X-Ways Software Technology AG

# *WinHex/ X-Ways Forensics*

*Integrated Computer Forensics Suite.*

*Data Recovery and IT Security Tool.*

*Editor Hexadecimal de Archivos, Discos y RAM*

Manual

# Índice


**1 Introducción ..... 1**

1.1 Acerca de WinHex y X-Ways Forensics ..... 1
1.2 Aviso Legal ..... 1
1.3 Licencias ..... 3
1.4 Differences between WinHex and X-Ways Forensics ..... 4

**2 Información General ..... 4**

2.1 Editores Hexadecimales ..... 4
2.2 Bytes Significativos ..... 5
2.3 Tipos de Datos Enteros ..... 6
2.4 Tipos de Datos Reales ..... 6
2.5 Tipos de Fecha ..... 7
2.6 ANSI-/IBM-ASCII ..... 8
2.7 Checksums ..... 9
2.8 Digests ..... 9
2.9 Consideraciones Técnicas ..... 10

**3 Forensic Features ..... 11**

3.1 Case Management ..... 11
3.2 Evidence Objects ..... 12
3.3 Log & Report Feature ..... 13
3.4 Navegador del Directorio ..... 14
3.5 Internal Viewer ..... 16
3.6 Registry Report ..... 18
3.7 Refined Volume Snapshots ..... 19
3.8 Mode Buttons ..... 20
3.9 Logical Search ..... 21
3.10 Hash Database ..... 22
3.11 Time Zone Concept ..... 23

**4 Trabajando con el Editor Hexadecimal ..... 24**

4.1 Inicio Rapido ..... 24
4.2 Introducción de Caracteres ..... 25
4.3 Modos de Edición ..... 25
4.4 Barra de Estado ..... 26
4.5 Scripts ..... 26
4.6 WinHex API ..... 27
4.7 Editor de Disco ..... 27
4.8 Editor de RAM ..... 29
4.9 Edición de Plantillas ..... 29
4.10 Consejos Útiles ..... 30

**5 Recuperación de Archivos ..... 31**

5.1 File Recovery with the Directory Browser ..... 31
5.2 File Recovery by *Name* ..... 31
5.3 File Recovery by *Type* ..... 32

5.4 File Type Definitions ..... 34
5.5 Manual Data Recovery ..... 35

**6 Referencia del Menú ..... 36**

6.1 Menú Archivo ..... 36
6.2 Menú Edición ..... 37
6.3 Menú Búsqueda ..... 38
6.4 Menú Posición ..... 39
6.5 Menú Ver ..... 40
6.6 Menú Herramientas ..... 41
6.7 Herramientas de Archivo ..... 43
6.8 Menú Especialista ..... 44
6.9 Menú Opciones ..... 48
6.10 Menú Ventanas ..... 48
6.11 Menú Ayuda ..... 49
6.12 Menú Contextual de Windows ..... 49
6.13 Directory Browser Context Menu ..... 50

**7 Opciones ..... 53**

7.1 Opciones Generales ..... 53
7.2 Directory Browser Options ..... 56
7.3 Opciones Deshacer ..... 58
7.4 Opciones de Seguridad ..... 58
7.5 Opciones de Búsqueda ..... 59
7.6 Opciones de Reemplazo ..... 61

**8 Miscelánea ..... 62**

8.1 Bloque ..... 62
8.2 Modicar Datos ..... 63
8.3 Conversiones ..... 64
8.4 Wiping and Initializing ..... 65
8.5 Clonar Disco ..... 66
8.6 Images and Backups ..... 67
8.7 Gestor de Backup ..... 69
8.8 Gestor de Posiciones ..... 69
8.9 Intérprete de Datos ..... 70

**Apéndice A: Definición de Plantillas ..... 71**

1 Cabecera ..... 71
2 Cuerpo: Declaración de Variables ..... 72
3 Cuerpo: Comandos Avanzados ..... 74
4 Cuerpo: Flexible Integer Variables ..... 75

**Apéndice B: Script Commands ..... 76**

**Apéndice C: Disk Editor Q&A ..... 83**

**Apéndice D: Registro Maestro de Arranque ..... 84**

**Apéndice E: Surplus Sectors ..... 85**

# 1 Introducción

## 1.1 Acerca de WinHex y X-Ways Forensics



X-Ways Software Technology AG
Stefan Fleischmann
Carl-Diem-Str. 32
32257 Bünde
Germany
Fax: +49 721-151 322 561

Web: http://www.x-ways.net
Product homepage: http://www.x-ways.net/winhex/
Ordering: http://www.x-ways.net/winhex/order.html
Support Forum: http://www.winhex.net

E-mail address: mail@x-ways.com

Registered in Bad Oeynhausen (HRB 7475). CEO: Stefan Fleischmann. Board of directors (chairwoman): Dr. M. Horstmeyer.

WinHex apareció por primera vez en 1995. Este manual ha sido compilado a partir de la ayuda en línea de WinHex v12.85, aparecido 3-2006.

La traducción al español es obra de José María Tagarro Martí.

Se soportan los siguientes sistemas operativos: Windows 2000, Windows XP (recommended); Windows 2003 Server

Varios Institutos Nacionales de los Estados Unidos (p.e. el Laboratorio Nacional de Oak Ridge en Tennessee), la Universidad Politécnica de Viena, la Universidad Politécnica de Munich (Instituto de Ciencias de la Computación, German Aerospace Center), Microsoft Corp., Hewlett Packard, Toshiba Europa, Siemens Business Services, Siemens VDO AG, Infineon Technologies Flash GmbH & Co. KG, Ericsson, National Semiconductor, Novell Inc., Ontrack Data International Inc., KPMG Forensic, Ernst & Young, Lockheed Martin, BAE Systems, TDK Corporation, Seoul Mobile Telecom, Visa International y muchas otras compañías e institutos científicos son ya usuarios registrados. Visite la página web de WinHex para obtener más información sobre cómo registrarse.

## 1.2 Aviso Legal

Esta publicación está diseñada para proporcionar información precisa y fidedigna acerca de las materias objeto de su utilización. En cualquier caso, el autor ni ofrece garantía alguna ni acepta ninguna responsabilidad con respecto al programa o al manual.

**License Agreement**

Your use, distribution, or installation of a software product published by X-Ways Software Technology AG indicates your acceptance of this license agreement. If you do not agree to any of the terms, then do not install, distribute or use the product.

A trial version may be only used for evaluation purposes. Purchasing one license authorizes you to install one copy of the full version of the software on a single machine. Additional licenses authorize you to install and use the full version on additional machines at the same time. Exception: *Forensic* licenses for WinHex/X-Ways Forensics do not impose an upper limit on the number of own computers with installations of the software, only on the number of concurrent uses on different computers.

This software, and all accompanying files, data, and materials, are distributed “as is” and with no warranties of any kind, whether express or implied, to the maximum extent permitted by applicable law. The user must assume the entire risk of using the program, knowing in particular that this software is not designed or intended for use in hazardous environments requiring fail-safe performance, where its failure to perform, misuse or inability to use adequately can reasonably be expected to lead to death, personal injury, or severe physical or environmental damage. In no event shall X-Ways Software Technology AG, or its officers, directors, employees, affiliates, contractors, or subsidiaries be liable for any direct, indirect, incidental, consequential, or punitive damages whatsoever arising out of the use or inability to use the software, to the maximum extent permitted by applicable law. Any liability will be limited exclusively to refund of purchase price by X-Ways Software Technology AG.

You may not rent, lease, modify, translate, reverse-engineer, decompile or disassemble the software or create derivative works based on it without prior explicit permission. All rights of any kind in the software product which are not expressly granted in this license agreement are entirely and exclusively reserved to and by X-Ways Software Technology AG.

No component of the software (except the WinHex API) must be accessed by other applications or processes.

Should any part of this agreement be or become invalid, such invalidity shall not affect the validity of the remaining provisions of the agreement.

**Acknowledgements**

Los algoritmos „Pukall Cipher 1“ (PC 1) y „Pukall Stream Cipher Hash Function“ tienen copyright de Alexandre Pukall. El código fuente está disponible en `http://www.multimania.com/pc1/`, `http://www.multimania.com/cuisinons/progs/` y en `http://www.freecode.com`.

The MD5 message digest is copyright by RSA Data Security Inc.

La librería de compresión „zlib“ tiene copyright de Jean-loup Gailly y Mark Adler. Página web: `ftp://ftp.cdrom.com/pub/infozip/zlib/zlib.html`

X-Ways Forensics contains software by Igor Pavlov, www.7-zip.com.

Outside In® Viewer Technology © 1991-2006 Stellent Chicago, Inc. All rights reserved.

Parts of the registry viewer are copyright by Markus Stephany, dumphive_[at]_mirkes_[dot]_de. All rights reserved. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Markus Stephany nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided by the copyright holders and contributors “as is” and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright owner or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damaged (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

## 1.3 Licencias

To use WinHex as a full version, you need a base license for personal, professional, or specialist use. If you are going to use WinHex on multiple machines, you will also need additional licenses. The full version will save files larger than 200 KB, write disk sectors, edit virtual memory and show no evaluation version reminders. It will reveal its license status on start-up and in the About box.

- Personal licenses are available at a reduced price únicamente para fines no comerciales, sin pertenecer a ninguna compañía, institución o gobierno.

- Professional licenses allow usage of the software in any environment (en su empresa, en una organización o en una administración pública). Professional licenses provide the ability to execute scripts and to use the WinHex API.

- Specialist licenses in addition to this allow to use the Specialist Tools menu section, to fully interpret Ext2, Ext3, CDFS/ISO9660, and UDF 1.0 media, and enable support for RAID and dynamic disks. Particularly useful for IT security specialists. Plus X-Ways Replica 1.3, a DOS-based forensically sound disk cloning and imaging software is included.

- Forensic licenses in addition to this allow to use the powerful case managing and report generating capabilities of WinHex, the internal viewer and additionally a powerful viewer component, the gallery view and advanced features of the drive contents table, comments in the directory browser, plus ReiserFS, Reiser4, HFS, HFS+, and UFS support. Furthermore,

they allow to read and write evidence files (.e01). Particularly useful for computer forensic examiners. The forensic edition of WinHex is called X-Ways Forensics. Also includes X-Ways Replica 2.35, with advanced disk cloning and imaging capabilities.

Please see the Order page of the web site (http://www.x-ways.net/winhex/order.html) on how to order your licenses.

## 1.4 Differences between WinHex and X-Ways Forensics

A forensic license offers numerous additional features over a specialist license. WinHex and X-Ways Forensics can be operated with the same forensic license. If so, they are identical, except for the following:

- WinHex (winhex.exe) always identifies itself as WinHex in the user interface, X-Ways Forensics (xwforensics.exe) as X-Ways Forensics. The program help and the manual, however, statically refer to “WinHex” in most cases.

- X-Ways Forensics only allows to open those files in an editable mode that are located on the drives that contain the current case, the general folder for temporary files, or the installation folder, for decoding/decryption/conversion purposes, etc. All other files, image files, virtual memory, and disks in general, are strictly opened in view mode (read-only), to enforce forensic procedures, where no evidence must be altered in the slightest. Similarly, only the above-mentioned drives are considered legitimate output folders where files can be saved. This strict write protection of X-Ways Forensics ensures that no original evidence can possibly be altered accidentally, which is a crucial aspect in court proceedings.

- Certain files (see http://www.x-ways.net/winhex/setup.html for details) are not part of the WinHex download, but owners of forensic licenses can copy them from X-Ways Forensics to enable the full feature set known from X-Ways Forensics in WinHex as well. Using WinHex instead of X-Ways Forensics can be desirable when not bound by strict forensic procedures and when in need to work more aggressively on files, disks, or images, e.g. repairing boot sectors etc., or when working with multiple clones where one clone is declared a working copy and cleared for write access.

# 2 Información General

## 2.1 Editores Hexadecimales

Un editor hexadecimal es capaz de mostrar completamete el contenido de cada tipo de archivo. A diferencia de un editor de texto, uno hexadecimal incluso muestra los códigos de control (p.e. los

caracteres de salto de línea y retorno) y el código ejecutable, usando un número de dos dígitos basado en el sistema de numeración hexadecimal.

Considere un byte como una secuencia de 8 bits. Cada bit puede ser un 0 o un 1, luego toma uno de dos estados posibles. Por lo tanto un byte puede tener $2•2•2•2•2•2•2•2 = 2^8 = 256$ valores diferentes. Como 256 es el cuadrado de 16, el valor de un byte puede estar definido por un número de dos dígitos en sistema hexadecimal, donde cada dígito representa un cuarteto (o nibble) de un byte, es decir, 4 bits. Los dieciséis dígitos utilizados en el sistema de numeración hexadecimal son 0-9, A-F.

Se puede cambiar el valor de un byte cambiando los dos dígitos del modo hexadecimal. También es posible introducir el carácter que tiene asignado un valor de byte dereminado en un juego de caracteres (véase Introducir Caracteres). Se permiten toda clase de caracteres (letras, signos de puntuación, etc.). Por ejemplo, un byte cuyo valor decimal es 65 aparece como 41 en notación hexadecimal (4•16+1=65) y tiene la letra A asignada en el modo de texto. Esto se debe a que el juego de caracteres ASCII asigna a la letra mayúscula A el valor decimal 65.

Cuando se editan archivos de un tipo determinado (por ejemplo archivos ejecutables), es esencial no alterar el *tamaño* del archivo. Modificar las direcciones del código ejecutable y los datos que están incluidos en ellos causa graves daños en tales archivos. Observe que cambiar el contenido de un archivo es con carácter general la causa de que la aplicación correspondiente se comporte de manera anormal. Aunque es bastante seguro editar los extractos de texto de un archivo, en cualquier caso es recomendable hacer archivos de backup (copias de seguridad) antes de hacerlo.

El comando „Búsqueda Combinada“ fue especialmente diseñado para editar archivos creados por juegos de ordenador para guardar la situación de las partidas. Si conoce el valor de una misma variable en dos de esos archivos, es posible encontrar el offset (la posición) en que ese dato ha sido grabado. Por ejemplo, si un archivo fue grabado cuando tenía 5 vidas y el otro cuando tenía 7, busque simultáneamente el valor hexadecimal 05 en el primero y 07 en el segundo archivo.

## 2.2 Bytes Significativos

Los microprocesadores difieren en la posición del byte menos significativo. Los procesadores Intel®, MIPS®, National Semiconductor y VAX lo colocan en primer lugar. Un valor multi-byte se almacena en la memoria desde el valor de byte más pequeño hasta el mayor. Por ejemplo, el valor hexadecimal 12345678 se guarda como 78 56 34 12. Este formato se conoce como *little-endian* (el pequeño al final).

Sin embargo, los procesadores Motorola et Sparc colocan el byte menos significativo en último lugar. Un valor multi-byte se almacena desde el valor mayor hasta el más pequeño. Por ejemplo, el valor hexadecimal 12345678 se guarda como 12 34 56 78. Este formato se conoce como *big-endian* (el grande al final).

## 2.3 Tipos de Datos Enteros

| Formato/Tipo | Rango | Ejemplo |
|---|---|---|
| signed 8 bit | -128...127 | FF = -1 |
| unsigned 8 bit | 0...255 | FF = 255 |
| signed 16 bit | -32,768...32,767 | 00 80 = -32,768 |
| unsigned 16 bit | 0...65,535 | 00 80 = 32,768 |
| signed 24 bit | -8,388,608...8,388,607 | 00 00 80 = -8,388,608 |
| unsigned 24 bit | 0...16,777,215 | 00 00 80 = 8,388,608 |
| signed 32 bit | -2,147,483,648...2,147,483,647 | 00 00 00 80 = -2,147,483,648 |
| unsigned 32 bit | 0...4,294,967,295 | 00 00 00 80 = 2,147,483,648 |
| signed 64 bit | $-2^{63}$ ($\approx -9 \cdot 10^{18}$)...$2^{63}-1$ ($\approx 9 \cdot 10^{18}$) | 00 00 00 00 00 00 00 80 = $-2^{63}$ |

Mientras no se indique lo contrario, los números multi-byte se almacenan en formato little-endian, significando por tanto que el primer byte de un número es el menos significativo y el último el más significativo. Este es el formato común de los ordenadores que ejecutan Microsoft Windows. De este modo, se da la paradoja de que los valores hexadecimales 10 27 se pueden interpretar como el número hexadecimal 2710 (decimal: 10.000).

El Intérprete de Datos es capaz de reconocer datos de todos los tipos enteros mencionados anteriormente.

## 2.4 Tipos de Datos Reales

| Tipo | Rango | Precisión [Dígitos] | Bytes |
|---|---|---|---|
| Float (Single) | $\pm 1.5^{-45}...3.4^{38}$ | 7-8 | 4 |
| Real | $\pm 2.9^{-39}...1.7^{38}$ | 11-12 | 6 |
| Double (Double) | $\pm 5.0^{-324}...1.7^{308}$ | 15-16 | 8 |
| Long Double (Extended) | $\pm 3.4^{-4932}...1.1^{4932}$ | 19-20 | 10 |

Los nombres de los tipos proceden del lenguage de programación C. Los nombres correspondientes en Pascal se especifican entre paréntesis. El tipo Real existe sólo en Pascal. El Intérprete de Datos es capaz de convertir valores hexadecimales de una ventana del editor en números reales de estos cuatro tipos y viceversa.

En el ordenador, un número real F se representa con una mantisa M y un exponente E, donde $M \times 2^E = F$. Tanto M como E son valores enteros con signo por si mismos. Los cuatro tipos de datos difieren en su rango de valores (es decir, el número de vits reservado para el exponente) y en su precisión (el número de bits reservados para la mantisa).

En los sistemas basados en Intel®, los cálculos con número reales se realizan con un coprocesador matemático, mientras el procesador principal espera. El Intel® 80x87 utiliza una

precisión de 80 bits para sus cálculos, mientras que los procesadores RISC a menudo disponen de 64 bits de prescisión.

## 2.5 Tipos de Fecha

El Intérprete de Datos soporta los siguientes formatos de fecha:

- **MS-DOS (4 bytes)**

Los dos primeros bytes almacenan la hora y los dos últimos la fecha. Este formato lo utilizan varias funciones de llamada DOS, el sistema de archivos FAT y muchas utilidades de sistema como los compresores de archivos, por ejemplo.

| Bits | Contenido |
|---|---|
| 0-4 | Segundo dividido por 2 |
| 5-10 | Minuto (0-59) |
| 11-15 | Hora (0-23 en un reloj de 24 horas) |
| 16-20 | Día del mes (1-31) |
| 21-24 | Mes (1 = Enero, 2 = Febrero, etc.) |
| 25-31 | Offset de año desde 1980 |

- **Win32 FILETIME (8 bytes)**

La estructura FILETIME es un entero de 64 bits que representa el número de intervalos de 100 nanosegundos transcurridos desde del 1 de enero de 1601. Este es el sistema usado por el API de Win32.

- **OLE 2.0 (8 bytes)**

Un valor real (más exactamente del tipo double) cuya parte entera determina el número de días transcurridos desde el el 30 de diciembre de 1899. La parte fraccionaria se interpreta como la duración de un día (luego ¼ serían las 6 de la mañana). Este es el formato de fecha del estándar OLE 2.0, usado entre otros programas por MS Excel.

- **ANSI SQL (8 bytes)**

Dos valores enteros consecutivos de 32 bits cada uno. El primero determina el número de días transcurridos desde el 17 de noviembre de 1858. El segundo es el número de intervalos de 100 microsegundos transcurridos desde medianoche. Este es el estándar ANSI SQL utilizado en multitud de bases de datos (por ejemplo InterBase 6.0).

- **UNIX, C, FORTRAN (4 bytes)**

Un valor entero de 32 bits determina el número de segundos transcurridos desde el 1 de enero de 1970. Este tipo de datos se utilizó en UNIX, en los programas C y C++ ("time_t") y en los

programas FORTRAN. En algunos sistemas se ha definido como el número de *minutos* transcurridos desde el 1 de enero de 1970. Las opciones del Intérprete de Datos permiten cambiar entre estas dos definiciones.

- **Macintosh HFS+ Date & Time (4 bytes)**

A 32-bit integer value that determines the number of seconds since January 1, 1904 GMT (HFS: local time). The maximum representable date is February 6, 2040 at 06:28:15 GMT. The date values do not account for leap seconds. They do include a leap day in every year that is evenly divisible by 4.

- **Java (8 bytes)**

A 64-bit integer value that specifies the number of milliseconds since January 1, 1970. Principally stored in big endian, which is the typical byte order in Java.

## 2.6 ANSI-/IBM-ASCII

ANSI-ASCII es un estándar del American National Standards Institute utilizado por las aplicaciones Windows. MS-DOS utiliza el juego de caracteres IBM-ASCII (también conocido como juego de caracteres OEM). Estos dos juegos de caracteres difieren en su segunda mitad, en los caracteres con valores ASCII mayores de 127.

Cuando quiera ver o editar un archivo procedente de un programa DOS, desactive la opción „Usar ANSI-ASCII“ del menú Opciones.

Utilice el comando „Convertir“ del menú Editar para convertir archivos de texto de un juego de caracteres a otro.

Los primeros 32 valores ASCII no definen caracteres imprimibles, sino códigos de control:

| Hex | Control Code | Hex | Control Code |
|---|---|---|---|
| 00 | Null | 10 | Data Link Escape |
| 01 | Start of Header | 11 | Device Control 1 |
| 02 | Start of Text | 12 | Device Control 2 |
| 03 | End of Text | 13 | Device Control 3 |
| 04 | End of Transmission | 14 | Device Control 4 |
| 05 | Enquiry | 15 | Negative Acknowledge |
| 06 | Acknowledge | 16 | Synchronous Idle |
| 07 | Bell | 17 | End of Transmission Block |
| 08 | Backspace | 18 | Cancel |
| 09 | Horizontal Tab | 19 | End of Medium |
| 0A | Line Feed | 1A | Substitute |
| 0B | Vertical Tab | 1B | Escape |

| | | | |
|---|---|---|---|
| 0C | Form Feed | 1C | File Separator |
| 0D | Carriage Return | 1D | Group Separator |
| 0E | Shift Out | 1E | Record Separator |
| 0F | Shift In | 1F | Unit Separator |

## 2.7 Checksums

Una checksum (suma de control) es un número característico utilizado para la verificación de la autenticidad de los datos. Dos archivos con un mismo checksum son con toda seguridad idénticos (byte a byte). Calcular y comparar el checksum de un archivo antes y después de una transmisión puede revelar errores durante el proceso. Un checksum coincidente indica que los archivos son (con toda probabilidad) idénticos. En cualquier caso, un archivo puede ser manipulado teniendo en cuenta que su checksum debe permanecer intacto. En estos casos, cuando se necesita detectar modificaciones malintencionadas (cuando no fruto del azar) se utilizan los llamados digests.

En WinHex, los checksum se calculan al abrir (opcional, véarse Opciones de Seguridad) o analizar (véase menú Herramientas) un archivo. Después de modificarlo, las sumas pueden ser recalculadas pulsando **ALT+F2**.

El checksum es simplemente la suma de todos los bytes de un archivo, calculada en un acumulador de 8 bits, 16 bits, 32 bits o 64 bits. El CRC (Código de Redundancia Cíclica) se basa en un algoritmo más sofisticado que es incluso más seguro que el estándar.

Por ejemplo: si una transmisión altera dos bytes de un archivo de tal manera que las modificaciones se compensan (byte uno +1, byte dos –1), el checksum permanece inalterado, pero el CRC cambia.

## 2.8 Digests

Los llamados digests son, al igual que los checksum, un número característico utilizado para la verificación de la autenticidad de datos. Pero los digest van aún más allá, ya que detectan cualquier cambio que haya ocurrido en los archivos.

Es factible manipular cualquier dato de manera que el checksum no cambie, y verificar su valor en tal caso conduciría a pensar que nuestros datos no han cambiado cuando en realidad si lo han hecho. Por lo tanto, los digest se usan en vez de los checksum cuando se buscan modificaciones malintencionadas (cuando no fruto del azar) de los datos originales. Es computacionalmente imposible encontrar dato alguno que corresponda a un digest dado. Incluso también es computacionalmente imposible encontrar dos fragmentos de datos a los que corresponda el mismo digest.

Por supuesto, las modificaciones aleatorias (como las causadas por una transmisión defectuosa)

también pueden ser detectadas cuando se usasn digests, pero los checksum son suficientes y ofrecen un servicio más eficiente para este propósito, ya que su cálculo es mucho más rápido.

WinHex incorporates the widely known 128-bit MD5 message digest, SHA-1, SHA-256, and PSCHF (Pukall Stream Cipher Hash Function).

## 2.9 Consideraciones Técnicas

- Especificaciones técnicas

| | |
|---|---|
| Cantidad de memoria adicional por cada macro: | 0.5 KB |
| Número máximo de ventanas: | 1000 (WinNT/2000), 500 (Win9x/Me) |
| Tamaño máximo de archivo o disco: | ≈2000 GB |
| Número máximo de instancias simultáneas del programa: | 99 |
| Número máximo de posiciones: | limited by RAM only |
| Número máximo de entradas de teclado reversibles: | 65535 |
| Complejidad de encriptación: | 128 bit |
| Longitud de los digest en los backups: | 128/256 bit |
| Juegos de caracteres soportados: | ANSI-/IBM-ASCII, EBCDIC, Unicode (limited) |
| Presentación del offset: | hexadecimal/decimal |

- En la mayoría de las ocasiones, una barra de progreso muestra el porcentaje completado de una operación. Sin embargo, durante las operaciones de búsqueda y reemplazo, indica la posición relativa en el archivo o disco actual.

- La interfaz de usuario tiene mejor aspecto si *no* se utiliza el tamaño de fuente extra grande en su sistema Windows.

- WinHex espera que su sistema esté funcionando en modo little-endian.

- Las claves que se especifican para operaciones de encriptación/desencriptación se guardan en el disco duro. En caso de que la opción de seguridad correspondiente esté activada, la clave se almacena encriptada en la RAM durante el tiempo que WinHex esté funcionando.

- Las operaciones de búsqueda y reemplazo generalmente funcionan más rápido si está activada la opción para distinguir mayúsculas de minúsculas y no se utilizan comodines.

- Cuando realice búsquedas con la opción „contar las ocurrencias“ activada o cuando reemplace sin confirmación, para un algoritmo de búsqueda existen generalmente dos modos de comportarse cuando encuentra una ocurrencia, lo que en muchos casos puede conducir a resultados diferentes. El siguiente ejemplo le ayudará a comprender:

  Las letras *ana* se buscan en la palabra „banana“. La primera ocurrencia se ha encontrado en el segundo carácter.

1ª alternativa: El algoritmo continúa la búsqueda desde el tercer carácter, de manera que *ana* se encuentra de nuevo en el cuarto carácter.

2ª alternativa: Las tres letras *ana* encontradas en la palabra „banana“ son descartadas. Las restantes letras *na* no contienen más ocurrencias de *ana*.

WinHex está programado de la segunda manera, ya que es la que conduce a resultados más razonables cuando se cuentan o reemplazan ocurrencias. De todas maneras, si continúa una búsqueda con la tecla **F3** o selecciona la opción „confirmar al encontrar“, el algoritmo se ajusta a la primera alternativa.

# 3 Forensic Features

## 3.1 Case Management

The integrated computer forensics environment in WinHex can be used with a forensic license of WinHex only. It offers complete case management, automated log and report file generation, and various additional features such as gallery view, category view, filename/file type mismatch detection, HPA detection, and skin color detection in pictures.

When starting up WinHex for the first time, you are asked whether to run it with the forensic interface. This means the “Case Data” window is displayed, WinHex is run in read-only mode, and you are asked to make sure the folders for temporary files and for case data are set correctly, in order to prevent WinHex from writing files to the wrong drive.

In order to work with a case, make sure the "Case Data" window is visible on the left of the main window. If not, enable View | Show | Case Data.

From the File menu, you may create a new case (start from scratch), open an existing case, close the active case, save the active case, back up the case file and the entire case folder in a ZIP archive, or automatically generate a case report. A case is stored in a .xfc file (xfc stands for X-Ways Forensics Case) and in a subfolder of the same name, just without the .xfc extension. This subfolder and its child folders are created automatically when the case is created. You may select the base folder for your cases in General Options. It is not necessary to explicitly save a case, unless you need to be sure it is saved at a given time. A case is saved automatically at latest when you close it or exit the program.

In the case properties window, you may name a case according to your own conventions (e.g. title or number). The date and time you create a case is recorded and displayed. The internal case filename is displayed as well. You may enter a description of the case (of arbitrary length) and the examiner's name, the examiner's organization's name and address. You may enable or disable the automated log feature for the whole case. Optionally, the evidence object subfolders in the case

folder are always suggested as default output folders for files recovered/copied off a file system. You may wish to disable that feature if your preference is to copy files from various evidence objects into the same output folder.

The most powerful concept in X-Ways Forensics, that allows to systematically and completely review files on computer media, is the so-called *refined volume snapshot*. It is possible to refine the standard volume snapshot for all evidence objects of a case in one step, and to search all evidence objects with volume snapshots logically with the help of the virtual global case root window. Note that it is possible to generate a flat overview of all existing and deleted files from all subdirectories on an partition or image file of a partition by recursively exploring the root directory. In order to explore a directory recursively (i.e. list its contents plus the contents of all its subdirectories plus their subdirectories), *right*-click the directory in the directory tree in the Case Data window. In order to *tag* a directory, you can click it with the middle mouse button in the directory tree.

In order to completely *delete* a case, you need to delete its .xfc file and the corresponding directory with the same name and all its subdirectories.

## 3.2 Evidence Objects

You may add any currently attached computer medium (such as hard disk, memory card, USB stick, CD-ROM, DVD, ...), any image file, or ordinary file to the active case. It will then be permanently associated with this case (unless you remove it from the case later), displayed in the tree-like case structure, and designated as an *evidence object*. A subfolder is created in the case folder for each evidence object, where by default files will be saved that you recover from that evidence object, so it will always be obvious from which object exactly (and from which case) recovered files originate.

In the evidence object properties window, you may enter a title or number for that evidence object according to your own conventions. The date and time it was associated with the active case is recorded and displayed. The internal designation of the evidence object is displayed as well as its original size in bytes. You may enter comments of arbitrary length that apply to the evidence objects, and a technical description of it is added by WinHex automatically (as known from the Medial Details Report command in the Specialist menu). You may have WinHex calculate a hash (checksum or digest) on the evidence object and verify it later, so that you can be sure that data authenticity has not been compromised in between. Hashes stored in evidence files are imported automatically when added to a case. You may disable the automated log feature for a specific evidence object if the log feature is enabled for the case as a whole.

Ways how to add files or media to a case: The "Add" commands in the case data window's File menu. The "Add" command in the edit window's tab's context menu. The "Add" command of a directory browser's item's context menu.

**Sub-elements**

All evidence objects in turn have further elements associated with them. There is a list of annotations/bookmarks, initially blank, where you may specially mark and comment an unlimited number of positions of interest, specifically for the evidence object. See Position Manager. Up to 32 contents tables can be associated with an evidence object. They are created by the Specialist menu commands Create Drive Contents Table and Create Directory Contents Table. They show the files of a volume including those in subdirectories in a single flat view, optionally grouped by file categories. From an evidence object's context menu you can also create special report contents tables, which are initially blank and to which you can add notable files via the directory browser's context menu (Position section). There is an item in the report table's context menu that allows you to toggle inclusion in the report.

Finally, if you extract free space, slack space, or text from a volume (using Specialist menu commands), the resulting files will show up in the case tree below the corresponding evidence object as well.

## 3.3 Log & Report Feature

**Logs**

When enabled in the case and the evidence properties window, WinHex obstinately logs all activities performed when the case is open. That allows you to easily track, reproduce, and document the steps you have followed to reach a certain result, for your own information and for the court room.

The following is recorded:

- when you a select a menu item, the command title (or at least an ID), and the name of the active edit window, if not an evidence object, preceded by the keyword "Menu",
- when a message box is displayed, the message text and what button you pressed (OK, Yes, No, or Cancel), preceded by the keyword "MsgBox",
- when a small progress indicator window is displayed, its title (like "Recovering files...") and whether the operation was completed or aborted, preceded by the keyword "Operation",
- a screenshot of each displayed dialog window with all selected options, e.g. for a complex operation that follows, preceded by the window's title,
- original source path of each recovered file,
- destination path of each recovered file when recovered with the directory browser or the Access button menu,
- the extensive log produced by Clone Disk and File Recovery by Type,
- your own entries (free text) that you add with the Add Log Entry command, either to the case as a whole or to a certain evidence object.

All actitivities are logged with their exact date and time, internally in FILETIME format with 100-nanosecond interval precision. Logs are by default associated with the case as a whole. However, logs of activities that apply to a certain evidence object are directly associated with that evidence object. This determines where they appear in a report. Screenshots are saved as .png

files in the "log" subfolder of a case folder. They can optionally be converted to black & white images, which allows to print them in a cost-effective way along with the report.

**Reports**

You may create a report from the File menu of the Case Data window. The report is saved as an HTML file and can thus be displayed and opened in a variety of applications. For example, you may view it in your favorite Internet browser and open and further process it in MS Word.

The report starts with the general case title and details, followed by a list of hyperlinks to the individual evidence object sections. For each evidence object, the report specifies its title, details, and description, your comments, your annotations, and the evidence object related log. The report ends with the general log.

## 3.4 Navegador del Directorio

On logical drives and partitions formatted with FAT12, FAT16, FAT32, NTFS, Ext2, Ext3, ReiserFS, Reiser4, HFS, HFS+, UFS, CDFS/ISO 9660/Joliet, or UDF, WinHex offers a *directory browser*, which resembles the Windows Explorer's right-hand list. It can be disabled or enabled by clicking the checkbox next to the Access button. The directory browser lists existing files and directories first, then deleted files and directories. Compressed files are displayed in blue, encrypted files in green (NTFS only). Right-clicking any item in the directory browser brings up a context menu with commands for opening a file or directory, exploring a directory, locating the beginning of a file or directory on the disk, locating the corresponding directory entry (FAT) or file record (NTFS), listing the allocated clusters in a separate window, and for easily recovering a lost or existing file or directory. The latter can recreate entire directory structures. Double-clicking executes the default action (locating the data in the Sectors view, listing clusters, and exploring in the case of a directory).

Deleted files and directories are represented in the directory browser with lighter icons. Question mark icons indicate that the original file or directory contents may be still available. Deleted objects that WinHex knows are no longer accessible (either because their first cluster has been reallocated or because they have a size of 0 bytes) have icons crossed out in red.

The directory browser can sort files and directories in ascending or descending order, and still reveals the previous sort criterion with a lighter arrow. For example, if you first click the filename column and then the filename extension column, files with the same extension will internally still be sorted by name.

When orphaned objects are found, e.g. files that have been deleted and whose original path is unknown, they are listed in a special fictitious directory “Path unknown”. With a specialist or forensic license, there are fictitious files in the root directory that allow you to conveniently address free space (unallocated clusters), volume slack, and file system areas (if applicable).

**Columns**

| | |
|---|---|
| Filename | Name of the listed file or directory. Allows to filter based on file masks. The filter expression may consist of several file masks, delimited by semicolons, like *.jpg;*.gif. Up to two asterisks allowed per mask if they are located at the beginning and the end of a filename mask. You may exclude files using file masks that start with a colon (:). For example, you may include all files except NTFS system files by providing the following masks: *;:$* (or simply :$*). Another example: All files with names that start with the letter "A", but do not contain the word "garden": A*;:*garden*. |
| Ext. | Filename extension. The part of the filename that follows the last dot, if any. |
| Type | If the header signature of a file was not specifically checked (see Refine Volume Snaphot), this is merely a repetition of the filename extension and displayed in gray. Otherwise, if the file signature verification revealed the true nature of the file, a typical extension of that type will be output. That extension will be displayed in black if it is still the same as the actual extension of the file, or in blue if the actual extension does not match the type of the file. (forensic license only) |
| Status | The status of the preceding Type column. Initially “not verified”. After verifying file types based on signatures or after previewing files: If a file is very small, the status is “don't care”. If neither the extension nor the signature is known to the file type signature database, the status is “not in list”. If the signature matches the extension according to the database, the status is “confirmed”. If the extension is referenced in the database, yet the signature is unknown, the status is “not confirmed”. If the signature matches a certain file type in the database, however the extension matches a different file type or there is no extension at all, the status is “newly identified”. (forensic license only) |
| Category | File type category corresponding to the file type, according to the definition in “File Type Categories.txt”. (forensic license only) |
| Path | Path of the file or directory, starting with a backward slash if the path is known, based on a volume's root, or starting with a question mark if the exact path is unknown. |
| Size | Size of the file or directory, without slack. |
| Created* | The date and time the file or directory was created on the volume it resides on. Not available on Linux filesystems. |
| Modified* | The date and time the file or directory was last modified on the volume it resides on. On FAT, time precision is 2-second intervals only. On CDFS, the only available date and time stamp is listed in this column altough it does not necessarily indicate last modification. |
| Accessed* | The date and time the file or directory was last read or otherwise accessed on the volume it resides on. On FAT, only the date is recorded. |
| Record update* | The date and time the file's or directory's FILE record (on NTFS) or inode (Linux filesystems) was last modified. These are filesystem data structures that contain the file's meta data. |

| | |
|---|---|
| Deletion* | The date and time the file or directory was deleted. Available on Linux filesystems only. |
| Attr. | DOS/Windows attributes on FAT/NTFS filesystems, Unix/Linux permissions on Ext2/Ext3/Reiser/HFS+ filesystems, plus some proprietary symbols. See below. |
| 1st cluster | The number of the cluster that contains the beginning file the file's or directory's data. Sorting by 1st cluster means to sort by physical location on the disk. |
| ID | The identifier assigned to the file or directory by the file system or by WinHex. Not necessarily unique. |
| Int. ID | The internal identifier of a file or directory in the volume snapshot. Items added to a volume snapshot last have the highest identifiers. |
| SC% | Skin color percentage. Optionally available for contents tables. Indicates the degree pictures are composed of skin tones. Sorting by this column is the most efficient way to discover traces of e.g. child pornography. |
| Hash | The file's hash value, if computed. |
| Hash set | In the internal hash database, the name of the hash set that the file's hash value, if available, belongs to. |
| Category | The category of the hash set that the file's hash value, if available, belongs to. Either "irrelevant", "notable", or blank. |

*Please note that for FAT volumes, all timestamps are displayed unmodified, for all other volumes the time zone concept applies.

**Attributes**

A = to be archived
R = read-only
H = hidden
S = system
P = junction point
C = compressed at file system level
c = compressed in an archive (ZIP, RAR, ...)
E = encrypted at file system level
e = encrypted in an archive (ZIP, RAR)
e? = possibly encrypted or compressed, according to the entropy test

The built-in priority when sorting by the Attr. column in descending order is as follows:

1) Document-level encryption
2) User-level encryption (e.g. in a ZIP archive)
3) NTFS filesystem encryption
4) User-level encryption supposed (flagged by the entropy test)
5) Unix/Linux SUID
6) Unix/Linux SGID

7) NTFS reparse/junction points
8) NTFS alternate data streams
9) File slack (listed separately only in evidence file containers)
10) User-level compression (e.g. in a ZIP archive)
11) NTFS filesystem compression
12) NTFS $EFS attributes
13) NTFS INDX/BTM attributes
14) HFS/HFS+ resource forks
15) Unix/Linux symlink
16) Unix/Linux other special file
17) ordinary DOS/Windows attributes and Linux permissions

## 3.5 Internal Viewer

The internal viewer can be invoked with the “View” command in the Tools menu and in the directory browser's context menu. It shows picture files of various file formats (see Gallery View) and the internal structure of Windows registry files. If you try to view a file that is not supported by the internal viewer, the first defined external viewer is invoked instead.

There is an additional viewer component that integrates seamlessly and allows to conveniently view more than 200 (!) file formats (such as MS Word/Excel/PowerPoint/Access/Works/Outlook, HTML, PDF, CorelDraw, StarOffice, OpenOffice, ...) directly in WinHex and X-Ways Forensics. This component is provided to all owners of forensic licenses issued for v12.05 and later. More information online.

**Registry Viewer**

MS Windows maintains an internal database called registry which contains all important settings for the local system and installed software in a tree-like structure. The data is persistently stored in files called registry hives. You can open and view hives without importing them into your own active registry. Supported formats are Win9x/Me/NT/2k/XP hives. Win9x and WinMe hives are located in the files "user.dat", "system.dat", and their backups. WinNT, Win2k, and WinXP hives are located in the file "ntuser.dat" in a user profile and in the directory \system32\config.

Up to 16 hives can be opened in the registry viewer at the same time. Since Win9x/Me and WinNT/2k/XP registries have different internal structures, their hives cannot be opened and viewed at the same time. If a different format is encountered, only the hive that was opened last will be displayed in the window.

With a right-click a popup menu can be opened anywhere in the window, which lets you invoke the commands "Search" and "Continue Search". Clicking "Search" pops up a dialog that lets you specify a search expression and where you want to search. You can browse either keys or names or values or all of them. The search starts at the topmost root and spans all opened hives. "Continue Search" finds the next match after at least one match has been found. (The currently selected element is not relevant for where the search continues). In the right-hand window the

popup menu also contains the command "Copy" which lets you copy the value of the selected element to the clipboard.

## 3.6 Registry Report

From within the registry viewer, WinHex can create a HTML-based report, listing values of possibly relevant registry keys, when you invoke the command "Create Registry Report" in the pop-up menu. The registry keys that are to be reported in all open hives are specified in a text file called "Reg Report Keys.txt". The registry files you view must have their original names, or else the report may fail. You may edit the list of registry keys in this files to tailor the report to your own needs.

**Format of entries in "Reg Report Keys.txt"**

(*operating system shortcut*) (*tabstop*) (*registry path*) (*tabstop*) (*description*) (*linefeed*)

*operating system shortcuts*:
9x: Windows 9x/Me
NT: Windows NT/2000/XP

*registry path*:
Full path of registry keys

HKLM: HKEY_LOCAL_MACHINE
HKCU: HKEY_CURRENT_USER

If an asterisk ("*") is provided as the last key, all keys on the same level and deeper and their values will be included in the report.

example:
NT HKLM\Software\Microsoft\Windows\CurrentVersion* report whole Windows branch

If you wish to report a particular value that exists in all subkeys of a certain key, you can as well write an "*" for all subkeys and include the value after that.

example:
9x HKCU\Identities*\UserID UserID value of every identity

The generated report contains the registry path with its timestamp (Windows NT/2000/XP only), the filename of the registry hive that the key was found in, the description that was provided in the "Reg Report Keys" file, and the value.

## 3.7 Refined Volume Snapshots

The Specialist menu allows to expand the *standard* volume snapshot (=the overview of files presented in the directory tree and in the directory browser) in various ways. Requires a specialist or forensic license. Refining volume snapshots serves as the same purpose and offers similar features as creating a drive contents table. Volume snapshots are superior to contents tables, though, since they can be examined both as a flat list (when exploring recursively) or directory-wise.

**Particularly thorough file system search**

- FAT12/FAT16/FAT32: This option searches for orphaned subdirectories (subdirectories that are no longer referenced by any other directory).
- NTFS: This option searches for file records in sectors that do not belong to the current MFT. Such file records can be found e.g. after a partition has been recreated, reformatted, moved, resized, or defragmented.
- ReiserFS, Reiser4: Searches for deleted files (which are not included in the standard volume snapshot at all).
- Other: no difference

Taking a *thorough* volume snapshot is possibly a lengthy operation, depending on the size of the volume, and for that reason this is not the standard procedure when opening volumes.

The "**File header search**" option causes files to be included in the list that can still be found in free or used drive space based on their file header signature and are no longer referenced by file system data structures. You are asked to select certain file types for detection, specify output filenames etc. as known from File Recovery by Type. Files found with this method will be included in the volume snapshot only if there is no other file in the volume snapshot with the same start cluster number or if they are not aligned at cluster boundaries, to avoid duplicates. Files found with this method are listed with a generic filename and size as detected by the File Recovery by Type mechanism.

**Hash values** can be computed for all files listed in a contents table. In addition to this, a forensic license allows to **match** the hash values against individually selected (or simply all) hash sets in the internal hash database. The filter can then later be used to hide known irrelevant files. Irrelevant files can optionally be hidden right away and excluded from further processing as part of volume snapshot refinement.

Only a forensic license allows to separately list and examine files in **ZIP, RAR**, ARJ, GZ, TAR, and BZIP **archives**, as long as the archives are not encrypted. The contents of archives in archives can be included as well, but no further level.

A forensic license allows you to **verify file types based on signatures**, i.e. detect filename/file type mismatches in files. For example, if someone has concealed an incriminating JPEG picture by naming it "invoice.xls" (wrong filename extension), the recognized file type "jpg" is stated in the Type column of the directory browser. For more information see the description of the columns Type and Status. The file signatures and extensions used for mismatch detection are

defined in the accompanying file type definition database, which you may fully customize. Please note that the link between the current data in unallocated clusters and deleted files and their filenames is weak, so false alerts might be displayed if a deleted file's clusters have been re-allocated to another file of a different type in the meantime.

A forensic license additionally allows to calculate the **percentage of skin colors in pictures**. This can be done for the same file types also supported by the gallery view, both for output to the directory browser and to a file. For example, if a forensic examiner is looking for traces of child pornography, sorting images by skin color percentage in category view may accelerate your work immensely because it renders checking the mass of 0%..9% skin color percentage pictures obsolete (e.g. thousands of browser cache garbage files). Please note that there may be false positives, i.e. skin-like colors of a non-skin surface. Pictures that cannot be correctly scanned for skin colors (e.g. too large, corrupt file or black-and-white) will be listed with a question mark instead of the skin color percentage.

A forensic license allows to optionally search for **JPEG and PNG pictures embedded in documents** such as MS Word, PDF, MS PowerPoint, MS Excel as well as in thumbs.db thumbnail buffers. Such pictures can be found by their file header signature and will be listed with generic names as "Embedded 1....jpg", "Embedded 2....png", etc. When including files in archives or pictures embedded in documents, the host files will be displayed as fictitious directories for convenient browsing.

A forensic license allows to optionally perform **file format specific and statistical encryption tests**. With an entropy test, each existing file is checked whether it is encrypted or compressed. If the test is positive (the entropy exceeds a certain threshold), the file is flagged with “e?” in the attribute column, to indicate that it might deserve special attention. Typical example: Encrypted container files, which can be mounted by encryption programs like PGP Desktop, BestCrypt, or DriveCrypt as drive letters. The entropy test is not applied to ZIP, RAR, TAR, GZ, BZ, 7Z, ARJ, JPG, PNG, GIF, TIF, MP3 and MPG files, which are well-known to be compressed internally. This test is not needed to detect that files are encrypted at the NTFS file system level or inside archives. Secondly, documents with the extensions .doc (MS Word 4...2003), .xls (MS Excel 2...2003), .ppt, .pps (MS PowerPoint 97-2003), .mpp (MS Project 98-2003), and .pdf (Adobe Acrobat) are checked for file format specific encryption. If positive, these files are flagged with “e!” in the attribute column. This check requires that the separate viewer component is active.

## 3.8 Mode Buttons

When examining a logical drive, partition, or image file with a file system supported by WinHex, there are four buttons that determine the display in the lower half of the window, below the directory browser.

**Sectors**

The default view that shows the binary data in all sectors as hexadecimal code, ASCII text, or both, along with an offset column.

**Preview**

Checks the file signature of the file currently selected in the directory browser. If found to be a picture (supported file types see below), the picture is displayed, otherwise an ASCII text extract from the beginning of the file. The result of the signature check (whether it matches the filename extension or not) is displayed in the status bar. By double-clicking the preview, you get a full-size view of a picture, where you may zoom in and out using the keys + and -. Even incomplete pictures (e.g. files incompletely recovered because of fragmention) can usually be displayed partially.

**Gallery**

Checks the file signature of all the files in the currently visible portion of the directory browser. If found to be a picture, a thumbnail is displayed, otherwise a brief summary (filename, size, signature). By scrolling in the directory browser, the gallery view scrolls as well. You may switch the directory even while the thumbnails are still loading. By double-clicking a thumbnail, you get a full-size view of a picture, where you may zoom in and out using the keys + and -. Even incomplete pictures (e.g. file incompletely recovered because of fragmention) can usually be displayed partially.

Supported picture file types: BMP, JPG/JPEG, JPEG 2000, PNG, GIF, TIF, TGA, PCX, WMF, EMF, MNG, JBG

**Calendar** (timeline view)

Gives a convenient overview of when the files/directories selected in the directory browser were created in a file system (red), last modified (blue), and last accessed (green), in the form of a calendar. Each day with a time stamp for at least one file or directory is filled in the calendar with the corresponding color. Weekends (Saturdays and Sundays) are specially marked. Hover the mouse over a day to find out which files exactly are represented and to see the corresponding times. If the list for a certain day is too lengthy to be displayed completely, you can still sort the directory browser in a suitable way and find out there.

Example: During which period of time were JPEG files created on a volume? Either right-click the root directory in the directory tree (case data window) to recursively list all existing files or create a contents table, then sort by filename extension, select all JPEG files, then enable the calendar view, and watch out for red bars.

## 3.9 Logical Search

The directory browser's context menu allows logical simultaneous search operations in files and folders that are selected in the directory browser (specialist or forensic license only). The Logical Search command is not available in a recursive view if directories are included in that recursive

view and among the selected items. That's because in a recursive view XWF cannot branch into a subdirectory and return to the same recursive view.

Advantages:

+ The search scope can be limited to certain files and folders, also certain files and folders that are part of a contents table.

+ Searching in files (usually = in the cluster chains allocated to files) as a "logical" kind of search will find search term occurrences even if the search term happens to be physically split in a fragmented file (occurs at the end and the beginning of discontiguous clusters) and even if the file is compressed at the NTFS file system level and optionally even if it is part of an archive (ZIP, RAR, GZ, TAR, BZ2, 7Z, and ARJ, if not encrypted, forensic license only). Files in archives are searched only if either 1) they have been specifically included in the volume snapshot and they are part of a recursive view in the directory browser or 2) the corresponding checkbox is checked in a non-recursive view.

+ The text contained in PDF (Adobe), WPD (Corel WordPerfect), CDR (Corel Draw), and VSD (Visio) files can automatically be extracted and decoded prior to search, such that their plaintext will be searched as well. Potential search hits in such files would otherwise be missed because these file types typically store text in an encoded, encrypted or otherwise garbled way. This feature requires the separate viewer component to be active for the decoding and text extraction part.

+ Files in which the searm term occurs can be automatically opened or added to a dedicated contents table.

Unallocated space can be included in a logical search by including the fictitious file "Free Space" in the root directory, file system areas by including the fictitious file of the same name. Slack space is included depending on the directory browser options.

- Only a physical search can cover the transition from slack space to directly following free space.

## 3.10 Hash Database

Only available with a forensic license. The internal hash database, once created, consists of 257 binary files with the extension .xhd (X-Ways Hash Database). The storage folder is selected in the General Options dialog. The hash database is organized in a very efficient way, which maximizes performance when matching hash values. It is up to you to decide, around what hash type the database is built (MD5, SHA-1, SHA-256, ...).

Each hash value in the hash database belongs to one or more hash sets. Each hash set belongs to either the category “known good”/“harmless”/“irrelevant” or “known bad”/“malicious”/“relevant”/“notable”.

Hash values of files can be calculated and matched against the hash database when creating a contents table. The directory browser's optional columns “Hash Set” and "Category” will then reveal for each file to which hash set and category it belongs, if any (which allows you to sort by these aspects and filter out irrelevant files easily). Please note that if a hash value is contained in multiple hash sets, only the first matching hash set will be displayed in the hash set column.

The Tools menu allows you to

- manage the active hash database: create a new (empty) one, view the list of hash sets, rename and delete hash sets, toggle the hash set category, and verify the integrity of the hash database (F8)
- import a single hash set text file (NSRL RDS 2.x, HashKeeper, and ILook text files are supported)
- import all the hash set text files in a certain folder and all its subfolders (ditto), optionally into a single internal hash set whose name you have to specify
- delete the active hash database, e.g. to start a fresh one with new hash sets and/or a new hash type.

The Create Hash Set command in the directory browser's context menu allows you to create your own hash sets in the internal hash database. The hash database supports up to 65,535 hash sets. Future versions will allow you to export hash sets in the hash database to the NSRL RDS format.

## 3.11 Time Zone Concept

The following applies to WinHex and X-Ways Forensics when operated with a specialist or forensic license.

Since v12.8, X-Ways Forensics no longer employs Windows' logic for converting UTC to local filetimes and displays timestamps independently of the time zone selected in the examiner's system's Control Panel. When working with a case, the time zone selected for that case applies globally to the entire program (selectable in the Case Properties), otherwise the one selected in the General Options dialog. When working with a case, optionally it is possible to specify different time zones per evidence object, so that you can always see local filetimes even for media that were used in different time zones, if preferable. Note that the timestamps are converted for display only. That means, in a recursive view in the case root that covers multiple media, sorting is based on absolute UTC timestamps. Optionally, the actually used conversion bias can be displayed as well (see directory browser options).

Timestamps on FAT volumes are never converted as they are not available in UTC, but in one or several unknown local time zones.

Report tables and export lists are output in local time, for your convenience, and internally converted back to UTC when read. They remember based on which time zone they were output, so importing them yields correct timestamps even if you had changed the time zone for that evidence object or the entire case in the meantime. Report tables created by earlier versions of X-Ways Forensics, on the other hand, are imported assuming they were created in the time zone

selected in the examiner's system's Control Panel, which is consistent behavior.

The time zone definitions can be adjusted, if necessary. Please note that changing these definitions in any dialog window affects the definition of time zones throughout the program.

WinHex and X-Ways Forensics still employs the standard Windows conversion technique, which depends on the time zone selected in the user's system's Control Panel:

- in File | Properties, where the timestamps of files on the user's own system can be accessed/changed,
- for its case logging features,
- generally when operated without a specialist or forensic license, and
- when operated without the file “timezone.dat”.

You can tell that either of the two latter is true if the time zone button in the General Options dialog is not available or visible.

# 4 Trabajando con el Editor Hexadecimal

## 4.1 Inicio Rapido

The so-called Start Center is a dialog window that is optionally displayed at startup and is meant as a simplified control panel for beginning your work. It allows to quickly open files, disks, memory modules, and folders as well as up to 255 recently edited documents (16 by default, left-hand list). These may be files, folders, logical drives or physical disks. When opened again, WinHex restores the last cursor position, the scrolling position, and the block (if defined) of each document, unless the corresponding option is disabled.

From the Start Center you are also able to access *projects* and *cases* (right-hand top list). A project consists of one or more documents to edit (files or disks). It remembers the editing positions, the window sizes and positions and some display options. By saving a window arrangement as a project you can continue to work in several documents right where you left them, with a single click only. This is especially useful for recurring tasks. When you load a project, all currently opened windows are automatically closed first.

Besides, WinHex automatically saves the window arrangement from the end of a WinHex session as a project, and can re-create it next time at startup. Each project is stored in a .prj file. It can be deleted or renamed right within the Start Center (context menu or DELETE/F2 key).

Last not least, the Start Center is the place where to manage *scripts*. You may check, edit, create, rename, and delete scripts using the context menu. To execute a script, double-click it or single-click it and click the OK button.

## 4.2 Introducción de Caracteres

En modo hexadecimal sólo pueden introducirse caracteres hexadecimales (‘0‘...‘9‘, ‘A‘...‘F‘). En modo texto se puede introducir cualquier clase carácter: letras, números, signos de puntuación y caracteres especiales (p.e. ‘«‘, ‘]‘, ‘^‘). Utilice el programa de Windows charmap.exe para encontrar combinaciones de teclas para dichos caracteres (p.e. Alt-1-8-7 para ‘»‘). La fuente de letra „WinHex“ incluso soporta el símbolo del Euro (€).

## 4.3 Modos de Edición

The details panel displays for each file/disk, in which mode it was opened. The details panel's context menu allows to selectively change the edit mode of the active window.

**Modo de sólo lectura:** Recommended for computer forensic examinations. In order to enforce strict forensic procedures, the only mode available in X-Ways Forensics, except for files in the current case's directory and in the general folder for temporary files, to allow to decode, decrypt, and convert them, etc. Los archivos o discos abiertos en este modo no pueden ser editados, sólo examinados. En otras palabras, están protegidos contra escritura.

**Modo de edición por defecto:** Las modificaciones de los archivos o discos abiertos en este modo se guardan en archivos temporales. Estos son creados dinámicamente. El comando „Guardar“ del menú Archivo actualiza el archivo original o el disco.

**Modo de edición directa:** Sea cuidadoso cuando abra archivos o discos en este modo. Todas las modificaciones (entradas desde teclado, llenar/borrar bloques, pegar el portapapeles, reemplazos, etc.) se escriben directamente en el *archivo original sin confirmación.* No es necesario guardar el archivo manualmente despues de haberlo modificado. En vez de eso, los cambios se guardan constante y automáticamente (por última vez al cerrar la ventana). De cualquier modo, puede utilizar el comando Guardar para asegurarse de que el buffer es vaciado en un momento determinado.

El modo de edición directa es recomendable en aquellos casos en que la transferencia de datos entre el archivo temporal y el original (y viceversa) consuma demasiado tiempo o espacio en disco. Este puede ser el caso al abrir muchos archivos grandes o cuando edite enormes cantidades de datos. Como no se encesitan archivos temporales en este modo de edición, generalmente es más rápido que el modo por defecto. El modo de edición directa es el único disponible cuando se utiliza el editor de RAM.

Incluso en el modo de edición directa la creación de un archivo temporal es inevitable cuando se altera el tamaño de un archivo.

## 4.4 Barra de Estado

La barra de estado muestra la siguiente información acerca de un archivo:

1. Número de página actual y número de páginas totales (editor de disco: sectores)
2. Posición actual (offset)
3. Conversión decimal de los valores hexadecimales de la posición actual
4. Principio y final del bloque actual (si está definido)
5. Tamaño del bloque actual en bytes

Pulse en las celdas de la barra de estado para...
1. Moverse a otra página/sector,
2. Moverse a otro offset,
3. Definir el tipo de entero para la conversión decimal y
4. Definir el bloque.

Pulse con el botón derecho en la barra de estado para copiar la parte de información que desee desde allí al portapapeles.

Pulsar con el botón derecho el segundo campo de la barra de estado permite cambiar entre presentación absoluta (por defecto) o relativa del offset. Esto es útil cuando examine datos que consistan en registros de una longitud prefijada. Después de especificar dicha longitud en bytes, la barra de estado muestra el número de registro actual y el offset interno del mismo.

Pulsar con el botón derecho el tercer campo de la barra de estado permite copiar los cuatro valores hexadecimales de la posición actual al portapapeles, pero en orden inverso. Esto es de utilidad para hacer un seguimiento de punteros.

## 4.5 Scripts

Most of the functionality of WinHex can be used in an automated way, e.g. to speed up recurring routine tasks or to perform certain tasks on unattended remote computers. The ability to execute scripts other than the supplied sample scripts is limited to owners of a professional license or higher. Scripts can be run from the Start Center or the command line. While a script is executed, you may press Esc to abort. Because of their superior possibilities, scripts supersede routines, which were the only method of automation in previous versions of WinHex.

WinHex scripts are text files with the filename extension ".whs". They can be edited using any text editor and simply consist of a sequence of commands. It is recommended to enter one command per line only for reasons of visual clarity. Depending on the command, you may need to specify parameters next to a command. Most commands affect the file or disk presented in the currently active window.

Vea Apéndice B for a description of currently supported script commands.

## 4.6 WinHex API

The WinHex API (application programming interface) allows to use the advanced capabilities of the WinHex Hex Editor programmatically from your own C++, Delphi, or Visual Basic programs. In particular, it provides a convenient and simple interface for random access to files and disks.

Developing software that uses the WinHex API requires a valid *professional* or *specialist* WinHex license. Additionally, you need import declarations for your programming language of choice, the library file "whxapi.dll", and the API documentation. Please find those files and more detailed information online at http://www.winhex.com/winhex/api/.

You may also *distribute* both any software that makes use of the WinHex API and WinHex itself. There are two ways how to distribute WinHex:

1. Distribute the unlicensed WinHex version. For the API to work, your customer has to purchase professional or specialist licenses according to the number of WinHex installations needed.
-or-
2. Recommended: distribute a special API version of WinHex that is configured to only provide the API functionality and that is available at a reduced price. You may place your order online at http://www.winhex.com/winhex/api/. Volume discount available on request (please specify the number of licenses you are interested in). One WinHex API license needed per end user computer. The product will be licensed to you, you will be the actual owner of the licenses, but any of your customers may use them. The end user does not have to take care of anything related to WinHex.

## 4.7 Editor de Disco

El editor de disco, que forma parte del menú Herramientas, permite acceder a disquetes y discos duros por debajo del nivel del sistema de archivos. Los discos consisten de sectores (por lo común unidades de 512 bytes). Se puede acceder al disco de manera lógica (controlada por el sistema operativo) o física (controlada por la BIOS). En la mayoría de los ordenadores se puede acceder incluso a unidades CD-ROM y DVD. There is an optional raw mode for optical drives that allows to read from audio CDs and also the complete 2352-byte sectors on data CDs (CD-ROM and Video CDs) that contain error correction codes.

Opening a *logical drive* means opening a contiguous formatted part of a disk (a partition) that is accessible under Windows as a drive letter. It's also called a "volume". WinHex relies on Windows being able to access the drive. Opening a *physical disk* means opening the entire medium, as it is attached to the computer, e.g. a hard disk including *all* partitions. It could also called the "raw device". The disk normally does not need to be properly formatted in order to

open it that way.

Usually it is preferable to open a logical drive instead of a physical disk, because more features are provided in this case. For example, “clusters” are defined by the file system, the allocation of clusters to files (and vice versa) is known to WinHex, “free space” and “slack space” have a meaning. Only if you need to edit sectors outside a logical drive (e.g. the master boot record), if you wish to search something on several partitions of a hard disk at the same time, or if a partition is damaged or formatted with a file system unknown to Windows, so Windows is unable to make it accessible as a drive letter, you would open the physical disk instead. Via the menu that appears when clicking the “Access” button, you may also open individual partitions from within a physical disk. WinHex understands both conventional MBR partitioning and Windows 2000's dynamic disks as organized by the LDM (Logical Disk Manager, specialist and forensic licenses only). All dynamic volume types are supported: simple, spanned, striped, and RAID 5. Holding the Ctrl key when opening hard disks disables detection and special handling of dynamic volumes and ensures the hard disk is treated like it has been partitioned in the conventional way.

Tenga en cuenta las siguientes limitaciones:

- Bajo Windows NT/2000/XP, se necesitan privilegios de administrador para acceder directamente a los discos duros.
- Bajo Windows 9x, se deben cumplir ciertos requisitos para porder acceder a unidades CD-ROM y DVD (vea Apéndice C).
- Las funciones de reemplazo no están disponibles.
- Los sectores de CD-ROM y DVD no se pueden grabar.
- El editor de disco no opera en unidades remotas (de red).

El apéndice E de este manual proporciona algunas especificaciones del registro maestro de arranque, que puede ser modificado con el editor de disco.

Editar el espacio dispobible en disco (Windows 95/98/Me)

Bajo Windows 95/98/Me es posible editar el espacio no utilizado de una unidad lógica. Las limitaciones mencionadas anteriormente no se aplican en este caso, ya que WinHex crea un archivo que usa la totalidad del espacio disponible en la unidad seleccionada. De este modo, es posible editar este archivo en el modo de edición directa ya que la integridad de los datos en las partes utilizadas del disco no puede verse afectada de ninguna manera.

Puede utilizar esta función para recuperar datos borrados accidentalmente mientras no hayan sido sobreescritos por archivos nuevos. Busque los datos, márquelos como el bloque actual y copielos en un archivo nuevo. Por supuesto, los datos que han sido eliminados mediante el comando Borrado Irreversible de WinHex no pueden encontrarse en las partes no usadas del disco.

**Guardar Sectores:** Su utilización es análoga al comando Guardar del menú Archivo. Escribe todas las modificaciones realizadas en el disco. Tenga en cuenta que, dependiendo de los cambios que haya realizado, esto puede dañar gravemente la integridad de los datos del disco. Si la opción de deshacer correspondiente está activada, se creará un backup de los sectores afectados antes de que sean sobreescritos.

Este comando no puede aplicarse a discos hasta que no se registre.

## 4.8 Editor de RAM

El editor de RAM le permite examinar the physical RAM/main memory (under Windows 2000/XP, with administrator rights only) y la memoria virtual de un proceso (es decir, un programa que está siendo ejecutado). Todas las páginas de memoria afectadas por este proceso se presentan en un bloque continuo. Las páginas no utilizadas (libres o reservadas) son by default, but optionally included and displayed with "?" characters. With no gaps, you may compare memory dumps to files exactly with one another (absolute and virtual addresses are identical), e.g. to examine stack and heap states or observe virusses.

Seleccione uno de los procesos listados. Entonces podrá acceder bien a la llamada memoria primeria o a la memoria completa, o a uno de los módulos cargados. La memoria primaria es la utilizada por los programas para casi todas sus necesidades. Generalmente contiene el módulo principal del proceso (el archivo EXE) y la pila. La „memoria completa" contiene toda la memoria virtual de un proceso incluyendo la parte de memoria que está compartida con todos los procesos excepto los módulos del sistema. Bajo Windows 95/98/Me los módulos del sistema pueden listarse opcionalmente en el arbol de procesos. Los módulos de sistemas están definidos como aquellos que son cargados más allá de la barrera de 2 GB (como kernel32.dll, gdi32.dll). Estos módulos son compartidos por todos los procesos que se están ejecutando.

Tenga en cuenta las siguientes limitaciones:

- Cuidado: sólo puede deshacer la entrada desde teclado.
- La memoria virtual de los procesos de 16 bits sólo es accesible en parte bajo Windows 95/98/Me.
- La edición sólo es posible en el modo directo.
- Los módulos de sistema de Windows 95/98/Me sólo pueden ser *examinados* en modo de sólo lectura, y no *manipulados.*

Las opciones relevantes del editor RAM son „Comprobar alteraciones de la memoria virtual" y „Direcciones Virtuales".

## 4.9 Edición de Plantillas

Una plantilla es un cuadro de diálogo que proporciona un sentido a la edición de estructuras de datos específicas en un modo más cómodo y seguro que el obtenido con la edición hexadecimal directa. La edición se realiza en dos cuadros separados. Los cambios tienen lugar cuando pulsamos la tecla **ENTER** o cuando abandonamos la plantilla (después de pedir confirmación). Los datos pueden proceder de un archivo, de sectores del disco o de la memoria virtual. Especialmente al editar bases de datos, es posible que prefiera definir una plantilla a su medida para acceder fácilmente a los registros. You will find the command to print a template in the system menu.

Una *definición de plantilla* se almacena en un archivo de texto. El *editor de plantillas* le permite escribir definiciones de plantilla y le ofrece comprobación de sintaxis integrada. Una definición de plantilla contiene principalmente declaraciones de variables, que son similares a aquellas que aparecen en el código fuente de los lenguages de programación. La sintaxis se explica con detalle en el Apéndice A. Los tipos de datos soportados incluyen todos los enteros más comunes, los reales y las variables booleanas, los tipos de fecha, valores hexadecimales, binarios, caracteres y cadenas de texto. Las listas tanto de variables individuales como de grupos de variables también pueden ser utilizadas.

La capacidad de moverse libremente hacia delante y hacia atrás en los datos hace el uso de plantillas especialmente flexible

- La misma variable puede ser interpretada y manipulada en muchas maneras.
- Las secciones con datos irrelevantes pueden ser ignoradas.

El *gestor de plantillas* muestra todos los archivos de texto del directorio de WinHex que contienen definiciones de plantillas. Se muestran el título de la plantilla junto con la descripción, el nombre del archivo y la fecha y hora de la última modificación. Pulse el botón Aplicar para mostrar una plantilla que utilice la definición seleccionada para los datos existentes en la ventana actual del editor y en la posición actual. Puede asimismo crear una nueva definición de plantilla, borrar o editar una existente.

WinHex incorpora diferenes ejemplos de plantillas.

## 4.10 Consejos Útiles

- Si selecciona un bloque con el ratón, puede pulsar dos veces el botón derecho para anularlo.
- Puede definir un bloque de datos utilizando el teclado (**SHIFT**+arrow keys o **ALT**+**1** y **ALT**+**2**).
- Use la tecla **TAB** para cambiar del modo hexadecimal al de texto y viceversa.
- Use la tecla **INS** para cambiar del modo overwrite al de insert y viceversa.
- **CTRL**+**Q** cierra todas las ventanas.
- **ENTER** muestra el Start Center.
- **ESC** aborta la operación en curso si la hay, en caso contrario anula la selección de bloque, dismisses an active dialog or template window.
- **PAUSE** detiene o continúa la operación en curso.
- **F11** repite el último comando Ir A Offset. **CTRL**+F11 works in the opposite direction (from the current position).
- **ALT**++ is a variant of the Go To Offset command specifically to jump a certain number of sectors *down*.
- **ALT**+- is another variant specifically to jump a certain number of sectors *up*.
- **SHIFT**+**F7** alterna entre los juegos de caracteres.
- (**SHIFT**+)**ALT**+**F11** repite el último comando Mover Bloque.
- **CTRL**+**SHIFT**+**M** invokes an open evidence object's annotations.
- **ALT**+**F2** recalcula el auto-hash despues de que un fichero haya sido modificado.

- **ALT**+**LEFT** and **ALT**+**RIGHT** allow for switching between records within a template (just as the "<" and ">" buttons). **ALT**+**HOME** and **ALT**+**END** access the first and the last record, respectively.
- **ALT**+**G** moves the cursor in the edit window to the current template position and closes the template window.
- **CTRL**+**F9** opens the Access button menu (disk edit windows only).
- WinHex acepta tanto nombres de archivo de línea de comandos como „arrastrar y soltar“.
- Use las scripts para hacer su trabajo con WinHex más eficiente.
- Puede indicar the name de una script como un parámetro de línea de comandos.
- "Invalid input": After dismissing this error message box, the blinking cursor indicates what parameter provided by you is invalid and needs to be corrected.
- Cambie de la presentación de offset hexadecimal a decimal pulsando sobre los números.
- Pruebe a pulsar en las celdas de la barra de estado (botones izquierdo y derecho del ratón).

# 5 Recuperación de Archivos

## 5.1 File Recovery with the Directory Browser

Most obviously, deleted files and directories that are listed in the directory browser can be recovered easily and selectively with the directory browser’s context menu. See chapter “directory browser”.

## 5.2 File Recovery by *Name*

This is a very easy to use mass file recovery function, part of the the Disk Tools menu. Requires that you have opened a logical drive or a single partition of a physical disk with the disk editor. Works on FAT12, FAT16, FAT32, and NTFS drives. You may specify one or more filename patterns that cover all the files you wish to retrieve, e.g.:

Letter to Mr. Smith.doc
Invoice*.pdf
m*.xls
Image*.gif
*.tif

You may exclude files using filename patterns that begin with a colon (:). For example, you may include all files except NTFS system (which always start with a dollar sign) files by providing the following patterns:

*
:$*

Please note that files that were moved to the recycle bin prior to permanent deletion are internally

renamed by Windows, where only the filename extension remains the same, so using wildcards will often come handy (e.g. *.jpg instead of abc.jpg). Unlike File Recovery by Type, this function will also restore the file date & time and its attributes.

Optionally this function recovers/copies only those files that currently exist in the file system (from a user's pointer of view) or that are considered non-existent (deleted or otherwise lost).

Alternatively to using the file allocation table of a FAT drive, WinHex can optionally also rely on files not being fragmented, recovering them as a continuous stream of consecutive clusters.

Check "Intercept invalid filenames" to prevent a failure of the recovery because of filenames with characters considered as invalid by the file system. Useful for example if you wish to recover files that had filenames in a non-western language with a western-language Windows version. This option will rename such a file if necessary to ensure that it can be recreated.

On an NTFS drive, if the file you are looking for cannot be found, it may help to enable the "thorough" search. It is not enabled by default because it takes significantly more time.

You must also specify an output folder where to recreate the original file(s). Important: make sure this folder is on a different drive. Specifying a folder on the same drive where you are recovering from could easily overwrite disk space where deleted files reside that you still wish to recover! That way they would be lost forever. It might also lead to a loop, if WinHex repeatedly "recovers" files that it has just recreated.

## 5.3 File Recovery by *Type*

Another data recovery function in the Disk Tools menu. This recovery method is also referred to as "file carving". It searches for files that can be recognized by a characteristic file header signature (a certain sequence of byte values). Because of this approach, File Recovery by Type does not depend on the existence of functional file system structures. When found based on the signature, the files are saved to the output folder that is specified by the user. Optionally, recovered files of each type are put into their own subfolder (...\JPEG, ...\HTML, etc.). Note that File Recovery by Type assumes contiguous file clusters, so produces corrupt files in case the files were originally stored in a fragmented way. A log file "File Recovery by Type.log" about the selected parameters and the recovery results is written to the output folder for verification purposes.

Since no use is made of a possible presence of a (functional or non-functional) file system, the original *file sizes* are principally *unknown* to this algorithm, and so are the original *filenames*. That is why the resulting files are named according to the following pattern: Prefix[X]id0000.ext. "Prefix" is an optional prefix you provide. "id" is a unique character combination that identifies an entry in the file type definition database (aa = 1st entry, ab = 2nd entry, ...). "0000" is an incrementing number per file type. "ext" is the filename extension that corresponds to the file header signature according to the file definition database. WinHex can often detect if recovered JPEG, GIF, and files of some other types, are corrupt or incomplete (caused e.g. by file

fragmentation). If this is the case, it will mark these files as corrupt in the log file and insert a capital X in the output filename. If the user-supplied file size limit is found to be too small for certain files, this will be noted in the log file as well. The output filename prefix may optionally contain a placeholder *%d*, which will be replaced by the drive name. This is useful if you apply File Recovery by Type to multiple drives at a time and wish to be able to easily distinguish files from different drives even without checking the log file.

The algorithm tries to determine the original size of JPEG, GIF, PNG, BMP, TIFF, CDR, AVI, WAV, ZIP, MS Word, MS Excel, MS PowerPoint, RTF, and HTML files by examining their data structure, roughly limited by the user-supplied maximum size. The corresponding entries in the file type definition database must not be altered in order for the size and type detection to work for these file types. For other types, the files are recovered at the exact size specified by the user as the maximum (in KB). Be generous when specifying this size because whereas files recovered "too large" can still be opened by their associated application, truncated files often can't be and are obviously incomplete.

Technically it is possible to select as many file types for simultaneous recovery as you like. However, if you e.g. recover MS Office and AVI files at the same time and the MS Office files you expect are around a few KB and the AVI files around a 1 GB in size, using a single global maximum file size would not be a good idea. That's why optionally you can define an individual default size for each file type in the file type definition database.

By default, file headers are only searched at *cluster* boundaries because the beginning of a cluster is the only place where a file can start in a cluster-based file system. However, you may also select to search for sector-aligned file headers. This is useful to find files from a previously existing volume with a different cluster layout. If performed on a physical medium or raw file with no cluster layout defined, WinHex searches at sector boundaries anyway if cluster boundaries are selected. There is yet another possibility, a thorough byte-level search. This is necessary when recovering files from backup files or tapes, or JPEG files from within MS Word documents, where they are not aligned at cluster or sector boundaries. This comes at the cost of a possibly increased number of false positives, though, misidentified file signatures occurring randomly on a media, not indicating the beginning of a file.

You may limit the scope of the recovery to a currently selected block if necessary and/or to allocated or unallocated space (option available on a logical drive or volume). E.g. in order to recover files that were deleted, you select to recover from unallocated space only. Files that are not accessible any more because of file system errors may still be stored in clusters that are considered as in use.

The option "Ext2/Ext3 block logic" causes this recovery method to deviate from the standard assumption of no fragmention in that it will follow the typical Ext block pattern, where e.g. the 13th block from the header of the file is considered an indirect block that references the following data blocks. This option has no effect when applied to partitions that WinHex knows have a file system other than Ext2 and Ext3 or when a header is found that is not block-aligned.

Specialist and forensic licenses only: If you enable the option “No actual recovery, just list found

files”, no files are actually output and no log file is written. Files are only listed in the directory browser and output to a contents table, e.g. for inspection with the gallery viewer and for selective recovery from within the directory browser.

## 5.4 File Type Definitions

"File Type Signatures.txt" is a tab-delimited text file that serves as a file type definition database for contents tables and for the File Recovery by Type command.

WinHex comes with various preset file type signatures. You may fully customize the file type definitions and add your own ones, either in "File Type Signatures.txt" itself or you create additional such files of the same format named "File Type Signatures *.txt", which will be loaded as well and have the benefit that they will not be overwritten when you install the next update. Up to 255 entries are supported altogether.

When you click the Customize button to edit the file "File Type Signatures.txt", by default WinHex opens the file in MS Excel. This is convenient because the file consists of columns separated by tabs. If you edit the file with a text editor, be sure to retain these tabs, as WinHex relies on their presence to properly interpret the file type definitions. MS Excel retains them automatically. After editing the file type definitions, you need to exit the dialog window and invoke the File Recovery by Type or Create Drive Contents Table menu command again to see the changes in the file type list.

**1st column: File Type**

A human-readable designation of the file type, e.g. "JPEG". Everything beyond the first 19 characters is ignored.

**2nd column: Extensions**

One or more file type extensions typically used for this file type. E.g. "jpg;jpeg;jpe". Specify the most common extension first because that one will be used by default for naming recovered files. Everything beyond the first 45 characters is ignored.

**3rd column: Header**

A unique header signature by which files of this file type can be recognized. May be specified in either ASCII or hex (e.g. 0xFFD8FF means the bytes 0xFF 0xD8 0xFF). Header signatures up to 16 bytes in size are supported. To find out characteristic file header signatures in the first place, open several existing files of a certain type in WinHex and look for common byte values near the beginning of the file at identical offsets.

**4th column: Offset**

The relative offset within a file at which the signature occurs. Often simply 0.

**5th column: Footer**

Optional. A signature (constant byte sequence) that reliably indicates the end of a file. May be specified in either ASCII or hex. A footer signature may help to force a recovery with the correct file size. Still, the recovery algorithm does not search for the footer further than the number of bytes specified as the maximum file size, starting from the header. Footer signatures up to 8 bytes in size are supported.

**6th column: Default in KB**

Optional. A file type specific default maximum file size in KB that can override the global maximum file size specified in the File Recovery by Type dialog window. Useful because e.g. an MPEG video could be more around 1 GB in size, where a Windows icon file (.ico) could be around 1 KB in size.

## 5.5 Manual Data Recovery

It is possible to restore lost or logically deleted files (or more general: data) that are merely marked as deleted in the file system, but have not been *physically* erased (or overwritten).

Open the logical drive where the deleted file resided on using the disk editor. Principally you can recreate such a file by selecting the disk sectors, that were allocated to the file, as the current block and saving them using the menu command Edit | Copy Block | Into New File. But it may prove difficult to *find* the sectors where the file is still stored. There are two general ways to accomplish this:

1. In case you know a snippet of the file you are looking for (e.g. the characteristic signature in the header of a JPEG file or the words “Dear Mr. Smith” in a MS Word document), search it on the disk using the common search commands (“Find Text” or “Find Hex Values”). This is a very simple and safe way, and can be recommended to anyone.

2. In case you only know the filename, you will need some knowledge about the filesystem on the disk (FAT16, FAT32, NTFS, ...) to find traces of former directory entries of the file and thereby determine the number of the first cluster that was allocated to the file. Detailed information on file systems is available on the WinHex web site. The following applies to all FAT variants:

   If the directory that *contained* the file (let's call that directory “D”) still exists, you can find D on the disk using Tools | Disk Tools | List Directory Clusters. The factory template for FAT directory entries that comes with WinHex will then be helpful to find out the number of the first cluster that was allocated to the deleted file in that directory. Otherwise, if D has been deleted as well, you need to find the contents of D (using the directory entry template) starting with the directory that contained D.

Deleted files and directories are marked with the character “å” (hexadecimal: E5) as the first letter in their name.

You may encounter the problem that the file to recover is fragmented, i. e. not stored in subsequent contiguous clusters. On FAT drives, the next cluster of a file can be looked up in the file allocation table at the beginning of the drive (simple templates to do this can be found on the web site), but this information is erased when a file is deleted.

# 6 Referencia del Menú

## 6.1 Menú Archivo

**Nuevo:** Sste comando se utiliza para crear un archivo. El archivo en abierto (en principio) en el modo de edición por defecto. Debe especificar el tamaño deseado para el archivo.

**Abrir:** Le permite abrir uno o más archivos. Puede elegir un modo de edición en caso de que no esté predeterminado en el menú Opciones.

**Guardar:** Guarda el archivo actualmente mostrado en el disco. En el modo de edición directa el uso de este comando no es necesario. Cuando utilice el editor de disco este comando se llama „Guardar Sectores“.

**Guardar Como:** Guarda el archivo actualmente mostrado bajo un nombre.

**Crear Backup:** Véase „Backups“

**Cargar Backup:** Seleccione un image o archivo de backup (archivo WHX) cuyo contenido (un archivo o sectores del disco) quiera restaurar.

**Gestor de Backup:** Véase „Backups“

**Ejecutar:** Ejecuta el archivo actual si es ejecutable, o en caso contrario el programa asociado.

**Imprimir:** Use este comando para imprimir un archivo, sectores del disco o el contenido de la RAM. Defina el rango de impresión mediante offsets. Puede seleccionar y configurar una impresora, así como el juego de caracteres para la impresión. Además puede elegir el tamaño de fuente que desee o bien aceptar el recomendado por WinHex. Este tamaño se calcula como sigue: resolución de impresión (por ejemplo 720 dpi) / 6 (120 en este ejemplo). Si lo desea puede introducir un comentario que será impreso al final.
En caso de que necesite más flexibilidad a la hora de imprimir, puede definir un bloque y copiarlo utilizando „Edición->Copiar->Pantalla del Editor“ como texto con el mismo formato hexadecimal de la pantalla del editor. Puede más tarde pegar el contenido del portapapeles en su procesador de textos preferido. Debería tener un aspecto perfecto con el tipo de letra „Courier

New“ y un tamaño de 10 pt.

**Propiedades:** Le permite editar el tamaño, la fecha y los atributos de un archivo (bajo Windows NT también de un directortio). Atributos válidos son: A (archivo), S (sistema), H (oculto), R (sólo lectura). Después de introducir nuevos valores en cualquiera de los apartados (tamaño, fecha o atributos), pulse la **ENTER** para aplicar las modificaciones.

**Abrir Carpeta:** Este comando se utiliza para abir varios archivos que cumplen unos determinados requisitos en un momento dado. Seleccione la carpeta en la que va a abir los archivos y opcionalmente busque en los subdirectorios. Puede especificar una máscara de archivo (por ejemplo „w*.exe;x*.dll“). También hay una opción que permiten abir sólo aquellos archivos que contengan un cierto texto o unos ciertos valores hexadecimales. Para este propósito aparecerán los mismos cuadros de diálogos del comando de búsqueda. Si WinHex no está configurado para editar los archivos en un modo determinado, también puede seleccionarlo ahora.

**Guardar Ficheros Modificados:** Todos los archivos que hayan sido modificados son escritos en el disco.

**Guardar Todos los Ficheros:** Todos los archivos que no hayan sido abiertos en modo de sólo lectura son escritos en el disco.

**Salir:** Use este comando para finalizar WinHex. Se le preguntará si desee guardar las modificaciones realizadas en archivos o discos.

## 6.2 Menú Edición

**Deshacer:** Invierte la última modificación realizada, en caso de que la correspondiente opción deshacer esté activada.

**Cortar:** Borra el bloque actual del archivo y lo pone en el portapapeles. Los datos que sigan al bloque subirán hasta el donde estaba el principio del bloque.

**Copiar Bloque/Todo/Sector:**

- **Normal:** Copia el actual bloque/archivo entero/sector en el portapapeles. El contenido del mismo puede más tarde pegarse o guardarse.
- **En un Archivo Nuevo:** Copia los datos directamente en un nuevo archivo (no a través del portapapeles). Por ejemplo, este comando puede utilizarse para recuperar un archivo perdido a partir de los sectores del disco.
- **Valores Hexadecimales:** Copia los datos como una serie de valores hexadecimales concatenados.
- **Pantalla del Editor:** Copia los datos como texto, formateado como si estuviese en la pantalla del editor hexadecimal, es decir, con una columna de offset, una hexadecimal y otra de texto.
- **Código Fuente en C/Pascal:** Copia los datos como código fuente en formato C/Pascal en el portapapeles.

**Pegar Portapapeles:** Inserta el contenido del portapapeles en la posición actual de un archivo. Los datos que sigan a esta posición serán desplazados hacia deltante.

**Escribir Portapapeles:** Copia el contenido del portapapeles en la posición y archivo actuales. Los datos en esta posición son sobreescritos. Si el archivo termina antes de pegar todo el contenido, el tamaño se incrementa hasta donde sea necesario para completar la operación

**Pegar Portapapeles en un Archivo Nuevo:** Crea un nuevo archivo con el contenido del portapapeles.

**Vaciar portapapeles:** Este comando se utiliza para liberar la memoria utilizada por el portapapeles.

**Borrar:** Borra el bloque actual del archivo. Los datos que sigan al bloque subirán hasta donde estaba el inicio del bloque. El portapapeles no se ve afectado por este comando. Si el bloque está definido de la misma manera en todos los archivos abiertos (es decir, empieza y termina en los mismos offsets), el comando puede ser incluso aplicado a todos los archivos abiertos a la vez.

**Pegar Bytes Cero:** Use este comando para insertar bytes cero en la posición actual de un archivo.

**Definir Bloque:** Esta función es accesible desde el menú y desde la barra de estado. Un cuadro de diálogo le permitirá especificar los limites deseados para el bloque. El comando también puede ser aplicado a todos los archivos

**Seleccionar Todo:** Define el principio y el final del archivo actual como los límites del bloque.

**Convertir:** Véase Conversiones

**Modicar Datos:** véase más adelante

**Rellanar Bloque/Archivos/Sectores de Disco:** véase más adelante (Wiping and Initializing)

## 6.3 Menú Búsqueda

**Encontrar Texto:** Este comando se usa para buscar una cadena de texto especificada de hasta 50 caracteres en el archivo actual, disco o sección de RAM (véase Opciones de Búsqueda). Specialist and forensic licenses only: identical to Simultaneous Search, unless Shift key is pressed.

**Encontrar Valores Hexadecimales:** Este comando se usa para buscar una secuencia de hasta 50 valores hexadecimales de dos caracteres (véase Opciones de Búsqueda).

**Reemplazar Texto:** Use este comando para reemplazar ocurrencias de una cadena especificada

con otra cadena (cada una de hasta 50 caracteres). Véase Opciones de Reemplazo.

**Reemplazar Valores Hexadecimales:** Funciona exactamente igual que el comando Reemplazar Texto, pero se aplica a una secuencia de valores hexadecimales (50 como máximo). Véase Opciones de Reemplazo.

**Búsqueda Combinada:** Proporciona un complejo mecanismo de búsqueda. En el archivo actual y en un segundo archivo se busca un mismo offset, donde cada archivo contiene los respectivos valores hexadecimales especificados

**Valor Entero:** Introduzca un entero (con los límites del tipo signed 64-bit). Esta función busca datos en el archivo actual que puedan interpretarse como este entero.

**Valor Real:** Introduzca un número real (por ejemplo, 12,34 = 0,1234 * $10^2$ = 0,1234E2) y seleccione un tipo de datos real. Esta funciónbusca datos en el archivo actual que puedan ser interpretados como este valor real.

**Extractos de Texto:** Use este comando para buscar una secuencia de letras (a-z, A-Z), dígitos (0-9) y/o signos de puntuación. Es útil cuando, por ejemplo, se desea traducir extractos de texto ocultos en alguna parte de un archivo con código ejecutable.
Indique la precisión de la búsqueda especificando cómo de larga debe ser una secuencia de caracteres para ser reconocida. Pulse „Aceptar juego de caracteres Unicode“ para obligar al algoritmo a aceptar bytes cero entre dos caracteres.

**Continuar Búsqueda Global:** Este comando se usa para continuar una operación de búsqueda global (por ejemplo, una operación de búsqueda aplicada a todos los archivos abiertos) en el siguiente archivo.

**Continuar Búsqueda:** Le permite continuar una operación de búsqueda en el archivo y posición actuales.

## 6.4 Menú Posición

**Ir A Offset:** Mueve la posición actual hasta el offset especificado. Normalmente esto se hace con relación al principio del archivo (offset 0). También puede mover el cursor con relación a la posición actual (hacia delante o hacia atrás) o con respecto al final del archivo (hacia atrás). Un offset puede especificarse en bytes (por defecto), words (2 bytes), doublewords (4 bytes), records (if defined), or sectors. Pulse **F11** para repetir el movimiento de la última posición.

**Ir A Página/Sector:** Mueva la posición actual hasta la página o sector especificado. Please note that the data area on FAT drives starts with cluster #2.

**Go To FAT Entry/FILE Record:** Jump to a certain entry in the file allocation table on a FAT drive or to a certain FILE record in the master file table on an NTFS drive, respectively.

**Mover Bloque:** Mueve la *selección* del bloque actual (no los datos del bloque) hacia delante o hacia atrás. Especifique la distancia en bytes. Pulse **ALT+F11** para repetir el último movimiento de bloque y pulse **SHIFT+ALT+F11** para invertir el movimiento. Este comando puede facilitar la edición de un archivo que consista en registros homogéneos de una longitud fija.

WinHex keeps a history of your offset jumps within a document and allows to go **back** and **foward** in the chain later.

**Ir A...**

**Inicio del Archivo:** Muestra la primera página del archivo actual y mueva la posción actual al offset 0.

**Final del Archivo:** Muestra la última página del archivo actual y mueve la posición actual hasta el último byte (offset = tamaño de archivo – 1).

**Inicio de Bloque:** Mueve la posición actual al principio del bloque actual.

**Final de Bloque:** Mueve la posición actual hasta el final del bloque actual.

**Marcar Posición:** Marca la posición actual y por lo tanto permite que más tarde pueda encontrarla.

**Borrar Marcador:** Borra el marcador de la pantalla.

**Ir A Marcador:** Mueve la posición actual hasta el marcador definido por el comando Marcar Posición.

**Gestor de Posiciones:** véa más abajo.

## 6.5 Menú Ver

**Sólo Pantalla de Texto**: Oculta la pantalla de datos hexadecimal y usa todo el ancho de la ventana del editor para mostrar la pantalla de texto ASCII.

**Sólo Pantalla Hexadecimal**: Oculta la pantalla de texto ASCII y usa todo el ancho de la ventana del editor para mostrar la pantalla hexadecimal.

**Ver de Registro:** When editing subsequent data records of the same size (for instance, table entries of a database) you may now have WinHex display every other record with a different background color, as a kind of visual aid. The color can be selected in the General Options dialog. Also, WinHex offers to display the current record number and the offset within that record (relative offset) in the status bar, based the record size and the offset of the first record as

specified. Esto es útil cuando examine datos que consistan en registros de una longitud prefijada. Después de especificar dicha longitud en bytes, la barra de estado muestra el número de registro actual y el offset interno del mismo.
If any of the two record features is enabled, the Go To Offset command allows moving the current position in units of the current record size.

**Mostrar**: The Case Data window is part of the forensic user interface of WinHex (X-Ways Forensics). The **directory browser** is available for logical drives/partitions opened with the disk editor. Cluster lists can optionally be displayed for any file or folder that you double-click in the directory browser. El **Intérprete de Datos** es una pequeña ventana que proporciona "servicios de traducción" para los datos localizados en la posición actual del cursor. La **barra de herramientas** puede mostrarse opcionalmente. El **control tab** hace cada ventana de edición accesible mediante un simple clic de ratón. La **sección de detalles** proporciona información en profundidad sobre cualquier objeto (archivo, disco, RAM).

**Gestor de Plantillas**

**Tablas**: Proporciona cuatro tablas de conversión (véase ANSI-/IBM-ASCII).

**Líneas & Columnas**

**Sincronice Desplazamiento**: Synchronizes up to four tiled windows on identical absolute offsets. Hold the Shift key when enabling this feature to tile the windows horizontally instead of vertically.

**Sincronice y Comparar**: Synchronizes up to four windows and visually displays byte value differences. If no more than two windows are involved, WinHex maintains the initial distance between the offsets of the first shown byte in these windows when scrolling. Not synchronizing on absolute offsets is useful for example when comparing two copies of the file allocation table, which are obviously at different offsets. You may skip to the next or to the previous byte value difference by clicking the extra buttons that are provided in one of the two edit windows.

**Restaurar**: Redraws the contents of the current edit window. In case the current file was updated by an external program, WinHex offers to dismiss any changes made in WinHex and reload the file from scratch.

## 6.6 Menú Herramientas

**Abrir Disco:** Vea el capítulo „Editor de Disco“.

**Clonar Disco:** Vea el capítulo „Clonar Disco“.

**Recuperte Archivos:** Vea Apéndice D.

**Take New Volume Snapshot:** Disponible para partitions with one of the supported file systems. WinHex recorre todas las cadenas de clústeres y por lo tanto genera un mapa de la unidad. Esto permite a WinHex to fill the directory browser y especificar la ubicación de cada sector o clúster (archivo, directorio, FAT, libre). Se recomienda invocar este comando otra vez después de operaciones de archivo en disco para actualizar la informaci´n mostrada por WinHex. Véase Opciones de Seguridad.

**Inicializar de Espacio Libre:** Puede que información confidencial se encuentre almacenada en el espacio libre del disco como consecuencia de una operación normal de borrado o copia. El espacio libre en el disco puede ser inicializado (por razones de seguridad). Esta operación borra de manera irreversible todos los datos de las partes no utilizadas del disco y hace imposible recuperarlos. On NTFS drives, WinHex will also offer to wipe all currently unused $Mft (Master File Table) file records, as they may still contain names and fragments of files previously stored in them.

**Initialize Slack Space:** Overwrites slack space (the unused bytes in the respective last clusters of all cluster chains, beyond the actual end of a file) with zero bytes. This may be used in addition to "Initialize Free Space" to securely wipe confidential data on a drive or to minimize the space a compressed disk backup (like a WinHex backup) requires. Close any running or resident program that may write to the disk prior to using this command.

**Initialize MFT Records:** On NTFS drives, WinHex can clear all currently unused $Mft (Master File Table) file records, as they may still contain names and fragments of files previously stored in them.

**Scan For Lost Partitions:** Formerly existing hard disk partitions that were not automatically found when opening a physical hard disk and are not listed in the Access button menu may be found and properly identified with this command. This command searches for a master boot record and boot sector signature (0x55AA), optionally only from the first sector that follows the last (location-wise) partition that was already found, and lists newly found partitions in the Access button menu.

**Interpret as Partition Start:** When you find the start sector of a volume (e.g. lost partition) on a physical disk, this menu command allows you to make such a partition easily accessible via the Access button menu. If no known file system is detected starting at the currently displayed sector, you will be asked for the number of sectors that you wish to include in the newly defined partition.

**Parámetros del Disco:** Using this command on a physical disk, you may override the number of cylinders, heads, and sectors per track as recognized by WinHex. This can be useful to access surplus sectors at the end of the disk (in case they were not detected by WinHex), or to adjust the CHS coordinate system to your needs. Use this command on a logical drive to override the total number of clusters WinHex detects on that drive. This can prove useful when examining huge DVDs, which are detected as 2 GB media under Windows 9x.

**Abrir RAM:** Vea el capítulo „Editor de RAM“.

**View:** Available only with a forensic license. Invokes the internal viewer.

**External Viewer:** Invokes external file viewing programs such as Quick View Plus etc., as selected in the Options menu, and opens the current file.

**Invocar X-Ways Trace:** Available only if X-Ways Trace is installed. This software can analyze the Internet Explorer's index.dat history file and the Windows recycle bin's info2 files.

**Calculadora:** Ejecuta la calculadora de Windows „calc.exe“. Cambiarla al modo científico es una opción altamente recomendable.

**Conversor Hexadecimal:** Le permite convertir número hexadecimales en números decimales y viceversa. Simplemente escriba el número y pulse **ENTER**.

**Tablas:** Proporciona cuatro tablas de conversión (véase ANSI-/IBM-ASCII).

**Analizar Bloque/Archivo:** Examina los datos del bloque/archivo actual y cuenta las ocurrencias de cada valor de byte (0...255). El resultado se muestra gráficamente con líneas verticales proporcionales. El número de ocurrencias y el porcentaje para cada valor de byte se muestran cuando se mueve el ratón sobre la correspondiente línea vertical.
Puede utilizar este comando, por ejemplo, para identificar datos de un tipo desconocido. Datos de audio, datos comprimidos, código ejecutable, etc., producen gráficas características. Please note that this feature is not intended for use with gigabytes of data. El checksum estándar de 32 bit y el CRC32 de los datos seleccionados también se muestran.
Use el menú de contexte de la ventana para activar o desactivar la consideración de los bytes cero, to print the analysis window, or to export the analysis to a text file.

**Calcular Hash:** Calculates one of the following checksums/digest of the entire current file, disks, or the currently selected block: 8-bit, 16-bit, 32-bit, 64-bit checksum, CRC16, CRC32, MD5, SHA-1, SHA-256, or PSCHF.

## 6.7 Herramientas de Archivo

**Concatenar:** Seleccione varios archivos de origen que serán copiados en un único archivo de destino. Los archivos de origen no se verán afectados.

**Partir:** Este comando crea múltiples archivos de destino utilizando el contenido de un único archivo de origen. Especifique un tamaño para cada archivo de destino. El archivo de origen no se ve afectado por esta operación.

**Unir:** Seleccione dos archivos de origen y un archivo de destino. Los bytes/words de los archivos de origen se escribirán alternativamente en el archivo de destino. El primer byte/word del destino será el del primer archivo de origen especificado. Utilice esta función para crear un archivo con bytes/word pares e impares procedentes de archivos diferentes (por ejemplo en la programación

de EPROM).

**Diseccionar:** Seleccione un archivo de origen y dos archivos de destino. Los bytes/words del archivo de origen se escribirán alternativamente en los archivos de destino. El primer byte/word del origen se copiará en el primer archivo de destino especificado. Utilice esta función para crear dos archivos separados conteniendo cada uno de ellos bien los bytes/words pares bien los impares del archivo original (por ejemplo en la programación de EPROM).

**Comparar:** Este comando se utiliza para comparar dos edit windows (archivos o discos) byte a byte. Decida si quiere ser informado de las coincidencias o de las diferencias entre ellos. You may indicate how many bytes to compare. Si lo desea, la operación terminará después de haber encontrado un cierto número de diferencias o de bytes idénticos. El informe resultante se guardará como un archivo de texto, pero tenga en cuenta que su tamaño puede aumentar considerablemente.
The comparison starts at the respective offsets specified for each edit window. These offsets may differ, such that e.g. the byte at offset 0 in file A is compared to the byte at offset 32 in file B, the byte at offset 1 with the one at offset 33, etc. When you select an edit window for comparison, the current position will automatically be entered in the "From offset" box.
There is yet another compare function in WinHex: you may also compare edit windows visually and synchronize scrolling in these windows (see View menu).

**Borrado Irreversible:** Este comando se utiliza para borrar el contenido de un archivo de manera irreversible, de manera que no pueda ser recuperado por medio de programas específicos. Cada archivo seleccionado se rellenará according to the current settings, después se cambiará su longitud a cero y por último se borrará. La entrada del nombre de archivo también se borrará. Incluso los intentos de recuperación profesionales serán inútiles. Lógicamente, este comando debe ser aplicado a aquellos archivos cuyo contenido sea confidencial, cuando deban ser destruidos. *Available in WinHex only, not in X-Ways Forensics.*

## 6.8 Menú Especialista

*Specialist and forensic licenses only.*

**Refine Volume Snapshot:** see separate chapter

**Create Drive Contents Table:** see separate chapter

**Simultaneous Search:** A parallel search facility, that lets you specify a virtually unlimited list of search terms, one per line (physical search). The search terms are either text strings or hex values (specified with a 0x prefix). They are searched simultaneously, and their occurrences can be archived in the Position Manager. WinHex will save the offset of each occurrence, the search term, the name of the file or disk searched, and in the case of a logical drive the cluster allocation as well (i.e. the name and path of the file that is stored at that particular offset, if any).
That means e.g. a forensic examiner is now able to systematically search through multiple hard drives and disk images in a single pass for words like

- drug
- cocaine
- (street synonym #1 for cocaine)
- (street synonym #2 for cocaine)
- (street synonym #3 for cocaine)
- (street synonym #3 for cocaine, alternative spelling)
- (name of dealer #1)
- (name of dealer #2)
- (name of dealer #3)

at the same time. When searching a logical drive, this will narrow down the examination to a list of files upon which to focus. If you do not have WinHex archive the occurrences, you may use the F3 key to continue the search.

**Create Directory Contents Table:** Works like Create Drive Contents Table, but for a user-selected directory and its subdirectories only. You will find this command only in the directory browser's context menu, when right-clicking a directory.

**Media Details Report:** Shows information about the currently active disk or file and lets you copy it e.g. into a report you are writing. Most extensive on physical hard disks, where details for each partition and even unallocated gaps between existing partitions are pointed out. Under Windows 2000 and XP, WinHex also reports the password protection status of ATA disks.
Forensic license only: WinHex is able to detect hidden host-protected areas (HPAs, a.k.a. ATA-protected areas) and device configuration overlays (DCO areas) on IDE hard disks up under Windows 2000 and XP. A message box with a warning will be displayed in case the disk size has been artificially reduced. At any rate, the real total number of sectors according to ATA, if it can be determined, is listed in the details report.

**Interpret Image File As Disk:** Treats a currently open and active disk image file as either a logical drive or physical disk. This is useful if you wish to closely examine the file system structure of a disk image, extract files, etc. without copying it back to a disk. If interpreted as a physical disk, WinHex can access and open the partitions contained in the image individually as known from “real” physical hard disks.
WinHex is even able to interpret *spanned* raw image files, that is, image files that consist of separate segments of any size. For WinHex to detect a spanned image file, the first segment may have an arbitrary name and a non-numeric extension or the extension “.001”. The second segment must have the same base name, but the extension “.002”, the third segment “.003”, and so forth. Both the Create Disk Image command and the DOS cloning tool X-Ways Replica are able to image disks and produce canonically named file segments. Image segmentation is useful because the maximum file size supported FAT file systems is limited.
In some rare cases WinHex may be unable to correctly determine whether the first sector in an image is the sector that contains a master boot record or already a boot sector, and consequently interprets the image structure in a wrong way. If so, hold the Shift key when invoking this command. That way WinHex will ask you and not decide on its own. When the segments of a raw image are spread across two different drives, you may hold the Control key to be able to specify the other storage location.
With a *forensic* license, WinHex can also interpret evidence files (.e01 images), which can be

created with the Create Disk Image command.

**Assemble RAID System:** WinHex can internally destripe RAID level 0 and level 5 systems consisting of up to 5 components (physical hard disks or images). That way it is not necessary to use scripts that unstripe and export RAID systems to a new image, saving you time and drive space. Components that are available as images need to be opened and interpreted before you use this function. You need to select the components in the correct order. WinHex lets you specify the strip size in sectors (often 128) and different RAID header sizes per component (often simply 0). You can usually tell that either the component order, the stripe size, the stripe pattern, or the RAID header size is incorrect when no partitions are detected or partitions with unknown file systems or with file systems that cannot be interpreted properly.

When you add an assembled RAID system to a case (and optionally partitions opened from such a RAID system), the selected RAID configuration parameters are saved with the evidence objects, which allows to access the RAID system instantly in later sessions (forensic licenses only).

In RAID level 5, data is not only striped across all component disks in a rotating pattern, but also interspersed with parity blocks for redundancy. RAID level 5 is implemented in different ways by different RAID controller manufacturers in that they employ different stripe/parity patterns. The supported patterns are the following:

Backward Parity (Adaptec)
Component 0: 0 2 P
Component 1: 1 P 4
Component 2: P 3 5

Backward Dynamic Parity (AMI)
Component 0: 0 3 P
Component 1: 1 P 4
Component 2: P 2 5

Backward Delayed Parity (HP/Compaq)
Component 0: 0 2 4 6 8 10 12 14
Component 1: 1 3 5 7 P P P P
Component 2: P P P P 9 11 13 15

Forward Parity
Component 0: P 2 4
Component 1: 0 P 5
Component 2: 1 3 P

If one of the RAID component disks is not available, you can assemble a RAID 5 system nonetheless because one component is redundant. Simply select a dummy substitute (one of the *other, available* components of the same RAID system) as the *missing* component and declare that component "missing".

**Gather Free Space:** Traverses the currently open logical drive and gathers all unused clusters in a destination file you specify. Useful to examine data fragments from previously existing files that have not been deleted securely. Does not alter the source drive in any way. The destination file must reside on another drive.

**Gather Slack Space:** Collects slack space (the unused bytes in the respective last clusters of all cluster chains, beyond the actual end of a file) in a destination file. Otherwise similar to Gather Free Space. WinHex cannot access slack space of files that are compressed or encrypted at the file system level.

**Gather Inter-Partition Space:** Captures all space on a physical hard disk that does not belong to any partition in a destination file, for quick inspection to find out if something is hidden there or left from a prior partitioning.

**Gather Text:** Recognizes text according to the parameters you specify and captures all occurrences from a file, a disk, or a memory range in a file. This kind of filter is useful to considerably reduce the amount of data to handle e.g. if a computer forensics specialist is looking for leads in the form of text, such as e-mail messages, documents, etc. The target file can easily be split at a user-defined size. This function can also be applied to a file with collected slack space or free space, or to damaged files in a proprietary format than can no longer be opened by their native applications, like MS Word, to recover at least unformatted text.

**Evidence File Container:** Only available with a forensic license: Allows to create a new file container, open an existing one, and close the active file container. An evidence file container is a raw image file formatted with the XWFS file system by X-Ways AG. Files selected in the directory browser can be added to the active file container with the directory browser's context menu. Certain technical metadata (e.g. the original cluster allocation) are lost, however, name, path, size, attributes, timestamps, deletion state, and especially the contents of the file are fully retained in a file container. So when you need to pass on selected files (even from different evidence objects) that are of particular relevance to a case, in a single handy archive, to other persons involved in that case, who do not need to or must not see irrelevant files, this feature comes highly recommended. Evicence file containers can be interpreted and conveniently examined like other image files with X-Ways Forensics 12.6 and later. Please note that only files that are part of the volume snapshot can be added to a container. Once added, a file cannot be removed any more in this version. Hold the Shift key when invoking this command in order to specifically add file slack to the container as well. Hold the Ctrl key to add only a file's slack, not the file itself. Depending on the Security Options, your virus scanner might be able to prevent that viruses will be added to an evidence file container.

**Bates-Number Files:** Bates-numbers all the files within a given folder and its subfolders for discovery or evidentiary use. A constant prefix (up to 13 characters long) and a unique serial number are inserted between the filename and the extension in a way attorneys traditionally label paper documents for later accurate identification and reference.

**Trusted Download:** Solves a security problem. When transferring unclassified material from a classified hard disk drive to unclassified media, you need to be certain that it will have no

extraneous information in any cluster or sector "overhang" spuriously copied along with the actual file, since this slack space may still contain classified material from a time when it was allocated to a different file. This command copies files in their current size, and no byte more. It does not copy entire sectors or clusters, as conventional copy commands do. Multiple files in the same folder can be copied at the same time.

**Highlight Free Space/Slack Space:** Displays offsets and data in softer colors (light blue and gray, respectively). Helps to easily identify these special drive areas. Works on FAT, NTFS, and Ext2/Ext3 partitions.

## 6.9 Menú Opciones

**Opciones Generales:** véase más adelante

**External Programs:** Here you can specify what external file viewing programs you would like to invoke from inside WinHex using the Tools menu. Also the installation path of the viewer component that is included in forensic licenses for v12.05 and later can be specified here (by default: subdirectory ..\viewer). The viewer component can also be specifically enabled or disabled.

**Opciones de Deshacer:** véase más adelante

**Opciones de Seguridad:** véase más adelante

**Opciones del Intérprete de Datos:** véase Intérprete de Datos

**Modo Edición:** Allows you to select the edit mode globally. (The details panel's context menu allows to select the edit mode specifically for an active edit window.)

**Juego de Caracteres:** Le permite seleccionar el juego de caracteres que se utilizará para la presentación en pantalla y la entrada de teclado entre ANSI ASCII, IBM ASCII y EBCDIC. También puede utilizar el atajo **SHIFT+F7**. EBCDIC (originario de los mainframes IBM) no está disponible en impresión.

## 6.10 Menú Ventanas

**Gestor de Ventanas:** Muestra todas las ventanas y proporciona un método para „cambio instantáneo entre ventanas“. También puede cerrar ventanas y guardar archivos.

**Save Arrangement As Project:** Writes the current window constellation into a project file. From the Start Center you will then be able to load the project and restore editing positions in each document at any time, to conveniently continue your work right where you left it or to begin your work in case of a recurring task.

**Cerrar Todo:** Cierra todas las ventanas y con ellas los archivos abiertos, discos y secciones de RAM que se estén mostrando.

**Cerrar Todo Sin Confirmación:** Cierra todas las ventanas y con ellas los archivos abiertos y discos, pero sin darle la oportunidad de guardar las modificaciones realizadas.

**Cascada/Mosaico:** Posiciona en Cascada/Mosaico todas las ventanas de edición.

**Minimizar:** Minimiza todas las ventanas.

**Reorganizar Iconos:** Este comando reordena todas las ventanas.

## 6.11 Menú Ayuda

**Contenido:** Muestra los contenidos del archivo de ayuda.

**Configuración:** Cambia el idioma de la interfaz de usuario entre alemán, inglés, francés y español.

**Inicializar:** Utilice este comando para restaurar las opciones por defecto del programa.

**Desinstalar:** Utilice este programa para eliminar WinHex de sus sistema. Este comando funciona correctamente incluso aunque no instalará WinHex utilizando el programa de instalación.

**Online:** Abre la página web de WinHex en su navegador, the support forum, the Knowledge Base, or the newsletter subscription page.

**Acerca de WinHex:** Displays information about WinHex (the program version, your license status, and more).

## 6.12 Menú Contextual de Windows

La interfaz de Windows muestra el menú de contexto cuando el usuario pulsa un objeto con el botón derecho del ratón. WinHex está presente en dicho menú únicamente si activa la opción correspondiente (véase „Opciones Generales“).

**Editar con WinHex:** Abre el archivo seleccionado en WinHex.

**Abrir en WinHex:** Le permite abrir todos los archivos de la carpeta seleccionada en WinHex, exactamente igual que con el comando Abrir Carpeta del menú Archivo.

**Editar Disco:** Abre el disco seleccionado en el editor de disco de WinHex. Si mantiene pulsada

la tecla **SHIFT** durante la operación, en vez de la unidad lógica se abrirá el disco físico correspondiente, caso de existir. (Esta última opción no está disponible bajo Windows NT).

WinHex proporciona su propio menú contextual en la barra de estado, el Intérprete de Datos y el gestor de posición.

## 6.13 Directory Browser Context Menu

The directory browser context menu allows the user to directly interact with the currently *selected* files/directories, notably *not* the *tagged* items. There are a number of menu commands which are available depending on the selected items. Double clicking files and directories will, depending on the circumstances, either call "View", "Explore" or the associated external program.

**View**

This command allows viewing the selected file with WinHex' internal viewers for Windows Registry files and various graphical file formats. For other files, the mode of operation depends on the installed components: If X-Ways Trace is installed, and the file is either an “info2” file used by the Windows Recycle Bin or Internet Explorer's “index.dat” or Mozilla's/Firefox's “history.dat” or Opera's “dcache4.url”, X-Ways Trace is invoked for these files. If the X-Ways Forensics external viewer component is active, all other files are sent to that viewer. If it is not, the first installed external program will be called instead.

Exceptions to all of the above are files beyond 2 GB in size and NTFS system files. These are always opened as data windows.

**Explore**

Only available for directories and archives (ZIP, RAR, TAR...), this command allows navigating into them within the directory browser. Double-clicking archives or directories does the same. A command that allows listing the contents of directories as well as their subdirectories at the same time can be found in the directory tree's context menu instead (in the Case Data window, "Explore recursively").

**External Programs**

Allows sending the selected file(s) to one of the external programs currently configured or the file's associated program in the current Windows installation. This association is determined based on file extension as is usual within Windows.

**Recover/Copy**

Allows copying the selected files from their current location to a location available for a standard Windows file dialog, e.g. out of an interpreted image file or from a local disk. This can be applied to both existing and deleted files and directories. When working with an active case and if

logging is enabled, the copy/recovery process is documented in the case log. Both the source and the target paths are recorded.

**Edit Comment**

Use this command to add a comment to an item in the directory browser or to edit or remove an existing comment. After entering comments, you can conveniently set the filter such that only commented items are shown or only items with specific comments, e.g. those with a certain relevance. Requires a forensic license.

**Add to Noteworthy Table/Tag/Untag Item**

In the directory browser of an evidence object, you include files in a report table of noteworthy items. These files will then also be listed in the case report. Having them in a dedicated contents table allows to conveniently copy/recover them in a single step at a later point of time or get a gallery overview of these files specifically. In order to remove files from the dedicated contents table of noteworthy files, use the "Delete from list" command in the directory browser context menu when that contents table is loaded or press the Del key on your keyboard. The click the floppy disk icon to save.

Tagging files means highlighting them visually. The visual highlighting can be undone with the context menu command "Untag item". Refining the volume snapshot can be limited to tagged files.

**Add to Active Case**

Performs the same operation as Recover/Copy but at the same time, the resulting file(s) will be added to the current case as evidence objects.

**Export List**

Outputs the current contents of the directory browser to a tab-delimited text file, which can be easily imported and further processed e.g. in MS Excel. The output format is the same as for a so-called report table. However, this command is also available when not working with a case. Requires a specialist license.

**Hide**

You may hide selected items or hide all but the tagged items. If actually filtered out, hidden files are excluded from the directory browser, the gallery view, logical searches, copying actions, additions to an evidence file container, etc. If you are only allowed to examine the contents of certain directories, you could initially hide all files in all other directories to ensure that. Refining the volume snapshot can be limited to files that are not hidden. Hidden items are actually filtered out only if the corresponding filter is enabled in the directory browser options.

**Position**

The Position group of commands allows interactions with the currently selected file on a generally more technical level. It allows accessing the file's (or directory's) first cluster on the disk in the sectors view, accessing its related information like MFT record in NTFS or Inode in Ext2/Ext3 and also sorting the files by their physical order on disk: "Sort by directory entry location" (FAT), "Sort by Inode Offset" (Ext2/Ext3) or "Sort by MFT ID" (NTFS), respectively, allow to see files and folders in the order in which they physically appear in file system data structures (directory entries, the MFT, or Inode tables).

The Position menu also allows calling for a file's or directory's cluster list, i.e. the cluster list window will be opened and filled with the selected item's cluster list, and it allows deleting the selection from contents tables. The deletion of items from a contents table can be made permanent by clicking the floppy disk icon that will appear in the directory browser's caption line. You may also mark items as to be hidden in the volume snapshot (see directory browser Filter options). The menu also allows to add or move files to special report contents tables.

When you are examining files based on their contents only, where filenames, timestamps, deletion status and other meta-data are of no relevance, then you can use the "Remove duplicates" command to remove duplicated files from a contents table or to hide duplicated files from the currently listed part of a volume snapshot, based on hash values (if hash values were calculated). This command will first sort by hash values.

**Logical Search**

See chapter of that name.

**Create Hash Set**

Creates a hash set of the currently selected files and directories and their subdirectories directly within the internal hash database.

**Create Directory Contents Table**

Creates a contents table just like a drive contents table except that it exclusively focuses on files located within the directory currently selected and its subdirectories.

**Print**

If the separate viewer component is active, you may select files for printing. You will be prompted for each file.

**Open**

Opens currently selected files or directories in separate data windows. In the case of a directory, the directory's data structures will be opened.

# 7 Opciones

## 7.1 Opciones Generales

**1ª columna:**

- At startup, WinHex can optionally **show** the **Start Center** or **restore** the **last window arrangement** (all windows with their sizes and the positions as you left them in the precedent WinHex session).

- Specify the number of **recent**ly opened **documents** to remember and to **list** in the Start Center (255 at max.). Up to 9 of them are also listados al ginal del menú Archivo.

- Si lo desea, **WinHex** aparecerá en el **menú contextual** de Windows. La interfaz muestra este menú cuando el usuario pulsa sobre algún objeto con el botón derecho del ratón. El menú de WinHex aparecerá al pulsar sobre archivos, carpetas y discos. Si esta opción no está completamente seleccionada, no habrá menú contextal para los archivos.

- La opción **Permitir múltiples instancias** le permite ejecutar WinHex más de una vez al mismo tiempo. Si la opción no está activada, WinHex pondrá la ventana principal de la instancia actualmente en ejecución en segundo plano en vez de crear una nueva instancia de programa.

- **No actualizar fecha del archivo** significa que WinHex will preserve the last modification time when a modified file is saved with File | Save or Save As.

- If **Check for surplus sectors** is disabled, WinHex will not try to search for surplus sectors when a physical hard disk is opened. When additional sectors are found, WinHex will remember them the next time you open the disk. You may enforce a new check by holding the Shift key while opening the disk. Checking for surplus sectors may cause very long delays, strange behavior or even damage to the Windows installation on *some very few* systems. Only under Windows XP surplus sectors are included automatically, which renders this option obsolete.

- Since v12.7 SR-8, WinHex by default **sorts** and enumerates disk **partitions** by their physical **location**.

- If **Auto-detect deleted partitions is enabled**, WinHex tries to identify obvious deleted partitions automatically in gaps between existing partitions and in unpartitioned space directly following the last partition, when opening physical hard disks. Such additionally detected partitions will be listed in the Access button menu and marked as deleted. Please note that deleted partitions detected in gaps between existing partitions cause the partition numbering to be changed. E.g. an existing partition #3 might become partition #4 if a deleted partition is detected on the disk before it.

- The **alternative access method 1** for physical hard disks under Windows 2000/XP may allow to access hard disks formatted with an unconventional sector size or other media that cannot be accessed otherwise. Note that it may be slower than the regular access method. If considerably slower, WinHex will notify you of this and recommend to revert to the standard access method. **Access method 2** affects physical hard disks only as well, under Windows 2000/XP. Both alternative methods allow you to specify a timeout in milliseconds after which read attempts will be aborted. This can be useful on disks with bad sectors, where an attempted read access to a single sector could otherwise cause a delay of many seconds or minutes.

- By default, **edit windows** are not **opened** in a **maximized** state.

- On a right click, **WinHex** can bring up a special **context menu**, the regular edit menu, or define the end of the current block. If this option is disabled, you can still bring up the context menu if you hold the Shift key while right-clicking.

- Si selecciona **Mostrar iconos de archivo**, los iconos almacenados en un archivo se mostrarán en la columna de información. Si un archivo no contiene iconos se mostrará el icono del *tipo* de archivo, a menos que esta opción no esté „completamente“ seleccionada.

- La tecla **ENTER** puede utilizarse para introducir hasta cuatro valores hexadecimales de dos dígitos. Un ejemplo útil es **0x0D0A**, el cual se interpreta como un marcardor de final de línea en el entorno Windows (Unix: 0x0D). The Start Center could then still be opened using **SHIFT**+**ENTER**.

- Decida si desea utilizar la tecla **TAB** para cambiar del modo hexadecimal al decima y viceversa, o para introducir el carácter TAB (0x09). En cualquier caso, **TAB**+**SHIFT** siempre le permitirá cambiar el modo numérico actual.

**2º columna:**

- Indique la **carpeta** donde se crearán los **archivos temporales**.

- Indique la **carpeta** donde se crearán los **archivos de backup** (.whx).

- Indique la **carpeta** donde se crearán los **archivos de proyecto, script y case**.

- Specify the **folder** in which to maintain the **internal hash database**.

- **Reduced user interface**: Available when operated with a forensic license. Slightly reduces and simplifies the menu structure. The checkbox has a third state ("forensic lite interface"), which is meant for investigators in law enforcement
  - who are specialized in areas such as white-collar crime, tax fraud, etc.
  - who do not need profound knowledge of computer forensics
  - who do not need technical insights that WinHex and XWF are well-known to offer

- who receive e.g. convenient-to-handle X-Ways evidence file containers from well-versed computer forensics examiners with only selected files from various sources (e.g. "all documents that contain the keywords x and y"), with obviously irrelevant stuff already filtered out
- who need to review hundreds of electronic documents, identify relevant ones, add comments to them, identify logical structures and connections between them with the help of their comments, and print documents, all within the same environment with a few mouse clicks, which saves the time to extract and load each document in its associated application
- who may or may not need to work in an environment severely restricted by the system administrator anyway

The "forensic lite" interface lacks many advanced technical options, to allow for easier access to non-technical personnel. Forensic licenses that only allow to use the "forensic lite" interface are available at 50% the regular rate on request.

- If the creation of thumbnails for **pictures within** large solid RAR **archives** for **gallery** view is too slow, you may want to disable it.

- You may specify your **preferred thumbnail size** in pixels. WinHex will decrease the size automatically if needed to ensure that at least as many files are displayed in the gallery view as are displayed in the currently visible section of the directory browser.

- When gallery view is enabled, WinHex can optionally continue **loading thumbnails in the background** when the current view is full, if the number of files in the current directory is not too big.

**3ª columna:**

- Non-printable **characters** with a character set value smaller than **0x20** can be represented by a user-defined other character.

- Los **offsets** pueden mostrarse y referenciarse en notación decimal o **hexadecimal**. Esta opción es válida para todo el programa.

- Cuando utilice el editor de RAM es lógico hacer que WinHex muestre **direcciones virtuales** en vez de offsets partiendo de cero. Esto se hace siempre al utilizar notación hexadecimal. El cuadro de diálogo del comando Ir A Offset también funcionará con direcciones virtuales.

- Se mostrarán los **separadores** de página y sector. Si la opción está activada sólo parcialmente, solo se mostrarán los separadores de sector.

- Decida cuantos **octetos** deben mostrarse **por línea**.

- Decida cuantos **octetos** deben mostrarse en un mismo **grupo**. Los mejores resultados se obtienen en la mayoría de los casos utilizando potencias de 2.

- Especifique cuantas **líneas** deben **desplazarse** cuando se utilice un **roll-mouse** o ratón de rueda (en caso de disponer de él).

- **NTFS: MFT auto coloring:** Highlights the various elements in FILE records of the NTFS file system, when the blinking cursor is located within such a record, to facilitate navigation and understanding. Requires a specialist or forensic license.

- Seleccione el **color** que se usará como **fondo** al seleccionar un **bloque**. Sólo podrá cambiarlo si la opción „Utilizar colores de Windows“ está desactivada.

- Select a **color** used as the **background** of every other fixed-length **record**, if record presentation is enabled (see Position menu).

- Select the default **color** for newly created **annotations**/positions/bookmarks.

- You may want WinHex to **hilite modified bytes**, i.e. display altered parts of a file, disk, or memory in a different color, so you can distinguish between original data and changes you have made so far. You may select the hilite color.

- Puede elegir una **fuente** para el modo ANSI-ASCII. La fuente WinHex incorpora el juego de caracteres de Windows completo (incluso los caracteres ™ y € y otros signos diversos).

- **Mostar barra de progreso de Windows** reemplaza la barra de progreso de WinHex con la típica barra común a la mayoría de programas de Windows.

- Por último pero no por ello menos importante, puede seleccionar uno de **estilos**.

Las opciones de fábrica pueden restaurarse utilizando el comando Inicializar del menú Ayuda.

## 7.2 Directory Browser Options

- **Grouping files and directories** in the directory browser is optional.

- **Grouping existing and deleted** items in the directory browser is optional. There are two possibilities how to enable this feature, either potentially recoverable deleted files (marked with a question mark) and known unrecoverable files (marked with an X) are internally grouped as well or not.

- By default, files **recover**ed via the directory browser are recreated in the output folder **including** their original **path**.

- Optionally, X-Ways Forensics can **append** the presumed **correct ext**ension when copying files (to a hard disk or to a container) after the signature check.

- Files can optionally be **opened**, **searched**, and **copied**/recovered including their **slack**.

- By default, **Ctrl+A** (select all) in the directory browser **includes** non-existent **files** whose first cluster has been reallocated. However, that behavior can be disabled so that e.g. less clusters are covered twice when you search logically in selected files.

- **Recursive selection statistics** reveal how many subdirectories, files and how much data are in a directory when you select it in the directory browser.

- Listing sub**dir**ectorie**s** when **exploring recursively** is optional.

- The **directory browser** can optionally be displayed with a **grid**.

- There is an option to display timestamps with **tenths of seconds**. Useful for the file systems NTFS and FAT that provide for this precision in all or some timestamps. Note that fractions of seconds are not retained when you export file lists or report tables.

- Optionally, the actually used **time zone conversion bias**, including daylight saving where appropriate, can be displayed right in the timestamp columns in the directory browser.

- **Loading contents tables** or report tables into the directory browser is faster if the items in such tables are not re-associated with the volume snapshot. If you opt for faster loading, the tagging status, comments, hash values, corresponding hash set and category will not be visible or changeable for items in a table.

Various columns in the directory browser are optional. They are displayed if they have a non-zero column width or hidden if their width is zero.

**Filters**

The following can be dynamically filtered out:

- Deleted files and directories. Useful if you are merely interested in data in existing files.
- Existing files. Useful if you are merely interested in recovering lost files.
- Files and directories that have been marked as to be hidden in the volume snapshot. (All such marks can be removed.)
- $EFS attributes, non-directory INDX streams and BMP attributes on NTFS volumes.

You may also define filters based on criteria such as filenames, file type categories, and matching hash set categories. Whenever an active filter actually filters out files or directories in the directory browser, this is flagged with a filter icon in the directory browser's header line. Please note that report tables are not automatically reloaded to reflect new filter settings.

## 7.3 Opciones Deshacer

La disponibilidad del comando „Deshacer“ depende de las siguineste opciones:

- Especifique cuantas acciones seguidas pueden ser invertidas por el comando Deshacer. Esta opción no afecta el número de entradas de teclado reversibles, que sólo está limitado por la cantidad de RAM disponible.

- Para ahorrar tiempo y espacio en su disco duro, puede especificar un tamaño de archivo límite. Si un archivo es mayor, no se crearán backups y el comando Deshacer sólo estára disponible para las entradas desde teclado.

- Los backups creados automáticamente para uso interno con el comando Deshacer son borrados por WinHex cuando se cierra el archivo, si la opción correspondiente está completamente seleccionada. Si está sólo parcialmente seleccionada, entoces se borrarán al salir de WinHex.

- Para todos los tipos de operaciones de edición debe decidir si deben ser reversibles o no. Si lo son, un backup interno se creará antes de que la operación tenga lugar.

## 7.4 Opciones de Seguridad

El **caché de lectura de disco** acelera el acceso secuencial al disco por parte del editor. Esta opción está especialmente recomendada cuando se desplace por los sectores de un CD-ROM o disquete, ya que el número de accesos físicos necesarios se reduce notablemente.

With the option **Keep volume snapshots between sessions** enabled, all information on file systems in opened volumes collected by WinHex (Disk Tools menu and/or Specialist menu) remains in the folder for temporary files even when WinHex terminates. WinHex can then reuse drive maps during later sessions. Volume snapshots of evidence objects are always kept, regardless of this setting.

**Quick snapshots without cluster allocation** speeds up taking a volume snapshot (in particular for the file systems Ext2, Ext3 and ReiserFS, and in particular also when the volume snapshot files are created across a slow USB 1.1 interface or network), however, causes WinHex to lose its ability to tell each sector’s and cluster’s allocation (for which file it is used). This will also disable the functionality of the Sync button and the preview mode when reviewing search hits. You may use the command “Take New Volume Snapshot” of the Tools menu to update the view of a volume, e.g. after unchecking this option.

Utilive la opción **Integridad de la memoria virtual** para asegurarse de que el editor de RAM comprueba la estructura de la memoria virtual antes de cada operación de *lectura* y *escritura*. Si la estructura de la misma ha cambiado, se podrá prevenir un posible error de lectura. Seleccionar esta opción puede suponer una importante pérdida de velocidad en los sistemas Windows NT.

Cuando edite la „memoria completa“ de un proceso, WinHex generalmente *nunca* comprueba las alteraciones antes de cada lectura, incluso aunque esta opción esté marcada.

Antes de guardar las modificaciones producidas en un archivo existente (por ejemplo, antes de **actualizar** el **archivo**), se le pedirá **confirmación**. Para evitar este comportamiento de WinHex, desactive la opción correspondiente.

Optionally, files can be **indirectly** added to evidence file **containers**, via your own hard disk. That means they are not copied directly into the container, but to your folder for temporary files first (cf. General Options), and only then from there into the container. This can be beneficial because it allows a resident antivirus software to intercept these files (check them for viruses, disinfect/disarm them, rename them, move/delete/lock them, etc.), so that it prevents viruses from making it into a container. The resulting container is free of known viruses (depending on the antivirus software in use) and can reasonably be passed on to and used in an environment with higher sensitivity, higher security requirements, and/or less sophisticated virus protection.

La **clave** requerida para la encriptación y desencriptación puede introducirse en un cuadro de edición normal, pero opcionalmente puede hacer que se muestren **asteriscos** en vez de los caracteres que teclea. En ese caso tendrá que confirmar la clave en un segundo cuadro de diálogo para evitar errores de mecanografía.

Por defecto, la **clave** se **almacena** en la **memoria** principal (en modo encriptado) durante el tiempo que WinHex esté funcionando, de manera que nunca tendrá que teclearla de nuevo si la utiliza en varias ocasiones. Posiblemente prefiera que WinHex borre la clave de la memoria después de cada utilización.

Decide whether or not WinHex shall **prompt before executing a script**, or only before executing a script via the command line.

## 7.5 Opciones de Búsqueda

**Distinguir mayúsculas de minúsculas:** Si esta opción está activada, WinHex distingue entre los caracteres en mayúsculas y en minúsculas, de manera que „Opcion“ no se encontrará en la palabra „opcionalmente“.

**Juego de caracteres Unicode:** El texto especificado se buscará utilizando los 256 ANSI-ASCII caracteres equivalentes Unicode, donde el byte de orden superior es 0. The simultaneous search allows to search for the same text at the same time in Unicode and ASCII. For this to work, the checkbox needs to be “half” checked.

Puede especificar un **comodín** (un carácter o un valor hexadecimal de dos dígitos), que represente un byte. Por ejemplo esta opción puede utilizarse para encontrar tanto „Speck“ como „Spock“ cuando busque „Sp?ck“ si ha indicado que el signo de interrogación sea el comodín.

**Palabras completas solamente:** La cadena buscada será reconocida sólo si está separada de otras palabras, por ejemplo por espacios en blanco o signos de puntuación. Si esta opción está activada, „tomato“ no se encontrará en „automaton“.

**Sentido de búsqueda:** Decida si quiere que WinHex busque desde el inicio hasta el final, o bien hacia arriba o hacia abajo desde la posición actual.

**Condición: Offset modulo x = y:** El algoritmo de búsqueda acepta ocurrencias de la cadena a buscar sólo en los offset que cumplan las condiciones especificadas. Por ejemplo, si está buscando un dato que típicamente se encuentra en el décimo byte de un sector del disco duro, puede especificar x=512, y=10. Si está buscando datos con alineación DWORD, especifique x=4, y=0 para reducir el número de ocurrencias.

**Búscar sólo en el bloque:** La operación de búsqueda está limitada al bloque actual.

**Búscar en todos las ventanas abiertas:** La operación de búsqueda se aplicará a todos las ventanas abiertas. Pulse F4 para continuar buscando en el archivo siguiente. Si „Búscar sólo en el bloque“ está activada a la vez, la operación de búsqueda se limitará al bloque actual dentro de cada uno de los archivos.

**Contar ocurrencias/Guardar posiciones de ocurrencias:** Hace que WinHex no muestre cada ocurrencia, sino que las cuente. Si esta opción está completamente activada, WinHex incluirá todas las ocurrencias en el gestor de posiciones.

**Search for “non-matches”:** In “Find Hex Values” you may specify a single hex value with an exclamation mark as a prefix (e.g. !00) to make WinHex stop when it encounters the first byte value that *differs*.

**GREP syntax:** Available with the Simultaneous Search and Logical Search only. Regular expressions are a powerful search tool. A single regular expression may match many different words. The following characters have a special meaning in regular expressions, as explained below: ( ) [ ] { } | \ . # + ?. Where these special characters are to be taken literally, you need to prefix them with a backslash character (\).

The | operator is used to denote alternative matches. You can use the regular expression *car (wheel|tire)* to search for the words "car wheel" and "car tire". Any match must equal the parts before, after, or between any | operators present. The effect of | is only limited by parentheses.

. and # are wildcards: . matches any character, # matches any numeric character. You can define sets of characters with the help of square brackets: [xyz] will match any of the characters x, y, z. [^xyz] will match any character except x, y, or z. You can define ranges of characters using a hyphen: [a-z] matches any lower-case letter. [^a-z] matches all characters except lower-case letters. The listing may comprise individually listed characters and ranges at the same time: [aceg-loq] matches a, c, e, g, h, i, j, k, l, o, and q. All characters except [, ], -, and \ are taken literally between square brackets, even the wildcard characters . and #.

Byte values that correspond to ASCII characters that cannot be easily produced with a keyboard can be specified in decimal or hexadecimal notation: For example, \032 and \x20 are both equivalent to the space character in the ASCII character set. This kind of notation is supported even in between square brackets. E.g. [\000-\x1f] matches non-printable ASCII characters.

Multiplier characters (*, +, and ?) indicate that the preceding character(s) may or must occur more than once (see below). Complex example: a(b|cd|e[f-h]i)*j matches aj, abj, acdj, aefij, aegij, aehij, abcdj, and abefij.

. A period matches any single character.
# A pound sign matches any numeric character [0-9].
\nnn A byte value specified with three decimal digits (0...255)
\xnn A byte value specified with two hexadecimal digits (0…FF).
E.g. \x0D\x0A is the Windows line break.
? Matches one or zero occurrences of the preceding character or set.
* Matches any number of occurrences of the preceding character, including zero time.
+ A plus sign after a character matches any number of occurrences of it except zero.
[XYZ] Characters in brackets match any one character that appears in the brackets.
[^XYZ] A circumflex at the start of the string in brackets means NOT.
[A-Z] A dash within the brackets signifies a range of characters.
\ Indicates that the following special GREP character is to be treated literally.
{X,Y} Repeats the preceding character or group of characters X-Y times.
(ab) Functions like a parenthesis in a mathematical expression. Groups ab together for + and *.
a | b The pipe acts as a logical OR. So it would read "a or b".

## 7.6 Opciones de Reemplazo

**Preguntar al encontrar:** WinHex esperará que tome una decisión cuando encuentre una ocurrencia. Podrá entonces reemplarla, continuar la búsqueda o abortar la operación

**Reemplazar todo:** Todas las ocurrencias serán reemplazadas automáticamente.

**Distingir mayúsculas de minúsculas:** Los caracteres que van a ser reemplazados se buscarán utilizando esta opción (véase Opciones de Búsqueda).

**Juego de caracteres Unicode:** Los caracteres especificados se buscarán y reemplazarán en formato Unicode (véase Opciones de Búsqueda).

Puede especificar un carácter o un valor hexadecimal de dos dígitos como comodín. Esto se hace habitualmente en la cadena de búsqueda, pero si la cadena de sustitución contine un comodín, el carácter en la posición correspondiente no será reemplazado. Por lo tanto, „black“ y „block“ pueden ser reemplazados simultáneamente con „crack“ y „crock“ (utilizando „bl?ck“ y „cr?ck“ como cadenas de búsqueda y sustitución respectivamente).

**Palabras completas sólamente:** La cadena a buscar se reconoce sólo si está separada de otras palabras, por ejemplo por signos de puntuación o espacios. Si esta opción está activada, „tomato“ no se reemplazará en „automaton“.

**Sentido de búsqueda:** Especifique si quiere que WinHex reemplace desde el principio hasta el final o bien desde la posición actual hacia arriba o hacia abajo.

**Reemplazar sólo en el bloque:** La operación de sustitución tiene lugar sólo en el bloque actualmente seleccionado.

**Reemplazar en todos los archivos abiertos:** La operación de sustitución se aplica a todos los archivos que no estén abiertos en modo de sólo lectura. Si la opción „Reemplazar sólo en el bloque“ está activada al mismo tiempo, la sustitución se limitará al bloque indicado dentro de cada uno de los archivos.

Consejo:

WinHex es capaz de reemplazar una cadena o una secuencia de valores hexadecimales con otra de diferente longitud. Se le preguntará cual de los siguientes métodos debe aplicarse:

**Primer método:** Los datos siguientes a la ocurrencia se moverán tantos bytes como la diferencia de longitud, de manera que el tamaño del archivo cambia. Este método no debe aplicarse a ciertos tipos de archivo, como los ejecutables. Incluso es posible *no* especificar *nada* como cadena de sustitución, con lo que todas las ocurrencias se *borrarán* del archivo.

**Segundo método:** La cadena de sustitución se escribe en el archivo en la posición de la ocurrencia. Si dicha cadena es más corta que la buscada, los caracteres en exceso de ésta última permanecerán en el archivo. En el otro caso los bytes que sigan a la ocurrencia serán sobreescritos (en tanto no se alcance el final del archivo). El tamaño del archivo no se ve afectado en absoluto por esta operación.

# 8 Miscelánea

## 8.1 Bloque

Puede marcar una parte de un archivo abierto como un „bloque“. Dicha parte puede ser luego manipulada por diversas funciones en el menú de edición exactamente igual que las selecciones hechas en otros programas de Windows. Si no se ha definido ningún bloque, generalmente las funciones se aplicarán a todo el archivo.

La posición actual y tamaño del bloque se muestran en la barra de estado. Pulsando dos veces con el botón derecho del ratón o pulsando la tecla **ESC** eliminará el bloque (no el contenido).

## 8.2 Modicar Datos

Utilice este comando para modificar los datos de un bloque o de todo el archivo, en caso de que no haya ningún bloque definido. En esta versión de WinHex hay tres tipos de modificaciones de datos disponibles. Puede sumar un número entero constante a cada elemento de los datos, invertir los bits, aplicar una operación XOR a los datos junto con un número hexadecimal (una clase sencilla de encriptación), OR o AND, bits are shifted logically o reflejar los bytes. By shifting bits, you can simulate inserting or removing single bits at the beginning of the block. You may also shift entire *bytes* (currently to the left only, by entering a negative number of bytes). This is useful if you wish to cut bytes from a very huge file in in-place mode, which would otherwise require the creation of a huge temporary file.

Reflejar Bytes

Este comando asume que todos los datos consisten en elementos de 16 bits (o de 32 bits en su caso) e intercambia los bytes más significativos con los menos significativos (o al revés en su caso). Utilicelo para convertir datos en formato big-endian en little-endian y viceversa.

Suma

Indique un número positivo o negativo, decimal o hexadecimal, y se sumará a cada elemento del bloque seleccionado. También debe definir el formato del número entero: el tamaño (1, 2 o 4 bytes) y el tipo (signed o unsigned).

Hay dos modos de proceder si el resultado de la suma está fuera del rango del formato de entero seleccionado. O bien el límite del rango se toma con el nuevo valor (I) o bien se ignora el exceso (*carry*) (II).

Ejemplo: formato unsigned de 8 bits
I. FF + 1 → FF (255 + 1 → 255)
II. FF + 1 → 00 (255 + 1 → 0)

Ejemplo: formato signed de 8 bits
I. 80 - 1 → 80 (-128 - 1 → -128)
II. 80 - 1 → 7F (-128 - 1 → +127)

- Si decide utilizar el primer método, WinHex le informará de cúanto se ha sobrepasado el límite del rango.

- El segundo método asegura que la operación sea reversible. Simplemente -x en vez de x con el mismo formato de entero y podrá recuperar los datos originales.

- Cuando utilice el segundo método no habrá diferencia entre utilizar un formato de entero

signed o utilizar un formato unsigned.

## 8.3 Conversiones

WinHex proporciona el comando Convertir en el menú Edición para facilitar las conversiones entre diferentes formatos de datos, además de para encriptación y desencriptación. La conversión puede ser aplicada opcionalmente a todos los archivos abiertos en vez de limitarse al archivo actualmente mostrado. Los formatos marcados con un asterisco (*) sólo pueden ser convertidos como un archivo completo, no como un bloque. Se reconocen los siguientes formatos:

- ANSI-ASCII, IBM-ASCII (dos juegos de caracteres ASCII diferentes)
- EBCDIC (un juego de caracteres de los mainframes IBM)
- Caracteres en mayúsculas/minúsculas (ANSI-ASCII)
- Binario* (datos sin formato)
- Hex ASCII* (representación hexadecimal de datos sin formato como texto ASCII)
- Intel Hex* (=Extended Intellec; datos Hex ASCII en un formato especial con checksums, etc.)
- Motorola S* (=Extended Exorcisor; ditto)
- Base64*
- UUCode*

Observe:
- Cuando convierta datos Intel Hex o Motorola S, los checksum internos de esos formatos no se comprueban.
- Depending on the file size, the smallest possible output subformat is chosen automatically. Intel Hex: 20-bit or 32-bit. Motorola S: S1, S2, or S3.
- When converting from binary to Intel Hex or Motorola S, only memory regions not filled with hexadecimal FFs are translated, to keep the resulting file compact.

The Convert command can also decompress raw data from any number of complete 16-cluster units compressed by the NTFS file system* and can stretch packed 7-bit ASCII to readable 8-bit ASCII*.

Encriptación/Desencriptación

Specify a string consisting of 1-16 characters as the encryption/decryption key. The more characters you enter, the safer is the encryption. The key itself is not used for encryption and decryption, instead it is digested to the actual key. The key is not saved on your hard disk. If the corresponding security option is enabled, the key is stored in an encrypted state in the RAM as long as WinHex is running.

Se recomienda especificar una combinación de al menos 8 caracteres como clave de encriptación. The key is case-sensitive. No utilice palabras de ningún idioma, es mejor utilizar una combinación aleatorio de letras, signos de puntuación y dígitosl. Tenga en cuenta que las claves de encriptación distinguen entre mayúsculas y minúsculas. Recuerde que no podrá recuperar los

datos encriptados sin la correspondiente clave. Sea cuidadoso, ya que la clave de desencriptación que introduzca no será comprobada antes de desencriptar los datos.

Encryption algorithms available in WinHex:

• State-of-the-art 256-bit AES/Rijndael, in counter (CTR) mode. This encryption algorithm uses a 256-bit key that is digested with SHA-256 from the 512-bit concatenation of the SHA-256 of the key you specify and 256 bits of cryptographically sound random input (“salt”). The file is expanded by 48 bytes to accommodate the 256 bits of salt, and a randomized 128-bit initial counter.
WinHex allows you to encrypt not only an entire file, but also a block of data only. In that case you are warned, however, that no salt is used and no random initial counter is used, so you must not reuse your key to encrypt other data with the same encryption method. The size of the block is left unchanged.

• Pukall Cipher 1 (PC 1), usando una clave de 128 bits (=el digest de 128 bits de la clave que especifique).

## 8.4 Wiping and Initializing

For securely erasing (shredding) data, and also simply for filling files or disk sectors with certain byte values, WinHex offers the following options:

**Rellenar con valores hexadecimales:** Especifique 1, 2, 3, 4, 5, 6, 12, 15 o 16 valores hexadecimales de 2 caracteres, que se copiarán repetidamente en el bloque actual, el archivo entero o todos los sectores del disco, respectivamente.

**Rellenar con bytes aleatorios:** Especifique un intervalo decimal (de 0 a 255) para los números aleatorios, que serán copiados repetidamente en el bloque actual, el archivo completo o todos los sectores del disco, respectivamente. Los números aleatorios tendrán una distribución de Laplace.

En caso de que en todos los archivos abiertos esté o no esté definido un bloque, el comando puede opcionalmente aplicarse a todos esos archivos a la vez.

To maximize security, if you wish to totally wipe (sanitize) slack space, free space, unused NTFS records, or an entire media, you may want to apply more than one pass for overwriting disk space (up to three).

According to the Clearing and Sanitization Matrix, the standard outlined in the U.S. Department of Defense (DoD) 5220.22-M operating manual, method "c", a hard disk or floppy disk can be cleared by overwriting (once) all addressable locations with a single character. This is usually the hexadecimal value 0x00, but can be any other value. To sanitize hard disks according to method "d", overwrite all addressable locations with a character, its complement, then a random character, and verify. (This method is not approved by the DoD for sanitizing media that contain

top secret information.)

The "DoD" button configures WinHex for sanitization, such that it will first overwrite with 0x55 (binary 01010101), then with its complement (0xAA = 10101010), and finally with random byte values.

The "0x00" button configures WinHex for simple initialization, wiping once with zero bytes.

## 8.5 Clonar Disco

Copia un número definido de sectores desde un disco origen a uno de destino. Ambos discos deben tener el mismo tamaño de sector (or alternatively from a disk image file or to a disk image file). Para *duplicar* eficazmente una unidad (es decir, copiar todos los sectores de la unidad), active la opción correspondiente y los números de sectores correspondientes aparecerán automáticamente. El disdo de destino no debe ser de menor tamaño que el de origen.

Clonar Disco ofrece opciones que controlan el comportamiento del programa cuando se encuentran sectores defectuosos en el disco de origen.

- Por defecto, se le informará del error y se le preguntará si desea continuar o abortar la operación. "Log silencioso" crea un archivo de log de toda la operación in the folder for temporary files (filename “Cloning Log.txt”), incluyendo un informe de sectores que no han podido leerse (which cannot be copied), e impide que WinHex informe de cada error por separado. Esto puede resultar de utilidad, por ejemplo para forenses de ordenador.
- WinHex puede bien dejar el sector de destino correspondiente a un sector de origen dañado sin cambiar o puede rellenarlo con an ASCII pattern you specify (e.g. your initials, or something like "BAD "). Leave the pattern edit box blank to fill such sectors with *zero* bytes. BTW, the chosen pattern is also used to display a bad sector's contents in the disk editor.
- Bad sectors often occur in contiguous groups, and each attempt to read a bad sector usually takes a long time. You may have WinHex avoid such damaged disk areas. When a bad sector is encountered, WinHex can skip a number of subsequent sectors you specify (25 by default). This is useful if you wish to accelerate the cloning process and if you do not care about some actually readable sectors not making it to the clone.

El clonado estándar de discos no es la opción correcta si lo que desea hacer es duplicar un disco de una unidad removible (por ejemplo, un disquete) cuando dispone de una sola de estas unidades. Lo que debería hacer en tal caso es *una imagen del disco* (también podría denominarse clonación de disco "retardada"). Dicha imagen puede ser restaurada en un disco diferente. El efecto es idéntico a la clonación de disco.

When you specify a file named “dev-null” as the destination, the data will only be read and not copied anywhere (and you will be warned of this). This is useful if you are interested in the report about bad sectors, but do not wish to actually clone or image a disk.

You may try “simultaneous I/O” if the destination is not the same physical medium as the source.

Offers a chance to accelerate the cloning process by up to 30%.

Hay dos maneras de  hacer una imagen de un disco:

- El cuadro de diálogo de clonación de disco le permite copiar sectores de un disco en un archivo de imagen directamente, sin cabeceras y posteriormente recuperarlas de igual manera. Combinada con el modo "log silencioso", esta es la manera recomendada si desea crear un backup cuando existen sectores dañados en el disco de origen.
- For options like compression, hashing, and file splitting, please use the image and backup functionality. For easy recovery, a backup file includes information on its contents: sector numbers, source disk etc.

Clonar o hacer una imagen de un disco que contiene una instalación activa de Windows puede producir copias diferentes del original. En cualquier caso, por favor, asegúrese de que en el disco de origen ningún otro programa escriba durante la operación de clonación/backup/restauración o por el propio Windows. Es recomendable mover el directorio TEMP a una unidad diferente. El archivo de intercambio (swap) debería crearse también en una unidad distinta.

Make sure no other program or service can write to the partition you are going to clone. E.g. check for defragmentation tools running in the background and deactivate them for the duration of the cloning/backup/restoration. Under Windows NT/2000/XP it is recommended to unmount the partition as a logical drive/drive letter.

After cloning a logical NTFS drive, you may need to reboot your system or run “chkdsk /f” on the target drive in order to see the new contents in Windows (this clears all of Windows’ internal buffers).

Cloning or imaging with WinHex makes exact sector-wise, forensically sound copies, including all unused space and slack space. WinHex cannot dynamically change partition sizes or adapt to destination disks larger or smaller than the source disks. This can be done by tools like PartitionMagic.

In order to reduce the space a backup occupies as much as possible, you can initialize unused drive space before making the backup. This is because sectors that consist but of zero values barely increase the backup size when compression is enabled.

# 8.6 Images and Backups

El comando Crear Backup del menú Archivo allows to create a backup or image of the currently open logical drive, physical disk, or individual file. There are three possible output file formats, each with unique advantages.

| File format: | **WinHex Backup** | **Evidence File** | **Raw Image** |
|---|---|---|---|
| Filename extension: | .whx | .e01 | e.g. .dd |
| Interpretable as disk: | no | yes | yes |
| Splittable: | yes | yes | yes |

| | | | |
|---|---|---|---|
| Compressible: | yes | yes | no |
| Encryptable: | yes | yes | no |
| Optional hash: | integrated | integrated | separate |
| Optional description: | integrated | integrated | no |
| Range of sectors only: | yes | no | no |
| Applicable to files: | yes | no | no |
| Automated maintenance: | Backup Manager | no | no |
| Compatibility: | no | (yes) | yes |
| Required license: | none | forensic | personal |

The major advantage of evidence files and raw images is that they can be interpreted by WinHex like the original disks (with the command in the Specialist menu). This also makes them suitable for usage as evidence objects in your cases. This holds true for evidence files in particular because they can store an optional description and an integrated hash for later automated verification. Raw images have the benefit that they can be easily exchanged between various forensic tools. All output file formats support splitting into segments of a user-defined size. A segment size of 650 MB e.g. is suitable for archiving on CD-R. Evidence files are *required* to be split at 2025 MB at max.

The encryption algorithm optionally used in evidence files is exceptionally strong: 256-bit AES/Rijndael, in counter (CTR) mode. This allows for random read access within evidence files. This encryption algorithm uses a 256-bit key that is digested with SHA-256 from the 512-bit concatenation of the SHA-256 of the key you specify and 256 bits of cryptographically sound random input (“salt”), which is stored in the header of the evidence file. The 128-bit counter is randomized and incremented per encryption block. The block size of AES is 128 bits. An additional SHA-256 is stored in the header as well and used later to determine whether a password, specified by the user for decryption, is correct or not. The SHA-256 algorithm is applied to a concatenation of the salt, hash x, and hash y to compute this password verification hash, where hash x is the SHA-256 of the user-supplied password and hash y is the SHA-256 of the concatenation of the user-supplied password and hash x.

El algoritmo de encriptación in WinHex backups es „Pukall Cipher 1“ (PC 1), usando una clave de 128 bits que es en realidad un digest de 256 bits de la concatenación del digest de la clave de 128 bits que ha introducido y del digest de 128 bits de una entrada aleatoria. La entrada aleatoria es guardada en el archivo WHX para la posterior desencriptación.

Si decide que WinHex asigne automáticamente el nombre al archivo WHX, dicho archivo se guardará en la carpeta de backups (véase Opciones Generales), named with the next free “slot” according to the Backup Manager's naming conventions (“*xxx*.whx”) y estará disponible en el gestor de backup. Si especifica explícitamente una ruta y un nombre de archivo, podrá restaurar el backup más tarde utilizando el comando „Cargar Backup“, and in case of split backups WinHex will automatically append the volume number to the filenames

WinHex utiliza el algoritmo de compresión „Deflate“ que es parte de la popular librería de propósito general *zlib*. Este algoritmo consiste de una compresión LZ77 y una codificación Huffman. El factor de compresión es el mismo del ZIP. La documentación completa acerca del formato de archivo WHX está disponible en la página web de WinHex accesible en http://www.winhex.com.

## 8.7 Gestor de Backup

Muestra una lista de los backups WinHex previamente creados. Los artículos de dicha lista pueden listarse en orden cronológico o alfabético. Seleccione el backup que desea restaurar. Cuando la restauración haya finalizado, se mostrará el archivo original o el contenido del sector de disco.

You can restore the backup
- into a temporary file first such that you will still need to save it,
- escribirse directamente en el disco, o
- o en un archivo nuevo.

In the case of disk sectors puede asimismo especificar un disco de destino diferente o un número de sector de destino distinto. También puede extraer sólo una parte de los sectores contenidos en el backup. (De cualquier modo, los sectores del principio de un backup *comprimido* no pueden saltarse durante la restauración). Si el backup se guardó con la opción de checksum y/o digest, la autenticidad de los datos se comprobará antes de que los sectores sean escritos directamente en el disco.

El gestor de backup también le permite borrar backups que no necesitará más adelante. Los backups que se crearon para uso interno por el comando Deshacer pueden ser borrados automáticamente por WinHex (véase Opciones de Deshacer).

Los archivos de backup que son mantenidos por el gestor de backup están ubicados en la carpeta especificada en el cuadro de diálogo Opciones Generales. Sus nombres son „xxx.whx“ donde xxx es un número de identificación único de tres dígitos. Este número se muestra en la última columna de la lista del gestor de backup.

## 8.8 Gestor de Posiciones

El gestor de posiciones mantiene una lista de offsets de un archivo o un disco y sus correspondientes descripciones, also called *annotations*, also used for search hits. Navigating from one entry to the next is easy if you press Ctrl+Left and Ctrl+Right. Puede introducir nuevas posiciones así coo editar o borrar entradas ya existentes. Si un offset en especial en un archivo es importante porque tiene que editarlo en más de una ocasión, introdúzcalo en el gestor de posiciones. De este modo será mucho más fácil encontrarlo más adelante y no tendrá que recordarlo. Descriptions may be up to 8192 characters in size. Una descripción apropiada podría ser „¡El trapicheo de datos comienza aquí!“. Optionally all positions maintained by the Position Manager can be *highlighted* in the editor window in a unique color you specify, and their descriptions displayed in yellow tooltip windows when the mouse cursor is moved over them. You may also add or edit positions with the context menu of an edit window or by clicking the

middle mouse button in an edit window.

Pulse el botón derecho del ratón para ver un menú contextual in the Position Manager. Dicho menú proporciona comandos adicionales que le permitirán borrar, cargar y guardar posiciones (y guardar como HTML). Si la lista de posiciones ha cambiado, se guardará en el archivo *WinHex.pos* al salir de WinHex.

La documentación completa sobre el formato de archivo POS está disponible en la página web de WinHex accesible en http://www.winhex.com.

## 8.9 Intérprete de Datos

El Intérprete de Datos es una pequeña ventana que proporciona „servicios de traducción“ para los datos localizados en la posición actual del cursor. El diálogo opciones le permite especificar los tipos de datos a interpretar. Actualmente hay disponibles 9 tipos de datos enteros (by default in decimal notation, optionally hexadecimal or octal), el formato binario (8 bits de un byte), cuatro tipos reales, assembler opcodes (Intel®), y 6 tipos de fecha.

El Intérprete de Datos es también capaz de traducir todos los tipos de datos (excepto los assembler opcodes) de nuevo en valores hexadecimales. Pulse dos veces sobre un número de la ventana del Intérprete de Datos, introduzca un nuevo valor y **ENTER**. El Intérprete de Datos escribirá los correspondientes valores hexadecimales en la posición actual del cursor en la ventana de edición.

Pulse con el botón derecho sobre el intérprete de datos para acceder al menú contextual. Este le permitirá cambiar entre el formato de traducción little-endian y big-endian de los tipos de datos enteros y reales. You may also choose between decimal, octal, or hexadecimal integer representation. This plus the digit grouping can also be selected in the Data Interpreter Options dialog.

Consejos:

- Algunos valores hexadecimales no pueden convertirse en números reales. Para dichos valores, el Intérprete de Datos muestra el mensaje NAN (**n**ot **a n**umber).
- Algunos valores hexadecimales no pueden convertirse en fechas válidas. Los rangos de valores de los diferentes tipos de fecha son más o menos amplios.
- Hay redundancias en el juego de instrucciones Intel® que aparecen en el Intérprete de Datos como una duplicación tanto de opcodes como de comandos. Las intrucciones reales (en coma flotante) se muestran generalmente como F***.
- Puede encontrar más detalles en Intel® Architecture Software Developer’s Manual Volume 2: Instruction Set Reference, disponible en formato PDF en Internet.

# Apéndice A: Definición de Plantillas

## 1 Cabecera

La cabecera de una definición de plantilla tiene el siguiente formato:

```
template „titulo“
[description „descripción“]
[applies_to (file/disk/RAM)]
[fixed_start offset]
[sector-aligned]
[requires offset „valores hexadecimales“]
[big-endian]
[hexadecimal/octal]
[read-only]
[multiple [tamaño global fijado]]
// Ponga aquí sus comentarios sobre la plantilla.
begin
      declaración de variables
end
```

Los elementos entre corchetes son opcionales. El orden de los elementos es irrelevante. Las expresiones deben ir siempre entre comillas si contienen espacios. Los comentarios pueden aparecer en cualquier parte de la definición de plantilla. Los caracteres que siguen a una doble barra son ignorados por el intérprete (parser).

La sentencia `applies_to` debe ir seguida de una y sólo una de las palabras file, disk o RAM. WinHex mostrará un mensaje de aviso si intenta aplicar una plantilla a datos de origen distinto al especificado en la misma.

While by default templates start interpreting the data at the current cursor position when applied, an optional `fixed_start` statement ensures interpretation always starts at the specified absolute offset within the file or disk.

Si la plantilla se aplica a un disco, la palabara clave `sector-aligned` asegura que la interpretación de la plantilla comience al principio del sector actual sin tener en cuenta la posición exacta del cursor.

De forma similar a la sentencia `applies_to`, la sentencia `requires` permite a WinHex prevenir la aplicación errónea de una definición de plantilla a datos para los que no está preparada. Especifique un offset y una cadena de valores hexadecimales de longitud arbitraria que identifique los datos para los que la plantilla está pensada. Por ejemplo, un registro de arranque maestro (MBR) válido puede reconocerse por los valores hexadecimales 55 AA en el offset 0x1FE, y un archivo ejecutable por los valores 4D 5ª („MZ“) en el offset 0x0. Puede haber multitud de sentencias `applies_to` en la cabecera de la definición, y todas son tenidas en

cuenta.

La sentencia `big-endian` provoca que todos los enteros multi-byte y las variables booleanas en la definición de plantilla se lean y escriban en el orden big-endian (byte más significativo primero).

La sentencia `hexadecimal` provoca que todos los enteros en la definición de plantilla se lean y escriban en notación hexadecimal.

La sentencia `read-only` nos asegura que la plantilla puede utilizarse únicamente para examinar estructuras y no para manipularlas. Los controles de edición de la plantilla serán desactivados.

Si la sentencia `multiple` se escribe en la cabecera, WinHex permitirá que nos desplacemos a los registros de datos cercanos mientras se muestra la plantilla. Esto requiere que WinHex conozca el tamaño del citado registro. Si no se especifica como un parámetro para la sentencia `multiple`, WinHex asumirá que el tamaño total de la estructura de una plantilla (=registro) es la posición actual al final de la interpretación de la plantilla menos la posición de edición de base. Si la estructura tiene un tamaño variable, por ejemplo el tamaño de un array o parámetros de moviemiento que estén determinados dinámicamente por le valor de variables, WinHex no podrá examinar los registros de datos precedentes.

# 2 Cuerpo: Declaración de Variables

El cuerpo de una definición de plantilla consiste básicamente en declaraciones de variables, de manera similar a los lenguajes de programación. Una declaración tiene la forma

```
type „title“
```

donde `type` puede ser uno de lo siguientes:

- `int8, uint8 = byte, int16, uint16, int24, uint24, int32, uint32, int64,`
- `uint_flex`
- `binary,`
- `float = single, real, double, longdouble = extended,`
- `char, char16, string, string16,`
- `zstring, zstring16,`
- `boole8 = boolean, boole16, boole32`
- `hex,`
- `DOSDateTime, FileTime, OLEDateTime, SQLDateTime, UNIXDateTime = time_t, JavaDateTime`

`title` debe ir encluido entre comillas únicamente si contiene espacios en blanco. `title` no debe consistir únicamente en dígitos. WinHex no distingue entre caracteres en minúsculas y en mayúsculas. Como máximo pueden utilizarse 41 caracteres para identificar una variables.

`type` can be preceded by at most one member of each of the following modifier groups:

```
big-endian      little-endian
hexadecimal     decimal         octal
read-only       read-write
```

Los modificadores solo afectan a la varible que les sigue inmediatamente. Serán redundantes si aparecen en la cabecera también.

El número al final del nombre de un tipo indica el tamaño de cada variable (cadenas: de cada carácter) en bits. WinHex soporta caracteres y cadenas Unicode con `char16` y `string16`. De cualquier modo, los caracteres Unicode que no sean los 256 primeros caracteres equivalentes ANSI no están soportados. El máximo tamaño de una cadena que pueda ser editada mediante una plantilla es 8192 bytes.

Los tipos `string`, `string16`, y `hex` necesitan un parámetro adicional que especifique el número de elementos. Este parámetro puede ser una constante o una variable previamente declarada. Si es una constante, puede estar especificada en formato hexadecimal, es decir, será reconocida si está precedida por `0x`.

También puede declarar arrays de variables poniendo el tamaño del array entre corchetes a continuación del tipo o el título. Specify "unlimited" as the array size to make the template stop only when the end of file is encountered. Las siguientes líneas declaran una cadana ASCII de tamaño dinámico, cuya longitud depende de la variable precedente:

```
uint8       „len“
char[len]   „A string“
```

Podría conseguirse el mismo resultado con las siguientes declaraciones:

```
byte        „len“
string len „A string“
```

El carácter „~“ puede utilizarse para marcar el lugar donde más tarde se ubicará el número de elemento actual del array (véa más adelante). Esto no se aplica a arrays de variables `char`, ya que éstas son automáticamente convertidas en cadenas.

Numerical parameters of `string`, `string16`, and hex variables as well as array size expressions may be specified in mathematical notation. They will be processed by the integrated formula parser. Such expressions need to be enclosed in brackets. They must not contain space characters. They may make use of previously declared integer variables whose names do not contain space characters either. Supported operations are addition (+), subtraction (-), multiplication (*), integer division (/), modular division (%), bitwise AND (&), bitwise OR (|), and bitwise XOR (^). Valid mathematical expressions are for example (5*2+1) or (len1/(len2+4)). The result is always an integer and must be a positive number.

`zstring` y `zstring16` are null-terminated strings whose size is determined dynamically at run-time.

# 3 Cuerpo: Comandos Avanzados

Cuando se incluyen entre corchetes, varias declaraciones de variables constituyen un bloque que puede utilizarse repetidamente como una unidad. Observe que, en cualquier caso, ese bloque no debe ser *anidado* en la implementación actual. El carácter ~ puede utilizarse en el nombre de una variable para marcar el lugar donde posteriormente se ubicará el contador de la repetición actual. La `numbering` es opcional y define donde comenzará la cuenta (0 por defecto).

```
numbering  1
{
byte       „len“
string len „String No. ~“
}[10]
```

En este ejemplo los nombres de variable actual en la plantilla “String No. 1”, “String No. 2”, ..., “String No. 10”. Instead of a constant number of repetitions (10 in this example), you may also specify “unlimited”. In that case WinHex will repeat the block until the end of file is encountered. “ExitLoop” can be used to break out of a loop at any time.

"IfEqual" is useful for the comparison of two expressions. Operands can be either both numerical values, be it constant values in decimal notation, integer variables or a formulas, or byte sequences given as text or hex values which are compared byte by byte. ASCII string expressions must be enclosed in quotation marks, hex sequences must be preceded by a "0x" identifier. Formulas need to be enclosed in brackets.

```
{
byte       Value
IfEqual    Value 1
     ExitLoop
EndIf
} [10]
```

An "IfEqual" command block is terminated with an "EndIf" statement. If the compared expressions are equal, template interpretation continues after "IfEqual". Optionally, "IfEqual" can be followed by an "Else" statement. The template processor branches into the "Else" block if the expressions are not equal. "IfEqual" commands must not be nested. "IfGreater" is similar to "IfEqual". The condition is true if the first expression is greater than the second. Strings and hex values are compared lexicographically.

Para facilitar la lectura y el análisis de la plantilla, puede definir grupos de variables que estén separados por un espacio en blanco en el cuadro de diálogo:

```
section    „...Section Title...“
...
endsection
```

Las sentencias `section`, `endsection`, y `numbering` no avanzan la posición actual de los datos que van a ser interpretados.

Hay dos comandos que no declaran variables, pero se usan específicamente para cambiar la posición actual. Esto puede hacerse para saltarse datos irrelevantes (movimiento hacia delante) o para poder acceder a ciertas variables en más de una ocasión como tipos diferentes (movimiento hacia atrás). Utilice la sentencia `move n` para saltar `n` bytes desde la posición actual, donde `n` puede ser negativo. `goto n` se desplaza a la posición especificada de manera absoluta desde el principio de la interpretación de la plantilla (debe ser positivo).

El siguiente ejemplo demuestra como acceder a una variable tanto como un entero de 32 bits como una cadena de valores hexadecimales de cuatro partes:

```
int32      „Disk serial number (decimal)“
move -4
hex 4      „Disk serial number (hex)“
```

# 4 Cuerpo: Flexible Integer Variables

A special variable type supported by templates is `uint_flex`. This type allows to compose an unsigned integer value from various individual bits within a 32-bit (4-byte) range in an arbitrary order and is even more flexible than a so-called bit field in the C programming language.

uint_flex requires an additional parameter string in inverted commas that specifies exactly which bits are used in which order, separated by commas. The bit listed first becomes the most significant bit (high value bit) in the resulting integer, and it is not interpreted as a + or - indicator. The bit listed last becomes the least significant bit in the resulting integer.

The bits are counted starting with 0. Bit 0 is the bit that is the least significant bit of the 1st byte. Bit 31 is the most significant bit of the fourth byte. Thus, the definition is based on little-endian philosophy.

For example,
`uint_flex "15,14,13,12,11,10,9,8,7,6,5,4,3,2,1,0" "Standard 16-bit integer"`
is exactly the same as `uint16`, the common unsigned 16-bit integer variable.

`uint_flex "31,30,29,28,27,26,25,24,23,22,21,20,19,18,17,16,15,14,13,12, 11,10,9,8,7,6,5,4,3,2,1,0" "Standard 32-bit integer"`
is exactly the same as `uint32`, the common unsigned 32-bit integer variable.

The benefit of `uint_flex`, though, is that the number, the position, and the usage order of all

bits can be chosen arbitrarily. For example,
`uint_flex "7,15,23,31" "An unusual 4-bit integer"`
composes a 4-bit integer out of the respective most significant bits of each of the four bytes involved. If these four bytes happen to be
F0 A0 0F 0A =
**1**1110000 **1**0100000 **0**0001111 **0**0001010,
bit 7 is **1**, bit 15 is **1**, bit 23 is **0**, and bit 31 is **0**.
So the resulting uint_flex is **110**0 = **1***8 + **1***4 + **0***2 + **0***1 = 12.

# Apéndice B: Script Commands

Script commands are case-*in*sensitive. Comments may occur anywhere in a script file and must be preceded by two slashes. Parameters may be 255 characters long at most. Where in doubt because hex values, text strings (or even integer numbers) are accepted as parameters, you may use inverted commas (quotation marks) to enforce the interpretation of a parameter as text. Inverted commas are *required* if a text string or variable name contains one or more space characters, so that all characters between the inverted commas are recognized as constituting *one* parameter.

Whereever numerical parameters are expected (integer numbers), the integrated formula parser allows you to use mathematical expressions. Such expressions need to be enclosed in brackets. They must not contain space characters. They may make use of variables that can be interpreted as integer numbers. Supported operations are addition (+), subtraction (-), multiplication (*), integer division (/), modular division (%), bitwise AND (&), bitwise OR (|), and bitwise XOR (^). Valid mathematical expressions are for example (5*2+1), (MyVar1/(MyVar2+4)), or (-MyVar).

The following is a description of currently supported script commands, including example parameters.

**Create "D:\My File.txt" 1000**
Creates the specified file with an initial file size of 1000 bytes. If the file already exists, it is overwritten.

**Open "D:\My File.txt"**
**Open "D:*.txt"**
Opens the specified file(s). Specify "?" as the parameter to let the user select the file to open.

**Open C:**
**Open D:**
Opens the specified logical drive. Specify ":?" as the parameter to let the user select a logical drive or physical disk to open.

**Open 80h**
**Open 81h**
**Open 9Eh**

Opens the specified physical media. Floppy disk numbering starts with 00h, fixed and removable drive numbering with 80h, optical media numbering with 9Eh.

Optionally, you may pass a second parameter with the Open command that defines the edit mode in which to open the file or media ("in-place" or "read-only").

**CreateBackup**
Creates a backup of the active file in its current state.

**CreateBackupEx 0 100000 650 true "F:\My backup.whx"**
Creates a backup of the active disk, from sector 0 through sector 1,000,000. The backup file will be split automatically at a size of 650 MB. Compression is enabled (“true”). The output file is specified as the last parameter.
If the backup file should not be split, specify 0 as the third parameter. To disable compression, specify “false”. To have the Backup Manager automatically assign a filename and place the file in the folder for backup files, specify "" as the last parameter.

**Goto 0x128**
**Goto MyVariable**
Moves the current cursor position to the hexadecimal offset 0x128. Alternatively, an existing variable (up to 8 bytes large) can be interpreted as a numeric value, too.

**Move -100**
Moves the current cursor position 100 bytes back (decimal).

**Write "Test"**
**Write 0x0D0A**
**Write MyVariable**
Writes the four ASCII characters "Test" or the two hexadecimal values "0D0A" at the current position (in overwrite mode) and moves the current position forward accordingly (i.e. by 4 bytes). Can also write the contents of a variable specified as the parameter.

**Insert "Test"**
Functions just as the "Write" command, but in *insert* mode. Must only be used with *files*.

**Read MyVariable 10**
Reads the 10 bytes from the current position into a variable named "MyVariable". If this variable does not yet exist, it will be created. Up to 32 different variables allowed. Another way to create a variable is the Assign command.

**ReadLn MyVariable**
Reads from the current position into a variable named "MyVariable" until the next line break is encountered. If the variable already exists, its size will be adjusted accordingly.

**Close**
Closes the active window without saving.

**CloseAll**
Closes all windows without saving.

**Save**
Saves changes to the file or disk in the active window.

**SaveAs "C:\New Name.txt"**
Saves the file in the active window under the specified path. Specify "?" as the parameter to let the user select the destination.

**SaveAll**
Saves changes in all windows.

**Terminate**
Aborts script execution.

**Exit**
Terminates script execution and ends WinHex.

**ExitIfNoFilesOpen**
Aborts script execution if no files are already opened in WinHex.

**Block 100 200**
**Block "My Variable 1" "My Variable 2"**
Defines the block in the active window to run from offset 100 to offset 200 (decimal). Alternatively, existing variables (each up to 8 bytes large) can be interpreted as numeric values.

**Block1 0x100**
Defines the block beginning to be at the hexadecimal offset 0x100. A variable is allowed as the parameter as well.

**Block2 0x200**
Defines the block end to be at the hexadecimal offset 0x200. A variable is allowed as the parameter as well.

**Copy**
Copies the currently defined block into the clipboard. If no block is defined, it works as known from the Copy command in the Edit menu.

**Cut**
Cuts the currently defined block from the file and puts it into the clipboard.

**Remove**
Removes the currently defined block from the file.

**CopyIntoNewFile "D:\New File.dat"**
**CopyIntoNewFile "D:\File +MyVariable+.dat"**
Copies the currently defined block into the specified new file, without using the clipboard. If no block is defined, it works as known from the Copy command in the Edit menu. Can copy disk sectors as well as files. The new file will not be automatically opened in another edit window. Allows an unlimited number of "+" concatenations in the parameter. A variable name will be interpreted as an integer if not be larger than 2^24 (~16 Mio.). Useful for loops and file recovery.

**Paste**
Pastes the current clipboard contents at the current position in a file, without changing the current position.

**WriteClipboard**
Writes the current clipboard contents at the current position in a file or within disk sectors, without changing the current position and by overwriting the data at the current position.

**Convert *Param1 Param2***
Converts the data in the active file from one format into another one. Valid parameters are ANSI, IBM, EBCDIC, Binary, HexASCII, IntelHex, MotorolaS, Base64, UUCode, LowerCase, and UpperCase, in combinations as known from the conventional Convert menu command.

**AESEncrypt "My Password"**
Encrypts the active file or disk, or selected block thereof, with the specified key (up to 32 characters long) with AES.

**AESDecrypt "My Password"**
Decrypts the active file or disk.

**Find "John" [*MatchCase MatchWord Down Up BlockOnly SaveAllPos Unicode Wildcards*]**
**Find 0x1234 [*Down Up BlockOnly SaveAllPos Wildcards*]**
Searches in the active window for the name John or the hexadecimal values 0x1234, respectively, and stops at the first occurrence. Other parameters are opional. By default, WinHex searches the entire file/disk. The optional parameters work as known from usual WinHex search options.

**ReplaceAll "Jon" "Don" [*MatchCase MatchWord Down Up BlockOnly Unicode Wildcards*]**
**ReplaceAll 0x0A 0x0D0A [*Down Up BlockOnly Wildcards*]**
Replaces all occurrences of either a string or hexadecimal values in the active file with something else. Can only be applied to a disk if in in-place mode.

**IfFound**
A boolean value that depends on whether or not the last Find or ReplaceAll command was successful. Place commands that shall be executed if something was found after the IfFound command.

**IfEqual MyVariable "Hello World"**
**IfEqual 0x12345678 MyVariable**

**IfEqual MyVariable 1000**
**IfEqual MyVariable MyOtherVariable**
**IfEqual MyVariable (10*MyOtherVariable)**
Compares either two numerical integer values (each of them being a constant value, an integer variable or a mathematical expression) or two variables, ASCII strings, or hexadecimal values at the binary level. Comparing two objects at the binary with a different length always returns False as the result. If equal, the following commands will be executed. If conditions must not be nested.

**IfGreater MyVariable "Hello World"**
**IfGreater 0x12345678 MyVariable**
**IfGreater MyVariable 1000**
**IfGreater MyVariable MyOtherVariable**
**IfGreater MyVariable (10*MyOtherVariable)**
Accepts the same parameters as IfEqual. If the first one is greater than the second one, the following commands will be executed. If conditions must not be nested.

**Else**
May occur after IfFound or IfEqual. Place commands that shall be executed if nothing was found or if the compared objects are not equal after the Else command.

**EndIf**
Ends conditional command execution (after IfFound or IfEqual).

**ExitLoop**
Exits a loop. A loop is defined by braces. Closing braces may be followed by an integer number in square brackets, which determines the number of loops to execute. This is may also be a variable or the keyword "unlimited" (so the loop can only be terminated with an ExitLoop command). Loops must not be nested.

Example of a loop:
{ Write "Loop" }[10] will write the word "Loop" ten times.

**Label ContinueHere**
Creates a label named "ContinueHere"

**JumpTo ContinueHere**
Continues script execution with the command following that label.

**NextObj**
Switches cyclically to the next open window and makes it the "active" window. E.g. if 3 windows are open, and window #3 is active, NextObj will make #1 the active window.

**ForAllObjDo**
The following block of script commands (until **EndDo** occurs) will be applied to all open files and disks.

**CopyFile C:\A.dat D:\B.dat**
Copies the contents of C:\A.dat into the file D:\B.dat.

**MoveFile C:\A.dat D:\B.dat**
Moves the file C:\A.dat to D:\B.dat.

**DeleteFile C:\A.dat**
Surprisingly, deletes C:\A.dat.

**InitFreeSpace**
**InitSlackSpace**
Clears free space or slack on the current logical drive, respectively, using the currently set initialization settings. InitSlackSpace switches the drive temporarily to in-place mode, thus saving all pending changes.

**InitMFTRecords**
Clears unused MFT FILE records on the current logical drive if it is formatted with NTFS, using the currently set initialization settings. Simply does nothing on other file systems. The changes are written immediately to the disk.

**Assign MyVariable 12345**
**Assign MyVariable 0x0D0A**
**Assign MyVariable "I like WinHex"**
**Assign MyVariable MyOtherVariable**
Stores the specified integer number, binary data, ASCII text, or other variable's contents in a variable named "MyVariable". If this variable does not yet exist, it will be created. Up to 32 different variables allowed. Another way to create a variable is the Read command.

**SetVarSize MyVariable 1**
**SetVarSize MyVariable 4**
Explicitly sets the allocated memory size of a variable at a given time, in bytes. This can be useful e.g. for variables that hold integer values and that are the result of a calculation, if this value is to be written to a binary file with a fixed-length structure. Without SetVarSize, no assumption must be made about the size of the variable. For instance, the number 300 could be stored in any number of bytes larger than 1. If the new size set by SetVarSize is smaller than the old size, the allocated memory is truncated. If the new size is larger, the allocated memory is expanded. At any rate, the value of the persisting bytes is retained.

**GetUserInput MyVariable "Please enter your name:"**
Stores the ASCII text or binary data (0x...) specified by the user at script execution time (128 bytes at max.) in a variable named "MyVariable". The user is prompted by the message you provide as the second parameter. If this variable does not yet exist, it will be created. Other ways to create variables: Assign, Read.

**GetUserInputI MyIntegerVariable "Please enter your age:"**
Works like GetUserInput, but accepts and stores only integer numbers.

**Inc MyVariable**
Interprets the variable as an integer (if not larger than 8 bytes) and increments it by one. Useful for loops.

**Dec MyVariable**
Interprets the variable as an integer (if not larger than 8 bytes) and decrements it by one.

**IntToStr MyStr MyInt**
**IntToStr MyStr 12345**
Stores the decimal ASCII text representation of the integer number specified as the second parameter in a variable specified as the first parameter.

**StrToInt MyInt MyStr**
Stores the binary representation of the integer number specified as a decimal ASCII string in the second parameter in a variable specified as the first parameter.

**StrCat MyString MyString2**
**StrCat MyString ".txt"**
Appends one string to another. The second parameter may be a variable or a constant string. The first parameter must be a variable. The result will be saved in the variable specified by the first parameter and must not be longer than 255 characters.

**GetClusterAlloc MyStr**
May be applied to a logical volume. Retrieves a textual description of the current position's allocation, e.g. which file is stored in the current cluster, and saves that description in the specified variable.

**GetClusterAllocEx IntVar**
May be applied to a logical volume. Retrieves an integer value that indicated whether the cluster at the current position is allocated (1) or not (0), and saves that description in the specified variable.

**GetClusterSize IntVar**
May be applied to a logical volume. Retrieves the cluster size and saves that value in the specified integer variable.

**InterpretImageAsDisk**
Treats a raw image, Encase image or evidence file like the original physical disk or partition. Requires a specialist or forensic license.

**CalcHash HashType MyVariable**
**CalcHashEx HashType MyVariable**
Calculates a hash as known from the command in the Tools menu and stores it in the specified variable (which will be created if it does not yet exist). The HashType parameter must be one of the following: CS8, CS16, CS32, CS64, CRC16, CRC32, MD5, SHA-1, SHA-256, PSCHF.

CalcHashEx in addition displays the hash in a dialog window.

**MessageBox "Caution"**
Displays a message box with the text "Caution" and offers the user an OK and a Cancel button. Pressing the Cancel button will abort script execution.

**ExecuteScript "ScriptName"**
Executes another script from within a running script, at the current execution point, e.g. depending on a conditional statement. Calls to other scripts may be nested. When the called script is finished, execution of the original script will be resumed with the next command. This feature can help you structure your scripts more clearly.

**Turbo On**
**Turbo Off**
In turbo mode, most screen elements are not updated during script execution and you are not able to abort (e.g. by pressing Esc) or pause. This accelerates the script by up to 75% if a lot of simple commands such as Move and NextObj are executed in a loop.

**Debug**
All the following commands must be confirmed individually by the user.

**UseLogFile**
Error messages are written into the log file "Scripting.log" in the folder for temporary files. These messages are not shown in a message box that requires user interaction. Useful especially when running scripts on unattended remote computers.

**CurrentPos**
**GetSize**
**unlimited**
are keywords that act as a placeholders and may be used where numeric parameters are required. On script execution, CurrentPos stands for the current offset in the active file or disk window and GetSize for its size in bytes. unlimited actually stands for the number 2,147,483,647.

# Apéndice C: Disk Editor Q&A

**¿Cómo puedo acceder a sectores de unidades CD-RW?**

DirectCD and PacketCD must not be installed on the Windows system.

**¿Cómo puedo acceder a sectores de unidades CD-ROM y DVD bajo Windows 9x?**

Por favor, asegúrese de que se cumplen los siguientes requisitos:

1. Un driver para Windows de la unidad de CD-ROM/DVD debe estar instalado. Un driver para MS-DOS no sería suficiente.
2. La interfaz ASPI debe estar instalada. Es posible que tenga que copiar manualmente el archivo wnaspi32.dll en su directorio Windows\System. Encontrará el archivo en su CD de instalación de Windows. El programa shareware WinZip (disponible en http://www.winzip.com) es adecuado para extraer ficheros de los archivos CAB.
3. El driver del CD-ROM/DVD debe soportar la forma en que WinHex intenta leer los sectores. La mayoría de las modernas unidades SCSI y ATAPI son adecuadas.

**¿Cómo puedo hacer que WinHex detecte una PC Card ATA Flash Disk/Unidad PCMCIA como un disco físico bajo Windows 9x?**

Panel de Control de Windows -> Sistema -> Administrador de dispositivos -> Seleccione su unidad PCMCIA -> Clic en "Propiedades" -> Busque una opción con un nombre parecido a "Dispositivo Int 13h". La forma de presentar la casilla de activación puede variar entre las distintas versiones de Windows. Si es posible, active la opción y reinicie su ordenador.

# Apéndice D: Registro Maestro de Arranque

El Registro Maestro de Arranque (Master Boot Record – MBR en inglés) este ubicado en el principio físico del disco duro y puede editarse a través del editor de disco. Consiste de un codigo maestro de carga (446 bytes) seguido de cuatro registros de partición de estructura idéntica. Finalmente, la firma hexadecimal 55 AA completa un Registro Maestro de Arranque válido.

El formato de un registro de partición es el siguiente:

| Offset | Tamaño | Descripción |
|---|---|---|
| 0 | 8 bit | Un valor de 80 indica partición activa |
| 1 | 8 bit | Cabeza del inicio de la partición |
| 2 | 8 bit | Sector de inicio de la partición (bits 0-5) |
| 3 | 8 bit | Pista de inicio de la partición (bits 8,9 en sector de inicio como bits 6,7) |
| 4 | 8 bit | Indicador del sistema operativo |
| 5 | 8 bit | Cabeza de final de la partición |
| 6 | 8 bit | Sector de final de la partición (bits 0-5) |
| 7 | 8 bit | Pista de final de la partición (bits 8,9 en sector de final como bits 6,7) |
| 8 | 32 bit | Sectores que preceden a la partición |
| C | 32 bit | Longitud de la partición en sectores |

**Indicadores de sistema operativo:**
(hexadecimales, incomplete list)

| | |
|---|---|
| 00 | Empty partition-table entry |
| 01 | DOS FAT12 |

| | |
|---|---|
| 04 | DOS FAT16 (up to 32M) |
| 05 | DOS 3.3+ extended partition |
| 06 | DOS 3.31+ FAT16 (over 32M) |
| 07 | Windows NT NTFS, OS/2 HPFS, Advanced Unix |
| 08 | OS/2 v1.0-1.3, AIX bootable partition, SplitDrive |
| 09 | AIX data partition |
| 0A | OS/2 Boot Manager |
| 0B | Windows 95+ FAT32 |
| 0C | Windows 95+ FAT32 (using LBA-mode INT 13 extensions) |
| 0E | DOS FAT16 (over 32 MB, using INT 13 extensions) |
| 0F | Extended partition (using INT 13 extensions) |
| 17 | Hidden NTFS partition |
| 1B | Hidden Windows 95 FAT32 partition |
| 1C | Hidden Windows 95 FAT32 partition (using LBA-mode INT 13 extensions) |
| 1E | Hidden LBA VFAT partition |
| 50 | OnTrack Disk Manager, read-only partition |
| 51 | OnTrack Disk Manager, read/write partition |
| 81 | Linux |
| 82 | Linux Swap partition, Solaris (Unix) |
| 83 | Linux native file system (ext2fs/xiafs) |
| 85 | Linux EXT |
| 86 | FAT16 volume/stripe set (Windows NT) |
| 87 | HPFS fault-tolerant mirrored partition, NTFS volume/stripe set |
| BE | Solaris boot partition |
| C0 | DR-DOS/Novell DOS secured partition |
| C6 | Corrupted FAT16 volume/stripe set (Windows NT) |
| C7 | Corrupted NTFS volume/stripe set |
| F2 | DOS 3.3+ secondary partition |

# Apéndice E: Surplus Sectors

This term is used in WinHex in the following way:

Surplus sectors on a logical drive are those few sectors at the end that do not add to a full cluster and thus cannot be used by the OS (and thus by no conventional application program either).

Surplus sectors on a physical disk are those sectors at the end that are located outside the regular disk geometry scheme (because they do not add to a full cylinder/header/track entity), which is why they are usually not used by any partition or the operating system (or any conventional application program).

Surplus sectors have nothing to do with "bad" or damaged sectors or sectors a hard disk internally uses as a replacement for sectors found to be faulty.