INTERNATIONAL STANDARD

ISO/IEC 13346-2

First edition
1995-12-15

Information technology — Volume and file structure of write-once and rewritable media using non-sequential recording for information interchange —

Part 2:
Volume and boot block recognition

Technologies de l'information — Structure de volume et de fichier de moyens d'écriture unique et de réécriture utilisant un enregistrement non séquentiel pour l'échange d'information —

Partie 2: Reconnaissance de volume et de «boot block»

Reference number
ISO/IEC 13346-2:1995(E)

Contents ..... Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialised system for worldwide standardisation. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organisation to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organisations, governmental and non-governmental, in liaison with ISO and IEC, also take part in this work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication of an International Standard requires approval by at least 75% of the national bodies casting a vote.

International Standard ISO/IEC 13346 was prepared by the European Association for Standardizing Information and Communication Systems, ECMA, (as Standard ECMA-167) and was adopted, under a special "fast-track procedure", by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by National Bodies of ISO and IEC.

Introduction

ISO/IEC 13346 is a volume and file structure standard for interchanging files and as such, it is a peer to existing volume and file structure standards such as ISO 9293 and ISO 9660. It is rather different from those standards in at least two important ways. Firstly, it offers much more functionality, mainly because of user needs for increased character set support and for more powerful file system features. Secondly, it acknowledges the separate concerns of booting, volume structure and file system structure. Rather than bundling these different functions together, ISO/IEC 13346 carefully segregates these functions into separate parts and describes in detail how those parts fit together. It is expected that future volume and file structure standards will fit into this framework, rather than building other distinct and incompatible formats.

ISO/IEC 13346 is published in five Parts. Part 1 - general - specifies references, definitions, notations and basic structures used in the other four Parts. This Part of ISO/IEC 13346 - volume and boot block recognition - specifies formats and system requirements for recognising the volume structures on a medium and booting from a medium. Part 3 - volume structure - specifies how to record various volume-related entities such as volumes, volume sets and logical volumes. Part 4 - file structure - specifies how to record and interpret files, both file data and file attributes, and file hierarchies within logical volumes. Part 5 - record structure - specifies how to record and interpret file data encoded as records.

This page intentionally left blank

Information technology - Volume and file structure of write-once and rewritable media using non-sequential recording for information interchange -

Part 2:
Volume and boot block recognition

Section 1: General

1 Scope

ISO/IEC 13346 specifies a format and associated system requirements for volume and boot block recognition, volume structure, file structure and record structure for the interchange of information on media between users of information processing systems.

The media shall be recorded as if the recording of sectors may be done in any order.

NOTE 1 - The medium is not restricted to being of only one type; the type of medium may be either write once, or read only, or rewritable, or a combination of these types.

ISO/IEC 13346 consists of the following five Parts:

Part 1: General

Part 2: Volume and Boot Block Recognition

Part 3: Volume Structure

Part 4: File Structure

Part 5: Record Structure

Annex A - ICB Strategies, is part of ISO/IEC 13346-4.

This Part of ISO/IEC 13346 specifies a format and associated system requirements for volume and boot block recognition by specifying:

- volume recognition;
- boot descriptors intended for use to bring a system to a known state;
- levels of medium interchange;
- requirements for the processes which are provided within information processing systems, to enable information to be interchanged between different systems; for this purpose, this Part of ISO/IEC 13346 specifies the functions to be provided within systems which are intended to originate or receive media which conform to this Part of ISO/IEC 13346.

2 Parts references

The first digit of a reference within ISO/IEC 13346 identifies the Part, e.g. 2/5 refers to clause 5 in this Part of ISO/IEC 13346, and figure 4/3 refers to figure 3 in ISO/IEC 13346-4.

3 Part interface

This clause specifies the interface of this Part of ISO/IEC 13346 to other standards or Parts.

3.1 Input

This Part of ISO/IEC 13346 requires the specification of the following by another standard or Part.

- A standard for recording (see 1/5.10).
- The address of the initial sector in the volume (see 2/8.1.1).
- A volume recognition space (see 2/8.2).

3.2 Output

This Part of ISO/IEC 13346 specifies the following which may be used by other standards or Parts.

– identification of certain standards (see 2/9.1.2) used to record information in the volume.

– information that may be used to bring a system to a known state.

4 Conformance

4.1 Conformance of a medium

A medium shall be in conformance with ISO/IEC 13346 when it conforms to a standard for recording (see 1/5.10) and information recorded on sectors of the medium conform to the specifications of ISO/IEC 13346-1 and one or more of Parts 2, 3, 4 and 5. A statement of conformance shall identify the sectors of the medium on which information is recorded according to the specifications of ISO/IEC 13346, and the Parts and the levels of medium interchange (see2/10, 3/11, and 4/15) to which the contents of those sectors of the medium conform.

4.2 Conformance of an information processing system

An information processing system shall be in conformance with ISO/IEC 13346 if it meets the requirements specified in ISO/IEC 13346-1 and one or more of Parts 2, 3, 4 and 5 either for an originating system (see 2/12, 3/13, 4/17 and 5/11) or for a receiving system (see 2/13, 3/14, 4/18 and 5/12) or for both types of system. A statement of conformance shall identify the Parts, and the levels of the requirements for each of those Parts, which can be met by the system.

5 Definitions

For the purposes of this Part of ISO/IEC 13346, the definitions given in ISO/IEC 13346-1 (see 1/5) and the following definition apply.

5.1 **extent:** A set of sectors, the sector numbers of which form a continuous ascending sequence. The address, or location, of an extent is the number of the first sector in the sequence.

6 Notation

The notation of ISO/IEC 13346-1 (see 1/6) applies to this Part of ISO/IEC 13346.

7 Basic types

The basic types of ISO/IEC 13346-1 (see 1/7) apply to this Part of ISO/IEC 13346.

Section 2: Requirements for the medium for volume and boot block recognition

8 Volume recognition

8.1 Arrangement of data on a volume

8.1.1 Sector numbers

Each sector of a volume shall be identified by a unique sector number. Sector numbers shall be consecutive integers assigned in an ascending sequence, in the order of ascending physical address of the volume as specified in the relevant standard for recording (see 1/5.10). Sector number 0 shall be assigned to the initial sector of the volume as specified in 2/3.1.

8.2 Volume recognition space

A volume recognition space shall be a contiguous sequence of sectors. The bytes in the volume recognition space shall be numbered with consecutive integers assigned in ascending sequence. The numbering shall start from 0 which shall be assigned to the first byte of the first sector of the volume recognition space. The numbering shall continue through successive bytes of that sector and then through successive bytes of each successive sector, if any, of the volume recognition space.

8.3 Volume recognition area

A volume recognition area shall be recorded in the volume recognition space. A volume recognition area shall consist of a volume recognition sequence (see 2/8.3.1) recorded in consecutively numbered sectors starting with the first byte of the first sector that begins after byte number 32 767 of the volume recognition space. This Part of ISO/IEC 13346 does not specify the interpretation of the information recorded in the volume recognition space other than in the volume recognition area of the volume recognition space.

8.3.1 Volume recognition sequence

A volume recognition sequence shall consist of a consecutively recorded sequence of one or more Volume Structure Descriptors (see 2/9.1) recorded according to the schema shown in figure 2/1.

Each Volume Structure Descriptor shall specify a standard or clause which shall specify the interpretation of the contents of the descriptor and the value of *n* (see figure 2/1).

The first Volume Structure Descriptor of the sequence shall be recorded beginning at the first byte of the first sector of the volume recognition area in which it is recorded. Each successive Volume Structure Descriptor of the sequence shall be recorded beginning at the first byte of the sector with the next higher sector number than that of the last sector constituting the previous Volume Structure Descriptor of the sequence.

NOTE 2 - The volume recognition sequence is terminated by the first sector which is not a valid descriptor, rather than by an explicit descriptor. This sector might be an unrecorded or blank sector.

```
[volume recognition sequence]{
   <CD-ROM Volume Descriptor Set>0+1
   [Extended Area]{
         <Beginning Extended Area Descriptor> 1+
         { <Volume Structure Descriptor> | <Boot Descriptor> } n+
         <Terminating Extended Area Descriptor> 1+
   } 0+
}
```

Figure 1 - Volume recognition sequence schema

8.3.1.1 CD-ROM Volume Descriptor Set

A CD-ROM Volume Descriptor Set shall be a set of consecutively recorded Volume Structure Descriptors whose Standard Identifier fields shall not contain “BEA01” and shall be interpreted according to ISO 9660.

8.4 Recording of descriptors

All the descriptors in this Part of ISO/IEC 13346 shall be recorded so that the first byte of the descriptor coincides with the first byte of a sector. All space, if any, after the last byte of the descriptor up to the end of the sector containing the last byte of the descriptor is reserved for future standardisation and shall be recorded as all #00 bytes.

9 Volume recognition structures

9.1 Volume Structure Descriptor

The Volume Structure Descriptor shall be recorded in the format shown in figure 2/2.

BP	Length	Name	Contents
0	1	Structure Type	Uint8 (1/7.1.1)
1	5	Standard Identifier	bytes
6	1	Structure Version	Uint8 (1/7.1.1)
7	2 041	Structure Data	bytes

Figure 2 - Generic Volume Structure Descriptor format

9.1.1 Structure Type (BP 0)

The number in this field shall specify the type of the Volume Structure Descriptor. The interpretation of the number shall be specified by the Standard or clause identified in the Standard Identifier field.

9.1.2 Standard Identifier (BP 1)

This field shall specify the interpretation of the Volume Structure Descriptor as shown in figure 2/3.

Identifier	Interpretation
"BEA01"	According to clause 2/9.2.
"BOOT2"	According to clause 2/9.4.
"CD001"	According to ISO 9660.
"CDW02"	According to ISO/IEC 13490.
"NSR02"	According to clause 3/9.1 of ISO/IEC 13346.
"TEA01"	According to clause 2/9.3.

Figure 3 - Volume Structure Descriptor interpretation

All other values are reserved for future standardisation.

9.1.3 Structure Version (BP 6)

The number in this field shall specify the version of the Volume Structure Descriptor. The interpretation of the number shall be specified by the Standard or clause identified in the Standard Identifier field.

9.1.4 Structure Data (BP 7)

The interpretation of this field shall be specified by the Standard or clause identified in the Standard Identifier field.

9.2 Beginning Extended Area Descriptor

The Beginning Extended Area Descriptor shall be recorded in the format shown in figure 2/4.

BP	Length	Name	Contents
0	1	Structure Type	Uint8 (1/7.1.1) = 0
1	5	Standard Identifier	bytes = "BEA01"
6	1	Structure Version	Uint8 (1/7.1.1) = 1
7	2 041	Structure Data	#00 bytes

Figure 4 - Beginning Extended Area Descriptor format

9.2.1 Structure Type (BP 0)

This field shall specify 0.

9.2.2 Standard Identifier (BP 1)

This field shall specify "BEA01".

9.2.3 Structure Version (BP 6)

This field shall specify the version of this descriptor. The value 1 shall indicate the structure of this Part of ISO/IEC 13346.

9.2.4 Structure Data (BP 7)

This field shall be reserved for future standardisation and all bytes shall be set to #00.

9.3 Terminating Extended Area Descriptor

The Terminating Extended Area Descriptor shall be recorded in the format shown in figure 2/5.

BP	Length	Name	Contents
0	1	Structure Type	`Uint8` (1/7.1.1) = 0
1	5	Standard Identifier	bytes = "TEA01"
6	1	Structure Version	`Uint8` (1/7.1.1) = 1
7	2 041	Structure Data	#00 bytes

Figure 5 - Terminating Extended Area Descriptor format

9.3.1 Structure Type (BP 0)

This field shall specify 0.

9.3.2 Standard Identifier (BP 1)

This field shall specify "TEA01".

9.3.3 Structure Version (BP 6)

This field shall specify the version of this descriptor. The value 1 shall indicate the structure of this Part of ISO/IEC 13346.

9.3.4 Structure Data (BP 7)

This field shall be reserved for future standardisation and all bytes shall be set to #00.

9.4 Boot Descriptor

The Boot Descriptor shall be recorded in the format shown in figure 2/6.

BP	Length	Name	Contents
0	1	Structure Type	`Uint8` (1/7.1.1) = 0
1	5	Standard Identifier	bytes = "BOOT2"
6	1	Structure Version	`Uint8` (1/7.1.1) = 1
7	1	Reserved	#00 byte
8	32	Architecture Type	`regid` (1/7.4)
40	32	Boot Identifier	`regid` (1/7.4)
72	4	Boot Extent Location	`Uint32` (1/7.1.5)
76	4	Boot Extent Length	`Uint32` (1/7.1.5)
80	8	Load Address	`Uint64` (1/7.1.7)
88	8	Start Address	`Uint64` (1/7.1.7)
96	12	Descriptor Creation Date and Time	`timestamp` (1/7.3)
108	2	Flags	`Uint16` (1/7.1.3)
110	32	Reserved	#00 bytes
142	1 906	Boot Use	bytes

Figure 6 - Boot Descriptor format

9.4.1 Structure Type (BP 0)

This field shall specify 0.

9.4.2 Standard Identifier (BP 1)

This field shall specify "BOOT2".

9.4.3 Structure Version (BP 6)

This field shall specify the version of this descriptor. The value 1 shall indicate the structure of this Part of ISO/IEC 13346.

9.4.4 Reserved (BP 7)

This field shall be reserved for future standardisation and shall be set to #00.

9.4.5 Architecture Type (BP 8)

This field shall specify an identification of a system which can recognise and act upon the contents of the Boot Identifier field. If this field contains all #00 bytes, no such system is identified.

9.4.6 Boot Identifier (BP 40)

This field shall specify an identification of an implementation which can recognise and act upon the contents of the Boot Extent Location, Boot Extent Length, Load Address, Start Address and Boot Use fields. If this field contains all #00 bytes, no such implementation is identified.

9.4.7 Boot Extent Location (BP 72)

This field shall specify the address of an extent of the volume containing boot information. If the Boot Extent Length field contains 0, then no boot extent is specified and this field shall contain 0.

NOTE 3 - If no boot extent is specified, the information needed to boot might be recorded in the Boot Use field.

9.4.8 Boot Extent Length (BP 76)

This field shall specify the length, in bytes, of the extent identified by the Boot Extent Location field.

9.4.9 Load Address (BP 80)

This field shall specify the memory address at which the information in the extent specified by the Boot Extent field should be copied.

9.4.10 Start Address (BP 88)

This field shall specify the memory address to which control should be transferred after the information specified by the Boot Extent field has been copied into memory.

9.4.11 Descriptor Creation Date and Time (BP 96)

This field shall specify the date and time of the day at which the information in this descriptor was recorded.

9.4.12 Flags (BP 108)

This field shall specify certain characteristics of the Boot Descriptor as shown in figure 2/7.

Bit	Interpretation
0	Erase: For any Boot Descriptor with the same contents of the Architecture Type and Boot Identifier fields as this Boot Descriptor and recorded in any lower numbered sectors of the volume recognition sequence than the sectors that this Boot Descriptor is recorded in: if set to ZERO, shall mean that this Boot Descriptor overrides any such Boot Descriptor; if set to ONE, shall mean that any such Boot Descriptor (including this Boot Descriptor) shall be ignored.
1-15	Shall be reserved for future standardisation and all bits shall be set to ZERO.

Figure 7 - Boot Descriptor characteristics

9.4.13 Reserved (BP 110)

This field shall be reserved for future standardisation and all bytes shall be set to #00.

9.4.14 Boot Use (BP 142)

This field shall be reserved for implementation use, and its contents are not specified by this Part of ISO/IEC 13346.

NOTE 4 - The Boot Descriptor is designed to allow a generic boot program. Such a boot program would scan for Boot Descriptors with a matching Architecture Type (which might represent combinations of processor type and memory management), and after examining the Boot Identifier, which might encode the operating system type and options, present the user with a choice of operating systems to boot. As this Part of ISO/IEC 13346 cannot mandate any specific implementation behaviour, the recommended interpretation of the Boot Descriptor, that is, read an extent of sectors from the volume into memory at a specified location and then transfer execution to another specified location, is optional.