

SMPTE STANDARD

D-Cinema Operations —
Auxiliary Resource
Presentation List



Table of Contents	Page
Foreword	2
Intellectual Property	2
1 Scope	3
2 Conformance Notation	3
3 Normative References	3
4 Glossary	4
5 Overview (Informative)	4
5.1 Usage	4
5.2 Composition Playlist Mapping	4
5.3 Show Playlist Mapping	4
6 Document Description	5
6.1 Namespace	5
6.2 Graphical Representation	6
6.3 Schema Elements	6
7 Schema Definition	8
8 Example (Informative)	9
9 XML Diagram Legend (Informative)	10
9.1 Element Symbols	10
9.2 Model Symbols ("compositors")	11
9.3 Types	11
9.4 Model Groups and References	11
Annex A Bibliography (Informative)	13

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 Part XIII of its Administrative Practices.

SMPTE ST 430-11 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 standard defines a document for specifying the location of resources on a digital cinema server and the corresponding position at which they should be presented during the play out of a single composition or a show comprised of multiple compositions.

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.

A conformant implementation according to this document is one that includes all mandatory provisions ("shall") and, if implemented, all recommended provisions ("should") as described. A conformant implementation need not implement optional provisions ("may") and need not implement them as described.

Unless otherwise specified, the order of precedence of the types of normative information in this document shall be as follows: Normative prose shall be the authoritative definition; Tables shall be next; followed by formal languages; then figures; and then any other language forms.

3 Normative References

Note: All references in this document to other SMPTE documents use the current numbering style (e.g. SMPTE ST 429-7:2006) although, during a transitional phase, the document as published (printed or PDF) may bear an older designation (such as SMPTE 429-7-2006). Documents with the same root number (e.g. 429-7) and publication year (e.g. 2006) are functionally identical.

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.

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).

Internet Engineering Task Force (IETF) (November 1996) RFC1738 - Uniform Resource Locators (URL).

Internet Engineering Task Force (IETF) (1996, November). RFC 2396 – Uniform Resource Identifiers (URI): Generic Syntax.

Internet Engineering Task Force (IETF) (2005, July). RFC 4122 – A Universally Unique Identifier (UUID) URN Namespace.7.

SMPTE ST 429-7:2006, D-Cinema Packaging — Composition Playlist.

4 Glossary

The following paragraphs define the acronyms used in this document.

ACS: Auxiliary Content Server

CPL: Composition Playlist

DCS: Digital Cinema Server

HTTP: Hypertext Transfer Protocol

RPL: Auxiliary Resource Presentation List

5 Overview (Informative)

This standard defines a schema for the interchange of the list of auxiliary resources available to the playout of one or more compositions. The Auxiliary Resource Presentation List File identifies the auxiliary content resources needed for the current show or composition and their relative position on the timeline.

5.1 Usage

The RPL provides a list of auxiliary resources available in the next composition or sequence of compositions to be played. It is presented by a Digital Cinema Server (DCS) to an Auxiliary Content Server (ACS) at some time before playout begins. An ACS will obtain timeline updates from the DCS and determine which resources should be presented.

5.2 Composition Playlist Mapping

Each ReelResources element within the RPL correlates to a single reel within a CPL. Each ReelResource element corresponds to a single Reel Asset within a CPL. While any reel asset could be mapped into the RPL, the intention is to map lower bit rate resources not “rendered” by the DCS, e.g. subtitles, captions, and special effects. Resources are intended to be pre-fetched before start of playout to allow for synchronous playout with the first frame of content.

5.3 Show Playlist Mapping

Multiple compositions can be played in sequence to create a “show” timeline. To ensure continuity across CPL boundaries, an RPL can be created to represent the entire show timeline. Multiple ReelResources elements, representing the reels for multiple CPLs, can be present in a single RPL. In this case, the TimelineOffset attribute of the ReelResources element identifies the show-relative offset of the referenced resources.

6 Document Description

6.1 Namespace

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/430-11/2010/RPL”. This namespace name conveys both structural and semantic version information, and serves the purpose of a traditional version number field.

XML namespace names used in this standard are identified in Table 1. Namespace names are represented as Uniform Resource Identifier (URI) values [RFC 2396]¹.

Table 1 – XML Namespaces

Qualifier	URI
rpl	http://www.smpte-ra.org/schemas/430-11/2010/RPL

The URI listed in Table 1 is normative, whereas the namespace qualifier value (used in Table 1 and elsewhere in this standard) is not normative. Thus, namespace qualifier values may be replaced in instance documents by any arbitrary XML compliant namespace qualifier, meaning that conformant implementations shall expect any XML compliant namespace qualifier value that is associated with a URI from Table 1.

Datatypes from other schemas that are used in this document will be prefixed with the appropriate namespace qualifier (e.g. `xs:dateTime`). See [XML Schema Part 2: Datatypes] for further information about these types.

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.

6.2 Graphical Representation

Below is a graphical representation of the ResourcePresentationList schema.

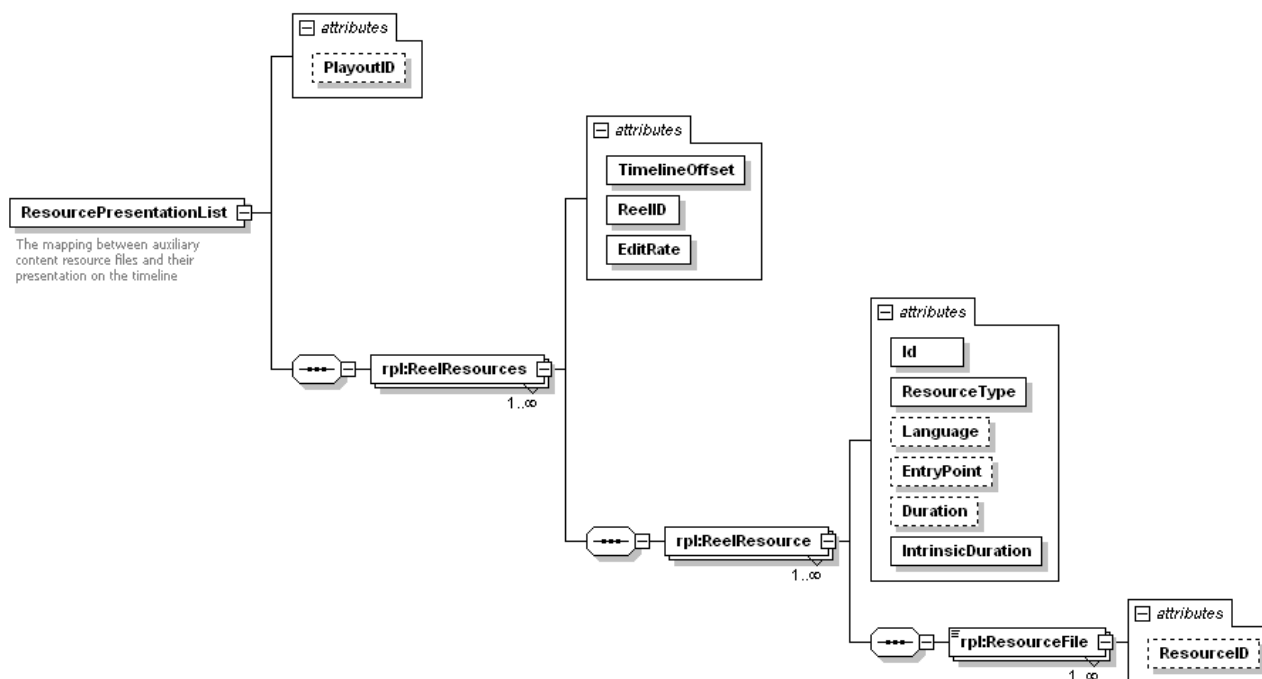


Figure 1 – ResourcePresentationList schema representation

6.3 Schema Elements

6.3.1 PlayoutID [optional]

The optional PlayoutID attribute is used to correlate the RPL to the current playout. The use of the PlayoutID is intended to prevent stale (incorrect) auxiliary resources from playing out with the current presentation.

6.3.2 ReelResources

Each ReelResources element defines the list of auxiliary resources that may be presented at the same point in the timeline. It contains attributes that apply to all child resources.

6.3.2.1 TimelineOffset

The TimelineOffset attribute defines the position (in edit units) in the timeline at which the child resources shall begin presentation.

6.3.2.2 ReelID

The ReelID identifies the Reel ID in the CPL associated with the ReelResources.

6.3.2.3 EditRate

The EditRate attribute defines the number of edit units per second for the associated CPL Reel.

6.3.3 ReelResource

The ReelResource element identifies all the attributes for a given auxiliary resource.

6.3.3.1 Id

The Id attribute identifies the Resource UUID.

6.3.3.2 ResourceType

The ResourceType attribute corresponds to the element name of a child element of the AssetList element in the CPL. Example ResourceTypes are MainSubtitle, MainCaption, ClosedCaption, and OpenCaption.

6.3.3.3 Language [optional]

The optional Language attribute corresponds to the Language attribute for assets from the same referenced CPL track.

6.3.3.4 EntryPoint [optional]

The optional EntryPoint attribute identifies the entry point in edit units from which this resource shall be offset. Each EntryPoint attribute corresponds to the EntryPoint attribute for the associated asset from the referenced CPL. Playout of this resource starts EntryPoint edit units from the start of this resource. If this attribute is not present, a value of 0 shall be assumed.

6.3.3.5 Duration [optional]

The optional Duration attribute identifies the duration of the playable region of the associated asset from the referenced CPL. The resource stops playing at EntryPoint plus Duration edit units. If this attribute is not present, asset playback shall stop after $(\text{IntrinsicDuration} - \text{EntryPoint})/\text{EditRate}$ seconds.

6.3.3.6 IntrinsicDuration

The IntrinsicDuration attribute defines the Native Duration of the associated asset from the referenced CPL. The IntrinsicDuration is the duration, in edit units, of the entire resource, including portions that are not to be played, such as that portion before EntryPoint.

6.3.4 ResourceFile

The ResourceFile element identifies a URL for a resource file. The URL may be absolute, including the protocol (e.g. HTTP), host IP address, path, and filename, or may be relative to the location of this file (specifying a relative or absolute path and the filename).

6.3.4.1 ResourceID [optional]

The optional ResourceID attribute identifies the UUID of this resource. This UUID is used to correlate resources referenced by UUID to their filename.

7 Schema Definition

The XML Schema document presented in this section normatively defines the structure of an RPL using a machine-readable language. While this schema is intended to faithfully represent the structure presented in the normative prose portions (Sections 3 through 6) of this document, conflicts in definition may occur. In the event of such a conflict, the normative prose shall be the authoritative expression of the standard.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:rpl="http://www.smpte-ra.org/schemas/430-11/2010/RPL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:dcml="http://www.smpte-
ra.org/schemas/433/2008/dcmlTypes/" targetNamespace="http://www.smpte-ra.org/schemas/430-
11/2010/RPL" elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:import namespace="http://www.smpte-ra.org/schemas/433/2008/dcmlTypes/"
schemaLocation="./dcmlTypes.xsd"/>
  <xs:element name="ResourcePresentationList">
    <xs:annotation>
      <xs:documentation>The mapping between auxiliary content resource files and their
presentation on the timeline</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ReelResources" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ReelResource" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="ResourceFile" maxOccurs="unbounded">
                      <xs:complexType>
                        <xs:simpleContent>
                          <xs:extension base="xs:anyURI">
                            <xs:attribute name="ResourceID" type="dcml:UUIDType" use="optional"/>
                          </xs:extension>
                        </xs:simpleContent>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:sequence>
              <xs:attribute name="Id" type="dcml:UUIDType" use="required"/>
              <xs:attribute name="ResourceType" type="xs:string" use="required"/>
              <xs:attribute name="Language" type="xs:language" use="optional"/>
              <xs:attribute name="EntryPoint" type="xs:long" use="optional"/>
              <xs:attribute name="Duration" type="xs:long" use="optional"/>
              <xs:attribute name="IntrinsicDuration" type="xs:long" use="required"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="TimelineOffset" type="xs:unsignedLong" use="required"/>
      <xs:attribute name="ReelID" type="dcml:UUIDType" use="required"/>
      <xs:attribute name="EditRate" type="dcml:RationalType" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:attribute name="PlayoutID" type="xs:unsignedInt" use="optional"/>
</xs:schema>
```


8 Example (Informative)

The RPL example below represents a show with 4 trailer compositions, each 120 seconds in duration, and 1 feature composition, 40 minutes in duration with 2 reels of 20 minutes each. Each composition has an edit rate of 24/1. Captions are only present in the feature composition, so the timeline offset for the first resource in the feature is set to $4 * 120 \text{ seconds} * 24 \text{ edit units/second} = 11520 \text{ edit units}$.

In this example, each ReelResources element corresponds to a single reel in the feature composition.

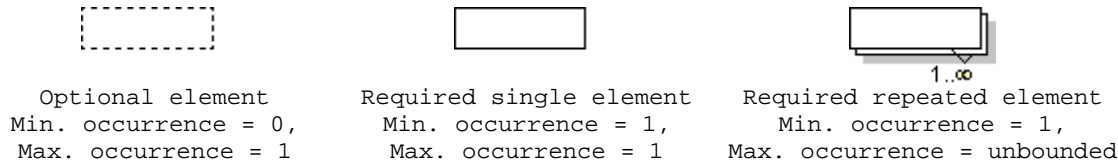
```
<?xml version="1.0" encoding="UTF-8"?>
<rpl:ResourcePresentationList PlayoutID="49520318" xsi:schemaLocation="http://www.smpte-
ra.org/schemas/430-11/2010/RPL AuxResourcePresentationList.xsd" xmlns:rpl="http://www.smpte-
ra.org/schemas/430-11/2010/RPL" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <rpl:ReelResources ReelID="urn:uuid:2fd9f048-5646-4b75-b0e7-8839d28a4813" EditRate="24 1"
TimelineOffset="11520">
    <rpl:ReelResource Language="en-us" Duration="28800" EntryPoint="0" ResourceType="ClosedCaption"
Id="urn:uuid:2fd9f048-5646-4b75-b0e7-8839b1a395c9" IntrinsicDuration="28800">
      <rpl:ResourceFile>http://192.168.1.1/reel1/caption_en-us_r1.xml</rpl:ResourceFile>
    </rpl:ReelResource>
    <rpl:ReelResource Language="fr" Duration="28800" EntryPoint="0" ResourceType="ClosedCaption"
Id="urn:uuid:2fd9f048-5646-4a54-b2e7-8839b1a39872" IntrinsicDuration="28800">
      <rpl:ResourceFile>http://192.168.1.1/reel1/caption_fr_r1.xml</rpl:ResourceFile>
    </rpl:ReelResource>
    <rpl:ReelResource Duration="28800" EntryPoint="0" ResourceType="ClosedSubtitle"
Id="urn:uuid:2fd9f048-5646-4b75-b0e7-8839b1a395d0" IntrinsicDuration="28800">
      <rpl:ResourceFile>http://192.168.1.1/reel1/closedSubtitle_r1_file1.xml</rpl:ResourceFile>
    </rpl:ReelResource>
  </rpl:ReelResources>
  <rpl:ReelResources ReelID="urn:uuid:2fd9f048-5646-4b75-b0e7-883959ae2c4a" EditRate="24 1"
TimelineOffset="40320">
    <rpl:ReelResource Language="en-us" Duration="28800" EntryPoint="0" ResourceType="ClosedCaption"
Id="urn:uuid:2fd9f048-5646-4b75-b0e7-8839b1a395f0" IntrinsicDuration="28800">
      <rpl:ResourceFile>http://192.168.1.1/reel2/caption_en-us_r2.xml</rpl:ResourceFile>
    </rpl:ReelResource>
    <rpl:ReelResource Language="fr" Duration="28800" EntryPoint="0" ResourceType="ClosedCaption"
Id="urn:uuid:2fd9f048-5646-4a54-b2e7-8839b1a39874" IntrinsicDuration="28800">
      <rpl:ResourceFile>http://192.168.1.1/reel2/caption_fr_r2.xml</rpl:ResourceFile>
    </rpl:ReelResource>
    <rpl:ReelResource Duration="28800" EntryPoint="0" ResourceType="ClosedSubtitle"
Id="urn:uuid:2fd9f048-5646-4b75-b0e7-8839b1a39432" IntrinsicDuration="28800">
      <rpl:ResourceFile>http://192.168.1.1/reel2/closedSubtitle_r2_file1.xml</rpl:ResourceFile>
    </rpl:ReelResource>
  </rpl:ReelResources>
</rpl:ResourcePresentationList>
```

9 XML Diagram Legend (Informative)

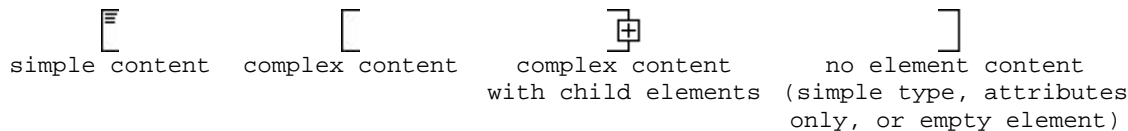
The following provides a legend for notation used in diagrams depicting XML structures.

9.1 Element symbols

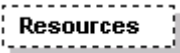
In the schema design diagrams presented above of this document, only the elements are drawn. Attributes are not visible. The cardinality of the element (0..1, 1 exactly, 0..n, 1..n) is indicated by the border of the elements. Optional elements are drawn with a dashed line, required elements with a solid line. A maximum occurrence greater one is indicated by a double border.



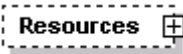
The content model of elements is symbolized on the left and right side of the element boxes. The left side indicates whether the element contains a simple type (text, numbers, dates, etc.) or a complex type (further elements). The right side of the element symbol indicates whether it contains child elements or not:



9.1.1 Examples



Optional single element without child elements. Minimum Occurrence = 0, Maximum Occurrence = 1, content = complex.



As above, but with child elements. the "plus" at the right side indicates the presence of one or more undisplayed child elements.



Mandatory single element. Minimum Occurrence = 1, Maximum Occurrence = 1, content = complex, no child elements (i.e. this denotes an *empty element*). The gray or green text below the element displays the xml-schema annotation associated with the element.



Mandatory multiple element containing child elements (content = complex). This element must occur at least once (Minimum Occurrence = 1) and may occur as often as desired (Maximum Occurrence = unbounded).

InternalNotes

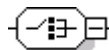
Mandatory single element with containing simple content (e.g. text) or mixed complex content (e.g. text with xhtml markup). Minimum Occurrence = 1, Maximum Occurrence = 1, type = xsd:string (for example), content = simple. The three lines in the upper left corner are used for both text and numeric content.

9.2 Model Symbols ("compositors")

A sequence of elements. The elements must appear exactly in the sequence in which they appear in the schema diagram.



A choice of elements. Only a single element from those in the choice may appear at this position.

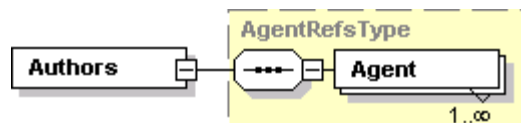


The "all" model, in which the sequence of elements is not fixed.

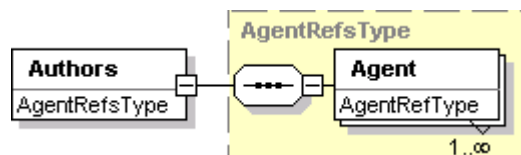


9.3 Types

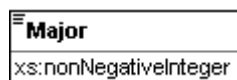
If an element refers to a complex global type, the type is shown with a border and yellow background. You can click on the gray type name shown at the top to jump to the type definition itself.



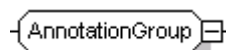
Depending on the settings in the xml editor when generating the schema diagrams, the type name may be shown in the line below the element name:



In that case, the type names of simple types are shown as well:



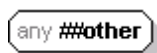
9.4 Model Groups and References



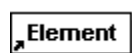
An *element group* is a named container with one or several elements. The group of elements can be reused at multiple places in the schema. Model groups are invisible in the instance document (in contrast to types,

which require). Model groups have been used sparingly since they do not map to a feature in object-oriented programming languages (unless they support multiple inheritance).

Important note on reading the diagrams for model groups: If the model group symbol is drawn with simple lines (i.e. not dashed), this does not imply that the elements in the model group are required. The optionality of the group depends on the optionality of elements contained in the model group. (Model groups can be made optional, e.g. to make a model group with required elements optional in some cases, but this has not been used.)



The "any" group is a special kind of model group. It is a placeholder for elements not defined in the schema. The "any" element defines points where the schema can be extended. After the "Any" keyword the namespace from which the elements may come is defined, for example, "##other" specifies that the extension elements may come from any namespace, except from the current schema namespace.



Element references are indicated through a link arrow in the lower left corner. They are similar to references to model groups within a schema, but instead of refining the model group, they directly refer to a single global element. The global element can then be reused in multiple places.

Annex A (Informative) **Bibliography**

SMPTE ST 430-10:2010, D-Cinema Operations — Auxiliary Content Synchronization Protocol