



# Asset Ordering, Delivery and Tracking

**Showing changes from 2/21/19**

---

## CONTENTS

1	Introduction .....	1
1.1	Overview.....	1
1.2	Document Organization .....	2
1.3	Document Notation and Conventions .....	3
1.3.1	XML Conventions .....	3
1.3.2	General Notes .....	4
1.4	Normative References .....	5
1.5	Informative References.....	5
1.6	Best Practices for Maximum Compatibility.....	5
2	General Types Encoding .....	7
2.1	Attribute Groups .....	7
2.1.1	RangeAttributes.....	7
2.2	Simple Types.....	7
2.3	Message and Terms Types .....	7
2.3.1	DeliverySource-type and DeliveryReverseSource-type.....	7
2.3.2	DeliveryHandling-type .....	10
2.3.3	DeliveryInstructions-type .....	10
2.3.4	DeliveryParams-type .....	10
2.3.5	DeliveryIdentification-type .....	11
2.4	Types that reference objects directly .....	12
2.4.1	DeliveryObjectReference-type.....	12
2.5	Technical Characteristics.....	14
2.5.1	Interpretation of terms within Technical Attributes .....	14
2.5.2	TechAudio-type .....	15
2.5.3	TechVideo-type .....	16
2.5.4	TechSubtitle-type.....	19
2.5.5	TechCard-type.....	20
2.5.6	DeliveryImage-type.....	21
2.5.7	TechMetadata-type.....	22
2.5.8	TechContainer-type .....	23
3	Content Delivery Requirements .....	24
3.1	Requirements Structure .....	24
3.1.1	Scope .....	24
3.1.2	Profiles .....	25
3.1.3	Profile Examples.....	26
3.2	DeliveryRequirement-type .....	28
3.3	CategoryRules-type .....	29
3.3.1	TerritoryRules-type .....	30
3.3.2	Language Rules .....	30
3.3.3	Rating Rules.....	32
4	Profiles.....	33
4.1	Administrative Profile.....	33

4.2	Product Profiles .....	33
4.2.1	Profiles-type .....	33
4.2.2	ProductProfile-type .....	34
4.2.3	ProductProfileInfo-type .....	34
4.2.4	ProductPromotional-type .....	35
4.2.5	ProductSupplemental-type .....	35
4.3	Artwork Profiles .....	35
4.3.1	ArtworkProfile-type .....	36
4.3.2	ArtworkImage-type .....	36
4.4	Technical Profiles .....	37
4.4.1	TechnicalProfile-type .....	38
5	Asset Order.....	40
5.1	AssetOrder-type .....	40
5.1.1	AssetOrderObject-type .....	40
5.1.2	AssetOrderTerms-type .....	41
6	Asset Availability (ASM).....	42
6.1	AssetAvailability-type.....	42
6.1.1	AssetAvailabilityObject-type .....	43
7	QC Report.....	45
7.1	QCReport-type .....	45
7.1.1	QCError-type .....	46
7.1.2	QCErrorDescription-type .....	46
7.1.3	QCCategoryError-type.....	47
8	Product Status .....	51
8.1	Asset Status .....	52
8.1.1	Progress Codes.....	53
8.2	Logs.....	53
8.2.1	ProductLog-type .....	53
8.2.2	ProductLogEvent-type .....	53
9	NOTES .....	55



This work is licensed under a [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/).

**NOTE:** No effort is being made by the Motion Picture Laboratories to in any way obligate any market participant to adhere to Common Metadata. Whether to adopt the Common Metadata in whole or in part is left entirely to the individual discretion of individual market participants, using their own independent business judgment. Moreover, Motion Picture Laboratories disclaims any warranty or representation as to the suitability of the Common Metadata for any purpose, and any liability for any damages or other harm you may incur as a result of subscribing to this Common Metadata.



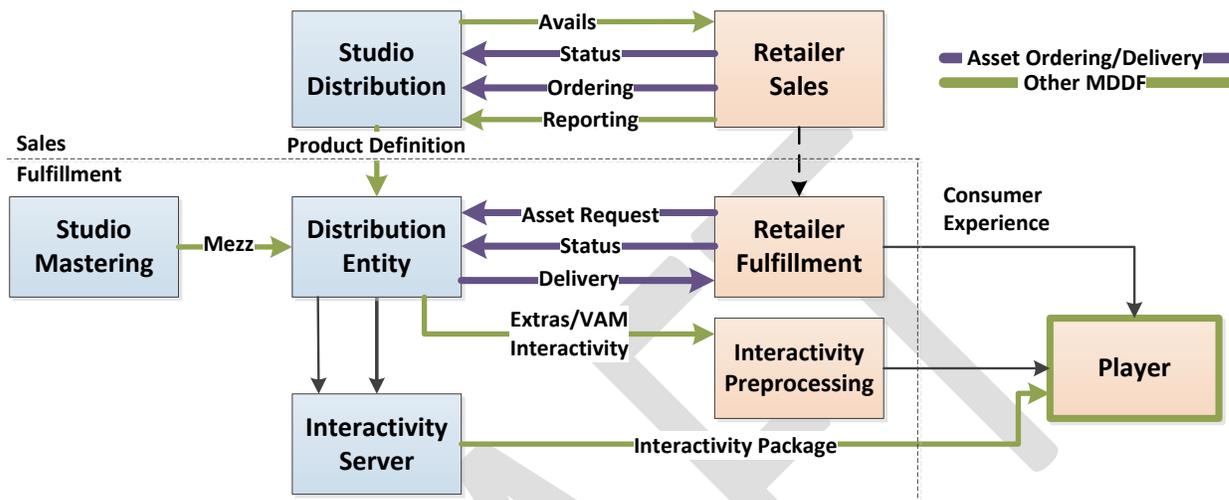
---

## REVISION HISTORY

Version	Date	Description
1.0		Original Version

## 1 INTRODUCTION

This document defined data used in the delivery of assets, within the MovieLabs Digital Distribution Framework (MDDF). The following illustration shows the MDDF flow, with Asset Ordering and Delivery data shown in purple.



This specification is designed to work with other MDDF specifications or with proprietary/legacy specifications.

### 1.1 Overview

The Asset Ordering and Delivery Process is addressed in three parts

- Rights Management – Generation and delivery of Avails or Title List
- Asset Planning – All processes associated with determining which assets (audio, video, subtitles, artwork, metadata, etc.) will be delivered
- Asset Delivery – Processes associated with the delivery of assets

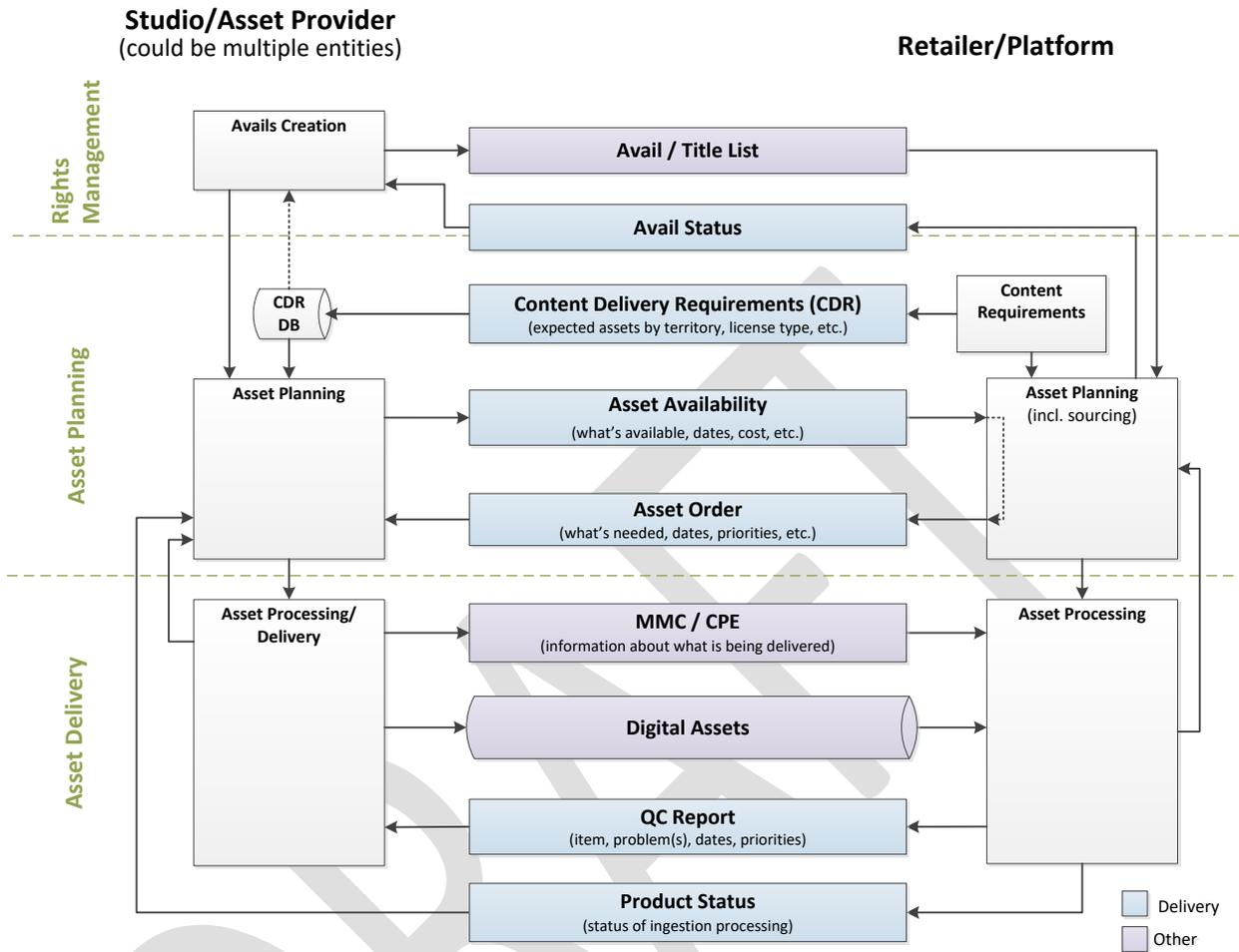
These are illustrated in [Figure 1](#) below.

The Rights Management process is covered by Avails and is not further discussed in this document. See [www.movielabs.com/md/avails](http://www.movielabs.com/md/avails) for more information.

Asset Planning is further divided into asset policies that span Avails, and Avail-specific or titles-specific asset selection. Asset policies are captured in “Content Delivery Requirements”. Avail or title-specific requests are included in Avail Confirmations, Asset Orders, and Asset [Status Manifests Availability](#).

Asset Delivery has several parts including a Media Manifest Core (MMC) delivery spec, the assets themselves, QC failure reports that document issues, and Asset Status information. MMC is documented elsewhere ([www.movielabs.com/md/mmc](http://www.movielabs.com/md/mmc)), and this specification is neutral to assets delivered—we attempt to support almost any format. This specification documents the QC failure reports and Asset Status data.

**Figure 1: Asset Distribution Workflow**



## 1.2 Document Organization

This document is organized as follows:

1. Introduction—Provides background, scope and conventions
2. General Types Encoding
3. Content Delivery Requirements
4. Profiles
5. Asset Planning and Delivery
6. Asset Order
7. Asset ~~Status Manifest (ASM)~~Availability
8. QC Fail Report
9. Asset Status

---

## 1.3 Document Notation and Conventions

As a general guideline, the key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119]. That is:

- “MUST”, “REQUIRED” or “SHALL”, mean that the definition is an absolute requirement of the specification.
- “MUST NOT” or “SHALL NOT” means that the definition is an absolute prohibition of the specification.
- “SHOULD” or “RECOMMENDED” mean that there may be valid reasons to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- “SHOULD NOT” or “NOT RECOMMENDED” mean that there may be valid reasons when the particular behavior is acceptable, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- “MAY” or “OPTIONAL” mean the item is truly optional, however a preferred implementation may be specified for OPTIONAL features to improve interoperability.

Terms defined to have a specific meaning within this specification will be capitalized, e.g. “Track”, and should be interpreted with their general meaning if not capitalized.

Normative key words are written in all caps, e.g. “SHALL”.

Normative requirements need not use the formal language above.

### 1.3.1 XML Conventions

XML is used extensively in this document to describe data. It does not necessarily imply that actual data exchanged will be in XML. For example, JSON may be used equivalently.

This document uses tables to define XML structure. These tables may combine multiple elements and attributes in a single table. Although this does not align with schema structure, it is much more readable and hence easier to review and to implement.

Although the tables are less exact than XSD, the tables should not conflict with the schema. Such contradictions should be noted as errors and corrected.

#### 1.3.1.1 Naming Conventions

This section describes naming conventions for Common Metadata XML attributes, element and other named entities. The conventions are as follows:

- Names use initial caps, as in InitialCaps.
- Elements begin with a capital letter, as in InitialCapitalElement.

- Attributes begin with a lowercase letter, as in `initialLowercaseAttribute`.
- XML structures are formatted as Courier New, such as `md:id-type`
- Names of both simple and complex types are followed with “-type”

### 1.3.1.2 Structure of Element Table

Each section begins with an information introduction. For example, “The Bin Element describes the unique case information assigned to the notice.”

This is followed by a table with the following structure.

The headings are

- Element—the name of the element.
- Attribute—the name of the attribute
- Definition—a descriptive definition. The definition may define conditions of usage or other constraints.
- Value—the format of the attribute or element. Value may be an XML type (e.g., “string”) or a reference to another element description (e.g., “See Bar Element”). Annotations for limits or enumerations may be included (e.g., “int [0..100]” to indicate an XML `xs:int` type with an accepted range from 1 to 100 inclusively)
- Card—cardinality of the element. If blank, then it is 1. Other typical values are 0..1 (optional), 1..n and 0..n.

The first row of the table after the header is the element being defined. This is immediately followed by attributes of this element, if any. Subsequent rows are child elements and their attributes. All child elements (i.e., those that are direct descendants) are included in the table. Simple child elements may be fully defined here (e.g., “Title”, “”, “Title of work”, “xs:string”), or described fully elsewhere (“POC”, “”, “Person to contact in case there is a problem”, “md:ContactInfo-type”). In this example, if POC was to be defined by a complex type defined as `md:ContactInfo-type`. Attributes immediately follow the containing element.

Accompanying the table is as much normative explanation as appropriate to fully define the element, and potentially examples for clarity. Examples and other informative descriptive text may follow. XML examples are included toward the end of the document and the referenced web sites.

### 1.3.2 **General Notes**

All required elements and attributes must be included.

When enumerations are provided in the form ‘enumeration’, the quotation marks (‘’) should not be included.

UTF-8 [RFC3629] encoding shall be used when ISO/IEC 10646 (Universal Character Set) encoding is required.

---

## 1.4 Normative References

- [Avails] Content Availability Metadata, TR-META-AVAIL, <http://www.movielabs.com/md/avails>
- [CM] Common Metadata, TR-META-CM, <http://www.movielabs.com/md/md>
- [CMM] Common Media Manifest Metadata, TR-META-MMM, <http://www.movielabs.com/md/manifest>
- [MEC] Media Entertainment Core, TR-META-MEC, , <http://www.movielabs.com/md/mec/>
- [EIDR] Entertainment Identifier Registry (EIDR), <http://eidr.org/resources/>
- [QCVocab] Quality Control (QC) Vocabulary, <http://www.movielabs.com/md/qcvocabulary>
- [TR-META-CR] *Common Metadata Content Ratings*. [www.movielabs.com/md/ratings](http://www.movielabs.com/md/ratings). Note that a specific version is not referenced as it is intended that the latest version will be used. Referencing specifications may selection a specific version of the referenced document.
- [TR-META-RS] Common Metadata Ratings Schema Definition, TR-META-RS, January 3, 2014, <http://www.movielabs.com/md/ratings/doc.html>
- [XML] “XML Schema Part 1: Structures”, Henry S. Thompson, David Beech, Murray Maloney, Noah Mendelsohn, W3C Recommendation 28 October 2004, <http://www.w3.org/TR/xmlschema-1/> and “XML Schema Part 2: Datatypes”, Paul Biron and Ashok Malhotra, W3C Recommendation 28 October 2004, <http://www.w3.org/TR/xmlschema-2/>

## 1.5 Informative References

## 1.6 Best Practices for Maximum Compatibility

Metadata typically evolves with the addition of new elements, attributes and vocabularies. Existing applications should be capable of accepting metadata, even though there might be more data than expected. Strict XML validation precludes an orderly evolution and can be counterproductive to the flexibility needed in real implementations.

Metadata specifications and schema updates are designed to support backwards compatibility. For example, element and attributes can be added, but required elements are not removed; or more generally ordinality of elements and attributes can be widened but not narrowed. Values are not changed in either syntax or semantics. Therefore, we strongly encourage implementations to either be diligent in tracking to the latest version, or follow the backwards compatibility rules provided here.

An XML document is considered compatible if its structure does not preclude the extraction of data from the document. For example, a document with additional elements and attributes do not preclude schema parsing and data extraction.

- 
- Do not reject compatible XML documents, unless they fail schema validation against the definition for an exact version/namespace match.
  - Extract data from compatible XML documents whenever possible
  - It is allowable to ignore elements and attributes whose presence is not allowed in the specification and schema versions against which the implementation was built. For example, if the original schema allows one instance and three instances are found, the 2nd and 3rd instance may be ignored.

We will try to update metadata definitions such that following these rules work consistently over time. Sometimes, changes must be made that are not always backwards compatible, so we will do our best to note these.

## 2 GENERAL TYPES ENCODING

### 2.1 Attribute Groups

#### 2.1.1 RangeAttributes

Element	Attribute	Definition	Value	Card.
	<b>RangeAttributes-attr</b>			
	rangeCondition	Range Condition. See below.	xs:string	0..1
	rangeRank	Relative ranking within equal rangeCondition, or if rangeCondition is unspecified. 0 is highest rank.	xs:nonNegativeInteger	0..1

RangeCondition defines the range of acceptable technical parameters. RangeCondition is an xs:string and typically an attribute (@rangeCondition).

When values are expressed,

Acceptable values for @rangeCondition are as follows

- ‘min’ – Represents minimum requirement. If numeric, lower values are not accepted.
- ‘max’ – Represents the maximum acceptable value. If numeric, higher values are not accepted.
- ‘preferred’ – Represents preferred condition or value.
- ‘acceptable’ – Represents a condition or value that is acceptable but not desired. There may be negative consequences of using this condition, such as lower quality.

### 2.2 Simple Types

Currently, there are no Simple Types in this schema.

### 2.3 Message and Terms Types

#### 2.3.1 DeliverySource-type and DeliveryReverseSource-type

DeliverySource-type and DeliveryReverseSource-type provides information about who provided the request, the former from studio to retailer, the latter from retailer to studio (reverse channel). This can include both the Service Provider who generated the document as well as the studio/retailer(s) for whom the document was prepared. This construct is useful to avoid ambiguity when requests come from service providers.

If requests are made on behalf of multiple retailers or storefronts, multiple Retail instances can be included.

Element	Attribute	Definition	Value	Card.
<b>DeliverySource-type</b>				
ServiceProvider		Service Provider delivering document	md:OrgName-type	0..1
Publisher		Publisher for whom the document was created	md:OrgName-type	0..n
DeliveryContact		Contact information for this document, typically from a Service Provider.	md:ContactInfo-type	0..1

Element	Attribute	Definition	Value	Card.
<b>DeliveryReverseSource-type</b>				
ServiceProvider		Service Provider delivering document	md:OrgName-type	0..1
Retailer		Retailer for whom the document was created	md:OrgName-type	0..n
DeliveryContact		Contact information for this document, typically from a Service Provider.	md:ContactInfo-type	0..1

### 2.3.2 ~~DeliveryTerms-type~~

~~Terms allows arbitrary terms to be specified.~~

~~The precise interpretation is subject to the mutual agreement of parties involved, although guidance is provided within.~~

~~Each term is a name/value pair with the name expressed as termName and the value expressed as one of Money, Event, Duration or text depending on the data contained within the term. If data cannot be otherwise expressed, the any##other element can be used.~~

~~Note that this object is based on Avails Terms. Syntax and semantics are intended to be identical. However, to avoid the need to reference the Avails schema, this complex type is repeated here.~~

Element	Attribute	Definition	Value	Card
DeliveryTerms-type				
	termName	Identifies the term. Enumeration is below. termName is case insensitive (i.e., case shall be ignored).	xs:string	
Money		Used when termName refers to a term expressed in terms of money.	md:Money-type	(choice)
Event		Used when termName refers to a term expressed in terms of a date, or date and time. See Section <b>Error! Reference source not found.</b>	xs:union(xs:date, xs:dateTime)	
Duration		Used when termName refers to a term expressed in terms of a time duration.	xs:duration	
Text		Used when a term can be expressed in text and it is not one of the other term types.	xs:string	
Boolean		Used when term can be expressed as True or False	xs:boolean	
URI		Used for URIs, including identifiers.	xs:anyURI	
Language		Used for language.	xs:language	
ID		Any identifier	md:id-type	
YearDateTime		Year, date or date+time. For time-only use Time.	md:YearDateOrTime	
Time		Time. May include time zone.	xs:time	
Region		Geographic area	md:Region-type	
<any>		Any other element. Used when a term cannot practically be expressed with one of the other element choices.	any ##other	

The Term specified is indicated by termName with the following conditions. Only one instance of each term may be included unless otherwise specified.

Following is a Terms template. Values may be filled in here, or in Best Practices.

termName	Interpretation	Element used
<td>	<td>	<td>

### 2.3.32.3.2 DeliveryHandling-type

Element	Attribute	Definition	Value	Card.
<b>DeliveryHandling-type</b>				
Comments		Any comments. Should be included if ExceptionFlag='true'	xs:string	0..1
ExceptionFlag		Indicates message requires human attention	xs:boolean	0..1
ResponseDate		Expected response date	xs:date	0..1
	datelsTarget	If 'true' indicates ResponseDate is not a hard deadline. Details determined bilaterally.	xs:boolean	0..1

### 2.3.42.3.3 DeliveryInstructions-type

DeliveryInstructions-type extends DeliveryHandling-type to include OrderID. This is for cases where an order applies. Note that not all uses of DeliveryHandling-type apply to an Order (e.g., Avails-related requests).

Element	Attribute	Definition	Value	Card.
<b>DeliveryInstructions-type</b>			delivery:DeliveryHandling-type	
OrderID		Order identifier	md:id-type	0..1

### 2.3.52.3.4 DeliveryParams-type

DeliveryParams-type includes delivery parameters that are common across media types, metadata, promotional, supplemental and other materials.

Element	Attribute	Definition	Value	Card.
<b>DeliveryParams-type</b>				
LeadTime		Lead time for deliverables relative to window start date. Negative	xs:duration	0..1

		values represent time before window.		
	durationIsTarget	If 'true' LeadTime is a target; that is, not a fixed duration	xs:boolean	0..1
Priority		Priority of request. Lower number is higher priority, with 0 being the highest.	xs:nonNegativeInteger	0..1
AdditionalInstructions		Any additional instructions	xs:string	0..1
Terms		Any additional terms	<a href="#">delivery:DeliveryTerms</a> <a href="#">md:Terms</a> -type	0..n

LeadTime is expressed as a negative duration for deliverables that occur prior to the window (the typical case).

durationIsTarget indicate that LeadTime are aspirational. The degree to which this must be honored is subject to bilateral service level agreements.

Priority is specified relative to a given DueDate. Note that processing of Priority will require Best Practices that define factors to optimize when prioritizing deliveries of different types across different dates (i.e., factoring in urgency versus importance).

### 2.3.5 DeliveryIdentification-type

Delivery Identification allows an object such as status or QC report to refer to the relevant set of deliverables. For example, if the delivery is associated with a particular Avail in France, one would use the ALID and Region to refer to the delivery. If the data is specific to a language or format profile, the Language and FormatProfile elements can be used. TransactionID (same as AvailID in Excel) is an efficient means of referring to a specific Avail over (Transaction element in XML, or row in Excel).

<u>Element</u>	<u>Attribute</u>	<u>Definition</u>	<u>Value</u>	<u>Card.</u>
<a href="#">DeliveryParams-type</a>				
<a href="#">ALID</a>		<a href="#">ALID</a>	<a href="#">md:id-type</a>	<a href="#">0..1</a>
<a href="#">EIDRURN</a>		<a href="#">EIDR in URN format</a>	<a href="#">xs:anyURI</a>	<a href="#">0..1</a>
<a href="#">AltID</a>		<a href="#">Alternate ID from Avail</a>	<a href="#">md:id-type</a>	<a href="#">0..n</a>
<a href="#">Region</a>		<a href="#">Region(s)</a>	<a href="#">md:Region-type</a>	<a href="#">1..n</a>
<a href="#">ExcludedRegion</a>		<a href="#">Excluded Region(s)</a>	<a href="#">md:Region-type</a>	<a href="#">0..n</a>

<u>Language</u>		<u>Language</u>	<u>xs:language</u>	<u>0..n</u>
	<u>asset</u>	<u>Corresponds with LocalizationOffering in Avails [Avails], Section 2.2.2.1 (i.e., 'sub', 'dub', 'subdub', 'any')</u>	<u>xs:string</u>	<u>0..1</u>
<u>FormatProfile</u>		<u>Format Profile as defined in Avails [Avails], Section 2.2.3</u>	<u>xs:string</u>	<u>0..n</u>
	<u>HDR</u>		<u>xs:string</u>	
	<u>WCG</u>		<u>xs:string</u>	
	<u>HFR</u>		<u>xs:string</u>	
	<u>NGAudio</u>		<u>xs:string</u>	
<u>TransactionID</u>		<u>Transaction ID from Avail</u>	<u>md:id-type</u>	<u>0..n</u>

## 2.4 Types that reference objects directly

### 2.4.1 DeliveryObjectReference-type

Element	Attribute	Definition	Value	Card.
<b>DeliveryObjectReference-type</b>				0..1
TrackReference		TrackReference per [Manifest], Section 2.2.3	xs:string	0..n
TrackIdentifier		TrackIdentifier per [Manifest], Section 2.2.3	md:ContentIdentifier-type	0..n
EIDRURN		EIDR identifier along with structural type	delivery:EIDRURN	0..n
TrackID		Reference track identifiers as per [Manifest]	delivery:DeliveryTrackID	0..n
FileInfo		Reference to a file	manifest:FileInfo-type	0..n
Container		Reference to a container	Manifest:ContainerInfo-typeReference-type	0..n

IMFRef		Reference to information in an Interoperable Master Format (IMF) file.	Delivery:DeliveryIMF-type	0..n
OtherIdentifier		Any other applicable identifier	md:ContentIdentifier-type	0..n

### 2.4.1.1 DeliveryTrackID-type

Allows tracks to be referenced

Element	Attribute	Definition	Value	Card.
<b>DeliveryTrackID-type</b>				
AudioTrackID		Audio track ID	manifest:AudioTrackID-type	(choice)
VideoTrackID		Video track ID	manifest:VideoTrackID-type	
SubtitleTrackID		SubtitleTrack ID	manifest:SubtitleTrackID-type	
ImageID		Image ID	manifest:ImageTrackID-type	
InteractiveTrackID		Interactive object (e.g., app) ID	manifest:InteractiveTrackID-type	
ContentID		Content ID	md:ContentID-type	
AncillaryTrackID		Ancillary track ID	manifest:AncillaryTrackID-type	
TextObjectID		Text object ID	manifest:TextObjectTrackID-type	

### 2.4.1.2 DeliveryIMFRef-type

References UUIDs for IMF CPLs, OPLs and virtual tracks.

Element	Attribute	Definition	Value	Card.
<b>DeliveryIMFRef-type</b>			extension of manifest:PresentationIMFRef-type	

NOTE: This object may need to be extended to reference other components of an IMF, particularly individual files. This specificity might be needed to more granularly request components or to report errors with more specificity.

## 2.5 Technical Characteristics

The TechCharacteristics-type defines a set of technical characteristics that can be used to define content characteristics or to refer to content by its characteristics.

Element	Attribute	Definition	Value	Card.
<b>TechCharacteristics-type</b>				
Audio		Parameters than define acceptable audio media delivery.	delivery:TechAudio-type	0..n
Video		Parameters than define acceptable video media delivery.	delivery:TechVideo-type	0..n
Subtitle		Parameters than define acceptable timed text media delivery.	delivery:TechSubtitle-type	0..n
Card		Parameters than define acceptable cards	delivery:TechCard-type	0..n
Image		Parameters that define acceptable image delivery, including artwork	delivery:TechImage-type	0..n
Metadata		Parameters than define acceptable metadata delivery.	delivery:TechMetadata-type	0..n
Container		Parameters than define acceptable containers.	delivery:TechContainer-type	0..n

### 2.5.1 Interpretation of terms within Technical Attributes

Technical Attributes contain parameters that correspond with technical characteristics of media files. Most of these correspond with technical values in Common Metadata [CM]. The full definitions are found in the referenced sections of Common Metadata.

When Type includes the note “Incl. @rangeCondition” then the type as defined in [CM] is extended to include an @rangeCondition attribute.

When a term is absent, there are no constraints. For example, if MaxFileSize is not specified, there are no limits on size. If Compliance is absent, there are no additional Compliance constraints. Generally speaking, only constrained parameters should be included. This makes the profile shorter and less complicated.

When a term is present, interpretation depends on the value of @rangeCondition.

When @rangeCondition is ‘preferred’, that is a suggestion, not a hard requirement.

For numeric values, elements values can be provided with @rangeCondition of ‘min’ and/or ‘max’. Values are inclusive. It is allowed to specify either or both of ‘min’ or ‘max’. With both are specified, media characteristic must fall within that limit (inclusive). When only a ‘min’ value is included, there is a fixed minimum but no maximum. With only a ‘max’ value is provided there is a fixed maximum with no minimum. There can be at most one ‘min’ value and one ‘max’ value.

For numeric values, any value with @rangeCondition of ‘preferred’ must be <= a ‘max’ value and >= a ‘min’ value. ‘min’ values must be <= ‘max’ values. At most one ‘min’ and one ‘max’ may be included. There is no limit on ‘preferred’ values. For example, 48kHz and 44.1kHz may both be ‘preferred’ values.

Non-numeric values may not have @rangeCondition = ‘max’ or ‘min’. This might be tempting for values such coded profiles, but it can sometimes be ambiguous.

## 2.5.2 TechAudio-type

References to Common Metadata types in this section refer to object in DigitalAssetImageData-type, as defined in [CM] section 5.2.3, with the same name.

Element	Attribute	Definition	Value	Card.
<b>TechAudio-type</b>				
	audioTechProfileName	Unique name of technical profile. If there is only one profile of this type and @default='true', this need not be included.	md:id-type	0..1
	Default	This profile is the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
	<u>purpose</u>	<u>Purpose of audio</u>	<u>xs:string</u>	<u>0..1</u>
Codec		As defined in [CM]	Incl. RangeAttributes	0..1
CodecType		As defined in [CM]	Incl. RangeAttributes	0..n
BitrateMax		As defined in [CM]	Incl. RangeAttributes	0..n
VBR		As defined in [CM].	Incl. RangeAttributes	0..1

SampleRate		As defined in [CM]	Incl. RangeAttributes	0..n
SampleBitDepth		As defined in [CM]	Incl. RangeAttributes	0..n
Channels		As defined in [CM]	Incl. RangeAttributes	0..n
ChannelMapping		As defined in [CM]	Incl. RangeAttributes	0..n
Compliance		As defined in [CM]	Incl. RangeAttributes	0..n
Loudness		As defined in [CM]	Incl. RangeAttributes	0..1
MaxFileSize		Maximum file size in bytes for file of this type	xs:nonNegativeInteger	0..1
<u>Language</u>		<u>Audio language. This does not apply to Content Delivery Requirements.</u>	<u>Md:DigitalAssetAudioLanguage-type</u>	<u>0..n</u>
Term		Additional terms that apply to this Profile	<u>delivery:DeliveryTermsmd:Terms-type</u>	0..n

### 2.5.3 TechVideo-type

References to Common Metadata types in this section refer to object in DigitalAssetVideoData-type, as defined in [CM] section 5.2.4, with the same name.

Element	Attribute	Definition	Value	Card.
<b>TechVideo-type</b>				
	videoTechProfile Name	Unique name of technical profile. If there is only one profile of this type and @default='true', this need not be included.	md:id-type	0..1
	<u>Defaultdefault</u>	Is this the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
	<u>purpose</u>	<u>Purpose of video</u>	<u>xs:string</u>	<u>0..1</u>
FrameCharacteristics		Frame constraints	delivery:TechVideoFrame-type	0..1
ColorCharacteristics		Color constraints	delivery:TechVideoColor-type	0..1

NextGenCharacteristics		Next Gen (i.e., HDR) characteristics	delivery:TechVideoNextGen-type	0..1
Type3D		As defined in [CM]	xs:string	0..1
MasterText		Defines the text allowed in the master	xs:string	0..1
	titles	Title text allowed. 'true' means allowed	xs:boolean	0..1
	credits	Credit text allowed. 'true' means allowed	xs:boolean	0..1
	scene	Scene setting text allowed, 'true' means allowed	xs:boolean	0..1
	forced	Force narrative text allowed. 'true' means allowed	xs:boolean	0..1
	textlessElements	Textless elements (i.e., video without text) provided in conjunction with texted video. 'true' means provided	xs:boolean	0..1
DiscreteCards		Indicates cards are delivered separately from video. If only certain cards are provided discretely, attributes indicate which ones are discrete. If cards are not discrete, they are appended to video and are part of the timeline.	xs:boolean	0..1
	dub	Dub cards are discrete	xs:boolean	0..1
	rating	Rating cards are discrete	xs:boolean	0..1
	territory	Territory-specific cards, such as anti-piracy and health cards, are discrete	xs:boolean	0..1
Compliance		As defined in [CM]	Incl. RangeAttributes	0..n
MaxFileSize		Maximum file size in bytes for file of this type	xs:nonNegativeInteger	0..1
Term		Additional terms that apply to this Profile	<a href="#">delivery:DeliveryTerms</a> <a href="#">md:Terms</a> -type	0..n

There are many definitions of terms like “semi-textless” based on what texted elements are allowed. The intent is to define what text elements are allowed in the video or need to be removed. For this purpose, we define text in terms of the following

- Titles – opening and closing
- Credits – opening and closing credits

- Scene Setting – Scene setting text such as location or time
- Forced narrative – Forced subtitles
- Photographic – Any text captured in a scene during production, such as billboards and street signs. Does not include VFX or animation-produced text. Production text is assumed to be part of the video, and is not considered in the context of texted or textless masters.

MasterText is encoded as follows. Note that most profiles prefer texted and/or semi-textless masters.

MasterText	Titles	Credits	Scene Setting	Forced Narrative	Additional
'Texted'	Allowed	Allowed	Allowed	Allowed	
'Semi-textless'	Allowed	Allowed	Allowed	Prohibited	
'Textless'	Prohibited	Prohibited	Prohibited	Prohibited	
'TextlessElements'	Allowed	Allowed	Allowed	Allowed	Textless elements are provided with texted master, typically appended
'Other'					Allowed text defined in attributes.

### 2.5.3.1 TechVideoFrame-type

References to Common Metadata types in this section refer to object in DigitalAssetVideoPicture-type, as defined in [CM] section 5.2.6, with the same name.

Element	Attribute	Definition	Value	Card.
<b>TechVideoFrame-type</b>				
Resolution		As defined in [CM]	Incl. RangeAttributes	0..n
AspectRatio		As defined in [CM]	Incl. RangeAttributes	0..n
PixelAspect		As defined in [CM]	Incl. RangeAttributes	0..1
FrameRate		As defined in [CM]	Incl. RangeAttributes	0..n
Progressive		As defined in [CM]	Incl. RangeAttributes	0..n
LetterboxAccepted		Letterbox and Pillarbox video is accepted. If 'false', only active pixels should be provided.	xs:boolean	0..1

### 2.5.3.2 TechVideoColor-type

References to Common Metadata types in this section refer to object in DigitalAssetVideoPicture-type, as defined in [CM] section 5.2.6, with the same name.

Element	Attribute	Definition	Value	Card.
<b>TechVideoColor-type</b>				
Colorimetry		As defined in [CM]	Incl. RangeAttributes	0..n
ColorSubsampling		As defined in [CM]	Incl. RangeAttributes	0..n
MasteredColorVolume		As defined in [CM]	Incl. RangeAttributes	0..n
BitDepth		As defined in [CM]	Incl. RangeAttributes	0..n

### 2.5.3.3 TechVideoNextGen-type

References to Common Metadata types in this section refer to object in DigitalAssetVideoPicture-type, as defined in [CM] section 5.2.6, with the same name.

Element	Attribute	Definition	Value	Card.
<b>TechVideoNextGen-type</b>				
LightLevel		As defined in [CM]	Incl. RangeAttributes	0..n
ColorVolumeTransform		As defined in [CM]	Incl. RangeAttributes	0..n

### 2.5.4 **TechSubtitle-type**

References to Common Metadata types in this section refer to object in DigitalAssetSubtitleData-type, as defined in [CM] section 5.2.7, with the same name.

Element	Attribute	Definition	Value	Card.
<b>TechSubtitle-type</b>				
	subtitleTechProfileName	Unique name of technical profile. If there is only one profile of this type and @default='true', this need not be included.	md:id-type	0..1

	<u>Default</u> <del>default</del>	Is this the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
	<u>purpose</u>	<u>Purpose of timed text</u>	<u>xs:string</u>	<u>0..1</u>
Type		As defined in [CM]	Incl. RangeAttributes	1..n
Format		As defined in [CM]	Incl. RangeAttributes	0..n
FormatType		As defined in [CM]	Incl. RangeAttributes	0..n
Compliance		As defined in [CM]	Incl. RangeAttributes	0..n
MaxFileSize		Maximum file size in bytes for file of this type	xs:nonNegativeInteger	0..1
Term		Additional terms that apply to this Profile	<u>delivery:DeliveryTerms</u> <u>md:Terms</u> -type	0..n

## 2.5.5 TechCard-type

Technical description for card, such as dub cards.

Element	Attribute	Definition	Value	Card.
<b>TechCard-type</b>				
	cardTechProfile Name	Unique name of technical profile. If there is only one profile of this type and @default='true', this need not be included.	md:id-type	0..1
	default	Is this the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
	<u>purpose</u>	<u>Purpose of card</u>	<u>xs:string</u>	<u>0..1</u>
DiscreteCards		Indicates whether Discrete Cards are required	xs:boolean	0..1
	RangeAttribute s-attr	Range Attributes (See Section 2.1.1)	xs:string	0..1
MustMatchVideoEncoding		Indicates whether cards must match video encoding	xs:boolean	0..1

MustMatchVideoDynamicRange		Indicates whether cards must match video dynamic range. For example, if video is HDR, must the cards be HDR.	xs:boolean	0..1
Compliance		Required compliance certifications. Encoded per definition in [CM], Section 3.17	md:Compliance-type	0..1
MaxFileSize		Maximum file size in bytes for file of this type	xs:nonNegativeInteger	0..1
Term		Additional terms that apply to this Profile	<del>delivery:DeliveryTerms</del> md:Terms-type	0..n

### 2.5.6 DeliveryImage-type

This object defines image technical characteristics. A set of image characteristics is called an Image Profile.

References to Common Metadata types in this section refer to object in DigitalAssetImageData-type, as defined in [CM] section 5.2.8, with the same name. Pixels are assumed to be square.

The image profile may be given a name in @imageProfileName. If this name is absent, it is assumed that all images will conform to this profile. Otherwise, artwork definitions must reference a named profile.

Element	Attribute	Definition	Value	Card.
<b>DeliveryImage-type</b>		Base type for this element is standard delivery parameters defined in DeliveryParams-type.	delivery:DeliveryParams-type (by extension)	
	imageTechProfileName	Unique name of technical profile. If there is only one profile of this type and @default='true', this need not be included.	md:id-type	0..1
	default	Is this the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
	<u>purpose</u>	<u>Purpose of image</u>	<u>xs:string</u>	<u>0..1</u>
Encoding		As per Common Metadata definition. One for each acceptable encoding method.	xs:string	0..n
AlphaAllowed		Is alpha channel supported (i.e., transparency). 'true' means yes. This must be absent or 'false' for encoding types that do not support alpha.	xs:boolean	0..1

DynamicRangeProfile		As defined in [CM]	xs:string	0..1
ColorGamutProfile		As defined in [CM]	xs:string	0..1
Compliance		As defined in [CM]	md:Compliance-type	0..1
MaxFileSize		Maximum file size in bytes for file of this type	xs:nonNegativeInteger	0..1
Term		Additional terms that apply to this Profile	<a href="#">delivery:DeliveryTerms</a> <a href="#">md:Terms</a> -type	0..n

### 2.5.7 TechMetadata-type

Element	Attribute	Definition	Value	Card.
<b>TechContainer-type</b>		Base type for this element is standard delivery parameters defined in DeliveryParams-type.	delivery:DeliveryParameters-type (by extension)	
	metadataTechProfileName	Unique name of technical profile. If there is only one profile of this type and @default='true', this need not be included.	md:id-type	0..1
	default	Is this the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
	<a href="#">purpose</a>	<a href="#">Purpose of metadata</a>	<a href="#">xs:string</a>	<a href="#">0..1</a>
Encoding		What is the metadata schema.	xs:string	1..n
	minVersion	Minimum version	xs:string	0..1
	maxVersion	Maximum version		
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1
MaxFileSize		Maximum file size in bytes for file of this type	xs:nonNegativeInteger	0..1
Term		Additional terms that apply to this Profile	<a href="#">delivery:DeliveryTerms</a> <a href="#">md:Terms</a> -type	0..n

Encoding is can be encoded with any value recognized by the recipient. However, Media Entertainment Core must be encoded as 'MEC'. minVersion and maxVersion indicate the version of that metadata type. For example, if any version of MEC 2.5 and beyond is acceptable, minVersion should be '2.5'.

### 2.5.8 TechContainer-type

References to Common Metadata types in this section refer to object in ContainerMetadataData-type, as defined in [CM] section 6.2, with the same name.

Element	Attribute	Definition	Value	Card.
<b>TechContainer-type</b>		Base type for this element is standard delivery parameters defined in DeliveryParams-type.	delivery:DeliveryParams-type (by extension)	
	containerTechProfileName	Unique name of technical profile. If there is only one profile of this type and @default='true', this need not be included.	md:id-type	0..1
	default	Is this the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
	<u>purpose</u>	<u>Purpose of container</u>	<u>xs:string</u>	<u>0..1</u>
ContainerType		As defined in [CM]	Incl. RangeAttributes	0..n
Compliance		As defined in [CM]	md:Compliance-type	0..1
MaxFileSize		Maximum file size in bytes for file of this type	xs:nonNegativeInteger	0..1
Term		Additional terms that apply to this Profile	<u>delivery:DeliveryTerms</u> <u>md:Terms-type</u>	0..n

## 3 CONTENT DELIVERY REQUIREMENTS

### 3.1 Requirements Structure

There are two parts to defining requirements: Scope (where the requirements apply) and Profiles (structured requirements).

#### 3.1.1 Scope

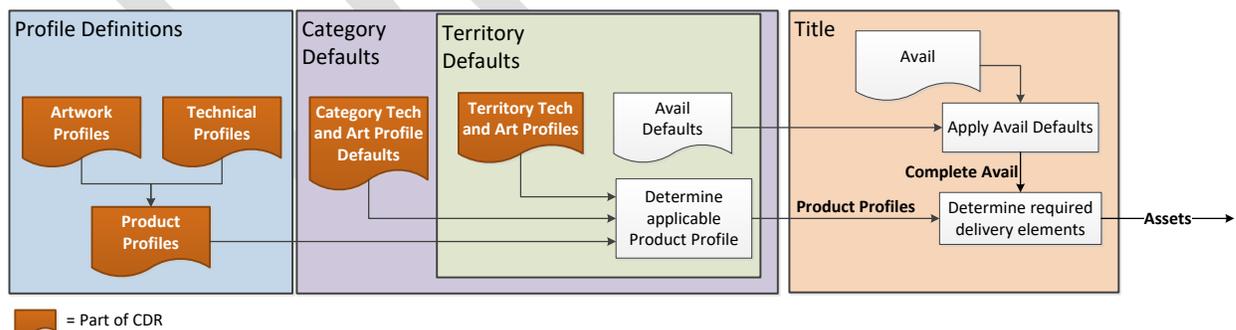
Scope defines where and when Profiles apply. Scope further divides into Territory and Category (TV, movies, etc.).

Territory is pretty straightforward. If the scope is worldwide, requirements apply everywhere, except where territory requirements are specified. This is an object model, where territories inherit the properties of the world, except where exceptions exist. There are specific rules that dictate what is inherited and what is not. {TBD}

Category defines what type of content, storefront, license model or other contextual parameter determines what rules apply. Like Region, Category is an object model where specifics inherit from their parent. For example, there could be a Category for TV and subcategory for Next-Day TV. Next-Day TV inherits most of its requirements (e.g., required artwork) from TV, but has different delivery timeframes.

Although inheritance can, at first, be daunting this is very much how people refer to content delivery requirements on paper.

This model is illustrated in the following figure. Profile Definitions define the Profiles for application within categories and territories. Category Defaults are the default profiles for the category. Territory Defaults are the defaults for profiles, *within* the Category. External to CDR are default Avail values (e.g., what languages are licensed by default within a territory). These Avail Defaults can be combined with an Avail for a Complete Avail (i.e., all the blanks filled in). Finally, all this information is combined to determine which assets apply.



Not shown in this illustration are *Administrative Profiles*.

Note that an earlier version of this specification included Language Profiles along with the others. It was determined that these are better handled in Avail Defaults. However, use cases might be discovered that will be best served by the inclusion of Language Profiles.

---

### 3.1.2 Profiles

A Profile describes requirements for some specific delivery. It takes several Profiles to fully describe a delivery.

Consider artwork for TV. It requires a collection of images with a particular aspect ratio and resolution; each with its own ‘purpose’. These are called Artwork Profiles. However, each image must comply with technical requirements such as encoding (JPEG, GIF, PNG), color encoding, maximum file size, and so forth. As all artwork images comply with a relatively small number of image specs, we have we have Image Profiles. Artwork Profiles simply refer to the applicable Image Profile.

Profiles come in the following categories

- Admin Profiles – Administrative rules such as lead times
- Language Profiles – Rules about localization, subs and dubs, and other language requirements as they apply to a territory
- Artwork Profiles – Sets of artwork, including resolutions, purpose, etc. [safe area?]
- Product Profiles – Definition of product-related deliverables, such as features, trailers, artwork, and bonus
- Technical Profiles – Audio, video, image, subtitle, and other digital asset technical descriptions

#### 3.1.2.1 Product Profiles

A Product Profile defines requirements for Feature (main feature), Promotional (ads, such as trailers) and Supplemental (bonus/extras/VAM). Each of these can have their own content requirements covering technical requirements, artwork, metadata and parameters specific to the type.

One would generally expect to have distinct Product Profiles for movies and TV. One could additionally have Product Profiles for deep catalog or tentpole titles. For example, deep catalog might have relaxed technical requirements. Tentpole titles might have additional expectations on artwork, trailers (Promotional) or bonus (Supplemental).

#### 3.1.2.2 Admin Profiles

Admin profiles address logistics issues such as lead time and priority. This sets general rules about delivery.

#### 3.1.2.3 Language Profiles

Language Profiles describe localization, including what artwork, metadata, audio, localized video, and other materials must be provided.

Language Profile is designed to provide defaults for information that would be found in EMA Avails [Avails]. Information in the Language Profile can be mapped directly to AllowedLanguages, AssetLanguage, LocalizationType, and RequiredFulfillmentLanguages.

### 3.1.2.4 Artwork Profiles

Each retail user interface has its own artwork requirements. Typically, there is a set of images for any given application. For example, movies might require 0.73 aspect ratio key art, while TV requires square key art. However, there can be more specific requirements, such as artwork for premium movies versus artwork for deep catalog movies.

Artwork Profiles are created for each set of images, each with a specific purpose (e.g., “cover1” or “hero2”). Purposes can correspond with MEC’s LocalizedInfo/ArtReference/@purpose, so when artwork is delivered you know exactly what you’re getting.

Image encoding (e.g., GIF/JPG/PNG, color space, etc.) is distinct from the Artwork Profile.

### 3.1.2.5 Technical Profiles

The following Technical Profiles are provided

- Audio
- Video
- Subtitle
- Image
- Cards
- Metadata
- Container

## 3.1.3 Profile Examples

### 3.1.3.1 Technical Profile

The following illustrates potential Technical Profiles. These profiles are described rather than encoded in XML. Many details are omitted for brevity.

Following are example video profiles:

Profile Name	Codec	Aspect Ratio	Color Space	Primaries	Sub-sampling	Bit depth	Frame Rate
HD ProRes	ProRes HQ	4:3, 1.66:1, 16:9, 1.85:1, 2:1, 2.20:1, 2.35:1, 2.39:1, 2.40:1	BT.709	BT.709	4:2:2 or 4:2:2	8-bit or 10-bit	23.976p, 24p, 25i, 25p, 29.97i, 29.97p, 30i, 30p, 60i
HD MPEG2	MPEG-2 Main or High	4:3, 1.66:1, 16:9, 1.85:1, 2:1, 2.20:1, 2.35:1, 2.39:1, 2.40:1	BT.709	BT.709	4:2:2 or 4:2:2	8-bit or 10-bit	23.976p, 24p, 25i, 25p, 29.97i, 29.97p, 30i, 30p, 60i



# Asset Ordering, Delivery and Tracking **DRAFT**

Ref: TR-META-AOD  
Version: v1.0 **DRAFT**  
Date: June 25, 2019

HD AVC	H.264 High	4:3, 1.66:1, 16:9, 1.85:1, 2:1, 2.20:1, 2.35:1, 2.39:1, 2.40:1	BT.709	BT.709	4:2:2 or 4:2:2	8-bit or 10-bit	23.976p, 24p, 25i, 25p, 29.97i, 29.97p, 30i, 30p, 60i
UHD	ProRes	4:3, 1.66:1, 16:9, 1.85:1, 2:1, 2.20:1, 2.35:1, 2.39:1, 2.40:1	BT.709	BT.709	4:2:2	10-bit	23.976, 24, 25, 29.97, 30, 60
UHDHDR	ProRes 422 HQ	4:3, 1.66:1, 16.9, 1.85:1, 2:1, 2.20:1, 2.35:1, 2.39:1, 2.40:1	BT.2100	P3	4:2:2	10-bit	23.976, 24, 25, 29.97, 30, 60

Following are example audio profiles

Profile Name	Codec	Channel Layout	Sample Rate	Bit Depth	Min Bitrate
PCM	PCM	'Mono', 'Mono, Mono', 'L,R', 'L,R,C,LFE,LS,RS', 'L,R,C,LFE,LS,RS,LRS,RRS'	48KHz	16, 24	
MPL2-S	MPEG-2 Layer II	'Mono', 'Mono, Mono', 'L,R'	48KHz	16, 24	384K
MPL2-MC	MPEG-2 Layer II	'L,R,C,LFE,LS,RS', 'L,R,C,LFE,LS,RS,LRS,RRS'	48KHz	16, 24	912K
AC-3-S	AC-3	'Mono', 'Mono, Mono', 'L,R'	48KHz	16, 24	192K
AC-3-MC	AC-3	'L,R,C,LFE,LS,RS', 'L,R,C,LFE,LS,RS,LRS,RRS'	48KHz	16, 24	448K
AAC-S	AAC	'Mono', 'Mono, Mono', 'L,R'	48KHz	16, 24	448K
AAC-MC	AAC	'L,R,C,LFE,LS,RS', 'L,R,C,LFE,LS,RS,LRS,RRS'	48KHz	16, 24	960K
Atmos	EAC3-Atmos				

Given these Audio and Video Profiles, a Technical Profile might look like the following. It references the other profiles. Note that an actual package definition would also reference subtitles.

Profile Name	Container	Video Profiles	Audio Profiles
MOV-HD	MOV	HD MPEG2, HD AVC	PCM, MPL2-S, MP2-MC, AC-3-S, AC-3-MC, AAC-S, AAC-MC
MOV-UHD	MOV	UHD, UHDHDR	PCM, MPL2-S, MP2-MC, AC-3-S, AC-3-MC, AAC-S, AAC-MC
ProRes-HD	ProRes	HD ProRes	PCM, AC-3-S, AC-3-MC, AAC-S, AAC-MC, Atmos
ProRes-UHD	ProRes	UHD, UHDHDR	PCM, AC-3-S, AC-3-MC, AAC-S, AAC-MC, Atmos

[[CHS NOTE TO SELF: Can we define all this as a Container, then just reference Containers from a Package. OR, do we need to map individual objects into a container.

How about this:

- Technical Profiles are collected into a Container
- Product Profiles are collected into Packages
  - Profiles in Packages reference Containers (e.g., feature references a package, promotional references a package, etc.)
  - Profiles probably need to reference Artwork and Metadata
  - Package →
    - Feature/Promotional/Supplemental
    - {Metadata + Artwork + AV} + {Tech Metadata + Tech Image + Tech Container} [Can also include Metadata and Image in Container?] ]]

### 3.2 DeliveryRequirement-type

DeliveryRequirements-type is the root definition of a ContentDeliveryRequirements element.

Element	Attribute	Definition	Value	Card.
ContentDeliveryRequirements-type				
	updateNum, workflow, updateDeliveryType, versionDescription	Common set of workflow attributes (defined in Common Metadata)	md:Workflow-attr	
Compatibility		Spec compatibility	manifest:Compatibility-type	
Source		Source of CRD	delivery:DeliveryReverseSource-type	
Publisher		Content provider who will fulfill content in accordance with these content delivery rules	md:OrgName-type	0..1
CDRID		Identifier for set of content delivery rules	md:id-type	0..1
Description		Description of content delivery rules set.	xs:string	0..1

AdminProfile		Applicable Admin Profiles	delivery:DeliveryAdminProfile-type	0..n
Profiles		Applicable Product, Technical and Artwork Profiles	delivery:Profiles-type	0..1
CategoryRules		Rules by Category and then Territory	delivery:CategoryRules-type	0..1
Instructions		Handling instructions. Includes exception flag.	delivery:Instructions-type	0..1

### 3.3 CategoryRules-type

Category Rules define the rules for one or more categories (e.g., Movie, TV or Next Day TV), and within that Category Territory Rules.

The referenced Product Profile within this object defines the default Product Profile for all territories. This can be superseded by Product Profile references within a TerritoryRules object.

Element	Attribute	Definition	Value	Card.
<b>DeliveryCategoryRules-type</b>				
ContentCategory		Content Category for rules defined in this object.	xs:string	0..1
ContentSubCategory		Additional specificity of Content Category for rules defined in this object.	xs:string	0..n
ProductProfileID		Reference to applicable Product Profile	md:id-type	0..1
TerritoryRules		Territory rules	delivery:TerritoryRules-type	0..1
Term		Additional terms that apply to this category and sub-category.	delivery:DeliveryTerms md:Terms-type	0..n

ContentCategory and ContentSubCategory define the scope of the CategoryRules object. When ContentDeliveryRequirements are used in conjunction with EMA Avails, ContentCategory values should correspond with Avails WorkType values. That allows an unambiguous linkage to Avails. ContentSubCategory can include values of WorkTypeDetail, values of EMA Avails LicenseTypeDescription (e.g., “Next Day TV” or “POD”), or other values that define handling (e.g., “Priority” and “Library”).

### 3.3.1 TerritoryRules-type

Territory rules apply across all categories within the territory, except when covered in category rules—category rules take precedence.

Element	Attribute	Definition	Value	Card.
<b>DeliveryTerritoryRules-type</b>				
Region		Region and Excluded Region define the territories where rules apply. They are encoded in accordance with Media Manifest [Manifest] Region and ExcludedRegion.	md:Region-type	(choice) 1..n
ExcludedRegion			md:Region-type	
TerritoryProductProfileID		Reference to the Product Profile that applies to territory or territories defined by Region and ExcludedRegion	md:id-type	0..1
<a href="#">LanguageRules</a>		<a href="#">Rules for languages, including original language and localizations</a>	<a href="#">delivery:DeliveryLanguageRules-type</a>	<a href="#">0..1</a>
RatingRules		Rules specific to content (parental control) ratings	delivery:DeliveryRatingRules-type	0..1
Terms		Additional terms	<a href="#">delivery:DeliveryTerms</a> <a href="#">md:Terms-type</a>	0..1

### 3.3.2 ~~RatingRules-type~~

#### 3.3.2 RatingRulesLanguage Rules

DeliveryLanguageRules-types defines requirements for assets and metadata in particular languages. This is used both in delivery requirements and in delivery orders.

For each language, this element defines which original or localized assets are required.

Any special requirement can be expressed through Terms.

<u>Element</u>	<u>Attribute</u>	<u>Definition</u>	<u>Value</u>	<u>Card.</u>
<a href="#">DeliveryLanguageRules-type</a>				

<a href="#">Original</a>		<a href="#">Rules for original language/original version (OV)</a>	<a href="#">xs:language, delivery:DeliveryLanguageRules-attr attribute group</a>	
<a href="#">Localization</a>		<a href="#">Rules for localized languages</a>	<a href="#">xs:language, delivery:DeliveryLanguageRules-attr attribute group</a>	<a href="#">0..n</a>
<a href="#">Terms</a>		<a href="#">Additional terms</a>	<a href="#">md:Terms-type</a>	<a href="#">0..1</a>

### [3.3.2.1 DeliveryLanguageRules-attr](#)

The [Delivery Language Rules Attribute Group](#) defines common attributes within [DeliveryLanguageRules-type](#)

<a href="#">Attribute Group</a>	<a href="#">Attribute</a>	<a href="#">Definition</a>	<a href="#">Value</a>	<a href="#">Card.</a>
<a href="#">DeliveryLanguageRules-attr</a>				
	<a href="#">audio</a>	<a href="#">Audio in this language is required or desired.</a>	<a href="#">xs:string</a>	<a href="#">0..1</a>
	<a href="#">timedText</a>	<a href="#">Timed text localization requirements as specified below</a>	<a href="#">xs:string</a>	<a href="#">0..1</a>
	<a href="#">SDH</a>	<a href="#">SDH Timed text localization requirements as specified below</a>	<a href="#">xs:string</a>	<a href="#">0..1</a>
	<a href="#">descriptive</a>	<a href="#">Descriptive audio is required or desired. See encoding information below.</a>	<a href="#">xs:string</a>	<a href="#">0..1</a>
	<a href="#">signed</a>	<a href="#">Video with signing is required or desired. See encoding information below.</a>	<a href="#">xs:string</a>	<a href="#">0..1</a>
	<a href="#">metadata</a>	<a href="#">Localized metadata is required or desired. See encoding information below.</a>	<a href="#">xs:string</a>	<a href="#">0..1</a>

[@audio is encoded as follows:](#)

- ‘required’ – Localized audio is required. Can be delivered in any format as opposed to ‘premium’ where premium formats are required. Default for Original.
- ‘premium’ – Localized asset is required in premium format (i.e., multichannel or object-based audio).
- ‘preferred’ – Localized audio is desired. It is not a requirement for launch.

@timedText and @SDH are encoded as follows:

- ‘required’ – Timed text is required.
- ‘preferred’ – Timed text is desired. It is not a requirement for launch.
- ‘both’ – Both language and ‘SDH’ subtitles are required. Both @timedText and @SDH must be encoded ‘both’
- ‘either’ – Either language or ‘SDH’ subtitles are required. Both @timedText and @SDH must be encoded ‘either’

@descriptive and @signed is encoded as follows:

- ‘required’ – Localized asset is required.
- ‘preferred’ – Localized asset is desired. It is not a requirement for launch.

### 3.3.3 Rating Rules

DeliveryRatingRules-types defines requirements for delivery of ratings related to the content in question.

Element	Attribute	Definition	Value	Card.
<b>DeliveryRatingRules-type</b>				
RatingRequired		A rating is required for this territory	xs:boolean	0..1
MaxRating		Maximum allowable rating. Multiple entries can be provided to define maximum rating in multiple rating systems.	md:ContentRatingDetail-type	0..n
Terms		Additional terms	<u>delivery:DeliveryTerms</u> md:Terms-type	0..1

## 4 PROFILES

A Profile is a collection of requirements. Currently, we refer to

- Administrative Profile – Lead times, priorities, and special instructions
- Product Profiles – Set of Artwork Profiles and Technical Profiles that apply to product category/categories and territory/territories.
  - Artwork Profiles – Sets of artwork types, resolutions, aspect ratios, and other descriptors
  - Technical Profiles – Technical requirements about files tracks

Once defined, a Profile is used as shorthand for these requirements. For example, one might have a “Benelux” profile for language requirements for Benelux countries, and an “HDR” profile for minimum HDR requirements.

Profiles can be referenced both as requirements and as part of deliveries. That is, a Content Delivery Requirements (CDR) document might define an “HDR” profile, an MMC delivery might refer to the assets as fulfilling part of the “HDR” Profile; and, an Asset **Status Manifest Availability** might indicate the “HDR” Profile has not yet been delivered.

### 4.1 Administrative Profile

Element	Attribute	Definition	Value	Card.
DeliveryAdminProfile-type			Extension of delivery: DeliveryParams-type	
	AdminProfileID	ID for this profile	xs:string	

### 4.2 Product Profiles

Product Profiles are collections of Artwork and Technical Profiles. The Profiles-type complex type contains Product Profiles and their subordinate Artwork and Product Profiles. Only Product Profiles are referenced externally to the Profiles object.

#### 4.2.1 Profiles-type

Element	Attribute	Definition	Value	Card.
Profiles-type				
ProductProfile		Product Profile definition	delivery:ProductProfile-type	1..n

ArtworkProfiles		Artwork Profiles	Delivery:ArtworkProfiles-type	0..1
TechnicalProfiles		Technical Profiles	Delivery:TechnicalProfiles-type	0..1

## 4.2.2 ProductProfile-type

This type defines a single Product Profile

Element	Attribute	Definition	Value	Card.
<b>ProductProfile-type</b>				
	productProfileID	Unique identifier for this Product Profile	md:id-type	0..1
	Default	Indicates whether this the default profile. If 'true', it is. If absent or 'false' it is not default. At most one instance can be the default	xs:boolean	0..1
Feature		Feature characteristics	delivery:ProductProfileInfo-type	0..n
Promotional		Promotional material characteristics	delivery:ProductPromotional-type	0..n
Supplemental		Supplemental material characteristics	delivery:ProductSupplemental-type	0..n

## 4.2.3 ProductProfileInfo-type

This type is the base type for Product Profiles. It contains data that is in all Product Profiles.

Element	Attribute	Definition	Value	Card.
<b>ProductProfile-type</b>				
	purpose	Propose of profile	xs:string	0..n
TechProfileName		Name of Technical Profiles that apply to this Product Profile	xs:string	0..n
ArtworkProfileName		Name of Artwork Profiles that apply to this Product Profile	xs:string	0..n
LocalizedMetadata		Whether localized metadata required for this Profile. 'true' means yes.	xs:boolean	0..1

#### 4.2.4 ProductPromotional-type

Product Profile information for promotional material, such as trailers and teasers.

Element	Attribute	Definition	Value	Card.
ProductPromotional-type		Base type for this element is default Product Profile data	delivery:ProductProfileInfo-type (by extension)	
IncludesTrailer		Indicates whether trailer is expected. 'true' means trailer is expected.	xs:boolean	0..1
LimitedAudience		Indicates limited audience promotional material is allowed (e.g., Red Band trailers)	xs:boolean	0..1

#### 4.2.5 ProductSupplemental-type

Supplementary material is any audiovisual, gallery, game, app, or other content that supplements the feature. Also referred to as Bonus and VAM (value added material).

Element	Attribute	Definition	Value	Card.
ProductSupplemental-type		Base type for this element is default Product Profile data	delivery:ProductProfileInfo-type (by extension)	
LocalizedBonus		Indicates whether supplemental material is expected to be localized to the territory. 'true' means supplemental material should be localized.	xs:boolean	0..1
IncludesBonus		Indicates whether supplemental material is expected. 'true' means supplemental material is expected.	xs:boolean	0..1

### 4.3 Artwork Profiles

This type defines a profile images each of which constitute artwork serving a 'purpose'. Typically, that purpose, defined in @purpose, corresponds with [CM] LocalizedInfo/ArtReference/@purpose.

An instance is included for each combination of @purpose and @imageProfileName. If @imageProfileName is absent, the default Image Profile is used. If there is only one TechImage-type/ImageProfile, it is the default. ImageProfile/@default = 'true', it is the default.

### 4.3.1 ArtworkProfile-type

When multiple instances of Aspect or Resolution are provided, each of those is required. Aspect should not be included for the same image.

Element	Attribute	Definition	Value	Card.
<b>ArtworkProfile-type</b>		Base type for this element is standard delivery parameters defined in DeliveryParams-type.	delivery:DeliveryParams-type (by extension)	
	ArtworkProfileID	Image profile name corresponding with ImageProfile in DeliveryImage-type	md:id-type	0..1
Image		Image with a given purpose that is part of this profile	Delivery:ArtworkImage-type	1..n

### 4.3.2 ArtworkImage-type

Element	Attribute	Definition	Value	Card.	
<b>ArtworkImage-type</b>					
	purpose	Image purpose	xs:string	0..1	
	imageProfileName	Unique image name. Note that @purpose could appear in multiple profiles.	xs:string	0..1	
ImageAspectRatio		Aspect ratio represented as a decimal number representing the ratio between the x-axis and y-axis dimensions. Note this definition is distinct from [CM] Picture/AspectRatio which is a string.	xs:decimal	1..n	Choice
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1	
Resolution		Resolution of image (fixed or minimum)	delivery:ArtworkResolution-type	1..n	
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1	
TextLocalization		Text localization constraints on images	xs:string	0..n	

TextLocalization indicates options for image text. Encoding includes

- ‘textfree’ – text is not allowed on the image
- ‘localized’ – text is allowed, but must be localized. ‘textfree’ images also accepted.
- ‘original’ – original version
- ‘preferred’ – textfree, localize, or original available image is acceptable. Generally, in the order of preference is text free or localized, then original, and then other versions.
- ‘any’ – any image localization will do
- *[CHS: Are there other options? Is this complete? Should it be checkboxes?]*

#### 4.3.2.1 PictureResolution-type

Defines the resolution for an artwork image or video picture in pixels. If resolution specifies a minimum (i.e. @absolute = ‘false’ or is absent), aspect ratio of width and height is fixed. That is, they both must scale together to maintain aspect ratio.

Element	Attribute	Definition	Value	Card.
<b>PictureResolution-type</b>				
Width		Width in pixels	x:integer	0..1
Height		Height in pixels	x:integer	0..1

## 4.4 Technical Profiles

The Technical Profiles is a collection of audio, video, subtitle, dub card, image, metadata and container profiles. Each component profile is defined independently so it can be reused across Technical Profiles.

The TechnicalProfiles-type defines a set of Technical Profiles (TechProfile). It relies on TechnicalAttributes-type for the detailed component profiles (Audio, Video, etc.).

Element	Attribute	Definition	Value	Card.
<b>DeliveryTechnicalProfiles-type</b>			Delivery:TechnicalAttributes-type (by extension)	
TechProfile		A Technical Profile.	delivery:TechnicalProfile-type	1..n

#### 4.4.1 TechnicalProfile-type

TechProfile-type defines a single Technical Profile.

Technical Profiles are a collection of audio, video, subtitle, card, metadata, image, and container profiles. These other profiles are included by reference via their Profile names.

Each profile name can optionally include a RangeAttributes to indicate whether the referenced profile requirements are hard requirements or desired condition. Interpretation of Range Attributes is defined in Section 2.1.12.5.1.

Element	Attribute	Definition	Value	Card.
<b>DeliveryTechnicalProfile-type</b>				
	TechProfileID	Unique identifier for this Technical Profile	md:id-type	1..n
AudioTechProfileName		Name of Audio Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1
VideoTechProfileName		Name of Video Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1
SubtitleTechProfileName		Name of Subtitle Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1
CardTechProfileName		Name of Card Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1
MetadataTechProfileName		Name of Metadata Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttributes-attr	Range Attributes (See Section 2.1.1)		0..1

ImageTechProfileName		Name of Image Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttribute s-attr	Range Attributes (See Section 2.1.1)		0..1
MetadataTechProfileName		Name of Metadata Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttribute s-attr	Range Attributes (See Section 2.1.1)		0..1
ContainerTechProfileName		Name of Container Profile that applies to this Technical Profile	xs:string	0..n
	RangeAttribute s-attr	Range Attributes (See Section 2.1.1)		0..1

## 5 ASSET ORDER

An Asset Order defines objects to be delivered.

### 5.1 AssetOrder-type

Element	Attribute	Definition	Value	Card.
<b>AssetOrder-type</b>				
	updateNum, workflow, updateDeliveryType, versionDescription	Workflow attributes	md:Workflow-attr	0..1
DeliveryID			md:id-type	0..1
Description		Description of request	xs:string	0..1
Source		Source of this request	delivery:DeliveryReverseSource-type	
Publisher		Publisher that originated content (i.e., generated the Avail)	md:orgName-type	0..1
<a href="#">ALIDIdentification</a>		<a href="#">ALID of content information to associate the order with the offer associated with this delivery.</a>	md:delivery:DeliveryIdentification-type	0..1
Asset		Identifies assets and specifies terms specific to that asset	delivery:AssetOrderObject-type	0..n
TermsAcrossAssets		Specifies terms that apply to all assets identified in the Asset object	delivery:AssetOrderTerms-type	0..n
Instructions		Any other instructions	xs:string	0..1

#### 5.1.1 AssetOrderObject-type

AssetOrderObject-type specifies the object to be delivered, and possibly terms specific to that object.

Element	Attribute	Definition	Value	Card.
---------	-----------	------------	-------	-------

<b>AssetOrderObject-type</b>			Delivery:AssetOrderTerms-type (by extension)	
<u>Purpose</u>		<u>Purpose of asset order</u>	<u>xs:string</u>	
ObjectReference		Reference to objects, such as specific tracks, requested	delivery:DeliveryObjectReference-type	0..n
ObjectDescription		Reference to objects, such as tracks, by description (e.g., <i>French dub</i> ).	delivery:DeliveryObjectDescription-type	0..n
<u>LanguageOrder</u>		<u>Order information by language</u>	<u>delivery:DeliveryLanguageRules-type</u>	<u>0..1</u>

### 5.1.2 AssetOrderTerms-type

Element	Attribute	Definition	Value	Card.
<b>AssetOrderTerms-type</b>				
StatusCode		Code that indicates order status for the object	xs:string	
ExpectedDelivery		Expected delivery date	md:YearDateOrTime-type	0..1
BusinessTerms		Business terms, such as cost to generate or deliver asset	<u>delivery:DeliveryTermsmd:Terms-type</u>	0..n
TechnicalTerms		Additional technical terms relating to asset delivery	<u>delivery:DeliveryTermsmd:Terms-type</u>	0..n
Instructions		Any other instructions	xs:string	0..1

StatusCode indicates how the request should be handled. For example, it could be a request that assets be delivered, it could be a request of estimated delivery, or it could be a request to price the delivery of assets. **TBD**

## 6 ASSET STATUS MANIFEST AVAILABILITY (ASM)

The Asset Status Manifest Availability describes the status of asset delivery from the studio to the retailer. This can include assets in any stage of delivery. Some conditions include

- Assets that have been delivered (retailer perspective on that delivery notwithstanding)
- Assets that are being prepared
- Assets that could potentially be provided by request (perhaps with a fee)

Note that asset status information is sent in both directions the mirror image of this object is Delivery Asset Status Asset Availability-type sent from the retailer to the studio.

### 6.1 Asset Status Manifest Asset Availability-type

Element	Attribute	Definition	Value
<u>Asset Status Manifest Asset Availability</u> -type			
	updateNum, workflow, updateDeliveryType, versionDescription	Workflow attributes	md:Workflow-attr
DeliveryID			md:id-type
Description		Description of request	xs:string
Source		Source of this request	delivery:DeliveryReverseSource-type
Publisher		Publisher that originated content (i.e., generated the Avail)	md:orgName-type
<u>ALID Identification</u>		<u>ALID of content information to associate the order with the offer associated with this delivery.</u>	md:idDelivery:DeliveryIdentification-type

AssetDisposition		Status of asset or group of assets	delivery:AssetStatusObjectAssetAvailabilityObject-type
Instructions		Any other instructions	xs:string

### 6.1.1 AssetStatusObjectAssetAvailabilityObject-type

Element	Attribute	Definition	Value	Card.
<u>AssetStatusObjectAssetAvailabilityObject-type</u>				
ObjectReference		Reference to objects, such as specific tracks, requested	delivery:DeliveryObjectReference-type	1..n (choice)
ObjectDescription		Reference to objects, such as tracks, by description (e.g., <i>French dub</i> ).	delivery:DeliveryObjectDescription-type	
StatusCode		Code that indicates status of asset or assets identified in ObjectReference or ObjectDescription	xs:string	
ExpectedDelivery		Expected delivery date	md:YearDateOrTime-type	0..1
BusinessTerms		Business terms, such as cost to generate or deliver asset	delivery:DeliveryTermsmd:Terms-type	0..n
TechnicalTerms		Additional technical terms relating to asset delivery	delivery:DeliveryTermsmd:Terms-type	0..n

---

Instructions		Any other instructions	xs:string	0..1
--------------	--	------------------------	-----------	------

StatusCode indicates the status of the particular asset. Values include (**TBD**)

- ‘available’ – Asset is available, but has not been requested
- ‘processing’ – Asset is being processed for delivery
- ‘delivered’ – Asset has been delivered and considered completed unless recipient indicates otherwise
- ‘rework’ – Being reworked following an QC report
- ‘rejected’ – Asset has been requested, but will not be delivered
- ‘recalled’ – Asset has been delivered, but has a problem and should not be used

## 7 QC REPORT

A Quality Control (QC) report provides information on anomalies associated with deliveries. This report provides the means to identify issues media, metadata and other files.

In the simplest form, the QC Report can identify the object in question and the convey associated issue. The QC Report also supports additional data associated with particular media types. For example, timecode ranges can be conveyed for any audio, video and timed text.

For uniformity, errors are reported using the standardized QC Vocabulary found in [QCVocab].

### 7.1 QCReport-type

Element	Attribute	Definition	Value	Card.
QCError-type				
	updateNum, workflow, updateDeliveryType, versionDescription	Common set of workflow attributes (defined in Common Metadata)	md:Workflow-attr	
Compatibility		Spec compatibility	manifest:Compatibility-type	
Source		Source of this object	delivery:DeliveryReverse Source-type	
Publisher		Publisher to whom the status is being sent	md:OrgName-type	0..1
<a href="#">Identification</a>		<a href="#">Information to associate the order with the offer associated with this delivery.</a>	<a href="#">Delivery:DeliveryIdentification-type</a>	
Description		Description of status (overview)	xs:string	0..1
QCError		Error report	delivery:QCError-type	1..n
Instructions		Handling instructions. Includes exception flag.	delivery:Instructions-type	0..1

## 7.1.1 QCError-type

QCError-type says one or more errors apply to one or more assets. The assets might be implicit (i.e., do not need to be explicitly stated in this object).

Element	Attribute	Definition	Value	Card.
<b>QCError-type</b>				
ErrorDescription		Description of the issue with the media and/or file	delivery:QCErrorDescription-type	1..n
MediaAsset		Media Asset that is the subject of the error	delivery:DeliveyrObjectReference-type	0..n

## 7.1.2 QCErrorDescription-type

QCError-Description-type provide information about the error. ErrorCategory and ErrorTerm are from QC Vocabulary [QCVocab].

In the form of CategorySpecific details on the specific error can be provided. For example, in anything time-based, start and/or end timecode can be provided. In video pictures or images a bounding box of the problem area can be described.

In some cases, full QC was not performed on an asset. This can be indicated in FullOrPartialQC.

ErrorReference is included to provide a reference to this specific error report.

Element	Attribute	Definition	Value	Card.
<b>QCErrorDescription-type-type</b>				
ErrorReference		Reference tag that can be used to refer to this error instance elsewhere	xs:string	0..1
ErrorCategory		Error Category, in accordance with QC Nomenclature [QCVocab]	xs:string	
ErrorTerm		Error Term in accordance with QC Nomenclature [QCVocab]	xs:string	
CategorySpecific		Additional data associated with error, based on Error Category.	delivery:QCCategoryError-type	0..n
Comments		Any additional comments	xs:string	0..1

FullOrPartialQC		Indicates whether assets was fully evaluated or if evaluation stopped at first error(s)	xs:string	0..1
-----------------	--	---	-----------	------

FullOrPartialQC is encoded as follows [CHS: should this just be a boolean?]

- ‘Full’ – QC was completed
- ‘Partial’ – QC was aborted once error(s) were found. Additional errors may be present.

### 7.1.3 QCCategoryError-type

This section contains additional information for errors that are specific to the type of object with an error. Value depends on the QC Nomenclature Category of the error.

Note that definitions are specific to Error Categories (e.g., Video or Audio), and not to specific Error Terms. It is assumed context is sufficient to interpret term-specific data. If not, Best Practices should be developed and/or notes can be put in the Comments field of the parent object.

Element	Attribute	Definition	Value	Card.
<b>DeliveryCategoryError-type</b>				
Audio		Audio Category error specifics	delivery:QCErrorAudio-type	(choice)
Video		Video Category error specifics	delivery:QCErrorVideo-type	
TimedText		TimedText Category error specifics	delivery:QCTimedText-type	
Avail		Avail Category error specifics	delivery:QCAvail-type	
Metadata		Metadata Category error specifics	delivery:QCErrorMetadata-type	
Artwork		Artwork Category error specifics	delivery:QCErrorArtwork-type	
Package		Package Category error specifics	delivery:QCErrorPackage-type	

[NOTE: Additional Categories are being defined (e.g., “Film”). These will need to be captured here.]

#### 7.1.3.1 QC Utility types

##### 7.1.3.1.1 QCTimeRange-type

Element	Attribute	Definition	Value	Card.
---------	-----------	------------	-------	-------

QCTimerange-type				
StartTimecode		Track timeline where issue starts.	manifest:Timecode-type	
EndTimecode		Track timeline where issue ends. Omit, if problem persists to end of timeline or if end is unknown	manifest:Timecode-type	0..1

### 7.1.3.1.2 QCXMLError-type

Indicates where in an XML document the problem exists. XPath defines the object. Or, if preferred, a line number can reference the object.

Element	Attribute	Definition	Value	Card.
<b>QCXMLError-type</b>				
XPath		XPath reference to object with issue(s)	xs:anyURI	0..1
LineNumber		Line number in file of issue	xs:positiveInteger	0..1

### 7.1.3.1.3 QCArea-type

Area of image or picture area where problem exists.

If issue is a single pixel, Width and Height should be 1.

Element	Attribute	Definition	Value	Card.
<b>QCArea-type</b>				
XOffset		In pixels, x-value of lower left corner of issue.	xs:decimal	
YOffset		In pixels, y-value of lower left corner of issue.	xs:decimal	
Width		In pixels, width of picture, inclusive of pixel marked by XOffset.		
Height		In pixels, height of picture, inclusive of pixel marked by YOffset.		

### 7.1.3.2 QCErrorAudio-type

Element	Attribute	Definition	Value	Card.
<b>QCErrorAudio-type</b>				
TimeRange		Time range where problem exists. If problem is entire range, do not include this element.	delivery:QCTimeRange-type	0..1
TimeOffset		For errors with alignment issues (e.g., AV Sync), the duration of offset. Negative means audio is ahead of video.	xs:duration	0..1

### 7.1.3.3 QCErrorVideo-type

Element	Attribute	Definition	Value	Card.
<b>QCErrorVideo-type</b>				
TimeRange		Time range where problem exists. If problem is entire range, do not include this element.	delivery:QCTimeRange-type	0..1
Area		Area picture where problem exists	delivery:QCArea-type	0..1

### 7.1.3.4 QCTimedText-type

Element	Attribute	Definition	Value	Card.
<b>QCErrorSubtitle-type</b>				
TimeRange		Time range where problem exists. If problem is entire range, do not include this element.	delivery:QCTimeRange-type	0..1
TimeOffset		For errors with alignment issues (e.g., subtitle Sync), the duration of offset. Negative means subtitle is ahead of video.	xs:duration	0..1

Text		Text that is in error		0..1
------	--	-----------------------	--	------

#### 7.1.3.5 QCErrorAvail-type

Element	Attribute	Definition	Value	Card.
<b>QCErrorMetadata-type</b>				
XMLError		Reference to location of XML Error	delivery:QCXMLError-type	

#### 7.1.3.6 QCErrorMetadata-type

Element	Attribute	Definition	Value	Card.
<b>QCErrorMetadata-type</b>				
XMLError		Reference to location of XML Error	delivery:QCXMLError-type	

#### 7.1.3.7 QCErrorArtwork-type

Element	Attribute	Definition	Value	Card.
<b>QCErrorArtwork-type</b>				
Area		Area picture where problem exists	delivery:QCArea-type	0..1
Text		Text on image that is in error		0..1

#### 7.1.3.8 QCErrorPackage-type

**TBD**

Element	Attribute	Definition	Value	Card.
<b>QCErrorPackage-type</b>				
Subobject		Object with package with issue		1..n

## 8 PRODUCT STATUS

Product Status provides the means for communicating status at every stage of delivery, from Avails through readiness to deliver content to a consumer.

Whether the status is for an Avail/Title List or for assets, there are two parts: What is the object being reported, and what is the status of that object. Avails are referenced by ALIDs and TransactionIDs (or AvailIDs). Assets are referenced by media asset references or file references.

ProductStatus-type is the defines the ProductStatus element.

Element	Attribute	Definition	Value	Card.
<b>ProductStatus-type</b>				
	updateNum, workflow, updateDeliveryType, versionDescription	Common set of workflow attributes (defined in Common Metadata)	md:Workflow-attr	
<u>DeliveryID</u>		<u>ID associated with the delivery</u>	<u>md:id-type</u>	<u>0..1</u>
Compatibility		Spec compatibility	manifest:Compatibility-type	
Source		Source of this Avail Defaults object	delivery:DeliveryReverse Source-type	
Publisher		Publisher to whom the status is being sent	md:OrgName-type	0..1
Description		Description of status (overview)	xs:string	0..1
<u>AvailStatusIdentification</u>		<u>Status of the information to tie this status to an Avail, including readiness to or other offer to consumer</u>	<u>delivery:ProductAvailStatusDeliveryIdentification-type</u>	<u>0..n</u>
<u>OverallProgressCode</u>		<u>Overall status</u>	<u>xs:string</u>	<u>0..1</u>
	<u>media</u>	<u>Progress code for media</u>	<u>xs:string</u>	<u>0..1</u>
	<u>artwork</u>	<u>Progress code for artwork</u>	<u>xs:string</u>	<u>0..1</u>
	<u>metadata</u>	<u>Progress code for metadata</u>	<u>xs:string</u>	<u>0..1</u>
AssetStatus		Status of asset delivery, including readiness to deliver assets to consumer	delivery:ProductAssetStatus-type	0..n

Instructions		Handling instructions. Includes exception flag.	delivery:Instructions-type	0..1
--------------	--	---	----------------------------	------

## 8.1 Avail Status

ProductAvailStatus-type provides status on the processing of an Avail or Title List.

Element	Attribute	Definition	Value	Card
ProductAvailStatus-type				
ALID		ALID of Avail whose status is being reported	md:id-type	
Region		Region and ExcludedRegion define the territories where rules apply. They are encoded in accordance with Media Manifest [Manifest] Region and ExcludedRegion. If absent, rules apply to all territories not covered by other TerritoryDefault instances.	md:Region-type	(choice) 1..n
ExcludedRegion			md:Region-type	
TransactionID		TransactionID association with Avail Transaction/@TransactionID being reported. If referring to Excel Avail, this corresponds with AvailID.	md:id-type	0..n
ProgressCode		Code that identifies progress step (TBD)	xs:string	
ErrorDescription		Description of error associated with progress	delivery:QCErrorDescription-type	0..1
Comments		Any additional comments	xs:string	0..1
Log		<u>Event Log</u> of previous events	delivery:DeliveryLogEventProductLog-type	0..1
Instructions		Handling instructions. Includes exception flag.	delivery:Instructions-type	0..1

### 8.28.1 Asset Status

ProductAssetStatus-type provides status of asset delivery processing.

Element	Attribute	Definition	Value	Card.
<b>ProductAssetStatus-type</b>				
MediaAsset		Reference to Asset	delivery:DeliveryObjectReference-type	0..n
ProgressCode		Code that identifies progress step (TBD)	xs:string	
ErrorDescription		Description of error associated with progress	delivery:QCErrordescription-type	0..1
Comments		Any additional comments	xs:string	0..1
Log		Log of previous events	delivery:DeliveryLogEvent-type	0..1
Instructions		Handling instructions. Includes exception flag.	delivery:Instructions-type	0..1

### 8.1.1 Progress Codes

TBD

(ready, missing, broken, in process, approved, etc.)

### 8.38.2 Logs

A log provides a history of events.

#### 8.3.18.2.1 ProductLog-type

A log is an ordered sequence of events. Log should be ordered from earliest to latest events.

Element	Attribute	Definition	Value	Card.
<b>ProductLog-type</b>				
Event		A reportable event	delivery:ProductLogEvent-type	1..n

#### 8.3.28.2.2 ProductLogEvent-type

Element	Attribute	Definition	Value	Card.
---------	-----------	------------	-------	-------

<b>ProductLogEvent-type</b>				
EventType		Type of event. Can be a Progress Code.	xs:string	
Timestamp		Time of event. Should be date or date plus time.	md:YearDateOrTime-type	
Description		Description of event	xs:string	0..1
ErrorReference		Reference to a specific error instance as defined in Progress/ErrorDescription	delivery:ProductAvailStat us-type	0..n

[CHS: Need to enumerate event types.]

---

## 9 NOTES

Make sure these are addressed:

- Indication that delivered content isn't to spec (kind of a waiver).
- Need an indication of what is missing. For example, is forced dubs required for video.
- Ordering something special (e.g., special trailers or artwork)
- Capacity planning and delivery timing?
- Flows
  - Standard delivery flow
  - Exception flows