

## **MPEG Interoperability Initiative: Packing List Specification**

*Application Specification for Digital Cinema Packaging (AS-DCP)*

**NOTICE**

**This document is provided without warranty as to its fitness for a particular purpose.**

# Contents

Page

1	Packing List SpecificationScope .....	2
2	Normative References.....	2
3	Overview.....	4
3.1	Use of XML Language.....	5
4	PackingList Structure .....	6
4.1	Id .....	6
4.2	AnnotationText [optional].....	6
4.3	IconId [optional].....	6
4.4	IssueDate.....	6
4.5	Issuer .....	6
4.6	Creator .....	6
4.7	AssetList .....	7
4.8	Signer [optional] .....	7
4.9	Signature [optional].....	7
5	Asset Structure.....	8
5.1	Id .....	8
5.2	AnnotationText [optional].....	8
5.3	Hash.....	8
5.4	Size .....	8
5.5	Type .....	8
5.6	OriginalFileName [optional].....	9
6	XML Schema .....	10
6.1	PackingList .....	10
6.2	Asset.....	10
7	Bibliography.....	10
8	Annex A - Sample [Informative] .....	11
9	Change History .....	11

## 1 Packing List Scope

This standard specifies the data format for interchange of a Packing List for Digital Cinema applications.

The electronic or physical form of a complete package described by a Packing List is beyond the scope of this standard. This mapping is defined in the Delivery Media Representation and Segmentation document [DDMR].

## 2 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.

[SHA1] Internet Engineering Task Force (IETF) RFC3174 (September 2001) “US Secure Hash Algorithm 1”  
<http://www.ietf.org/rfc/rfc3174.txt>

[MIME] Internet Engineering Task Force (IETF) RFC2045 (November 1996) *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies* URL <http://www.ietf.org/rfc/rfc2045.txt>

The Internet Engineering Task Force (IETF) RFC2046 (November 1996) *Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types* URL <http://www.ietf.org/rfc/rfc2046.txt>

[XML] World Wide Web Consortium (W3C) Recommendation (06 October 2000). *Extensible Markup Language (XML)* <http://www.w3.org/TR/2000/REC-xml-20001006>

[XSCH] World Wide Web Consortium (W3C) Recommendation (2 May 2001). *XML Schema Part 1: Structures* URL <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>

[DIGS] World Wide Web Consortium (W3C) Recommendation (12 February 2002). *XML-Signature Syntax and Processing* URL <http://www.w3.org/TR/2002/REC-xmlsig-core-20020212/>

[UUID] The Internet Engineering Task Force (IETF), A UUID URN Namespace [WWW document]. URL <http://www.ietf.org/internet-drafts/draft-mealling-uuid-urn-03.txt> (update)

MPEG Interop: Composition Playlist Specification

### 3 Overview

A packing list, depicted in abstract form in Figure 1, is a list of identification information about a distribution package. Said another way, a packing list describes a particular distribution package by enumerating its contents. A distribution package contains track files, one or more composition playlist files and possibly other files. The information contained in the packing list allows a receiver to validate the whole package upon receipt. The packing list's elements are not ordered.

Figure 1 illustrates the abstract form of a complete packing list for a trailer and a single reel feature.

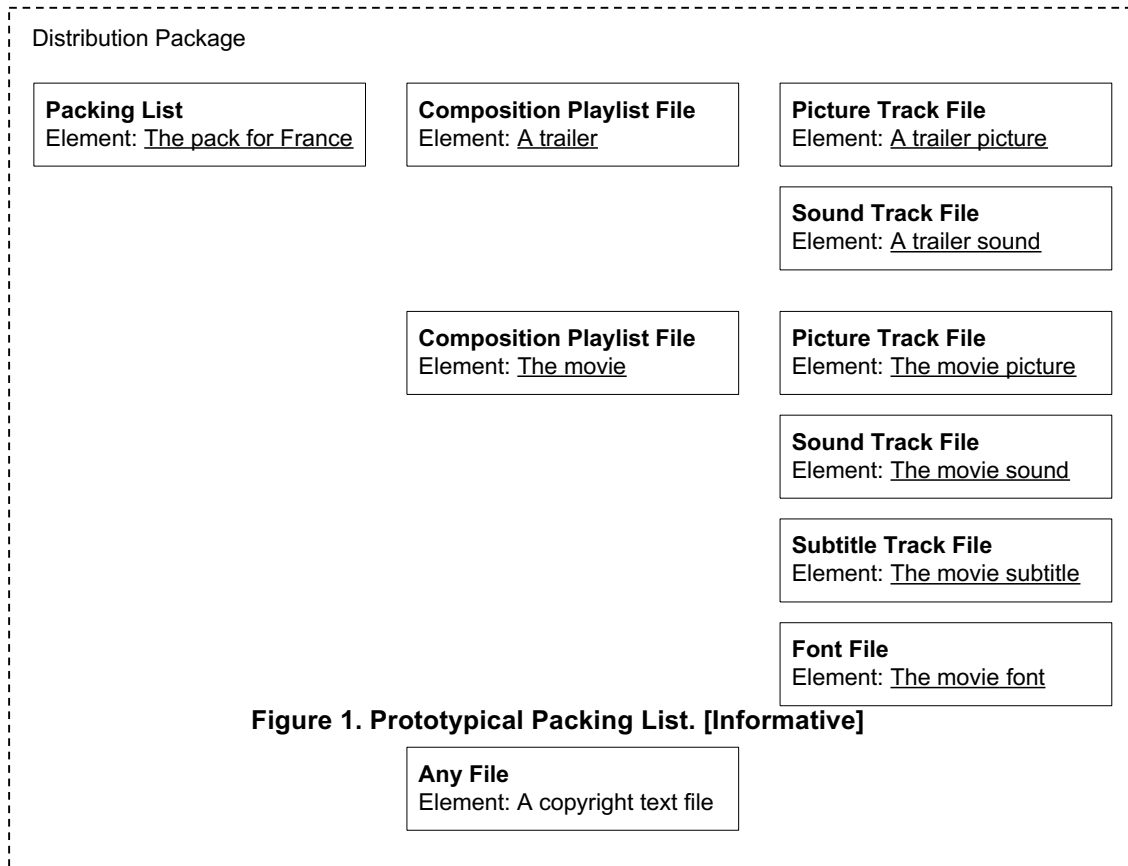


Figure 1. Prototypical Packing List. [Informative]

Figure 2 illustrates the abstract form of a partial packing list, in which a replacement “reel” is being sent to theatres. This may be created, for example, to distribute a change in the end credits of a feature. Note that this partial package includes picture, sound and sub-title track files, and a composition playlist file that references those files as well as files from a previously delivered package. The example package shown in Figure 2 is not related to the example package shown in Figure 1.

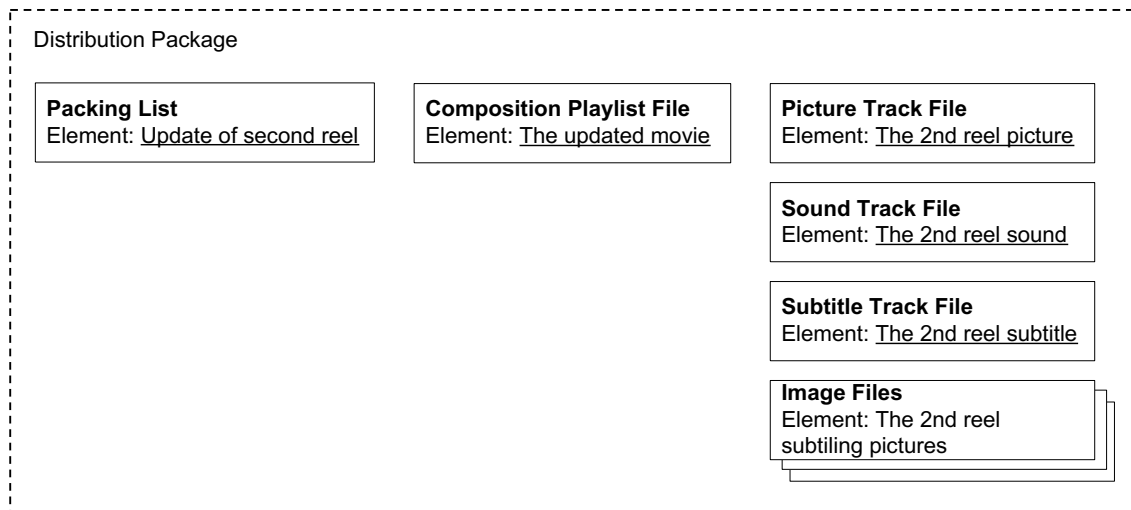


Figure 2.

Prototypical Partial Packing List. [Informative]

### 3.1 Use of XML Language

The structures defined in this document are represented using the XML language [XML], and specified using the W3C schema language [XSCH]. This version of the specification is associated with a unique XML namespace, namely <http://www.digicine.com/PROTO-ASDCP-PKL-20040311#>. This namespace conveys both structural and semantic version information, and serves the purpose of a traditional version number field.

The packing list shall be encoded using the UTF-8 character encoding scheme.

## 4 PackingList Structure

A packing list is encoded as an XML document [XML]. The top-level element is designated *PackingList*, and is described below. See the XML Schema declaration in Section 6 of this document for explicit type information.

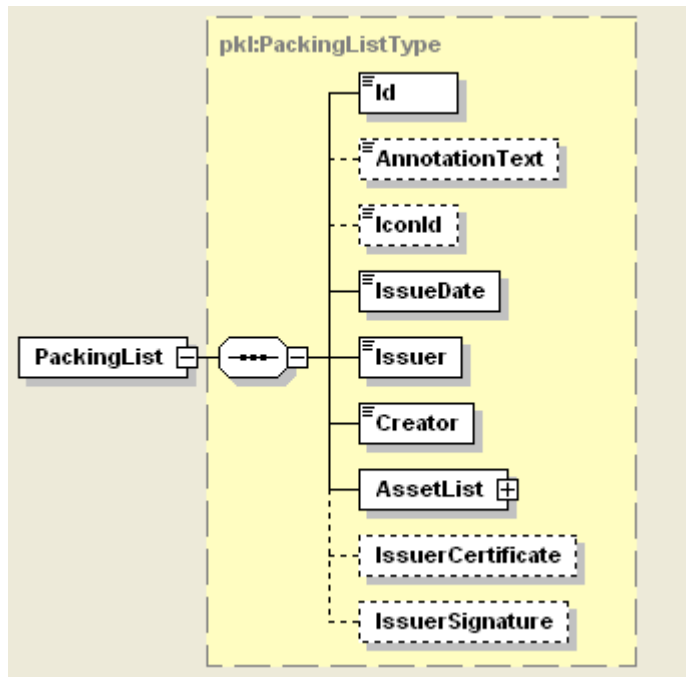


Figure 3. Packing List structure. Dotted lines denote an optional element.

### 4.1 Id

The *Id* parameter uniquely identifies the distribution package for asset management purposes. It is encoded as a UUID [UUID].

### 4.2 AnnotationText [optional]

The *AnnotationText* parameter is a free-form, human-readable annotation describing the distribution package. It is meant strictly as a displayed guidance for the user.

### 4.3 IconId [optional]

The *IconId* parameter uniquely identifies an external file containing a picture icon illustrating the packing list. It is encoded as a UUID. The icon may be rendered, for instance, from a frame of the underlying content.

### 4.4 IssueDate

The *IssueDate* parameter indicates the time and date at which the packing list was issued. The *IssueDate* is formatted according to the 'dateTime' format defined in <http://www.w3.org/XML/Schema> [XMLS].

### 4.5 Issuer

The *Issuer* parameter is a free-form, human-readable annotation describing the person or company who has created the packing list. It meant strictly as a displayed guidance for the user.

### 4.6 Creator

The *Creator* parameter is a free-form, human-readable annotation describing the system (hardware/software) used to create the packing list. It meant strictly as a displayed guidance for the user.

#### 4.7 AssetList

The *AssetList* parameter contains an unordered list of *Asset* elements (see Sec. 5 below) contained in the package. The structure of the *Asset* element is described in Section 5 of this document.

#### 4.8 Signer [optional]

The Signer parameter uniquely identifies the entity, and hence public key, that digitally signs the packing list. It shall be an instance of the KeyInfo type defined in W3C XML digital signature standard [DIGS]. The Key Info element shall contain an X509IssuerSerial element.

The X509IssuerSerial element contains two elements: the X509IssuerName and the X509SerialNumber.

The Signer parameter is required if the packing list is signed.

#### 4.9 Signature [optional]

The Signature parameter contains a digital signature authenticating the packing list. It shall be an instance of the SignatureType type defined in the W3C XML digital signature standard [DIGS]. The digital signature is enveloped and applies to the entire packing list. It is generated by the signer, as identified by the Signer parameter, using its private key.

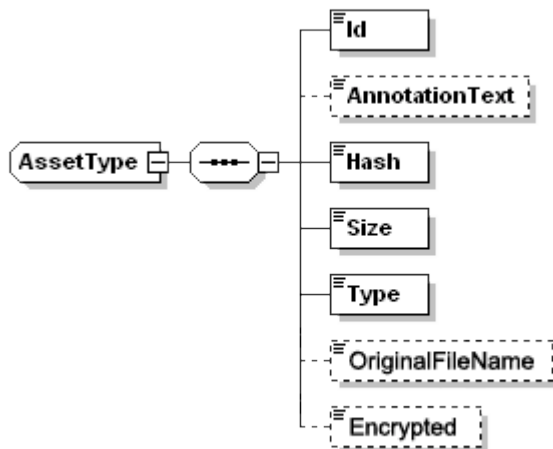
The standard Signature element is a highly flexible construct, which can adapt to a wide range of applications. For the purpose of the packing list, it shall satisfy the following constraints.

- The KeyInfo element shall be present, and contain the entire certificate chain for the signer.
- The Object element shall not be present
- The Reference element URI shall be set to "", since the signature is enveloped.
- The Reference element shall contain a single DigestMethod element, with its Algorithm attribute set to <http://www.w3.org/2000/09/xmldsig#sha1>.
- The Reference element shall contain a single Transform element, with its Algorithm attribute set <http://www.w3.org/2000/09/xmldsig#enveloped-signature>.
- The CanonicalizationMethod shall be <http://www.w3.org/TR/2001/REC-xml-c14n-20010315>.
- The SignatureMethod shall be <http://www.w3.org/2000/09/xmldsig#rsa-sha1>.

The entire certificate chain shall be carried in the KeyInfo element as a sequence of X509Data elements. Each of the X509Data elements shall correspond to one certificate in the chain, and contain one X509IssuerSerial element and one X509Certificate element.

## 5 Asset Structure

A packing list contains a set of files. Each file is described by an XML *Asset* element, described below. See the XML Schema declaration in Section 6 of this document for explicit type information.



**Figure 4. Asset Structure.** (remove 'encrypted') Dotted lines denote an optional element.

### 5.1 Id

The *Id* parameter uniquely identifies the asset for management purposes. It is represented by a UUID. The UUID must be extracted from the asset. The allocation of the UUID is done by the creator of the asset.

### 5.2 AnnotationText [optional]

The *AnnotationText* parameter, if present, is a free-form, human-readable annotation associated with the asset. It is meant strictly as a displayable guidance for the user. Here are some examples:

```
<AnnotationText>When Pigs Will Fly II - Swedish Package</AnnotationText>  
  
<AnnotationText>When Pigs Will Fly II - new reel 5 with outtakes -  
Swedish Package</AnnotationText>
```

### 5.3 Hash

The Hash parameter contains a hash of the underlying track file computed using the SHA-1 algorithm. When combined with the digital signature included in the composition playlist, it may be used to verify the integrity and authenticity of the underlying track file. The resulting 160 bit integer is encoded using base64 representation. SHA-1 is defined by IETF RFC3174 [SHA1]

### 5.4 Size

The Size parameter contains the size of the asset. This size is expressed in bytes.

### 5.5 Type

The *Type* parameter describes the MIME type of the asset file. It is meant both as a displayed guidance for the user and a machine-interpretable information for content reception processing. MIME types are defined in the IETF RFC2045 and RFC2046. The *Type* parameter may contain any valid MIME type string. MIME type strings for file formats defined by DC28.0 will be designated in the respective standards documents. MIME types defined for use within the MPEG Interop Initiative are listed in Table 2.



**Table 1. Asset Type (informative).**

Asset Type Enumeration	
<i>Type</i>	<i>Description</i>
<code>text/xml;asdcKind=CPL</code>	A composition playlist file
<code>application/x-smpte-mxf;asdcKind=Picture</code>	A picture track file
<code>application/x-smpte-mxf;asdcKind=Sound</code>	A sound track file
<code>application/ttf</code>	A TrueType font file for subtitling
<code>image/png</code>	A PNG image file for subtitling

**Table 2. Asset Type (normative)**

Asset Type Enumeration	
<i>Type</i>	<i>Description</i>
<code>text/xml;asdcKind=Subtitle</code>	A subtitle track File
<code>text/xml;asdcKind=ProjectorData</code>	A Projector Metadata track File
<code>application/x-pcf</code>	Projector configuration file

### **5.6 OriginalFileName [optional]**

The *OriginalFileName* parameter describes the original file name of the asset. It is meant strictly as a displayable guidance for the user.

## 6 XML Schema

```
<xs:schema targetNamespace=" http://www.digicine.com/PROTO-ASDCP-PKL-20040311#"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:pkl="
  http://www.digicine.com/PROTO-ASDCP-PKL-20040311#" xmlns:cpl="
  http://www.digicine.com/PROTO-ASDCP-CPL-20040511#"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
...
</xs:schema>
```

### 6.1 PackingList

```
<xs:element name="PackingList" type="pkl:PackingListType"/>
<xs:complexType name="PackingListType">
  <xs:sequence>
    <xs:element name="Id" type="cpl:UUID"/>
    <xs:element name="AnnotationText" type="cpl:UserText" minOccurs="0"/>
    <xs:element name="IconId" type="cpl:UUID" minOccurs="0"/>
    <xs:element name="IssueDate" type="xs:dateTime"/>
    <xs:element name="Issuer" type="cpl:UserText"/>
    <xs:element name="Creator" type="cpl:UserText"/>
    <xs:element name="AssetList">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="pkl:Asset" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="Signer" type="ds:KeyInfoType" minOccurs="0"/>
    <xs:element ref="ds:Signature" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

### 6.2 Asset

```
<xs:element name="Asset" type="pkl:AssetType"/>
<xs:complexType name="AssetType">
  <xs:sequence>
    <xs:element name="Id" type="cpl:UUID"/>
    <xs:element name="AnnotationText" type="cpl:UserText" minOccurs="0"/>
    <xs:element name="Hash" type="xs:base64Binary"/>
    <xs:element name="Size" type="xs:positiveInteger"/>
    <xs:element name="Type" type="xs:string"/>
    <xs:element name="OriginalFileName" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

## 7 Bibliography

Ref	Author, Date: Title (URL)
KMDC	SMPTE DC28.30 Key Management AHG Specification for Digital Cinema Role Certificate
TFIL	SMPTE DC28.20 Packaging AHG AS-DCP Track File Specification
DDMR	Delivery Media Representation and Segmentation

## 8 Annex A - Sample [Informative]

The following Packing List sample XML structure is a valid instance of the Packing List schema. It is not-functional and meant for informative purposes only. The optional *Signer* and *Signature* parameters were omitted for the sake of simplicity.

[probably needs updating – is cpl namespace needed?]

```
<?xml version="1.0" encoding="UTF-8"?>
<PackingList xmlns="http://www.digicine.com/PROTO-ASDCP-PKL-20040311#"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <Id>urn:uuid:00000000-0000-0000-0000-000000000000</Id>
  <AnnotationText>When Pigs Will Fly II - Swedish Package</AnnotationText>
  <IconId>urn:uuid:00000000-0000-0000-0000-000000000000</IconId>
  <IssueDate>2001-12-17T09:30:47-05:00</IssueDate>
  <Issuer>Distribution Center XYZ</Issuer>

  <AssetList>
    <Asset>
      <Id>urn:uuid:00000000-0000-0000-0000-000000000000</Id>
      <AnnotationText>Reel #1 Picture</AnnotationText>
      <Hash>54852082a163b46d8a8163b46d8a9de13ca88952</Hash>
      <Size>123456789</Size>
      <Type>application/x-smpte-mxf;asdcKind=Picture</Type>
    </Asset>
    <Asset>
      <Id>urn:uuid:00000000-0000-0000-0000-000000000000</Id>
      <AnnotationText>Reel #1 Sound</AnnotationText>
      <Hash>63b46d8a9de13ca8895254852082a163b46d8a81</Hash>
      <Size>56789</Size>
      <Type>application/x-smpte-mxf;asdcKind=Sound</Type>
    </Asset>
    <Asset>
      <Id>urn:uuid:00000000-0000-0000-0000-000000000000</Id>
      <AnnotationText>Composition List</AnnotationText>
      <Hash>2a163b46d8a8163b46d8a9de154852083ca88952</Hash>
      <Size>234</Size>
      <Type>text/xml;asdcKind=CPL</Type>
    </Asset>
  </AssetList>
</PackingList>
```

## 9 Change History

Ver	Date	By	Sect	Description
1	8 June 2004			- Original modification of SMPTE doc.
2	15 June 2004			- Addition of review comments
2.2	18 June 2004			- More review comments
2.3	13 August 2004			- Schema correction to Hash
2.4	27 May, 2005			- Added previously undefined MIME types in Table 2.
				-