

NAGRA MediaLive CMS/WFM

Generic interface for encoder control

Version 0.9

Status: Proposed

Filename	NagraMediaLiveCMS-EncoderControl-SOAP-Interface-v0.9.docx
Date	June 8, 2017
Author(s)	Sanchez Christophe
Owner	Nagravision

Copyright © 2017 NagraVision. All rights reserved.
CH-1033 Cheseaux, Switzerland.
www.nagra.com

All trademarks and registered trademarks are the property of their respective owners.

This document is supplied with an understanding that the notice(s) herein or any other contractual agreement(s) made that instigated the delivery of a hard copy, electronic copy, facsimile or file transfer of this document are strictly observed and maintained.

The information contained in this document is subject to change without notice.

Security Policy of NagraVision Kudelski Group

Any recipient of this document, without exception, is subject to a Non-Disclosure Agreement (NDA) and access authorization.

Contents

Contents	3
List of figures.....	4
Preface	5
1. Overview	7
1.1 API context.....	7
1.2 SOAP mechanism	8
1.2.1 MultiDrmCommon types	8
1.2.2 Fault management	8
1.3 Sequence diagrams illustrating API use	9
1.3.1 Basic sequence diagram	9
1.3.2 Cancel triggered by the CMS	9
1.3.3 Cancel triggered by the encoder	10
1.4 Encoding job state diagram.....	11
2. Interface specification	12
2.1 CreateEncodingJob	12
2.1.1 Description	12
2.1.2 Request.....	13
2.1.2.1 Xsd	13
2.1.2.2 Parameters description	14
2.1.3 Response.....	14
2.1.3.1 Xsd	14
2.1.3.2 Parameters description	15
2.1.4 Fault	15
2.1.5 Samples	16
2.1.5.1 Use case with single output file in clear (no adaptive streaming)	16
2.1.5.2 Use case with HLS	17
2.2 GetEncodingJobStatus.....	19
2.2.1 Description	19
2.2.2 Request.....	19
2.2.2.1 Xsd	19
2.2.2.2 Parameters description	19
2.2.3 Response.....	20
2.2.3.1 Xsd	20
2.2.3.2 Parameters description	20
2.2.4 Fault	20
2.2.5 Sample.....	21
2.2.5.1 Request	21
2.2.5.2 Response	21
2.3 CancelEncodingJob	22
2.3.1 Description	22
2.3.2 Request.....	22
2.3.2.1 Xsd	22
2.3.2.2 Parameters description	22

2.3.3	Response	22
2.3.3.1	Parameters description	22
2.3.4	Fault	22
2.3.5	Sample.....	23
2.3.5.1	Request	23
2.3.5.2	Response	23
2.4	GetEncodedAssetUrl	24
2.4.1	Description	24
2.4.2	Request.....	24
2.4.2.1	Xsd	24
2.4.2.2	Parameters description	24
2.4.3	Response	24
2.4.3.1	Xsd	24
2.4.3.2	Parameters description	24
2.4.4	Fault	25
2.4.5	Samples	26
2.4.5.1	Use case without adaptive streaming in clear.....	26
2.4.5.2	Use case with HLS	26
Glossary.....		28
Acronyms.....		28

List of figures

Figure 1: Basic call sequence	9
Figure 2: CMS Cancel Call Sequence	9
Figure 3: Encoder Cancel Call Sequence	10
Figure 4: Encoding Job State Diagram	11
Figure 5: Directory structure diagram with a single output file	17

Preface

Document purpose

This document describes the basis of the NAGRA MediaLive Encoder SOAP interface that will be used to drive encoding activities.

Document change policy

It is defined that the versioning of this document is based on three parts incremented as follow:

- The first part is incremented when a change in the SOAP message is performed
- The second part is incremented when the changes impact only the documentation (for instance a correction in an explanation or in the typography)

Used Conventions

When the document references an XML element within the text, this element is written in another font and is represented between angle brackets (e.g. <Content>).

Other references to the same word make refer to the Media Live CMS 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.

Related Documents

[1] Encoder-KSS API: Encoder-KSS_IRS.docx

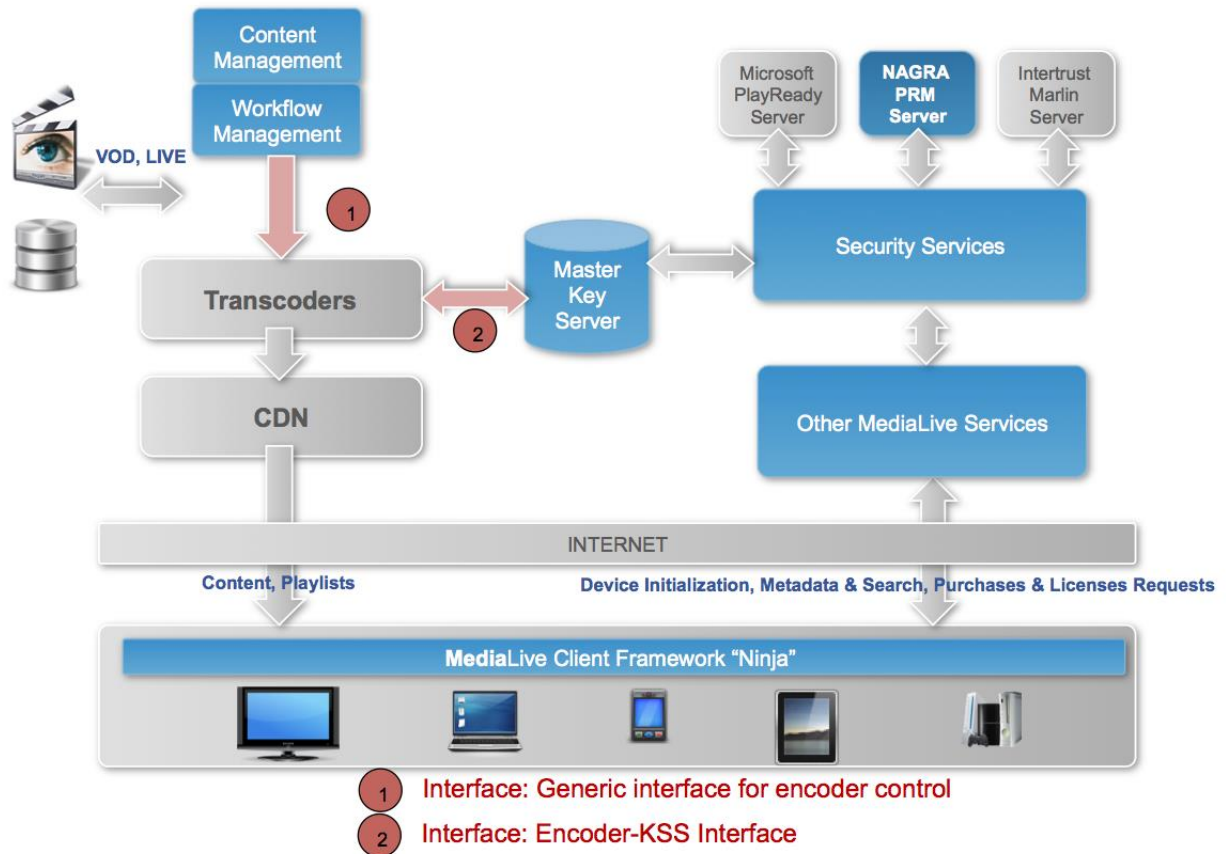
Document History

Change logs
0.9 / 2012.10.10 / Christophe Sanchez <ul style="list-style-type: none">Proposed version

1. Overview

1.1 API context

This document describes the API between the Nagra MediaLive WFM and the Transcoders (1).



Note

Interface (2) is covered in the document [1]

1.2 SOAP mechanism

The WSDL ([EncoderControl.wsdl](#)) is provided with xsd schema in order to validate (and type) the SOAP message.

For every SOAP method, a request, a response and a fault xsd are provided.

These xsd and wsdl are contained in [NagraMediaLiveCMS-EncoderControl-SOAP-Interface-v0.9.zip](#)

1.2.1 MultiDrmCommon types

Depending on the SOAP call, the [Subset MultiDrmCommonV1.xsd](#) can be included.

This xsd defines types used to transport data destined to the KSS.

These types are also used in the interface encoder-KSS (see [1])

1.2.2 Fault management

The fault xsd only allow validating the "detail" part of the fault message:

See example below in bold:

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header/>
  <env:Body>
    <env:Fault>
      <faultcode>env:Server</faultcode>
      <faultstring>Encoding Job Id does not exist</faultstring>
      <detail>
        <ns2:ExceptionWS xmlns:ns2="wfm:EncoderControlAPI/v1/schemas">
          <message>Encoding Job Id does not exist</message>
        </ns2:ExceptionWS>
      </detail>
    </env:Fault>
  </env:Body>
</env:Envelope>
```

Find below its definition:

[GenericException.xsd](#)

The "Fault" sections of this document list error messages that must at least be managed.

It is possible for the encoder vendor to specify additional error messages with the only constraint that they must be human readable.

Note

No specific behavior is planned to be done on error return on CMS side so no error code is managed.

1.3 Sequence diagrams illustrating API use

1.3.1 Basic sequence diagram

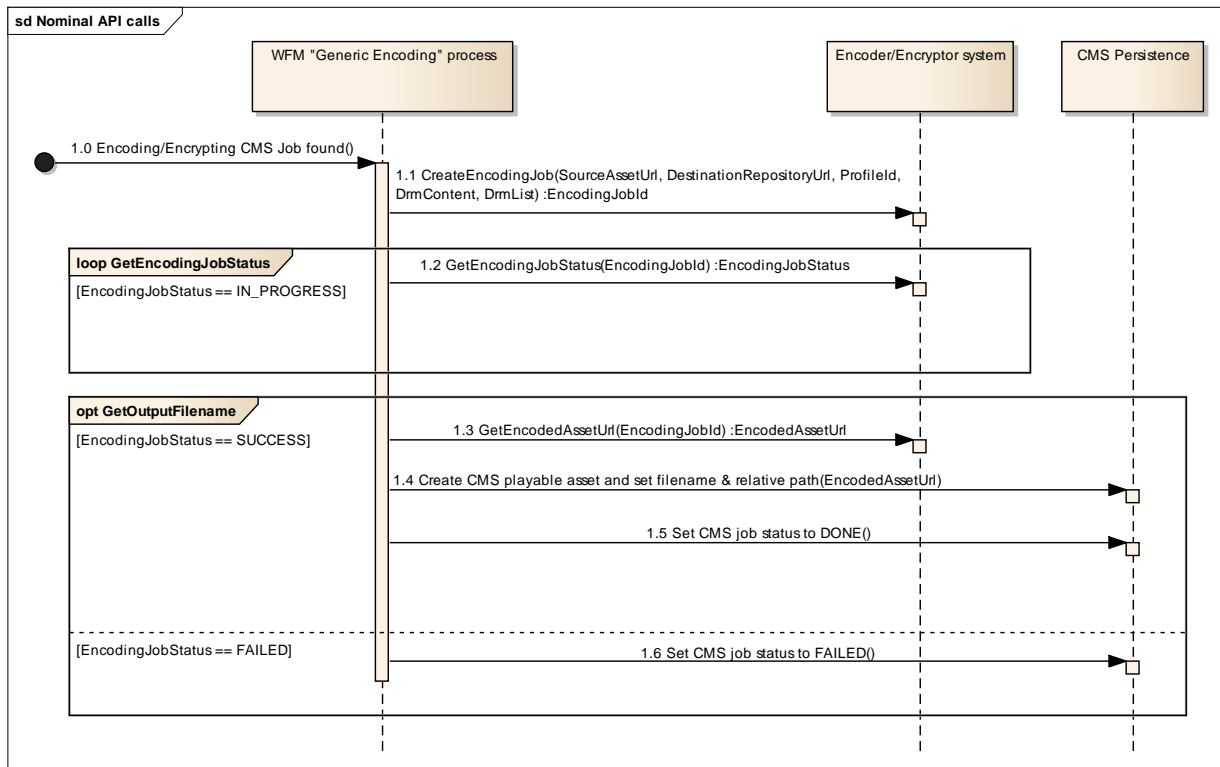


Figure 1: Basic call sequence

1.3.2 Cancel triggered by the CMS

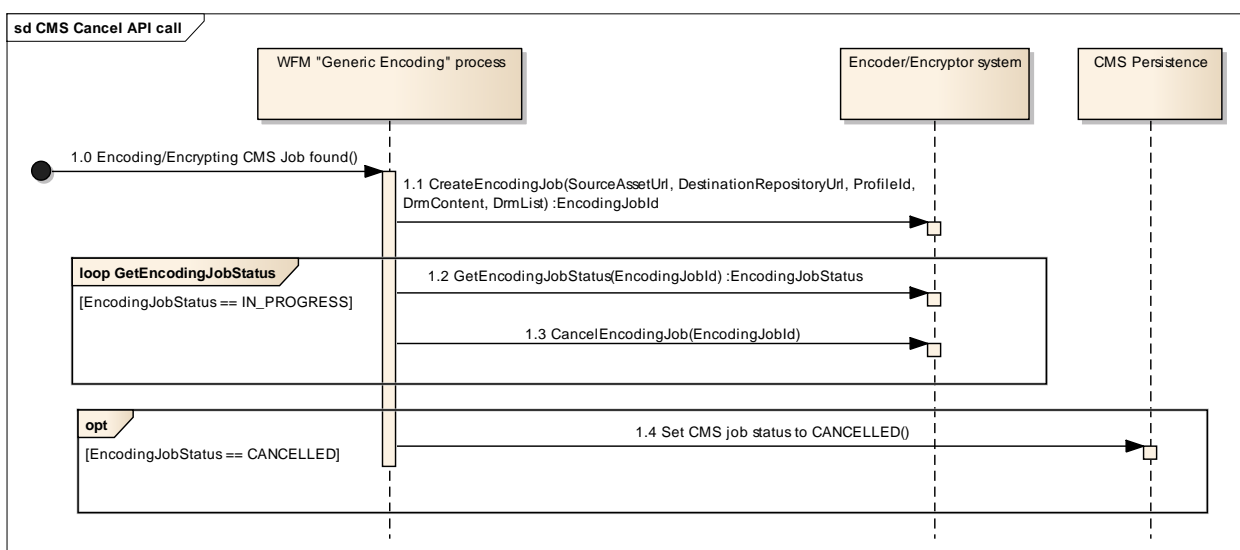


Figure 2: CMS Cancel Call Sequence

1.3.3 Cancel triggered by the encoder

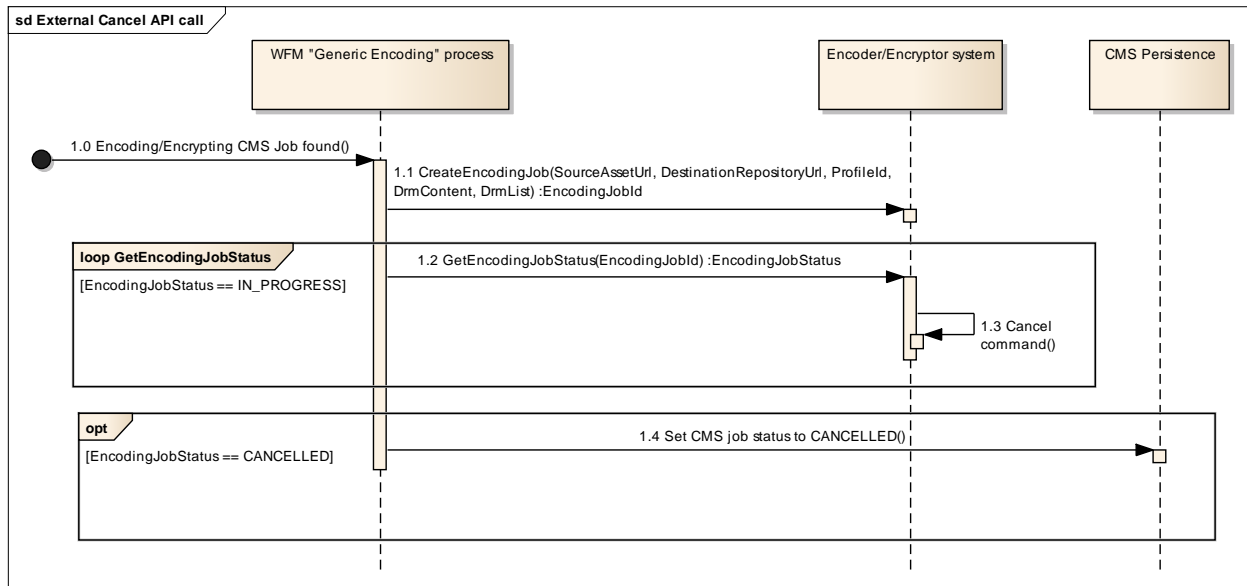


Figure 3: Encoder Cancel Call Sequence

1.4 Encoding job state diagram

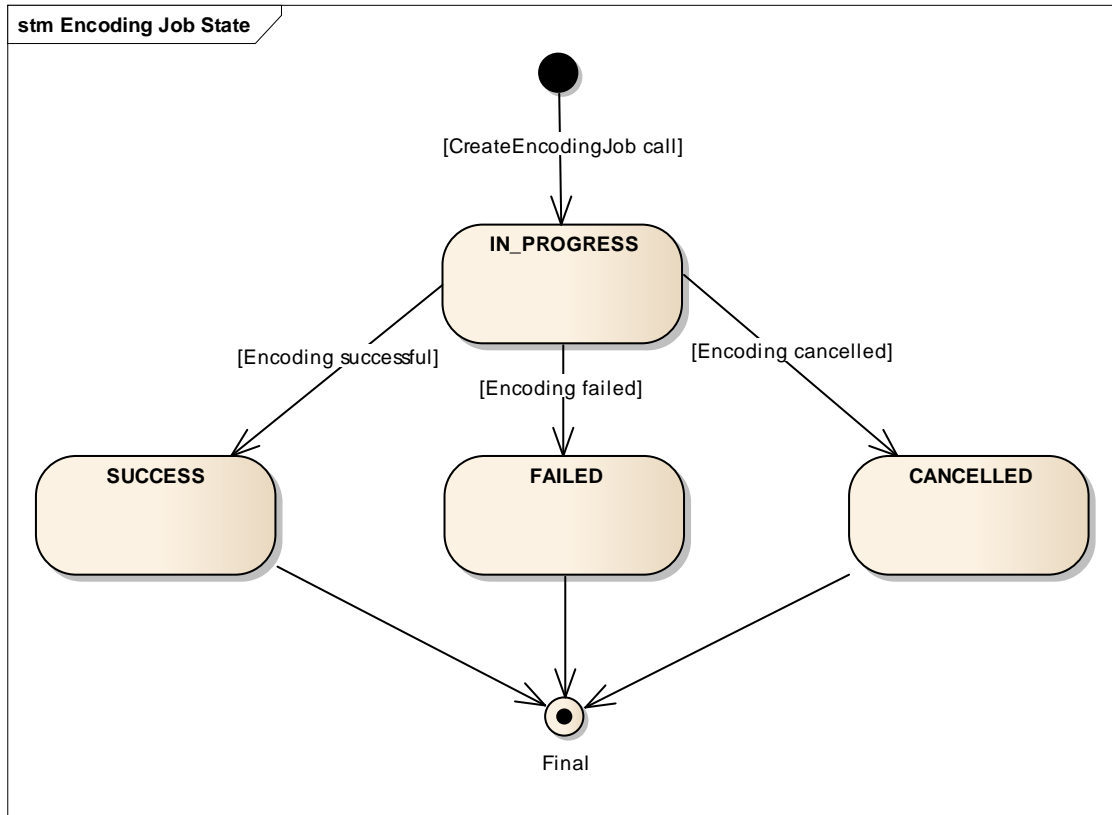


Figure 4: Encoding Job State Diagram

2. Interface specification

2.1 CreateEncodingJob

2.1.1 Description

The goal of this interface is to create and start an encoding job in the encoding system.

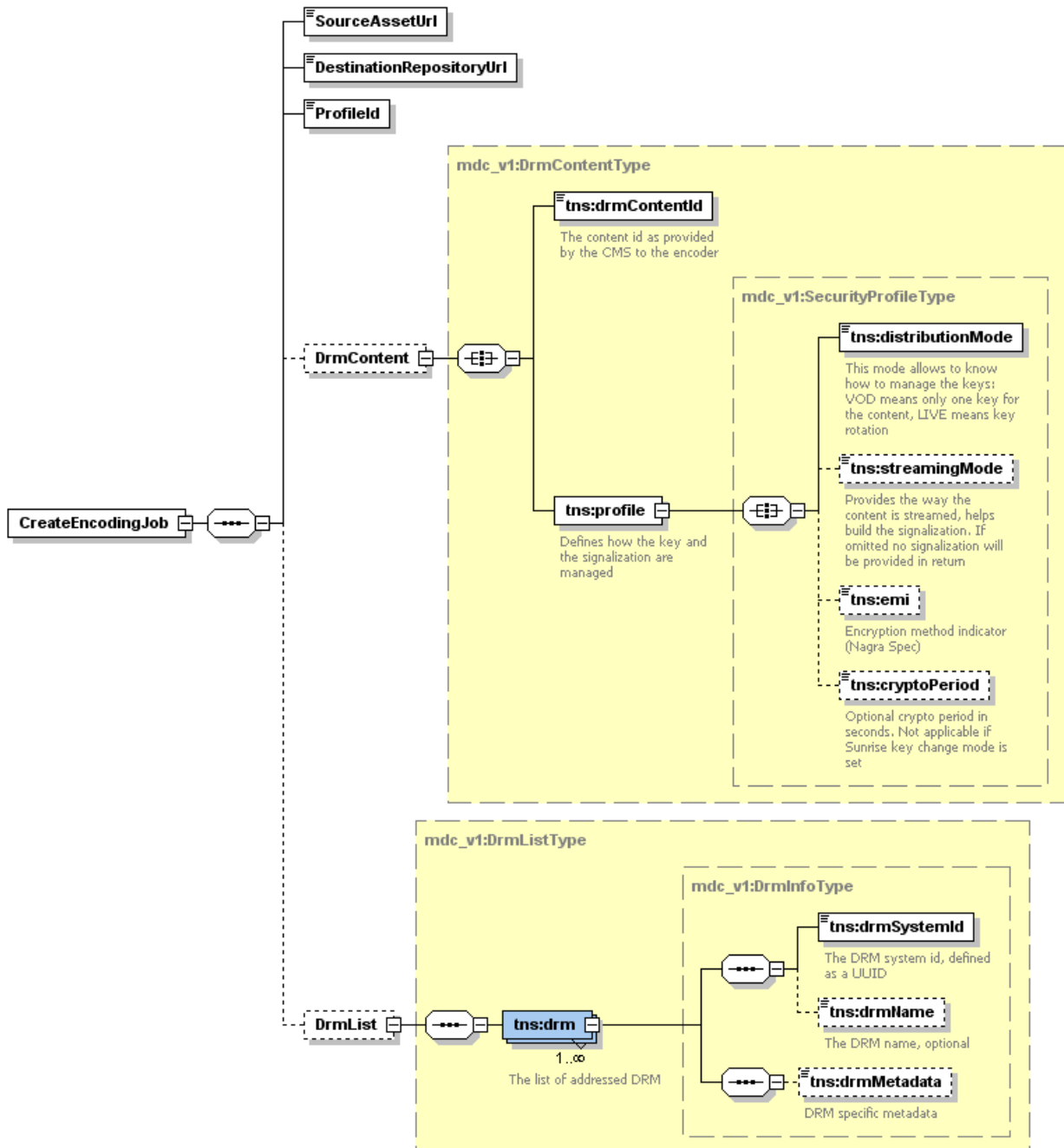
Once finished ("CANCELLED", "FAILED" or "SUCCESS"), the encoding job must be accessible (using "GetEncodingJobStatus" or "GetEncodingFilename") for a configurable period (typically 24 hours).

The encoder is in charge of purging all temporary files created during the encoding process once the encoding job is finished ("CANCELLED", "FAILED" or "SUCCESS").

2.1.2 Request

2.1.2.1 Xsd

[CreateEncodingJob.xsd](#)



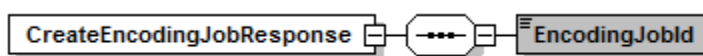
2.1.2.2 Parameters description

Name	Type	Description
SourceAssetUrl	xs:anyURI	URL of the source asset to encode. Local URL not supported.
DestinationRepositoryUrl	xs:anyURI	URL of the destination repository where to put the encoded asset. Local URL not supported.
ProfileId	xs:string	ID of the encoding profile to use. The format of the "ProfileId" parameter is dependent on the encoding system.
DrmContent	mdc_v1:DrmContentType	Content management and security profile information (See detailed description in [1]) Must be omitted if the "ProfileId" specified does not contain any encryption. Note: The "streamingMode" is mandatory in the context of this API.
DrmList	mdc_v1:DrmListType	The list of addressed DRMs used to protect the content (See detailed description in [1]). Must be omitted if the "ProfileId" specified does not contain any encryption. Note: An empty list means that the content is protected by all DRM supported by the system

2.1.3 Response

2.1.3.1 Xsd

[CreateEncodingJobResponse.xsd](#)



2.1.3.2 Parameters description

Name	Type	Description
EncodingJobId	xs:string	<p>ID of the encoding job as created by the encoding system.</p> <p>The format of the "EncodingJobId" parameter is dependent on the encoding system.</p>

2.1.4 Fault

A fault message will be returned in the following case:

- Source asset does not exist
- Source asset URL not accessible
- Destination URL not accessible
- Profile Id does not exist
- DrmContent is missing (only in case of profile with encryption)
- Drm <DrmName> is not accessible

See 1.2.2 for a fault response sample.

2.1.5 Samples

2.1.5.1 Use case with single output file in clear (no adaptive streaming)

2.1.5.1.1 Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:CreateEncodingJob>
      <ns:SourceAssetUrl>ftp://user:password@CIS:21/repository1/movie.ts</ns:SourceAssetUrl >
      <ns:DestinationRepositoryUrl>http://CDN:80/repository2/</ns:DestinationRepositoryUrl >
      <ns:ProfileId>456789</ns:ProfileId>
    </ns:CreateEncodingJob>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.5.1.2 Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:CreateEncodingJobResponse>
      <ns:EncodingJobId>110E8400-E29B-11D4-A716-446655449999</ns:EncodingJobId>
    </ns:CreateEncodingJobResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.5.1.3 Expected directory structure diagram

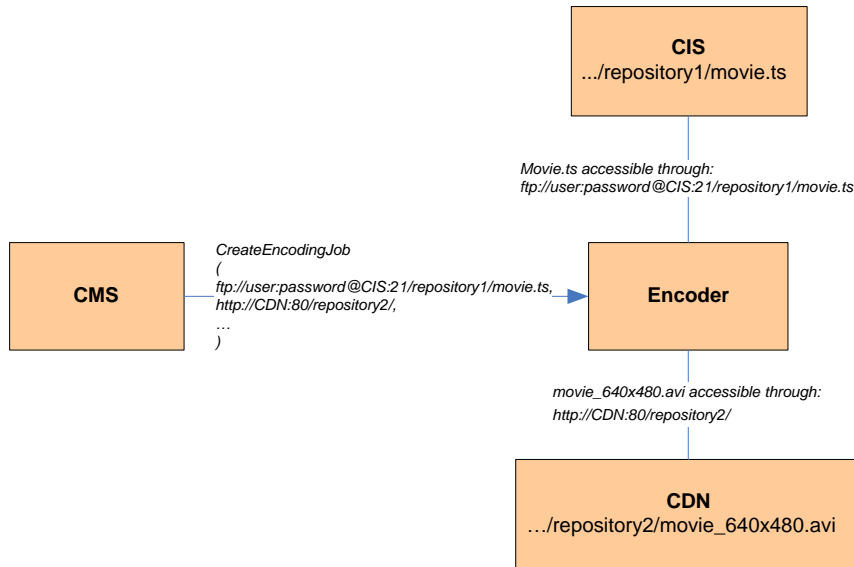


Figure 5: Directory structure diagram with a single output file

2.1.5.2 Use case with HLS

2.1.5.2.1 Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns1="wfm:EncoderControlAPI/v1/schemas" xmlns:mdc_v1="drm:MultiDrmCommon/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns1:CreateEncodingJob>
      <ns1:SourceAssetUrl>ftp://user:password@CIS:21/repository/movie.ts</ns1:SourceAssetUrl>
      <ns1:DestinationRepositoryUrl>http://CDN:80/movie_LYS000004979_201111202_113327/</ns1:DestinationRepositoryUrl >
      <ns1:ProfileId>111222</ns1:ProfileId>
      <ns1:DrmContent>
        <mdc_v1:drmContentId>123456789</mdc_v1:drmContentId>
        <mdc_v1:profile>
          <mdc_v1:distributionMode>VOD</mdc_v1:distributionMode>
          <mdc_v1:streamingMode>DASH</mdc_v1:streamingMode>
        </mdc_v1:profile>
      </ns1:DrmContent>
      <ns1:DrmList>
        <mdc_v1:drm>
          <mdc_v1:drmSystemId>10000000-0000-0000-0000-000000000001</mdc_v1:drmSystemId>
          <mdc_v1:drmName>NagraPrm</mdc_v1:drmName>
        </mdc_v1:drm>
        <mdc_v1:drm>
          <mdc_v1:drmSystemId>20000000-0000-0000-0000-000000000002</mdc_v1:drmSystemId>
          <mdc_v1:drmName>Marlin</mdc_v1:drmName>
        </mdc_v1:drm>
      </ns1:DrmList>
    </ns1:CreateEncodingJob>
  </soapenv:Body>
</soapenv:Envelope>
  
```

```

    </mdc_v1:drm>
  </ns1:DrmList>
</ns1:CreateEncodingJob>
</soapenv:Body>
</soapenv:Envelope>

```

2.1.5.2.2 Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ns="wfm:EncoderControlAPI/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:CreateEncodingJobResponse>
      <ns:EncodingJobId>110E8400-E29B-11D4-A716-446655440000</ns:EncodingJobId>
    </ns:CreateEncodingJobResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

2.1.5.2.3 Expected directory structure diagram

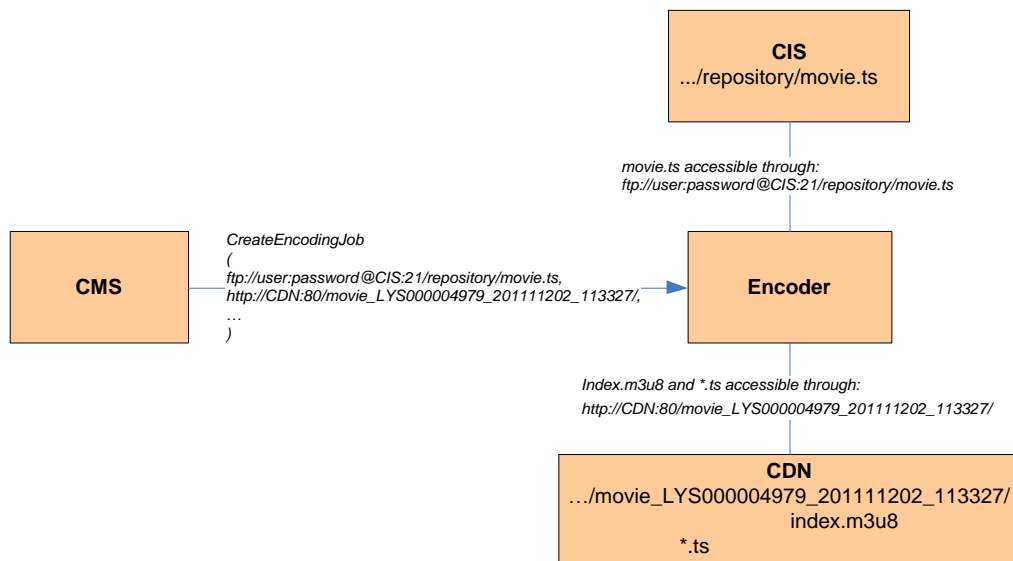


Figure 6: Directory structure diagram with HLS

2.2 GetEncodingJobStatus

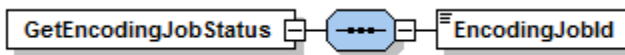
2.2.1 Description

The goal of this interface is to retrieve the encoding job status.

2.2.2 Request

2.2.2.1 Xsd

[GetEncodingJobStatus.xsd](#)



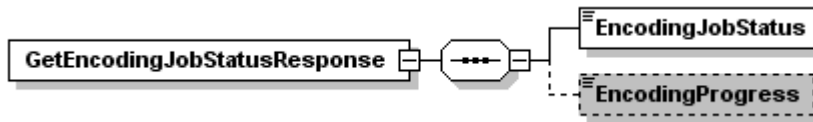
2.2.2.2 Parameters description

Name	Type	Description
EncodingJobId	xs:string	ID of the encoding job returned by "CreateEncodingJob" method. The format of the "EncodingJobId" parameter is dependent on the encoding system.

2.2.3 Response

2.2.3.1 Xsd

[GetEncodingJobStatusResponse.xsd](#)



2.2.3.2 Parameters description

Name	Type	Description
EncodingJobStatus	xs:string	<ul style="list-style-type: none"> - SUCCESS: The specified encoding job is successfully completed and the encoded asset is fully available on the destination repository ("DestinationUrl" used during the "CreateEncodingJob" method call) - FAILED: The specified encoding job has failed - IN_PROGRESS: The specified encoding job is in progress - CANCELLED: The specified encoding job has been cancelled
EncodingProgress	xs:unsignedShort	<p>Value in the range [0, 100] indicating the percentage of progress.</p> <p>This value is only relevant if "EncodingJobStatus" is "IN_PROGRESS".</p> <p>If omitted, the encoding progress is considered as not managed</p>

2.2.4 Fault

A fault message will be returned in the following case:

- Encoding Job Id does not exist

See 1.2.2 for a fault response sample.

2.2.5 Sample

2.2.5.1 Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <ns:GetEncodingJobStatus>  
      <ns:EncodingJobId>110E8400-E29B-11D4-A716-446655440000</ns:EncodingJobId>  
    </ns:GetEncodingJobStatus>  
  </soapenv:Body>  
</soapenv:Envelope>
```

2.2.5.2 Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <ns:GetEncodingJobStatusResponse>  
      <ns:EncodingJobStatus>IN_PROGRESS</ns:EncodingJobStatus >  
      <ns:EncodingProgress>52</ns:EncodingProgress>  
    </ns:GetEncodingJobStatusResponse>  
  </soapenv:Body>  
</soapenv:Envelope>
```

2.3 CancelEncodingJob

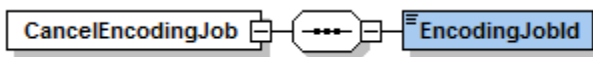
2.3.1 Description

The goal of this interface is to cancel an "IN_PROGRESS" encoding job. The concerned encoding job is set in the state "CANCELLED".

2.3.2 Request

2.3.2.1 Xsd

[CancelEncodingJob.xsd](#)



2.3.2.2 Parameters description

Name	Type	Description
EncodingJobId	xs:string	ID of the encoding job returned by "CreateEncodingJob" method. The format of the "EncodingJobId" parameter is dependent on the encoding system.

2.3.3 Response

2.3.3.1 Parameters description

No value is returned for this call.

2.3.4 Fault

A fault message will be returned in the following case:

- Encoding Job Id does not exist

See 1.2.2 for a fault response sample.

2.3.5 Sample

2.3.5.1 Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">  
  <soapenv:Header/>  
  <soapenv:Body>  
    <ns:CancelEncodingJob>  
      <ns:EncodingJobId>110E8400-E29B-11D4-A716-446655440000</ns:EncodingJobId>  
    </ns:CancelEncodingJob>  
  </soapenv:Body>  
</soapenv:Envelope>
```

2.3.5.2 Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">  
  <soapenv:Header/>  
  <soapenv:Body/>  
</soapenv:Envelope>
```

2.4 GetEncodedAssetUrl

2.4.1 Description

The goal of this interface is to retrieve the URL of the output encoded asset location once the encoding job is successfully completed (encoding job status is "SUCCESS").

The beginning of this URL must be equal to the "DestinationUrl" input parameter used during the "CreateEncodingJob" call.

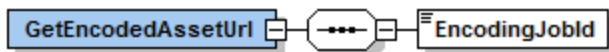
In the case of adaptive streaming, the encoded asset corresponds to the main playlist file.

In other cases, the encoded asset corresponds to the encoded output file itself.

2.4.2 Request

2.4.2.1 Xsd

[GetEncodedAssetUrl.xsd](#)



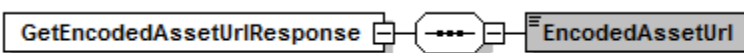
2.4.2.2 Parameters description

Name	Type	Description
EncodingJobId	xs:string	ID of the encoding job returned by "CreateEncodingJob" method. The format of the "EncodingJobId" parameter is dependent on the encoding system.

2.4.3 Response

2.4.3.1 Xsd

[GetEncodedAssetUrlResponse.xsd](#)



2.4.3.2 Parameters description

Name	Type	Description
------	------	-------------

EncodedAssetUrl	xs:anyURI	URL of the output encoded asset. Local URL not supported.
-----------------	-----------	--

2.4.4 Fault

A fault message will be returned in the following case:

- Encoding Job Id does not exist

See 1.2.2 for a fault response sample.

2.4.5 Samples

The following samples are based on the "CreateEncodingJob" ones (see 2.1.5).

2.4.5.1 Use case without adaptive streaming in clear

2.4.5.1.1 Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:GetEncodedAssetUrl>
      <ns:EncodingJobId>110E8400-E29B-11D4-A716-446655449999</ns:EncodingJobId>
    </ns:GetEncodedAssetUrl>
  </soapenv:Body>
</soapenv:Envelope>
```

2.4.5.1.2 Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:GetEncodedAssetUrlResponse>
      <ns:EncodedAssetUrl>http://CDN:80/repository2/movie\_640x480.avi</ns:EncodedAssetUrl >
    </ns:GetEncodedAssetUrlResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

2.4.5.2 Use case with HLS

2.4.5.2.1 Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:GetEncodedAssetUrl>
      <ns:EncodingJobId>110E8400-E29B-11D4-A716-446655440000</ns:EncodingJobId>
    </ns:GetEncodedAssetUrl>
  </soapenv:Body>
</soapenv:Envelope>
```

2.4.5.2.2 Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="wfm:EncoderControlAPI/v1/schemas">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:GetEncodedAssetUrlResponse>
      <ns:EncodedAssetUrl>http://CDN:80/movie\_LYS000004979\_201111202\_113327/index.m3u8
    </ns:EncodedAssetUrl>
  </ns:GetEncodedAssetUrlResponse>
```

</soapenv:Body>
</soapenv:Envelope>

Glossary

Name	Definition
Encoding	Used indifferently for encoding with or without encryption

Acronyms

Acronym	Stands For	Definition
CDN	Content Delivery network	Storage Server dedicated to deliver content to end user devices
CIS	Content Ingest Server	Storage Server dedicated to receive content/metadata to process
CMS	Content Management System	Backend system managing content, traffic and scheduling data
DRM	Digital Rights Management	Umbrella term to cover all sorts of active or passive mechanisms controlling the availability of locally stored digital content
ID	Identifier	Unique value associated to an element to identify it.
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
URL	Uniform Resource Locator	Address of a file (resource) accessible on the Internet.
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
WFM	Workflow Manager	Part of the CMS in charge of all external system interactions.

End of document