

The structure of a PKZip file

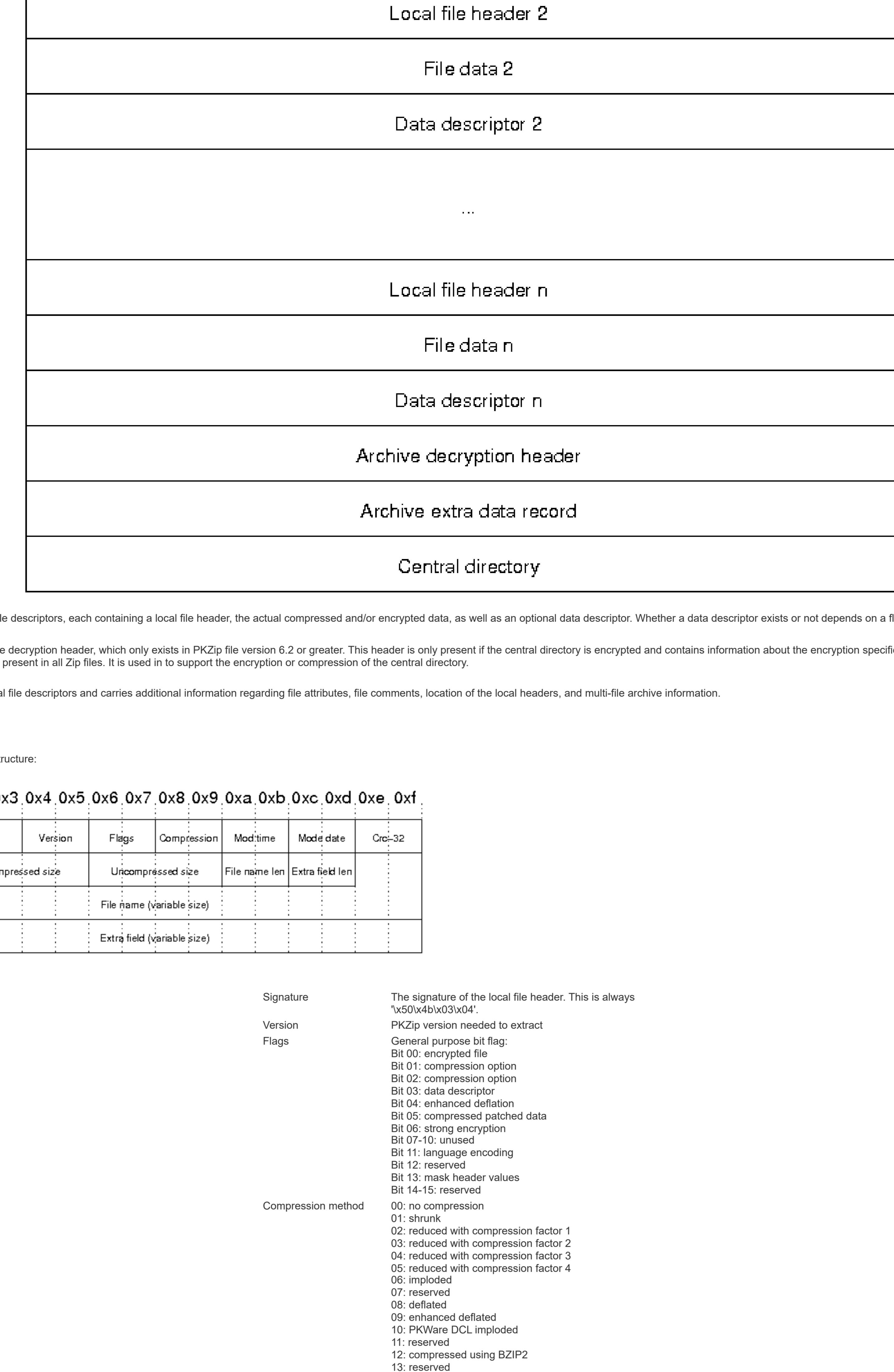
by Florian Buchholz

General structure
Local file headers
Data descriptor
Archive decryption header
Archive extra data record
Central directory
Printable Version

This document describes the on-disk structure of a PKZip (Zp) file. The documentation currently only describes the file layout format and meta information but does not address the actual compression or encryption of the file data itself. This documentation also does not discuss Zip archives that span multiple files in great detail. This documentation was created using the [official documentation](#) provided by PKWare Inc.

General structure

Each Zip file is structured in the following manner:



The archive consists of a series of local file descriptors, each containing a local file header, the actual compressed and/or encrypted data, as well as an optional data descriptor. Whether a data descriptor exists or not depends on a flag in the local file header.

Following the file descriptors is the archive decryption header, which only exists in PKZip file version 6.2 or greater. This header is only present if the central directory is encrypted and contains information about the encryption specification. The archive extra data record is also only for file version 6.2 or greater and is not present in all Zip files. It is used to support the encryption or compression of the central directory.

The central directory summarizes the local file descriptors and carries additional information regarding file attributes, file comments, location of the local headers, and multi-file archive information.

Local file headers

Each local file header has the following structure:

	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
0x0000	Signature	Version	Vers. needed	Flags	Compression	Modtime	Moddate									Crc-32
0x0010	Crc-32	Compressed size	Uncompressed size	File name len	Extra field len											
0x0020				File name (variable size)												
0x0030				Extra field (variable size)												

Signature	The signature of the local file header. This is always "v50x4b0x030x04".
Version	0x1 = 20 -> 2.0
Flags	no flags
Compression method	0x0: deflated
File modification time	0x7d1c = 0111110100011100 hour = 0111110100011100 = 15 minute = 0010101010101011 = 40 second = 0111110100011100 = 28 = 56 seconds 15:40:56
File modification date	0x0100 = 010101010101010111 year = 001101010101010111 = 26 month = 001101010101010111 = 10 day = 001101010101010111 = 11 10/11/2006
Crc-32 checksum	0x7d90fa16
Compressed size	0x45 = 69 bytes
Uncompressed size	0x4a = 74 bytes
File name length	0x04 = 4 bytes
Extra field length	0x01 = 1 byte
File name	"file1"
Extra field	Used to store additional information. The field consists of a sequence of header and data pairs, where the header has a 2 byte identifier and a 2 byte data size field.

Example

Our sample zip file starts with a local file header:

```
00000000 50 4b 03 04 14 00 00 00 08 01 7d 4b 35 a1 [PK.....]1K5...]
00000010 90 7d 45 00 00 00 4a 00 00 00 05 00 1c 7d 4b 35 a1 [J.....J.....]
00000020 6c 65 31 55 54 09 00 03 c7 48 2d 45 55 78 00 00 66 69
00000030 55 78 04 00 f1 f5 01 00 c9 c8 2c 56 00 00 92 [U.....V..-]
```

This results in the following fields and field values:

	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
0x0000	50	4b	03	04	14	00	00	08	00	1c	7d	4b	35	a1		
0x0010	90	7d	45	00	00	00	4a	00	00	00	05	00	15	00	66	69
0x0020	6c	65	31	55	54	09	00	03	c7	48	2d	45	55	78	00	00
0x0030	55	78	04	00	f1	f5	01	00	c9	c8	2c	56	00	00	92	

Data descriptor

The data descriptor is only present if bit 3 of the bit flag field is set. In this case, the CRC-32, compressed size, and uncompressed size fields in the local header are set to zero. The data descriptor field is byte aligned and immediately follows the file data. The structure is as follows:

	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb				
0x0000	0x0															
0x0010																

The example file does not contain a data descriptor.

Archive decryption header

This header is used to support the Central Directory Encryption Feature. It is present when the central directory is encrypted. The format of this data record is identical to the Decryption header record preceding compressed file data.

The structure is as follows:

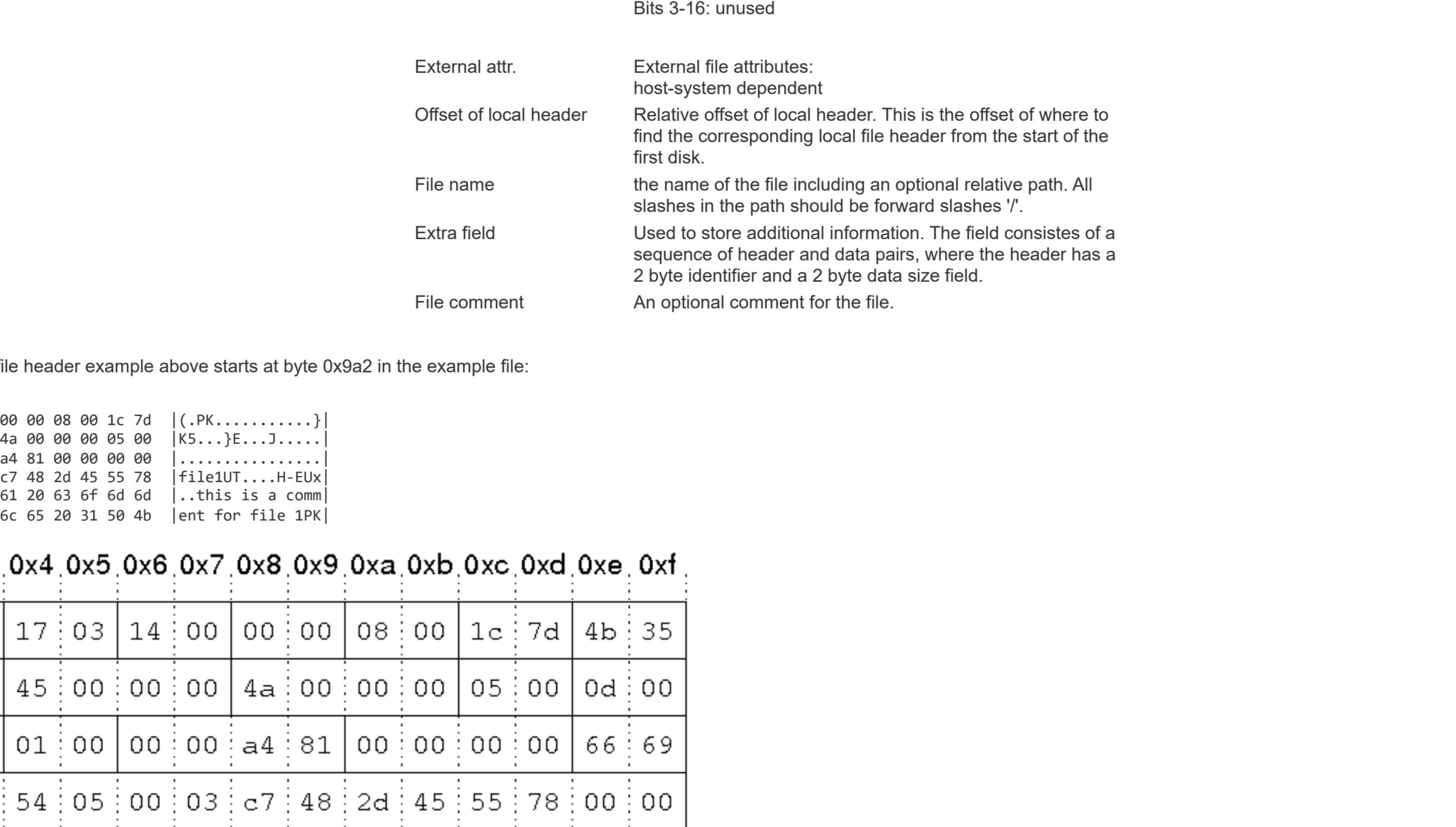
	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
0x0000	Signature	Version	Vers. needed	Flags	Compression	Modtime	Moddate									
0x0010	Crc-32	Compressed size	Uncompressed size	File name len	Extra field len											

Archive extra data record

This header is used to support the Central Directory Encryption Feature. When present, this record immediately precedes the central directory data structure. The size of this data record will be included in the Size of the Central Directory field in the End of Central Directory record.

Central directory

The central directory contains more metadata about the files in the archive and also contains encryption information and information about Zip64 (64-bit Zip archives). Furthermore, the central directory contains information about archives that span multiple files. The structure of the central directory is as follows:



The file headers are similar to the local file headers, but contain some extra information. The Zip64 entries handle the case of a 64-bit Zip archive, and the end of the central directory record contains information about the archive itself.

Central directory file header

The structure of the file header in the central directory is as follows:

	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
0x0000	Signature	Version	Vers. needed	Flags	Compression	Modtime	Moddate									
0x0010	Crc-32	Compressed size	Uncompressed size	File name len	Extra field len											
0x0020	File comment len	Disk # start	Internal attr.	Offset of local header												
0x0030				File name (variable size)												
0x0040				Extra field (variable size)												
0x0050				File comment (variable)												

End of central directory record

The structure of the end of central directory record is as follows:

	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
0x0000	Signature	Version	V													