BS ISO/IEC 20919:2021

BSI Standards Publication

Information technology — Linear tape file system (LTFS) Format specification

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 20919:2021

National foreword

This British Standard is the UK implementation of ISO/IEC 20919:2021. It supersedes BS ISO/IEC 20919:2016, which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee ICT/1, Information systems co-ordination.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2021
Published by BSI Standards Limited 2021

ISBN 978 0 539 12916 8

ICS 35.220.20

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 August 2021.

Amendments/corrigenda issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 20919:2021

INTERNATIONAL STANDARD

ISO/IEC 20919

Second edition
2021-07

Information technology — Linear tape file system (LTFS) Format specification

Technologies de l'information — Spécification du format de système de fichier à bande magnétique

Reference number
ISO/IEC 20919:2021(E)

Documents
Document Sharing Hub

© ISO/IEC 2021

Full document details: BS ISO/IEC 20919:2021

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2021

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 20919:2021

Foreword

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

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by the Storage Networking Industry Association (SNIA) (as Linear Tape File System (LTFS) Format Specification, Version 2.5) and drafted in accordance with its editorial rules. It was adopted, under the JTC 1 PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

This second edition cancels and replaces the first edition (ISO/IEC 20919:2016), which has been technically revised.

The main changes compared to the previous edition are as follows:

— Volume Advisory Locking — a method of marking a volume as locked against future modification;

— Percent Encoding — method of including previously reserved characters in names of files and directories;

— Incremental Indexes — a method of recording a smaller index containing only changes since the last index;

— File hashes — a method of recording fils hash information in the index in an interchangeable manner;

— Open For Write — a method of noting files in the index which were still open at the time the index is written to tape.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Documents
Document Sharing Hub

© ISO/IEC 2021 – All rights reserved

Full document details: BS ISO/IEC 20919:2021

USAGE

Copyright © 2020 SNIA. All rights reserved. All other trademarks or registered trademarks are the property of their respective owners.

The SNIA hereby grants permission for individuals to use this document for personal use only, and for corporations and other business entities to use this document for internal use only (including internal copying, distribution, and display) provided that:

1. Any text, diagram, chart, table or definition reproduced shall be reproduced in its entirety with no alteration, and,

2. Any document, printed or electronic, in which material from this document (or any portion hereof) is reproduced, shall acknowledge the SNIA copyright on that material, and shall credit the SNIA for granting permission for its reuse.

Other than as explicitly provided above, you may not make any commercial use of this document or any portion thereof, or distribute this document to third parties. All rights not explicitly granted are expressly reserved to SNIA.

Permission to use this document for purposes other than those enumerated above may be requested by e-mailing tcmd@snia.org. Please include the identity of the requesting individual and/or company and a brief description of the purpose, nature, and scope of the requested use.

All code fragments, scripts, data tables, and sample code in this SNIA document are made available under the following license:

BSD 3-Clause Software License

Copyright (c) 2020, The Storage Networking Industry Association.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of The Storage Networking Industry Association (SNIA) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

DISCLAIMER

The information contained in this publication is subject to change without notice. The SNIA makes no warranty of any kind with regard to this specification, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The SNIA shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this specification.

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub

Full document details: BS ISO/IEC 20919:2021

Revision History

Revision	Date	Sections	Originator	Comments
2.3.0 rev 1	3/18/14	Add section F.3	SNIA LTFS TWG	Edits during TWG F2F
2.3.0 rev 2	12/15/14	Add Changes per TWG	SNIA LTFS TWG	Edits adding Volume Advisory Locking, Volume UUID and LTFS Name Encoding
2.3.0 rev 3	3/18/15	Add changes per TWG	SNIA LTFS TWG	Edits Advisory Locking, Sha512,Media Pool Extended Attributes &MAM
2.3.0 rev 4	3/24/15	Add edits from TWG review	SNIA LTFS TWG	Cleanup some reference errors and minor editorial changes
2.3.0 rev 5	9/15/15	Add example 3 to Appendix F.1.6	SNIA LTFS TWG	Added additional description and example to describe reclamation of spanned file segments
2.3.0 rev 6	9/15/15	Add edits from TWG review	SNIA LTFS TWG	Revised wording in Appendix F.1.6 example 3
2.4.0 rev 0	11/14/15	Add work from TWG	SNIA LTFS TWG	Added changes listed in Change history
2.4.0 rev 1	1/12/17	Add edits from TWG review	SNIA LTFS TWG	Cleaned up some hyperlinks and minor editorial changes
2.4.0 rev 2	1/17/17	Add edits from TWG F2F	SNIA LTFS TWG	Minor editorial changes
2.4.0 rev3	6/12/17	Add edits from TWG	SNIA LTFS TWG	Added Advisory Locking changes
2.4.0 rev 4	6/13/17	Add Media Pool changes per TWG	SNIA LTFS TWG	Added additional extended attributes for Media Pool
2.4.0 rev 5	6/19/17	Add edits per TWG	SNIA LTFS TWG	Cleaned up a couple of typographical errors
2.4.0 rev 6	10/10/17	Add edits per TWG	SNIA LTFS TWG	Incorporated public review editorial comments
2.4.0 rev 7	10/18/17	Add edits per TWG	SNIA LTFS TWG	Minor editorial changes
2.4.0 rev 8	10/24/17	Annex G	SNIA LTFS TWG	Minor editorial change
2.4.0 rev 9	11/14/17	Annex G	SNIA LTFS TWG	Minor editorial change
2.5.0 rev 0	12/04/18	Sections 5 & 9, Annex B, etc.	SNIA LTFS TWG	Add incremental indexes; changes for sync behavior
2.5.0 rev 1	02/11/19	Add Annex H; edits per TWG	SNIA LTFS TWG	New Annex describing incremental indexes
2.5.0 rev 2	04/02/19	Section 9.2; Annex H	SNIA LTFS TWG	Corrected typos in example XML, updated flowchart in Annex H to clarify intent
2.5.1 rev 0	08/18/20	Section 3; Annex I	SNIA LTFS TWG	Editorial changes to References per feedback from ISO

Suggestion for changes or modifications to this document should be sent to the SNIA Linear Tape File System Technical Work Group at http://www.snia.org/feedback/.

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

Changes between v1.0 and v2.0.0

- Incremented version number to 2.0.0 and updated date to March 11, 2011.
- Improvements in specification text to remove ambiguity and clarify intention of the specification. These changes were made at several locations throughout the document.
- Improvements to clarify description of MAM parameters in Section 10 Medium Auxiliary Memory.
- Removed reference to a specific version of the Unicode standard in Section 7.5 Name pattern format. This removes any requirement to use specific versions of Unicode support code in an implementation.
- Improved description of Name pattern format to remove ambiguity in Section 7.5 Name pattern format.
- Added description of LTFS Format specification version numbering in Section 2.1 Versions.
- Updated XML Schema for Label and Index to match version number format in Annex A and Annex B.
- Added specification of minimum and recommended blocksize value for LTFS Volumes to Section 8.1.2 LTFS Label.
- Added definition of allowed version numbers to Section 8.1.2 LTFS Label and Section 9.2 Index.
- Added definition of fileoffset tag in Section 9.2 Index.
- Extended description in Section 6 Data Extents to support addition of fileoffset tag and associated functionality.
- Added definition of highestfileuid tag in Section 9.2 Index.
- Added definition of fileuid tag in Section 9.2 Index.
- Added definition of backuptime tag in Section 9.2 Index.
- Incremented version number in Application Client Specific Information (ACSI) structure shown in Section 10.3 Use of Volume Coherency Information for LTFS. This increment allows identification of LTFS Volumes written with a LTFS v1.0 compliant implementation. A widely used v1.0 implementation wrote ambiguous ACSI values due to an implementation bug.
- Added definition of extended attributes in the ltfs.* namespace in Annex C.
- Added description for handling unknown XML tags in Index to Section 9.2.13 Managing LTFS Indexes.

Changes between v2.0.0 and v2.0.1

- Incremented specification version number to 2.0.1.
- Updated specification date to August 17, 2011.
- Expanded historical record of changes between revisions of LTFS Format Specification.
- Improved description of constraints for two Indexes having the same generation number in Section 5.4.1 Generation Number to make it clear that differences in access time values is permitted between Indexes that are otherwise except for self pointer and index pointer values.
- Added note in Section 5.4.1 Generation Number to explicitly state that Index generation numbers may increase by integer values other than 1.
- Expanded description of the ltfs.sync extended attribute in Annex C . The expanded description explicitly states that this extended attribute triggers a sync of the in-memory data to the storage media. That is, the operation is analogous to a POSIX sync operation.

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

Changes between v2.0.1 and v2.1.0

- Incremented specification version number to 2.1.0.
- Updated specification date to October 18, 2012.
- Added definition of symlink tag in Section 9.2 Index.
- Added example of symlink tag use in Annex E.
- Added symlink tag to Annex B.
- Added description of “ltfs.vendor.X.Y” extended attribute namespace in Annex C .
- Added description of software metadata section in Annex C.
- Added description of drive metadata section in Annex C.
- Added ”ltfs.labelVersion” extended attribute in Annex C.
- Added ”ltfs.indexVersion” extended attribute in Annex C
- Added ”ltfs.mediaEncrypted” extended attribute in Annex C .
- Improved description of ”ltfs.mediaStorageAlert” extended attribute in Annex C.

Changes between v2.1.0 and v2.2.0

- Incremented specification version number to 2.2.0.
- Updated specification date to July 16, 2013.
- Changed “2010” to “2013” in XML examples.
- Editorial Cleanup.
- Changed “extentinfo” definition in Section 9.2 Index.
- Changed “symlink” definition in Section 9.2 Index.
- Added additional paragraph to “symlink” definition in Section 9.2 Index.
- Added general comments at start of Section 10 Medium Auxiliary Memory.
- Added Section 10.4 Use of Host-type Attributes for LTFS.
- Removed Section 9 Certification from document.
- Added “ltfs.mamBarcode” extended attribute in Volume Metadata.
- Added “ltfs.mamApplicationVendor” extended attribute in Volume Metadata.
- Added “ltfs.mamApplicationVersion” extended attribute in Volume Metadata.
- Added “ltfs.mamApplicationFormatVersion” extended attribute in Volume Metadata.
- Added new Annex F Interoperability Recommendation and added File Spanning and File Permissions subsections

Changes between v2.2.0 and 2.3.0 rev 1

- Add section F.3 Storing File Hash Values in LTFS
- Add Section 10.5 Volume Advisory Locking to Section 10 Medium Auxiliary Memory
- Added Volume UUID to Section 10.4 and Section 10.4.8
- Added LTFS Name Encoding to Sections 3.1, 7.4, 9.2.1 and 9.2.2
- Added SHA512 to Section F.3 Storing Hash Values in LTFS
- ISO document formatting changes

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

- Add Media Pool Extended Attributes and MAM

Changes between v2.3.0 rev 1 and v2.4.0

- Incremented specification version number to 2.4.0.
- Added new text to Section 10.5 Volume Advisory Locking
- Added new text to Annex C.4 VolumeMetadata, ltfs.volumeLockState
- Added new text for "openforwrite" to Section 9.2.9
- Added new entry to the file element list in Annex B
- Added new descriptive text to Annex E
- Added new example file "partialfile.bin" to Annex E
- Updated the compliance statement to "version 2.4.0" in Section 7.4 Name Format
- Added new percentencoding text to Section 7.4 Name Format
- Removed percentencoding definition from Section 9.2.7
- Removed percentencoding definition from Section 9.2.9
- Added new text to symlink definition in Section 9.2.9
- Removed percentencoding paragraph(s) from Section 9.2.10
- Removed the percentencoded entry from the directory, file and extended attributes sections in Annex B
- Changed the name definition in the directory section of Annex B
- Changed the name definition in the file section of Annex B
- Changed the symlink definition in the file section of Annex B
- Changed the key definition in the extendedattributes section of Annex B
- Added new NameType definition to Annex B
- Changed the entry for fileuid "10" in Annex E

Changes between v2.4.0 and v2.4.0 rev 1

- Incremented specification version number to 2.4.0 rev 1.
- Changed version from 2.3.0 to 2.4.0 in Section 2.1 Versions
- Changed version from 2.3.0 to 2.4.0 in Annex E
- Cleaned up some hyperlinks and minor editorial changes

Changes between v2.4.0 rev 1 and v2.4.0 rev 2

- Changed version from 2.3.0 to 2.4.0 throughout document except for Section 9.2.19
- Changed updatetime in Annex E to reflect correct date/time value for example shown
- Changed highestfileuid in Annex E to reflect correct value for example shown
- Changed lockType to locktype throughout document
- Changed NameType to nametype throughout document

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

Changes between v2.4.0 rev 2 and v2.4.0 rev 3

- Updated the definition of the volumeLockState virtual extended attribute in Section 10.5 and Annex C.4

Changes between v2.4.0 rev 3 and v2.4.0 rev 4

- Updated the definition of the Media Pool MAM attribute in Section 10.4 and Annex F.4.1.1

Changes between v2.4.0 rev 4 and v2.4.0 rev 5

- Fixed a couple of typographical errors

Changes between v2.4.0 rev 5 and v2.4.0 rev 6

- Corrections to Annex B – XML Schema
- Added Annex G – character representations
- Clarified in Section 7.4 that percent encoding does not apply to Table 13, and added reference to Annex G
- Added two further IBM contributors to the Acknowledgements

Changes between v2.4.0 rev 6 and v2.4.0 rev 7

- Changed title and reference of table G.1 in Annex G – applies to 2.3 and later
- Added clarifying note for symlinks in Annex G

Changes between v2.4.0 rev 7 and v2.4.0 rev 8

- Changed column heading in tables G.1 and G.2 to “symlink target name”
- Improved wording of Note 3 in Annex G

Changes between v2.4.0 rev 8 and v2.4.0 rev 9

- Changed column heading in tables G.1 and G.2 to “File name, directory name, filename pattern”
- Added corresponding Note 4 in Annex G

Changes between v2.4.0 rev 9 and v2.5.0

- Incremented specification version number to 2.5.0.
- Added new text to describe Full and Incremental Indexes, in particular in 5.4 Index Layout and 9 Index Format.
- Added new virtual extended attributes related to Incremental Indexes in C.4.
- Specify required behavior in response to writing ltfs.commitMessage and ltfs.sync VEAs in Annex C.

Changes between v2.5.0 rev 0 and v2.5.0 rev 1

- Added new informative Annex H describing background information on Incremental Indexes
- Editorial changes to clarify required vs optional elements in Incremental Indexes in 9 Index Format.
- Clarified that the required behavior for ltfs.commitMessage and ltfs.sync applies only to implementations supporting those VEAs.

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

Changes between v2.5.0 rev 1 and v2.5.0 rev 2

- Corrected typos in Incremental Indexes example XML in 9.2.2 and 9.2.12
- Corrected typos in Annex H and amended flowchart H.1 to clarify the intent

Changes between v2.5.0 rev 2 and v2.5.1 rev 0

- Moved informative references from Section 3 to new Annex I Bibliography
- Amended all references to SPC-3 / SPC-4 to SPC-5

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

Acknowledgements

The SNIA LTFS Technical Working Group, which developed and reviewed this specification, would like to recognize the significant contributions made by the following members:

EMC Corporation.....Don Deel
Hewlett Packard Enterprise .....Chris Martin
IBM.....David Pease
.....Ed Childers
.....Takeshi Ishimoto
.....Atsushi Abe
NetApp.....David Slik
Oracle Corporation.....Matthew Gaffney
.....Carl Madison
Quantum Corporation.....Paul Stone
..... Jim Wong
SNIA.....Arnold Jones

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub

Full document details: BS ISO/IEC 20919:2021

Contents

1 Introduction 15

2 Scope 16

2.1 Versions 16

2.2 Conformance 17

3 Normative references 18

4 Definitions and Acronyms 19

4.1 Definitions 19

4.2 Acronyms 21

5 Volume Layout 22

5.1 LTFS Partitions 22

5.2 LTFS Constructs 22

5.3 Partition Layout 23

5.4 Index Layout 24

6 Data Extents 27

6.1 Extent Lists 27

6.2 Extents Illustrated 27

6.3 Files Illustrated 29

7 Data Formats 32

7.1 Boolean format 32

7.2 Creator format 32

7.3 Extended attribute value format 32

7.4 Name format 32

7.5 Name pattern format 34

7.6 String format 34

7.7 Time stamp format 34

7.8 UUID format 35

8 Label Format 36

8.1 Label Construct 36

9 Index Format 39

9.1 Index Construct 39

9.2 Index 39

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub

Full document details: BS ISO/IEC 20919:2021

10 Medium Auxiliary Memory 53
10.1 Volume Change Reference 53
10.2 Volume Coherency Information 54
10.3 Use of Volume Coherency Information for LTFS 5433
10.4 Use of Host-type Attributes for LTFS 55
10.5 Volume Advisory Locking 57

Annex A (normative) LTFS Label XML Schema 59

Annex B (normative) LTFS Index XML Schemas 61
B.1 LTFS Full Index XML Schema 61
B.2 LTFS Incremental Index XML Schema 63

Annex C (normative) Reserved Extended Attribute definitions 66
C.1 Software Metadata 66
C.2 Drive Metadata 66
C.3 Object Metadata 66
C.4 Volume Metadata 67
C.5 Media Metadata 69

Annex D (informative) Example of Valid Simple Complete LTFS Volume 72

Annex E (informative) Complete Example LTFS Full Index 73

Annex F (normative) Interoperability Recommendations 78
F.1 Spanning Files across Multiple Tape Volumes in LTFS 78
F.2 File Permissions in LTFS 83
F.3 Storing File Hash Values in LTFS 86
F.4 LTFS Media Pools 87

Annex G (informative) Character representations 89

Annex H (informative) Incremental Indexes 92
H.1 Background 92
H.2 Backwards Compatibility 92
H.3 Traversing the Index Back Pointer Chain 93
H.4 Incremental Index Format 93
H.5 Processing Incremental Indexes 95
H.6 Miscellaneous 96

Annex I (informative) Bibliography 98

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

List of Figures

Figure 1 — LTFS Partition 22
Figure 2 — Label Construct 22
Figure 3 — Index Construct 23
Figure 4 — Partition Layout 23
Figure 5 — Complete partition containing data 24
Figure 6 — Back Pointer example 25
Figure 7 — Back Pointer example for Incremental Indexes 26
Figure 8 — Extent starting and ending with full block 28
Figure 9 — Extent starting with full block and ending with fractional block 28
Figure 10 — Extent starting and ending in mid-block 28
Figure 11 — File contained in a single Data Extent 29
Figure 12 — File contained in two Data Extents 29
Figure 13 — Shared Blocks example 30
Figure 14 — Sparse files example 30
Figure 15 — Shared data example 31
Figure 16 — Label construct 36
Figure 17 — Index Construct 39
Figure D.1 — Content of a simple LTFS volume 72
Figure H.1 — Processing an Incremental Index (flowchart) 97

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

List of Tables

Table 1 — Version elements 16
Table 2 — Version comparisons 16
Table 3 — Extent list entry starting and ending with full block 28
Table 4 — Extent list entry starting with full block and ending with fractional block 28
Table 5 — Extent list entry starting and ending in mid-block 28
Table 6 — Extent list entry for file contained in a single Data Extent 29
Table 7 — Extent list entry for a file contained in two Data Extents 29
Table 8 — Extent lists for Shared Blocks example 30
Table 9 — Extent list for sparse files example 31
Table 10 — Extent lists for shared data example 31
Table 11 — Creator format definitions 32
Table 12 — Reserved characters for name format 33
Table 13 — Characters which should be avoided for name format 33
Table 14 — Name percent-encoding 33
Table 15 — Time stamp format 35
Table 16 — VOL1 Label Construct 36
Table 17 — Volume Coherency Information 54
Table 18 — ACSI format for LTFS 54
Table 19 — Relevant Host-type Attributes for LTFS 55
Table 20 — Example of Host-type Attributes 57
Table 21 — Volume Locked MAM Attribute 58
Table 22 — Volume Locked MAM Attribute Values 58
Table F.1 — Hash Types 86
Table G.1 — Character representations : version 2.3 or later 89
Table G.2 — Character representations : version 2.2 or earlier 90

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

1 Introduction

This document defines a Linear Tape File System (LTFS) Format separate from any implementation on data storage media. Using this format, data is stored in LTFS Volumes. An LTFS Volume holds data files and corresponding metadata to completely describe the directory and file structures stored on the volume.

The LTFS Format has these features:

- An LTFS Volume can be mounted and volume content accessed with full use of the data without the need to access other information sources.
- Data can be passed between sites and applications using only the information written to an LTFS Volume.
- Files can be written to, and read from, an LTFS Volume using standard POSIX file operations.

The LTFS Format is particularly suited to these usages:

- Data export and import.
- Data interchange and exchange.
- Direct file and partial file recall from sequential access media.
- Archival storage of files using a simplified, self-contained or “self-describing” format on sequential access media.

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

2 Scope

This document defines the LTFS Format requirements for interchanged media that claims LTFS compliance. Those requirements are specified as the size and sequence of data blocks and file marks on the media, the content and form of special data constructs (the LTFS Label and LTFS Index), and the content of the partition labels and use of MAM parameters.

The data content (not the physical media) of the LTFS format shall be interchangeable among all data storage systems claiming conformance to this format. Physical media interchange is dependent on compatibility of physical media and the media access devices in use.

NOTE: This document does not contain instructions or tape command sequences to build the LTFS structure.

2.1 Versions

This document describes version 2.5.0 of the Linear Tape File System (LTFS) Format Specification.

The version number for the LTFS Format Specification consists of three integer elements separated by period characters of the form *M.N.R*, where *M*, *N* and *R* are positive integers or zero. Differences in the version number between different revisions of this specification indicate the nature of the changes made between the two revisions. Each of the integers in the format specification are incremented according to Table 1.

Table 1 — Version elements

Element	Description
M	Incremented when a major update has been made to the LTFS Format Specification. Major updates are defined as any change to the on-media format or specification semantics that are expected to break compatibility with older versions of the specification.
N	Incremented when a minor update has been made to the LTFS Format Specification. Minor updates are defined as any change to the on-media format or specification semantics that is not expected to break compatibility with older versions of the specification that have the same value for *M* in the version number.
R	Incremented when textual revisions are made to the LTFS Format Specification. Textual revisions are defined as revisions that improve the clarity of the specification document *without* changing the intent of the document. By definition, minor changes do not alter the on-media format or specification semantics.

NOTE 1: When any element of the specification version number is incremented, all sub-ordinate elements to the right are reset to zero. For example, if the version is 1.0.12 and N is incremented to 1, then R is set to zero resulting in version 1.1.0.

NOTE 2: The first public version of this document used version number 1.0. This value should be interpreted as equivalent to 1.0.0 in the version numbering defined in this document.

The result of comparison between two LTFS version numbers $M_A.N_A.R_A$ and $M_B.N_B.R_B$ is defined in Table 2.

Table 2 — Version comparisons

Conditional	Description
$M_A < M_B$	$M_A.N_A.R_A$ is an earlier version than $M_B.N_B.R_B$.
$M_A = M_B$ *and* $M_A < N_B$	$M_A.N_A.R_A$ is an earlier version than $M_B.N_B.R_B$.
$M_A = M_B$ *and* $N_A = N_B$ *and* $R_A < R_B$	$M_A.N_A.R_A$ is an earlier version than $M_B.N_B.R_B$. However, as defined above, changes that result only in a different *R* value are descriptive changes in the specification rather than on media changes.

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

2.2 Conformance

Recorded media claiming conformance to this format shall be in a consistent state when interchanged or stored. See Section 4.1.4.

Any implementation conforming to this specification should be able to correctly read Label and Index structures from all prior versions of this specification and write Label and Index structures conforming to the descriptions in this document. The current Label and Index structures are defined in Section 8 Label Format and in Section 9 Index Format.

NOTE: Where practical, any implementation supporting a given version value for M should endeavor to support LTFS volumes with version numbers containing higher values for N and R than those defined at the time of implementation.

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

3 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 14776-455, Information Technology - Small Computer System Interface (SCSI) – Part 455: SCSI Primary Commands - 5 (SPC-5) [ANSI INCITS.502-219]

SSC-4 SCSI Stream Commands – 4 (SSC-4) [ANSI INCITS 516-2013]

ISO 8601:2004 Data elements and interchange formats – Information interchange – Representation of dates and times – (UTC)

ISO/IEC 10646:2012: Information technology - Universal Coded Character Set (UCS) (UTF-8)

IETF RFC 4648, The Base16, Base32, and Base64 Data Encodings, http://www.ietf.org/rfc/rfc4648.txt

IETF RFC 4122, Universally Unique Identifier (UUID) URN Namespace http://www.ietf.org/rfc/rfc4122.txt

IETF RFC 3986, Uniform Resource Identifier (URI): Generic Syntax, http://www.ietf.org/rfc/rfc3986.txt

ANSI X3.27-1978 American National Standard Magnetic Tape Labels and File Structure for Information

W3C - Extensible Markup Language (XML) http://www.w3.org/XML

OSF CDE 1.1, Remote Procedure Call – Universal Unique Identifier (UUID) http://pubs.opengroup.org/onlinepubs/9629399/toc.pdf

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

4 Definitions and Acronyms

For the purposes of this document the following definitions and acronyms shall apply.

4.1 Definitions

4.1.1

Block Position

The position or location of a recorded block as specified by its LTFS Partition ID and logical block number within that partition.

The block position of an Index is the position of the first logical block for the Index.

4.1.2

Complete Partition

An LTFS partition that consists of an LTFS Label Construct and a Content Area, where the last construct in the Content Area is an Index Construct.

4.1.3

Content Area

A contiguous area in a partition, used to record Index Constructs and Data Extents.

4.1.4

Consistent State

A volume is consistent when both partitions are complete and the last Index Construct in the Index Partition has a back pointer to the last Full Index Construct in the Data Partition.

4.1.5

Data Extent

A contiguous sequence of recorded blocks.

4.1.6

Data Partition

An LTFS partition primarily used for data files.

4.1.7

File

A group of logically related extents together with associated file metadata.

4.1.8

Filesystem sync

An operation during which all cached file data and metadata is flushed to the media.

4.1.9

Full Index

A data structure that describes all valid data files in an LTFS volume. The Full Index is an XML document conforming to the XML schema shown in Annex B (normative) LTFS Index XML Schema.

4.1.10

Generation number

A positive decimal integer which shall indicate the specific generation of an Index within an LTFS volume.

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

4.1.11

Incremental Index

A data structure that describes changes made to the LTFS volume since the last index was written. The Incremental Index is an XML document conforming to the XML schema shown in Annex B (normative) LTFS Index XML Schema.

4.1.12

Index

Either a Full Index or an Incremental Index.

4.1.13

Index Construct

A data construct comprised of an Index and file marks.

4.1.14

Index Partition

An LTFS partition primarily used to store Index Constructs and optionally data files.

4.1.15

Label Construct

A data construct comprised of an ANSI VOL1 tape label, LTFS Label, and tape file marks.

4.1.16

Linear Tape File System (LTFS)

This document describes the Linear Tape File System Format.

4.1.17

LTFS Construct

Any of three defined constructs that are used in an LTFS partition. The LTFS constructs are: Label Construct, Index Construct, and Data Extent.

4.1.18

LTFS Label

A data structure that contains information about the LTFS partition on which the structure is stored. The LTFS Label is an XML document conforming to the XML schema shown in Annex A (normative) LTFS Label XML Schema.

4.1.19

LTFS Partition

A tape partition that is part of an LTFS volume. The partition contains an LTFS Label Construct and a Content Area.

4.1.20

LTFS Volume

A pair of LTFS partitions, one Data Partition and one Index Partition, that contain a logical set of files and directories. The pair of partitions in an LTFS Volume shall have the same UUID. All LTFS partitions in an LTFS volume are *related partitions*.

4.1.21

Medium Auxiliary Memory

An area of non-volatile storage that is part of an individual storage medium. The method of access to this non-volatile storage is standardized as described in the SPC-5 standard.

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

4.1.22

Partition Identifier (Partition ID)

The logical partition letter to which LTFS data files and Indexes are assigned.

The linkage between LTFS partition letter and physical SCSI partition number is determined by the SCSI partition in which the LTFS Label is recorded. The LTFS partition letter is recorded in the LTFS Label construct, and the SCSI partition number is known by the SCSI positional context where they were read/written.

4.1.23

Sparse file

A file that has some number of empty (unwritten) data regions. These regions are not stored on the storage media and are implicitly filled with bytes containing the value zero (0x00).

4.1.24

UUID

Universally unique identifier; an identifier use to bind a set of LTFS partitions into an LTFS volume.

4.1.25

Volume Change Reference (VCR)

A value that represents the state of all partitions on a medium.

4.1.26

Volume Advisory Locking

An indication that the LTFS volume has been locked against future modifications. This is a form of write protection under the control of host software rather than physical hardware.

4.2 Acronyms

ASCII American Standard Code for Information Interchange

CM Cartridge Memory

DCE Distributed Computing Environment

ISO International Organization for Standardization

LTFS Linear Tape File System

MAM Media Auxiliary Memory

NFC Normalization Form Canonical Composition

OSF Open Software Foundation

POSIX Portable Operating System Interface for Unix

T10/SSC-4 ISO/IEC 14776-334, SCSI Stream Commands - 4 (SSC-4) [T10/2123-D]

UTC Coordinated Universal Time

UTF-8 8-bit UCS/Unicode Transformation Format

UUID Universally Unique Identifier **W3C** World Wide Web Consortium **XML** Extensible Markup Language

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

5 Volume Layout

An LTFS volume is comprised of a pair of LTFS partitions. LTFS defines two partition types: data partition and index partition. An LTFS volume shall contain exactly one Data Partition and exactly one Index Partition.

5.1 LTFS Partitions

Each partition in an LTFS volume shall consist of a Label Construct followed by a Content Area. This logical structure is shown in Figure 1.

Figure 1 — LTFS Partition

The Label Construct is described in Section 5.2 LTFS Constructs and in Section 8 Label Format. The Content Area contains some number of interleaved Index Constructs and Data Extents. These constructs are described in Section 5.2 LTFS Constructs and in Section 9 Index Format. The precise layout of the partitions is defined in Section 5.3 Partition Layout.

5.2 LTFS Constructs

LTFS constructs are comprised of file marks and records. These are also known as ‘logical objects’ as found in T10 SSC specifications and are not described here. An LTFS volume contains three kinds of constructs.

- A Label Construct contains identifying information for the LTFS volume.
- A Data Extent contains file data written as sequential logical blocks. A file consists of zero or more Data Extents plus associated metadata stored in the Index Construct.
- An Index Construct contains an Index, which is an XML data structure which describes the mapping between files and Data Extents.

5.2.1 Label Construct

Each partition in an LTFS volume shall contain a Label Construct with the following structure. As shown in Figure 2, the construct shall consist of an ANSI VOL1 label, followed by a single file mark, followed by one record in LTFS Label format, followed by a single file mark. Each Label construct for an LTFS volume shall contain identical information except for the “location” field of the LTFS Label.

The content of the ANSI VOL1 label and the LTFS Label is specified in Section 8 Label Format.

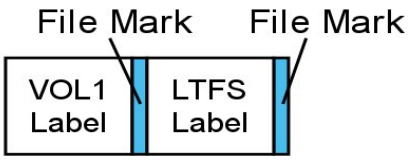

Figure 2 — Label Construct

5.2.2 Data Extent

A Data Extent is a set of one or more sequential logical blocks used to store file data. The “blocksize” field of the LTFS Label defines the block size used in Data Extents. All blocks within a Data Extent

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

shall have this fixed block size except the last block, which may be smaller.

The use of Data Extents to store file data is specified in Section 6 Data Extents.

5.2.3 Index Construct

Figure 3 shows the structure of an Index Construct. An Index Construct consists of a file mark, followed by an Index, followed by a file mark. An Index consists of a record that follows the same rules as a Data Extent, but it does not contain file data. That is, the Index is written as a sequence of one or more logical blocks of size “blocksize” using the value stored in the LTFS Label. Each block in this sequence shall have this fixed block size except the last block, which may be smaller. This sequence of blocks records the Index XML data that holds the file metadata and the mapping from files to Data Extents. The Index XML data recorded in an Index Construct shall be written from the start of each logical block used. That is, Index XML data may not be recorded offset from the start of the logical block.

Figure 3 — Index Construct

Indexes also include references to other Indexes in the volume. References to other Indexes are used to maintain consistency between partitions in a volume. These references (back pointers and self pointers) are described in Section 5.4 Index Layout.

The content of the Index is described in Section 9 Index Format.

5.3 Partition Layout

This section describes the layout of an LTFS Partition in detail. An LTFS Partition contains a Label Construct followed by a Content Area. The Content Area contains zero or more Data Extents and Index Constructs in any order. The last construct in the Content Area of a complete partition shall be an Index Construct.

Figure 4 illustrates an empty complete partition. It contains a Label Construct followed by an Index Construct. This is the simplest possible complete partition.

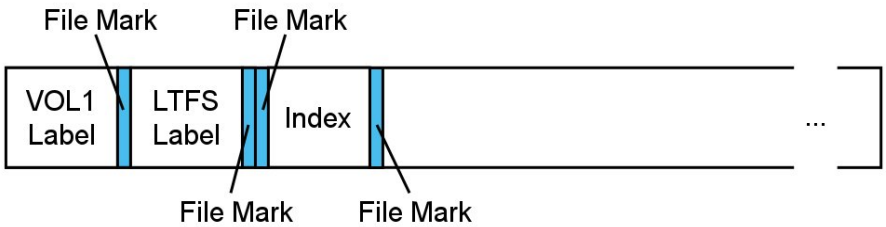

Figure 4 — Partition Layout

Figure 5 illustrates a complete partition containing data. The Content Area on the illustrated partition contains two Data Extents (the first extent comprising the block ‘A’, the second extent comprising blocks ‘B’ and ‘C’) and three Index Constructs.

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

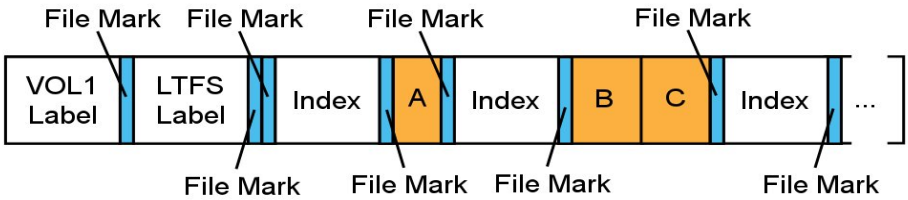

Figure 5 — Complete partition containing data

NOTE: There must not be any additional data trailing the end of the VOL1 Label, the LTFS Label, nor any Index on an LTFS Volume. The Label Construct must be recorded starting at the first logical block in each partition.

5.4 Index Layout

Each Index data structure contains information used to verify the consistency of an LTFS volume.

- A generation number, which records the age of this Index relative to other Indexes in the volume.
- A self pointer, which records the volume to which the Index belongs and the block position of the Index within that volume.
- A back pointer, which records the block position of the last Full Index present on the Data Partition immediately before this Index was written.
- An optional second back pointer, which will only be present if an Incremental Index has been written since the last Full Index on the Data Partition. If present this pointer records the block position of the most recent Incremental Index.

5.4.1 Generation Number

Each Index in a volume has a generation number, a non-negative integer that increases as changes are made to the volume. In any consistent LTFS volume, the Index with the highest generation number on the volume represents the current state of the entire volume and must be a Full Index. Generation numbers are assigned in the following way:

- Given two Indexes on a partition, the one with a higher block position shall have a generation number greater than or equal to that of the one with a lower block position.
- Two Indexes in an LTFS volume may have the same generation number if and only if their contents are identical except for these elements:
  - access time values for files and directories (described in Section 9.2 Index),
  - the self pointer (described in Section 5.4.2 Self Pointer), and
  - the back pointer (described in Section 5.4.3 Back Pointer).

NOTE: The value of the generation number between any two successive Indexes may increase by any positive integer value. That is, the magnitude of increase between any two successive Indexes is not assumed to be equal to 1.

The first Index on an LTFS Volume shall be generation number ‘1’.

5.4.2 Self Pointer

The self pointer for an Index is comprised of the following information:

- The UUID of the volume to which the Index belongs
- The block position of the Index

The self pointer is used to distinguish between Indexes and Data Extents. An otherwise valid Index with an invalid self pointer shall be considered a Data Extent for the purpose of verifying that a volume is valid and consistent. This minimizes the likelihood of accidental confusion between a valid Index and a Data Extent containing Index-like data.

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

5.4.3 Back Pointer

Each Index contains at most two back pointers, defined as follows.

- If the Index resides in the Data Partition, the full index back pointer shall contain the block position of the preceding Full Index in the Data Partition. If no preceding Index exists, no back pointer shall be stored in this Index. Back pointers are stored in the Index as described in Section 9.2 Index.
- If the Index resides in the Index Partition and has generation number N then the full index back pointer for the Index shall contain either the block position of a Full Index having generation number N in the Data Partition, or the block position of the last Full Index having at most generation number N−1 in the Data Partition. If no Index of generation number N-1 or less exists in the Data Partition, then the Index in the Index Partition is not required to store a back pointer.
- On a consistent volume, the final Index in the Index Partition shall contain a back pointer to the final Full Index in the Data Partition.
- On a volume containing Incremental Indexes, an index residing in the Data Partition may contain a second back pointer with the block position of the most recent Incremental Index on the Data Partition written since the last Full Index
- As a consequence of the rules above, no Index may contain a back pointer to itself or to an Index with a higher generation number.

On a consistent volume, the rules above require that the Indexes on the Data Partition and the final Index on the Index Partition shall form an unbroken chain of back pointers. Figure 6 illustrates this state for a volume not containing any Incremental Indexes, and Figure 7 illustrates the corresponding state for a volume which does contain Incremental Indexes. See Section 9 for more detail on full and incremental indexes.

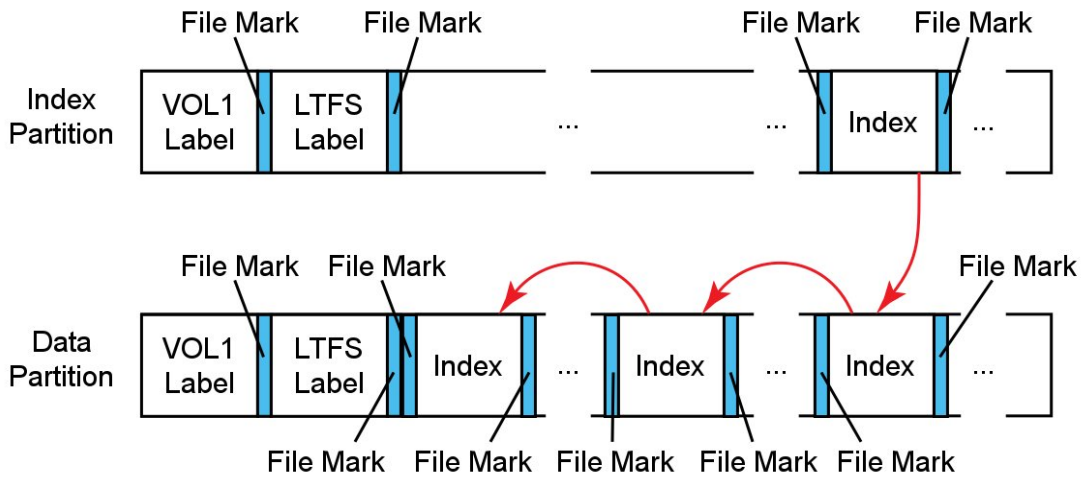

Figure 6 — Back Pointer example

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

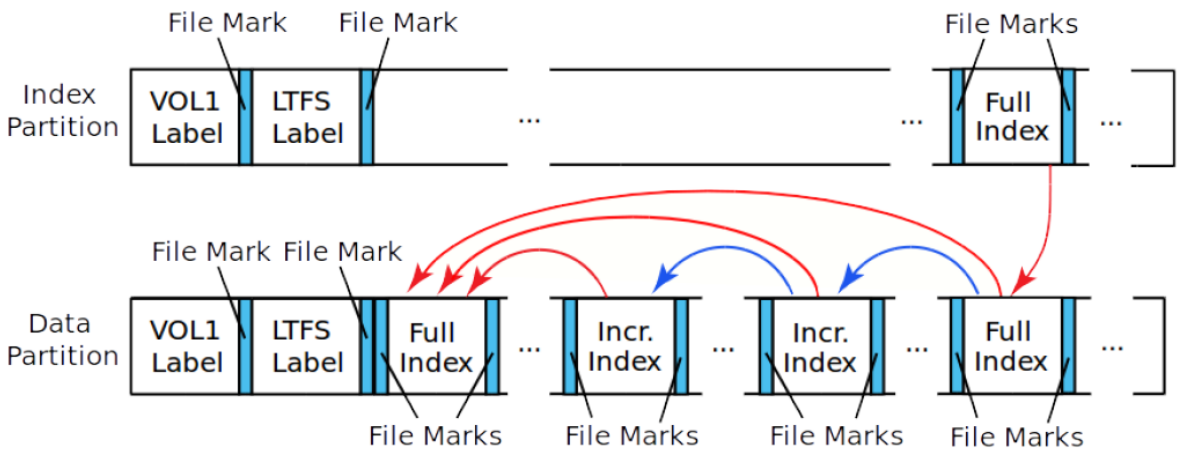

Figure 7 — Back Pointer example for Incremental Indexes

In **Figure 7** the red arrows represent back pointers to a Full Index, and the blue arrows represent back pointers to an Incremental Index.

© ISO 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

6 Data Extents

A Data Extent is a set of one or more sequential records subject to the conditions listed in Section 5.2.2 Data Extent. This section describes how files are arranged into Data Extents for storage on an LTFS volume. Logically, a file contains a sequence of bytes; the mapping from file byte offsets to block positions is maintained in an Index. This mapping is called the extent list.

6.1 Extent Lists

A file with zero size has no extent list.

Each entry in the extent list for a file encodes a range of bytes in the file as a range of contiguous bytes in a Data Extent. An entry in the extent list is known as an extent. Each entry shall contain the following information:

- **partition ID** – partition that contains the Data Extent comprising this extent.
- **start block** (start block number) – block number within the Data Extent where the content for this extent begins.
- **byte offset** (offset to first valid byte) – number of bytes from the beginning of the start block to the beginning of file data for this extent. This value shall be strictly less than the size of the start block. The use of byte offset is described in Section 6.2.3 Starting and ending Data Extent in mid-block.
- **byte count** – number of bytes of file content in this Data Extent.
- **file offset** – number of bytes from the beginning of the file to the beginning of the file data recorded in this extent.

NOTE: Version 1.0 of this specification did not explicitly include file offsets in the extent list. When interpreting LTFS Volumes written based on the Version 1.0 specification, the file offsets shall be determined as follows.

- The first extent list entry begins at file offset 0.
- If an extent list entry begins at file offset N and contains K bytes, the following extent list entry begins at file offset N + K.

These file extent rules for version 1.0 of the specification necessarily imply that the order of extents recorded in the Index shall be preserved during any subsequent update of the Index to another version 1.0 Index.

The inclusion of the File Offset value for each extent starting from version 2.0.0 of this specification removes the significance of the order in which extents are recorded in the Index.

Implementers are encouraged to record extents in the same logical order as they exist in the represented file.

In the extent list for any file, no extent may contain bytes that extend beyond the logical end of file. The logical end of file is defined by the file length recorded in the Index. Also, in any extent list for any file, there shall not exist any pair of extents that contain overlapping logical file offsets. That is, no extent is allowed to logically overwrite any data stored in another extent.

An extent list entry shall be a byte range within a single Data Extent; that is, it shall not cross a boundary between two Data Extents. This requirement allows a deterministic mapping from any file offset to the block position where the data can be found. On the other hand, two extent list entries (in the same file or in different files) may refer to the same Data Extent.

6.2 Extents Illustrated

This section illustrates various forms of extent list entries and the mapping from files to these extents. The illustrations are not exhaustive. Other combinations of starting and ending blocks are possible.

The LTFS Partition ID is an essential element of an extent definition. For simplicity, the LTFS Partition ID and File Offset are not shown explicitly in the extents lists illustrated in Table 3 — Extent list entry starting and ending with full block, Table 4, and Table 5 — Extent list entry starting and ending in mid-block. Note that not all extents in an extent list shall be on the same partition.

6.2.1 Starting and ending Data Extent with full block

Figure 8 illustrates an extent of 3 full size blocks contained within a Data Extent of 3 blocks, *N* through

© ISO 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021

N + 2.

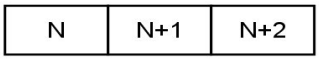

Figure 8 — Extent starting and ending with full block

The extent list entry for this extent is shown in Table 3.

Table 3 — Extent list entry starting and ending with full block

Start Block	Offset	Length
N	0	3 × *Blk*

NOTE: Blk is the length of a full-sized block.

6.2.2 Starting Data Extent with full block and ending with fractional block

Figure 9 illustrates an extent of 2 full-size blocks and one fractional block of *K* bytes, contained within a Data Extent of 2 full size blocks *N* and *N* + 1 and one fractional block *N* + 2.

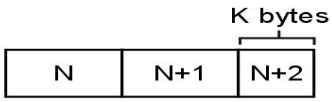

Figure 9 — Extent starting with full block and ending with fractional block

The extent list entry for this extent is shown in Table 4.

Table 4 — Extent list entry starting with full block and ending with fractional block

Start Block	Offset	Length
N	0	(2 × *Blk*) + *K*

NOTE: **K** is the length of the fractional block, where **K < Blk**

6.2.3 Starting and ending Data Extent in mid-block

Figure 10 illustrates an extent smaller than 3 blocks, contained within a Data Extent of 3 full size blocks. Valid data begins in block *N* at byte number *J* and continues to byte number *K* of block *N* + 2. The last block of the extent, block *N* + 2, may be a fractional block.

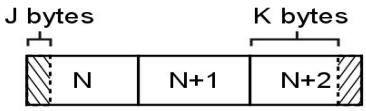

Figure 10 — Extent starting and ending in mid-block

The extent list entry for this extent is shown in Table 5.

Table 5 — Extent list entry starting and ending in mid-block

Start Block	Byte Offset	Byte Count
N	*J*	(*Blk* − *J*) + *Blk* + *K*

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 20919:2021