Nagravision VOD

XML VOD Catalogue

Version 4.23.0

Status: Approved

Filename	Import-CMS-VOD-Interface-v4.23.x.docx
Date	31-May-2016
Author(s)	Christophe Sanchez, Wendy Molyneaux, Salim Azzabi, Christophe Rickli
Client/Project	CMS 4.23
Owner	C. Sanchez

CH-1033 Cheseaux, Switzerland.
www.nagra.com

Contents

Preface

Audience

Nagravision product and engineering departments are the Internal attended audience.

Third party engineering departments for integration purpose are the External attended audience, and are called data provider within the document.

Document Structure

This document Nagravision describes the basis of the Nagravision import interfaces that will be used for VOD projects.

This document is a generic view of the Nagravision interfaces. For a specific project this document is always provided with another document that explains the customization of the project like the list of EPG fields, the available locales, the non-usage of some elements and the specific constraints on some attributes. Only both generic and project specific documents define clearly the Nagravision VOD interfaces in the project context.

Document Organization

This document is organized as follows:

Chapter 1: this chapter provides an overview of the import and explains the FTP mechanism used by the Media Live CMS agent tool to import automatically the XML files provided.

Chapter 2: this chapter describes the import XML file format. It is composed of the following sub chapters:

Chapter 2.1: information about XML file encoding.

Chapter 2.2: all the format information related to the entire file like the date format or EPG data that is used in all part of the XML describes in the subsequent chapters.

Chapter 2.3: the part of the XML related to the Content management, import of `<Content>`, <Promotion>, <Series> or <AdInsertionProfile> and all related elements.

Chapter 2.4: the part of the XML related to the VOD management, import of `<Catalogue>`, `<DefineNode>` or `<VodItem>` and all related elements.

Chapter 2.5: the part of the XML related to the Product management, import of `<Product>` and all related elements.

Chapter 3: this chapter describes the error XML file format.

Chapter 4: Some examples of import file. These examples are generic, they cannot be taken in account without a project specific adaptation.

Conventions Used

When the document references an XML element within the text, this element is written in another font and is placed between the XML tag delimiters like this: `<Content>`. Other references to the same word make refer to the Nagravision interpretation of this word, for this example, where the word is written in normal text font "Content" describes all the editorial and eventually technical information of a movie itself. Following the same idea, the attribute name is also written in another font like this: `scheduleDate`.

Conventions used inside the XML file are described in the §2.

Note

All the XML tags of the import file are indexed to easily find them. But the index works fine only if the format information (¶) of the document is hidden.

Note

This document references the Media Live CMS application (GUI, import tool, database…). The name Media Live CMS is the name of the Nagravision CMS application.

Document and Schema versionning convention

It is defined that the versionning of the Import Nagravision VOD Interface is based on three parts incremented as follow:

- The first part is incremented when a not-retrocompatible changes is performed (for instance when the root tag is changed from ScheduleProvider to LysisData)
- The second part is incremented when the changes impact the Schema (for instance add a new tag)
- The third part is incremented when the changes impact only the documentation (for instance a correction in an explanation or in the typography)

The Schema file has only a two parts version number and the document has the three parts.

Note

This chapter is added in the XSD too, in a comment.

List of Open Points

No table of figures entries found.

Related Documents

[1] ISO-639-2 language definition
ISO
[2] ISO-4217 currency definition
ISO
[3] ISO 3166-1 country definition
ISO

[4] ISO-8601 date format definition
ISO
[5] Import-CMS-VOD-Interface Schema
Import-CMS-VOD-Interface-v4.20.xsd
[6] CMS Data Fields Specification, EPG Fields for VOD and BTV
DataFields-CMS-v4.20.x.docm

Document History

Change logs
4.22.0 / 2017-07-03 / Neeraj Kumbhkar • Create Import-CMS-VOD-Interface-v4.22.x.docx based on Import-CMS-VOD-Interface-v4.21.x.doc

Document Reviewers

Name	Function	Review Date
Pedro Ferreira	CMS PLM	

Document Approvers

Name	Function	Approval Date
Christophe Sanchez	CMS Architect	

1. Overview

The goal of this interface is to get VOD content details from a data provider.

The interface proposed by Nagravision is an independent interface to exchange data with third party's systems. The standard chosen, based on XML standard, has, among other benefits, the following advantages:

- Flexible; however please note that changes to tags generally imply modifications to source code.
- Multilingual and multi-locale, as a language and country tag specifies which language is considered.

The Media Live CMS import tool for this interface will be configured as an agent that will check for incoming files on a regular basis. The file will be received with FTP. The Media Live CMS agent is an ftp client, with the ftp server being on the data provider system or eventually a third party's system. For integration testing and special cases it will also be possible for the Media Live CMS user to manually import a file.

1.1 File processing

The CMS import will be deployed as an agent that will run on a regular basis. When launched, the agent will retrieve the list of files to be processed by examining a specific configured directory, looking for data files whose name match a specific pattern (generally *.xml). This directory is located on a file server accessible to the CMS. FTP protocol is supported as well as direct file access.

Files will be processed in an order according to case sensitive alpha-numeric sorting of the file names. Therefore to ensure CMS processes files in the same order that they were generated it is proposed that the beginning of the file name contains the date and time of the file generation in a numeric format, year first.

Important:

In order to prevent the file access errors that would occur if the CMS agent were to try to process a file while it is still being generated or copied by the data provider, the data provider will perform a two stage provisioning operation. It will first generate or copy the file into a working directory and/or with a file extension that is different from that the CMS agent examines. Then secondly it will change directory and/or change file extension to correspond with what the CMS agent examines. If different directories are used, it is essential that they are on the same physical disk and file system so that the operating system can perform the change in an atomic action without provoking an extra data copy. In this way the CMS agent can only "see" the files once they are complete.

The CMS agent will make a local copy of the source file from the data provider's ftp server. Once this copy is done, it modifies the source file name adding the suffix ".progress" to show this file being processed.

When the CMS has processed the file, it returns an acknowledgement to the data provider by renaming the source file. The ".progress" suffix is removed and the suffix ".success" added if the import was successful, or ".failed" added when errors are detected during the import.

In the case of failure an error log file with the same name as the source file and an ".error" extension will be transmitted as described in section 3. In addition an alarm will be raised.

No log file is explicitly generated for successful import operations, however messages are logged in relevant monitoring view in the CMS GUI.

When processing multiple files, if an error is detected for one file, the import process will still try to import the subsequent files.

It is assumed that the data provider will take care of deleting the processed files once it has taken note of the acknowledgement.

After having successfully imported a file, provided that not other changes are made in the Media Live CMS database to the imported objects, the same file can be re-imported successfully.

2. XML file format

2.1 File encoding

The character encoding of the source file must corresponding to the "encoding" attribute of the first line of the xml file. Nagravision advises to use UTF8 encoding in order to easily manage ASCII and non-ASCII texts in a uniform way.

2.2 Common data

2.2.1 Locale

The locale abbreviation is the abbreviation of the different languages and countries defined in the XML document, usually, it is based on ISO-639 [1] 2 letters languages and ISO 3166-1 [3] 2 letters country ('en_UK' for British English, 'en_US' for American English, 'fr_FR' for French, or 'zh_SG' for Singapore Chinese). The locales used by the Nagravision system are defined by the configuration of the project (i.e. the Nagravision operator is guided in his possible choices of language).

2.2.2 Identifier

2.2.2.1 Technical identifier

Some elements in this schema have an `id` attribute (xsd IdType) for the following reasons:

a) To reference the element from an other element;

b) Only elements with an `id` could be updated or deleted. The data provider is the owner of this `id`; however Nagravision will use it as its Public ID that will be used to uniformly synchronize all the third parties' systems.

The `id` attribute is used to identify the item in the Media Live CMS database to be created, updated or deleted. When creating a new item the maximum length of this field is not validated by the XML syntax check phase of the import process, but will be checked by Media Live CMS's configurable validation applied when data is saved to the database. The maximum possible length is 50 characters. An ID should generally be composed of printable ASCII 7-bit characters without spaces. Depending of the projects, the format and length of this ID can varied.

The regexp pattern that validates the id is: [a-zA-Z0-9_\.\+]+[a-zA-Z0-9_\-\.\+]*

2.2.2.2 Human identifier

The `title` or `name` attribute is used to provide a unique working title for the item. This title will be shown in the Media Live CMS GUI so the user knows what item he is working with. It is intended that the item's title that will be displayed to the TV viewer be supplied in the relevant child EpgDescription. When creating or updating the item the maximum length of this field is not validated by the XML syntax check phase of the import process, but will be checked by Media Live CMS's configurable validation applied when data is saved to the database. The maximum possible length is 300 bytes when stored in the database, but the configurable validation counts the number of characters, taking into account multi-byte character encodings for languages like Chinese. The default maximum length is 100 characters.

2.2.3 Date formats

The date format is based on ISO-8601 [4]. **All dates are given in GMT time**. The mandatory formats are:

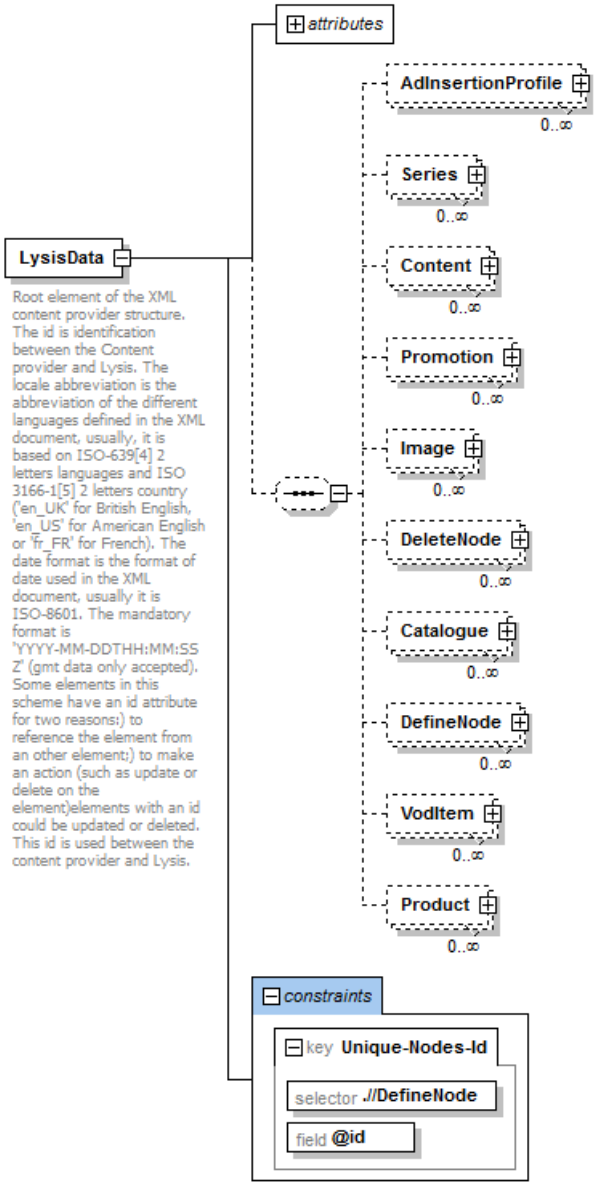
- SimpleType date: `yyyy-mm-dd`
- SimpleType time (GMT based): `hh:mm:ss`
- SimpleType gmtdatetime: `yyyy-mm-ddThh:mm:ssZ`

The dashes, colons, T and Z characters are mandatory field separators.

In terms of schema validation these types are a restriction of string, with the following pattern applied:

- date: `[0-9][0-9][0-9][0-9]-[0-1][0-9]-[0-3][0-9]`
- time (GMT based): `[0-2][0-9]:[0-6][0-9]:[0-6][0-9]`
- gmtdatetime: `[0-9][0-9][0-9][0-9]-[0-1][0-9]-[0-3][0-9]T[0-2][0-9]:[0-6][0-9]:[0-6][0-9]Z`

2.2.4 LysisData element



`<LysisData>` is the root element. It may contain any number of <AdInsertionProfile>, <Series>, <Content>, <Promotion>, <Image>, <DeleteNode>, <Catalogue>, <DefineNode>, <VodItem> or <Product> child elements. The data provider can choose how he organizes how many elements he puts in each separate xml file. He may put one element per file, or group all data in a single file. The restriction is if an element references another one, the referred element must be present in a previously imported file or at least in the same file. When deciding upon this organization it needs to be taken into account that if an error is found while processing a file, none of the data changes implied by that file will be applied to the Media Live CMS database. In the default "override" action mode (see attribute "action" in the following

element descriptions) Media Live CMS can accept processing a file that contains data already imported; however large scale re-processing of identical data should generally be avoided as its wastes processing resources. Therefore for optimal performance Nagravision advises providing XML files that contain only new items or updates & deletes, and grouping about 100 elements per input file.

Attributes details

Name	Type	Mandatory	Description and example
`id`	IdType, max 50 char, unique for the provider domain	Yes	Unique identification of the data provider. Eg: LYS1234
`scheduleDate`	GMT date and time	Yes	Date and time of file generation. Eg: 2005-02-25T22:30:00Z

The `id` attribute must match the ID of one of the company objects already existing in the Media Live CMS database that has the role "Broadcaster". All content items created from the input XML file will be associated to that company. This ID will provide the content attribute "companyId" subsequently exported to the VOD portal.

The `scheduleDate` attribute must be present and syntactically correct but its value is not further processed by the current Media Live CMS import tool. Nagravision recommends that the data provider supplies the timestamp of the file's generation for documentation purposes.

Note

A key constraint is defined to insure the unicity of the <DefineNode> IDs within the entire XML document.

2.2.5 SimpleType ActionEnumType

This type is an enumeration of string that can have the following values:

- delete
- override

This type is used as an attribute `action` that controls the behavior of the XML consumer application with respect to XML files containing data already supplied in previous files already entered into the Media Live CMS database.

The action **delete** allows the file producer to specify that the referenced element no longer exists. The file consumer will generate an error during the import process if deletion of this element would violate the consistency of its database. The the element does not exist in the Media Live CMS database a warning message will be logged and the import processing will continue.

The action **override** (default behavior if the action attribute is not present) causes the file consumer to create the data element if it does not already exist or to update existing elements.

The `id` attribute of the concerned XML element is used to identify the concerned data item.

2.2.6 SimpleType DurationType

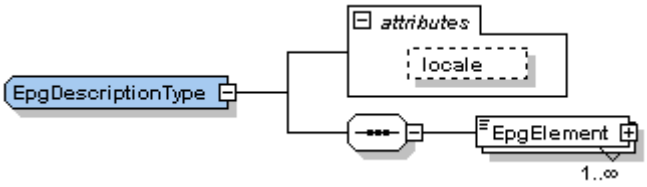
source

```
<xs:simpleType name="DurationType">
    <xs:annotation>
        <xs:documentation>Type for durations</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:long">
        <xs:minInclusive value="0"/>
    </xs:restriction>
</xs:simpleType>
```

This type is a simple description of duration type attributes.

2.2.7 EpgDescription element

The `<EpgDescription>` elements within <Series>, <Content>, <Promotion>, <Image>, <Catalogue>, <DefineNode>, <VodItem> and <Product> are defined by the type EpgDescriptionType:



Attributes details

Name	Type	Example	Mandatory	Description
`locale`	String, 5 chars, ISO-639-2 and ISO 3166-1, ascii	fr_FR or en_UK.	No	The locale in which the EPG description is written. 5 chars, ISO-639-2 and ISO 3166-1, ASCII explained in §2.2.1.

This element provides a container for supplying a set of generically defined data values. Each value is defined as a keyword – value pair in a `<EpgElement>` child element. Data values can be classified into two groups:

- Textual items that are translated into more than one language, typically titles and synopsis
- The rest – where the raw data value is language independent, for example parental rating code, production year.

Within a parent element, there will be one `<EpgDescription>` element for each language to carry the data from the first group, with the language specified by the `locale` attribute. All the text elements within such an `<EpgDescription>` element should be supplied in that language. The allowed locales are configured per customer project, and files containing a non-configured locale will be rejected as errors.

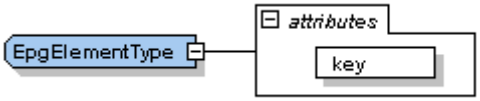
All the data values of the second group are supplied together in one `<EpgDescription>` element, and here the `locale` attribute will be omitted.

Reference document [6] provides the details of all the available fields, including which are localized. The fields that are defined to be localized must be given within an <EpgDescription> that specifies the "locale" attribute, and the non-localised ones must be supplied in the <EpgDescription> without attribute "locale". The CMS's import process will raise and error and reject the soure xml if this is not respected.

The value of the `locale` attribute is described in §2.2.1.

2.2.8 EpgElement element

The `<EpgElement>` elements within `<EpgDescription>` are defined by the type EpgElementType:



Attributes details

Name	Type	Example	Mandatory	Description
`key`	String, max 300 char, unique, UTF-8	title	Yes	Case sensitive name of the EPG field. The list of allowed keys is configured for each customer project.

This element provides the value of an EPG field of the item in which it is embedded.

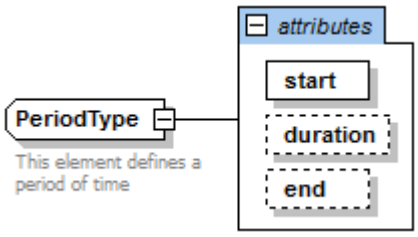
The content of the element is the field value. The tag may be empty, corresponding to specifying a "null" value for the field, which will effectively delete the field value from the Media Live CMS database if it exists.

The list of allowed field names and the format of their values are defined in the project specific document that accompanies this generic document.

The `key` attribute must exactly match the EPG field names configured in the Media Live CMS database. Unknown field names are considered an error.

2.2.9 Period element

The `<Period>` elements within `<VodItem> and <AddToProduct>` are defined by the type PeriodType:



Attributes details

Name	Type	Example	Mandatory	Description
`start`	GMT Date and time	2002-02-25T03:30:00Z	Yes	Date and time of the start of the period.
`end`	GMT Date and time	2002-02-25T04:30:00Z	No	Date and time of the end of the period. **(*)**
`duration`	DurationType, in seconds	3600	No	Duration in seconds of the period. **(*)**

(*): Either 'duration' or 'end' attribute is mandatory

Generally only one of the `end` or `duration` attributes is provided. If the `end` attribute is provided, the validity period of the element is between the start date and the end date. If the `duration` attribute is provided, the end date is calculated by added the duration to the start date.

Note

If both `end` and `duration` attributes are provided together, the calculated end date (start date + duration) must be equal to the provided end date otherwise an error is raise by the Media Live CMS import tool.

2.3 Content management

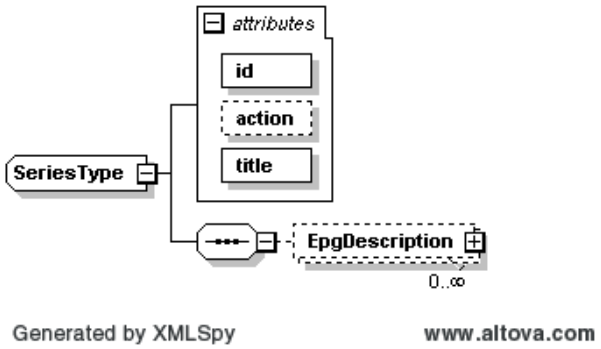
This chapter will regroup the <Series>, <Content>, <Promotion>, <Image> and <AdInsertionProfile> elements and their sub-elements. All these elements participate to the Content data.

If a content element is specified without a version, it will be imported as a content version of an auto-generated content, which is a clone of the specified content.

If an auto-generated main content is updated between two imports, all the modifications will be lost when the content that triggered its creation is updated in a new import. This content will be overridden.

2.3.1 Series element

The <Series> elements within <LysisData> are defined by the type SeriesType:



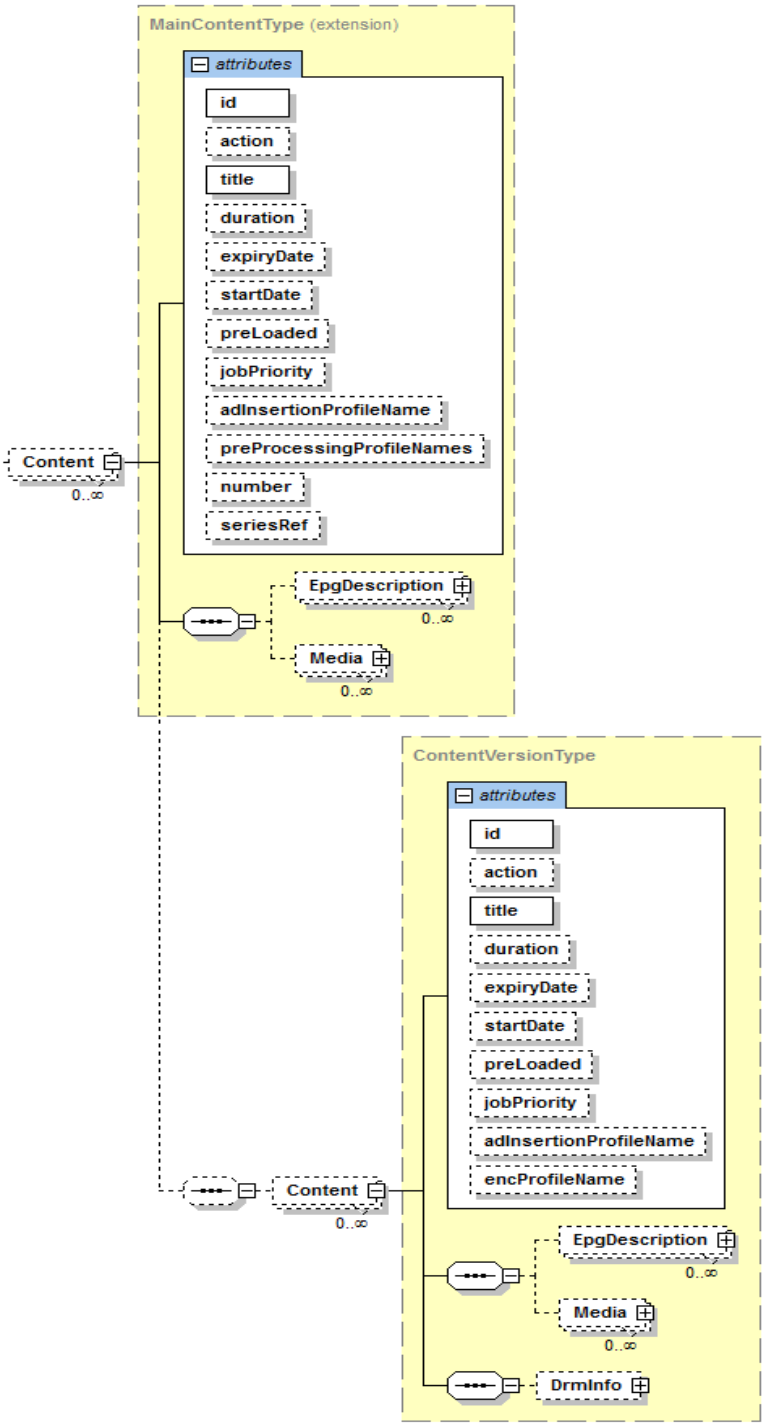
The data provider provides information about available Series. It includes the series EPG description.

Attributes details

Name	Type	Example	Mandatory	Description
id	IdType, max 50 char, unique for the provider domain	A12343	Yes	Unique identification of the series, see §2.2.2.1 for more information.
title	String, max 100 char, UTF-8	Desperate Housewife	Yes	Series working title, see §2.2.2.2
action	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information

2.3.2 Content element

The <Content> elements within <LysisData> are defined by the type ContentType:



Attributes details

Name	Type	Example	Mandatory	Description
`id`	IdType, max 50 char, unique for the provider domain	A12343	Yes	Unique identification of the content, see §2.2.2.1
`action`	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information
`title`	String, max 100 chars, UTF-8	A Sunny Summer	Yes	Movie title, see §2.2.2.2
`duration`	DurationType, default value : 0	7200	No	Movie approximate duration in seconds.
`number`	Integer	5	No (Forbidden for content versions)	The index of the episode in the series. If specified, *seriesRef* attribute must be filled. Must be omitted for episode/content versions. Zero value is allowed and is suggested when no "proper " number is available Note: this attribute is forbidden if the attribute *seriesRef* is not specified or is empty and is mandatory if *seriesRef* is present and not empty.
`seriesRef`	IdType, max 50 char or empty string ("")	LYS12343	No/Yes for episode (Forbidden for content versions)	The ID of this episode's Series. If specified and not empty, number attribute must be filled. If the content is not within a Series or is a version of a content, this attribute must be omitted.
`expiryDate`	GMT date and time	2004-03-01T00:00:00Z	No (Used for DCM generation)	The expiry date of the content. If no expiry date is specified, the Media Live CMS import tool will supply a default value (constant) equal to the maximum expiration date (January 1st, 2038). For a main content, this date is adjusted such that it is always greater or equal the max expiry date of its versions.

Name	Type	Example	Mandatory	Description
`startDate`	GMT date and time	2004-03-01T00:00:00Z	No	The start date of the content. If no start date is specified, the Media Live CMS import tool will supply a default value equal to the date of the import. For a main content, this date is adjusted such that it is always less or equal the min start date of its versions.
`encProfileName`	String	TabletEncodingProfile	No	Name of an existing encoding or live profile. The Encoding profiles are linkable to production versions only not on the main productions.
`preProcessingProfileNames`	String	PivotFileGeneration	No	Names of an preprocessing profile. The PreProcessing profiles are linkable to the main productions only and not on production versions. Multiple profile names can be supplied by semi-colon separated list of names of PreProcessing profiles.
`preLoaded`	Boolean, default value: false	true	No	Boolean flag. If a content is preloaded (asset files preloaded on destination devices), this flag is set to 'true'.
`jobPriority`	JobPriorityEnumType	3	No	The priority level of the future jobs created for this content. Must be an integer value.
`adInsertionProfileName`	String		No	The name of the Ad Insertion Profile to link to the content (See § 0). If attribute's value is empty, the profile is unlinked. If the attribute is not present, nothing is changed.

The data provider provides information about available content. It includes the movie meta-data and basic information as approximate duration.

These elements are used to provide the details of a main content, also called Production in Media Live CMS.

The `duration` attribute provides the approximate duration of the content, specified in seconds. The minimum and maximum durations are configured in the Media Live CMS system.

When deleting a content (`action` = "delete") the Media Live CMS import tool will only examine the `id` attribute, and use that to identify the content data to be removed from the database. All child XML components (`<EpgDescription>`) in the file will be ignored.

The `expiryDate` attribute is optional, it determines when the production expires and can be purged. In the Media Live CMS GUI the corresponding field is 'Purge date'.

The `seriesRef` defines the link between the `<Content>` and the `<Series>`.

If the seriesRef attribute is an empty string and the content already exists and is an episode, the content will be unlinked from its series and will become a standalone content. In this case, all inherited EPG values will be copied into the standalone content.

If the `seriesRef` attribute is present and is not empty and the content already exists but is not an episode, the content becomes an episode of the specified series. In this case, EPGs of the content will override EPGs inherited from the series.

If the `seriesRef` attribute exists and is not empty and the content already exists and is an episode of another series, the content will be un-assigned from its old series and assigned to the specified one.

The different episodes (`<Content>` element) within the same series could have the same `number`. If a `textual episode number` is needed, this textual number will be filled in an EPGElement field.

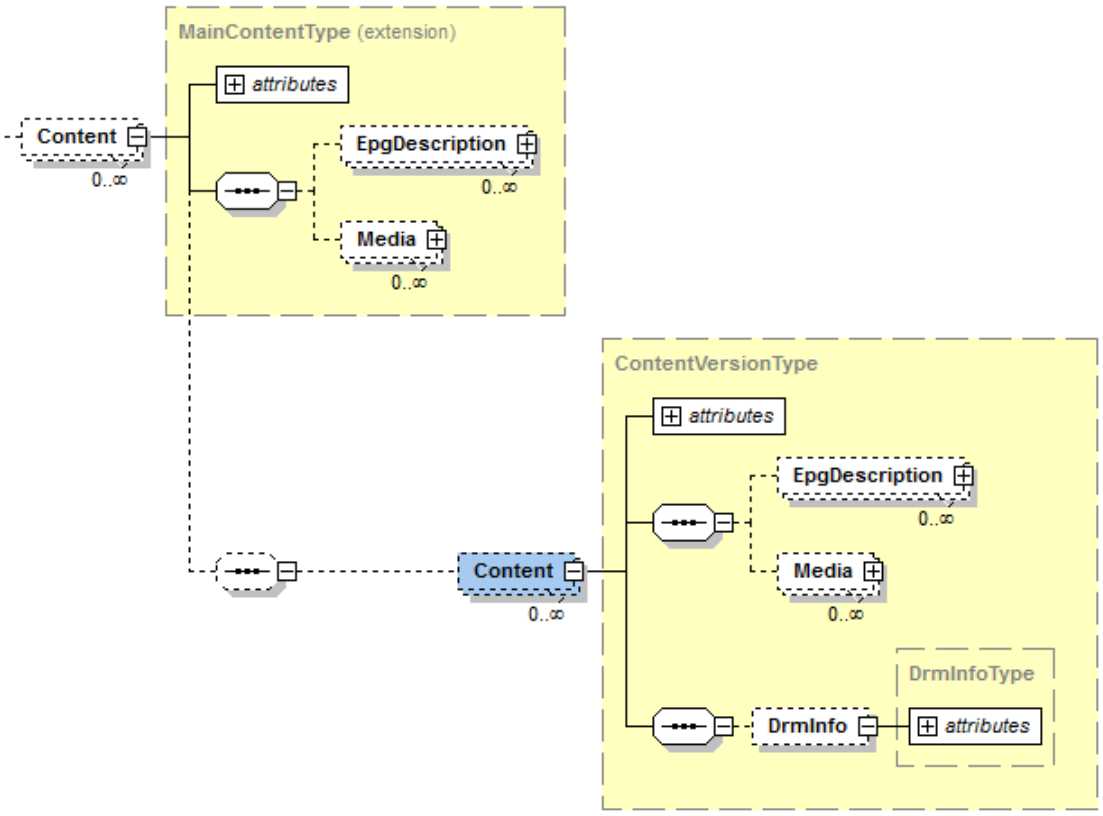
2.3.2.1 Main Content and Versions

Often a single item is not enough to clearly define content. For instance, for a same movie you might want to manage a high definition version and a standard version without duplicating the editorial (EPG) data. For the Production (main content), Media Live CMS can allow the operator to create multiple versions. To reflect this structure, the import format allows to add a `<Content>` inside another (but only on the first level) as represented in the following figure. This encapsulation is not represented in the ContentType definition to avoid a recursive encapsulation.

A `<Content>` that is created as a Content Version (inside another `<Content>`) cannot be re-imported outside the same Main Content. In the same idea, a Main Content cannot be re-imported inside another `<Content>`.

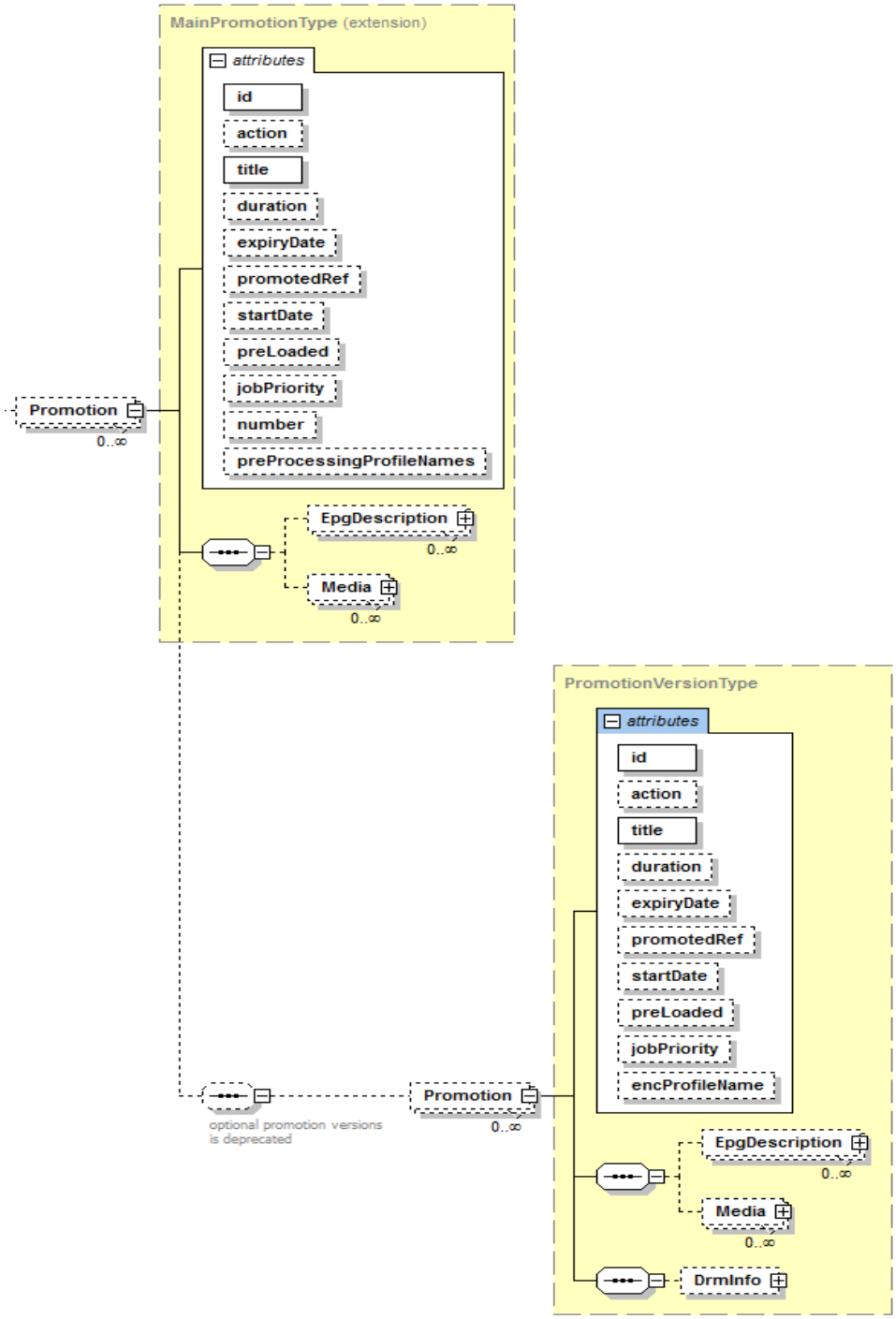
A `<Content>` that is created as a Content Version must not define any *seriesRef* or number attribute.

A `<Content>` that is created as a Content Version may have a `<DrmInfo>` (§2.3.4).



2.3.3 Promotion element

The `<Promotion>` elements within `<LysisData>` are defined by the type PromotionType:



Attributes details

Name	Type	Example	Mandatory	Description
`id`	IdType, max 50 char, unique for the provider domain	A12343	Yes	Unique identification of the promotion, see §2.2.2.1.
`action`	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information.
`title`	String, max 100 chars, UTF-8	Sunny Summer Promo	Yes	Promotion title, see §2.2.2.2.
`duration`	DurationType, default value : 0	120	No	Promotion approximate duration in seconds.
`expiryDate`	GMT date and time	2004-03-01T00:00:00Z	No	The expiry date of the promotion. If no expiry date is specified, the Media Live CMS import tool will supply a default value (constant) equal to the maximum expiration date (January 1st, 2038). For a main promotion, this date is adjusted such that it is always greater or equals the max expiry date of its versions.
`promotedRef`	IdType, max 50 char,	A123456	No	The ID of the promoted content.
`startDate`	GMT date and time	2004-03-01T00:00:00Z	No	The start date of the promotion. If no start date is specified, the Media Live CMS import tool will supply a default value equal to the date of the import. For a main promotion, this date is adjusted such that it is always less or equal the min start date of its versions.
`encProfileName`	String	TabletEncodingProfile	No	Name of an existing encoding or live profile. The Encoding profiles are linkable to promotion versions only not on the main promotions.

Name	Type	Example	Mandatory	Description
preProcessingProfileNames	String	PivotFileGeneration	No	Names of an preprocessing profile. The PreProcessing profiles are linkable to the main promotions only and not on promotion versions. Multiple profile names can be supplied by semi-colon separated list of names of PreProcessing profiles.
preLoaded	Boolean, default value: false	true	No	Boolean flag. If a promotion is preloaded (asset files preloaded on destination devices), this flag is set to 'true'.
jobPriority	JobPriorityEnumType	3	No	The priority level of the future jobs created for this content. Must be an integer value.

These elements are used to provide the details of promotional (trailer) content.

The duration attribute provides the approximate duration of the promotion, specified in seconds. The minimum and maximum durations are configured in the Media Live CMS system.

When deleting a promotion (action = "delete") the Media Live CMS import tool will only examine the id attribute, and use that to identify the promotion data to be removed from the database. All child XML components (<EpgDescription>) in the file will be ignored.

The expiryDate attribute is optional, it determines when the promotion expires and can be purged. In the Media Live CMS GUI the corresponding field is 'Purge date'.

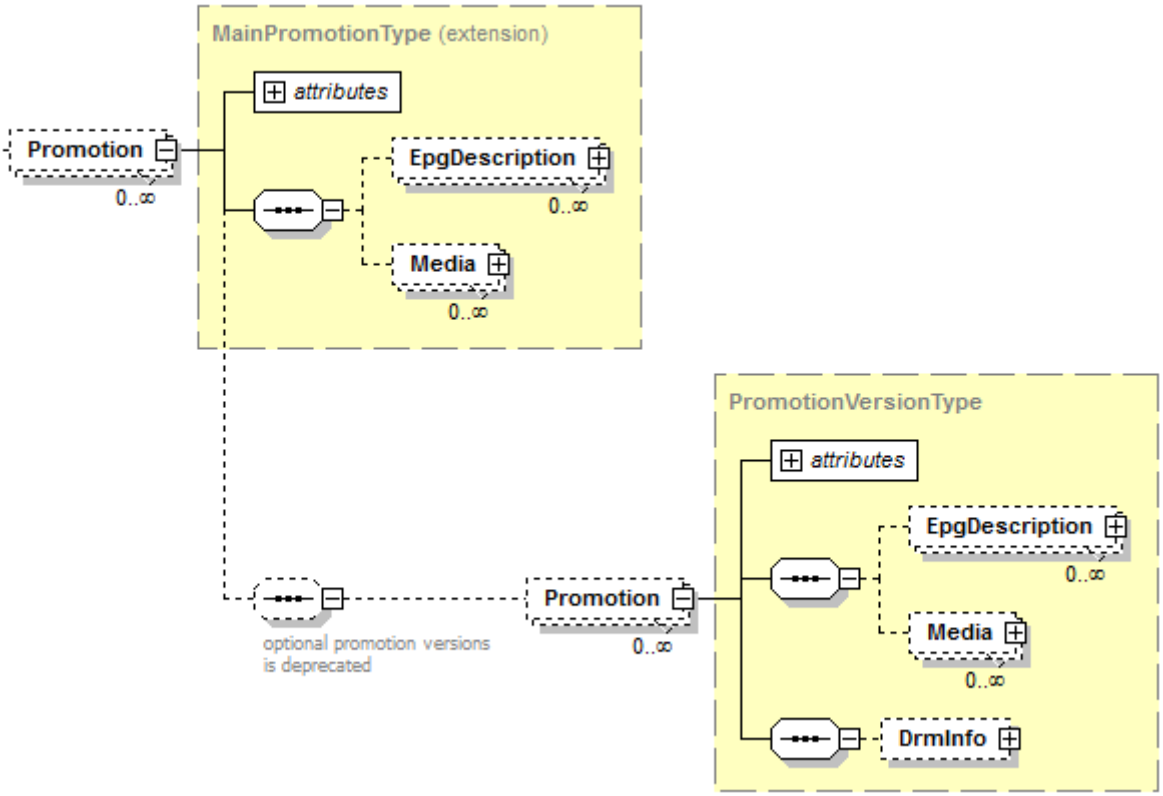
The promotedRef attribute is used to establish the link between the promotion and the content it promotes. The value of this field is the ID of a <Content> in the same file (but obviously not a deleted one) or a Production already existing in the Media Live CMS database. If the attribute is not present or its value is empty, the resulting promotion will be "stand-alone" in the Media Live CMS database after the import, specifically not linked to any Production. This means that for example, if an XML file is being supplied to just update a promotion's title, the ID of the promotion's main content must also be given in the file. If the promotion already exists in the Media Live CMS database and is linked to a different main content from the one specified in the XML file, the import processing will un-link it from the old main content, and link it to the newly specified main content.

2.3.3.1 Main Promotion and Versions

Often a single item is not enough to clearly define a promotion. For instance, for a same movie you might want to manage a high definition version and a standard version. For the Promotion (main promotion), Media Live CMS can allow the operator to create multiple versions. To reflect this structure, the import format allows to add a <Promotion> inside another (but only on the first level) as represented in the following figure. This encapsulation is not represented in the PromotionType definition to avoid a recursive encapsulation.

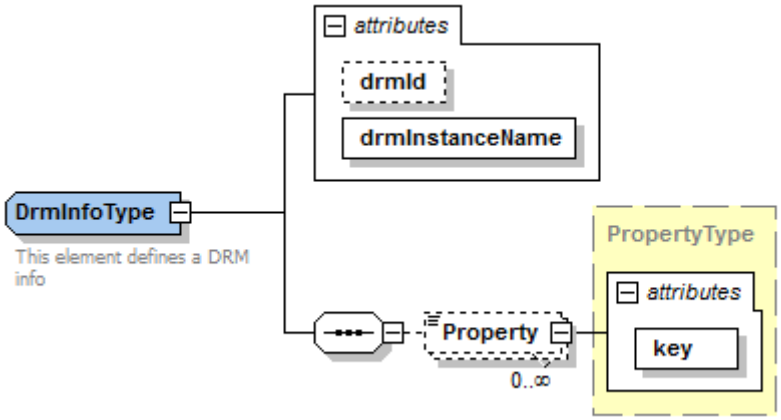
A <Promotion> that is created as a Promotion Version (inside another <Promotion>) cannot be re-imported outside the same Main Promotion. In the same idea, a Main Promotion cannot be re-imported inside another <Promotion>.

A <Promotion> that is created as a Promotion Version may have a <DrmInfo> (§2.3.4) .



2.3.4 DrmInfo element

The <DrmInfo> element within <Content> or <Promotion> is defined by the type DrmInfoType:



At each import, the <DrmInfo> information for a specific parent element will be imported. The drmId and drmInstanceName attributes are used to link a content version to a specific DRM system.

When a <DrmInfo> information has been set in a previous import, it can be updated or removed. All attributes can be updated by providing a new value. To delete a <DrmInfo> information, the drmInstanceName attribute must be provided with an empty value (in that case, drmId and retentionDuration attributes are ignored if provided).

If a <DrmInfo> information has been set in a previous import but has been omitted next, nothing is happening.

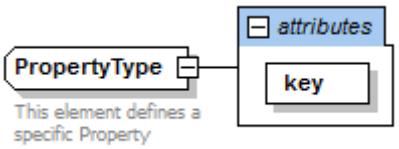
Attributes details

Name	Type	Example	Mandatory	Description
drmId	DrmIdType, max 256 chars	28	No	DRM information identifier.
drmInstanceName	String, max 100 chars	PRM_Nagra	Yes	Name of the DRM instance assigned to the content version.

This element provides a container in order to supply a set of metadata using the <Property> child element.

2.3.5 Property element

The `<Property>` elements within `<DrmInfo>` are defined by the type PropertyType:



This element provides metadata information related to the parent XML element using a key-value format. The value of the property corresponds to the element body.

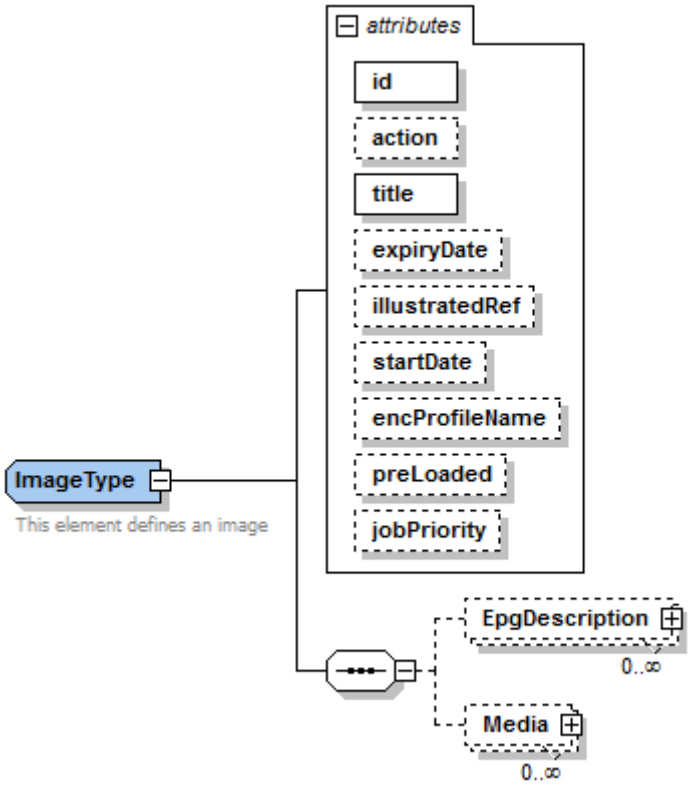
The tag may be empty, corresponding to a "null" value for the metadata, which will effectively delete the metadata value from the Media Live CMS database if it exists.

The list of allowed keys and their format are defined in [6].

The `key` attribute must exactly match one of the allowed metadata configured in the Media Live CMS database. Unknown metadata are considered as errors.

2.3.6 Image element

The <Image> elements within <LysisData> are defined by the type ImageType:



Attributes details

Name	Type	Example	Mandatory	Description
id	IdType, max 50 char, unique for the provider domain	A12343	Yes	Unique identification of the image, see §2.2.2.1.
action	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information.
title	String, max 100 chars, UTF-8	Sunny Summer Poster	Yes	Image title, see §2.2.2.2.

Name	Type	Example	Mandatory	Description
`expiryDate`	GMT date and time	2004-03-01T00:00:00Z	No	The expiry date of the image. If no expiry date is specified, the Media Live CMS import tool will supply a default value (constant) equal to the maximum expiration date (January 1st, 2038). For a main image, this date is adjusted such that it is always greater or equals the max expiry date of its versions.
`illustratedRef`	IdType, max 50 char,	A123456	No	The ID of the linked content.
`startDate`	GMT date and time	2004-03-01T00:00:00Z	No	The start date of the image. If no start date is specified, the Media Live CMS import tool will supply a default value equal to the date of the import. For a main image, this date is adjusted such that it is always less or equal the min start date of its versions.
`encProfileName`	String	TabletEncodingProfile	No	Name of an existing encoding or live or preprocessing profile. The PreProcessing profiles are linkable to the main images only and not on image versions.
`preLoaded`	Boolean, default value: false	true	No	Boolean flag. If an image is preloaded (asset files preloaded on destination devices), this flag is set to 'true'.
`jobPriority`	JobPriorityEnumType	3	No	The priority level of the future jobs created for this image. Must be an integer value.

These elements are used to provide the details of an image.

When deleting an image (`action` = "delete") the Media Live CMS import tool will only examine the `id` attribute, and use that to identify the image data to be removed from the database. All child XML components (`<EpgDescription>`) in the file will be ignored.

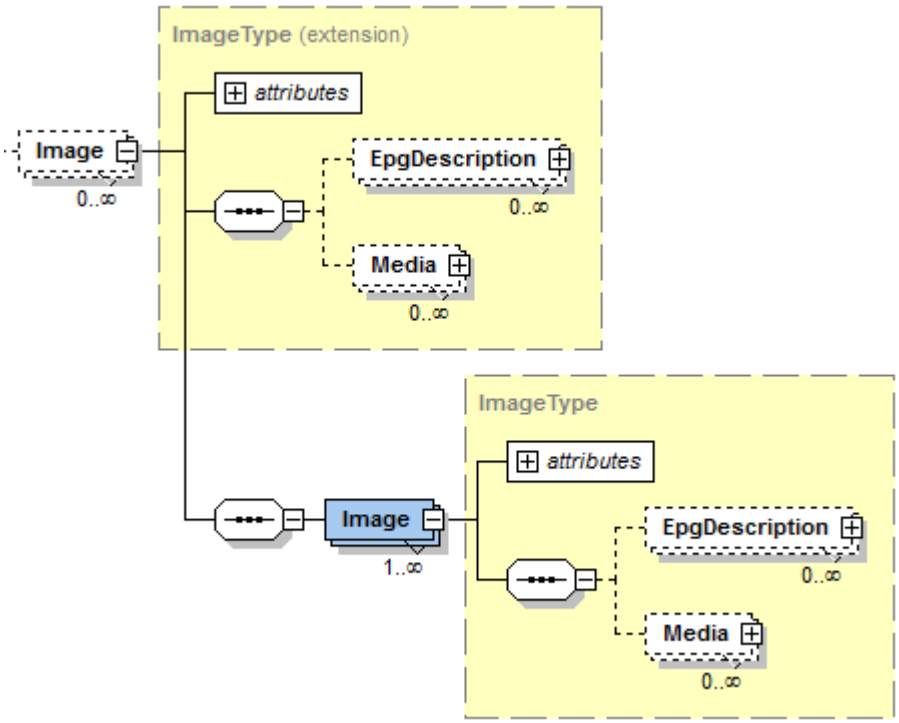
The `expiryDate` attribute is optional, it determines when the image expires and can be purged. In the Media Live CMS GUI the corresponding field is 'Purge date'.

The `illustratedRef` attribute is used to establish the link between the image and a series or main content. The value of this field is the ID of a `<Content>` in the same file (but obviously not a deleted one) or a Main Production/Main Promotion/Series already existing in the Media Live CMS database. If the attribute is not present or its value is empty, the resulting image will be "stand-alone" in the Media Live CMS database after the import, specifically not linked to any Production/Promotion/Series. This means that for example, if an XML file is being supplied to just update an image's title, the ID of the image's linked content/series must also be given in the file. If the image already exists in the Media Live CMS database and is linked to a different content/series from the one specified in the XML file, the import processing will un-link it from the old content/series, and link it to the newly specified content/series.

2.3.6.1 Main Image and Versions

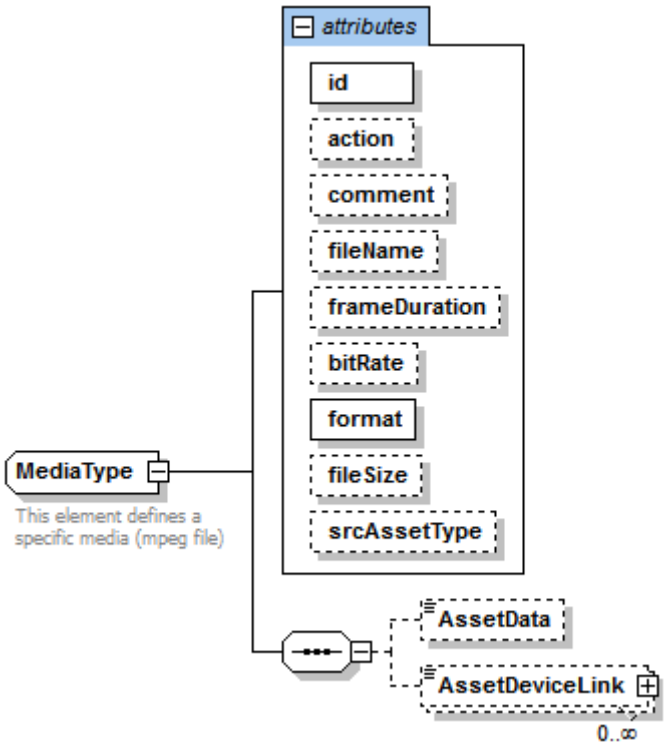
A single item is not enough to clearly define an image. For instance, for a same movie you might want to manage a high definition version and a standard version. For the Image (main image), Media Live CMS force the operator to create at least one image version. To reflect this structure, the import format ensures to have at least an `<Image>` inside the main one as represented in the following figure. This encapsulation is not represented in the ImageType definition to avoid a recursive encapsulation.

An `<Image>` that is created as an Image Version (inside another `<Image>`) cannot be re-imported outside the same Main Image. In the same idea, a Main Image cannot be re-imported inside another `<Image>`.



2.3.7 Media element

The <Media> elements within <Promotion>, <Image> or <Content> are defined by the type MediaType:



At each import, the <Media> information for a specific parent element will be imported. Including frame length and file name.

Attributes details

Name	Type	Example	Mandatory	Description
id	IdType, max 50 char, unique for the provider domain	A12343	Yes	Unique identification of the media, see §2.2.2.1.
action	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information.
comment	String, max 3000 char, unicodeUTF-8	Sunny summer media encoded the 2004-06-21	No	Operational comments of the media.
fileName	String, max 255 char, ascii	sunny_summer.mpg	No	Name of the file on the storage device.
frameDuration	DurationType, default value : 0	180000	No	The duration of the Media in frames.

Name	Type	Example	Mandatory	Description
bitRate	Integer	375000	No	The bit rate in bits/sec.
format	FormatEnum	AV_ClearTS	Yes	The format of the media used to indentify its type.
fileSize	Integer	1048576	No	The file size in bytes.
srcAssetType	SrcAssetType, list of value defined in mapping parameters	Clear_Asset_HD	Only for reference assets. Not used by other asset types	Value of SrcAssetType metadata (for Reference asset only)

AssetData is a CDATA object encoded in base64.

In order to specify asset type (HD, SD…) an attribute'srcAssetType' has to be present on media element which are reference assets

Concerning the location of the asset, attributes 'location' and 'storageName' have to be used together on a media element. If one is present, the other is mandatory. A default value is configurable for these two parameters in "Import VOD content provider 2.0" algorithm parameter set.

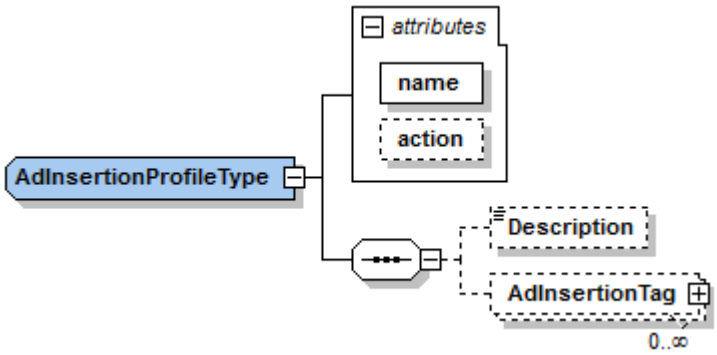
2.3.8 AdInsertionProfile elements (Ad Signalling)

The <AdInsertionProfile> and <AdInsertionTag> sub-elements are part of the Ad Signalling feature introduced with CMS4.4.

2.3.8.1 AdInsertionProfile element

An Ad Insertion Profile holds a list of Ad Insertion Points (see § 2.3.8.3). This list is considered as a snapshot (that means all new tags are added, all no more present tags are removed, all updated tags are overidden).

The <AdInsertionProfile> is defined by the type AdInsertionProfileType:



Attributes details

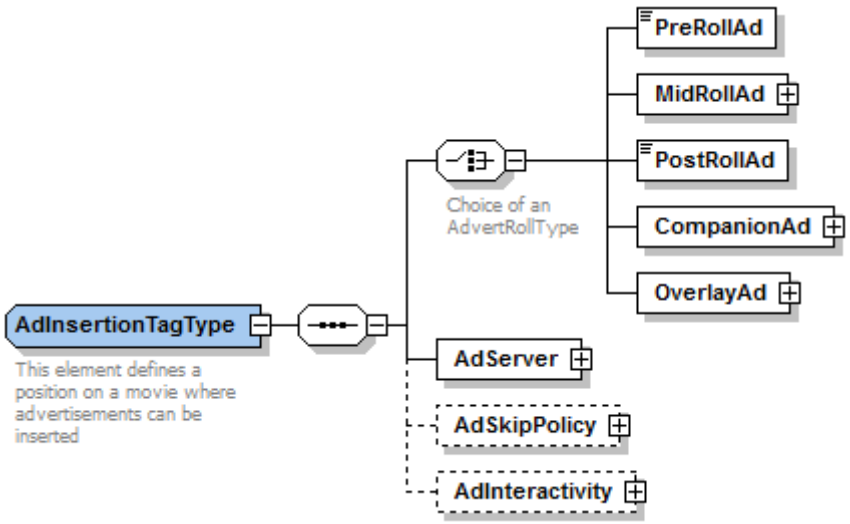
Name	Type	Example	Mandatory	Description
`name`	String, max 50 char, unicode UTF-8		Yes	Unique name of the profile.
`action`	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information.

2.3.8.2 Description element

Textual description of the profile. This optional element has a String value, max 3000 char, unicode UTF-8.

2.3.8.3 AdInsertionTag element

An Ad Insertion Tag is used to describe a position on a movie where advertisements can be inserted. The `<AdInsertionTag>` is defined by the type AdInsertionTagType:



2.3.8.4 PreRollAd and PostRollAd Elements

These elements represent two distinct possibilities among all types of advertisement roll defined by the <AdInsertionTag> element.

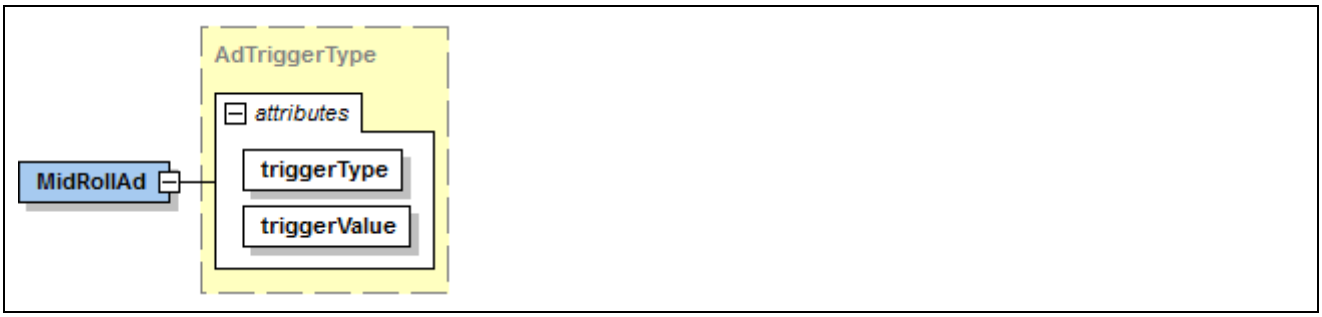
PreRollAd	PostRollAd

Attributes details

These elements define no attributes.

2.3.8.5 MidRollAd element (AdTriggerType)

This element represents one possibility among all types of advertisement roll defined by the <AdInsertionTag> element.

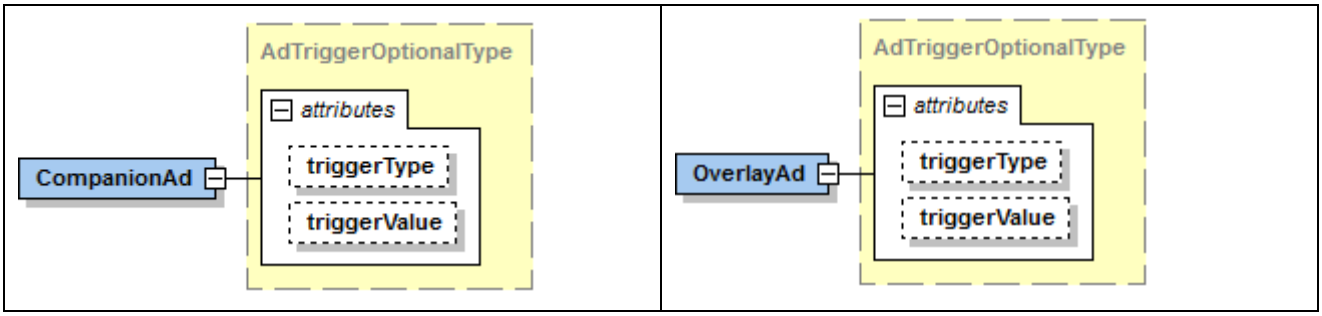


Attributes details

Name	Type	Description	Examples	
triggerType	AdTriggerEnum (§ 2.3.8.10)	Type of trigger.	after-sec	after-percentage
triggerValue	Integer, max is 86400 (after-sec) or 100 (after-percentage)	Time indication as when the advert is started. The value is either expressed in seconds or in percentage.	600	60

2.3.8.6 CompanionAd and OverlayAd Elements

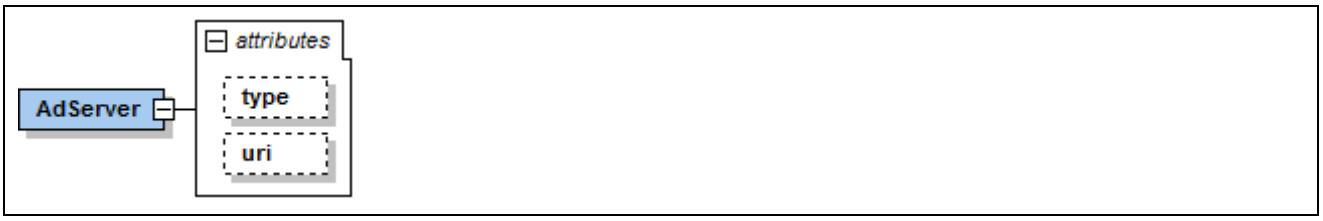
These elements represent two distinct possibilities among all types of advertisement roll defined by the <AdInsertionTag> element.



Attributes details

See "Attributes details" in § 2.3.8.5.

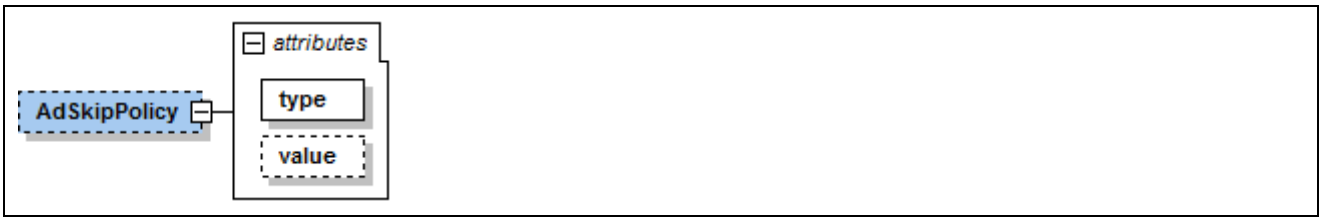
2.3.8.7 AdServer Element



Attributes details

Name	Type	Description	Examples
type	AdServerEnum (§ 2.3.8.11)	Type of AdServer (internal, external).	internal
uri	String	AdServer URI	http://AdServer.com/requestAd.asp

2.3.8.8 AdSkipPolicy Element



Attributes details

Name	Type	Description	Examples
type	AdSkipPolicyEnum (§2.3.8.12)	(skip-after-sec, no-skip)	skip-after-sec
value	Integer	Value in seconds. Valid with type 'skip-after-sec'	30

2.3.8.9 AdInteractivity Element

Attributes details

Name	Type	Description	Examples
type	AdInteractivityEnum (§ 2.3.8.13)	(interactive, not-interactive, any)	interactive

2.3.8.10 AdTriggerEnum simpleType

```
<xs:simpleType name="AdTriggerEnum">
	<xs:annotation>
		<xs:documentation>Enumerates the types of AdTrigger.</xs:documentation>
	</xs:annotation>
	<xs:restriction base="xs:string">
		<xs:enumeration value="after-sec"/>
		<xs:enumeration value="after-percentage"/>
	</xs:restriction>
</xs:simpleType>
```

2.3.8.11 AdServerEnum simpleType

```
<xs:simpleType name="AdServerEnum">
	<xs:annotation>
		<xs:documentation>Enumerates the types of AdServer.</xs:documentation>
	</xs:annotation>
	<xs:restriction base="xs:string">
		<xs:enumeration value="internal"/>
		<xs:enumeration value="external"/>
	</xs:restriction>
</xs:simpleType>
```

2.3.8.12 AdSkipPolicyEnum simpleType

```
<xs:simpleType name="AdSkipPolicyEnum">
	<xs:annotation>
		<xs:documentation>Enumerates the types of AdSkipPolicy.</xs:documentation>
	</xs:annotation>
	<xs:restriction base="xs:string">
		<xs:enumeration value="skip-after-sec"/>
		<xs:enumeration value="no-skip"/>
	</xs:restriction>
</xs:simpleType>
```

2.3.8.13 AdInteractivityEnum simpleType

```
<xs:simpleType name="AdInteractivityEnum">
	<xs:annotation>
		<xs:documentation>Enumerates the types of AdInteractivity.</xs:documentation>
	</xs:annotation>
	<xs:restriction base="xs:string">
		<xs:enumeration value="interactive"/>
		<xs:enumeration value="not-interactive"/>
		<xs:enumeration value="any"/>
	</xs:restriction>
</xs:simpleType>
```

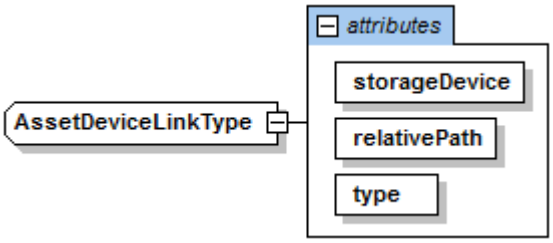
2.3.9 FormatEnum simpleType

source

```
<xs:simpleType name="FormatEnum">
    <xs:annotation>
        <xs:documentation>Type of media format.</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="AV_ClearTS"/>
        <xs:enumeration value="AV_EncryptedTS"/>
        <xs:enumeration value="AV_PlaylistName"/>
        <xs:enumeration value="AV_HarmonicOSPlaylistName"/>
        <xs:enumeration value="Data_PMT"/>
        <xs:enumeration value="AV_Dummy"/>
        <xs:enumeration value="Image_Basic"/>
        <xs:enumeration value="Data_Index"/>
    </xs:restriction>
</xs:simpleType>
```

2.3.10 AssetDeviceLink element

The `<AssetDeviceLink>` elements within <Media> are defined by the type AssetDeviceLinkType:



Name	Type	Example	Mandatory	Description
`storageDevice`	Device	CIS	Yes	Name of the device where the asset is stored. Device must existe withing Media Live CMS.
`relativePath`	String	AssetFolder/	Yes	Relative path of the asset. Asset full path consists of the concatenation of an access point path and this relative path.
`type`	ADLEnumType	Source	Yes	Type of the asset device link (ADLEnumType).

If an <AssetDeviceLink> with the same type is already present on the asset, it will be updated. Otherwise, a new entry will be created.

No ADL will be deleted during import process.

2.3.11 ADLEnumType simpleType

source

```
<xs:simpleType name="ADLEnumType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Source"/>
    <xs:enumeration value="Destination"/>
    <xs:enumeration value="Archive"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>
```

2.3.12 JobPriorityEnumType simpleType

source

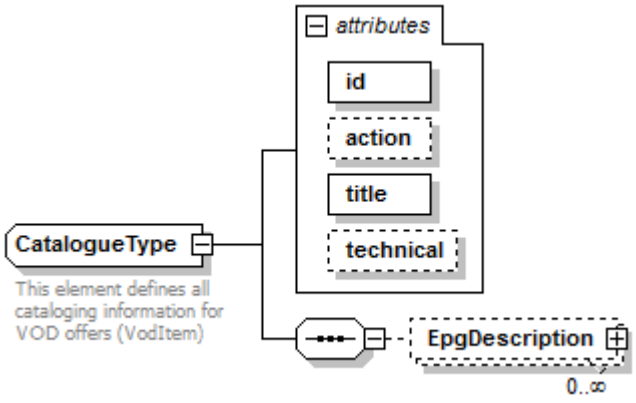
```
<xs:simpleType name="JobPriorityEnumType">
  <xs:annotation>
    <xs:documentation>Priority level enumeration</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:integer">
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
    <xs:enumeration value="4"/>
    <xs:enumeration value="5"/>
  </xs:restriction>
</xs:simpleType>
```

2.4 VOD data

This chapter will regroup the <Catalogue>, <DefineNode> and <VodItem> elements and their sub-elements. All these elements participate to the VOD data.

2.4.1 Catalogue element

The <Catalogue> elements within <LysisData> are defined by the type CatalogueType:



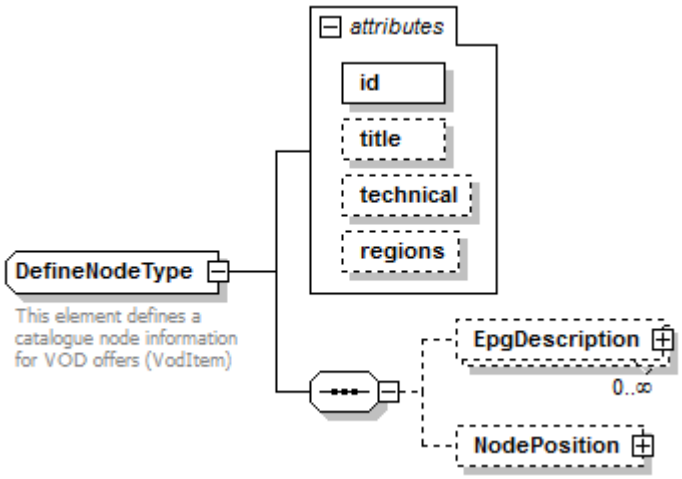
Attributes details

Name	Type	Example	Mandatory	Description
id	IdType, max 50 char, unique for the provider domain	A12343	Yes	Unique identification of the catalogue, see §2.2.2.1.
action	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information.
title	String, max 100 chars, UTF-8	VOD	Yes	Catalogue title, see §2.2.2.2.
technical	Boolean	0	No	0 if the catalogue is a browsable catalogue, 1 if the catalogue is technical and should not be displayed as such.

The catalogue is the top level of the heiarchical structure for browsing VOD content. The internal node structure will be built using <DefineNode> elements.

2.4.2 DefineNode element

The `<DefineNode>` elements within `<LysisData>` are defined by the type NodeType:



Attributes details

Name	Type	Example	Mandatory	Description
`id`	IdType, max 50 char	A12343	Yes	Unique identification of the node, see §2.2.2.1.
`title`	String, max 100 chars, UTF-8	Africa	Yes	Node title, see §2.2.2.2.
`technical`	Boolean	0	No	0 if the node is a browsable tree node, 1 if the node is technical and should not be displayed as such.
`regions`	String, list of semi-colon separated values	France;Belgium	No	A list of regions where the node is available (semi-colon separated values)

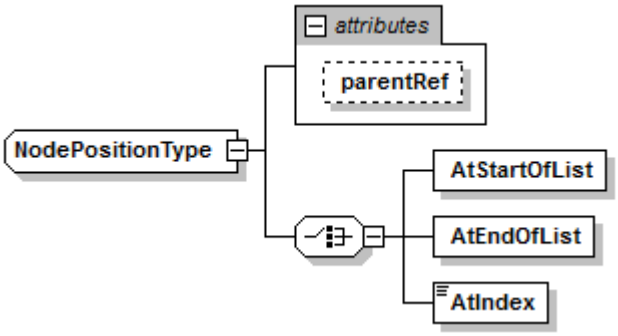
These elements are used to construct the heirachical structure of nodes within the catalogue for content browsing.

When creating a new node it is mandatory to supply a value for the `title` attribute, a nested `<NodePosition>` element and the EPG fields that are specified as mandatory for nodes in reference document [6].

This element can to used to selectively update fields of existing nodes. When updating an existing node, for fields that are not explicitly included in the xml file, the values already existing the in the CMS database will be preserved. For example, to just update one of the EPG fields of an existing node, just the node's ID and the corresponging nested `<EpgDescription>` + `<EpgElement>` need to be provided. Similarly, an existing node may be moved within its catalogue by just providing the node's ID and child `<NodePosition>` giving the new location.

2.4.3 NodePosition element

The <NodePosition> elements within <DefineNode> are defined by the type NodePositionType:



Attributes details

Name	Type	Example	Mandatory	Description
parentRef	IdType, max 50 char	A12343	No	Unique identification of the parent node, see §2.2.2.1.

The <NodePosition> element has one attribute parentRef which will contain the ID of a catalogue or node already existing in the CMS database or defined earlier in the same xml file. So the node is placed as the child of an already existing catalogue or node.

When creating a new node it is mandatory to provide a non-empty parentRef attribute value.

With this method of positioning, new nodes can be added within the hierarchical structure by specifiying just the direct parent, there is no need to provide the entire hierarchical path. In addition, if an existing node is to be re-positioned within the same parent, i.e. just a re-ordering of child nodes, parentRef may be omitted.

Child elements <AtStartOfList> and <AtEndOfList> have no attributes and no payload.

Child element <AtIndex> has no attributes. Its payload is a non-negative integer.

The order of sibling nodes within the same parent (the catalogue for first level nodes, or a node for those deeper within the structure) is recorded in the CMS database. This ingest interface provides three choices to specify the position of a node:

<AtStartOfList> The node is positioned as the first child, i.e. before all other child nodes within the same parent.

<AtEndOfList> The node is positioned as the last child, i.e. after all other child nodes within the same parent.

<AtIndex> The node is positioned between two existing child nodes, immediately before the one that is currently at the n^{th} position in the list, where n is this element's payload value. Position indexes are counted starting from 0.

Within the same xml file different options may be mixed, so different nodes may be positioned by different choices.

It is possible to use the choices `<AtStartOfList>` and `<AtEndOfList>` multiple times for nodes within the same parent within the same xml file. In this case the order of their enclosing <DefineNode> elements in the xml file determines the final order of the nodes, together with whether choice `<AtIndex>` is also used. To keep positioning simple (and easily predictable), it is advised not to mix start/end list placement with index based, and in addition not to use `<AtStartOfList>` multiple times.

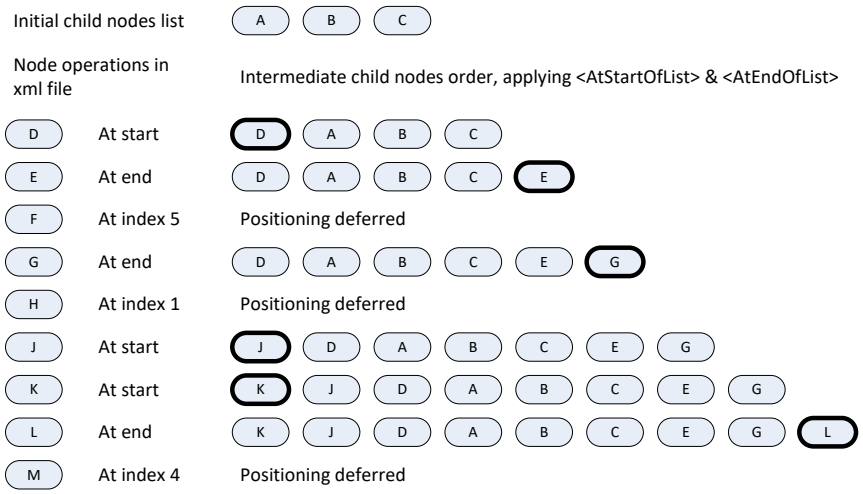
The following logic applies for placing the child nodes within a parent:

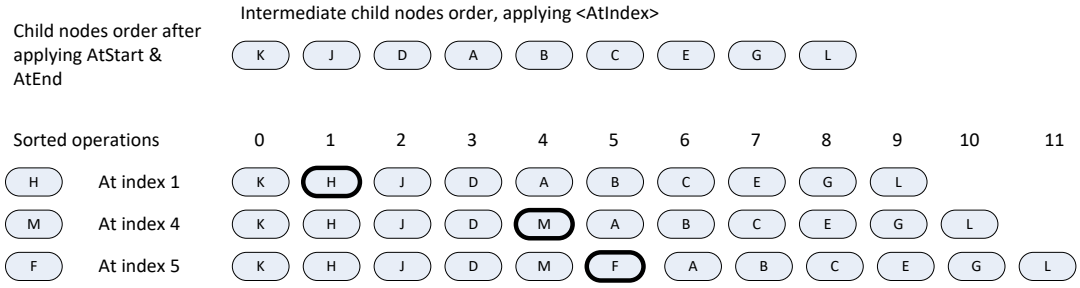
As the source xml is processed sequentially, each `<AtStartOfList> and <AtEndOfList>` items are encountered, the nodes are positioned accordingly.

The <AtIndex> elements are retained in a separate list but not applied immediately. Once the whole xml file has been read, the positioning of these elements is done. For each parent node the child nodes to be positioned are processed by increasing sorted order of the index values (i.e. not necessarily the order the elements occurred in the XML). Notice that existing nodes get "pushed" down the list as new nodes are added.

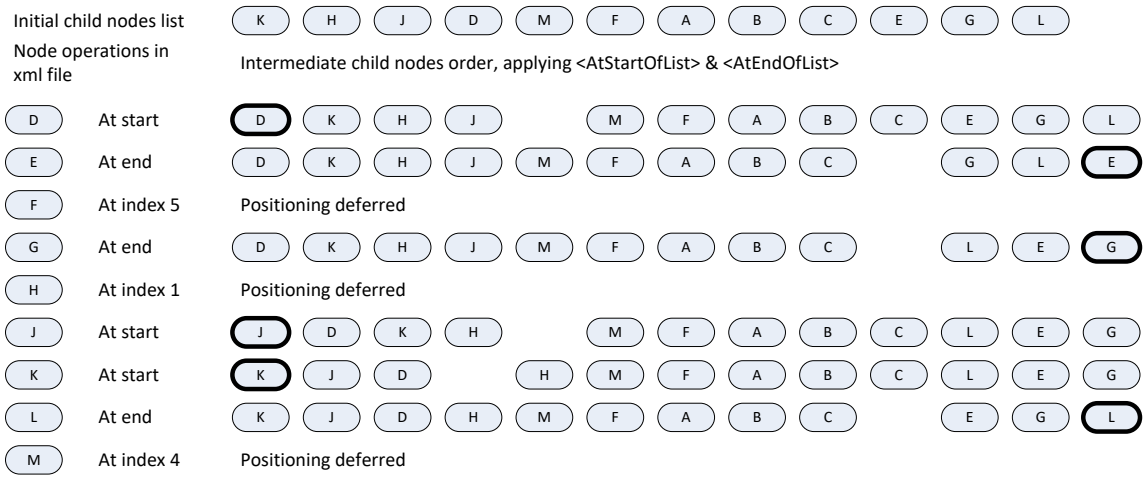
First the `<AtStartOfList>` nodes are processed in order and are successively positioned before any existing nodes.

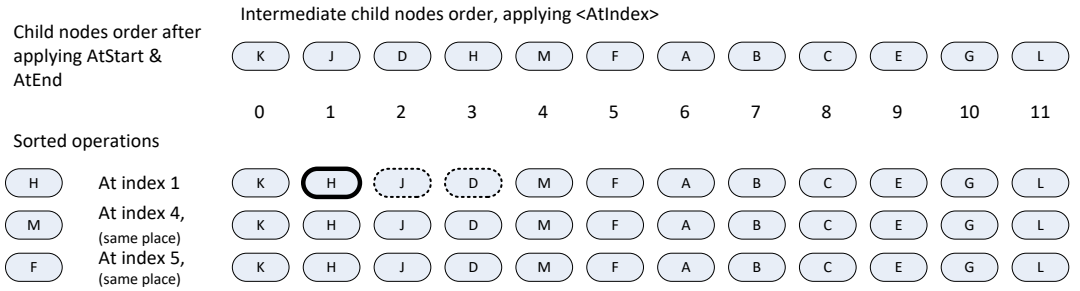
The diagrams below illustrate this process.





The defined logic also means that if the same xml file is re-ingested a second time, although there may be some internal shuffelling of the elements the final result is the same. As shown in the diagrams below for the same input sequence as above.





If an index value is specified that is greater than the number of child nodes of the parent, at the time that node is added, the node get placed at the end of the list of children.

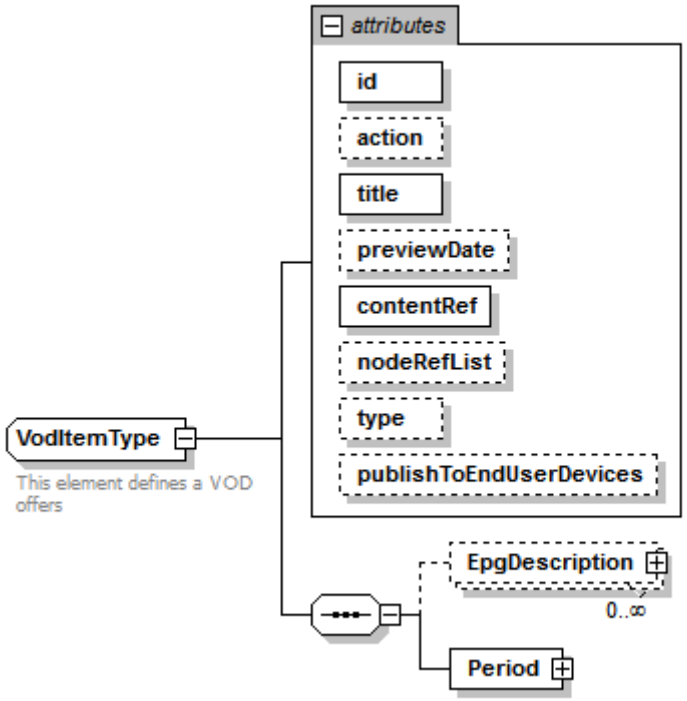
The above examples show the behavior when adding new nodes. It is also possible to move existing nodes. For an existing node if its current location is different to that specified via the `<NodePosition>`, it is removed from its old location and placed at the new one.

Note:

The index values are only used during the ingest process, they are not persisted in the CMS database.

2.4.4 VodItem element

The `<VodItem>` elements within `<LysisData>` are defined by the type VodItemType:



Each VOD offer describes the availability of content (Production, Promotion or Image) during a period of time. These elements keep a reference to its associated content.

Attributes details

Name	Type	Example	Mandatory	Description
`id`	IdType, max 50 char, unique for the provider domain	A234554	Yes	Unique id of the VOD item, see §2.2.2.1.
`action`	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information.
`title`	String, max 100 chars, UTF-8	The Matrix april offer.	Yes	Technical identification of the VOD Item, see §2.2.2.2.

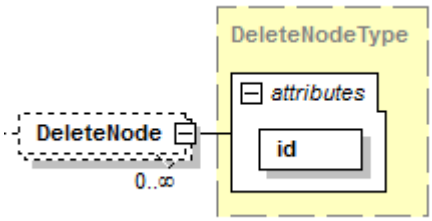
Name	Type	Example	Mandatory	Description
previewDate	GMT Date and time	2004-12-12T06:00:00Z	No	The date and time when the offer becomes available for preview. If the preview date is not specified it will be set to the start date of the VOD item.
contentRef	IdType, max 50 char	LYS000000124	Yes	Refers Content/Promotion/Image through Content/Promotion/Image.ID. This allows many VOD item to refer to same Content/Promotion/Image.
nodeRefList	String, list of semi-colon separated ids, each id must conforme to rules defined for id attribute	A12234;A12235;A12236	No	The list of Nodes in which this offer will be displayed (semi-colon separated values of Nodes id) **(+).** The nodes in the list will replace the existing nodes the vod item is currently linked to. This list cannot be empty.
type	List of Strings separated by semicolons	catchup;truevod	No	The semicolon separated list could contains any combinaison of the following values : catchup, truevod, pushvod and pullvod
publishToEndUserDevices	boolean	true	No	"true" if the VOD item must be published to the end user devices, "false" otherwise. If there isn't any value provided, then it will be imported as "false".

The contentRef defines the link between the <VodItem> and the <Content>. It is important to undertand that the <Content> linked to <VodItem> cannot be changed when defined. An update of a <VodItem> will be refused if the contentRef does not refer to the same <Content> as the previous import.

If there is no nodeRefList, vod items will be placed in nodes configured in the APS ("DefaultNodeIds"), the link between the vod item and other nodes already existing in the database will not be removed.

2.4.5 DeleteNode element

The <DeleteNode> elements within <LysisData> are defined by the type DeleteNodeType:



Attributes details

Name	Type	Example	Mandatory	Description
id	IdType, max 50 char, unique for the provider domain	A123445	Yes	Unique identification of the node, see §2.2.2.1.

Note

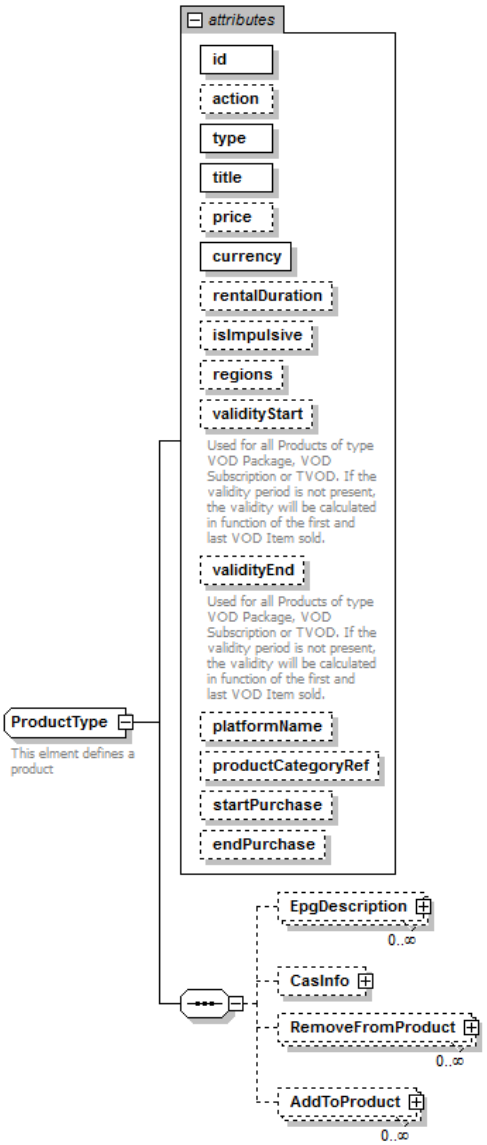
When<DeleteNode> elements are provided in an import file, all the Nodes with those ids provided as part of <DeleteNode> elements will be deleted.

2.5 Product data

This chapter will regroup the `<Product>` element and its sub-elements. All these elements participate to the Product data.

2.5.1 Product element

The `<Product>` elements within `<LysisData>` are defined by the type ProductType:



Attributes details

Name	Type	Example	Mandatory	Description
id	IdType, max 50 char, unique for the provider domain	A234554	Yes	Unique identifier of the product, see §2.2.2.1.
action	ActionEnumType, default value : override	override	No	Action to apply to the element, see §2.2.5 for more information.
type	(single, subscription, multiple,EST)	single	Yes	single product references only one element that is sold (MOD), subscription references a set of VOD offers based on a monthly fee (SVOD), multiple refers to packages made out of several VOD offers (MOD) , EST references one element that for consumption whilst that device is offline (local playback). See §2.5.1.1 for the values.
title	String, 100 chars, UTF-8.	Star Wars	Yes	Working title of the product, see §2.2.2.2.
price	Price (xs:decimal), default value : 1, min value: 0.00,	1.99	No	A non-negative numeric value. If the attribute "price" and "ProductCategoryRef" are provided at the same time, the product direct price will override the product category price definition
currency	Currency (xs:string)	CHF	No	The currency of the price. 3 letter, upper case, currency code from ISO [2]. Allowed individual values are listed in the CMS mapping table TYPES.CurrencyCodeISO and this configuration can be adapted on-site.
rentalDuration	DurationHourMinuteInSeconds, default value : 0	48	No	The max rental allowed in seconds.
isImpulsive	Boolean, default value: false	true	No	Attribute that indicate if the product is impulsive. Default value is false.

Name	Type	Example	Mandatory	Description
regions	GMT Date and time	Belgium;France; Luxembourg	No	A list of regions where the product is available (semi-colon separated values).. An empty value means the product is targeted to all regions, without having to list them.
validityStart	GMT Date and time	2002-02-25T04:30:00Z	No	Used for all Products of type VOD Package, VOD Subscription or TVOD. If the validity period is not present, the validity will be calculated in function of the first and last VOD Item sold.
validityEnd	GMT Date and time	2002-02-25T04:30:00Z	No	Used for all Products of type VOD Package, VOD Subscription or TVOD. If the validity period is not present, the validity will be calculated in function of the first and last VOD Item sold.
platformName				The name of the existing platform.The referenced platform will replace the existing platform the product is currently linked to. If the attribute is empty, the product will be unlinked from its platform.
productCategoryRef	xs:string	LYS12345	No	Public Id of the product category. If specified, and the product "price" attribute is not specified, for a new product, the product price will be the one of the product category. If specified, and the product "price" attribute is not specified, for an old product, the product price will not be the one of the product category. If specified, and the product "price" attribute is specified, the product price will not be the one of the product category.
startPurchase	GMT Date and time	2002-02-25T04:30:00Z	No	Used only for EST products. Start of the purchase period.

Name	Type	Example	Mandatory	Description
endPurchase	GMT Date and time	2002-02-25T04:30:00Z	No	Used only for EST products. End of the purchase period.

The <Product> element allows the definition of different VOD offers. It includes information such as price, currency and rental duration and optional regions.

The title is the working title and not the billing title. The billing title will be displayed in the end user and in this case will be part of the <EpgDescription> related to the <Product>.

Attributes validityStart and validityEnd have to be used in pairs, either both present or absent.

Child <CasInfo> element (see §2.5.2) may be included inside <Product> to link it with a CAS security device.

2.5.1.1 SimpleType ProductEnumType

This type is an enumeration of string that can have the following values:

- subscription
- single
- multiple
- EST

This type is used as an attribute that defines the type of product.

2.5.1.2 Single product (TVOD)

Single products represent a simple VOD PPV product. It must be linked to one and only one VodItem. The creation of a single product is done as following:

Example

```
<Product id="XXX200000001" action="override" currency="EUR" price="2.0"
rentalDuration="86400" type="single">
      <AddToProduct elementKind="VodItem" elementId="XXX100000001"></AddToProduct>
</Product>
```

The VodItem with ID XXX100000001 can be define in the same XML file or in a previous imported XML file (In fact VodItem XXX100000001 must exist in Media Live CMS)

2.5.1.3 Electronic Sell Through (EST)

EST producs must be linked to one and only one VodItem. They have an infinite validity period (validity dates are null), no rental duration (also set to null) and no Cas Info. The creation of an EST product is done as following:

Example

```
<Product id="XXX200000001" action="override" currency="EUR" price="2.0" endPurchase="2023-12-
18T08:00:00Z" startPurchase="2010-05-21T06:00:00Z" type="EST">
      <AddToProduct elementKind="VodItem" elementId="XXX100000001"></AddToProduct>
```

```
</Product>
```

2.5.1.4 Multiple product

Multiple products represent a package product. It must be linked to one or multiple VodItem. The definition of a multiple product can be done in the same or in multiple XML files:

Example

In the first file:

```
<Product id="XXX200000002" action="override" currency="EUR" price="10.0"
rentalDuration="172800"   type="multiple">
      <AddToProduct elementKind="VodItem" elementId="XXX100000001"/>
      <AddToProduct elementKind="VodItem" elementId="XXX100000002"/>
</Product>
```

And in another file:

```
<Product id="XXX200000002" action="override" currency="EUR" price="10.0"
rentalDuration="172800" type="multiple">
      <AddToProduct elementKind="VodItem" elementId="XXX100000003"/>
</Product>
```

You can see that you must use the action override when you want to add a new VodItem in an existing multiple product. You can also use `<RemoveFromProduct>` to remove a VodItem from the product.

2.5.1.5 Subscription product (SVOD)

Subscription products are very similar to multiple products. You can add or remove VodItem/Node exactly as you do for multiple products.

2.5.2 Element <CasInfo>

The element <CasInfo> within <Product> is defined by the type CasInfoType:

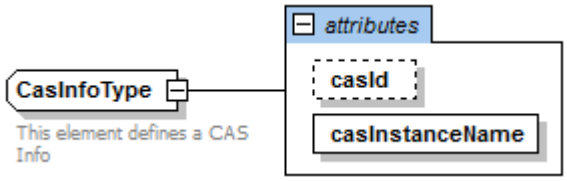


Table 1 <CasInfo> attributes details		
Name	**Type**	**Description and typical value**
casId	xs:string optional	The identifier of this product within the CAS. This is the technicall ID for all conditional access related operations. Max length 50 characters.
casInstanceName	xs:string mandatory	The name that uniquely identifies the CAS server managing access to the content sold by this product. The corresponding CAS security device must already

		exist in the CMS database. Will be unique amongst all devices. May not start or end with whitespace. Maximum length is 300 characters.

The **casInstanceName** attribute must match the corresponding CMS CAS security device object. Synchronisation of the configuration of these names is out of the scope of this interface.

A configuration parameter of the CMS import algorithm defines the names of the devices the data provider is authorized to use. Attempts to create, update or delete products that are attached to a CAS device that the data provider is not authorized to use will raise an error and cause the import to fail. The access restriction does not apply to OTT products so all data providers have full access to all OTT products.

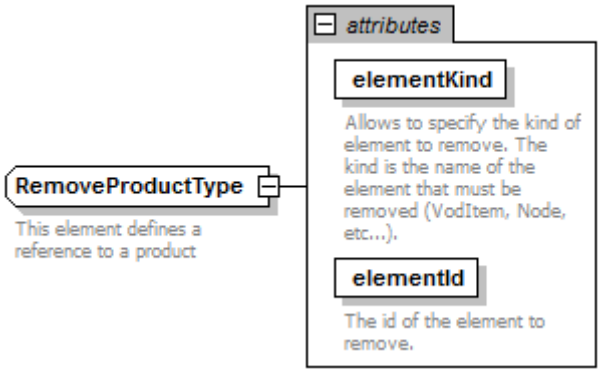
If the configuration parameter of the CMS import algorithm doesn't defines the names of the devices the data provider is authorized to use (is empty) then the access restriction does not apply.

When a <CasInfo> information has been set in a previous import, it can be updated or removed. All attributes can be updated by providing a new value. To delete a <CasInfo> information, the casInstanceName attribute must be provided with an empty value (in that case, casId attribute is ignored if provided).

The **casID** value should be unique within the CAS/DRM security devices, but this rule does not apply to HybridCASDevices.

2.5.3 RemoveFromProduct element

The `<RemoveFromProduct>` elements within `<Product>` are defined by the type RemoveProductType:



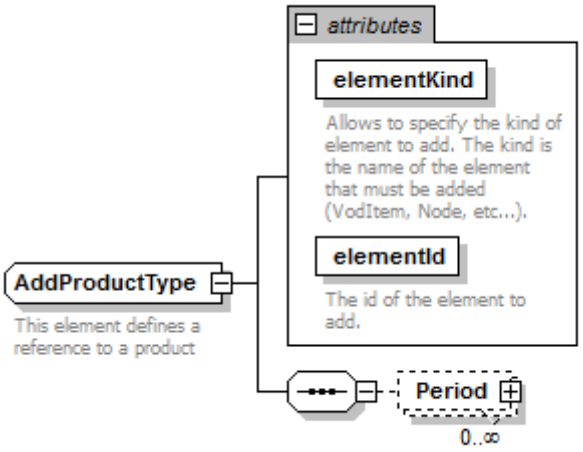
Attributes details

Name	Type	Example	Mandatory	Description
`elementKind`	VodItem, Node	VodItem	Yes	Define the element to remove from the product. For VOD import only VodItem and Node are allowed.
`elementId`	IdType, max	A234554	Yes	The ID of element to remove.

	50 char			

2.5.4 AddToProduct element

The `<AddToProduct>` elements within `<Product>` are defined by the type AddProductType:



Attributes details

Name	Type	Example	Mandatory	Description
`elementKind`	VodItem, Node	VodItem	Yes	Define the element to add in the product. For VOD import only VodItem and Node are allowed.
`elementId`	IdType, max 50 char	A234554	Yes	The ID of element to add.

Note

The `<AddToProduct>` is provided only when a new item must be added to the product. For a single product it is just at the creation, but this point is important for the subscription and package products. For instance we import a package (multiple products) with two items in it. The next day the product is updated to add a third item, this update will contain only one `<AddToProduct>` with the new item.

The elementId attribute can never be the id of a VodItem referring an Image, because a VodItem referring an Image can't be sold. In other terms, we cannot add a VodItem referring an image to a Product.

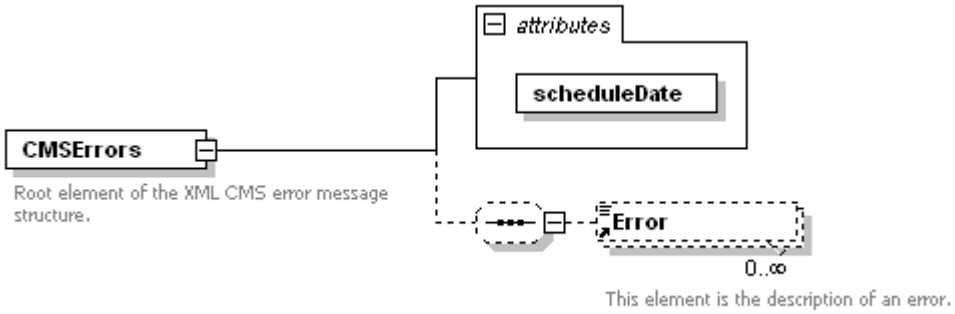
The associated period elements allow defining a set of availability periods of the specified VodItem/Node for the product.

The Period element cannot be defined on a single TVOD product, the import algorithm will finish in error.

3. Error file format

In the case where the CMS import process detects an error causing it to reject the input xml file an error file with the same name as the source file and an ".error" extension will be generated and written in the acknowledgement failure directory.

The error file is an XML file with the following structure



<CMSErrors> attributes details		
Name	**Type**	**Description and typical value**
scheduleDate	gmtdatetime Mandatory	See 2.2.3 Provides a reference date for the error XML file. Will be the date and time that the CMS generates the file. Eg "2011-08-04T08:15:58Z"

The child <Error> elements have type "xs:string", with undefined maximum length. Multiple <Error> elements may be supplied, one for each error detected.

3.1 Error file sample

In case of problems, the response should include a description in an xml format (content-type: text/xml).

```
<?xml version="1.0" encoding="UTF-8"?>
<CMSErrors scheduleDate="2004-07-01T22:00:00Z">
      <Error>Error description 1</Error>
      <Error>Error description 2</Error>
      <Error>Error description 3</Error>
</CMSErrors>
```

3.2 Error file xsd schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
	<xs:element name="CMSErrors">
		<xs:annotation>
			<xs:documentation>Root element of the XML CMS error message
structure.
			</xs:documentation>
		</xs:annotation>
		<xs:complexType>
			<xs:sequence minOccurs="0">
				<xs:element ref="Error" minOccurs="0" maxOccurs="unbounded"/>
			</xs:sequence>
			<xs:attribute name="scheduleDate" type="gmtdatetime" use="required"/>
		</xs:complexType>
	</xs:element>
	<xs:element name="Error">
		<xs:annotation>
			<xs:documentation>This element is the description of an error.
			</xs:documentation>
		</xs:annotation>
		<xs:complexType>
			<xs:simpleContent>
				<xs:extension base="xs:string"/>
			</xs:simpleContent>
		</xs:complexType>
	</xs:element>
	<xs:simpleType name="gmtdatetime">
		<xs:annotation>
			<xs:documentation>A ISO 8601 compatible gmt datetime
                                Format : yyyy-mm-ddThh:mm:ssZ
			</xs:documentation>
		</xs:annotation>
		<xs:restriction base="xs:string">
			<xs:pattern value="[0-9][0-9][0-9][0-9]-[0-1][0-9]-[0-3][0-9]T[0-
2][0-9]:[0-6][0-9]:[0-6][0-9]Z"/>
		</xs:restriction>
	</xs:simpleType>
</xs:schema>
```

4. XML Samples

4.1 Import file Sample

This interface is very flexible. You can import a global VOD catalogue with all products in one file or you can import VOD item per VOD item in different files. An easy solution is to import one file per content. For content, you will import a file like the following examples.

4.1.1 Import file sample without editorial data

```
<LysisData id="CMS4X" scheduleDate="2005-11-01T00:00:00Z">
	<!-- id should be 3 letters + 9 numbers (3 letters are provider specific, numbers must be unique for a content provider) : XXX000000000 -->
	<Content action="override" duration="1800" id="XXX100000001" title="Movie A" encProfileName="Encryption_HD">
		<EpgDescription locale="en_GB">
			<EpgElement key="Title">Movie A</EpgElement>
		</EpgDescription>
		<EpgDescription>
			<EpgElement key="Rating">10</EpgElement>
		</EpgDescription>
		<Media action="override" comment="Technical comments" fileName="movieA.ts" fileSize="123456" frameDuration="123132" id="XXX200000001" format="AV_ClearTS"/>
	</Content>
	<VodItem id="XXX300000001" action="override" title="Movie A offer 1" contentRef="XXX100000001" nodeRefList="LYS000000963">
		<Period start="2015-11-01T00:00:00Z" end="2015-12-31T00:00:00Z"/>
	</VodItem>
	<Product id="XXX400000001" action="override" currency="EUR" price="2.0" rentalDuration="86400" type="single" title="Prod_1" validityStart="2015-12-01T00:00:00Z" validityEnd="2015-12-03T00:00:00Z">
		<AddToProduct elementKind="VodItem" elementId="XXX300000001"/>
	</Product>
	<Product id="XXX400000002" action="override" currency="EUR" price="2.5" rentalDuration="172800" type="single" title="Prod_2" validityStart="2015-12-03T00:00:00Z" validityEnd="2015-12-13T00:00:00Z">
		<AddToProduct elementKind="VodItem" elementId="XXX300000001"/>
	</Product>
</LysisData>
```

4.1.2 Import file sample with editorial data

```
<LysisData id="CMS4X" scheduleDate="2007-01-24T19:05:16Z" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
	<!-- id should be 3 letters + 9 numbers (3 letters are provider specific, numbers must be unique for a content provider) : XXX000000000 -->
	<Content id="XXX100000001" title="Les Brigades du Tigre" duration="7500" action="override" encProfileName="Encryption_HD">
		<!-- french language editorial data -->
		<EpgDescription locale="en_GB">
			<EpgElement key="Title">Les Brigades du Tigre</EpgElement>
			<EpgElement key="Synopsis">En 1907, une vague de crimes sans précédent ensanglante la Belle Epoque. Face aux bandits d'un nouveau siècle, le Ministre de l'Intérieur Georges Clemenceau crée une force de police à leur mesure : les Brigades Mobiles. En 1912, la France entière les connaît sous un autre nom : les Brigades du Tigre.</EpgElement>
```

```
                        <EpgElement key="Description">Film inspiré de la série Les Brigades
du Tigre.</EpgElement>
                        <EpgElement key="ShortTitle">Brigades du Tigre</EpgElement>
                        <EpgElement key="Copyright">2006 TFM Distribution</EpgElement>
                </EpgDescription>
                <!-- editorial data -->
                <EpgDescription>
                        <EpgElement key="Rating">10</EpgElement>
                        <EpgElement key="Directors">Jérôme Cornuau</EpgElement>
                        <EpgElement key="Producers">Manuel Munz</EpgElement>
                        <EpgElement key="Actors">Clovis Cornillac;Diane Kruger;Edouard
Baer</EpgElement>
                        <EpgElement key="Year">2006</EpgElement>
                        <EpgElement key="Language">fra</EpgElement>
                        <EpgElement key="AudioMode">5.1</EpgElement>
                        <EpgElement key="Aspect">Widescreen</EpgElement>
                        <EpgElement key="Definition">SD</EpgElement>
                        <EpgElement key="Studio">TFM Distribution</EpgElement>
                        <EpgElement key="Categories">Suspense</EpgElement>
                        <EpgElement key="Countries">FR</EpgElement>
                        <EpgElement key="Subtitles">Eng-Standard</EpgElement>
                </EpgDescription>
                <Media id="XXX200000001" fileName="3345_3800.ts" fileSize="4000000"
frameDuration="187500" comment="Programme en streaming" action="override"
format="AV_ClearTS"/>
        </Content>
        <!-- nodeRef place the VOD offer in the shop / sub shop structure. The list of
available node ids is defined during platform integration -->
        <VodItem id="XXX300000001" action="override" title="Brigades du Tigre"
contentRef="XXX100000001" nodeRefList="LYS000000963">
                <EpgDescription locale="en_GB">
                        <EpgElement key="DisplayPriority">0</EpgElement>
                </EpgDescription>
                <Period duration="14947199" start="2016-09-07T00:00:00Z" end="2017-02-
26T23:59:59Z"/>
        </VodItem>
        <Product id="XXX400000001" title="Les Brigades du Tigre" action="override"
currency="EUR" price="4.99000" rentalDuration="186400" type="single"
regions="France;Belgium">
                <AddToProduct elementKind="VodItem" elementId="XXX300000001"/>
        </Product>
</LysisData>
```

4.1.3 Import file sample with series editorial data

```
<LysisData id="CMS4X" scheduleDate="2007-01-24T19:05:16Z"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <!-- id should be 3 letters + 9 numbers (3 letters are provider specific, numbers
must be unique for a content provider) : LLL000000000 -->
        <Series id="LLL100000001" title="Desperate Housewife" action="override">
                <!-- french language editorial data -->
                <EpgDescription locale="fr_FR">
                        <EpgElement key="Title">Desperate Housewife</EpgElement>
                        <EpgElement key="Synopsis">Desperate Housewife synopsis</EpgElement>
                </EpgDescription>
                <!-- editorial data -->
                <EpgDescription>
                        <EpgElement key="Categories">Romance</EpgElement>
```

```
                <EpgElement key="Rating">10</EpgElement>
            </EpgDescription>
</Series>
      <Content id="LLL200000001" number="0" seriesRef="LLL100000001" title="The
beginning" duration="7500" action="override" encProfileName="Encryption_HD">
            <!-- french language editorial data -->
            <EpgDescription locale="fr_FR">
                <EpgElement key="Title">The beginning</EpgElement>
                <EpgElement key="Synopsis">Episode synopsis</EpgElement>
                <EpgElement key="Description">Episode description</EpgElement>
            </EpgDescription>
            <!-- editorial data -->
            <EpgDescription>
                <EpgElement key="Rating">10</EpgElement>
                <EpgElement key="Directors">Jérôme Cornuau</EpgElement>
                <EpgElement key="Producers">Manuel Munz</EpgElement>
                <EpgElement key="Actors">Clovis Cornillac;Diane Kruger;Edouard
Baer</EpgElement>
                <EpgElement key="Year">2006</EpgElement>
                <EpgElement key="Language">fra</EpgElement>
                <EpgElement key="AudioMode">5.1</EpgElement>
                <EpgElement key="Aspect">Widescreen</EpgElement>
                <EpgElement key="Definition">SD</EpgElement>
                <EpgElement key="Studio">TFM Distribution</EpgElement>
                <EpgElement key="Countries">FR</EpgElement>
                <EpgElement key="Subtitles">Eng-Standard</EpgElement>
            </EpgDescription>
            <Media id="LLL300000001" fileName="3345_3800.ts" fileSize="4000000"
frameDuration="187500" comment="Programme en streaming" action="override"
format="AV_ClearTS"/>
      </Content>
      <!-- nodeRef place the VOD offer in the shop / sub shop structure. The list of
available node ids is defined during platform integration -->
      <VodItem id="LLL400000001" action="override" title="The beginning"
contentRef="LLL200000001" nodeRefList="LYS000000963">
            <EpgDescription locale="fr_FR">
                <EpgElement key="DisplayPriority">0</EpgElement>
            </EpgDescription>
            <Period duration="14947199" start="2016-09-07T00:00:00Z" end="2017-02-
26T23:59:59Z"/>
      </VodItem>
      <Product id="LLL500000001" action="override" currency="EUR" price="4.99000"
rentalDuration="186400" type="single" regions="France;Belgium" title="The beginning">
            <AddToProduct elementKind="VodItem" elementId="LLL400000001"/>
      </Product>
</LysisData>
```

4.1.4 Import file sample TVOD and SVOD products

```
<LysisData id="CMS4X" scheduleDate="2007-01-24T19:05:16Z"
xmlns:xsl="http://www.w3.org/2001/XMLSchema-instance" >
      <!-- id should be 3 letters + 9 numbers (3 letters are provider specific, numbers
must be unique for a content provider and defined during integration) : XXX000000000 -->
      <Content id="XXX100000001" title="Les Brigades du Tigre" duration="7500"
action="override" encProfileName="Encryption_HD">
            <EpgDescription locale="en_GB">
                <EpgElement key="Title">Les Brigades du Tigre</EpgElement>
```

```
                        <EpgElement key="Synopsis">En 1907, une vague de crimes sans
précédent ensanglante la Belle Epoque. Face aux bandits d'un nouveau siècle, le Ministre
de l'Intérieur Georges Clemenceau crée une force de police à leur mesure : les Brigades
Mobiles. En 1912, la France entière les connaît sous un autre nom : les Brigades du
Tigre.</EpgElement>
                        <EpgElement key="Description">Film inspiré de la série Les Brigades
du Tigre.</EpgElement>
                        <EpgElement key="ShortTitle">Brigades du Tigre</EpgElement>
                        <EpgElement key="Copyright">2006 TFM Distribution</EpgElement>
                </EpgDescription>
                <!-- editorial data -->
                <EpgDescription>
                        <EpgElement key="Rating">10</EpgElement>
                        <EpgElement key="Directors">Jérôme Cornuau</EpgElement>
                        <EpgElement key="Producers">Manuel Munz</EpgElement>
                        <EpgElement key="Actors">Clovis Cornillac;Diane Kruger;Edouard
Baer</EpgElement>
                        <EpgElement key="Year">2006</EpgElement>
                        <EpgElement key="Language">fra</EpgElement>
                        <EpgElement key="AudioMode">5.1</EpgElement>
                        <EpgElement key="Aspect">Widescreen</EpgElement>
                        <EpgElement key="Definition">SD</EpgElement>
                        <EpgElement key="Studio">TFM Distribution</EpgElement>
                        <EpgElement key="Categories">Suspense</EpgElement>
                        <EpgElement key="Countries">FR</EpgElement>
                        <EpgElement key="Subtitles">Eng-Standard</EpgElement>
                </EpgDescription>
                <!-- the movie ts physical file -->
                <Media id="XXX200000001" fileName="3345_3800.ts" fileSize="4000000"
frameDuration="187500" comment="Programme en streaming" action="override"
format="AV_ClearTS"/>
        </Content>
        <!-- nodeRef place the VOD offer in the shop / sub shop structure. The list of
available node ids will be defined during platform integration -->
        <VodItem id="XXX300000001" action="override" title="Brigades du Tigre"
contentRef="XXX100000001" nodeRefList="LYS000000963">
                <EpgDescription locale="en_GB">
                        <EpgElement key="DisplayPriority">0</EpgElement>
                </EpgDescription>
                <Period duration="14947199" start="2016-09-07T00:00:00Z" end="2017-02-
26T23:59:59Z"/>
        </VodItem>
        <!-- single TVOD (1 film for 4.99) -->
        <Product id="XXX400000001" action="override" currency="EUR" price="4.99000"
rentalDuration="186400" type="single" title="Les Brigades du Tigre">
                <AddToProduct elementKind="VodItem" elementId="XXX300000001"/>
        </Product>
        <!-- SVOD (all movies of this offer for 12.99 / rental period) -->
        <Product id="XXX600000001" action="override" currency="EUR" price="12.99000"
title="Cinema" type="subscription">
                <AddToProduct elementKind="VodItem" elementId="XXX300000001"/>
        </Product>
</LysisData>
```

4.1.5 Import file sample with SVOD products

```
<LysisData id="CMS4X" scheduleDate="2007-01-24T19:05:16Z"
xmlns:xsl="http://www.w3.org/2001/XMLSchema-instance" >
	<!-- id should be 3 letters + 9 numbers (3 letters are provider specific, numbers
must be unique for a content provider and defined during integration) : XXX000000000 -->
	<Content id="XXX100000021" title="Cartoon 1" duration="7500" action="override"
encProfileName="Encryption_HD">
		<EpgDescription locale="en_GB">
			<EpgElement key="Title">Cartoon 1</EpgElement>
			<EpgElement key="Synopsis">Cartoon 1 synopsis.</EpgElement>
			<EpgElement key="Description">Cartoon 1 description.</EpgElement>
			<EpgElement key="ShortTitle">Cartoon 1</EpgElement>
			<EpgElement key="Copyright">2006 Walt Disney</EpgElement>
		</EpgDescription>
		<!-- editorial data -->
		<EpgDescription>
			<EpgElement key="Rating">10</EpgElement>
			<EpgElement key="Year">2006</EpgElement>
			<EpgElement key="Language">fra</EpgElement>
			<EpgElement key="AudioMode">5.1</EpgElement>
			<EpgElement key="Aspect">Widescreen</EpgElement>
			<EpgElement key="Definition">SD</EpgElement>
			<EpgElement key="Studio"> Walt Disney</EpgElement>
			<EpgElement key="Categories">Children</EpgElement>
			<EpgElement key="Countries">FR</EpgElement>
			<EpgElement key="Subtitles">Eng-Standard</EpgElement>
		</EpgDescription>
		<!-- the movie ts physical file -->
		<Media id="XXX200000021" fileName="3345_3821.ts" fileSize="4000000"
frameDuration="187500" comment="Programme en streaming" action="override"
format="AV_ClearTS"/>
	</Content>
	<Content id="XXX100000022" title="Cartoon 2" duration="7500" action="override"
encProfileName="Encryption_HD">
		<EpgDescription locale="en_GB">
			<EpgElement key="Title">Cartoon 2</EpgElement>
			<EpgElement key="Synopsis">Cartoon 2 synopsis.</EpgElement>
			<EpgElement key="Description">Cartoon 2 description.</EpgElement>
			<EpgElement key="ShortTitle">Cartoon 2</EpgElement>
			<EpgElement key="Copyright">2006 Walt Disney</EpgElement>
		</EpgDescription>
		<!-- editorial data -->
		<EpgDescription>
			<EpgElement key="Rating">10</EpgElement>
			<EpgElement key="Year">2006</EpgElement>
			<EpgElement key="Language">fra</EpgElement>
			<EpgElement key="AudioMode">5.1</EpgElement>
			<EpgElement key="Aspect">Widescreen</EpgElement>
			<EpgElement key="Definition">SD</EpgElement>
			<EpgElement key="Studio"> Walt Disney</EpgElement>
			<EpgElement key="Categories">Children</EpgElement>
			<EpgElement key="Countries">FR</EpgElement>
			<EpgElement key="Subtitles">Eng-Standard</EpgElement>
		</EpgDescription>
		<!-- the movie ts physical file -->
```

```
            <Media id="XXX200000022" fileName="3345_3822.ts" fileSize="4000000"
frameDuration="187500" comment="Programme en streaming" action="override"
format="AV_ClearTS"/>
      </Content>
      <Content id="XXX100000023" title="Cartoon 3" duration="7500" action="override"
encProfileName="Encryption_HD">
            <EpgDescription locale="en_GB">
                  <EpgElement key="Title">Cartoon 3</EpgElement>
                  <EpgElement key="Synopsis">Cartoon 3 synopsis.</EpgElement>
                  <EpgElement key="Description">Cartoon 3 description.</EpgElement>
                  <EpgElement key="ShortTitle">Cartoon 3</EpgElement>
                  <EpgElement key="Copyright">2006 Walt Disney</EpgElement>
            </EpgDescription>
            <!-- editorial data -->
            <EpgDescription>
                  <EpgElement key="Rating">10</EpgElement>
                  <EpgElement key="Year">2006</EpgElement>
                  <EpgElement key="Language">fra</EpgElement>
                  <EpgElement key="AudioMode">5.1</EpgElement>
                  <EpgElement key="Aspect">Widescreen</EpgElement>
                  <EpgElement key="Definition">SD</EpgElement>
                  <EpgElement key="Studio"> Walt Disney</EpgElement>
                  <EpgElement key="Categories">Children</EpgElement>
                  <EpgElement key="Countries">FR</EpgElement>
                  <EpgElement key="Subtitles">Eng-Standard</EpgElement>
            </EpgDescription>
            <!-- the movie ts physical file -->
            <Media id="XXX200000021" fileName="3345_3823.ts" fileSize="4000000"
frameDuration="187500" comment="Programme en streaming" action="override"
format="AV_ClearTS"/>
      </Content>
      <!-- nodeRef place the VOD offer in the shop / sub shop structure. The list of
available node ids will be defined during platform integration -->
      <VodItem id="XXX300000021" action="override" title="Cartoon 1"
contentRef="XXX100000021" nodeRefList="LYS000000963">
            <EpgDescription locale="en_GB">
                  <EpgElement key="DisplayPriority">0</EpgElement>
            </EpgDescription>
            <Period duration="14947199" start="2016-09-07T00:00:00Z" end="2017-02-
26T23:59:59Z"/>
      </VodItem>
      <VodItem id="XXX300000022" action="override" title="Cartoon 2"
contentRef="XXX100000022" nodeRefList="LYS000000963">
            <EpgDescription locale="en_GB">
                  <EpgElement key="DisplayPriority">0</EpgElement>
            </EpgDescription>
            <Period duration="14947199" start="2016-09-07T00:00:00Z" end="2017-02-
26T23:59:59Z"/>
      </VodItem>
      <VodItem id="XXX300000023" action="override" title="Cartoon 3"
contentRef="XXX100000023" nodeRefList="LYS000000963">
            <EpgDescription locale="en_GB">
                  <EpgElement key="DisplayPriority">0</EpgElement>
            </EpgDescription>
            <Period duration="14947199" start="2016-09-07T00:00:00Z" end="2017-02-
26T23:59:59Z"/>
      </VodItem>
      <!-- SVOD (all cartoons of this offer for 12.99 / rental period) -->
```

```
	<Product id="XXX600000002" action="override" currency="EUR" price="12.99000"
title="Cinema" type="subscription">
		<AddToProduct elementKind="VodItem" elementId="XXX300000021"/>
		<AddToProduct elementKind="VodItem" elementId="XXX300000022"/>
		<AddToProduct elementKind="VodItem" elementId="XXX300000023"/>
	</Product>
</LysisData>
```

4.1.6 Import file with Content Version

```
<LysisData id="CMS4X" scheduleDate="2007-01-24T19:05:16Z"
xmlns:xsl="http://www.w3.org/2001/XMLSchema-instance">
	<!-- id should be 3 letters + 9 numbers (3 letters are provider specific, numbers
must be unique for a content provider and defined during integration) : XXX000000000 -->
	<Content id="ABC100000001" title="Cartoon 11" duration="7500" action="override">
		<EpgDescription locale="en_GB">
			<EpgElement key="Title">Cartoon 11</EpgElement>
			<EpgElement key="Synopsis">Cartoon 11 synopsis.</EpgElement>
			<EpgElement key="Description">Cartoon 11 description.</EpgElement>
			<EpgElement key="ShortTitle">Cartoon 11</EpgElement>
			<EpgElement key="Copyright">2006 Walt Disney</EpgElement>
		</EpgDescription>
		<!-- editorial data -->
		<EpgDescription>
			<EpgElement key="Rating">10</EpgElement>
			<EpgElement key="Year">2006</EpgElement>
			<EpgElement key="Language">fra</EpgElement>
			<EpgElement key="AudioMode">5.1</EpgElement>
			<EpgElement key="Aspect">Widescreen</EpgElement>
			<EpgElement key="Definition">SD</EpgElement>
			<EpgElement key="Studio"> Walt Disney</EpgElement>
			<EpgElement key="Categories">Children</EpgElement>
			<EpgElement key="Countries">FR</EpgElement>
			<EpgElement key="Subtitles">Eng-Standard</EpgElement>
		</EpgDescription>
		<!-- the movie ts physical file -->
		<Content id="ABC100000011" title="Cartoon 11 (version 1)" duration="7500"
action="override" encProfileName="Encryption_SD" preLoaded="true">
			<EpgDescription>
				<EpgElement key="ShortTitle">Cartoon 11 (SD)</EpgElement>
			</EpgDescription>
			<!-- editorial data -->
			<EpgDescription>
				<EpgElement key="Definition">SD</EpgElement>
			</EpgDescription>
			<!-- the movie ts physical file -->
			<Media id="ABC200000011" fileName="3345_3821.ts" fileSize="4000000"
frameDuration="187500" format="AV_ClearTS"/>
			<!—Security Information -->
			<DrmInfo drmId="30" drmInstanceName="DRM_PlayReady"/>
		</Content>
		<Content id="ABC100000021" title="Cartoon 11 (version 2)" duration="7500"
action="override" encProfileName="Encryption_HD" preLoaded="true">
			<EpgDescription>
				<EpgElement key="ShortTitle">Cartoon 11 (HD)</EpgElement>
			</EpgDescription>
			<!-- editorial data -->
```

```
                    <EpgDescription>
                          <EpgElement key="Definition">HD</EpgElement>
                    </EpgDescription>
                    <!-- the movie ts physical file -->
                    <Media id="ABC200000021" fileName="3345_3822.ts" fileSize="8000000"
frameDuration="187500" format="AV_ClearTS"/>
              </Content>
       </Content>
</LysisData>
```

Acronyms

Acronym	Stands for	Definition
ASCII	American Standard Code for Information Interchange	The most popular coding method used by small computers for converting letters, numbers, punctuation and control codes into digital form. ASCII is a 7-bit character set encoding that contains characters for uppercase and lowercase English, American English
CAS	Conditional Access System	The set of features (and the components that implement them) which provide selective access to broadcast services on a per-subscriber basis. See conditional access.
CAS	Conditional Access System	The overall security system for providing and preventing access to digital interactive television system. It is composed of an IMS, ciphering units and a CA/DRM Server.
CMS	Content Management System	Backend system managing content, traffic and scheduling data
CMS	Card Management System	Part of Nagravision SMS managing the smart cards.
EPG	Electronic Program Guide	Depending on the context, refers either to the set-top box application providing a display of the channel schedule on the subscriber TV screen, or either to the whole schedule process: schedule information building in the IMS, transmission on the network
FTP	File transfer protocol	Standard Internet protocol and application to transmit files between machines. FTP uses the TCP/IP protocols. FTP is commonly used to transfer Web pages from their creator to the computer that acts as their server for everyone on the Internet.
GMT	Greenwich Mean Time	Today called Coordinated Universal Time (UTC). Reference time for CAS.
GUI	Graphical User Interface	Graphical (rather than purely textual) user interface to a computer
ID	Identifier	Unique value associated to an element to identify it.
ISO	International Organization for Standardization	ISO edicts norms about everything. For CAS, used norms concern in particular language and country codes.
MOD	Multimedia on Demand	feature offering all digital multimedia contents, such as video, audio, games, software, Ebook
NDA	Non-Disclosure Agreement	A legal contract between at least two parties that outlines confidential material, knowledge, or information that the parties wish to share with one another for certain purposes, but wish to restrict access to by third parties
PMT	Program Map Table	PSI table, MPEG-2 normalized. Identifies and indicates the locations of the streams that make up each MPEG service.
PPV	Pay Per View	A programme only accessible by buying the specific entitlement. A PPV can be a unique programme (e.g. live sport event) or a programme within an NVOD sequence (e.g. film).
PPV	Pay Per View	CAS feature allowing subscribers to purchase one-time viewing events or group of events. There are two types of PPV: pre-booked PPV and impulse PPV.
STB	Set Top Box	Television interface device receiving, demultiplexing and descrambling the television signal. Houses the smart card that gives authorization for descrambling and provides the control word. Syn: decoder, receiver, integrated receiver device, set-top unit,
TV	Television	

TVOD	True Video On Demand	True or real Video On Demand (VOD) feature: allows a subscriber to select a movie on a VideO Server (VOS).
UTF-8	UCS Transforming Format 8	UTF-8 is an alternative coded representation form for all the characters of the UCS. It can be used to transmit text data through communication systems which assume that individual octets in the range 00 to 7F have a definition according to ISO/IEC 4873. UTF-8 is a good way to go for using Unicode under Unix-style operating systems.
VOD	Video On Demand	umbrella term for a wide set of technologies whose common goal is to enable individuals to select video streams from a central server for viewing on a television or computer screen
XML	eXtensible Markup Language	XML is standard language used to create extensible data structures (XML schemas) and documents compatible with these structures (XML documents). Allows designers to create their own customized tags, enabling the definition, transmission, validation, and i

End of document