QXL Hard Disc Format

A QXL Hard disc file is made up of three parts:

- The header which describes the format and settings for the disc;
- The map which holds the list of all the clusters on the data area where various files and directories lives;
- The data area which is where your data is saved.

In some documents you may see "group" used rather than "cluster" - I tend to favour the latter, but I also use both, so just remember that they are the same.

The Header

The header is the very first 64 bytes of the disc. It is part of the map and has the following structure:

Field	Offset	Size	Purpose
Qwa.id	$0000	Long	'QLWA' - Identification bytes.
Qwa_name	$0004	2 + 20	First word is the name size, then 20 bytes for the space padded name.
Qwa_spr0	$001A	Word	Spare - set to zero.
Qwa_uchk	$001C	Long	Update check. The high word is the random number, the low word the update count.
Qwa_intl	$0020	Word	Interleave factor - usually zero.
Qwa_sctg	$0022	Word	Sectors per cluster.
Qwa_sctt	$0024	Word	Sectors per track - usually zero.
Qwa_trkc	$0026	Word	Tracks per cylinder - usually zero.
Qwa_cyld	$0028	Word	Cylinders per drive - usually zero.
Qwa_ngrp	$002A	Word	Number of clusters.
Qwa_fgrp	$002C	Word	Number of free clusters.
Qwa_sctm	$002E	Word	Sectors per map.
Qwa_nmap	$0030	Word	Number of maps.
Qwa_free	$0032	Word	First free cluster.
Qwa_root	$0034	Word	Root directory number.
Qwa_rlen	$0036	Long	Root directory length.
Qwa_fsct	$003A	Long	First sector for this partition - usually zero.
Qwa_park	$003E	Word	Park cylinder - usually zero.

Quite a few of the fields in the header will always be zero, and those are flagged above.

The remainder can be calculated from the initial size of the disc at formatting time. The following list shows the various calculations carried out to fill in the header.

- `Qwa_sctg`, sectors per cluster, is MB / 32 rounded up and must be a minimum of 4. This means that a file on a QXL hard disc will always be at least 2048 bytes on the disc.
- `Qwa_ngrp`, number of clusters, is MB * 2048 / sectors per cluster (`qwa_sctg`).
- `Qwa_sctm`, sectors per map, is ((numberOfGroups (`qwa_ngrp`) * 2) + 64) / 512 rounded up.
- `Qwa_root`, the root directory number, is sectors per map (`qwa_sctm`) / sectors per group (`qwa_sctg`) rounded up.
- `Qwa_rlen`, root directory length, is always 64 after a format.

- `Qwa_free`, the first free cluster, is root directory number (`qwa_root`) + 1.
- `Qwa_fgrp`, the number of free clusters, is number of clusters (`qwa_ngrp`) - root directory number (`qwa_root`) - 1

So, for a freshly formatted 30 Mb QXL file, we have the following calculations:

- `Qwa_sctg`, sectors per cluster, is 30/32 = 0.9375 ⇒ 1 ⇒ 4.
- `Qwa_ngrp`, number of clusters, is 30*2048/4 = 15360.
- `Qwa_sctm`, sectors per map, is ((15360*2)+64)/512 = 30874/512 = 60.125 ⇒ 61.
- `Qwa_root`, the root directory number, is 61/4 = 15.25 ⇒ 16.
- `Qwa_rlen`, root directory length, is 64.
- `Qwa_free`, the first free cluster, is 16+1 = 17.
- `Qwa_fgrp`, the number of free clusters, is 15360 - 16 -1 = 15343.

The Map

This area of the disc is nothing more than, at most, 65,536, 16 bit unsigned words, starting on the disc as address 64_{dec} or $\$40_{hex}$. The offset into this long list of words is the file or directory number, the map is always file zero and has offset zero, or the first word in the map.

The next file on the disc is always the root directory and it is given the file number that is exactly the same number of sectors that make up the map. For example, a freshly formatted 30 MB disc will require a cluster size of 4 sectors, 2,048 bytes, and will use the first 16 clusters (for 61 sectors) for it's own use. These are clusters 0 through 15 inclusive, the root directory is therefore file number 16.

When any file holds more than one cluster, the clusters are effectively a linked list with the first one holding the cluster number of the second and so on, the final cluster's entry in the map, holds a zero to indicate the end of the list.

Looking back at the above example for a 30 MB disc, the first 16 words in the map, would read as follows:

```
$0001, $0002, $0003, $0004, $0005, $0006, $0007, $0008
$0009, $000A, $000B, $000C, $000D, $000E, $000F, $0000
```

The entry in the map for the root directory would be next, and would be a single word of zero as a directory is only 64 bytes long when created, so it's first cluster is also its last.

The file number is recorded in the directory for each file created, and that number is the index into the map where the list of blocks making up the file are to be found in the data area. Each entry in the map "points" to a cluster of data within the data area of the hard disc and each cluster is always contiguous - clusters cannot be split.

The way that the map "points" to the data is simply:

```
file_number * sectors_per_cluster * sector_size
```

On all QXL hard discs the sector size is 512 bytes and the sectors per cluster is obtained from the `qwa_sctg` field in the header. In the 30 MB example here, the map, file zero, would start at address:

```
0 * 4 * 512 = 0.
```

Finding the second, and subsequent cluster would be as follows:

- Index into the map at entry = file_number;
- The word in the map at that index is the next cluster number;
- If the next cluster number is zero, you are done;
- If not, feed the number into the above equation as the new file_number;
- Access the disc at the address to get the data for the next cluster;
- Repeat until you hit a zero word in the map.

The map may extend up to a total of 65536 cluster entries, but it may have less than this. This is the maximum number and if a QXL file is formatted then the cluster size in the header, `qwa_sctg`, will be incremented until the map contains 65536 or less, entries.

The Data Area

Files on the disc are saved to this part of the file, and are saved in clusters of a number of sectors, each of which is 512 bytes in size. The header defines how many of these sectors are to be found in each cluster, amongst other things.

Data files cannot be smaller in size than a single cluster. And a cluster is bigger depending on the size of the hard disc file itself.

The map holds the file's number and this is also the pointer to the first cluster of the file's data on the hard disc.

The first 64 bytes of the first cluster of every data file or directory, contain what is supposed to be a copy of the file's entry in the directory, the file header, but may contain whatever random garbage that was in the cluster when it was last in use as part of a file.

Directories

Directories are where file information is kept. Each entry is 64_{dec} or $\$40_{hex}$ bytes in size and there is a copy of this entry, supposedly, at the start of each file's first cluster. This doesn't always appear to be the case. This "spare" 64 bytes at the start applies to the directory as well - the first 64 bytes are not part of the directory structure itself, at least, not in a meaningful way.

The structure of a directory entry is as shown in the following table and applies to the root directory and any subsequently created sub-directories.

Field	Offset	Size	Purpose
Hdr_flen	$0000	Long	File length, including an additional 64 bytes for the header.
Hdr_accs	$0004	Byte	Access control byte - usually zero.
Hdr_type	$0005	Byte	File Type.
Hdr_data	$0006	Long	Program dataspace if executable file.
Hdr_xtra	$000A	Long	Extra information.
Hdr_name	$000E	2 + 36	First word is the name size, then 36 bytes for the space padded name.
Hdr_date	$0034	Long	Update date.
Hdr_vers	$0038	Word	Version number.
Hdr_flid	$003A	Word	File number.
Hdr_bkup	$003C	Long	backup date.

Directory entries for files which have been deleted are not removed from the directory, only the file's length (`hdr_flen`) and the word defining the size of the file name (at `hdr_name`) are zero'd to indicate a deleted file.

A file's access byte is usually always zero.

A file's type byte (`hdr_type`) indicates whether the file is a directory (255), a data file (0), an executable file (1), or an object module awaiting the linker to create an executable file (2). Other file types have been made available over the years, but the ones above are the standard ones.