

SMPTE STANDARD

D-Cinema Packaging —
Asset Mapping and
File Segmentation



Table of Contents	Page
Foreword	2
Intellectual Property	2
1 Scope	3
2 Conformance Notation	3
3 Normative References	3
4 Overview	4
4.1 Ingesting a Mapped File Set (Informative)	5
4.2 XML File Structure	5
5 AssetMap Structure.....	6
5.1 Id	6
5.2 AnnotationText [optional]	7
5.3 Creator	7
5.4 VolumeCount	7
5.5 IssueDate	7
5.6 Issuer	7
5.7 AssetList.....	7
6 Asset Structure.....	7
6.1 Id	8
6.2 AnnotationText [optional]	8
6.3 PackingList [optional]	8
6.4 ChunkList	8
7 Chunk Structure	8
7.1 Path.....	9
7.2 VolumeIndex [optional]	9
7.3 Offset [optional]	9
7.4 Length [optional].....	9
8 VolumeIndex Structure.....	9

9 Media-Specific Constraints.....10

9.1 Mapped File Set Definition 10

9.2 Path URI 10

9.3 XML Encoding 10

9.4 Asset Map Location 10

10 Asset Map Sample (Informative) 11

11 XML Schema 12

Annex A Basic Map Profile v2 (Normative) 14

A.1 Mapped File Set Definition 14

A.2 Path 14

A.3 XML Encoding 14

A.4 Asset Map Location.....14

A.5 Example Ingest Algorithm (Informative).....15

Annex B XML Diagram Legend (Informative).....16

B.1 Element Symbols 16

B.2 Model Symbols ("Compositors") 17

B.3 Types.....17

B.4 Model Groups and References 18

Foreword

SMPTE (the Society of Motion Picture and Television Engineers) is an internationally-recognized standards developing organization. Headquartered and incorporated in the United States of America, SMPTE has members in over 80 countries on six continents. SMPTE’s Engineering Documents, including Standards, Recommended Practices and Engineering Guidelines, are prepared by SMPTE’s Technology Committees. Participation in these Committees is open to all with a bona fide interest in their work. SMPTE cooperates closely with other standards-developing organizations, including ISO, IEC and ITU.

SMPTE Engineering Documents are drafted in accordance with the rules given in its Standards Operations Manual.

SMPTE ST 429-9 was prepared by Technology Committee 21DC.

Intellectual Property

At the time of publication no notice had been received by SMPTE claiming patent rights essential to the implementation of this Standard. However, attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. SMPTE shall not be held responsible for identifying any or all such patent rights.

1 Scope

This document specifies a generic method of mapping asset identifiers (UUID) to locations (URI). An application of the method to common hierarchical filesystems, the Basic Map Profile v2, is specified.

Note: Earlier versions of this specification allowed assets to be split among multiple volumes. This functionality was not found to be useful in practice and is removed from this version. The schema remains however unchanged to minimize impact on implementations.

2 Conformance Notation

Normative text is text that describes elements of the design that are indispensable or contains the conformance language keywords: "shall", "should", or "may". Informative text is text that is potentially helpful to the user, but not indispensable, and can be removed, changed, or added editorially without affecting interoperability. Informative text does not contain any conformance keywords.

All text in this document is, by default, normative, except: the Introduction, any section explicitly labeled as "Informative" or individual paragraphs that start with "Note:"

The keywords "shall" and "shall not" indicate requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.

The keywords, "should" and "should not" indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document. The keyword "reserved" indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword "forbidden" indicates "reserved" and in addition indicates that the provision will never be defined in the future.

3 Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent edition of the standards indicated below.

SMPTE ST 429-8:2007, D-Cinema Packaging — Packing List

Internet Engineering Task Force (IETF) (2005, January). RFC 3986 — Uniform Resource Identifiers (URI): Generic Syntax

Internet Engineering Task Force (IETF) (2005, July). RFC 4122 — A Universally Unique Identifier (UUID) URN Namespace.

World Wide Web Consortium (W3C) (2004, February 4). Extensible Markup Language (XML) 1.0 (Third Edition).

World Wide Web Consortium (W3C) (2004, October 28). XML Schema Part 1: Structures (Second Edition).

World Wide Web Consortium (W3C) (2004, October 28). XML Schema Part 2: Datatypes (Second Edition).

World Wide Web Consortium (W3C) (2009, December 8). Namespaces in XML,

4 Overview

The packing list [SMPTE ST 429-8] specifies the contents of a distribution package.

The package and its Packing List are specified independently of the underlying media, e.g. hard disk partition. The Packing List contains the UUID identifiers of the assets in the Package, not the location of those assets on the underlying media. The Asset Map provides the link between the asset identifier UUIDs and the files containing the assets. This layered approach has two distinct advantages. First, it simplifies specification of the Packing List by removing dependence on past, current or future media formats and constraints. Second, it allows the package and its underlying assets to be created once and mapped onto various target media as circumstances require.

The Asset Map structure, depicted in Figure 1, provides a mapping from the Packing List Id and asset Ids to actual locations within a Mapped File Set, which consists of the Asset Map and all referenced assets. An Asset Map can contain mappings for more than one package, but does not contain a partial mapping for a particular package. The Asset Map provides special identification for Packing List assets so that they can be located without scanning all assets within the package.

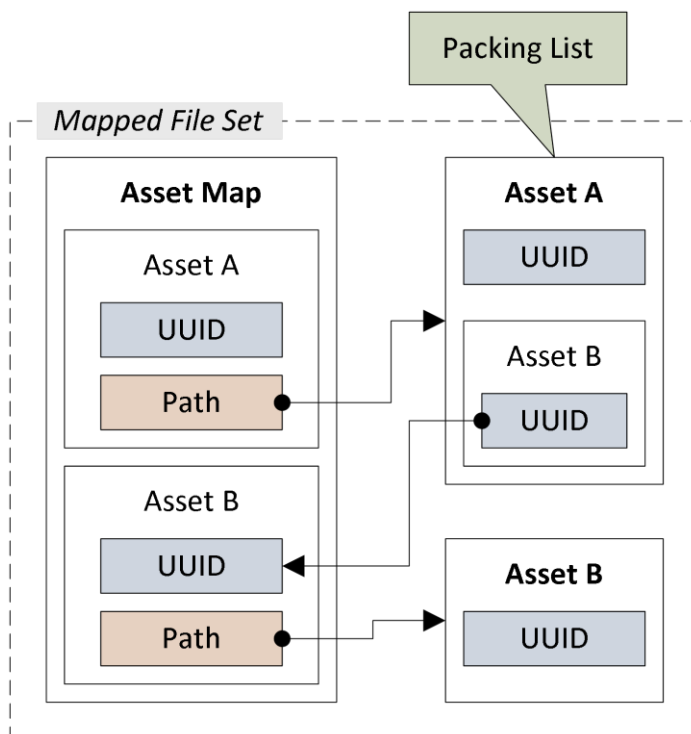


Figure 1 – Relationship between the Packing List and Asset Map structures

The definition of Mapped File Set and the means of locating the Asset Map are defined by a normative Map Profile (see Section 9) for that media type.

Annex A specifies a Map Profile for use with common hierarchical filesystems.

4.1 Ingesting a Mapped File Set (Informative)

The process of reading a package by an implementation is termed ingest. A system can provide a simple or elaborate ingest control interface — allowing coarse or fine-grained selection of items to be ingested — but the basic process will typically be the same.

As illustrated using pseudo-code in Figure 2, a Mapped File Set is ingested by first opening its Asset Map document. The means of identifying the Asset Map on the Mapped File Set is defined by the normative Map Profile (see Section 9) for that media type. The Asset Map is used to locate the Packing List(s) which detail the contents of the available package(s). Assets are chosen from the Packing List(s), and the selected assets are ingested.

```
Ingest Mapped File Set:
  open the Asset Map

  for each Packing List in the Asset Map
    open the Packing List

    for each Asset in the Packing List
      if this Asset is wanted
        locate the corresponding Asset structure in the Asset Map

        read the Asset and test the message digest
```

Figure 2 – Example ingest process

4.2 XML File Structure

The structures defined in this document are represented using the Extensible Markup Language (XML) [XML 1.0], and specified using XML Schema [XML Schema Part 1: Structures] and [XML Schema Part 2: Datatypes]. This specification shall be associated with a unique XML namespace name [Namespaces in XML]. The namespace name shall be the string value "<http://www.smpte-ra.org/schemas/429-9/2007/AM>". This namespace name conveys both structural and semantic version information, but does not serve the purpose of a traditional version number field.

Table 1 lists the XML namespace names used in this specification. Namespace names are represented as Uniform Resource Identifier (URI) values [RFC 3986]¹.

¹ Readers unfamiliar with URI values as XML namespace names should be aware that although a URI value begins with a “method” element (“http” in this case), the value is designed primarily to be a unique string and does not necessarily correspond to an actual on-line resource. Applications implementing this standard should not attempt to resolve URI values on-line.

Table 1 – XML Namespaces

Qualifier	URI
am	http://www.smpte-ra.org/schemas/429-9/2007/AM
xs	http://www.w3.org/2001/XMLSchema

The URIs found in Table 1 are normative. The namespace qualifier values (also called namespace prefixes in XML jargon) used in Table 1 and elsewhere in this document, namely "am" and "xs", are not normative. Specifically, they may be replaced in instance documents by any XML compliant namespace prefix. In other words, implementations shall expect any arbitrary XML compliant namespace prefix value that is associated with a URI from Table 1.

5 AssetMap Structure

An Asset Map document is an XML document with an AssetMap element as its root element. The elements and structure of an AssetMap element are illustrated in Figure 3 and defined in the subsections that follow.

An Asset Map may contain mappings for more than one package, but shall not contain a partial mapping for a particular package, i.e. a package shall not span multiple distinct Asset Maps.

A Mapped File Set shall contain exactly one Asset Map.

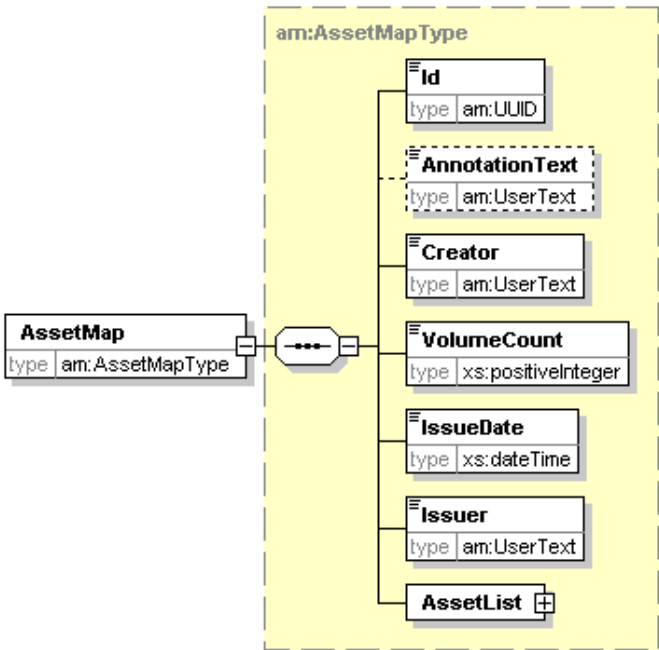


Figure 3 – AssetMap structure

5.1 Id

The Id element uniquely identifies the Asset Map structure. It is encoded as a urn:UUID [RFC 4122].