

OpenCable™ Specifications

OpenCable Content Definition Format (CoDF)

OC-SP-CoDF-I02-110325

ISSUED

Notice

This OpenCable specification is the result of a cooperative effort undertaken at the direction of Cable Television Laboratories, Inc., for the benefit of the cable industry and its customers. This document may contain references to other documents not owned or controlled by CableLabs. Use and understanding of this document may require access to such other documents. Designing, manufacturing, distributing, using, selling, or servicing products, or providing services, based on this document may require intellectual property licenses from third parties for technology referenced in this document.

Neither CableLabs nor any member company is responsible to any party for any liability of any nature whatsoever resulting from or arising out of use or reliance upon this document, or any document referenced herein. This document is furnished on an "AS IS" basis and neither CableLabs nor its members provides any representation or warranty, express or implied, regarding the accuracy, completeness, noninfringement, or fitness for a particular purpose of this document, or any document referenced herein.

© 2007 - 2011 Cable Television Laboratories, Inc.
All rights reserved.

Document Status Sheet

Document Control Number:	OC-SP-CoDF-I02-110325			
Document Title:	OpenCable Content Definition Format (CoDF)			
Revision History:	I01 – Released 01/14/08 I02 – Released 03/25/11			
Date:	March 25, 2011			
Status:	Work in Progress	Draft	Issued	Closed
Distribution Restrictions:	Author Only	CL/Member	CL/Member/ Vendor	Public

Key to Document Status Codes:

Work in Progress	An incomplete document, designed to guide discussion and generate feedback that may include several alternative requirements for consideration.
Draft	A document in specification format considered largely complete, but lacking review by Members and vendors. Drafts are susceptible to substantial change during the review process.
Issued	A stable document, which has undergone rigorous member and vendor review and is suitable for product design and development, cross-vendor interoperability, and for certification testing.
Closed	A static document, reviewed, tested, validated, and closed to further engineering change requests to the specification through CableLabs.

Trademarks:

Advanced Digital Cable™, CableCARD™, CableHome®, CableLabs®, CableNET®, CableOffice™, CablePC™, DCAS™, DOCSIS®, DPoE™, EBIF™, eDOCSIS™, EuroDOCSIS™, EuroPacketCable™, Go2BroadbandSM, M-Card™, M-CMTS™, OCAP™, OpenCable™, PacketCable™, PCMM™, and tru2way® are marks of Cable Television Laboratories, Inc. All other marks are the property of their respective owners.

Contents

1	SCOPE	1
2	REFERENCES	2
2.1	NORMATIVE REFERENCES	2
2.2	INFORMATIVE REFERENCES	2
2.3	REFERENCE ACQUISITION	3
3	CONVENTIONS	4
3.1	PROFILE LANGUAGE	4
4	ABBREVIATIONS AND ACRONYMS	5
5	INTRODUCTION	6
6	DATA MODEL	9
6.1	CONTENT UPDATES AND REPLACEMENT	9
6.1.1	<i>Carousel Item Processing</i>	9
6.1.2	<i>EISS Processing</i>	10
7	METADATA (INFORMATIVE)	12
7.1	REPRESENTATIVE MODELS	12
7.1.1	<i>Metadata overview of a national distribution package</i>	13
7.1.2	<i>Metadata overview requiring localization at the acquisition point</i>	14
7.1.3	<i>Template Overview</i>	15
7.2	METADATA CONSIDERATIONS	15
7.2.1	<i>Primary CoDF</i>	15
7.2.2	<i>One or more secondary CoDF</i>	15
7.2.3	<i>Media Files</i>	16
7.2.4	<i>Template Files</i>	16
7.2.5	<i>Distribution Requirements</i>	16
7.2.6	<i>Addressability and Localization</i>	16
7.2.7	<i>CoDF Override</i>	16
ANNEX A	XML SCHEMA (NORMATIVE)	17
APPENDIX I	EXAMPLE REPRESENTATIONS OF CODF SEMANTICS (INFORMATIVE)	18
APPENDIX II	USE CASES	19
II.1	ETV – NO LOCALIZATION	19
II.1.1	<i>ETV Application – Simple Case</i>	19
II.1.2	<i>ETV Application – Complex Case 1</i>	19
II.2	ETV – WITH LOCALIZATION	19
II.2.1	<i>ETV Application – Complex Case 2a</i>	19
II.2.2	<i>ETV Application – Complex Case 2b</i>	20
II.2.3	<i>ETV Application – Complex Case 3</i>	20
II.2.4	<i>ETV Application – Complex Case 4</i>	20
II.2.5	<i>ETV Resource Generation – Simple Case</i>	20
II.2.6	<i>ETV Resource Generation – Complex Case 1</i>	21
II.2.7	<i>ETV Trigger Alignment – Simple Case</i>	21
II.3	OCAP-RELATED USE CASES	21

II.3.1 Unbound OCAP Application Deployment 21

II.3.2 Unbound OCAP Application Deployment – Data Update..... 22

II.3.3 Unbound OCAP Application Deployment – New Application Revision..... 22

II.3.4 Unbound OCAP Application Deployment – Multiple Sets of Java Zip Files Revision..... 22

II.3.5 Bound java applications 22

APPENDIX III EXAMPLE IMPLEMENTATIONS OF USE CASES (INFORMATIVE).....**23**

APPENDIX IV REVISION HISTORY**24**

Figures

Figure 5–1 - Components of a distribution package 7

Figure 5–2 - Representative Architecture 8

Figure 7–1 - National Distribution with all components..... 13

Figure 7–2 - Content elements requiring Metadata descriptions with localization 14

Figure 7–3 - Metadata template 15

1 SCOPE

This document defines a format by which interactive TV applications may be distributed to content delivery platforms. For instance, information authored in this format by an advertiser could be distributed to multiple cable systems, in order to support a national interactive advertising campaign.

This document specifies the Content Definition Format (CoDF), a standard format for application distribution between application providers and application playout systems. This standard specifies the format for defining the file system content, application signaling, signing, and carousel configuration of applications. It further specifies deployment instructions for those applications, including their timing and destinations.

2 REFERENCES

2.1 Normative References

Note: Information contained in these normative references is required for all implementations. Notwithstanding, intellectual property rights may be required to use or implement these normative references.

The following documents contain provisions which, through reference in this text, constitute provisions of the present document. References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific:

- For a specific edition reference, subsequent revisions do not apply.
- For a non-specific edition reference, the latest version applies.

References	Specific Edition	Description
[CoDF XSD]	I02	OC-CoDF-I02.xsd, March 25, 2011, Cable Television Laboratories, Inc.
[CoDF EXMPL]		http://www.cablelabs.com/advancedadvertising/specifications/codf.html , Cable Television Laboratories, Inc.
[ETV-AM]		OpenCable Enhanced TV Application Messaging Protocol 1.0, OC-SP-ETV-AM1.0-I06-110128, January 28, 2011, Cable Television Laboratories, Inc.
[ETV-BIF]		OpenCable Enhanced TV Binary Interchange Format 1.0, OC-SP-ETV-BIF1.0-I06-110128, January 28, 2011, Cable Television Laboratories, Inc.
[OCAP]		OpenCable Application Platform Specification (OCAP) Profile 1.x, OC-SP-OCAP1.1.5-110224, February 24, 2011, Cable Television Laboratories, Inc.
[OHI]		OpenCable Headend Common Download and Unbound Application Signaling Interface Specification, OC-SP-OHI-I01-061208, December 8, 2006, Cable Television Laboratories, Inc.
[RFC 4648]		IETF RFC 4648, The Base16, Base32, and Base64 Data Encodings, October 2006.

2.2 Informative References

This specification uses the following informative references.

References	Specific Edition	Description
[CONTENTv3.0]		CableLabs Content 3.0 Specification, MD-SP-CONTENTv3.0-I01-100812, August 12, 2010, Cable Television Laboratories, Inc.
[DVB-MHP 1.0.3]	ETSI TS 101 812 V1.3.1 (2003-06)	DVB Multimedia Home Platform 1.0.3, DVB-MHP 1.0.3, ETSI TS 101 812 V1.3.1 (2003-06); http://www.etsi.org/services_products/freestandard/home.htm

2.3 Reference Acquisition

- Cable Television Laboratories, Inc., 858 Coal Creek Circle, Louisville, CO 80027; Phone +1-303-661-9100; Fax +1-303-661-9199; <http://www.cablelabs.com>
- European Telecommunications Standards Institute (ETSI), 650, route des Lucioles, 06921 Sophia-Antipolis Cedex, FRANCE, Phone: +33 (0)4 92 94 42 00, <http://www.etsi.org/WebSite/Standards/StandardsDownload.aspx>

3 CONVENTIONS

The following conventions are used in this specification:

- The Courier New font type is used to indicate code examples, names of properties, and other information that **MUST** be entered exactly as-is: `code example font`
- **Boldfaced** text is used as emphasis.

3.1 Profile Language

Throughout this document, the words that are used to define the significance of particular requirements are capitalized. These words are:

"SHALL"	This word means that the item is an absolute requirement of this specification.
"SHALL NOT"	This phrase means that the item is an absolute prohibition of this specification.
"SHOULD"	This word means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before choosing a different course.
"SHOULD NOT"	This phrase means that there may exist valid reasons in particular circumstances when the listed behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
"MAY"	This word means that this item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.

4 ABBREVIATIONS AND ACRONYMS

This specification uses the following abbreviations and acronyms.

AIT	Application Information Table
CoDF	Content Definition Format
EBIF	ETV Binary Interchange Format
EISS	ETV Integrated Signaling Stream
ETV	Enhanced TV
NVMEM	Non-volatile Memory
OCAP	OpenCable Application Platform
XAIT	Extended Application Information Table

5 INTRODUCTION

This specification defines a Content Definition Format (CoDF) that describes the various components that may comprise a simple to complex application. An application may be based on the [ETV-BIF] or [OCAP] specifications. Components can be application resources, data, or signaling. The CoDF syntax includes instructions for how to integrate an application, data and signals into a stream, with or without a video/audio component. The means by which application resources and associated data are distributed is outside the scope for this specification.

An application may consist of a wide range of components, applications, data, and signaling, from a simple ETV or OCAP application to be integrated with a 30-second advertisement that gets delivered to a single market, to an application that is to be delivered to multiple markets via multiple acquisition points.

CoDF defines the means to control application integration via directives delivered in a single or a set of CoDF files. CoDF also enables the insertion of stream events into an existing program.

The various components of application content and packaging are illustrated in Figure 5–1. The definition of packaging is out of scope for this specification.

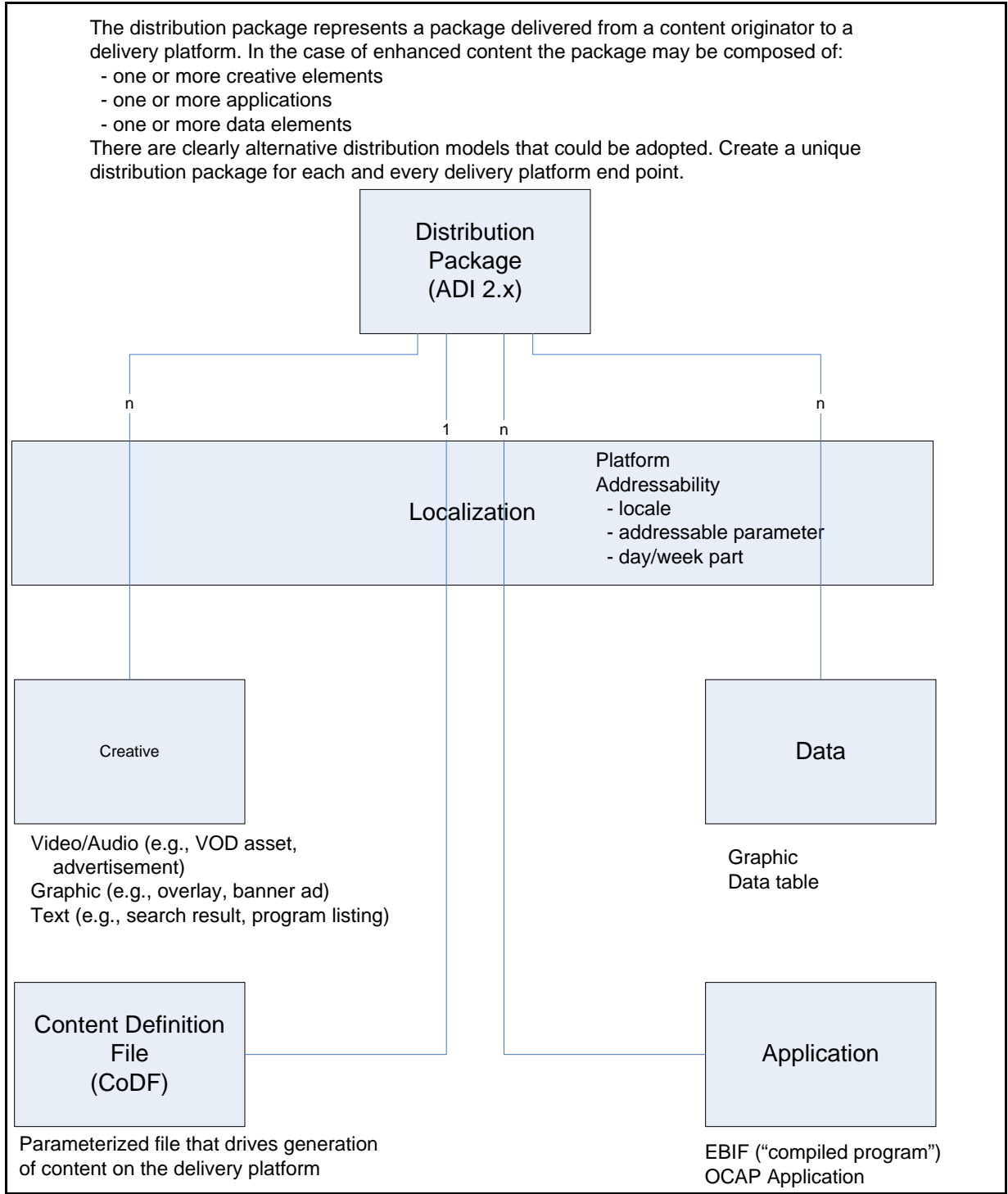


Figure 5-1 - Components of a distribution package

An overall architecture is represented in Figure 5–2 and illustrates the various components of a system for delivery and distribution of content. The figure also indicates which formats or interfaces are defined in this specification.

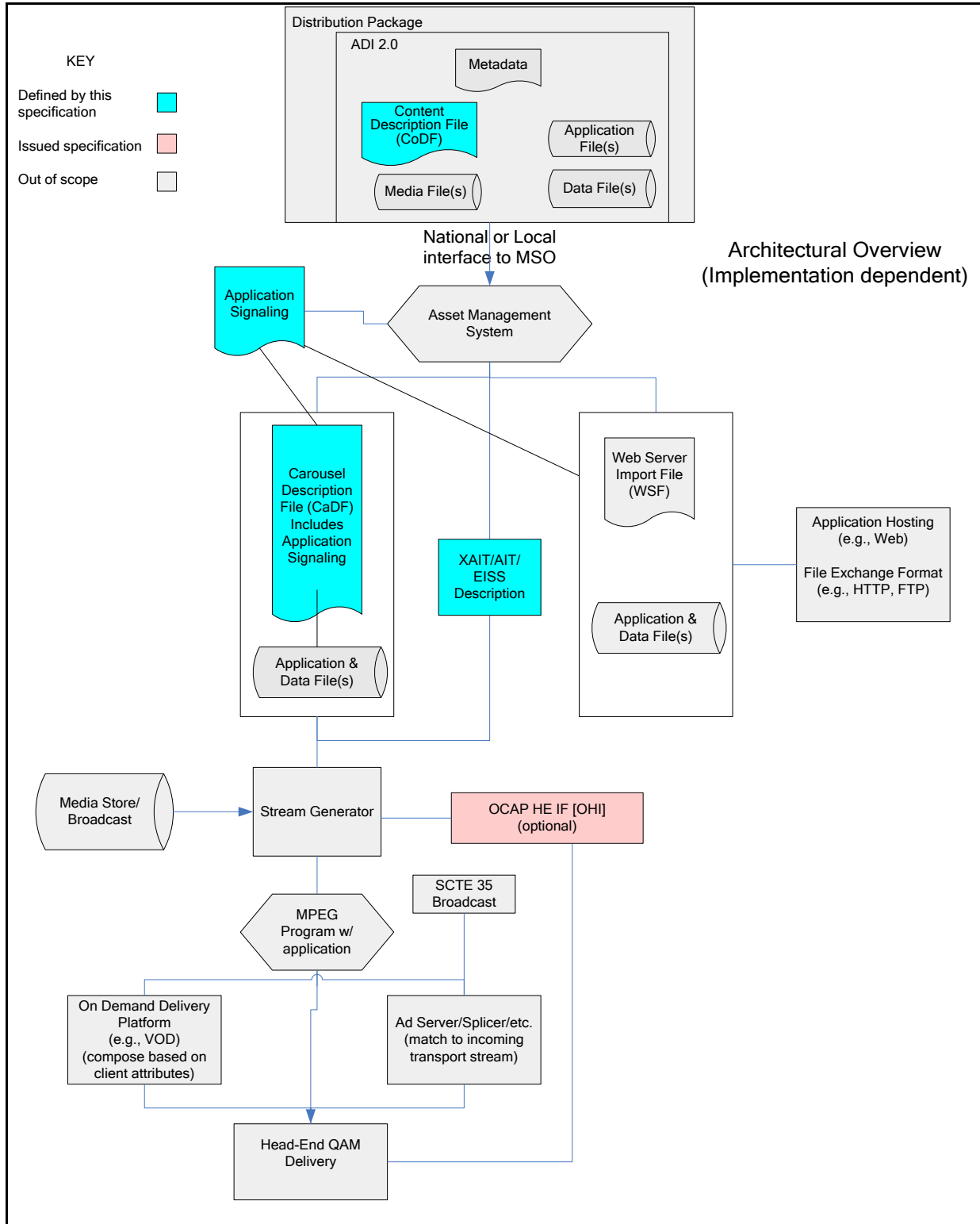


Figure 5–2 - Representative Architecture

6 DATA MODEL

The Content Definition Format (CoDF) data model is defined in [CoDF XSD].

Distribution of interactive TV applications to content delivery platforms SHALL conform to the semantics of the CoDF XML schema in [CoDF XSD].

6.1 Content Updates and Replacement

There are several models for applying CoDF updates or replacements. For example, two CoDF files may be submitted together to create an initial carousel delivery; subsequent to submitting the initial CoDF file or files, another CoDF file is sent to update all or a portion of the active carousel; or, an initial CoDF file may be submitted to create the initial carousel with one or more updates being applied from the same CoDF file sometime later to change the carousel contents. In this second case, Carousel and EISS content can be updated, replaced, or deleted by supplying a new CoDF document. In the last case, the CoDF file or files submitted can describe a carousel over an extended period of time (e.g., for a 60-second advertising break, the CoDF file may define the carousel for the first 30 seconds followed by a different carousel for the second 30-seconds). When a new CoDF document is received, the CoDF consumer SHALL observe the rules described below.

Note that there may be temporal considerations when performing Carousel and EISS matching, and that matching should only occur if the items are scheduled concurrently. The last case described above applies to a template-driven CoDF file generation process where the carousel items for the first 30 seconds and the second 30 seconds share a number of comment elements but are not scheduled concurrently.

6.1.1 Carousel Item Processing

A new carousel is considered to match an existing one when the StreamConfigName and ContentReference attributes contain the same string value, using case-sensitive string matching.

- a. If the new Carousel's ReplaceContent attribute is set to TRUE, then
 - a. If a matching Carousel exists, match up the Items contained in the new Carousel with the existing ones using the Identity attribute
 - i. For each matching Item,
 1. If the new Item's Delete attribute is TRUE, then remove the Item from the Carousel;
 2. Otherwise, replace the old Item on the Carousel with the new Item;
 - ii. For each non-matching Item that exists only in the old Carousel, remove the Item from the Carousel;
 - iii. For each non-matching Item that exists only in the new Carousel, append the Item to the Carousel;
 - b. If no matching Carousel exists, then append the Carousel to the list of Carousels;
- b. If the new Carousel's ReplaceContent attribute is set to FALSE, then
 - a. For each matching Carousel, match up the Items contained in the new Carousel with the existing ones using the Identity attribute
 - i. For each matching Item, replace the old Item with the new Item;
 - ii. For each non-matching Item that exists only in old carousel, leave the Item as-is;

- iii. For each non-matching Item that exists only in the new Carousel, append the Item to the Carousel;
- b. If no matching Carousel exists, then append the Carousel to the list of Carousels;

6.1.2 EISS Processing

A new EISS is considered to match an existing one when the StreamConfigName and ContentReference attributes contain the same string value, using case-sensitive string matching.

- a. If the new EISS's ReplaceContent attribute is set to TRUE, then
 - a. If a matching EISS exists,
 - i. Match up the EISSTable elements contained in the new EISS with the existing ones using the EISSTable's Identity attribute
 - 1. For each matching EISSTable,
 - a. If the new EISSTable's Delete attribute is TRUE, then remove the EISSTable from the EISS;
 - b. Otherwise, replace the old EISSTable in the EISS with the new EISSTable;
 - 2. For each non-matching EISSTable that exists only in the old EISS, remove the EISSTable from the EISS;
 - 3. For each non-matching EISSTable that exists only in the new EISS, append the EISSTable to the EISS;
 - ii. Match up the EISSDescriptor elements contained in the new EISS with the existing ones using the EISSDescriptor's Identity attribute
 - 1. For each matching EISSDescriptor,
 - a. If the new EISSDescriptor's Delete attribute is TRUE, then remove the EISSDescriptor from the EISS;
 - b. Otherwise, replace the old EISSDescriptor in the EISS with the new EISSDescriptor;
 - 2. For each non-matching EISSDescriptor that exists only in the old EISS, remove the EISSDescriptor from the EISS;
 - 3. For each non-matching EISSDescriptor that exists only in the new EISS, append the EISSDescriptor to the EISS;
 - b. If no matching EISS exists, then append the EISS to the list of EISSs;
- b. If the new EISS's ReplaceContent attribute is set to FALSE, then
 - a. For each matching EISS,
 - i. Match up the EISSTable elements contained in the new EISS with the existing ones using the EISSTable's Identity attribute
 - 1. For each matching EISSTable,
 - a. If the new EISSTable's Delete attribute is TRUE, then remove the EISSTable from the EISS;
 - b. Otherwise, combine the contents of the new and old EISSTables;
 - 2. For each non-matching EISSTable that exists only in the old EISS, leave the EISSTable as-is;

7 METADATA (INFORMATIVE)

This section is provided to inform metadata definition teams on assumptions made in the creation of this specification. As part of the metadata definition, there are a number of areas that should be accounted for:

- Packaging – How a CoDF file and associated binary files (EBIF, OCAP, etc.) are packaged for delivery.
- Distribution – How the packaged CoDF and associated binary files are distributed. The [CONTENTv3.0] specification includes provisions for specifying one or more delivery locations for a package.
- Localization/Addressability – In those cases where a different application is delivered to different physical locations, psychographic/demographic groups, etc., metadata will be necessary to communicate the scope of delivery.
- Delivery instructions – While the CoDF may specify the preferred delivery instructions as part of the metadata, external metadata may be included as part of the interface to the delivery platform to override the preferences. The overrides may be necessary to meet business agreements, infrastructure capabilities or limitations, destination device capabilities/limitations, etc.

The following section includes some representation models that touch on each of the areas. The last section provides some additional information to be considered.

7.1 Representative Models

This section provides three representative models to be considered as part of the metadata definition process. The three models are:

- national distribution that includes localizations as part of the nationally distributed package,
- national distribution that support localization at the acquisition point,
- the last model uses a template which is out of scope for CoDF but is a potential distribution model that may integrate with pre-defined CoDF packages at the acquisition or delivery platform.

7.1.1 Metadata overview of a national distribution package

Figure 7–1 illustrates the various components that may comprise advertising content destined for various locales. A locale represents a delivery platform (e.g., VOD server, carousel server, etc.) that services all or a subset of set-top devices connected to a service delivery network. Metadata is required to control the distribution and selection of these components for delivery to the various locales. The metadata may include addressability and/or delivery platform directives to further control the final delivery of the application.

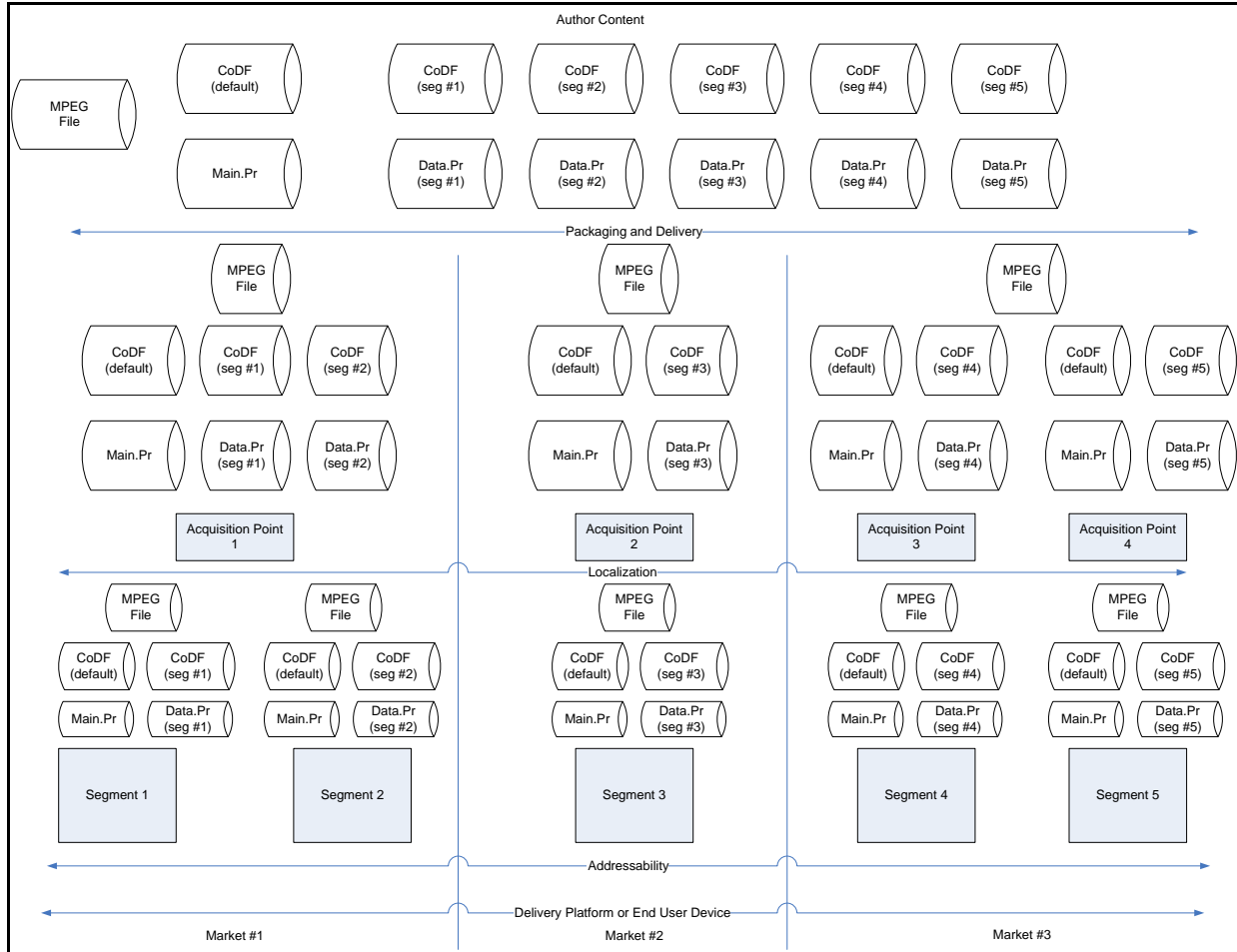


Figure 7–1 - National Distribution with all components

7.1.2 Metadata overview requiring localization at the acquisition point

Figure 7–2 illustrates the various components that may comprise advertising content destined for various locales. A locale represents a delivery platform (e.g., VOD server, carousel server, etc.) that services all or a subset of set-top devices connected to a service delivery network. Metadata is required to organize these components for distribution to the various locales.

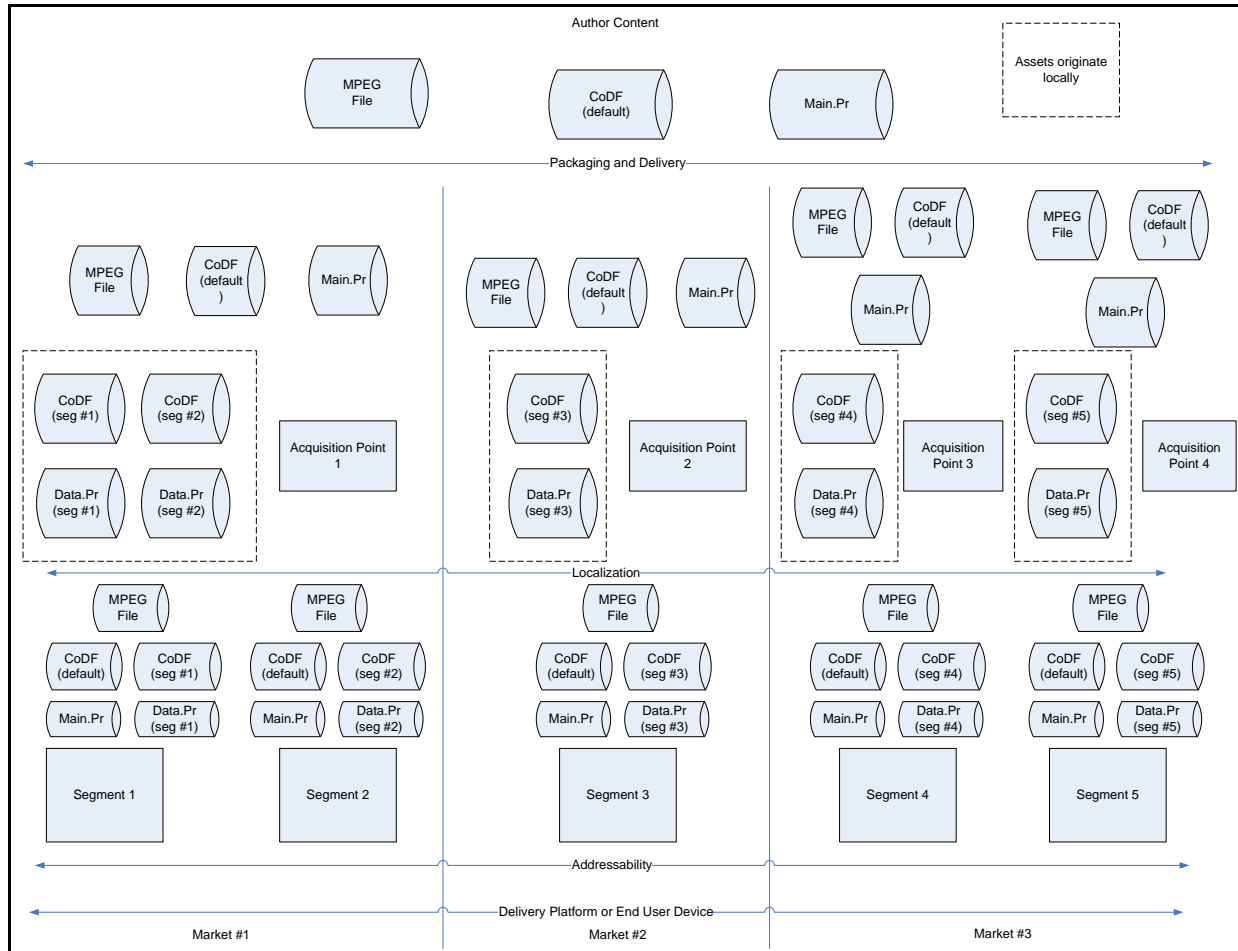


Figure 7–2 - Content elements requiring Metadata descriptions with localization

7.1.3 Template Overview

Figure 7–3 illustrates the general organization of content elements requiring metadata descriptions.

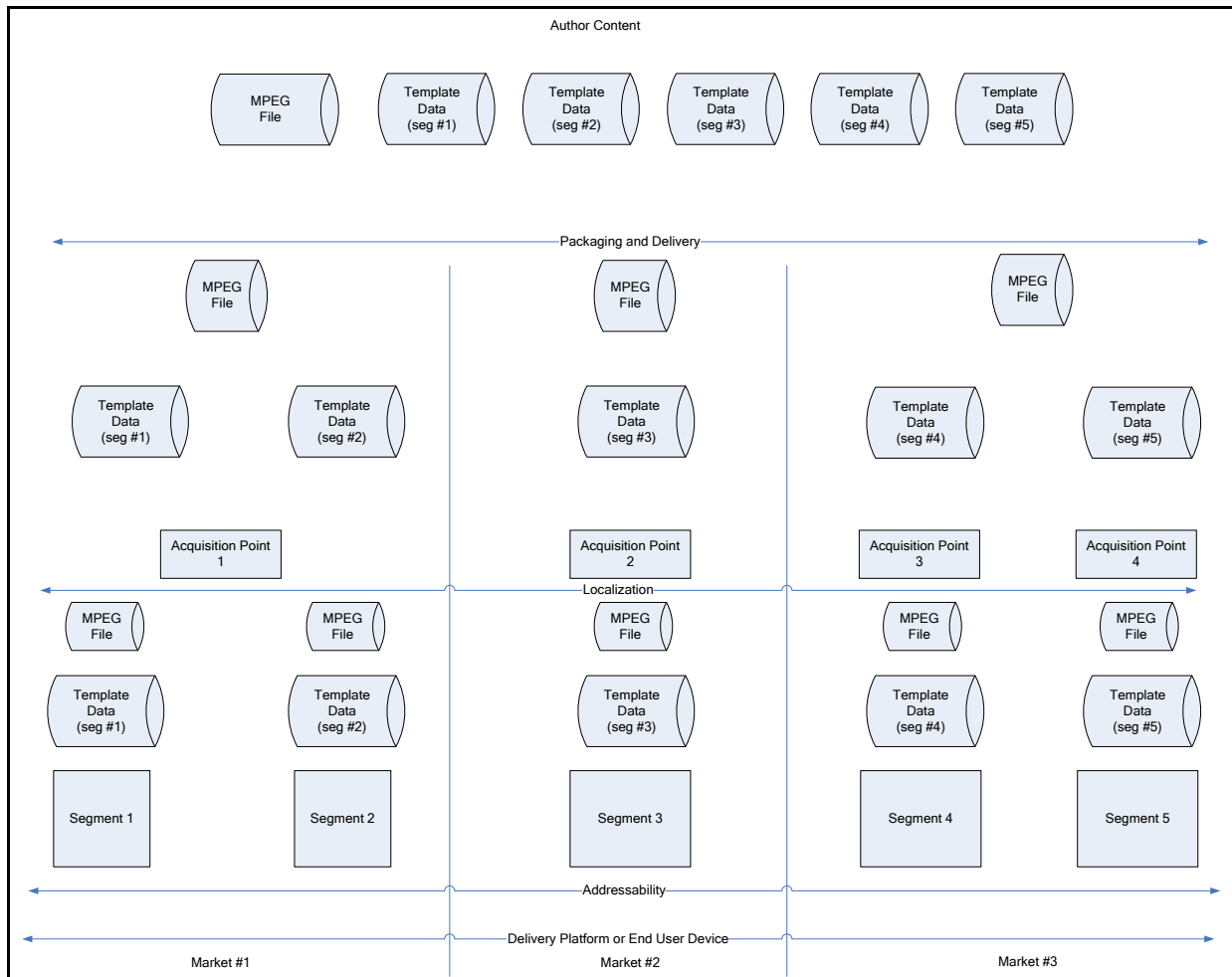


Figure 7–3 - Metadata template

7.2 Metadata Considerations

This section contains some specific considerations the metadata definition may want to consider.

Note: The definition of this metadata is out of scope for this specification.

7.2.1 Primary CoDF

Allow for one or more primary CoDF. Each primary CoDF will be constrained by locale and/or addressable parameters.

7.2.2 One or more secondary CoDF

Allow for one or more secondary CoDF. Each primary CoDF will be constrained by locale and/or addressable parameters.

After the primary CoDF is selected, one or more secondary CoDFs may be combined to create a finished delivery element.

7.2.3 Media Files

Similar to the primary CoDF, one or more media files may be included. Locale and/or addressable parameters will constrain the media file(s) applied.

7.2.4 Template Files

Similar to the primary CoDF, one or more populated template files may be included. Locale and/or addressable parameters will constrain the populated template file(s) applied. Template files are mentioned in this specification solely to inform the metadata specification authors. The definition of template files is out of scope for this specification.

7.2.5 Distribution Requirements

The [CONTENTv3.0] specification supports the concept of a distribution list. The metadata may include this or a similar construct to control the delivery and/or receipts of CoDF and related content files.

7.2.6 Addressability and Localization

While distribution metadata is utilized to control the delivery of CoDF, related content files, and/or other media files into an acquisition point, additional metadata to direct content preparation tools (i.e. application striper) or delivery platforms (i.e. on demand server) may be required to allow finer level of control over the content delivered to a specific subscriber or group of subscribers.

7.2.7 CoDF Override

One or more of the parameters in the CoDF file may be overridden via metadata. The overrides will adhere to this namespace. For example, the bit rate in the original CoDF may exceed infrastructure limitations of the delivery platform in place.

Annex A XML Schema (Normative)

The formal data definition is found in [CoDF XSD].

Appendix I Example representations of CoDF semantics (Informative)

Examples of CoDF data expressions can be found in [CoDF EXMPL].

Appendix II Use cases

This informative appendix contains ETV and OCAP-related use cases.

II.1 ETV – no localization

II.1.1 ETV Application – Simple Case

- Baseline Profile
- Single Set of Resource Files

Ad sales operations have an existing installed base of MPEG encoding subsystems that are used to encode/prepare advertisements to be placed on an ad server or video on demand server. The equipment comprising the subsystem normally includes a control station, a tape deck and an encoder (integrated or external). In the simplest use case, the local ad sales operation wants to be able to add an ETV enhancement to the content. This could be accomplished by the workstation ingesting a file generated by an ETV application generation tool. Alternatively, a stand-alone application server could take the MPEG file output and ETV file and combine into a new MPEG file.

The ETV application tool should not need to be aware of the length of the target asset (don't mandate that the ETV tool create an MPEG file). Separate metadata may be defined to communicate to the MPEG file generator:

- Delay from the start to start integrating the EBIF elements
- Bit rate to insert the EBIF elements at – the bit rate may vary from the start to finish
- The point in the content to stop and/or disable the application.

II.1.2 ETV Application – Complex Case 1

- Range of profiles
- Multiple resource files for different profiles

An ad sales agency has an ETV development tool that can generate applications that can be targeted to a range of set top box profiles as well as create resources targeted to one or more profiles. The ad sales agency will generate an ETV file or file set comprising the full set of profiles and resources. This ETV file or file set will be shipped along with a betamax tape of the advertising content to be encoded by the system operator.

The system operator will encode the content and, as part of the workflow, add in the ETV content that matches the equipment deployed at their site. A given operator may have different equipment at each site. The final MPEG file will be sent to the ad server, on demand server or other equipment for delivery to the client.

The ETV file or file set may include a metadata file that describes the contents of the file or fileset for automated ingest, validation and resource selection.

II.2 ETV – with localization

II.2.1 ETV Application – Complex Case 2a

- Range of profiles
- Multiple resource files for different profiles

- Multiple resource files for different locales

As in the previous use case (Complex Case 1), ETV files are delivered to the system operator along with the betamax tape. Additionally, the content provider includes a set of resources packaged into a file or set of files that contain localizations. Localizations may include graphics, tables of dealer locations, localized pricing, etc.

Once received by the system operator, the content is encoded and the appropriate application and resource files that match the equipment and location are combined into a finished MPEG file. Metadata will be required to assist local site personnel or processes to efficiently create the final MPEG file(s).

II.2.2 ETV Application – Complex Case 2b

- Range of profiles
- Multiple resource files for different profiles
- Resource files for a given locale are delivered independently of the original core application

In this use case, rather than localization files being distributed with the original material (see Complex case 2a), they are received independently. In this case each local business may have a generation tool to create the local resource and forward to the system operator

II.2.3 ETV Application – Complex Case 3

- Range of profiles
- Accommodate the insertion of one or more resources within a downstream system

This use case is similar to "Complex Case 1", except an MPEG file is distributed by the content provider containing the applications and resources for a range of profiles. Additionally, as in "Complex Case 2b", a localization file is provided from an alternate source (e.g., local business).

In addition to metadata describing how to select the correct combination of applications and resources, the application encoding should make it easy for a downstream system to easily identify and combine the application and resource files, including localizations, to create the finished MPEG.

II.2.4 ETV Application – Complex Case 4

- Range of profiles in a live feed
- Accommodate the insertion of one or more resources within a downstream system

In this use case, the content is received from the content provider within the live feed. At receipt at the system operator, the following processes must be accommodated:

1. Selecting the correct set of application and resources required to meet the needs of the local plant
2. Insertion or replacement of resource to meet localization requirements.

It is anticipated that metadata will be sent in advance that will identify the content in the live feed (e.g., Ad-id, ISAN, etc.) and the processes to be completed.

The file and delivery formats must accommodate the required processing in near real time.

II.2.5 ETV Resource Generation – Simple Case

- Generate a standalone resource file

A local business or other content contributor must be able to create localization files to be delivered to the system operator.

II.2.6 ETV Resource Generation – Complex Case 1

- Multiple resource files for different locales

A local business or other content contributor must be able to create a complex localization file that can provide different resources for different profiles and/or locales.

II.2.7 ETV Trigger Alignment – Simple Case

A local business creates a standard MPEG file that highlights the separate sections of their business. The business engages with a content creation company to create a high quality video/audio experience and has EBIF triggers placed at the correct locations within the content file.

On a periodic basis the business creates an updated ETV application and resource file set that displays messages based on receipt of the triggers. This file is delivered to the system operator so the content can be refreshed.

While not within scope of this effort, it is anticipated that voiceovers could also be updated.

II.3 OCAP-related use cases

II.3.1 Unbound OCAP Application Deployment

1. Application developer provides a new unbound application to an MSO. This application contains the following:
 - a. Java-classes for the application;
 - b. Fonts for the application;
 - c. a set of common images;
 - d. a set of common screen definitions;
 - e. 3 different sets of images (corresponding to 3 different Guide UIs deployed in MSO's network); note: this means that appearance must conform with the Guide
 - f. 2 different sets of screen definition files corresponding to the device type (DVR or non DVR);
 - g. URIs for gathering some dynamic data (e.g., new feed), and data update interval;
 - h. XAIT information for the apps (e.g., OrgId, AppId, priority, parameters, BaseDir, Classpath, MainClass)
2. Network operator receives the application (and metadata, assets), inserts this application into the asset management system, integrates supplied metadata with the local environment (e.g., XAIT), and adds the application to the to-be-tested list.
3. Testing lab administrator provisions the test labs to test various configurations of the application on various test systems. Testing lab administrator reviews the application descriptions and specifies various targeting criteria to make sure the right set of the application data are downloaded to the right set of devices. Application is tested in two different types of networks: one using http based delivery and another based on carousel based delivery. During this phase live feed of dynamic data is used.
4. Once the testing is successfully completed testing, lab administrator marks the application to be ready for trials.
5. Network operator chooses a set of trial sites and provisions the asset management system to include this application in the list of applications available to those sites. Network administrator also sets various targeting criteria to make sure the right data set is downloaded to the right device.
6. The asset management system interfaces with multiple carousel generators to provide application information (see #1 in this section), application files. It also provides information for XAIT generation, e.g.,

various targeting criteria. The asset management system also interfaces with http servers to host the application files.

7. After the trial, the network operator provisions the asset management system to deploy the application on several sites.

II.3.2 Unbound OCAP Application Deployment – Data Update

1. Application developer provides a new set of data-files corresponding to an application. The data-set is tagged with targeting descriptions as in the use-case in section II.3.1. None of the other aspects of the application has changed.
2. Operator receives the data, inserts the data to the asset management system. Seeing that this data does not need additional testing provisions this new data-set for deployment.
3. Asset management system interfaces with carousel-generators and http servers to update the data.

II.3.3 Unbound OCAP Application Deployment – New Application Revision

Same as the use-case in II.3.1, but in this case a new revision of an already deployed application is received from the application developer. In this use-case, we have to consider the case where one version of the application is being deployed in some devices, where as the older version is being deployed in all the other devices. Similarly, consider the case where one system is trialing the new revision of the app, whereas all the other systems have deployed the earlier version. Also consider the case where the operator schedules a time-period to switch from the old version to the new version.

II.3.4 Unbound OCAP Application Deployment – Multiple Sets of Java Zip Files Revision

Same as the use-case in II.3.1, but in this case there are multiple sets of java files (specified as separate sets of zip files), each set targeting a device characteristic.

II.3.5 Bound java applications

- A national advertiser with a local presence wants to support replacing the national weekly specials with a local version.
- A national information channel (news, weather, financial services, etc.) wants to support the insertion of detail regional/local information, logos, etc.
- Content provider provides everything in stream.
- Headend adds events to content provider provided stream.
- Update of application/content.
- Applications can be updated without updating content.
- Content can be updated without updating application.

Appendix III Example implementations of Use Cases (Informative)

Example implementations of use cases can be found in [CoDF EXMPL].

Appendix IV Revision History

The following ECNs were incorporated into version I02 of this specification:

ECN	Title of EC	Date Accepted
CoDF-N-10.1551-2	Document Restructure	3/25/11
CoDF-N-10.1552-5	Updates for ETV I06	3/25/11
CoDF-N-10.1570-2	Clarification of anyAttribute	3/25/11
CoDF-N-10.1583-1	Enforce Proper Values Within Usage Context	3/25/11
CoDF-N-10.1584-2	ETV Application Argument	3/25/11
CoDF-N-10.1585-2	Usability Improvements	3/25/11
CoDF-N-10.1586-2	Addition of Version Type	3/25/11
CoDF-N-11.1639-2	Content Updates and Replacement	3/25/11

