

The ISO 9660 File System

This article describes the ISO 9660 file system format used on compact disc read only memory (CD-ROM). CD-ROMs have become so popular (and cheap) that its market share grew exponential over the last years. Therefore, it is worthwhile to examine the file system used on CD-ROMs. What makes it different to other file systems such as the UNIX File System (UFS) used on e.g. SunOS systems?

Gratien D?haese
Alcatel Bell
Switching Systems Division
May 1995

Introduction

The audio compact disks are only one decade old, but surprisingly enough it pushed the vinyl records completely from the market. It was in the beginning of the 1980s that Philips and Sony introduced the Compact Disc Digital Audio (CD-DA) Standard, better known as the Red Book standard.

It was also Philips and Sony who introduced in 1984 the CD-ROM (Compact Disc Read Only Memory) standard, which is commonly known as the Yellow Book standard.

The computer industry immediately saw the benefits of CD-ROMs, namely:

- cheaper in production than tapes
- cheaper in shipping to customers
- less vulnerable to dust, fingerprints, magnetic fields than tapes
- large capacity, more than 600 Mbytes
- cannot be overwritten by accident, because it is a read-only medium

Therefore, it did not take long before the CD-ROM became quiet popular among developers and customers.

However, to make a CD-ROM was until last year problematic because for mastering a CD-ROM special equipment was needed. These so called CD-writers have become affordable now, so that making a CD-ROM master (which can also be read with normal CD drives) is no big deal anymore. Making duplicates (the silver CD-ROMs) from a master (the golden CD-ROM) costs about 60 BEF a piece, which is a fair price!

The only problem was that CD-ROMs were not interchangeable amongst different computer architectures. The ISO 9660 standard was created to define the external characteristics of data on a CD-ROM to make it architecture independent. The first standard did not quite go far enough, so an *ad hoc* committee of hardware and software suppliers met at the High Sierra Hotel in Nevada (USA) and drew up a proposal for the ?High Sierra? format for CD-

ROM file structure. The previous ISO 9660 standard and the High Sierra file structure were combined into a complete ISO 9660 standard (1988).

There are almost 6 million CD readers sold until now, which proves that CD-ROMs are a very popular medium among the commercial and non-commercial end-users.

Colour books of standards

The Compact Disc Digital Audio (CD-DA) standard (or just CD) made by Philips and Sony in the early 1980s became the de facto standard for all audio discs, and means that any CD plays on any audio CD drive. This standard became known as the Red Book. The Red Book specifies that the audio data is on the CD in one or more tracks. Each track is normally one song. These tracks are further subdivided into sectors that are 1/75 of a second in length and contain 2352 bytes of audio data in digital form. A maximum of 99 audio tracks may be placed on a standard Red Book disc.

In addition to the 2352 bytes of audio data, the Red Book specifies the addition of 2 layers of error detection and error correction code (EDC/ECC). The compact disc utilises the Cross Interleave Reed-Solomon Code (CIRC) in its first two layers of error protection. If a disc gets scratched or dirty and a laser cannot read the data, the CD player uses the CIRC to recreate the music.

Each sector is also assigned 98 control bytes, which control the timing information that the CD player uses to display the length of each song.

The Yellow Book standard was introduced by Philips and Sony in 1984 which defined the Compact Disc Read Only Memory (CD-ROM) layout.

The Yellow Book further defined the Red Book by adding two new types of tracks.

The track type defined in the Red Book is:

- CD-Audio, for audio music.

The two new track types defined in the Yellow Book are:

- CD-ROM Mode 1, usually used for computer data.
- CD-ROM Mode 2, usually used for compressed audio data, and video/picture data. Also, usually further defined as XA (eXtended Architecture).

The CD-ROM Mode 1 and Mode 2 tracks use the Red Book specifications as a foundation. The difference between the Red Book and the Yellow Book is a redefinition of the 2352 byte Red Book data area.

Furthermore, the Yellow Book CD-ROM Mode 1 and Mode 2 use the same track layout as the Red Book specification, including the error correction and control bytes. The fundamental difference between the two Yellow Book CD-ROM modes is the way in which they use the main data segment.

The Yellow Book CD-ROM Mode 1 defines the ISO 9660 and non-ISO 9660 standards. The ISO 9660 compliant CD-ROMs are readable by any kind of (modern) operating systems, such as DOS, UNIX, MacOS, AmigaDOS and other OSes.

The CD-ROM Mode 1 divides the 2352 byte data area defined by the Red Book standards into the following:

- 12 bytes of synchronisation
- 4 bytes of header information
- 2048 bytes of user information
- 288 bytes of error correction and detection codes.

The first 16 bytes contain the synchronisation and header information that the computer uses to determine which sector it is reading. The following 2048 bytes contain the actual user data. Together, these two subdivisions comprise the full 2352 byte portion of the Red Book standard.

The last 288 bytes carry an additional layer of error correction and direction code. This additional layer, which is found only in Mode 1, provides the reliability that is needed for certain types of computer data.

CD-ROM Mode 2 redefines the use of the 2352 byte data area as follows:

- 12 bytes of synchronisation
- 4 bytes of header information
- 2336 bytes of user data.

The main advantage of Mode 2 is that it provides an additional 14 per cent of the user data space per sector (2336 versus 2048 bytes). The reason is that Mode 2 does not have the additional EDC and ECC error correction data of Mode 1.

Mode 2 discs are normally used in extended architecture (XA) format. Even without XA, there are still two layers of error correction as defined in the Red Book standard. CD-ROM Mode 2 discs can be read by a standard CD-ROM drive, but require special software to decode and strip the user data from each sector.

CD-ROM Mode 2 allows compressed audio data and video/picture data to be incorporated on the disc, thanks to the alignment of the byte layout. The drawback is that a CD-ROM drive reading this data cannot read computer data while it's playing audio.

The next step in CD technology was to create a file format that lent itself to the incorporation of audio and video/picture data. To define this extension to the Yellow Book standard, Sony and Philips produced the Compact Disc

Read Only Memory Extended Architecture (CD-ROM/XA). The XA disc has compressed audio and computer data interleaved on the same track, so it can read the computer data and play audio on the same time.

This was a dramatic improvement on existing Yellow Book technology, and marks the point from which application discs that made best use of CD-ROM technology started to develop. CD-ROM/XA Mode 2 is subdivided into Form 1 (for computer data) and Form 2 (for compressed audio data and video/picture data).

The Compact Disc Interactive (CD-I) Media standard was released in 1987 by Philips. This standard specifies the CD-I disc layout and an operating system called CD-RTOS. This specification is known as the Green Book standard. Like CD-ROM/XA, this standard allows for the interleaving of computer data and compressed audio on the same track. The CD-I track is not shown in the table of contents on the disc. This prevents audio players from playing the CD-I track. The sector layout of a CD-I disc is identical to CD-ROM/XA. A CD-I system consists of a stand-alone CD-I player connected to a TV set.

Remember, the main drawback of a CD-ROM is, at least for some people, that it is a read-only medium (the ROM part of its name)! Writable mediums were needed to fulfil new (created) needs. In Frankfurt (Germany) a group was formed (guess the name) - the Frankfurt Group - which includes Philips, Sony, Kodak and others to take CD-ROM into the writable market. This became the Orange Book standard defining a CD that lets users write audio and/or data to disc. Part 1 of the Orange Book describes a Compact Disc-Magneto Optical (CD-MO) where data can be written, erased and rewritten. Part 2 describes a Compact Disc Write Once (CD-WO) where data can be written but not erased. The CD-WO is better known under its name CD-R where R stands for recordable. CD writers are becoming quite popular these days (and affordable).

CD-ROM Hardware

CD-ROM drives in general function like any other write-protected disk drive, and the device driver level appear quite ordinary.

CD-ROMs do not have a fixed number of sectors in a fixed-arm position. Instead, an inward spiral of records is arranged to maintain minimal latency from record to record.

Furthermore, CD-ROMs are not random access, but rather sequential access, which is why they tend to be slow, because the head hunts to find the desired record. The sectors are indexed by track in the same way cylinder, head, and sector indices are used in a hard-disk drive. With SCSI, these sectors are hidden by logical address translation in the drive's controller. Unlike with hard drives, a separate head is not used for timing information, so each time a CD-ROM is moved for random access it must search up and down the spiral in the vicinity the head let down to find the desired sector.

CD-ROM sector sizes are large, usually 2048 bytes per sector (larger sizes are possible). Therefore, the concept of *logical sectors* is introduced. Each logical sector, with a constant logical sector size (= the sector size), starts in a different sector from any other logical sector.

File Systems on CD-ROM

A CD-ROM may be mastered with any kind of information on it. Sun Microsystems, for example, uses the Berkeley UNIX UFS file systems on many CD-ROMs. This makes them only usable on Sun equipment, which is no big deal for a bootable CD-ROM with an operating system on it, but for distributing general information it's a big limitation.

However, because CD-ROMs are especially suited to volume publishing of information, a standard file system useful across many kinds of architecture is very desirable. Before there was a standard on this matter some were using the High Sierra format on CD-ROM, which arranged file information in a dense, sequential layout to minimise nonsequential access.

The High Sierra file system format uses a hierarchical (eight levels of directories deep) tree file system arrangement, similar to UNIX and MS-DOS. High Sierra has a minimal set of file attributes (directory or ordinary file and time of recording) and name attributes (name, extension, and version). The designers realised they could never get people to agree on a unified definition of file attributes, so the minimum *common* information was encoded, and a place for future optional extensions (system use area) was defined for each file.

High Sierra was soon adapted (with changes) as an international standard (ISO 9660-1988), and the ISO 9660 file system format is now used throughout the industry.

The ISO 9660 File System

An ISO 9660 CD-ROM is described in Figure 1.

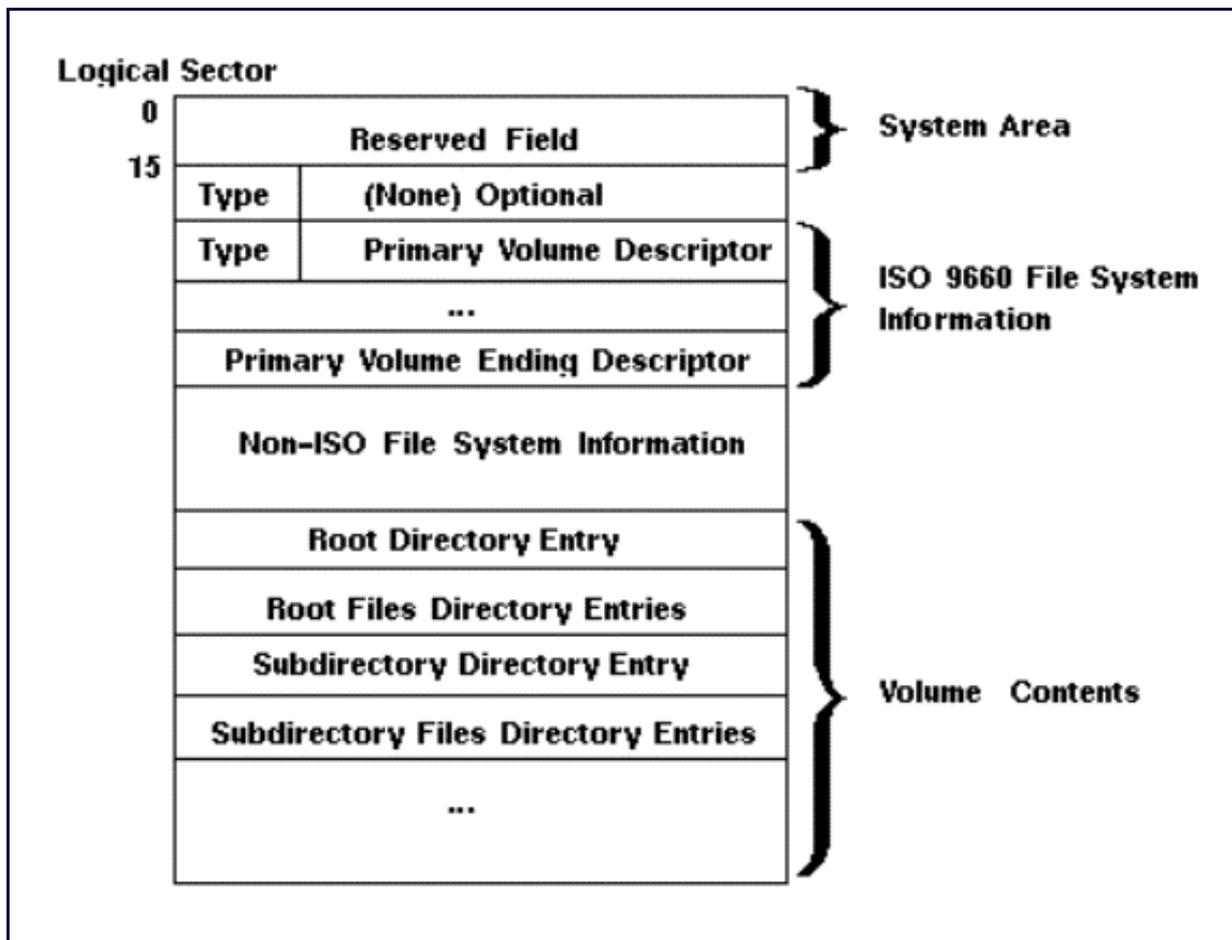


Figure 1: ISO 9660 CD-ROM

A reserved field at the beginning of the disk is present for use in booting CD-ROM on a computer (system area). As a matter of fact its use was not specified by the ISO 9660 standard, but generally it is used for boot information.

Immediately afterwards, a series of volume descriptors details the contents and kind of information contained on the disk (something like the partition table of MS-DOS).

A volume descriptor describes the characteristics of the file system information present on a given CD-ROM, or volume. It is divided into two parts;

- the type of volume descriptor, and
- the characteristics of the descriptor.

The volume descriptor is constructed in this manner so that if a program reading the disk does not understand a particular descriptor, it can just skip over it until it finds one it recognises, thus allowing the use of many different types of information on one CD-ROM. Also, if an error were to render a descriptor unreadable, a subsequent redundant copy of a descriptor could then allow for fault recovery. When checking CD-ROMs with a *dump* utility we find each descriptor back in a single logical sector on itself, and also a backup of the descriptor a few logical sectors further.

The minimum requirement is that it has a primary descriptor describing the ISO 9660 file system and an ending descriptor (a variable length table that contains information on how many other descriptors are present).

Little/Big Endian

In order to accommodate the two common byte orders, Big Endian (680x0, Sparc) and Little Endian (80x86, Rx000), ISO 9660 has data types which allow either and consequently are twice as big.

For example, the 32-bit integer (0x11223344) is represented as the byte sequence (0x44, 0x33, 0x22, 0x11, 0x11, 0x22, 0x33, 0x44), which is essentially a binary palindrome.

This method is often referred to as 733 (section 7.3.3 from the ISO 9660 standard, both byte orders) and is represented in Figure 2.

Figure 2: Both Byte Orders



Figure 2 illustrates, the Big Endian addressing model assigns or maps the lowest address to the highest-order (that is, the most significant or leftmost) data byte of a multibyte-scalar data item. The Little Endian addressing model assigns or maps the lowest address to the lowest-order (least significant or right-most) data byte of a multibyte-scalar data item.

ISO 9660 Primary Volume Descriptor

The ISO 9660 primary volume descriptor describes the characteristics of the ISO standard file system information present on a given CD-ROM (refer to Figure 3).

The ISO 9660 primary volume descriptor acts much like the superblock of the UNIX file system, providing details on the ISO 9660 compliant portion of the disk. Contained within the primary volume descriptor is the root directory record describing the location of the contiguous root directory. (As in UNIX, directories appear as files for the operating system's special use). Directory entries are successively stored within this region. Evaluation of the ISO 9660 filenames is begun at this location. The root directory is stored as an extent, or sequential series of sectors, that contains each of the directory entries appearing in the root. In addition, since ISO 9660 works by segmenting the CD-ROM into logical blocks, the size of these blocks is found in the primary volume descriptor as well.

A CD-ROM is only compliant to the ISO 9660 file system standard if there is a primary descriptor, and when there is an ending descriptor available (e.g., the volume descriptor constitute a variable length table which contains information on how many other descriptors are present).

It is possible to have many kind of file systems and information arrangements on a single CD-ROM. However, while many other kinds of descriptors can be used to optionally record non-ISO defined information contents, the primary volume descriptor is always present. It is even possible to have a Mixed Mode disc, containing audio tracks and data tracks on the same disc. The most common type of Mixed Mode discs is one where the first track on the disc is Mode 1 data (ISO 9660 or non-ISO 9660), and the remaining tracks on the disc are audio tracks. Another possibility is the so called Hybrid disc, which contains an ISO 9660 part and a non-ISO 9660 part (e.g. Apple HFS format). The popular magazine on CD-ROM called "CD-ROM Today" is an example of an Hybrid disc.

Referring back to Figure 3, the first entry is the Volume Descriptor Type (*type*), where it can have the following values:

- **Number 0**: shall mean that the Volume Descriptor is a Boot Record
- **Number 1**: shall mean that the Volume Descriptor is a Primary Volume Descriptor
- **Number 2**: shall mean that the Volume Descriptor is a Supplementary Volume Descriptor
- **Number 3**: shall mean that the Volume Descriptor is a Volume Partition Descriptor
- **Numbers 4-254** are reserved
- **Number 255**: shall mean that the Volume Descriptor is a Volume Descriptor Set Terminator.

```

struct iso_primary_descriptor {
    unsigned char type           [ISODCL ( 1, 1)]; /* 711 */
    unsigned char id            [ISODCL ( 2, 6)];
    unsigned char version       [ISODCL ( 7, 7)]; /* 711 */
    unsigned char unused1       [ISODCL ( 8, 8)];
    unsigned char system_id     [ISODCL ( 9, 40)]; /* unsigned chars */
    unsigned char volume_id     [ISODCL (41, 72)]; /* unsigned chars */
    unsigned char unused2       [ISODCL (73, 80)];
    unsigned char volume_space_size [ISODCL (81, 88)]; /* 733 */
    unsigned char unused3       [ISODCL (89, 120)];
    unsigned char volume_set_size [ISODCL (121, 124)]; /* 723 */
    unsigned char volume_sequence_number [ISODCL (125, 128)]; /* 723 */
    unsigned char logical_block_size [ISODCL (129, 132)]; /* 723 */
    unsigned char path_table_size [ISODCL (133, 140)]; /* 733 */
    unsigned char type_1_path_table [ISODCL (141, 144)]; /* 731 */
    unsigned char opt_type_1_path_table [ISODCL (145, 148)]; /* 731 */
    unsigned char type_m_path_table [ISODCL (149, 152)]; /* 732 */
    unsigned char opt_type_m_path_table [ISODCL (153, 156)]; /* 732 */
    unsigned char root_directory_record [ISODCL (157, 190)]; /* 9.1 */
    unsigned char volume_set_id   [ISODCL (191, 318)]; /* unsigned chars */
    unsigned char publisher_id    [ISODCL (319, 446)]; /* achars */
    unsigned char preparer_id     [ISODCL (447, 574)]; /* achars */
    unsigned char application_id  [ISODCL (575, 702)]; /* achars */
    unsigned char copyright_file_id [ISODCL (703, 739)]; /* 7.5 dchars */
    unsigned char abstract_file_id [ISODCL (740, 776)]; /* 7.5 dchars */
    unsigned char bibliographic_file_id [ISODCL (777, 813)]; /* 7.5 dchars */
    unsigned char creation_date   [ISODCL (814, 830)]; /* 8.4.26.1 */
    unsigned char modification_date [ISODCL (831, 847)]; /* 8.4.26.1 */
    unsigned char expiration_date [ISODCL (848, 864)]; /* 8.4.26.1 */
    unsigned char effective_date  [ISODCL (865, 881)]; /* 8.4.26.1 */
    unsigned char file_structure_version [ISODCL (882, 882)]; /* 711 */
    unsigned char unused4         [ISODCL (883, 883)];
    unsigned char application_data [ISODCL (884, 1395)];
    unsigned char unused5         [ISODCL (1396, 2048)];
};

```

Figure 3: File structure of an ISO 9660 primary volume descriptor

The second entry is called the Standard Identifier (*id*) and is set to ?CD001? for a CD-ROM compliant to the ISO 9660 standard.

Another interesting field is the Volume Space Size (*volume_space_size*) which contains the amount of data available on the CD-ROM. It is recorded according the 733 method (see Figure 4).

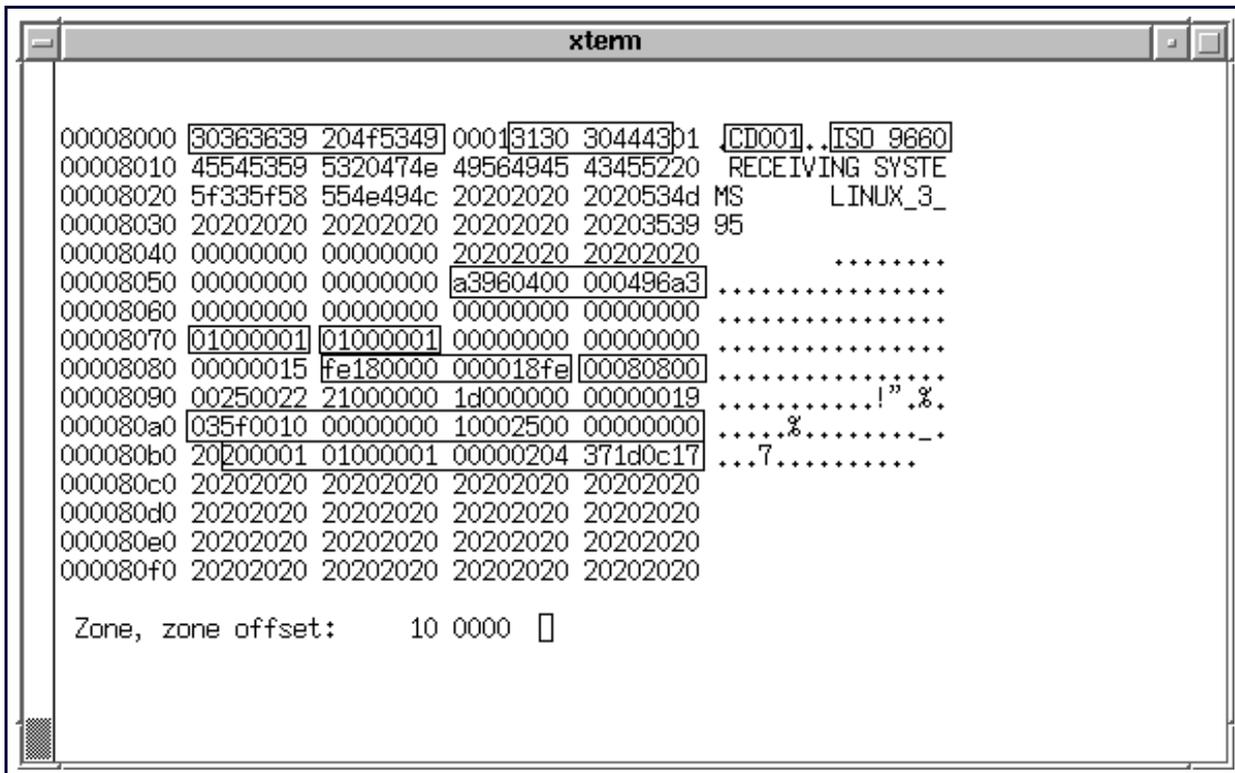


Figure 4: A dump of a file structure of an ISO 9660 primary volume descriptor

Figure 4 is a dump of a CD-ROM (made by the Belgian UNIX systems Users Group (B.U.U.G.) containing Linux 95) starting at logical sector 16 and we see the first 256 bytes (of the 2048 bytes). The rectangles made in Figure 4 are respectively the Standard Identifier, System Identifier, Volume Space Size (check if you can find the binary palindrome), Volume Set Size, Volume Sequence Number, Logical Block Size, Path Table Size, and the Directory Record for the Root Directory. Together with Figure 3 you can follow the parameters in Figure 4. Also, together with Figure 4 and Figure 5 you should be able to discover to way the Directory Record of the Root Directory is represented on a CD-ROM (explained in following paragraphs).

Directory-entry Format

A directory entry is a data structure that describes the characteristics of a file or directory, beginning with a length octet describing the size of the entire entry. Entries themselves are of variable length, up to 255 octets in size. Attributes for the file described by the directory entry are stored in the directory entry itself (unlike UNIX).

In Figure 5, the root directory entry is a variable length object, so that the name can be of variable length. (No other part in the directory entry is of variable length).

```

struct iso_directory_record {
    unsigned char length           [ISODCL (1, 1)]; /* 711 */
    unsigned char ext_attr_length [ISODCL (2, 2)]; /* 711 */
    unsigned char extent          [ISODCL (3, 10)]; /* 733 */
    unsigned char size            [ISODCL (11, 18)]; /* 733 */
    unsigned char date            [ISODCL (19, 25)]; /* 7 by 711 */
    unsigned char flags           [ISODCL (26, 26)];
    unsigned char file_unit_size  [ISODCL (27, 27)]; /* 711 */
    unsigned char interleave      [ISODCL (28, 28)]; /* 711 */
    unsigned char volume_sequence_number [ISODCL (29, 32)]; /* 723 */
    unsigned char name_len        [ISODCL (33, 33)]; /* 711 */
    unsigned char name            [38];
};

```

Figure 5: Data Structure of a CD-ROM file system directory entry

File Attributes

File attributes are very simple in ISO-9660. The most important file attribute is determining whether the file is a directory or an ordinary file. File attributes for the file described by the directory entry are stored in the directory entry and optionally, in the extended attribute record. The name length field specifies how long the name is and is limited to 31 characters.

Filenames

Filenames in ISO-9660 correspond to a DOS-like representation, with an uppercase, fixed-size base name, a delimiter (a period) to separate filenames from the extension, and a three-letter extension name (also uppercase). Directory names contain maximum 8 characters and do not have extensions.

With filenames the extension may be followed another delimiter (a semicolon) and a revision number of the file. For example, a typical filename would be FOO.BAR;1. There are additional restrictions on the type of allowed characters beyond that of alpha characters (0-9 and _).

The choice of filename is thus restricted to allow for the vast number of different systems that existed at the time the standard was determined. While the directory entries allow much larger names than this, the characteristics and size of the filename were developed to achieve level-one compliance with original High Sierra format.

Unfortunately, many systems with ISO 9660 capability are not compatible with the naming conventions. For example, on a UNIX system, a semicolon is used as a command delimiter in the shell interpreter.

To by-pass these problems the Rock Ridge Interchange Protocol (RRIP) was designed to allow users of POSIX and other UNIX like systems to remain much of the directory information that is in the native file system. This is important because there systems use directory entries for much more than just pointing to files. Directory entries can point to other entries (symbolic links) or to device drivers that are linked to peripheral devices such as hard disks, tape drives and CD-ROM drives (device files). The directory entry includes information that lets the system know what type of file it is dealing with, whether it is a regular file, directory, symbolic link, or device file. It also has information regarding who has permission to read, write and execute each file. Most of these systems are multi-users systems, and you would not want just anyone to be able to write to the device file that contains your operating system, because they could accidentally erase the entire operating system. On the other hand, permission may not set to tight, because it can make CD-ROM unusable for users when they have no read access to files.

File Pathname Traversal

There are two ways to locate a file on an ISO 9660 file system. One way is to successively interpret the directory names and look through each directory file structure to find the file (much the way MS-DOS and UNIX work to find a file). The other way is through the use of a precompiled table of paths, where all the entries are enumerated in the

successive contents of a file with the corresponding entries. Some systems do not have a mechanism for wandering through directories, they obtain a match by consulting the table.

While a large linear table seems a bit arcane, it can be of great value, as you can quickly search without wandering across the disk (thus reducing seek time).

File Contents

The ISO 9660 standard says practically nothing about the contents of files themselves - they can contain any kind of data one wishes to store.

Although the ISO 9660 standard allows an optional extended attributes record (XAR) stored at the beginning of the file's extent which can contain additional file attribute information. Extension attributes are simply a way to extend the attributes of files. Since attributes vary according to the user, most everyone has a different opinion on what a file attribute should specify.

The Rock Ridge extensions is an example of extended attributes to make POSIX alike file attributes (much like UNIX). Rock Ridge can also be used in a networked situation, since a single CD-ROM can be exported to a variety of different operating systems viewing the same files, while appearing to be in the local system's native file structure format. In sum, Rock Ridge is heading in the same 'universal' direction of other file systems like the Network File System (NFS).

Conclusions

A lot of people are aware of the ISO 9660 standard and its significance in sharing CD-ROM data between different platforms. ISO compliant CD-ROMs are interchangeable and can be used on any type of system and architecture. However, the minimalism that helped make the ISO 9660 standard successful may sometimes be too minimal for specific applications (such as distributing POSIX based, bootable CD-ROMs). Because ISO 9660 does not adequately support the POSIX file system, the Rock Ridge Group was formed to develop ISO 9660:1988 extensions, which take advantage of the system-use area of the directory record (provided for in ISO 9660) to store complete POSIX file system information.

Extensions to ISO 9660 can make a CD-ROM appear like a given target operating system (such as a POSIX compliant file system). By encoding these extensions (using the sharing-use protocols), you can allow for separate sets of attributes for the same file system. This lets you organize extended information for different systems (such as VMS, DOS, and UNIX) in a nonconflicting way. Also, any system that only understands ISO 9660 without any extensions can still gain access to the files and obtain the exact same contents of data for a file. It is quite simple, if the extensions are not understood, they are not used at all. You get the best of both worlds: ISO compatibility and interoperability, and POSIX operating system transparency and functionality.

Technology is not standing still also, because Philips and Sony just proposed a new CD-ROM standard which could contains 2.3 Gbytes of data (almost 4 times more than current CD-ROM standard). Probably the current ISO 9660 standard has to be adapted too with new extension attributes for new applications (to be defined).

References