

# **VMware Virtual Disk (VMDK) format specification**

*Analysis of the VMware Virtual Disk (VMDK) format*

By Joachim Metz <joachim.metz@gmail.com>

## Summary

A VMware Virtual Disk (VMDK) file is used by VMware virtualization software to contain storage media data. This specification is based on the VMDK specification by VMware and was complimented by reverse engineering.

This document is intended as a working document for the VMDK file format specification. Which should allow existing Open Source forensic tooling to be able to process this file type.

## Document information

**Author(s):** Joachim Metz <joachim.metz@gmail.com>

**Abstract:** This document contains information about the VMware Virtual Disk file format

**Classification:** Public

**Keywords:** VMDK, Virtual Disk File, COWD, copy-on-write disk file

## License

Copyright (c) 2009-2014, Joachim Metz <joachim.metz@gmail.com>.  
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

## Version

Version	Author	Date	Comments
0.0.1	J.B. Metz	September 2009	Initial version.
0.0.2	J.B. Metz	September 2012	Additional information, email and license update.
0.0.3	J.B. Metz	September 2012	Additional information.
0.0.4	J.B. Metz	May 2013	Additional information and small changes.
0.0.5	J.B. Metz	May 2013	Additional information from VMWare Player 9 images.
0.0.6	J.B. Metz	October 2013	Additional information regarding descriptor files.
0.0.7	J.B. Metz	December 2013	Additional information regarding compressed VMDK sparse extent data files, with thanks to V. Ramasamy.
0.0.8	J.B. Metz	April 2014	Additional information regarding Stream-Optimized Compressed Sparse Extent that contain GD_AT_END and VMDK version 3.
0.0.9	J.B. Metz	April 2014	Additional information regarding corruption scenarios.
0.0.10	J.B. Metz	July 2014	Additional information regarding descriptor file and change track file.

# Table of Contents

1. Overview.....	1
1.1. Test version.....	1
1.2. Disk types.....	1
1.3. Delta links.....	2
2. The descriptor file.....	2
2.1. Header.....	2
2.1.1. Encodings.....	3
2.1.2. Disk type.....	3
2.2. Extent descriptions.....	4
2.2.1. Extent descriptor.....	5
2.2.2. Extent access mode.....	5
2.2.3. Extent type.....	5
2.3. Change tracking file section.....	6
2.4. Disk database.....	6
2.4.1. Virtual hardware version.....	7
2.4.2. The disk adapter type.....	7
3. The RAW extent data file.....	8
4. The VMDK sparse extent data file.....	8
4.1. File header.....	8
4.1.1. Flags.....	10
4.1.2. Compression method.....	10
4.2. Markers.....	10
4.3. The marker.....	11
4.3.1. Marker type.....	11
4.3.2. Compressed grain marker.....	11
4.3.3. End of stream marker.....	12
4.3.4. Grain table marker.....	12
4.3.5. Grain directory marker.....	12
4.3.6. Footer marker.....	12
4.3.7. Footer.....	13
4.3.8. Notes.....	13
4.4. Descriptor.....	13
4.5. Grain directory.....	13
4.6. Grain table.....	14
4.7. Grain data.....	14
4.7.1. Compressed grain data.....	14
4.8. Changed block tracking (CBT).....	15
5. The COWD sparse extent data file.....	15
5.1. File header.....	15
5.2. Grain directory.....	16
5.3. Grain table.....	16
6. Change tracking file.....	17
7. Corruption scenarios.....	17
Appendix A. References.....	18
Appendix B. GNU Free Documentation License.....	18

# 1. Overview

A VMware Virtual Disk (VMDK) files are used by VMware virtualization software to contain storage media data.

A VMDK disk image consist of multiple files:

- descriptor file
- extent data files
  - RAW extent data file
  - VMDK sparse extent data file
  - COWD sparse extent data file

Characteristics	Description
Byte order	little-endian
Date and time values	
Character string	Narrow character string. Assumed to be UTF-8 by default other encodings are indicated in descriptor file.

A sector is considered 512 bytes.

## 1.1. Test version

The following version of programs were used to test the information within this document:

- VMWare Player 9

## 1.2. Disk types

The 2GbMaxExtentFlat (or twoGbMaxExtentFlat) disk image consists of:

- a descriptor file (<name>.vmdk)
- RAW data extent files (<name>-f###.vmdk), where ### is contains a decimal value starting with 1.

The 2GbMaxExtentSparse (or twoGbMaxExtentSparse) disk image consists of:

- a descriptor file (<name>.vmdk)
- VMDK sparse data extent files (<name>-s###.vmdk), where ### is contains a decimal value starting with 1.

The monolithicFlat disk image consists of:

- a descriptor file (<name>.vmdk)
- RAW data extent file (<name>-f001.vmdk)

The monolithicSparse disk image consists of:

- VMDK sparse data extent file (<name>.vmdk) also contains the descriptor file data.

The vmfs disk image consists of:

- a descriptor file (<name>.vmdk)
- RAW data extent file (<name>-flat.vmdk)

The vmfsSparse differential disk image consists of:

- a descriptor file (<name>.vmdk)
- COWD sparse data extent files (<name>-delta.vmdk)

TODO describe more disk types

### 1.3. Delta links

A delta link is similar to a differential image where the image contains the changes (or delta) in comparison of a parent image. According to [VMDK] one delta image can chain to another delta image.

Name <name>-delta.vmdk

## 2. The descriptor file

The descriptor file is a text based file that contains the following information:

- comment and empty lines (optional)
- the header
- the extent descriptions
- the change tracking file
- the disk database (DDB)

Note that the descriptor file can contains leading and trailing whitespace. Lines are separated by a line feed character (0x0a). And leading comment (starting with #) and empty lines.

### 2.1. Header

The header of a descriptor file looks similar to the data below.

```
# Disk DescriptorFile
version=1
CID=12345678
parentCID=ffffffff
createType="twoGbMaxExtentSparse"
```

The header consists of the following values:

Value	Description
# Disk DescriptorFile	File signature Section header
version	The format version 1, 2 or 3
encoding	The used string encoding (for the descriptor file) See section: 2.1.1 Encodings
CID	Content identifier A random 32-bit value updated the first time the content of the virtual disk is modified after the virtual disk is opened. A value of 'fffffffe' (-2) represents that the long

Value	Description
	content identifier should be used?
parentCID	The content identifier of the parent A 32-bit value identifying the parent content. A value of 'ffffffff' (-1) represents no parent content.
isNativeSnapshot	<b>TODO</b> Seen values “no” Seen in VMWare Player 9 descriptor file uncertain when this was introduced
createType	The disk type See section: 2.1.2 Disk type
parentFileNameHint	Contains the path to the parent image. This value is only present if the image is a differential image (delta link).

### 2.1.1. Encodings

It is unknown which encodings are supported, currently it is assumed that at least the Windows codepages are supported and that the default is UTF-8.

Value	Description
UTF-8	UTF-8
windows-1252	Windows codepage 1252 Seen in VMWare Player 9 descriptor file uncertain when this was introduced.

### 2.1.2. Disk type

Value	Description
2GbMaxExtentFlat (twoGbMaxExtentFlat)	The disk is split into fixed-size extents of maximum 2 GB. The extents consists of RAW extent data files.
2GbMaxExtentSparse (twoGbMaxExtentSparse)	The disk is split into sparse (dynamic-size) extents of maximum 2 GB. The extents consists of VMDK sparse extent data files.
custom	<b>TODO</b> Descriptor file with arbitrary extents , used to mount v2i-format.
fullDevice	The disk uses a full physical disk device.
monolithicFlat	The disk is a single RAW extent data file.
monolithicSparse	The disk is a single VMDK sparse extent data

Value	Description
	file.
partitionedDevice	The disk uses a full physical disk device, using access per partition.
streamOptimized	The disk is a single compressed VMDK sparse extent data file. (Unknown if more than one extent data file is allowed)  Note from [VMDK] Compressed sparse extents with embedded LBA, useful for OVF streaming.
vmfs	The disk is a single RAW extent data file. This is similar to the "monolithicFlat". The maximum size depends on the block size used to format the VMFS3.
vmfsEagerZeroedThick	The disk is a single RAW extent data file. The disk is pre-allocated on VMFS, with all blocks zeroed when created.
vmfsPreallocated	The disk is a single RAW extent data file. The disk is pre-allocated on VMFS, with blocks zeroed on first use.
vmfsRaw	The disk uses a full physical disk device. Special raw disk for ESXi hosts, pass through only mode.
vmfsRDM (vmfsRawDeviceMap)	The disk uses a full physical disk device. Also referred to as Raw Device Map (RDM).
vmfsRDMP (vmfsPassthroughRawDeviceMap)	The disk uses a full physical disk device. Similar to the Raw Device Map (RDM), but sends SCSI commands to underlying hardware.
vmfsSparse	The disk is split into sparse (dynamic-size) extents. The extents consists of COWD sparse extent data files. Often used as a redo-log
vmfsThin	The disk is split into sparse (dynamic-size) extents. The extents consists of COWD sparse extent data files.

## 2.2. Extent descriptions

The extent descriptions of a VMDK descriptor file looks similar to the data below.

```
# Extent description
RW 4192256 SPARSE "test-s001.vmdk"
```

```
# Extent description
RW 1048576 FLAT "test-f001.vmdk" 0
```

The extent descriptions consists of the following values:



Value	Description
# Extent description	Section header
	Extent descriptors

### 2.2.1. Extent descriptor

The extent descriptor consists of the following values:

Value	Description
1 <sup>st</sup>	The access mode See section: 2.2.2 Extent access mode
2 <sup>nd</sup>	The number of sectors Likely 512 bytes per sector is always assumed
3 <sup>rd</sup>	The extent type See section: 2.2.3 Extent type
<i>If extent type is not ZERO</i>	
4 <sup>th</sup>	The filename of the VMDK extent data file The filename is relative to the location of the VMDK descriptor file
<i>Optional</i>	
5 <sup>th</sup>	The extent start sector Likely 512 bytes per sector is always assumed
<i>Seen in VMWare Player 9 in combination with a physical device extent on Windows</i>	
6 <sup>th</sup>	PartitionUUID
7 <sup>th</sup>	Device identifier

The extent offset is specified only for flat extents and corresponds to the offset in the file or device where the extent data is located. For device-backed virtual disks (physical or raw disks) the extent offset can be non-zero. For RAW extent data files the extent offset should be zero.

### 2.2.2. Extent access mode

The extent access mode consists of the following values:

Value	Description
NOACCESS	No access
RDONLY	Read only
RW	Read write

### 2.2.3. Extent type

The extent type consists of the following values:

Value	Description
FLAT	RAW extent data file

Value	Description
	Seen in VMWare Player 9 to be also used for devices on Windows
SPARSE	VMDK sparse extent data file
ZERO	Sparse extent that consists of 0-byte values
VMFS	RAW extent data file
VMFSSPARSE	COWD sparse extent data file
VMFSRDM	TODO Physical disk device that uses RDM?
VMFSRAW	TODO Physical disk device?

### 2.3. Change tracking file section

The change tracking file section was introduced in version 3 and looks similar to:

```
# Change Tracking File
changeTrackPath="test-flat.vmdk"
```

The change tracking file section consists of the following values:

Value	Description
# Change Tracking File	Section header
changeTrackPath	TODO The path to the change tracking file?

### 2.4. Disk database

The disk database of a VMDK descriptor file looks similar to the data below.

```
# The Disk Data Base
#DDB

ddb.virtualHWVersion = "4"
ddb.geometry.cylinders = "16383"
ddb.geometry.heads = "16"
ddb.geometry.sectors = "63"
ddb.adapterType = "ide"
ddb.toolsVersion = "0"
```

The disk database consists of the following values:

Value	Description
# The Disk Data Base #DDB	Section header
ddb.deletable	TODO "true"
ddb.virtualHWVersion	The virtual hardware version For VMWare Player and Workstation this seems

Value	Description
	to correspond with the application version
ddb.longContentID	The long content identifier 128-bit base16 encoded value, without spaces
ddb.uuid	Unique identifier 128-bit base16 encoded value, with spaces between bytes
ddb.geometry.cylinders	The number of cylinders
ddb.geometry.heads	The number of heads
ddb.geometry.sectors	The number of sectors
ddb.geometry.biosCylinders	The number of cylinders as reported by the BIOS Seen in VMWare Player 9 for a Device
ddb.geometry.biosHeads	The number of heads as reported by the BIOS Seen in VMWare Player 9 for a Device
ddb.geometry.biosSectors	The number of sectors as reported by the BIOS Seen in VMWare Player 9 for a Device
ddb.adapterType	The disk adapter type See section: 2.4.2 The disk adapter type
ddb.toolsVersion	TODO String containing the version of the installed VMWare tools version
ddb.thinProvisioned	TODO "1"

#### 2.4.1. Virtual hardware version

Value	Description
4	
6	
7	
9	VMWare Player/Workstation 9.0

#### 2.4.2. The disk adapter type

Value	Description
ide	TODO
buslogic	TODO
lsilogic	TODO

Value	Description
legacyESX	TODO

The buslogic and lsilogic values are for SCSI disks and show which virtual SCSI adapter is configured for the virtual machine. The legacyESX value is for older ESX Server virtual machines when the adapter type used in creating the virtual machine is not known.

### 3. The RAW extent data file

The RAW extent data file contains the actual disk data. The RAW extent data file can be a file or a device.

This type of extent data file is also known as Simple or Flat Extent.

### 4. The VMDK sparse extent data file

The VMDK sparse extent data file contains the actual disk data. The VMDK sparse extent data file consists of the following distinguishable elements:

- file header
- optional embedded descriptor
- secondary grain directory
  - secondary grain tables
- (primary) grain directory
  - (primary) grain tables
- grains

This type of extent data file is also known as Hosted Sparse Extent or Stream-Optimized Compressed Sparse Extent when markers are used.

Note that the actual layout can vary per file, e.g. Stream-Optimized Compressed Sparse Extent have seen to use secondary file headers.

Changes in version 2:

- added encrypted disk support (though this feature never seem to never have been implemented).

Changes in version 3:

- the size of extent files is no longer limited to 2 GiB;
- added support for persistent changed block tracking (CBT).

**Note:** CBT: the changeTrackPath setting in the descriptor file references a file that describes changed areas on the virtual disk.

#### 4.1. File header

The file header is 512 bytes of size and consists of:

offset	size	value	description
0	4	“KDMV”	Signature

offset	size	value	description
4	4	1, 2 or 3	Version
8	4		Flags See section: 4.1.1 Flags
12	8		Maximum data number of sectors (capacity)
20	8		Grain number of sectors The value must be a power of 2 and > 8
28	8		Descriptor sector number The sector number of the embedded descriptor file. The value is relative from the start of the file or 0 if not set.
36	8		Descriptor number of sectors The number of sectors of the embedded descriptor in the extent data file.
44	4	512	The number of grains table entries
48	8		Secondary (redundant) grain directory sector number The value is relative from the start of the file or 0 if not set.
56	8		Grain directory sector number The value is relative from the start of the file or 0 if not set.  Note that the value can be -1 see below for more information.
64	8		Metadata (overhead) number of sectors
72	1		Is dirty Value to determine if the extent data file was cleanly closed.
73	1	'\n'	Single end of line character
74	1	' '	Non end of line character
75	1	'\r'	First double end of line character
76	1	'\n'	Second double end of line character
77	2		Compression method
79	433	0	Padding

The end of line characters are used to detect corruption due to file transfers that alter line end characters.

According to [VMDK] the maximum data number of sectors (capacity) should be a multitude of the grain number of sectors. Note that this is not always the case.

If the grain directory sector number value is -1 (0xffffffffffffff) (GD\_AT\_END) in a Stream-

Optimized Compressed Sparse Extent there should be a secondary file header stored at offset -1024 relative from the end of the file (stream) that contains the correct grain directory sector number value.

#### 4.1.1. Flags

The flags consist of the following values:

Value	Identifier	Description
0x00000001		Valid new line detection test
0x00000002		Use secondary grain directory The secondary (redundant) grain directory should be used instead of the primary grain directory.
<i>As of format version 2</i>		
0x00000004		Use zeroed-grain table entry The zeroed-grain table entry overloads grain data sector number 1 to indicate the grain is sparse
<i>Common</i>		
0x00010000		Has compressed grain data The type of compression is described by compression algorithm.  Only used in combination with disk type: streamOptimized?
0x00020000		Has metadata The disk contains markers to identify every block of metadata or data and the markers for the virtual machine data contain a LBA  Only used in combination with disk type: streamOptimized?

#### 4.1.2. Compression method

The compression method consist of the following values:

Value	Identifier	Description
0x00000000	COMPRESSION_NONE	No compression
0x00000001	COMPRESSION_DEFLATE	Compression using deflate (RFC 1951)

#### 4.2. Markers

The markers are used in Stream-Optimized Compressed Sparse Extents. The corresponding flag must be set for markers to be present. An example of the layout of a Stream-Optimized Compressed Sparse Extent that uses markers is:

- file header
- embedded descriptor

- compressed grain markers
- grain table marker
- grain table
- grain directory marker
- grain directory
- footer marker
- secondary file header
- end-of-stream marker

### 4.3. The marker

The marker is 512 bytes of size and consists of:

offset	size	value	description
0	8		Value
8	4		Marker data size
<i>If marker data size equals 0</i>			
12	4		Marker type
16	496	0	Padding Unused bytes are set to 0.
<i>If marker data size &gt; 0</i>			
12	...		Compressed grain data

If the marker data size > 0 the marker is a compressed grain marker.

#### 4.3.1. Marker type

The marker type consist of the following values:

Value	Identifier	Description
0x00000000	MARKER_EOS	End-of-stream marker
0x00000001	MARKER_GT	Grain table (metadata) marker
0x00000002	MARKER_GD	Grain directory (metadata) marker
0x00000003	MARKER_FOOTER	Footer (metadata) marker

#### 4.3.2. Compressed grain marker

The compressed grain marker indicated that compressed data follows.

offset	size	value	description
0	8	0	Sector number where the block of compressed data is located within the virtual disk
8	4	> 0	Compressed grain data size
12	...		Compressed grain data Decompress with deflate (RFC 1951).

Note that the compressed grain data can be larger than the grain data size.

### 4.3.3. End of stream marker

The end-of-stream marker indicated the end of the virtual disk. Basically the end-of-stream marker is an empty sector block.

offset	size	value	description
0	8	0	Value
8	4	0	Marker data size
12	4	MARKER_EOS	Marker type See section: 4.3.1 Marker type
16	496	0	Padding

### 4.3.4. Grain table marker

The grain table marker indicates that a grain table follows the marker sector block.

offset	size	value	description
0	8	0	Value
8	4	0	Marker data size
12	4	MARKER_GT	Marker type See section: 4.3.1 Marker type
16	496	0	Padding
512	...		Grain table See section: 4.6 Grain table

### 4.3.5. Grain directory marker

The grain directory marker indicates that a grain directory follows the marker sector block.

offset	size	value	description
0	8	0	Value
8	4	0	Marker data size
12	4	MARKER_GD	Marker type See section: 4.3.1 Marker type
16	496	0	Padding
512	...		Grain directory See section: 4.5 Grain directory

### 4.3.6. Footer marker

The footer marker indicates that a footer follows the marker sector block.



offset	size	value	description
0	8	0	Value
8	4	0	Marker data size
12	4	MARKER_FOOTER	Marker type See section: 4.3.1 Marker type
16	496	0	Padding
512	...		Footer See section: 4.3.7 Footer

### 4.3.7. Footer

The footer is only used in Stream-Optimized Compressed Sparse Extents. The footer is the same as the file header. The footer should be the last block of the disk and immediately followed by the end-of-stream marker so that they together make up the last two sectors of the disk.

The header and footer differ in that the grain directory offset value in the header is set to -1 (0xffffffffffff) (GD\_AT\_END) and in the footer to the correct value.

### 4.3.8. Notes

The markers can be used to scan for the individual parts of the VMDK sparse extent data file if the stream has been truncated, but not that this can be very expensive process IO-wise.

## 4.4. Descriptor

Contains data similar to the descriptor file. See section: 2 The descriptor file.

## 4.5. Grain directory

The grain directory is also referred to as level-0 metadata.

The size of the grain directory is dependent on the number of grains in the extent data file. The number of entries in the grain directory can be determined as following:

```

number of grain directory entries = maximum data size
                                / ( number of grain table entries x grain
size )
if( maximum data size % ( number of grain table entries x grain size ) > 0 )
{
    number of entries += 1
}

```

The grain directory consists of 32-bit grain table offsets:

offset	size	value	description
0	4		Grain table sector number The value is relative from the start of the file or 0 if not set.

The grain directory is stored in a multitude of 512 byte sized blocks.

- A sector number of 0 indicates a the grain table is sparse or should be read from the parent.
- As of VMDK sparse extent data file version 2 if the “use zeroed-grain table entry” flag is set a sector number of 1 indicates the grain table is sparse.
- Any other value point to a sector number in the VMDK sparse extent data file.

## 4.6. Grain table

The grain table is also referred to as level-1 metadata.

The size of the grain table is variable of size. The number of entries in the grain table is stored in the file header. Note that the number of entries in the last grain table is dependent on the maximum data size and not necessarily the same as the value stored in the file header.

The grain directory consists of 32-bit grain table offsets:

offset	size	value	description
0	4		Grain data sector number The value is relative from the start of the file or 0 if not set.

The number of entries in a grain table and should be 512, therefore the size of the grain table is  $512 \times 4 = 2048$  bytes.

The grain table is stored in a multitude of 512 byte sized blocks.

- A sector number of 0 indicates a the grain data is sparse or should be read from the parent.
- As of VMDK sparse extent data file version 2 if the “use zeroed-grain table entry” flag is set a sector number of 1 indicates the grain data is sparse.
- Any other value point to a sector number in the VMDK sparse extent data file.

## 4.7. Grain data

In an uncompressed sparse extent data file the data is stored at the grain data sector number.

In a compressed sparse extent data file every non-sparse grain is (assumed to be) stored compressed.

### 4.7.1. Compressed grain data

The compressed grain data is variable of size and consists of:

offset	size	value	description
0	8		Media data sector number
8	4		Compressed data size
12	...		Compressed data Contains ZLIB compressed data (DEFLATE + ZLIB header)
...	...		Padding Unknown if this should be always 0-byte

offset	size	value	description
			values

The uncompressed data size should be the grain size or less for the last grain.

#### 4.8. Changed block tracking (CBT)

TODO need example data.

## 5. The COWD sparse extent data file

The copy-on-write disk (COWD) sparse extent data file contains the actual disk data. The COW sparse extent data file consists of the following distinguishable elements:

- file header
- grain directory
  - grain tables
- grains

This type of extent data file is also known as ESX Server Sparse Extent.

#### 5.1. File header

The file header is 2048 bytes of size and consists of:

offset	size	value	description
0	4	“COWD”	signature
4	4	1	Version
8	4	0x00000003	Flags (Unknown)
12	4		Maximum data number of sectors (capacity)
16	4		Grain number of sectors
20	4	4	Grain directory sector number The value is relative from the start of the file or 0 if not set.
24	4		Number of grain directory entries
28	4		The next free sector
<i>In root extent data file</i>			
32	4		The number of cylinders
36	4		The number of heads
40	4		The number of sectors
44	1016		Empty values
<i>In child extent data file</i>			
32	1024		Parent filename

offset	size	value	description
			UTF-8 or ASCII string with codepage?
1056	4		Parent generation
<i>Common</i>			
1060	4		Generation
1064	60		Name UTF-8 or ASCII string with codepage?
1124	512		Description UTF-8 or ASCII string with codepage?
1636	4		Saved generation
1640	8		Reserved
1648	4		Is dirty Value to determine if the extent data file was cleanly closed.
1652	396		Padding

Note that the parent filename seems not to be set in recent delta sparse extent files.

## 5.2. Grain directory

The grain directory is also referred to as level-0 metadata.

The size of the grain directory is dependent on the number of grains in the extent data file. The number of entries in the grain directory is stored in the file header.

The grain directory consists of 32-bit grain table offsets:

offset	size	value	description
0	4		Grain table sector number The value is relative from the start of the file or 0 if not set.

The grain directory is stored in a multitude of 512 byte sized blocks. Unused bytes are set to 0.

## 5.3. Grain table

The grain table is also referred to as level-1 metadata.

The size of the grain table is variable of size. The number of entries in a grain table is the fixed value of 4096.

The grain directory consists of 32-bit grain table offsets:

offset	size	value	description
0	4		Grain sector number The value is relative from the start of the

offset	size	value	description
			file or 0 if not set.

The grain table is stored in a multitude of 512 byte sized blocks. Unused bytes are set to 0.

## 6. Change tracking file

TODO; need more samples

offset	size	value	description
0	4	“\xa2\x72\x19\xf6”	Unknown (signature?)
4	4	1	Unknown (version?)
8	4		Unknown (empty values)
12	4	0x200	Unknown
16	8		Unknown
24	8		Unknown
32	4		Unknown
36	4		Unknown
40	4		Unknown
44	16		Unknown (GUID?)
60	...		Unknown (empty values?)

## 7. Corruption scenarios

The total size specified by the number of grain table entries is larger than size specified by the maximum number of sectors. Seen in VMDK images generated by qemu-img.

## Appendix A. References

[RFC1950]

Title: ZLIB Compressed Data Format Specification  
Version: 3.3  
Author(s): P. Deutsch, J-L. Gailly  
Date: May 1996  
URL: <http://www.ietf.org/rfc/rfc1950.txt>

[RFC1951]

Title: DEFLATE Compressed Data Format Specification  
Version: 1.3  
Author(s): P. Deutsch  
Date: May 1996  
URL: <http://www.ietf.org/rfc/rfc1951.txt>

[VMDK]

Title: Virtual Disk Format  
Author(s): WMWare  
Version: 1.1  
URL: <http://www.vmware.com/app/vmdk/?src=vmdk>

Title: Virtual Disk Format  
Author(s): WMWare  
Version: 5.0  
URL: <http://www.vmware.com/app/vmdk/?src=vmdk>

## Appendix B. GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.  
<<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

### 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any

textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

## **1. APPLICABILITY AND DEFINITIONS**

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in

formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

## **2. VERBATIM COPYING**

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

## **3. COPYING IN QUANTITY**

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated



version of the Document.

#### **4. MODIFICATIONS**

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the

Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

## **5. COMBINING DOCUMENTS**

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

## **6. COLLECTIONS OF DOCUMENTS**

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

## **7. AGGREGATION WITH INDEPENDENT WORKS**

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative

works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

## **8. TRANSLATION**

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

## **9. TERMINATION**

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

## **10. FUTURE REVISIONS OF THIS LICENSE**

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions

of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

## **11. RELICENSING**

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.