NAME

mkisofs – Creates a hybrid ISO9660/JOLIET/HFS filesystem with optional Rock Ridge attributes.

SYNOPSIS

/usr/sbin/mkisofs [-abstract *FILE*] [-A *application_id*] [-allow-lowercase] [-allow-multidot] [-b *eltorito_boot_image*] [-eltorito-alt-boot] [-no-boot] [-no-emul-boot] [-biblio *FILE*] [-boot-load-seg *segment_address*]

[-boot-load-size *load-sectors*] [-boot-info-table] [-c *boot_catalog*] [-C *last_sess_start,next_sess_start*] [-cache-inodes] [-no-cache-inodes] [-check-oldnames] [-check-session *FILE*] [-copyright *FILE*] [-d]

[-D] [-dir-mode *mode*] [-exclude-list *file*] [-f] [-file-mode *mode*] [-force-rr] [-G *generic_boot_image*] [-gid *gid*] [-graft-point] [-gui]

[-hard-disk-boot] [-hide *glob*] [-hide-list *file*] [-hidden *glob*] [-hidden-list *file*] [-hide-joliet *glob*] [-hide-joliet-list *file*] [-hide-joliet-trans-tbl] [-hide-rr-moved] [-input-charset *charset*]

[-iso-level *level*] [-J] [-jcharset *charset*] [-l] [-L] [-log-file *log_file*] [-m *glob*] [-M {*path*|*device*}] [-max-iso9660-filenames] [-N] [-new-dir-mode *mode*]

[-no-bak] [-no-iso-translate] [-no-rr] [-no-split-symlink-components] [-no-split-symlink-fields] [-o *filename*] [-output-charset *charset*] [-pad] [-no–pad] [-path-list *file*] [-p *preparer_id*]

[-P *publisher_id*] [-print-size] [-quiet] [-r] [-R] [-relaxed-filenames] [-sort *sort_file*] [-sysid *ID*] [-T | -table-name *TABLE_NAME*]

[-U] [-ucs-level *level*] [-uid *uid*] [-use-fileversion] [-v] [-V *volid*] [-volset *ID*] [-volset-seqno #] [-volset-size #] [-x *path*] [-z]

[-apple-hfs] [-auto *AutoStart_file*] [-boot-hfs-file *driver_file*] [-cap] [-cluster-size *size*] [-dave] [-double] [-ethershare] [-exchange]

[-hfs-creator *CREATOR*] [-hfs-type *TYPE*] [-hide-hfs *glob*] [-hide-hfs-list *file*] [-hfs-volid *hfs_volid*] [-hfs-unlock] [-hfs-bless *folder_name*] [-icon-position] [-input-hfs-charset *charset*] [-macbin]

[-mac-name] [-magic *magic_file*] [-map *mapping_file*] [-netatalk]
[-no-desktop] [-o *filenamepathspec pathspec*] [-output-hfs-charset *charset*]
[-part] [-prep-boot *FILE*] [-probe] [-root-info *FILE*] [-sfm] [-sgi] [-single]
[-ushare] [-xinet]

OPTIONS

`-abstract` *`FILE`*	Specifies the abstract file name. This parameter can also be set in the file `.mkisofsrc` with ABST=filename. If specified in both places, the command-line version is used.
`-A` *`application_id`*	Specifies a text string that is written into the volume header. This describes the application that is to be on the disc. There is space on the disc for 128 characters of information. This parameter can also be set in the file `.mkisofsrc` with APPI=id. If specified in both places, the command-line version is used.
`-allow-lowercase`	This option allows lower case characters to appear in ISO9660 filenames. This violates the ISO9660 standard, but it works on some systems, so use with caution.
`-allow-multidot`	This options allows more than one dot to appear in ISO9660 filenames. (A leading dot is not affected by this option; it is allowed by using the `-L` option.) This violates the ISO9660 standard, but it works on many systems, so use with caution.
`-b` *`eltorito_boot_image`*	Specifies the path and filename of the boot image to be used when making an "El Torito" bootable CD. The pathname must be relative to the source path specified to

	`mkisofs`.This option is required to make an "El Torito" bootable CD. The boot image must be exactly the size of either a 1.2, 1.44, or a 2.88 megabyte floppy, and `mkisofs` will use this size when creating the output ISO9660 filesystem. It is assumed that the first 512 byte sector will be read from the boot image (emulating a normal floppy drive). This will work, for example, if the boot image is a LILO-based boot floppy.
`-eltorito-alt-boot`	Start with a new set of El Torito boot parameters. This allows more than one El Torito boot on a CD. A maximum of 63 El Torito boot entries may be put on a single CD.
`-no-boot`	Specifies that the created El Torito CD is marked as not bootable. The system will provide an emulated drive for the image, but it will boot off a standard boot device.
`-no-emul-boot`	Specifies that the boot image used to create El Torito bootable CDs is a ’no emulation’ image. The system will load and execute this image without performing any disk emulation.
`-biblio` *`FILE`*	Specifies the bibliographic file name. This parameter can also be set in the file `.mkisofsrc` with BIBLO=filename. If specified in both places, the command-line version is used.
`-boot-load-seg` *`segment_address`*	Specifies the load segment address of the boot image for no-emulation El Torito CDs.

`-boot-load-size` *load_sectors*	Specifies the number of virtual (512-byte) sectors to load in no-emulation mode. The default is to load the entire boot file. Some BIOSes may have problems if this is not a multiple of 4.
`-boot-info-table`	Specifies that a 56-byte table with information of the CD-ROM layout will be patched in at offset 8 in the boot file. If this option is given, the boot file is modified in the source filesystem, so make a copy if the boot file cannot be easily regenerated! See the EL TORITO BOOT INFO TABLE section for a description of this table.
`-c` *boot_catalog*	Specifies the path and filename of the boot catalog to be used when making an El Torito bootable CD. The pathn must be relative to the source path specified to `mkisofs`. This option is required to make a bootable CD. This file is inserted into the output tree and is not created in the source filesystem, so be sure the specified filename does not conflict with an existing file, as it will be excluded. Usually a name like "boot.catalog" is chosen.
`-C` *last_sess_start,next_sess_start*	This option is needed when `mkisofs` is used to create a CDextra or the image of a second session or a higher level session for a multi session disk. The option `-C` takes two numbers separated by a comma. The first number is the sector number of the first sector in the last session of the disk that should be appended to. The second number is the starting sector number of the new session. The expected pair of numbers may be retrieved by calling `cdrecord -msinfo`. If the `-C` option is

	used in conjunction with the `-M` option, `mkisofs` will create a filesystem image that is intended to be a continuation of the previous session. If the `-C` option is used without the `-M` option, `mkisofs` will create a filesystem image that is intended to be used for a second session on a CDextra. This is a multisession CD that holds audio data in the first session and a ISO9660 filesystem in the second session.
`-cache-inodes`	Caches inode and device numbers to find hard links to files. If `mkisofs` finds a hard link (a file with multiple names), then the file will only appear once on the CD. This saves space on the CD. The option `-cache-inodes` is the default on UNIX operating systems. Be careful when using this option on a filesystem without unique inode numbers as it may result in files containing the wrong content on CD.
`-no-cache-inodes`	Does not cache inode and device numbers. This option is needed whenever a filesystem does not have unique inode numbers. It is the default on `Cygwin`. As the Microsoft operating system that runs below `Cygwin` is not POSIX compliant, it does not have unique inode numbers. Cygwin creates fake inode numbers from a hash algorithm that is not 100% correct. If `mkisofs` would cache inodes on Cygwin, it would believe that some files are identical although they are not. The result in this case are files that contain the wrong content if a significant amount of different files (> ~5000) is in inside the tree that is to be archived. This does not happen when the `-no-cache-inodes` option is used, but the disadvantage is

	that `mkisofs` cannot detect hardlinks anymore and the resulting CD image may be larger than expected.
`-check-oldnames`	Checks all filenames imported from old session for compliance with `mkisofs` ISO9660 file-naming rules. If this option is not present, only names with a length > 31 characters are checked as these files are a violation of the ISO9660 standard.
`-check-session` *`FILE`*	Checks all old sessions for compliance with `mkisofs` ISO9660 file-naming rules. This is a high-level option that is a combination of the options: `-M"` *`FILE`* `"-C 0,0 -check- oldnames` For the parameter *`FILE`* see desciption of `-M` option.
`-copyright` *`FILE`*	Specifies the copyright filename. This parameter can also be set in the file `.mkisofsrc` with COPY=filename. If specified in both places, the command-line version is used.
`-d`	Omits the trailing period from files that do not have a period. This violates the ISO9660 standard, but it works on many systems. Use with caution.
`-D`	Does not use deep directory relocation. This violates the ISO9660 standard, but works on many systems. Use with caution.
`-dir-mode` *`mode`*	Overrides the mode of directories used to create the image to *`mode`*. Specifying this option automatically enables Rock Ridge extensions.

`-exclude-list` *`file`*	A file containing a list of *`globs`* to exclude. See `-hide` *`glob`* for a definition of glob.
`-f`	Follows symbolic links when generating the filesystem. If this option is not specified, symbolic links are entered using the Rock Ridge extensions, if enabled. Otherwise the file is ignored.
`-file-mode` *`mode`*	Overrides the mode of regular files used to create the image to *`mode`*. Specifying this option automatically enables Rock Ridge extensions.
`-force-rr`	Does not use the automatic Rock Ridge attribute recognition from previous sessions.
`-G` *`generic_boot_image`*	Specifies the path and filename of the generic boot image to be used when making a generic bootable CD. The `generic_boot_image` will be placed on the first 16 sectors of the CD. The first 16 sectors are the sectors that are located before the ISO9660 primary volume descriptor.
`-gid` *`gid`*	Overrides the gid read from the source files to the value of *`gid`*. Specifying this option automatically enables Rock Ridge extensions.
`-graft-point`	Allows graft points for filenames. If this option is used, all filenames are checked for graft points. The filename is divided at the first unescaped equal sign. All occurrences of \\ and = characters must

	be escaped with `\\` if `-graft-points` has been specified.
`-gui`	Switches the behaviour for a GUI. To make the output more verbose.
`-hard-disk-boot`	Specifies that the boot image used to create El Torito bootable CDs is a hard disk image. The hard disk image must begin with a master boot record that contains a single partition.
`-hide` *`glob`*	Hides a *`glob`* from being seen on the ISO9660 or Rock Ridge directory. Multiple globs may be hidden. If *`glob`* matches a directory, then the contents of that directory will be hidden. All the hidden files will still be written to the output CD image file. Should be used with the `-hide-joliet` option. See DESCRIPTION for a definition of *`glob`*.
`-hide-list` *`file`*	A file containing a list of *`globs`* to be hidden with the `-hide` option. See DESCRIPTION for a definition of *`glob`*.
`-hidden` *`glob`*	Adds the hidden ISO9660 directory attribute for *`glob`*. This attribute will prevent *`glob`* from being listed on DOS-based systems if the `/A` flag is not used for the listing. Multiple globs may be hidden. See DESCRIPTION for a definition of *`glob`*.
`-hidden-list` *`file`*	A file containing a list of *`globs`* to get the hidden attribute with the `-hidden` option. See DESCRIPTION for a definition of *`glob`*.

`-hide-joliet` *`glob`*	Hides *`glob`* from being seen on the Joliet directory. Multiple globs may be hidden. If *`glob`* matches a directory, then the contents of that directory will be hidden. All the hidden files will still be written to the output CD image file. Should be used with the `-hide` option. See DESCRIPTION for a definition of *`glob`*
`-hide-joliet-list` *`file`*	A file containing a list of *`globs`* to be hiddenwith the `-hide-joliet` option. See DESCRIPTION for a definition of *`glob`*
`-hide-joliet-trans-tbl`	Hides the `TRANS.TBL` files from the Joliet tree. These files usually don't make sense in the Joliet World as they list the real name and the ISO9660 name which may both be different from the Joliet name.
`-hide-rr-moved`	Rename the directory `RR_MOVED` to `.rr_moved` in the Rock Ridge tree. It seems to be impossible to completely hide the `RR_MOVED` directory from the Rock Ridge tree. This option only makes the visible tree better to understand for people who don't know what this directory is for. If you do not need the `RR_MOVED` directory, use the `-D` option. Note that in case that the `-D` option has been specified, the resulting filesystem is not ISO9660 level-1 compliant and will not be readable on MS-DOS. See the `NOTES` section for more information on the `RR_MOVED` directory.
`-input-charset` *`charset`*	Defines the characters used in local file names. To get a list of valid charset names, call `mkisofs` —`-input-charset` help. To get a 1:1 mapping, you may use

	`-default` as charset name. The default initial values are *`cp437`* on DOS-based systems and *`iso8859-1`* on all other systems. See the `CHARACTER SETS` section for more details.
`-iso-level` *`level`*	Sets the ISO9660 conformance level. Valid numbers are 1, 2, 3. With level 1, files may consist only of one section and filenames are restricted to 8.3 characters. With level 2, files may consist only of one section. With level 3, no restrictions apply. With all ISO9660 levels all filenames are restricted to upper case letters, numbers and the underscore (_). The maximum filename length is restricted to 31 characters; the directory nesting level is restricted to 8; and the maximum path length is limited to 255 characters.
`-J`	Generates Joliet directory records in addition to regular ISO9660 file names. This is useful when the discs are to be used on Windows-NT or Windows-95 machines. The Joliet filenames are specified in Unicode and each path component can be up to 64 Unicode characters long. Note that Joliet is not standard. CDs that use only Joliet extensions but no standard Rock Ridge extensions generally can only be used on Microsoft Win32 systems. Furthermore, the fact that the filenames are limited to 64 characters and the fact that Joliet uses the UTF-16 coding for Unicode characters causes interoperability problems.
`-j charset` *`charset`*	Same as using `- input-charset` -*charset* and `-J` options. See `CHARACTER SETS` section for more details.

`-l`	Allows full 31 character filenames. Normally the ISO9660 filename will be in an 8.3 format, which is compatible with MS-DOS, even though the ISO9660 standard allows filenames of up to 31 characters. If you use this option, the disc may be difficult to use on a MS-DOS system. Use with caution.
`-L`	Allows ISO9660 filenames to begin with a period. Usually, a leading dot is replaced with an underscore in order to maintain MS-DOS compatibility. This violates the ISO9660 standard, but works on many systems. Use with caution.
`-log-file` *`log_file`*	Redirects all error, warning and informational messages to *`log_file`* instead of the standard error.
`-m` *`glob`*	Excludes *`glob`* from being written to CD-ROM. Technically, *`glob`* is matched against the *`d->d_name`* part of the directory entry. Multiple globs may be excluded. NOTE: The `-m` and `-x` options both work the same and use filename globbing. A file is excluded if either the last component matches or the whole path matches.
`-M` *`{path|device}`*	Specifies path to existing ISO9660 image to be merged. The alternate form takes a SCSI device specifier that uses the same syntax as the `dev=` parameter of `cdrecord`. The output of `mkisofs` will be a new session which gets written to the end of the image specified in the `-M` option. Typically this requires multi-session capability for the recorder and CD-ROM drive that you are

	attempting to write this image to. This option may only be used in conjunction with the `-C` option.
`-max-iso9660-filenames`	Allows 37 chars in ISO9660 filenames. This option forces the `-N` option as the extra name space is taken from the space reserved for ISO-9660 version numbers. This violates the ISO9660 standard, but works on many systems. Although a conforming application needs to provide a buffer space of at least 37 characters, disks created with this option may cause a buffer overflow in the reading operating system. Use with extreme care.
`-N`	Omits version numbers from ISO9660 file names. This violates the ISO9660 standard. Use with caution.
`-new-dir-mode` *`mode`*	Mode to use when creating new directories in the iso filesystem. The default mode is 0555.
`-no-bak`	Does not include backup files files on the ISO9660 filesystem. If the `-no-bak` option is specified, files that contain the characters ~ or # or end in `.bak` will not be included.
`-no-iso-translate`	Does not translate the characters # and ~ which are invalid for ISO9660 filenames. These characters are often used by Microsoft systems. This violates the ISO9660 standard, but works on many systems. Use with caution.
`-no-rr`	Does not use the Rock Ridge attributes from previous sessions. This may help to

	avoid trouble when `mkisofs` finds illegal Rock Ridge signatures on an old session.
`-no-split-symlink-components`	Does not split the SL components, but begins a new Continuation Area (CE) instead. This may waste some space.
`-no-split-symlink-fields`	Does not split the SL fields, but begin a new Continuation Area instead. This may waste some space.
`-o` *`filename`*	Specifies the name of the file to which the ISO9660 filesystem image should be written. This can be a disk file, a tape drive, or it can correspond directly to the device name of the optical disc writer. If not specified, `stdout` is used. Note that the output can also be a block special device for a regular disk drive, in which case the disk partition can be mounted and examined to ensure that the premastering was done correctly.
`-output-charset` *`charset`*	Outputs a character set that defines the characters that will be used in Rock Ridge file names. The default is the input charactset. See `CHARACTER SETS` section below for more details.
`-pad`	Pads the end of the ISO9660 track by 16 sectors (32kilobytes). If the total size then is not a multiple of 16 sectors, the needed number of sectors is added. If the option B is used, then there is a second padding at the end of the boot partitions. The padding is needed as many operating systems (e.g. Linux) implement read-ahead bugs in their filesystem I/O. These bugs result in read errors on one or more files that are located at the end of a

	track. They are usually present when the CD is written in Track at Once mode. To avoid problems with I/O error on the last file on the filesystem. The `-pad` option is the default.
`-no–pad`	Does not pad the end of the ISO9660 by 16 sectors (32kilobytes).
`-path-list` *`file`*	A file containing a list of *`pathspec`* directories and filenames added to the ISO9660 filesystem. This list of pathspecs is processed after any that appear on the command line. If the argument is –, then the list is read from the standard input. There must be at least one *`pathspec`* given on the command line as well.
`-p` *`preparer_id`*	Specifies a text string that is written into the volume header. This should describe the preparer of the CD-ROM, usually with a mailing address and phone number. There is space on the disc for 128 characters of information. This parameter can also be set in the file `.mkisofsrc` with `PREP=`. If specified in both places, the command line entry is used.
`-P` *`publisher_id`*	Specifies a text string that is written into the volume header. This should describe the publisher of the CD-ROM, usually with a mailing address and phone number. There is space on the disc for 128 characters of information. This parameter can also be set in the file `.mkisofsrc` with `PUBL=`. If specified in both places, the command line entry is used.
`-print-size`	Prints estimated filesystem size and exits. This option is needed for Disk At

	Once mode and with some CD-R drives when piping directly into `cdrecord.` In this case, the size of the filesystem must be known before the actual CD-creation is done. The option `-print-size` gets this size from a "dry-run" before the CD is actually written.
`-quiet`	Makes `mkisofs` even less verbose. No progress output is provided.
`-R`	Generates System Use Sharing Protocol records (SUSP) and Rock Ridge (RR) records using the RR protocol to further describe the files on the ISO9660 filesystem.
`-r`	Similar to the `-R` option, but file ownership and modes are set to more useful values. The`- uid` and `-gid` are set to zero, because they are usually only useful on the author's system, and not useful to the client. All the file read bits are set true, so that files and directories are globally readable on the client. If any execute bit is set for a file, all execute bits are set, so that executables are globally executable on the client. If any search bit is set for a directory, all search bits are set, so that directories are globally searchable on the client. All write bits are cleared, because the CD-ROM will be mounted read-only. Any special mode bits that are set, clear them, because file locks are not useful on a read-only file system, and set-id bits are not desirable for `-uid` 0 or `-gid` 0. When used on Win32, the execute bit is set on *all* files. This is a result of the lack of file permissions on Win32 and the Cygwin POSIX emulation

	layer. See also `-uid`, `-gid`, `-dir-mode`,`-file-mode` and `-new-dir-mode`.
`-relaxed-filenames`	Allows ISO9660 filenames to include digits, uppercase characters and all other 7 bit ASCII characters. This violates the ISO9660 standard, but works on many systems. Use with caution.
`-sort` *`sort_file`*	Sorts file locations on the media. Sorting is controlled by a file that contains pairs of filenames and sorting offset weighting. The higher the weighting, the closer to the beginning of the media the file is located. There can be only one space or tab character between the filename and the weight, and the weight must be the last characters on a line. The filename includes all the characters up to, but not including the last space or tab character on a line. This allows for space characters to be in or at the end of a filename.
`-sysid` *`ID`*	Specifies the system ID. This parameter can also be set in the file `.mkisofsrc` with SYSI=system_id. If specified in both places, the command line version is used.
`-T`	Generates a file `TRANS.TBL` in each directory on the CD-ROM, which can be used on non-Rock Ridge capable systems to establish the correct file names. The file also contains information that indicates the major and minor numbers for block and character devices, and each symlink has the name of the link file given.
`-table-name` *`TABLE_NAME`*	Specifies a translation table file name to be used by the `-T` option. If you do not

	specify a name, `TRANS.TBL` is used. If you are creating a multi-session image you must use the same name as in the previous session.
`-U`	Allows untranslated filenames, completely violating the ISO9660 standards. Forces on the –d, –l, –L, –N, –relaxed-filenames, –allow-lowercase, –allow-multidot and –no-iso-translate options. It allows more than one `.` character in the filename, as well as mixed case filenames. Use with extreme caution.
`-ucs-level` *`level`*	Sets the Unicode conformance level in the Joliet SVD. Valid values are 1, 2 or 3. The default level is 3.
`-uid` *`uid`*	Overrides the uid read from the source files to the value of *`uid`*. Specifying this option automatically enables Rock Ridge extensions.
`-use-fileversion`	Allows `mkisofs` to use file version numbers from the filesystem. If the option is not specified, `mkisofs` creates a version of 1 for all files. File versions are strings in the range from 1 to 32767. This option is the default on VMS.
`-v`	Verbose execution. If given twice on the command line, extra debug information is printed.
`-V` *`volid`*	Specifies the volume ID (volume name or label) to be written into the master block. This parameter can also be set in the file `.mkisofsrc` with `VOLI=id`. If specified

	in both places, the command line version is used. Note that if you assign a volume ID, this is the name that is assigned to the disc on a Microsoft Win32 platform.
`-volset` *`ID`*	Specifies the volume set ID. This parameter can also be set in the file `.mkisofsrc` with `VOLS=volset_id`. If specified in both places, the command line version is used.
`-volset-seqno` *`#`*	Sets the volume set sequence number to the number specified. The volume set sequence number is the index number of the current CD in a CD set. The option `-volset-size` must be specified before `-volset-seqno` on each command line.
`-volset-size` *`#`*	Sets the volume set size to #. The volume set size is the number of CD's that are in a CD set. The `-volset-size` option may be used to create CD's that are part of, for example, a Operation System installation set of CD's. The option `-volset-size` must be specified before `-volset-seqno` on the command line.
`-x` *`path`*	Excludes *`path`* from being written to CD-ROM. *`path`* is the complete pathname that results from concatenating the pathname given as command line argument and the path relative to this directory. Multiple paths may be excluded. Example: `mkisofs –o cd –x /local/dir1 –x /local/dir2 /local`. See the `-m` option for more information.
`-z`	Generates special System Use Sharing Protocol (SUSP) records for transparently

compressed files. This is only of use and interest for hosts that support transparent decompression. This is an experimental feature, and no hosts yet support this, but there are ALPHA patches for Linux that can make use of this feature.

HFS OPTIONS

`-apple`	Creates an ISO9660 CD with Apple's extensions. Similar to the `-hfs` option, except that the Apple Extensions to ISO9660 are added instead of creating an HFS hybrid volume.
`-auto`*`AutoStart_file`*	Makes the HFS CD use the QuickTime 2.0 Autostart feature to launch an application or document. The given filename must be the name of a document or application located at the top level of the CD. The filename must be less than 12 alphanumeric characters.
`-boot-hfs-file` *`driver_file`*	Installs the `driver_file` to make the CD bootable on a Macintosh. See the HFS BOOT DRIVER section.
`-cap`	Looks for AUFS CAP Macintosh files. Searches for CAP Apple/UNIX file formats only. Searching for the other possible Apple/UNIX file formats is disabled, unless other double-dash options are given.
`-cluster-size` *`size`*	Sets the size in bytes of the cluster or allocation units of PC Exchange files. Implies the `-exchange` option. See HFS MACINTOSH FILE FORMATS.

`-dave`	Looks for Thursby Software Systems DAVE Macintosh files.
`-double`	Looks for AppleDouble Macintosh files.
`-ethershare`	Looks for Helios EtherShare Macintosh files.
`-exchange`	Looks for PC Exchange Macintosh files.
`-hfs`	Creates an ISO9660/HFS hybrid CD. This option should be used in conjunction with the `-map`, `-magic` and the various *`double dash`* options given below.
`-hfs-creator` *`CREATOR`*	Sets the default CREATOR for all files. Must be exactly 4 characters. See HFS CREATOR/TYPE for more details.
`-hfs-type` *`TYPE`*	Sets the default TYPE for all files. Must be exactly 4 characters. See HFS CREATOR/TYPE for more details.
`-hide-hfs` *`glob`*	Hide *`glob`* from the HFS volume. The file or directory will still exist in the ISO9660 and/or Joliet directory.
`-hide-hfs-list` *`file`*	A file containing a list of *`globs`* to be hidden.
`-hfs-volid` *`hfs_volid`*	Volume name for the HFS partition. This is the name that is assigned to the disc on a Macintosh and replaces the *`volid`* used with the `-V` option
`-hfs-unlock`	By default, `mkisofs` will create an HFS volume that is *`locked`*. This option leaves the volume unlocked so that other applications (that is `hfsutils`)

	can modify the volume. See HFS PROBLEMS/LIMITATIONS below for warnings about using this option.
`-hfs-bless` *`folder_name`*	"Bless" the given directory (folder). This is usually the system folder and is used in creating HFS bootable CDs. The name of the directory must be the whole path name as `mkisofs` sees it; that is, if the given path specification is `./cddata` and the required folder is called `System Folder`, then the whole path name is `"./cddata/System Folder"`Use quotes if the name contains spaces.
`-icon-position`	Uses the icon position information, if it exists, from the Apple/UNIX file. The icons will appear in the same position as they would on a Macintosh desktop. Folder location and size on screen, its scroll positions, folder View (view as Icons, Small Icons, etc.) are also preserved.
`-input-hfs-charset` *`charset`*	Inputs charset that defines the characters used in HFS file names when used with the `-mac-name` option. The default charset is *`cp10000`* (Mac Roman). See CHARACTER SETS and HFS MACINTOSH FILE NAMES for more details.
`-macbin`	Looks for MacBinary Macintosh files.
`-mac-name`	Uses the HFS filename as the starting point for the ISO9660, Joliet and Rock Ridge file names. See HFS MACINTOSH FILE NAMES for more information.

`-magic` *`magic_file`*	Uses the *`magic_file`* to set the CREATOR and TYPE information for a file based on the file's magic number. The *`magic_file`* is only used if a file is not one of the known Apple/UNIX file formats, or the filename extension has not been mapped using the `-map` option. See HFS CREATOR/TYPE for more details.
`-map` *`mapping_file`*	Uses the *`mapping_file`* to set the CREATOR and TYPE information for a file based on the filename's extension. A filename is mapped only if it is not one of the known Apple/UNIX file formats. See HFS CREATOR/TYPE below.
`-netatalk`	Looks for NETATALK Macintosh files.
`-no-desktop`	Does not create empty Desktop files. New HFS Desktop files are created when the CD is used on a Macintosh and stored in the system folder By default, empty Desktop files are added to the HFS volume.
`-output-hfs-charset` *`charset`*	Outputs charset that defines the characters that will be used in the HFS file names. Defaults to the input charset. See CHARACTER SETS for more details.
`-part`	Generates an HFS partition table. By default, no partition table is generated, but some older Macintosh CD-ROM drivers need an HFS partition table on the CD-ROM to be able to recognize a hybrid CD-ROM.
`-prep-boot` *`FILE`*	PReP boot image file. Up to 4 are allowed. See README.prep_boot (Alpha)

`-probe`	Searches the contents of files for all the known Apple/UNIX file formats. See HFS MACINTOSH FILE FORMATS for more about these formats. However, the only way to check for `MacBinary` and `AppleSingle` files is to open and read them. Therefore this option may increase processing time. It is better to use one or more `double dash` options given below if the Apple/UNIX formats in use are known.
`-root-info` *`file`*	Sets the location, size on screen, scroll positions, folder View, and so on, for the root folder of an HFS volume. See README.rootinfo for more information. (Alpha)
`-sfm`	Looks for Microsoft's Services for Macintosh files (NT only) (Alpha)
`-sgi`	Looks for SGI Macintosh files.
`-single`	Looks for AppleSingle Macintosh files.
`-ushare`	Looks for IPT UShare Macintosh files.
`-xinet`	Looks for XINET Macintosh files.

DESCRIPTION

Use the `mkisofs` pre-mastering program to generate an ISO9660/JOLIET/HFS hybrid filesystem. The Hierarchical File System (HFS) is the native file system used on Macintosh computers. The image of this filesystem will be written to CD-ROM.

The `mkisofs` command generates the System Use Sharing Protocol records (SUSP) specified by the Rock Ridge Interchange Protocol (RR). This is used to further describe the files in the ISO9660 filesystem to a UNIX host, and it provides information such as longer filenames, uid/gid, posix permissions, symbolic links, block and character devices.

If Joliet or HFS hybrid command line options are specified, `mkisofs` will create additional filesystem metadata for Joliet or HFS. The file content in this case refers to the same data blocks on the media. It will generate a pure ISO9660 filesystem unless the Joliet or HFS hybrid command line options are given.

The `mkisofs` command can generate a *`true`* or *`shared`* HFS hybrid filesystem. The Hierarchical File System (HFS) is the native file system used on Macintosh computers. The same files are seen as HFS files when accessed from a Macintosh and as ISO9660 files when accessed from other machines.

As an alternative, `mkisofs` can generate the Apple Entensions to ISO9660 for each file. These extensions provide each file with CREATOR, TYPE and certain Finder Flags when accessed from a Macintosh. See HFS MACINTOSH FILE FORMATS.

A *`glob`* is a shell wild-card-style pattern that must match any part of the filename or path. The pathname does not include a trailing / character. For example, `mkisofs –o rom –m *.o –m core –m foobar` would exclude all files ending in `.o`, called `core` or `foobar` to be copied to CD-ROM. Note that if you had a directory called `foobar` it too (and of course all its descendants) would be excluded.

Multiple globs may be excluded. For example, `mkisofs –o rom –hfs –hide-hfs *.o –hide-hfs foobar` would exclude all files ending in `.o` or called `foobar` from the HFS volume. Note that if you had a directory called `foobar` it too (and of course all its descendants) would be excluded. The *`glob`* can also be a path name relative to the source directories given on the command line. For example, `mkisofs –o rom –hfs –hide-hfs src/html src` would exclude just the file or directory called `html` from the `src` directory. Any other file or directory called `html` in the tree would not be excluded. Should be used with the `-hide` and/or `-hide-joliet` options. In order to match a directory name, make sure the pathname does not include a trailing / character.

The `mkisofs` command takes a snapshot of a given directory tree and generates a binary image which corresponds to an ISO9660 or HFS filesystem when written to a block device.

Each file written to the ISO9660 filesystem must have a filename in the 8.3 format (8 characters, period, 3 characters, all upper case), even if Rock Ridge is in use. This filename is used on systems that are not able to make use of the Rock Ridge extensions (such as MS-DOS), and each filename in each directory must be different from the other filenames in the same directory. The `mkisofs` command generally tries to form correct names by forcing the UNIX filename to upper case and truncating as required, but often this yields unsatisfactory results when there are cases where the truncated names are not all unique. The `mkisofs`command assigns weightings to each filename, and if two names that are otherwise the same are found, the name with the lower priority is renamed to have a 3 digit number as an extension (where the number is guaranteed to be unique). An example of this would be the files foo.bar and foo.bar.~1~ - the file foo.bar.~1~ would be written as FOO000.BAR;1 and the file foo.bar would be written as FOO.BAR;1

When used with various HFS options, `mkisofs` will attempt to recognise files stored in a number of Apple/UNIX file formats and will copy the data and resource forks as well as any relevant finder information. See HFS MACINTOSH FILE FORMATS for more about formats `mkisofs` supports.

Note that `mkisofs` is not designed to communicate with the CD burner directly. Most burners have proprietary command sets that vary from one manufacturer to another.

The `cdrecord` utility is capable of burning an actual disc.

Most CD writers are very particular about timing. Once you start to burn a disc, you cannot let the buffer empty before you are done, or you will end up with a corrupt disc. Thus it is critical that you be able to maintain an uninterrupted data stream for the entire time that the disc is being written.

The `pathspec` is the path of the directory tree to be copied into the ISO9660 filesystem. Multiple paths can be specified, and `mkisofs` will merge the files found in all of the specified path components to form the CD-ROM image.

Specify the `-graft-points`option to graft the paths at points other than the root directory. You can graft files or directories onto the CD-ROM image with names different from what they have in the source filesystem.

For example, you want to include a local file `../old.lis`, in the CD-ROM image. Issue the command, `foo/bar/=../old.lis`. This includes the file `old.lis` in the CD-ROM image at `/foo/bar/old.lis`. If you enter the command as `foo/bar/xxx=../old.lis`then `mkisofs` puts the file `old.lis` in the CD-ROM image at `/foo/bar/xxx`.

The same sort of syntax can be used with directories as well. The `mkisofs` command creates any directories required such that the graft points exist on the CD-ROM image. The directories do not need to appear in one of the paths. By default, any directories that are created on the fly like this will have permissions 0555 and appear to be owned by the person running `mkisofs`. If you wish other permissions or owners of the intermediate directories, see the `-uid`,`- gid`, `-dir-mode`, `-file-mode` and `-new-dir-mode` options.

The `mkisofs` command will also run on Win9X/NT4 machines when compiled with Cygnus' cygwin (available from http://sourceware.cygnus.com/cygwin/). Therefore most references in this man page to `UNIX` can be replaced with `Win32`.

CHARACTER SETS

The `mkisofs` command processes file names in a POSIX compliant way as strings of 8-bit characters.

Modern UNIX operating systems use `UTF-8` coding for filenames. This coding allows to use the complete Unicode code set. Each 32-bit character is represented by one or more 8-bit characters.

For all non `UTF-8` coded operating systems, the actual character that each byte represents depends on the *`character set`* or *`codepage`* (which is the name used by Microsoft) used by the local operating system in use.

Because all operating systems and applications do not use the Unicode character set as the basis for file names in a unique way, it may be necessary to specify which character set your file names use and in which character set the file names should appear on the CD.

There are four options to specify the character sets:

`-input-charset`	Defines the local character set you are using on your host machine. Any character set conversions that take place will use this character set as the staring point. The default input character sets are *`cp437`* on DOS-based systems and *`iso8859-1`* on all other systems. If the `-J` option is given, then the Unicode equivalents of the input character set will be used in the Joliet directory. Using the

	`-jcharset` option is the same as using the `-input-charset` and `-J` options.
`-output-charset`	Defines the character set that will be used for the Rock Ridge names on the CD. Default is the input character set. This option is useful on a non-UNIX platform, for example, using `mkisofs` on a Microsoft Win32 machine to create Rock Ridge CDs.
`-input-hfs-charset`	Defines the HFS character set used for HFS file names decoded from any of the various Apple/UNIX file formats. This option is only useful when used with the `-mac-name` option. See HFS MACINTOSH FILE NAMES for more information. Default is *`cp10000`* (Mac Roman).
`-output-hfs-charset`	Defines the HFS character set used to create HFS file names from the input character set in use. In most cases this is the character set given with the `-input-charset` option. Default is the input HFS character set.

There are a number of character sets built in to `mkisofs`. To get a listing, use `mkisofs -input-charset` help.

Additional character sets can be read from a file for any of the character set options by giving a filename as the argument to the options. The given file will only be read if its name does not match one of the built-in character sets.

The format of the character set files is the same as the mapping files available from http://www.unicode.org/Public/MAPPINGS The format of these files is: Column #1 is the input byte code (in hex as 0xXX) Column #2 is the Unicode (in hex as 0xXXXX) Rest of the line is ignored. Any blank line, line without two (or more) columns in the above format or comment lines (starting with the # character) are ignored without any warnings. Any missing input code is mapped to Unicode character 0x0000.

Note that there is no support for 16 bit UNICODE (UTF-16) or 32 bit UNICODE (UTF-32) coding because this coding is not POSIX compliant. There should be support for UTF-8 UNICODE coding which is compatible to POSIX filenames and supported by moder UNIX implementations such as Solaris.

A 1:1 character set mapping can be defined by using the keyword *`default`* as the argument to any of the character set options. This is the behaviour of older (v1.12) versions of `mkisofs`.

The ISO9660 file names generated from the input filenames are not converted from the input character set. Any character that `mkisofs` can not convert will be replaced with a _ character.

HFS CREATOR/TYPE

A Macintosh file has two properties that define which application created the file, the *`CREATOR`* and what data the file contains, the *`TYPE`*. Both are 4 letter strings.

In summary, for all files, the default CREATOR is ’unix’ and the default TYPE is ’TEXT’. These can be changed by using entries in the `.mkisofsrc` file or by using the `-hfs-creator` and/or `-hfs-type` options.

If the a file is in one of the known Apple/UNIX formats (and the format has been selected), then the CREATOR and TYPE are taken from the values stored in the Apple/UNIX file.

Other files can have their CREATOR and TYPE set from their file name extension (the `-map` option), or their magic number, the `-magic` option. If the default match is used in the `mapping` file, then these values override the default CREATOR and TYPE.

The CREATOR and TYPE information is stored in all the Apple/UNIX encoded files. For other files it is possible to base the CREATOR and TYPE on the filename’s extension using a mapping file (the `-map` option) and/or using the *`magic number`* (usually a signature in the first few bytes) of a file (the `-magic` option). If both these options are given, then their order on the command line is important. If the `-map` option is given first, then a filename extension match is attempted before a magic number match. However, if the `-magic` option is given first, then a magic number match is attempted before a filename extension match.

If a mapping or magic file is not used, or no match is found then, the default CREATOR and TYPE for all regular files can be set by using entries in the `.mkisofsrc` file or using the `-hfs-creator` and/or `-hfs-type` options. The default values for CREATOR and TYPE are `unix` and `TEXT`.

The format of the *`mapping`* file is the same *`afpfile`* format as used by *`aufs`*. This file has five columns for the *`extension`*, *`file translation`*, *`CREATOR`*, *`TYPE`* and *`Comment`*. Lines starting with the # character are comment lines and are ignored, for example:

# Example	filename	mapping file		
#				
# EXTN	XLate	CREATOR	TYPE	Comment
	Raw	8BIM	TIFF	Photoshop TIFF image
	Ascii	BnHq	TEXT	BinHex file
	Raw	MSWD	WDBN	Word file
	Raw	TVOD	MooV	QuickTime Movie
*	Ascii	ttxt	TEXT	Text file

The *`EXTN`* column defines the UNIX filename extension to be mapped. The default mapping for any filename extension that doesn't match is defined with the `*` character.

The *`Xlate`* column defines the type of text translation between the UNIX and Macintosh file. It is ignored by `mkisofs` but is kept to be compatible with `aufs`(1). Although `mkisofs` does not alter the contents of a file, if a binary file has its TYPE set as TEXT, it *may* be read incorrectly on a Macintosh. Therefore a better choice for the default TYPE may be `????`.

The *`CREATOR`* and *`TYPE`* keywords must be 4 characters long and enclosed in single quotes.

The comment field is enclosed in double quotes. It is ignored by `mkisofs`, but is kept to be compatible with `aufs`.

The format of the *`magic`* file is almost identical to the `magic`(4) file used by the Linux `file`(1) command. The routines for reading and decoding the *`magic`* file are based on the Linux `file`(1) command.

This file in the following example has four tab-separated columns for the `byte offset`, `type`, `test` and `message`. Lines starting with the # character are comment lines and are ignored.

```
# Example       magic file
#
# off           type            test            message
0               string          GIF8            8BIM GIFf GIF image
0               beshort         0xffd8          8BIM JPEG image data
0               string          SIT!            SIT! SIT! StuffIt Archive
0               string          \037\235        LZIV ZIVU standard
                                                unix compress
0               string          \037\213        GNUz ZIVU gzip
                                                compressed data
0               string          %!              ASPS TEXT Postscript
0               string          \004%!          ASPS TEXT PC Postscript
                                                with a ^D to start
4               string          moov            txtt MooV QuickTime
                                                movie file (moov)
4               string          mdat            txtt MooV QuickTime
                                                movie file (mdat)
```

The format of the file is described in the `magic`(4) man page. The only difference here is that for each entry in the magic file, the message for the initial offset must be 4 characters for the CREATOR followed by 4 characters for the TYPE. White space is optional between them. Any other characters on this line are ignored. Continuation lines (starting with a '>') are also ignored.

Using the `-magic` option may significantly increase processing time as each file has to opened and read to find its magic number.

A full CREATOR/TYPE database can be found at `http://www.angelfire.com/il/szekely/index.html`

HFS MACINTOSH FILE FORMATS

Macintosh files have two parts called the `Data` and `Resource` fork. Either may be empty. UNIX, and many other OSs can only cope with files having one part or fork. Macintosh files also have a number of attributes associated with them, probably the most important are the TYPE and CREATOR. Again UNIX has no concept of these types of attributes.

For example, a Macintosh file may be a JPEG image where the image is stored in the Data fork and a desktop thumbnail stored in the Resource fork. It is usually the information in the data fork that is useful across platforms.

Therefore to store a Macintosh file on a UNIX filesystem, a way has to be found to cope with the two forks and the extra attributes, which are referred to as the `finder` information. Unfortunately, it seems that every software package that stores Macintosh files on UNIX has chosen a completely different storage method.

The Apple/UNIX formats that `mkisofs` partially supports are:

CAP AUFS format	Data fork is stored in a file. Resource fork is in subdirectory `.resource` with same filename as data fork. Finder info in `.finderinfo` subdirectory with same filename.
AppleDouble/Netatalk	Data fork is stored in a file. Resource fork stored in a file with same name prefixed with `%`. Finder info also stored in same `%` file. Netatalk uses the same format, but the resource fork/finderinfo stored in subdirectory `.AppleDouble` with same name as data fork.
AppleSingle	Data structures are similar to above, except both forks and finder information are stored in one file.
Helios EtherShare	Data fork is stored in a file. Resource fork and finder information are stored together in subdirectory `.rsrc` with same filename as data fork.
IPT UShare	Very similar to the EtherShare format, but the finder information is stored slightly differently.

MacBinary	Both forks and finder information are stored in one file.
Apple PC Exchange	Used by Macintosh systems to store Apple files on DOS (FAT) disks. Data fork stored in a file. Resource fork in subdirectory `resource.frk` (or `RESOURCE.FRK`). Finder info as one record in file `finder.dat` (or `FINDER.DAT`). Separate `finder.dat` for each data fork directory. Note: `mkisofs` requires the native FAT cluster size of the disk that the PC Exchange files are on (or have been copied from). This size is given by the `-cluster-size` option. The cluster or allocation size can be found by using the DOS utility `CHKDSK`. May not work with PC Exchange v2.2 or higher files (available with MacOS 8.1). DOS media containing PC Exchange files should be mounted as type `msdos`, not `vfat` when using Linux.
SGI/XINET	Used by SGI machines when they mount HFS disks. Data fork is stored in a file. Resource fork is in subdirectory `.HSResource` with same name. Finder info as one record in file `.HSancillary`. Separate `.HSancillary` for each data fork directory.
Thursby Software Systems DAVE	Allows Macintosh systems to store Apple files on SMB servers. Data fork is stored in a file. Resource fork is in subdirectory `resource.frk`. Uses the AppleDouble format to store resource fork.
Services for Macintosh	Format of files stored by NT Servers on NTFS filesystems. Data fork is stored as `filename`. Resource fork

stored as a NTFS *`stream`* called `filename:AFP_Resource`. The finder info is stored as a NTFS *`stream`* called `filename:Afp_AfpInfo`. These streams are normally invisible to the user.

The `mkisofs` command only partially supports the SFM format. If an HFS file or folder stored on the NT server contains an illegal NT character in its name, then NT converts these characters to Private Use Unicode characters. The characters are: " * / < > ? | also a space or period if it is the last character of the file name, character codes 0x01 to 0x1f (control characters) and Apple' apple logo.

Unfortunately, these private Unicode characters are not readable by the `mkisofs` NT executable. Therefore any file or directory name containing these characters will be ignored, including the contents of any such directory.

The `mkisofs` command will attempt to set the CREATOR, TYPE, date and possibly other flags from the finder info. Additionally, if it exists, the Macintosh filename is set from the finder info, otherwise the Macintosh name is based on the UNIX filename. See HFS MACINTOSH FILE NAMES section below.

When using the `-apple` option, the TYPE and CREATOR are stored in the optional System Use or SUSP field in the ISO9660 Directory Record in much the same way as the Rock Ridge attributes are. Apple extensions are added at the beginning of the existing Rock Ridge attributes so to get the Apple extensions, you get the Rock Ridge extensions as well.

The Apple extensions require the resource fork to be stored as an ISO9660 `associated` file. This is just like any normal file stored in the ISO9660 filesystem except that the associated file flag is set in the Directory Record (bit 2). This file has the same name as the data fork (the file seen by non-Apple machines). Associated files are normally ignored by other operating systems.

When using the `-hfs` option, the TYPE and CREATOR plus other finder information are stored in a separate HFS directory, not visible on the ISO9660 volume.

In most cases, it is better to use the `-hfs` option instead of the `-apple` option, as the latter imposes the limited ISO9660 characters allowed in filenames. However, the Apple extensions do give the advantage that the files are packed on the disk more efficiently and it may be possible to fit more files on a CD.

HFS MACINTOSH FILE NAMES

Where possible, the HFS filename that is stored with an Apple/UNIX file is used for the HFS part of the CD. However, not all the Apple/UNIX encodings store the HFS filename with the finderinfo. In these cases, the UNIX filename is used with escaped special characters. Special characters include `/` and characters with codes over 127.

Aufs escapes these characters by using `:` followed by the character code as two hex digits. Netatalk and EtherShare have a similar scheme, but uses `%` instead of a `:`.

If `mkisofs` command cannot find an HFS filename, then it uses the UNIX name, with any `%xx` or `:xx` characters (xx == two hex digits) converted to a single character code. If `xx` are not hex digits ([0-9a-fA-F]), then they are left alone, although any remaining `:` is converted to `%` as colon is the HFS directory separator. Care must be taken, as an ordinary UNIX file with `%xx` or `:xx` will also be converted. For example,

This:2fFile	converted to This/File
This:File	
This:t7File	converted to This%t7File

Although HFS filenames appear to support upper and lower case letters, the filesystem is case insensitive, that is the filenames `aBc` and `AbC` are the same. If a file is found in a directory with the same HFS name, then `mkisofs` will attempt, where possible, to make a unique name by adding _ characters to one of the filenames.

If an HFS filename exists for a file, then `mkisofs` can use this name as the starting point for the ISO9660, Joliet and Rock Ridge filenames using the `-mac-name` option. Normal UNIX files without an HFS name will still use their UNIX name.

If a MacBinary or PC Exchange file is stored as `someimage.gif.bin` on the UNIX filesystem, but contains a HFS file called `someimage.gif`, then this is the name that would appear on the HFS part of the CD. However, as `mkisofs` uses the UNIX name as the starting point for the other names, then the ISO9660 name generated will probably be *`SOMEIMAG.BIN`* and the Joliet/Rock Ridge would be *`someimage.gif.bin`*. Although the actual data in this case is a GIF image. This option will use the HFS filename as the starting point and the ISO9660 name will probably be *`SOMEIMAG.GIF`* and the Joliet/Rock Ridge would be *`someimage.gif`*.

Using the `-mac-name` option will not currently work with the `-T` option. The UNIX name will be used in the TRANS.TBL file, not the Macintosh name.

The character set used to convert any HFS file name to a Joliet/Rock Ridge file name defaults to *`cp10000`* (Mac Roman). The character set used can be specified using the `-input-hfs-charset` option. Other built-in HFS character sets are cp10006 (MacGreek), cp10007 (MacCyrillic), cp10029 (MacLatin2), cp10079 (MacIcelandandic) and cp10081 (MacTurkish).

Note that the character codes used by HFS file names taken from the various Apple/UNIX formats will not be converted as they are assumed to be in the correct Apple character set. Only the Joliet/Rock Ridge names derived from the HFS file names will be converted.

The existing `mkisofs` code will filter out any illegal characters for the ISO9660 and Joliet filenames, but as `mkisofs` expects to be dealing directly with UNIX names, it leaves the Rock Ridge names as is. But as `/` is a legal HFS filename character, the `-mac-name` option converts `/` to a _ in Rock Ridge filenames.

If the Apple extensions are used, then only the ISO9660 filenames will appear on the Macintosh. However, as the Macintosh ISO9660 drivers can use *`Level 2`* filenames, then you can use options like `-allow-multidot` without problems on a Macintosh. Take care naming the files. For example, *`this.file.name`* will be converted to *`THIS.FILE`*. That is, because there is only one `.`. Also, the filename *`abcdefgh`* will be seen as *`ABCDEFGH`* but *`abcdefghi`* will be seen as *`ABCDEFGHI.`*, that is, with a `.` at the end. All filenames will be in uppercase when viewed on a Macintosh.

HFS CUSTOM VOLUME/FOLDER ICONS

To give a HFS CD a custom icon, make sure the root (top level) folder includes a standard Macintosh volume icon file. To give a volume a custom icon on a Macintosh, an icon has to be pasted over the volume's icon in the "Get Info" box of the volume. This creates an invisible file called `Icon\r`, where `\r` is the carriage return character in the root folder.

A custom folder icon is very similar. An invisible file called `Icon\r` exits in the folder itself.

Probably the easiest way to create a custom icon that `mkisofs` can use is to format a blank HFS floppy disk on a Mac and paste an icon to its "Get Info" box. If using Linux with the HFS module installed, mount the floppy using a command like: `mount −t hfs /dev/fd0 /mnt/floppy` The floppy will be mounted as a CAP file system by default. Then run `mkisofs` using a command like: `mkisofs −−cap −o output source_dir /mnt/floppy` If you are not using Linux, then you can use the hfsutils utilities to copy the icon file from the floppy. However, care has to be taken, as the icon file contains a control character. For example, `hmount /dev/fd0 hdir −a hcopy −m Icon^V^M icon_dir/icon`, where `^V^M` is control-V followed by control-M. Then run `mkisofs` by using a command like: `mkisofs −−macbin −o output source_dir icon_dir`

The procedure for creating/using custom folder icons is very similar. Paste an icon to folder's "Get Info" box and transfer the resulting `Icon\r` file to the relevant directory in the `mkisofs` source tree.

You may want to hide the icon files from the ISO9660 and Joliet trees.

To give a custom icon to a Joliet CD, follow the instructions found at: http://www.cdrfaq.org/faq03.html

HFS BOOT DRIVER

It may be possible to make the hybrid CD bootable on a Macintosh.

A bootable HFS CD requires an Apple CD-ROM (or compatible) driver, a bootable HFS partition and the necessary System, Finder, `etc` files, and so on.

A driver can be obtained from any other Macintosh bootable CD-ROM using the *`apple_driver`* utility. This file can then be used with the `-boot-hfs-file` option.

The HFS partition, that is, the hybrid disk in our case must contain a suitable System Folder, again from another CD-ROM or disk.

For a partition to be bootable, it must have it's *`boot block`* set. The boot block is in the first two blocks of a partition. For a non-bootable partition the boot block is full of zeros. Normally, when a System file is copied to partition on a Macintosh disk, the boot block is filled with a number of required settings.

Therefore, the utility *`apple_driver`* also extracts the boot block from the first HFS partition it finds on the given CD-ROM and this is used for the HFS partition created by `mkisofs`.

> **Note**
>
> By using a driver from an Apple CD and copying Apple software to your CD, you become liable to obey Apple Computer, Inc. Software License Agreements.

EL TORITO BOOT INFORMATION TABLE

When the `-boot-info-table` option is given, `mkisofs` will modify the boot file specified by the `-b` option by inserting a 56-byte boot information table at offset 8 in the file. This modification is done in the source filesystem, so make sure you use a copy if this file is not easily recreated. This file contains pointers which may not be easily or reliably obtained at boot time.

The format of this table is as follows; all integers are in section 7.3.1 (little endian) format. Offset Name Size Meaning 8 bi_pvd 4 bytes LBA of primary volume descriptor 12 bi_file 4 bytes LBA of boot file 16 bi_length 4 bytes Boot file length in bytes 20 bi_csum 4 bytes 32-bit checksum 24 bi_reserved 40 bytes Reserved The 32-bit checksum is the sum of all the 32-bit words in the boot file starting at byte offset 64. All linear block addresses (LBAs) are given in CD sectors (normally 2048 bytes).

CONFIGURATION

The `mkisofs` command looks for the `.mkisofsrc` file, first in the current working directory, then in the user's home directory, and then in the directory in which the `mkisofs` binary is stored. This file is assumed to contain a series of lines of the form `TAG=`*`value`*, that you can specify certain options. The case of the tag is not significant. Some fields in the volume header are not settable on the command line but can be altered through this facility. Comments may be placed in this file, using lines which start with a hash (#) character.

`APPI`	The application identifier, describing the application that will be on the disc. There is space on the disc for 128 characters of information. May be overridden using the `-A` command line option.
`COPY`	The copyright information, often the name of a file on the disc containing the copyright notice. There is space in the disc for 37 characters of information. May be overridden using the `-copyright` command line option.
`ABST`	The abstract information, often the name of a file on the disc containing an abstract. There is space in the disc for 37 characters of information. May be overridden using the `-abstract` command line option.
`BIBL`	The bibliographic information, often the name of a file on the disc containing a bibliography. There is space in the disc for 37 characters of information. May be overridden using the `-bilio` command line option.
`PREP`	The preparer of the CD-ROM, usually with a mailing address and phone number. There is space on the disc for 128 characters of information. May be overridden using the `-p` command line option.
`PUBL`	The publisher of the CD-ROM, usually with a mailing address and phone number. There is space on the disc for 128 characters of information. May be

	overridden using the `-P` command line option.
`SYSI`	The System Identifier. There is space on the disc for 32 characters of information. May be overridden using the `-sysid` command line option.
`VOLI`	The Volume Identifier. There is space on the disc for 32 characters of information. May be overridden using the `-V` command line option.
`VOLS`	The Volume Set Name. There is space on the disc for 128 characters of information. May be overridden using the `-volset` command line option.
`HFS_TYPE`	The default TYPE for Macintosh files. Must be exactly 4 characters. May be overridden using the `-hfs-type` command line option.
`HFS_CREATOR`	The default CREATOR for Macintosh files. Must be exactly 4 characters. May be overridden using the `-hfs-creator` command line option.

The `mkisofs` command can also be configured at compile time with defaults for many of these fields. See the file `defaults.h`.

EXAMPLES

To create a ISO-9660 filesystem image in the file *`cd.iso`*, where the directory *`cd_dir`* will become the root directory if the CD:

```
% mkisofs −o cd.iso cd_dir
```

To create a CD with Rock Ridge extensions of the source directory *`cd_dir`*:

```
% mkisofs −o cd.iso −R cd_dir
```

To create a CD with Rock Ridge extensions of the source directory *cd_dir* where all files have at least read permission and all files are owned by *root*:

```
% mkisofs −o cd.iso −r cd_dir
```

To create a HFS hybrid CD with the Joliet and Rock Ridge extensions of the source directory *cd_dir*:

```
% mkisofs −o cd.iso −R −J −hfs cd_dir
```

To create a HFS hybrid CD from the source directory *cd_dir* that contains Netatalk Apple/UNIX files:

```
% mkisofs −o cd.iso −-netatalk cd_dir
```

To create a HFS hybrid CD from the source directory *cd_dir*, giving all files CREATOR and TYPES based on just their filename extensions listed in the file `mapping`:

```
% mkisofs −o cd.iso −map mapping cd_dir
```

To create a CD with the Apple Extensions to ISO9660 from the source directories *cd_dir* and *another_dir.* Files in all the known Apple/UNIX format are decoded and any other files are given CREATOR and TYPE based on their magic number given in the file `magic`:

```
% mkisofs −o cd.iso −apple −magic magic −probe \ cd_dir
another_dir
```

To put different files on the CD that all have the name README, but have different contents when seen as a ISO9660/RockRidge, Joliet or HFS CD.

Current directory contains:

```
% ls −F README.hfs README.joliet README.unix cd_dir/
```

The following command puts the contents of the directory *cd_dir* on the CD along with the three README files, but only one will be seen from each of the three filesystems, that is, the file README.hfs will be seen as README on the HFS CD and the other two README files will be hidden. Similarly for the Joliet and ISO9660/RockRidge CD.

```
% mkisofs −o cd.iso −hfs −J −r −graft-points \
−hide README.hfs −hide README.joliet \ −hide-joliet
README.hfs −hide-joliet README.unix \ −hide-hfs
README.joliet −hide-hfs README.unix \ README=README.hfs
README=README.joliet \ README=README.unix cd_dir
```

NOTES

The `mkisofs` command may safely be installed suid root. This may be needed to allow `mkisofs` to read the previous session when creating a multi session image.

If `mkisofs` is creating a filesystem image with Rock Ridge attributes and the directory nesting level of the source directory tree is too much for ISO-9660, `mkisofs` will do deep directory relocation. This results in a directory called `RR_MOVED` in the root directory of the CD. You cannot avoid this directory.

NOTE: The –m and –x option description should both be updated, they are wrong. Both now work identical and use filename globbing. A file is excluded if either the last component matches or the whole path matches.

RESTRICTIONS

HFS file/directory names that share the first 31 characters have _N' (N == decimal number) substituted for the last few characters to generate unique names.

When creating an HFS volume with the multisession options, `-M` and `-C`, only files in the last session will be in the HFS volume. i.e. `mkisofs` can not *add* existing files from previous sessions to the HFS volume.

Symbolic links (as with all other non-regular files) are not added to the HFS directory.

Hybrid volumes may be larger than pure ISO9660 volumes containing the same data.

Using the `-mac-name` option will not currently work with the `-T` option - the UNIX name will be used in the TRANS.TBL file, not the Macintosh name.

It is not possible to use the the `-sparc-boot` or `-generic-boot` options with the `-boot-hfs-file` or `-prep-boot` options.

Any files that have hard links to files not in the tree being copied to the ISO9660 filesystem will have an incorrect file reference count.

Does not check for SUSP record(s) in "." entry of the root directory to verify the existence of Rock Ridge enhancements. This problem is present when reading old sessions while adding data in multi-session mode.

Does not properly read relocated directories in multi-session mode when adding data. Any relocated deep directory is lost if the new session does not include the deep directory. Repeat by: create first session with deep directory relocation then add new session with a single dir that differs from the old deep path.

Does not re-use RR_MOVED when doing multi-session from TRANS.TBL

Does not create whole_name entry for RR_MOVED in multi-session mode.

AVAILABILITY

The `mkisofs` command is available as part of the cdrecord package from ftp://ftp.fokus.gmd.de/pub/unix/cdrecord/ `hfsutils` from ftp://ftp.mars.org/pub/hfs

AUTHOR

This page has been adapted from information provided by:

Joerg Schilling

Seestr. 110 D-13353

Berlin Germany

Additional information can be found at **`http://www.fokus.gmd.de/research/cc/glone/employees/jeorg.schilling/private/cdrecord.html`**

HFS MKHYBRID MAINTAINER

James Pearson

SEE ALSO

Commands: `cdrecord`(1), `apple_driver`(8)

Others: `magic`(5)