SMPTE ST 2021-4:2022
Revision of SMPTE ST 2021-4:2020

SMPTE STANDARD

Broadcast Exchange Format (BXF) Schema Documentation

Table of Contents

Approved April 4, 2022

Foreword

SMPTE (the Society of Motion Picture and Television Engineers) is an internationally-recognized standards developing organization. Headquartered and incorporated in the United States of America, SMPTE has members in over 80 countries on six continents. SMPTE"s Engineering Documents, including Standards, Recommended Practices, and Engineering Guidelines, are prepared by SMPTE"s Technology Committees. Participation in these Committees is open to all with a bona fide interest in their work. SMPTE cooperates closely with other standards-developing organizations, including ISO, IEC and ITU.

SMPTE Engineering Documents are drafted in accordance with the rules given in its Standards Operations Manual. This SMPTE Engineering Document was prepared by Technology Committee 34CS Media Systems, Control and Services.

Normative text is text that describes elements of the design that are indispensable or contains the conformance language keywords: "shall", "should", or "may". Informative text is text that is potentially helpful to the user, but not indispensable, and can be removed, changed, or added editorially without affecting interoperability. Informative text does not contain any conformance keywords.

All text in this document is, by default, normative, except: the Introduction, any section explicitly labeled as "Informative" or individual paragraphs that start with "Note:"

The keywords "shall" and "shall not" indicate requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.

The keywords "should" and "should not" indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document.

The keyword "reserved" indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword "forbidden" indicates "reserved" and in addition indicates that the provision will never be defined in the future.

A conformant implementation according to this document is one that includes all mandatory provisions ("shall") and, if implemented, all recommended provisions ("should") as described. A conformant implementation need not implement optional provisions ("may") and need not implement them as described.

Unless otherwise specified, the order of precedence of the types of normative information in this document shall be as follows: Normative prose shall be the authoritative definition; Tables shall be next; then formal languages; then figures; and then any other language forms.

SMPTE draws attention to the fact that it is claimed that compliance with this Standard may involve the use of one or more patents or other intellectual property rights (collectively, "IPR"). The Society takes no position concerning the evidence, validity, or scope of this IPR.

Each holder of claimed IPR has assured the Society that it is willing to License all IPR it owns, and any third party IPR it has the right to sublicense, that is essential to the implementation of this Standard to those (Members and non-Members alike) desiring to implement this Standard under reasonable terms and conditions, demonstrably free of discrimination. Each holder of claimed IPR has filed a statement to such effect with SMPTE. Information may be obtained from the Director, Standards & Engineering at SMPTE Headquarters.

Attention is also drawn to the possibility that elements of this Standard may be subject to IPR other than those identified above. The Society shall not be responsible for identifying any or all such IPR.

1 Scope

This document provides documentation of the BXF schemas specified in the ST2021-4a-2022.zip (the actual collection of XSD files that constitute the normative schema).

2 Terms and definitions

2.1
Programming Metadata Communication Protocol
PMCP

protocol defined for exchange of programming data, as specified in ATSC A/76B

3 Description of BXF schema

This documentation does not contain the actual schema, but instead provides a reference to the individual files that can be viewed with any number of tools, including most commonly used browsers as well as other third-party tools. In addition to the XML Schema Files (*.xsd) the user can also browse the schema using the HTML (*.html) files found in ST2021-4b-2022.zip.

Depending on the tools you use, different parts of the schema may appear in different formatting. The section below describes the meaning of various parts of the schema as you would see them in the HTML version.

The schema shall be normative and this document informative should any differences exist between the two.

4 Schema documentation

A number of graphics and symbols are used in the documentation to help describe the various elements of the schema and how the elements are related. These represent components and the relationships between schema components. The different components are represented by the following:

Single element – mandatory: Indicated by a rectangle with a solid border. The element name is inside the shape, as shown in Figure 1.

Figure 1 — Single element – mandatory.

Single element – optional: Indicated by a rectangle with a dashed border. The element name is inside the shape, as shown in Figure 2.

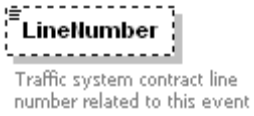


Figure 2 — Single element – optional.

Multiple elements: Indicated by a rectangle with a solid border with a number range representing the minimum and maximum number of occurrences possible, as shown in Figure 3. In the example, 0 to infinity (0..∞) is shown.

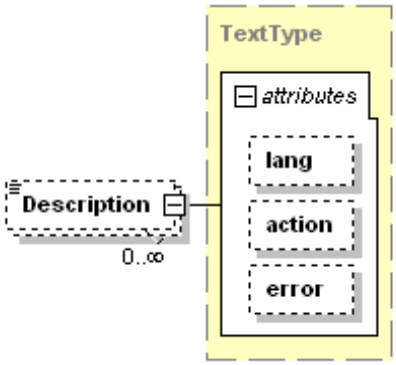


Figure 3 — Multiple elements.

Elements containing child elements: Indicated by a [+] or [-] on the element, representing an element containing additional attributes or elements. The [+] indicates that additional elements are available for display. The [-] indicates that the child elements are displayed, as shown in Figure 4.

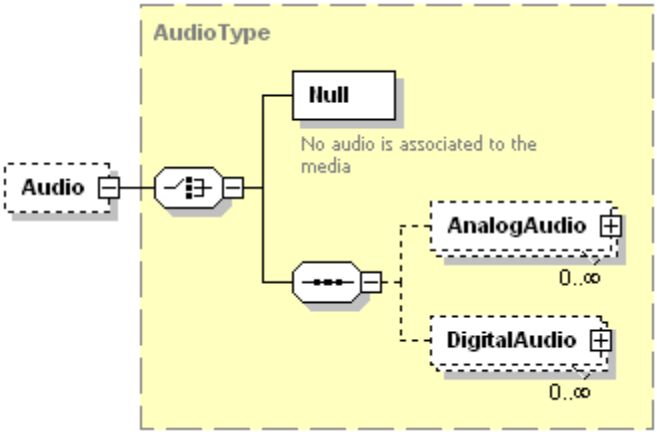


Figure 4 — Elements containing child elements.

Complex type: Indicated by a partial hexagon and a child element symbol, as shown in Figure 5.

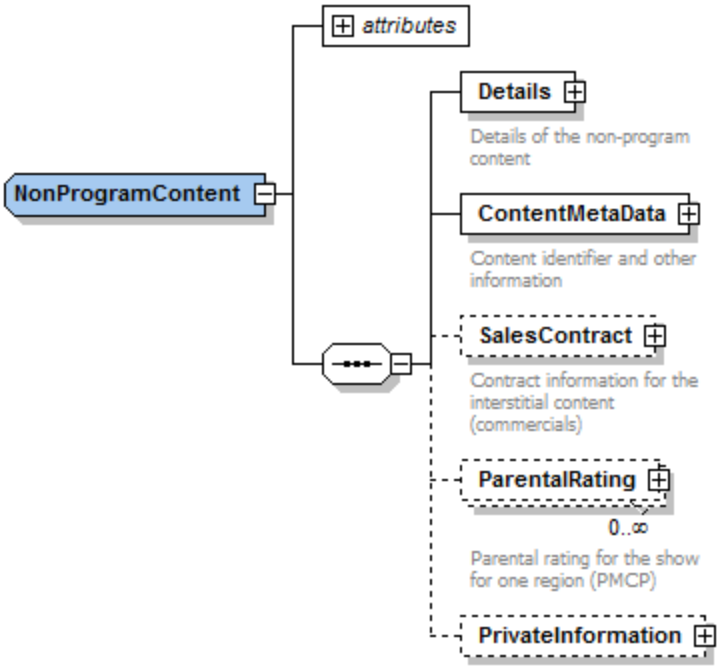


Figure 5 — Complex type.

Wildcards: Indicated by an octagon with any at the left, as shown in Figure 6.

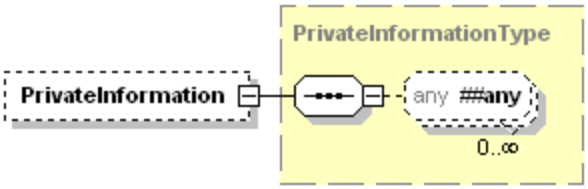


Figure 6 — Wildcards.

Attributes: Indicated by the word ‘attributes’, as shown in Figure 7.

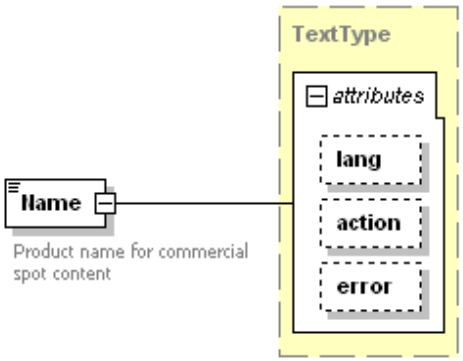


Figure 7 — Attributes.

The relationships between components are represented by symbols for sequence and choice.

Sequence: The sequence compositor shows that all elements occur in sequence, as shown in Figure 8.

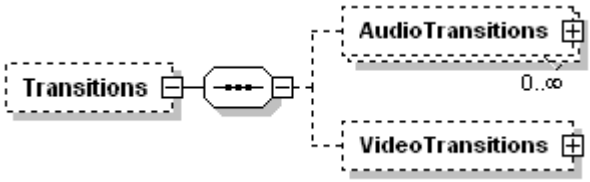


Figure 8 — Sequence.

Choice: The choice compositor shows the ‘or’ relationship between associated components (only one choice may be made), as shown in Figure 9.

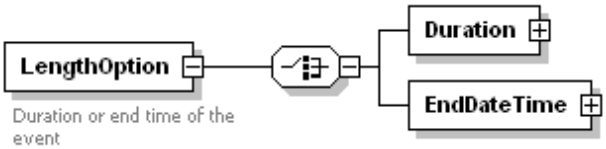


Figure 9 — Choice.

Solid lines: Solid lines connecting elements represent mandatory connections within the schema diagrams, as shown in Figure 10.

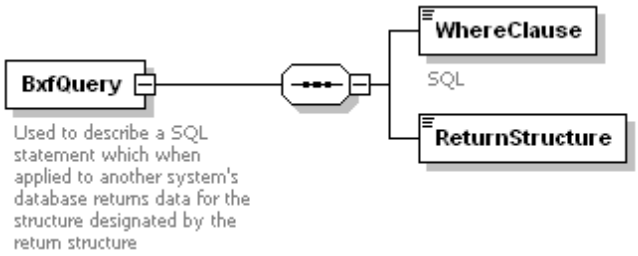


Figure 10 — Solid lines.

Dashed lines: Dashed lines between elements represent optional connections within the schema diagrams, as shown in Figure 11.

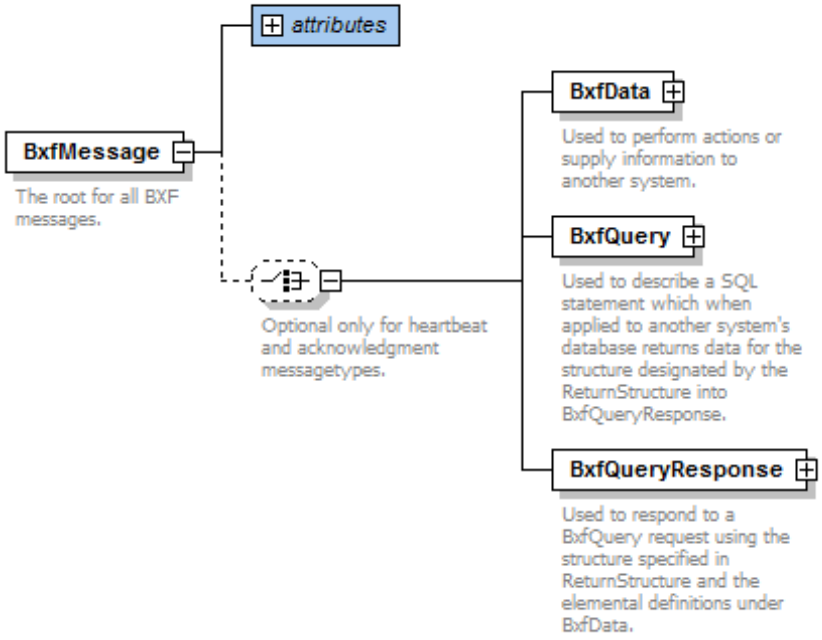


Figure 11 — Dashed lines.

4.1 Schema file list

The schema is composed of 33 files, as shown in Figure 12. The root file (starting point) is bxfshema.xsd and all other files are referenced from this file. A brief description of each file is given below in alphabetical order. This same description may also be contained in the schema for major elements, but some elements do not contain a description and were segregated into separate files for the convenience of managing the schema. Note that no new XSDs have been added as part of version 7.0, but there are several new elements added into contentmetadata.xsd, video.xsd and contentdelivery.xsd.

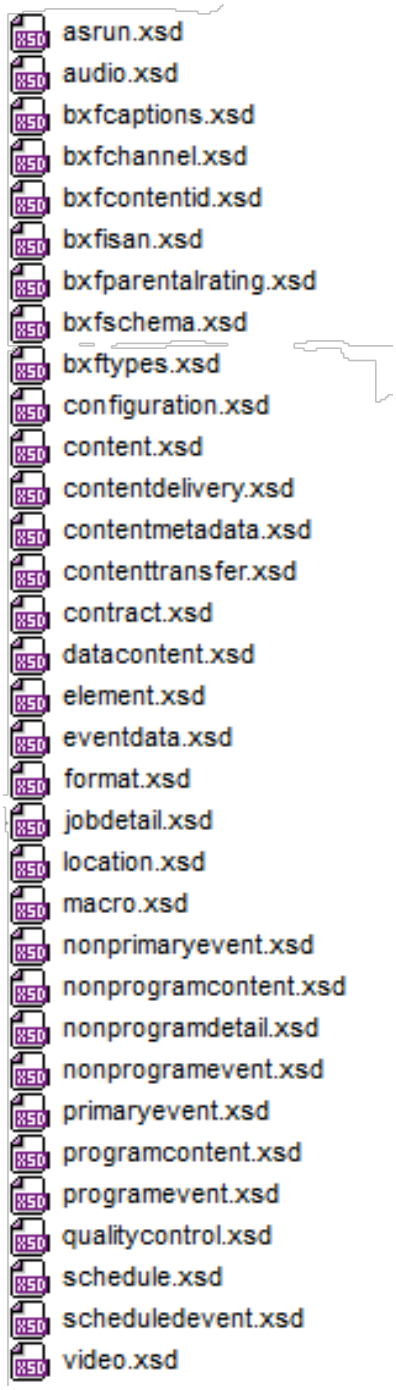


Figure 12 — Schema file list.

4.2 Header change in BXF 8.0

All XSD file headers have the same change regarding copyright and targetNamespace:

```
<!-- Copyright 2022 Society of Motion Picture and Television Engineers. All rights
reserved. -->

<xs:schema xmlns="http://smpte-ra.org/schemas/2021/2022/BXF"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:pmcp="http://www.atsc.org/XMLSchemas/pmcp/2007/3.1"
targetNamespace="http://smpte-ra.org/schemas/2021/2022/BXF"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="8.000">
```

In addition, a change has been made to require that all associated PMCP files, starting with this version, should be stored in a PMCP folder located within the main BXF folder. This will localize these files and remove the need to access the PMCP XSDs from a remote server hosted by the ATSC. At times, it has been reported that accessing the ATSC server had been inconsistent and resulted in errors for those using the BXF schema in their systems. As there is no expectation that any of these PMCP reference files will be modified in the future, SMPTE felt it best to fully incorporate them into the BXF distribution structure. ATSC is in full agreement with this change and has authorized SMPTE to distribute the appropriate PMCP files with v8.0.

The new <xs:import> located in the header of all the XSDs is now the following:

```
<xs:import namespace="http://www.atsc.org/XMLSchemas/pmcp/2007/3.1"
schemaLocation="PMCP/pmcp31.xsd"/>
```

Most of the BXF XSDs (25) had only their header changed as detailed above and are listed here. There are eight (8) XSDs with more substantial changes and these changes are detailed below.

Header Only Changes for these XSDs:

Asrun.xsd

BXFCaptions.xsd

BXFChannel.xsd

BXFContentID.xsd

BXFISAN.xsd

BXFParentalRating.xsd

BXFSchema.xsd

Configuration.xsd

Content.xsd

ContentTransfer.xsd

Contract.xsd

DataContent.xsd

EventData.xsd

Format.xsd

Location.xsd

Macro.xsd

NonPrimaryEvent.xsd

NonProgramContent.xsd

- NonProgramEvent.xsd
- PrimaryEvent.xsd
- ProgramContent.xsd
- ProgramEvent.xsd
- QualityControl.xsd
- Schedule.xsd
- ScheduleEvent.xsd

4.2.1 Audio.xsd(*)

The base for all audio definitions, as shown in Figure 13.

Contains:

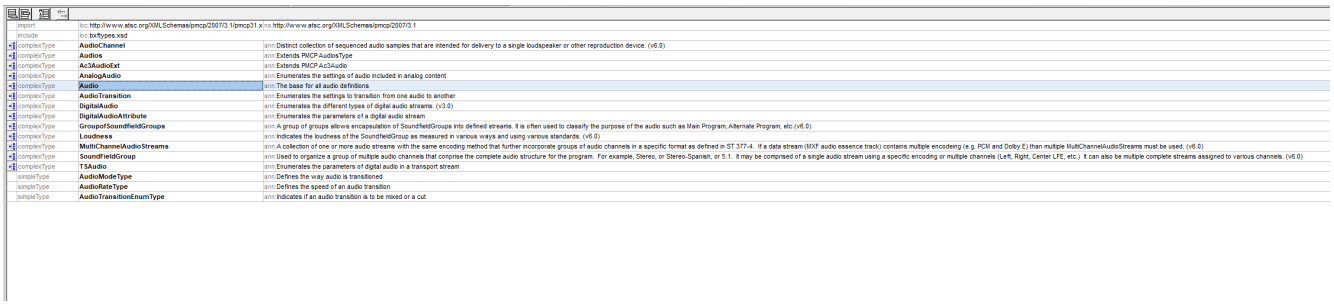


Figure 13 — Audio.xsd

4.2.1.1 BXF 8.0 Changes

Description of change: Added a new optional element to Audio to allow the indication of watermarking in an audio file, as shown in Figure 14. In addition, the node to which watermarking is attached was set to be unbounded. The Watermarking element is configured as an *xs:string* with five enumerations: *ATSC 3.0; BVS; Nielsen; OBID and Other*. In addition, the DigitialAudioAttribute was given a new optional attribute, "muteOnSound", configured as a boolean, which when set to True indicates that there is no sound on the audio track(s).

Text representations:

```
<xs:complexType name="Audio">
<xs:annotation>
     <xs:documentation>The base for all audio definitions</xs:documentation>
</xs:annotation>
     <xs:choice>
          <xs:element name="Null">
          <xs:annotation>
               <xs:documentation>No audio is associated with the
media</xs:documentation>
          </xs:annotation>
               <xs:complexType/>
          </xs:element>
               <xs:sequence maxOccurs="unbounded">
               <xs:annotation>
```

```
                    <xs:documentation>Added unbounded to allow mapping of audio to
watermarking. (v8.0)</xs:documentation>

                </xs:annotation>

                <xs:element name="AnalogAudio" type="AnalogAudio" minOccurs="0"
maxOccurs="unbounded"/>

                <xs:element name="DigitalAudio" type="DigitalAudio" minOccurs="0"
maxOccurs="unbounded">

                <xs:annotation>

                    <xs:documentation>Enumerates the different types of digital
audio streams. (v3.0)</xs:documentation>

                </xs:annotation>

                </xs:element>

                <xs:element name="MultiChannelAudioStreams"
type="MultiChannelAudioStreams" minOccurs="0" maxOccurs="unbounded">

                <xs:annotation>

                    <xs:documentation>A collection of one or more audio streams that
further incorporate groups of audio channels in a specific format as defined in SMPTE
ST 377M.</xs:documentation>

                </xs:annotation>

                </xs:element>

                <xs:element name="Watermarking" minOccurs="0" maxOccurs="unbounded">

                <xs:annotation>

                    <xs:documentation>Used to indicate which watermarking is
associated with the Audio Streams. (v8.0)</xs:documentation>

                </xs:annotation>

                    <xs:simpleType>

                        <xs:restriction base="xs:string">

                            <xs:enumeration value="ATSC 3.0"/>

                            <xs:enumeration value="BVS"/>

                            <xs:enumeration value="Nielsen"/>

                            <xs:enumeration value="OBID"/>
```

```
                                <xs:enumeration value="Other"/>

                            </xs:restriction>

                        </xs:simpleType>

                    </xs:element>

                </xs:sequence>

            </xs:choice>

…

<xs:attribute name="muteOnSound" type="xs:boolean">

<xs:annotation>

     <xs:documentation>True indicates that there is no sound on the audio track(s).
(v8.0)</xs:documentation>

</xs:annotation>

</xs:attribute>
```

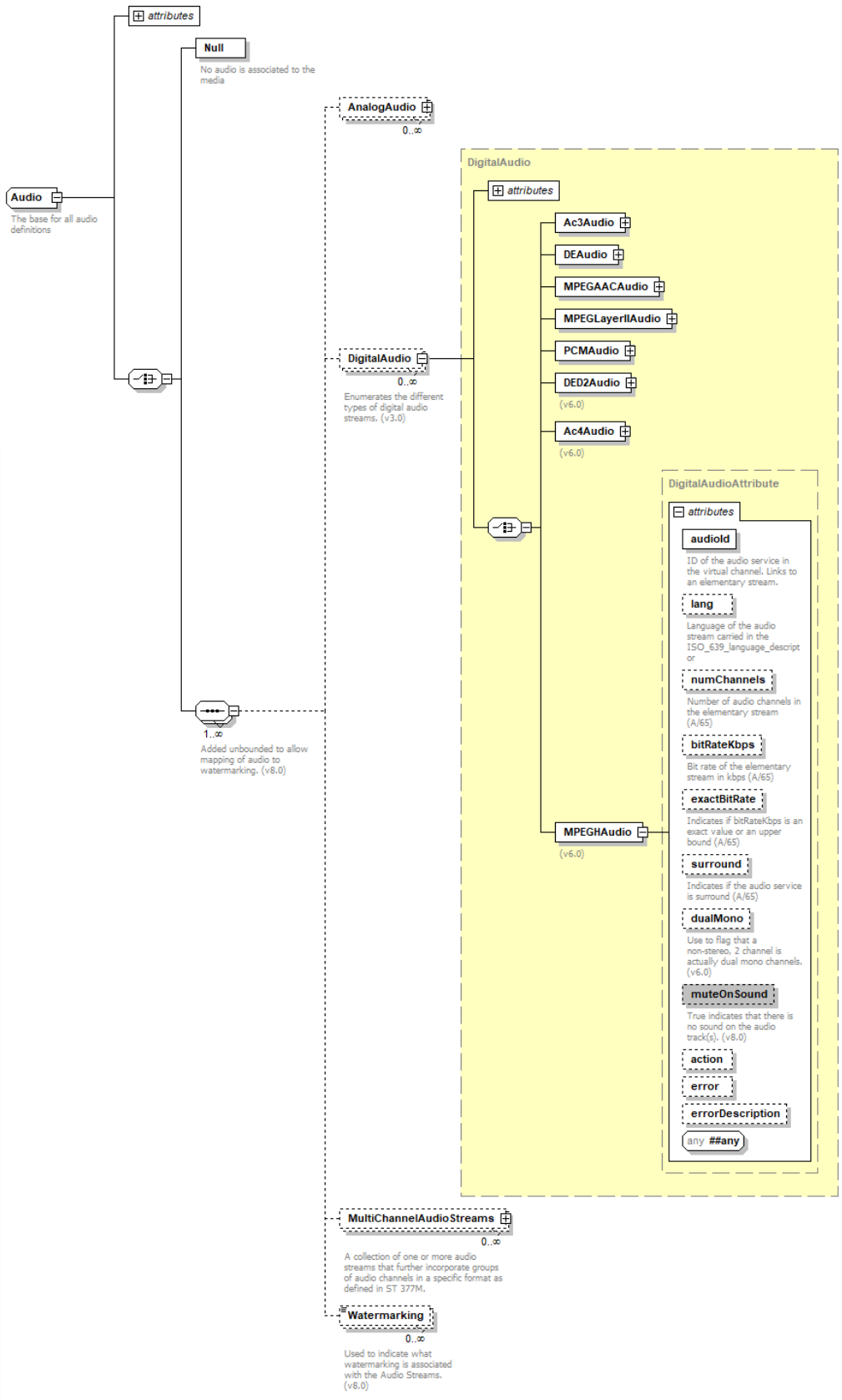
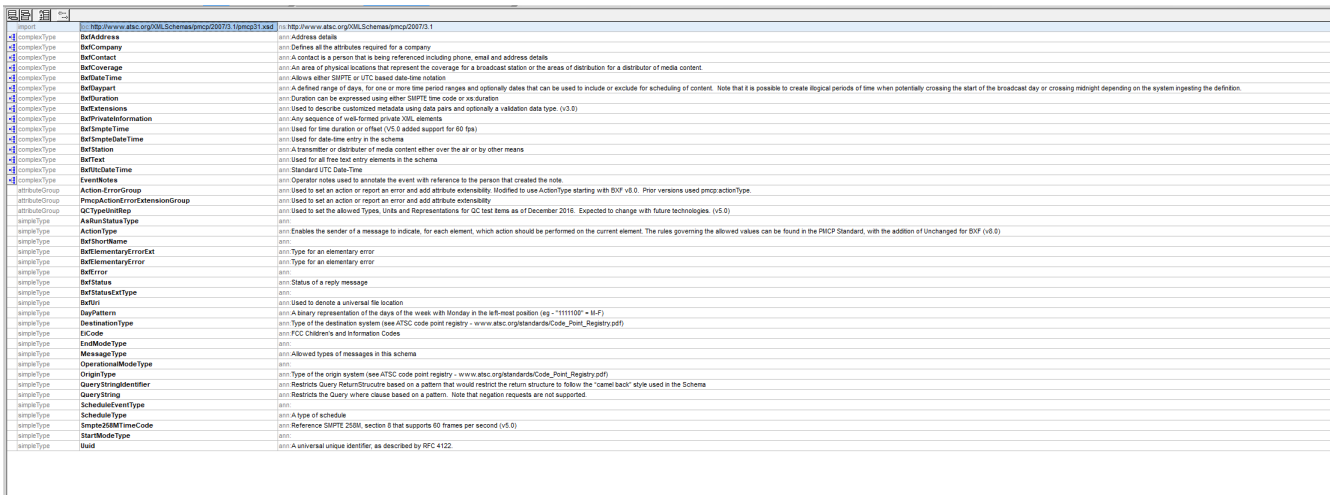


Figure 14 — Audio.xsd detail.

4.2.2 BxfTypes.xsd(*)

Used to organize all the utility elements that are used by most of the other schema files, as shown in Figure 15.

Contains:



import	http://www.atsc.org/XMLSchemas/pmcp/2007/3.1/pmcp31.xsd	ns http://www.atsc.org/XMLSchemas/pmcp/2007/3.1
complexType	BxfAddress	ann Address details
complexType	BxfCompany	ann Defines all the attributes required for a company
complexType	BxfContact	ann A contact is a person that is being referenced including phone, email and address details
complexType	BxfCoverage	ann An area of physical locations that represent the coverage for a broadcast station or the areas of distribution for a distributor of media content.
complexType	BxfDateTime	ann Allows either SMPTE or UTC based date-time notation
complexType	BxfDaypart	ann A defined range of days, for one or more time period ranges and optionally dates that can be used to include or exclude for scheduling of content. Note that it is possible to create illogical periods of time when potentially crossing the start of the broadcast day or crossing midnight depending on the system ingesting the definition.
complexType	BxfDuration	ann Duration can be expressed using either SMPTE time code or xs:duration
complexType	BxfExtensions	ann Used to describe customized metadata using data pairs and optionally a validation data type. (v3.0)
complexType	BxfPrivateInformation	ann Any sequence of well-formed private XML elements
complexType	BxfSmpteTime	ann Used for time duration or offset (V5.0 added support for 60 fps)
complexType	BxfSmpteDateTime	ann Used for date-time entry in the schema
complexType	BxfStation	ann A transmitter or distributer of media content either over the air or by other means
complexType	BxfText	ann Used for all free text entry elements in the schema
complexType	BxfUtcDateTime	ann Standard UTC Date-Time
complexType	EventNotes	ann Operator notes used to annotate the event with reference to the person that created the note.
attributeGroup	Action-ErrorGroup	ann Used to set an action or report an error and add attribute extensibility. Modified to use ActionType starting with BXF v8.0. Prior versions used pmcp:actionType.
attributeGroup	PmcpActionErrorExtensionGroup	ann Used to set an action or report an error and add attribute extensibility
attributeGroup	QCTypeUnitRep	ann Used to set the allowed Types, Units and Representations for QC test items as of December 2016. Expected to change with future technologies. (v5.0)
simpleType	AsRunStatusType	ann:
simpleType	ActionType	ann Enables the sender of a message to indicate, for each element, which action should be performed on the current element. The rules governing the allowed values can be found in the PMCP Standard, with the addition of Unchanged for BXF (v8.0)
simpleType	BxfShortName	ann:
simpleType	BxfElementaryErrorExt	ann Type for an elementary error
simpleType	BxfElementaryError	ann Type for an elementary error
simpleType	BxfError	ann:
simpleType	BxfStatus	ann Status of a reply message
simpleType	BxfStatusExtType	ann:
simpleType	BxfUri	ann Used to denote a universal file location
simpleType	DayPattern	ann A binary representation of the days of the week with Monday in the left-most position (eg - "1111100" = M-F)
simpleType	DestinationType	ann Type of the destination system (see ATSC code point registry - www.atsc.org/standards/Code_Point_Registry.pdf)
simpleType	EICode	ann FCC Children's and Information Codes
simpleType	EndModeType	ann:
simpleType	MessageType	ann Allowed types of messages in this schema
simpleType	OperationalModeType	ann:
simpleType	OriginType	ann Type of the origin system (see ATSC code point registry - www.atsc.org/standards/Code_Point_Registry.pdf)
simpleType	QueryStringIdentifier	ann Restricts Query ReturnStrucutre based on a pattern that would restrict the return structure to follow the "camel back" style used in the Schema
simpleType	QueryString	ann Restricts the Query where clause based on a pattern. Note that negation requests are not supported.
simpleType	ScheduleEventType	ann:
simpleType	ScheduleType	ann A type of schedule
simpleType	Smpte258MTimeCode	ann Reference SMPTE 258M, section 8 that supports 60 frames per second (v5.0)
simpleType	StartModeType	ann:
simpleType	tuid	ann A universal unique identifier, as described by RFC 4122.

Figure 15 — BxfTypes.xsd

4.2.2.1 BXF 8.0 Changes

Description of change: The element BxfDaypart/DaypartDetail added a new optional attribute, audienceDeficiencyUnit, which has the type boolean and when True indicates that the Daypart described is an ADU, as shown in Figure 16. In the AttributeGroup, Action-ErrorGroup, which is used throughout the application, previously used an imported PMCP based simpleType called pmcp:actionType to indicate the type of action attributes that could be set. With version 8.0, a new simpleType named ActionType has been added that is identical to the old PMCP version, except for the addition of the "unchanged" option.

Text representations:

```
<xs:attribute name="audienceDeficiencyUnit" type="xs:boolean">

<xs:annotation>

     <xs:documentation>If True indicates that it is an ADU.
(v8.0).</xs:documentation>

</xs:annotation>

</xs:attribute>



<xs:attributeGroup name="Action-ErrorGroup">

<xs:annotation>
```

```
      <xs:documentation>Used to set an action or report an error and add attribute
extensibility. Modified to use ActionType starting with BXF v8.0. Prior versions used
pmcp:actionType.</xs:documentation>

</xs:annotation>

      <xs:attribute name="action" type="ActionType" use="optional"/>

      <xs:attribute name="error" type="BxfError" use="optional"/>

      <xs:attribute name="errorDescription" type="xs:string" use="optional"/>

      <xs:anyAttribute namespace="##any" processContents="lax"/>

</xs:attributeGroup>



<xs:simpleType name="ActionType">

<xs:annotation>

      <xs:documentation>Enables the sender of a message to indicate, for each
element, which action should be performed on the current element. The rules governing
the allowed values can be found in the PMCP Standard, with the addition of Unchanged
for BXF (v8.0)</xs:documentation>

</xs:annotation>

      <xs:restriction base="xs:string">

            <xs:enumeration value="add"/>

            <xs:enumeration value="read"/>

            <xs:enumeration value="remove"/>

            <xs:enumeration value="unchanged"/>

            <xs:enumeration value="update"/>

      </xs:restriction>

</xs:simpleType>
```

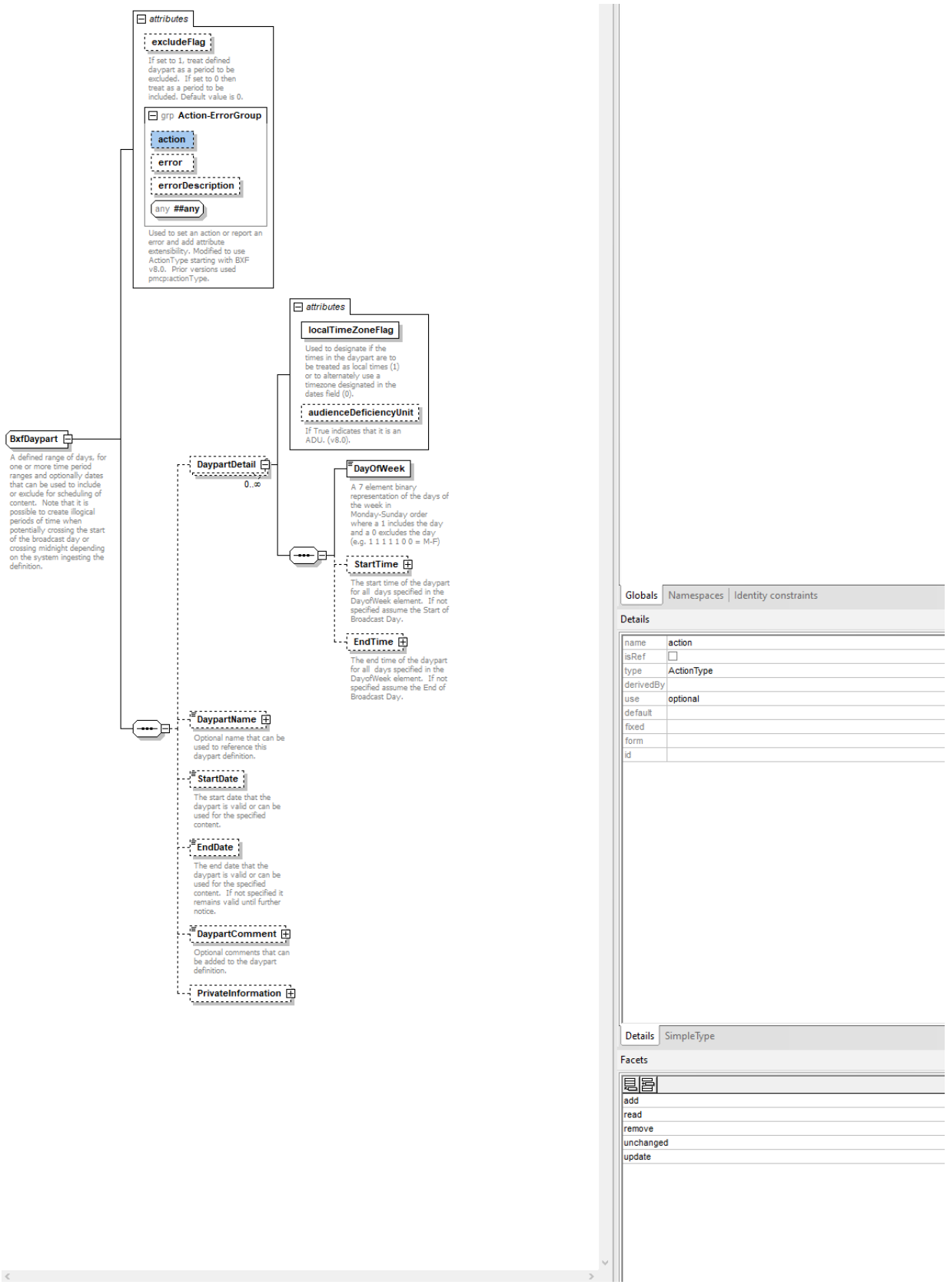


Figure 16 — BxfDaypart.xsd

4.2.3 ContentDelivery.xsd(*)

Used to specify the method by which media content should be created and formatted for downstream use by a designated media company, as shown in Figure 17. Media is often formatted in different manners depending on the planned use for that media downstream. ContentDelivery now supports both media designated to be aired as well as media content that is considered a Library Master, but may have additional modification before used for air. AirReadyMaster was released in v5.0 and the corresponding document for that version should be consulted for details. LibraryMaster was added in release v6.0 with some modifications that were made in v7.0. With V8.0 ContentDelivery adds support to designate the method of media delivery (service used) as well as the expected date and time of that delivery.

Contains:

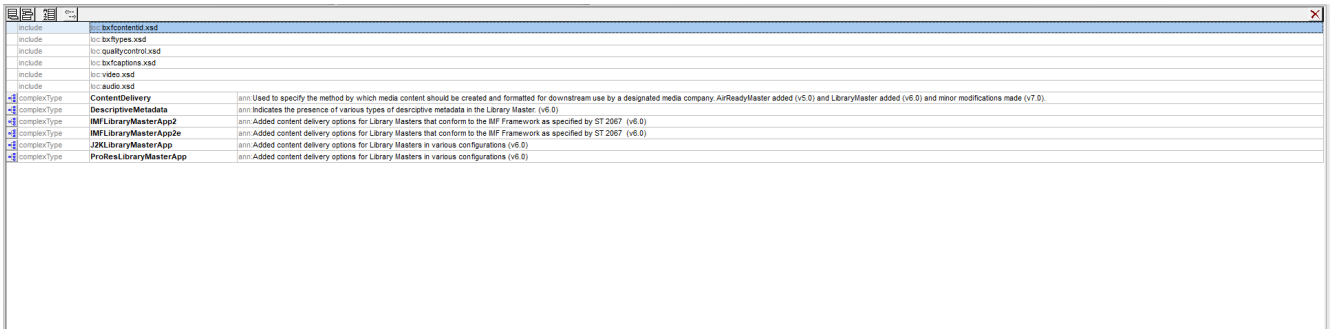


Figure 17 — ContentDelivery.xsd

4.2.3.1 BXF 8.0 Changes

Description of change: A new optional element, DeliveryDetails, has been added to ContentDelivery in order to allow the description of a DeliveryService and DeliveryDate of the media delivery, as shown in Figure 18.

Text representations

```
<xs:element name="DeliveryDetails" minOccurs="0">

<xs:annotation>

     <xs:documentation>Used to describe how the media is being delivered and when.
(v8.0)</xs:documentation>

</xs:annotation>

     <xs:complexType>

          <xs:sequence>

               <xs:element name="DeliveryService" type="BxfText" minOccurs="0"/>

               <xs:element name="DeliveryDate" type="BxfDateTime" minOccurs="0"/>

          </xs:sequence>

     </xs:complexType>

</xs:element>
```

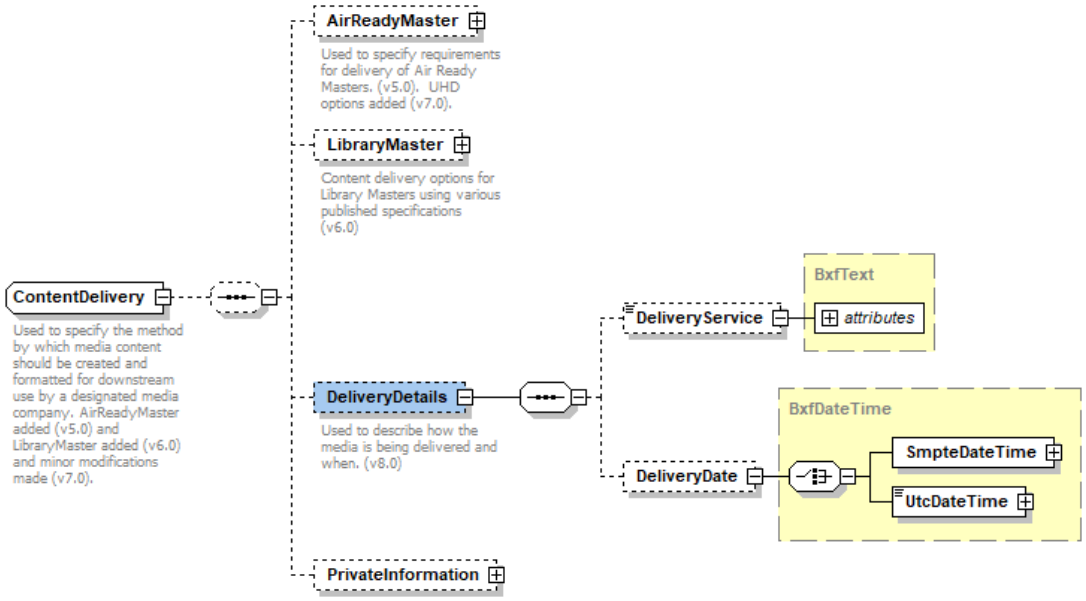


Figure 18 — ContentDelivery.xsd detail.

4.2.4 ContentMetadata.xsd(*)

Content identification and other details related to media and media location, as shown in Figure 19.

Contains:

include	loc:	audio.xsd	
include	loc:	bxfcaptions.xsd	
include	loc:	bxfcontentid.xsd	
include	loc:	datacontent.xsd	
include	loc:	location.xsd	
include	loc:	video.xsd	
include	loc:	macro.xsd	
include	loc:	bxftypes.xsd	
include	loc:	qualitycontrol.xsd	
include	loc:	contentdelivery.xsd	
complexType		**AlternateAudioContent**	ann: Used to provide alternate audio only media information
complexType		**BaseMedia**	ann: Enumerates the way content is stored or the method used to transmit
complexType		**BillBoard**	ann: Used to describe the different attributes of one or more BillBoards
complexType		**ContentDetail**	ann: Description of people, events, sports results, and word tags to search for things that represent this media (v5.0).
complexType		**ContentMetaData**	ann: Used to describe all the metadata for a single instance of content. (Note that this element name might be expected to be "ContentMetadata", but was left unchanged from previous versions in order to preserve backward compatibility. This is a documented exception as of v3.0)
complexType		**Media**	ann: Base Media combined with Media Location
complexType		**MediaLocation**	ann: Used to designate the physical location of a media essence and its quality
complexType		**UsagePolicy**	ann:

Figure 19 — ContentMetadata.xsd

4.2.4.1 BXF 8.0 Changes

Description of change: Under UsagePolicy, the optional element, FirstAirDate, has a new optional attribute, dayOfWeek, that allows the designation of the day of week (Mon-Sun order) that should be used to first air the program, as shown in Figure 20.

Text representation:

```
xs:element name="FirstAirDate" minOccurs="0">

<xs:annotation>

     <xs:documentation>First planned schedule date for this
content</xs:documentation>

</xs:annotation>

     <xs:complexType>

          <xs:simpleContent>

               <xs:extension base="xs:dateTime">

                    <xs:attribute name="dayOfWeek" type="DayPattern">

                    <xs:annotation>

                         <xs:documentation>Used to specify the actual day of week
for the first airdate. E.g., a 7 element binary representation of the days of the
week in Monday-Sunday order where a 1 includes the day and a 0 excludes the day

(e.g., 1 0 0 0 0 0 0 = Monday) (v8.0)</xs:documentation>

                    </xs:annotation>

                    </xs:attribute>

               </xs:extension>

          </xs:simpleContent>

     </xs:complexType>

</xs:element>
```

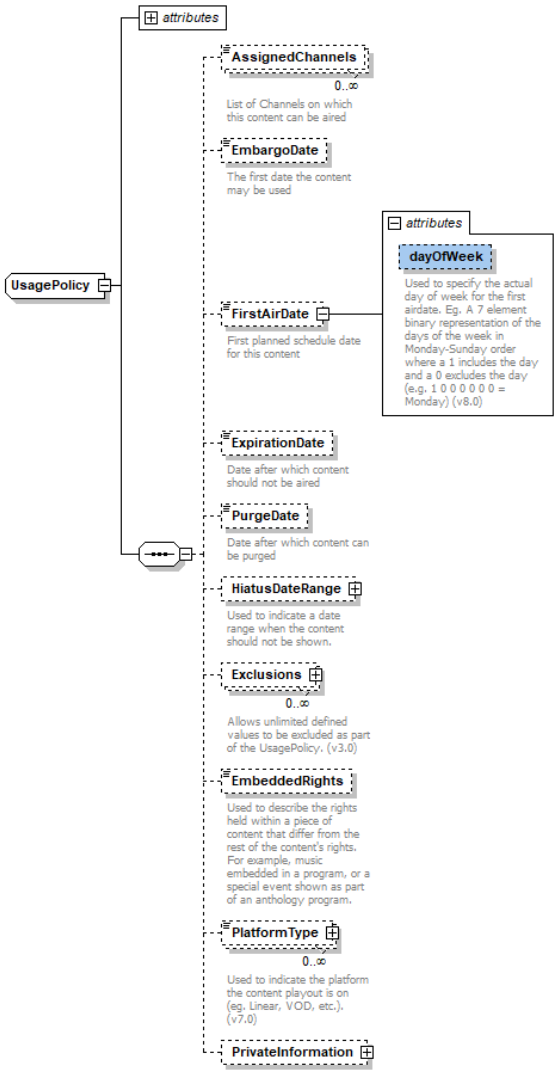
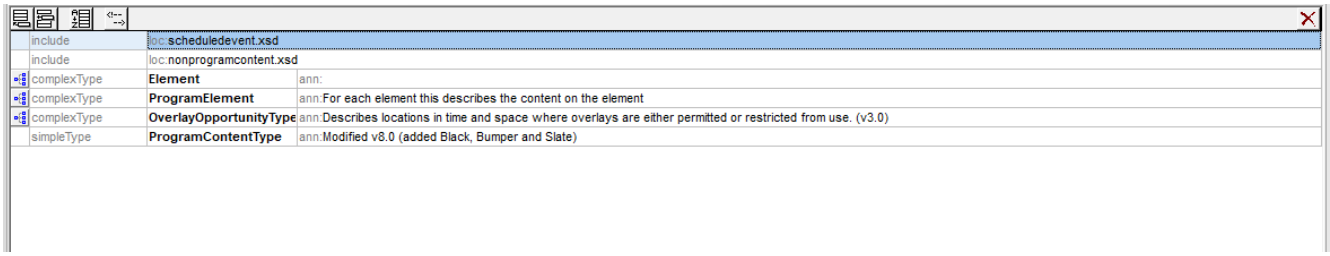


Figure 20 — UsagePolicy.xsd

4.2.5 Element.xsd(*)

If the program is being delivered in segmented form, this describes each segment's content, position and offset from the start of the program as well as other embedded content such as inserted barter content, as shown in Figure 21.

Contains:



include	loc:scheduledevent.xsd	
include	loc:nonprogramcontent.xsd	
complexType	**Element**	ann:
complexType	**ProgramElement**	ann:For each element this describes the content on the element
complexType	**OverlayOpportunityType**	ann:Describes locations in time and space where overlays are either permitted or restricted from use. (v3.0)
simpleType	**ProgramContentType**	ann:Modified v8.0 (added Black, Bumper and Slate)

Figure 21 — Element.xsd

4.2.5.1 BXF 8.0 Changes

Description of change: Under the node Element/ProgramElement/Type, three new enumerations have been added to the ProgramContentType base to allow for the selection of Black, Bumper and Slate. Note that ProgramSegment remains the default value, as shown in Figure 22.

Text representation:

```
<xs:element name="Type" default="ProgramSegment">
<xs:annotation>
     <xs:documentation>Either a program segment or an interactive element. Textless
added as a new option. (v6.0) Added Black, Bumper and Slate as new options.
(v8.0)</xs:documentation>
</xs:annotation>
     <xs:complexType>
          <xs:simpleContent>
               <xs:extension base="ProgramContentType"/>
          </xs:simpleContent>
     </xs:complexType>
</xs:element>


<xs:simpleType name="ProgramContentType">
<xs:annotation>
     <xs:documentation>Modified v8.0 (added Black, Bumper and
Slate)</xs:documentation>
</xs:annotation>
     <xs:restriction base="xs:string">
          <xs:enumeration value="Black"/>
          <xs:enumeration value="Bumper"/>
          <xs:enumeration value="Interactive"/>
          <xs:enumeration value="ProgramSegment"/>
          <xs:enumeration value="Slate"/>
          <xs:enumeration value="Textless"/>
     </xs:restriction>
</xs:simpleType>
```

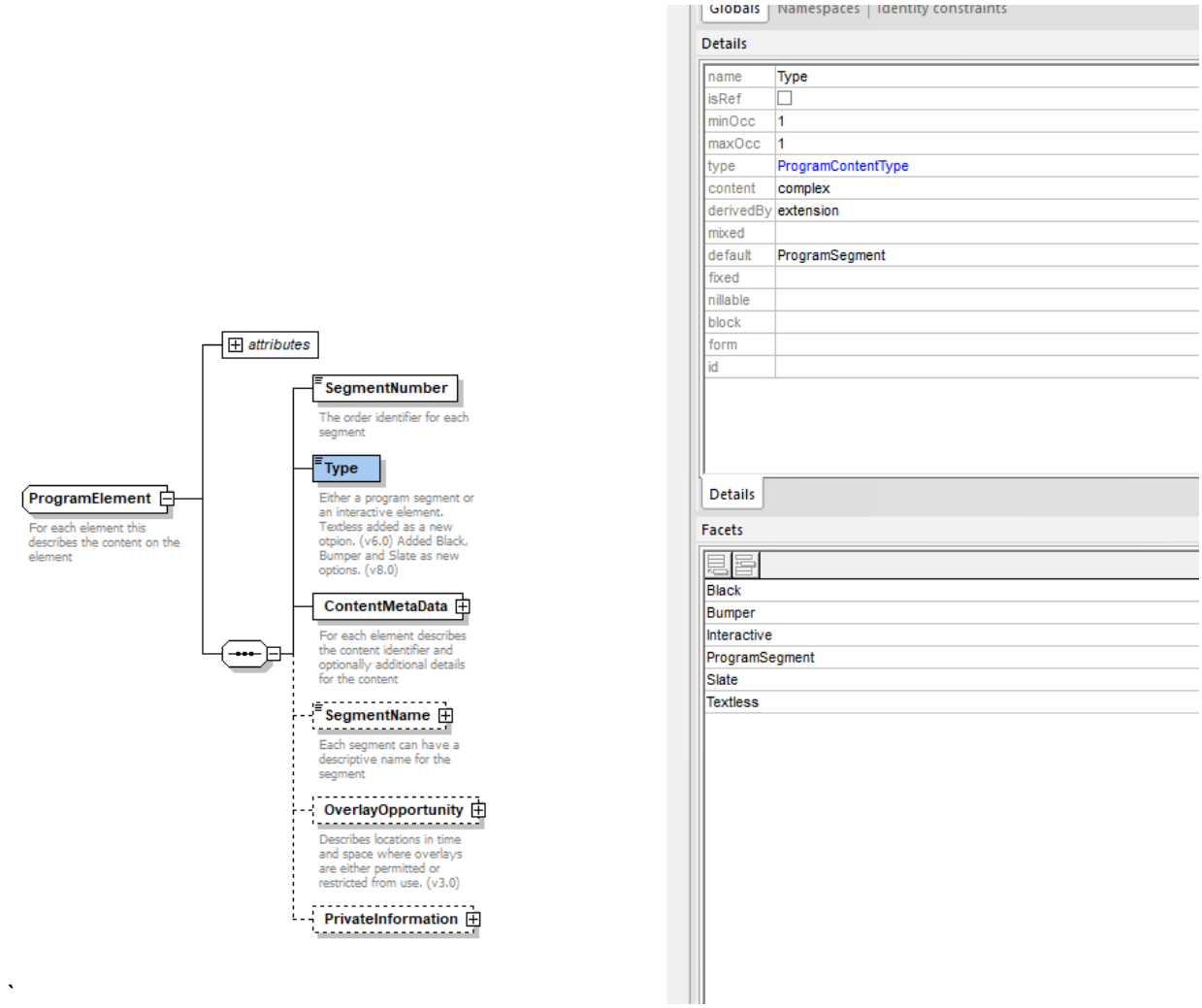


Figure 22 — ProgramElement.xsd

4.2.6 JobDetail.xsd(*)

Allows the message to assign job information. If more than one company or process is required for a job, enter multiple records and link using the jobId. If one job is dependent on the completion of another, link the jobs using jobId and jobDependency, as shown in Figure 23.

Contains:

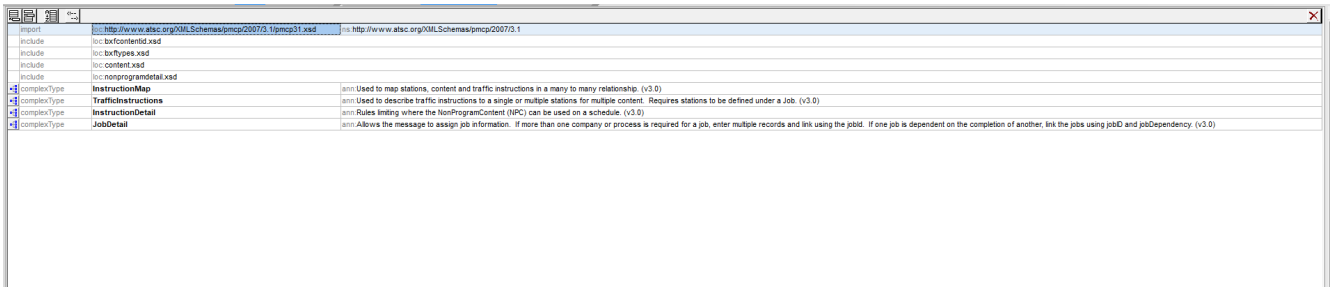


Figure 23 — JobDetail.xsd

4.2.6.1 BXF 8.0 Changes

Description of change: Under the node InstructionDetail/ContentRotation/RotationPattern, a modification to the annotation was made in order to clarify the way to use the field, but no functional change was made, as shown in Figure 24. A new optional element, RotationGroup, was added with the type xs:integer to allow the grouping of Bookends and Piggybacks. When paired using the same integer number in this field the traffic system can determine that two or more content records are related to one another. The actionType attribute for InstructionDetail was also extended to include the "Unchanged" option.

Text representation:

```
<xs:element name="RotationPattern" type="xs:string" minOccurs="0">

<xs:annotation>

     <xs:documentation>String of numbers (11223) or letters (ABBC) (with commas
optionally) indicating content order of placement by traffic system. (Annotation Only
v8.0)</xs:documentation>

</xs:annotation>

</xs:element>

…

<xs:element name="RotationGroup" type="xs:integer" minOccurs="0">

<xs:annotation>

     <xs:documentation>Used to group Bookends and Piggybacks.
(v8.0)</xs:documentation>

</xs:annotation>

</xs:element>



<xs:attribute name="actionType" use="required">

<xs:annotation>

     <xs:documentation>The type of action the receiving system could take with the
delineated instructions for the specific schedule including adding new instructions,
revising existing instructions or deleting previous instructions. Updated to include
Unchanged option in (V8.0).</xs:documentation>

</xs:annotation>

     <xs:simpleType>

          <xs:restriction base="xs:string">
```

```
                <xs:enumeration value="Add"/>
                <xs:enumeration value="Delete"/>
                <xs:enumeration value="Revision"/>
                <xs:enumeration value="Unchanged"/>
            </xs:restriction>
        </xs:simpleType>
</xs:attribute>
```

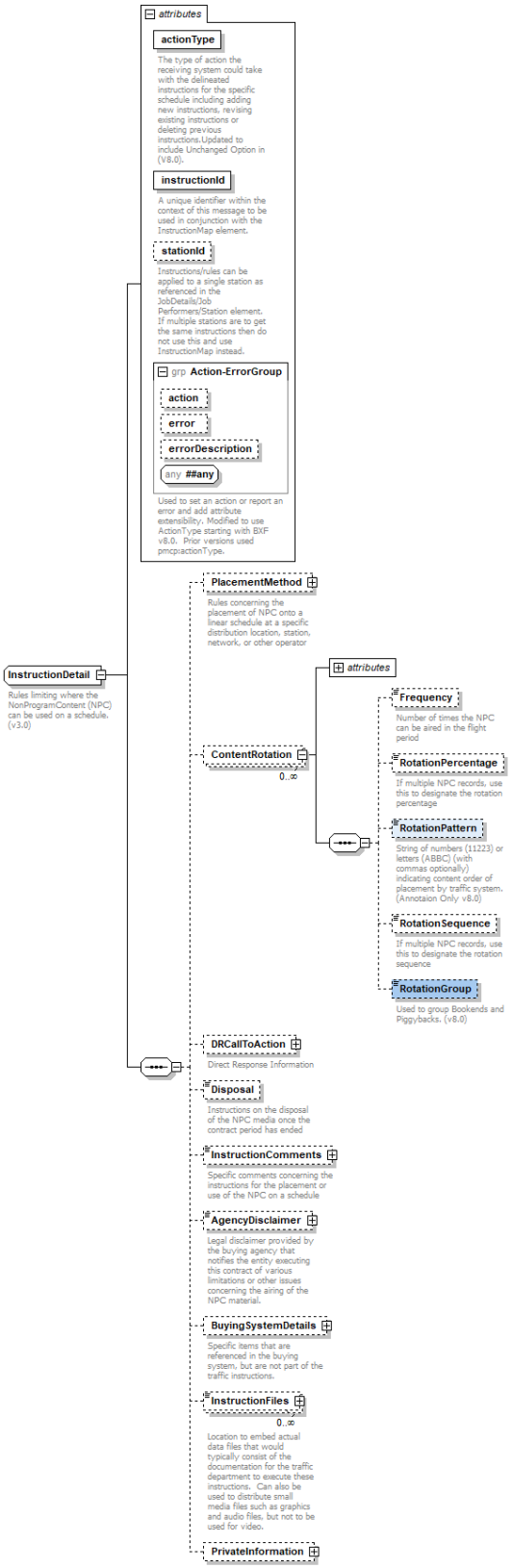


Figure 24 — InstructionDetail.xsd

4.2.7 NonProgramDetail.xsd(*)

Details of the non-program content, as shown in Figure 25.

Contains:

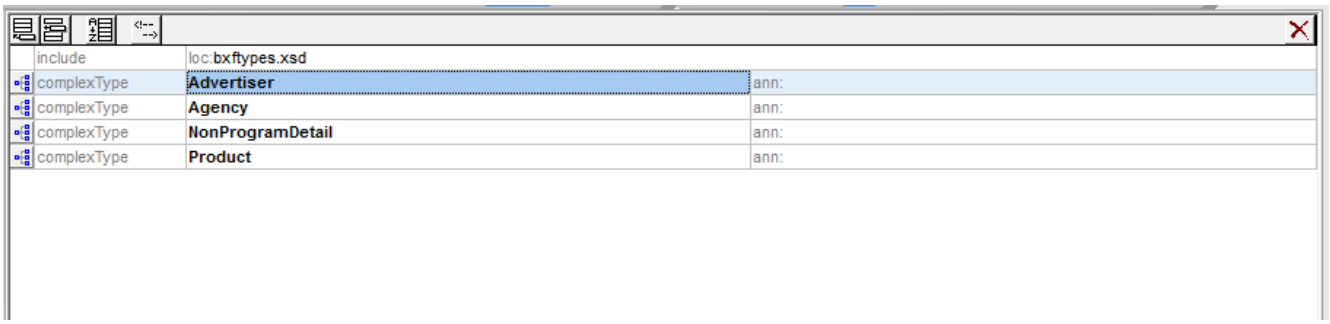


Figure 25 — NonProgramDetail.xsd

4.2.7.1 BXF 8.0 Changes

Description of change: AdType was modified to include three new enumeration options: Fee Spot, Double Barter and Optional Barter, as shown in Figure 26.

Text representation:

```
<xs:complexType name="NonProgramDetail">

     <xs:sequence>

          <xs:element name="AdType">

          <xs:annotation>

               <xs:documentation>Used to describe the type of advertisement being
placed (e.g., Commercial, General, Promo, PSA, etc.). Note: added "Commercial" as an
additional enumeration to be explicit from the General option. (v6.0) Added Fee Spot,
Double Barter and Optional Barter (v8.0)</xs:documentation>

          </xs:annotation>

               <xs:simpleType>

                    <xs:restriction base="xs:string">

                         <xs:enumeration value="Commercial"/>

                         <xs:enumeration value="Direct Response"/>

                         <xs:enumeration value="Double Barter"/>

                         <xs:enumeration value="Fee Spot"/>

                         <xs:enumeration value="General"/>

                         <xs:enumeration value="Paid Program"/>
```

```
                        <xs:enumeration value="Promo"/>
                        <xs:enumeration value="PSA"/>
                        <xs:enumeration value="Optional Barter"/>
                        <xs:enumeration value="Trade/Barter"/>
                        <xs:enumeration value="Other"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
…
        </xs:sequence>
        <xs:attributeGroup ref="Action-ErrorGroup"/>
</xs:complexType>
```

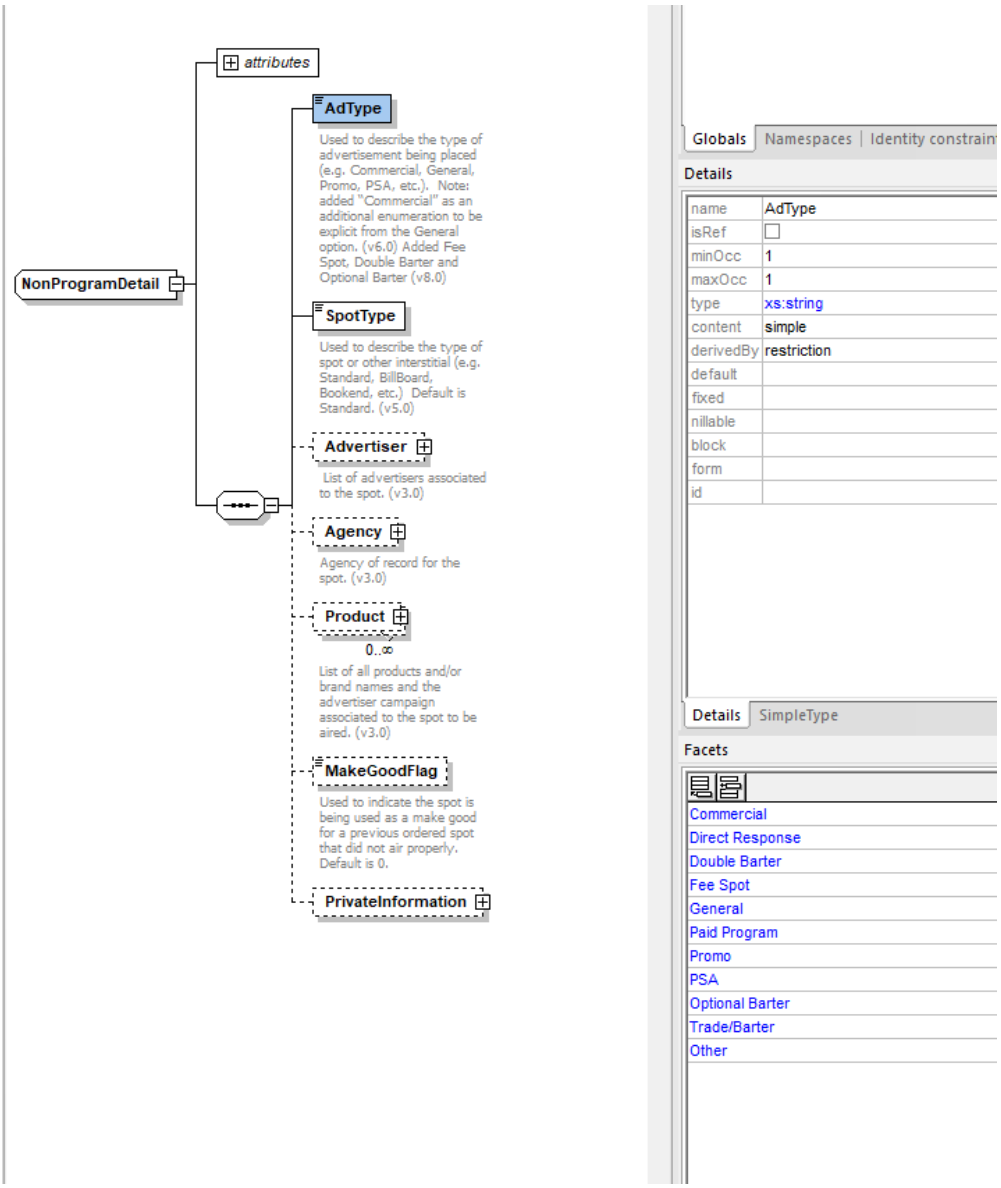
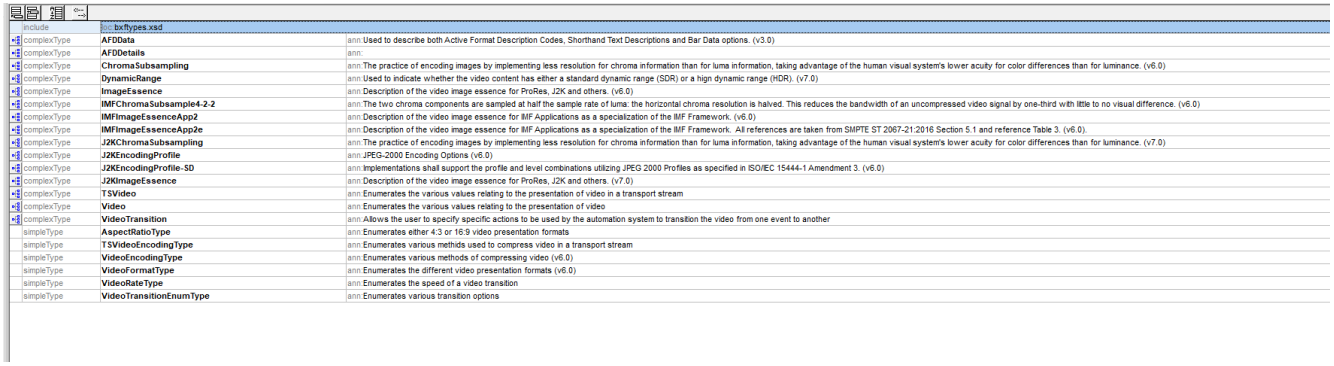


Figure 26 — NonProgramDetail.xsd detail.

4.2.8 Video.xsd(*)

Enumerates the various values relating to the presentation of video, as shown in Figure 27.

Contains:



include	bxftypes.xsd	
complexType	AFDData	ann: Used to describe both Active Format Description Codes, Shorthand Text Descriptions and Bar Data options. (v3.0)
complexType	AFDDetails	ann:
complexType	ChromaSubsampling	ann: The practice of encoding images by implementing less resolution for chroma information than for luma information, taking advantage of the human visual system's lower acuity for color differences than for luminance. (v6.0)
complexType	DynamicRange	ann: Used to indicate whether the video content has either a standard dynamic range (SDR) or a hign dynamic range (HDR). (v7.0)
complexType	ImageEssence	ann: Description of the video image essence for ProRes, J2K and others. (v6.0)
complexType	IMFChromaSubsample4-2-2	ann: The two chroma components are sampled at half the sample rate of luma: the horizontal chroma resolution is halved. This reduces the bandwidth of an uncompressed video signal by one-third with little to no visual difference. (v6.0)
complexType	IMFImageEssenceApp2	ann: Description of the video image essence for IMF Applications as a specialization of the IMF Framework. (v6.0)
complexType	IMFImageEssenceApp2e	ann: Description of the video image essence for IMF Applications as a specialization of the IMF Framework. All references are taken from SMPTE ST 2067-21:2016 Section 5.1 and reference Table 3. (v6.0).
complexType	J2KChromaSubsampling	ann: The practice of encoding images by implementing less resolution for chroma information than for luma information, taking advantage of the human visual system's lower acuity for color differences than for luminance. (v7.0)
complexType	J2KEncodingProfile	ann: JPEG-2000 Encoding Options (v6.0)
complexType	J2KEncodingProfile-SD	ann: Implementations shall support the profile and level combinations utilizing JPEG 2000 Profiles as specified in ISO/IEC 15444-1 Amendment 3. (v6.0)
complexType	J2KImageEssence	ann: Description of the video image essence for ProRes, J2K and others. (v7.0)
complexType	TSVideo	ann: Enumerates the various values relating to the presentation of video in a transport stream
complexType	Video	ann: Enumerates the various values relating to the presentation of video
complexType	VideoTransition	ann: Allows the user to specify specific actions to be used by the automation system to transition the video from one event to another
simpleType	AspectRatioType	ann: Enumerates either 4:3 or 16:9 video presentation formats
simpleType	TSVideoEncodingType	ann: Enumerates various methids used to compress video in a transport stream
simpleType	VideoEncodingType	ann: Enumerates various methods of compressing video (v6.0)
simpleType	VideoFormatType	ann: Enumerates the different video presentation formats (v6.0)
simpleType	VideoRateType	ann: Enumerates the speed of a video transition
simpleType	VideoTransitionEnumType	ann: Enumerates various transition options

Figure 27 — Video.xsd

4.2.8.1 BXF 8.0 Changes

Description of change: A new optional element, Watermarking, was added to Video that can be used to indicate that watermarking is present in the video, however, only one option of Other is allowed to indicate which method of watermarking is present, as shown in Figure 28. It is hoped that additional options will be added in future BXF versions.

Text representation:

```
<xs:element name="Watermarking" minOccurs="0" maxOccurs="unbounded">

<xs:annotation>

     <xs:documentation>Indicates the type(s) of watermarking applied to the video.
(v8.0)</xs:documentation>

</xs:annotation>

     <xs:simpleType>

          <xs:restriction base="xs:string">

               <xs:enumeration value="Other"/>

          </xs:restriction>

     </xs:simpleType>

</xs:element>
```

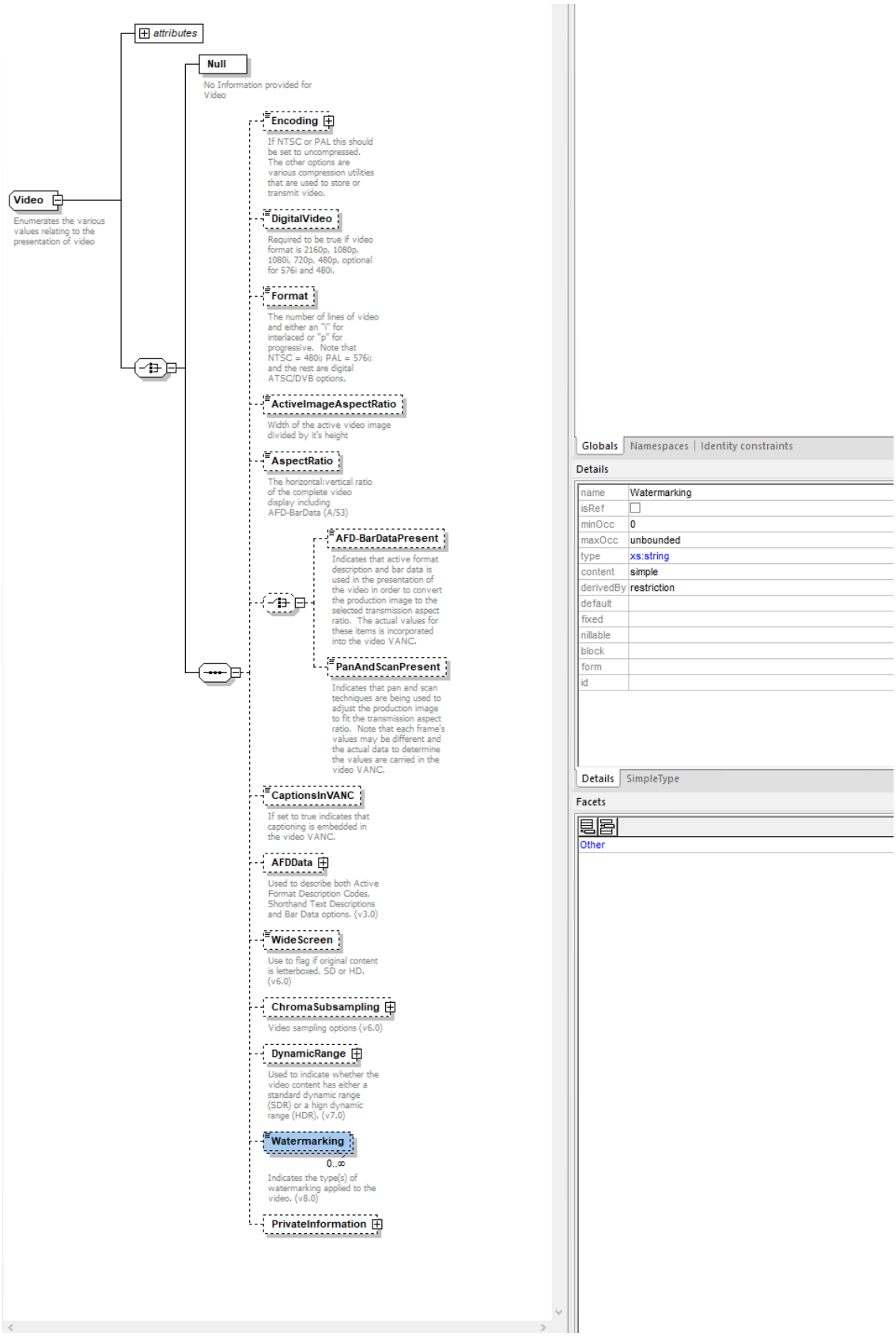


Figure 28 — Video.xsd detail.

Annex A Additional elements (Informative)

ST2021-4a-2022.zip – Schema XSD files

ST2021-4b-2022.zip – HTML representation of the schema