INTERNATIONAL STANDARD

ISO/IEC 15938-5

First edition
2003-05-15

AMENDMENT 5
2015-11-15

Information technology — Multimedia content description interface —

Part 5: Multimedia description schemes

AMENDMENT 5: Quality metadata, multiple text encodings, extended classification metadata

Technologies de l'information — Interface de description du contenu multimédia —

Partie 5: Schémas de description multimédia

AMENDEMENT 5: Métadonnées de qualité, encodages de texte multiples, métadonnées de classification étendues

Reference number
ISO/IEC 15938-5:2003/Amd.5:2015(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

Introduction

The Multimedia Preservation Application Format (MP-AF, ISO/IEC 23000-15) makes use of MPEG-7 MDS for descriptive and technical metadata. The following gaps in MPEG-7 MDS for preservation applications are addressed by this amendment:

- Support for identifying the script of a string. This is useful in cases where a language can be represented in multiple scripts (e.g., Japanese, Bulgarian). While the script identifier according to ISO 15924 can be represented as part of the language tag, introducing a separate attribute improves the modularity of the metadata to represent it separately.

- Support for multiple string encodings in a single XML document. In some cases strings (e.g., identifiers, file names) need to be kept in their original character encoding different to the encoding of the metadata document. In order to enable this and avoid problems with XML parsing, the content of the string is then base64 or percent encoded and the original character encoding is identified.

- Extended classification metadata, i.e. increasing the cardinality of some classification metadata elements and adding the option to identify the version of the content.

- Metadata for quality control. Detailed quality descriptors are only available for audio (MPEG-7 part 4/AMD 2), while similar tools for format/wrapper and visual quality checks and defects are missing. A representation compatible with the European Broadcasting Union's (EBU) Quality Control data model is needed.

This amendment defines solutions for filling these gaps in MDS. These extensions are expected to be useful also for applications beyond preservation.

The extensions are all optional, thus descriptions conforming to a version of MPEG-7 MDS prior to this amendment are valid against a version including this amendment.

Information Technology — Multimedia content description interface — Part 5: Multimedia description schemes, AMENDMENT 5: Quality metadata, multiple text encodings, extended classification metadata

1 Scope

This amendment extends text and classification descriptors for descriptive metadata for audiovisual content making use of existing datatypes and adding some elements and attributes, and adds support for quality metadata. A first extension improves internationalization for text-based metadata by identifying the script(s) used and enables carrying encoded strings represented in multiple character sets in the same XML document. A second improvement extends classification metadata in order to: (a) improve interoperability with existing metadata formats used in professional media production, (b) increase cardinality of some elements and (c) add attributes and a version element. Finally a third extension improves the tools for describing profiles for quality control of audiovisual media, as well as the results of quality analysis on wrapper, bitstream and baseband level.

NOTE The XML schema implementing this amendment as well as a consolidated XML schema for MPEG-7 including this amendment are contained in the electronic attachment.

2 Multiple Text Encodings

In 7.2.2, replace the entire text of the subclause with the following text:

7.2.2 Textual datatypes

7.2.2.1 Introduction

The `TextualBase` datatype provides a base from which other kinds of textual description tools are derived. It can also be used in the definitions of tools that contain textual descriptions. An optional attribute, xml:lang, is used to specify the language of the contained text.

The `Textual` datatype, derived from the `TextualBase` datatype, is a non-abstract type that is used for simple textual descriptions.

In cases where textual elements need to be stored in their native character set (different from the encoding of the XML document), the character set can be specified using the charset attribute. In order to allow for correct parsing of the document, the content of the textual element must be encoded in this case (using base64 or percent encoding).

7.2.2.2 Textual datatypes syntax

```
<!-- ########################################## -->
<!--  Definition of Textual datatypes (7.2.2)  -->
<!-- ########################################## -->
<!-- Definition of TextualBase datatype -->
<complexType name="TextualBaseType" abstract="true">
```

```
    <simpleContent>
        <extension base="string">
            <attribute ref="xml:lang" use="optional"/>
            <attribute name="phoneticTranscription" use="optional">
                <simpleType>
                    <list itemType="mpeg7:PhoneType"/>
                </simpleType>
            </attribute>
            <attribute name="phoneticAlphabet" type="mpeg7:phoneticAlphabetType"
                use="optional" default="sampa"/>
            <attribute name="charset" type="string" use ="optional"/>
            <attribute name="encoding" type="string" use ="optional"/>
            <attribute name="script" type="string" use ="optional"/>
        </extension>
    </simpleContent>
</complexType>

<!-- Definition of Textual datatype -->
<complexType name="TextualType">
    <simpleContent>
        <extension base="mpeg7:TextualBaseType"/>
    </simpleContent>
</complexType>
```

7.2.2.3 Textual datatypes semantics

Semantics of the TextualBaseType:

Name	Definition
`TextualBaseType`	Abstract base type for textual descriptions. This type includes the `xml:lang` attribute for identifying the language in which the description is written and can contain a phonetic transcription of the text
`xml:lang`	Indicates the language in which the textual description is written (optional). If no value is specified for this attribute, the value of the xml:lang attribute is inherited from the closest containing element that specifies a value for xml:lang.
`phoneticTranscription`	Describes a phonetic transcription of the text using a phonetic alphabet. The transcription is represented as a list of phones (i.e. `PhoneType`). The defintion for `PhoneType` is given in ISO/IEC 15938-4.
`phoneticAlphabet`	Describes the phonetic alphabet that is used for the transcription in phoneticTranscription, e.g. IPA (International Phonetic Association alphabet). The specification of the `phoneticAlphabetType` is given in ISO/IEC 15938-4.
`charset`	Specifies the character set of the string, if it does not conform to the character set of the XML document. In this case, the content of `TextualBaseType` must be encoded, and the `encoding` attribute must be present.
`encoding`	The encoding used to represent string in a charset different than the global character set of the XML document. Recommended

Name	Definition
	encodings are `base64` or `percent`.
`script`	Specifies the script used for representing the string, using the values defined in ISO 15924. This allows the correct identification of the script being used for languages that can be represented in multiple scripts. For modularity of representation, a separate attribute is provided as an alternative to coding the script in the `xml:lang` attribute. It is recommended to use the script attribute.

Semantics of the `TextualType`:

Name	Definition
`TextualType`	Describes a textual description. `TextualType` that extends the abstract `TextualBaseType`. This type includes the `xml:lang` attribute for identifying the language in which the description is written and can contain a phonetic transcription of the text

7.2.2.4 Textual datatypes example (informative)

The following example description excerpt shows the use of `Textual datatype` for describing the free text in English: "The Giants win the pennant."

```
<Textual xml:lang="en">
   The Giants win the pennant.
</Textual>
```

The following example description excerpt shows the use of `Textual datatype` for describing the name of the city of Moscow in Russian, using Cyrillic script.

```
<Textual xml:lang="ru" script="Cyrl">Москва</Textual>
```

3 Extended Classification Metadata

In 9.2.3, replace the entire text of the subclause with the following text:

9.2.3 Classification DS

9.2.3.1 Introduction

The `Classification DS` describes the classification of the multimedia content. The resulting descriptions facilitate searching and filtering of multimedia content based on user preferences (e.g, language, style, genre, and so forth) and service-oriented classifications (e.g, purpose, parental guidance, market segmentation, media review, and so forth).

9.2.3.2 Classification DS syntax

```
<!-- ######################################### -->
<!--  Definition of Classification DS (9.2.3)  -->
```

```
<!-- ######################################### -->

<!-- Definition of Classification DS  -->
<complexType name="ClassificationType">
    <complexContent>
        <extension base="mpeg7:DSType">
            <sequence>
                <element name="Form" type="mpeg7:ControlledTermUseType" minOccurs="0"
                    maxOccurs="unbounded" />
                <element name="Genre" minOccurs="0" maxOccurs="unbounded">
                    <complexType>
                       <complexContent>
                          <extension base="mpeg7:ControlledTermUseType">
                             <attribute name="type" use="optional" default="main">
                                <simpleType>
                                   <restriction base="NMTOKEN">
                                      <enumeration value="main"/>
                                      <enumeration value="secondary"/>
                                   </restriction>
                                </simpleType>
                             </attribute>
                             <attribute name="confidence" use="optional"
                                type="mpeg7:zeroToOneType" />
                          </extension>
                       </complexContent>
                    </complexType>
                </element>
                <element name="Subject" type="mpeg7:TextAnnotationType"
                       minOccurs="0" maxOccurs="unbounded"/>
                <element name="Purpose" type="mpeg7:ControlledTermUseType"
                       minOccurs="0" maxOccurs="unbounded"/>
                <element name="Language" type="mpeg7:ExtendedLanguageType"
                       minOccurs="0" maxOccurs="unbounded"/>
                <element name="CaptionLanguage" minOccurs="0" maxOccurs="unbounded">
                    <complexType>
                       <simpleContent>
                          <extension base="language">
                             <attribute name="closed" type="boolean" use="optional"
                                default="true"/>
                             <attribute name="supplemental" type="boolean"
                                use="optional" default="false"/>
                          </extension>
                       </simpleContent>
                    </complexType>
                </element>
                <element name="SignLanguage" minOccurs="0" maxOccurs="unbounded">
                    <complexType>
                       <simpleContent>
                          <extension base="language">
                             <attribute name="primary" type="boolean" use="optional"/>
                             <attribute name="translation" type="boolean"
                                use="optional"/>
                          </extension>
                    </simpleContent>
                    </complexType>
                </element>
                <element name="Release" minOccurs="0" maxOccurs="unbounded">
                    <complexType>
                       <sequence>
                       <element name="Region" type="mpeg7:regionCode"
```

```
                        minOccurs="0" maxOccurs="unbounded"/>
                    </sequence>
                    <attribute name="date" type="mpeg7:timePointType"
                            use="optional"/>
                </complexType>
            </element>
            <element name="Target" minOccurs="0">
                <complexType>
                    <sequence>
                        <element name="Market" type="mpeg7:ControlledTermUseType"
                            minOccurs="0" maxOccurs="unbounded"/>
                        <element name="Age" minOccurs="0">
                            <complexType>
                                <attribute name="min" type="nonNegativeInteger"
                                    use="optional"/>
                                <attribute name="max" type="nonNegativeInteger"
                                    use="optional"/>
                            </complexType>
                        </element>
                        <element name="Region" type="mpeg7:regionCode"
                            minOccurs="0" maxOccurs="unbounded"/>
                    </sequence>
                </complexType>
            </element>
            <element name="ParentalGuidance" type="mpeg7:ParentalGuidanceType"
                    minOccurs="0" maxOccurs="unbounded"/>
            <element name="MediaReview" type="mpeg7:MediaReviewType"
                    minOccurs="0" maxOccurs="unbounded"/>
            <element name="Version" type="mpeg7:TextAnnotationType" minOccurs="0" />
        </sequence>
    </extension>
  </complexContent>
</complexType>

<!-- Definition of ExtendedLanguage datatype -->
<complexType name="ExtendedLanguageType">
    <simpleContent>
        <extension base="language">
            <attribute name="type" use="optional" default="original">
                <simpleType>
                    <restriction base="NMTOKEN">
                        <enumeration value="original"/>
                        <enumeration value="dubbed"/>
                        <enumeration value="background"/>
                    </restriction>
                </simpleType>
            </attribute>
            <attribute name="supplemental" type="boolean"
                use="optional" default="false"/>
        </extension>
    </simpleContent>
</complexType>

<!-- Definition of ParentalGuidance datatype -->
<complexType name="ParentalGuidanceType">
    <sequence>
        <choice>
            <element name="ParentalRating" type="mpeg7:ControlledTermUseType"/>
            <element name="MinimumAge" type="nonNegativeInteger"/>
        </choice>
```

```
        <element name="Region" type="mpeg7:regionCode"
                minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
</complexType>

<!-- Definition of MediaReview datatype -->
<complexType name="MediaReviewType">
    <sequence>
        <element name="Rating" type="mpeg7:RatingType" minOccurs="0"/>
        <element name="FreeTextReview" type="mpeg7:TextualType"
                minOccurs="0" maxOccurs="unbounded"/>
        <element name="ReviewReference" type="mpeg7:RelatedMaterialType"
                minOccurs="0"/>
        <element name="Reviewer" type="mpeg7:AgentType" minOccurs="0"/>
    </sequence>
</complexType>
```

9.2.3.3 Classification DS semantics

Semantics of the `ClassificationType`:

Name	*Definition*
`ClassificationType`	Describes the classification of the multimedia content.
`Form`	Describes the production type of the document, such as, film, news program, magazine, documentary, etc (optional). An example of CS is `FormatCS`.
`Genre`	Describes what the multimedia content is about (broad classification), such as sports, politics, economics, etc (optional). An example of CS is the `GenreCS`.
`type`	Indicates the type of the genre of the multimedia content. The types of genres are defined as follows. • *main* – The specified genre is the main, or primary. This is the default value. • *secondary* – The specified genre is a secondary genre, such as a subgenre.
`confidence`	Confidence of determined genre (optional).
`Subject`	Describes the subjects (specific classifications) of the multimedia content (optional). The subjects allows a textual annotation to classify the multimedia content.
`Purpose`	Describes one or more purposes for which the multimedia content was created (optional). An example of CS is `IntentionCS`.
`Language`	Describes one or more languages of the spoken audio of the program (optional).
`CaptionLanguage`	Describes one or more languages of the caption information included with the program (optional). The type of the caption information associated with the program is denoted by the closed attribute. Closed

Name	*Definition*
	captions can be turned on or off by the user, while open captions (or subtitles) are part of the picture itself and remain visible.
`closed`	Indicates whether the specified caption is closed. Default value of the attribute is true; if the attribute is set to false, then the provided caption description refers to open captions/subtitles.
`supplemental`	Indicates whether the captions provide descriptions of the scene for the benefit of hearing or visually impaired, in addition to a direct translation of the spoken words. Closed captions may include such descriptive information, such as speaker identification, and non-speech sounds that would be missed. Default value of this attribute is *false*.
`SignLanguage`	Specifies the audio sign language provided for the multimedia content, and, optionally, qualifies the use of signing as a primary language or as a translation of the spoken dialogue (optional).
`primary`	Indicates if the sign language is the primary language of the content or not, i.e, if the content is produced specifically for the hearing impaired or not (optional).
`translation`	Indicates if the sign language is a translation of the spoken dialogue or not (optional).
`Release`	Describes the release date and region of the multimedia content (optional).
`Region`	Indicates the countries or regions in which the multimedia content was first released (optional). This locator may be different than the location where it was created.
`date`	Indicates the date on which the multimedia content was first released. This date may be different than the date(s) when it was created (optional).
`Target`	Describes the target of the multimedia content in terms of market classification, age and country or region (optional).
`Market`	Describes targeted markets of the multimedia content (optional). An example of CS is `TargetGroupCS`.
`Age`	Describes the targeted age range of the multimedia content (optional).
`min`	Indicated the minimum age of the targeted age range (optional).
`max`	Indicated the maximum age of the targeted age range (optional).
`Region`	Describes target countries or regions for the multimedia content (optional).
`ParentalGuidance`	Describes parental guidance classification of the multimedia content (optional).
`MediaReview`	Describes media reviews about the multimedia content (optional).

Name	*Definition*
`Version`	The version of the content (optional).

Semantics of the `ExtendedLanguageType`:

Name	*Definition*
`ExtendedLanguageType`	Specifies the language of spoken audio of the multimedia content.
`type`	Indicates whether the spoken language is the original language of the content, or has been dubbed. The following values are allowed. • *original* – The specified language is an original language of the spoken content. This is the default value. • *dubbed* – The specified language is a dubbed language of the spoken content. • *background* – The specified language is the original language heard in the background, while another (dubbed) language is heard in the foreground. Other values that are datatype-valid with respect to `mpeg7:termReferenceType` are reserved.
`supplemental`	Indicates whether the specified language exclusively provides an audio description of the multimedia content, for the benefit of the visually impaired. Default value of this attribute is false.

Semantics of the `ParentalGuidanceType`:

Name	*Definition*
`ParentalGuidanceType`	Describes parental guidance classification.
`ParentalRating`	Indicates the rating value according to the rating scheme. Examples of CSs are `JapanRatingCS`, `ICRAParentalRatingCS`, `MPAAParentalRatingCS`, `ICRAParentalRatingNudityCS`, `RIAAParentalRatingCS`, `ICRAParentalRatingSexCS`, `MPAAParentalRatingTVCS`, and `IRCAParentalRatingViolenceCS`.
`MinimumAge`	Indicates the minimum recommended age in years of the end user.
`Region`	Indicates the country or region for which the parental guidance description is specified (optional). If not specified, the parental guidance applies worldwide.

Semantics of the `MediaReviewType`:

Name	*Definition*
`MediaReviewType`	Describes the Media Review about the multimedia content.
`Rating`	Specifies the rating value and criterion used in the review (optional).
`FreeTextReview`	Describes a free-text review of the multimedia content without reference to a rating scheme (optional). There can be multiple instances of the review in different languages.
`ReviewReference`	Describes the Media Locator of the material from where the review may have been extracted or quoted (optional). For example, the TV magazine that published the review, the Films Guide from where the review was quoted, an interview from where the review was transcribed.
`Reviewer`	Describes the reviewer/critic of the multimedia content (optional). Since it is defined as `AgentType`, this field can describe a person, a group of persons or an organization.

4 Quality Control Metadata

Replace subclause 8.2.6 with the following text (allowing QualityRating to be optional):

8.2.6 Media Quality D

8.2.6.1 Introduction

The `MediaQuality D` describes quality rating information of multimedia content. The quality of AV content may decrease when the signal is compressed, transmitted, or converted. The MediaQuality D describes both subjective and objective quality ratings.

8.2.6.2 MediaQuality D syntax

```
<!-- ######################################## -->
<!-- Definition of MediaQuality D (8.2.6) -->
<!-- ######################################## -->
<!-- Definition of MediaQuality D -->
<complexType name="MediaQualityType">
  <complexContent>
    <extension base="mpeg7:DType">
      <sequence>
        <element name="QualityRating" minOccurs="0" maxOccurs="unbounded">
          <complexType>
            <complexContent>
              <extension base="mpeg7:RatingType">
                <attribute name="type" use="required">
                  <simpleType>
                    <restriction base="NMTOKEN">
                      <enumeration value="subjective"/>
                      <enumeration value="objective"/>
                    </restriction>
```

```
                </simpleType>
              </attribute>
            </extension>
          </complexContent>
        </complexType>
      </element>
      <element name="RatingSource" type="mpeg7:AgentType" minOccurs="0"/>
      <element name="RatingInformationLocator" type="mpeg7:ReferenceType"
               minOccurs="0" maxOccurs="unbounded"/>
      <element name="PerceptibleDefects" minOccurs="0">
        <complexType>
          <sequence>
            <element name="VisualDefects" type="mpeg7:ControlledTermUseType"
                     minOccurs="0" maxOccurs="unbounded"/>
            <element name="AudioDefects" type="mpeg7:ControlledTermUseType"
                     minOccurs="0" maxOccurs="unbounded"/>
          </sequence>
        </complexType>
      </element>
    </sequence>
  </extension>
 </complexContent>
</complexType>
<!-- Definition of Rating datatype -->
<complexType name="RatingType">
 <sequence>
  <element name="RatingValue" type="float"/>
  <element name="RatingScheme">
    <complexType>
      <complexContent>
        <extension base="mpeg7:TermUseType">
          <attribute name="best" type="float" use="optional"/>
          <attribute name="worst" type="float" use="optional"/>
          <attribute name="style" use="required">
            <simpleType>
              <restriction base="NMTOKEN">
                <enumeration value="higherBetter"/>
                <enumeration value="lowerBetter"/>
              </restriction>
            </simpleType>
          </attribute>
        </extension>
      </complexContent>
    </complexType>
  </element>
 </sequence>
</complexType>
```


8.2.6.3 MediaQuality D semantics

Semantics of the `MediaQualityType`:

Name	*Definition*
`MediaQualityType`	Describes the media quality of the Media Profile.
`QualityRating`	Describes the rating values and the criterion used to create the media quality ratings (optional).
`type`	Describes the type of media quality rating. The types of quality rating are defined as follows.

Name	*Definition*
	• subjective – The rating is subjective, that is, it is provided by human viewers. • objective – The rating is objective, that is, it is acquired using computational means.
`RatingSource`	Describes the source that provides the ratings (optional).
`RatingInformationLocator`	Indicates the locator for additional information about the quality rating method (optional).
`PerceptibleDefects`	Describes defects that are perceived in the media (optional).
`VisualDefects`	Describes the visual errors perceived in the media (optional). Errors are listed in the order of descending severity. An example of CS is `VisualDefectsCS`.
`AudioDefects`	Describes the audio errors perceived in the media (optional). Errors are listed in the order of descending severity. An example of CS is `AudioDefectsCS`.

Semantics of the `RatingType`:

Name	*Definition*
`RatingType`	Describes the rating value and the criterion used to create the ratings. For example, media quality rating and media reviews.
`RatingValue`	Indicates the actual numerical rating values assigned to the multimedia content.
`RatingScheme`	Describes the name of the rating method used to obtain the quality rating. An example of CS for media quality rating is `QualityRatingSchemeCS`.
`best`	Indicates the numerical rating value that means the best rating (optional).
`worst`	Indicates the numerical rating value that means the worst rating (optional).
`style`	Describes the relationship between the numerical rating values and media quality. The types of rating styles are defined as follows. • higherBetter – The rating style indicates that a higher value means better. • lowerBetter– The rating style indicates that a lower value means better.

Add the following subclause after 8.2.8:

8.2.9 Extended Media Quality Description Tools

This subclause describes a framework for representing metadata related to automatic or semi-automatic quality control. It provides tools for representing profiles specifying quality control steps to be performed as well as tools for describing results of quality control of media.

8.2.9.1 ExtendedMediaQuality D

8.2.9.1.1 Introduction

The extended media quality D is the container for a list of quality analysis item descriptors. It is proposed as an extension to MediaQualityD, in order to enable attaching it to any element of SegmentType or its derived types.

8.2.9.1.2 ExtendedMediaQuality datatype syntax

```
<!-- ############################################################ -->
<!-- Definition of ExtendedMediaQuality D                         -->
<!-- ############################################################ -->
<complexType name="ExtendedMediaQualityType">
    <complexContent>
        <extension base="mpeg7:MediaQualityType">
            <sequence>
                <element name="QCProfile" type="mpeg7:QCProfileType" minOccurs="0"
                        maxOccurs="1" />
                <element name="QCItemResult" type="mpeg7:QCItemResultType" minOccurs="0"
                        maxOccurs="unbounded" />
                <element name="Operator" type="mpeg7:PersonType" minOccurs="0"/>
                <element name="UsedTool" type="mpeg7:CreationToolType" minOccurs="0"/>
                <element name="Annotation" type="mpeg7:TextAnnotationType" minOccurs="0"/>
            </sequence>
        </extension>
    </complexContent>
</complexType>
```

8.2.9.1.3 ExtendedMediaQuality datatype semantics

Semantics of the `ExtendedMediaQualityType`:

Name	*Definition*
`ExtendedMediaQualityType`	Describes results for quality analysis items for the segment.
`QCProfile`	The profile describing what to analyze and providing the inputs for this QC process. It is recommended to include the used profile instance at least once in a document, preferably at the `ExtendedMediaQualityType` descriptor closest to the root element. At least one `QCProfile` or `QCItemResult` element shall be present.
`QCItemResult`	Result description for a single quality analysis item (e.g. Video Noise).
`Operator`	Identifies the operator who created or validated the quality description (optional).
`UsedTool`	Describes the tool that has been used for creating the quality

Name	*Definition*
	description (optional).
Annotation	Textual annotation about the quality analysis.

8.2.9.2 QC Profile Tools

8.2.9.2.1 Introduction

The quality control profile type specifies the set of QC items (e.g. Audio Noise and Audio Clicks) and their input parameters.

8.2.9.2.2 QC Profile description tools syntax

```
<!-- ############################################################ -->
<!-- Definition of QCProfile DS                    -->
<!-- ############################################################ -->
<complexType name="QCProfileType">
   <complexContent>
    <extension base="mpeg7:DSType">
      <sequence>
        <element name="Name" type="mpeg7:ControlledTermUseType" />
        <element name="Description" type="mpeg7:TextualType" minOccurs="0" />
        <element name="QCItem" minOccurs="0" maxOccurs="unbounded">
         <complexType>
            <sequence>
             <element name="Name" type="mpeg7:ControlledTermUseType" minOccurs="1"
                 maxOccurs="unbounded"/>
             <element name="Layer" minOccurs="0">
               <simpleType>
                 <restriction base="string">
                  <enumeration value="wrapper"/>
                  <enumeration value="bitstream"/>
                  <enumeration value="baseband"/>
                 </restriction>
               </simpleType>
             </element>
             <element name="Category" minOccurs="0">
               <simpleType>
                 <restriction base="string">
                  <enumeration value="regulatory"/>
                  <enumeration value="absolute"/>
                  <enumeration value="objective"/>
                  <enumeration value="subjective"/>
                 </restriction>
               </simpleType>
             </element>
             <element name="Severity" type=" nonNegativeInteger" minOccurs="0" />
             <element name="EssenceType" minOccurs="0" maxOccurs="unbounded">
               <simpleType>
                 <restriction base="string">
                  <enumeration value="video"/>
                  <enumeration value="audio"/>
                  <enumeration value="subtitle"/>
                  <enumeration value="metadata"/>
                 </restriction>
               </simpleType>
             </element>
             <element name="InputParameter" minOccurs="0" maxOccurs="unbounded"
```

```
                type="mpeg7:QCValueType" />
            <element name="ItemScope" type=" mpeg7:SpatioTemporalLocatorType" minOccurs="0"
                    maxOccurs="unbounded"/>
            </sequence>
          </complexType>
         </element>
         <element name="CheckResultRule" minOccurs="0">
            <simpleType>
             <restriction base="string">
               <enumeration value="AND"/>
               <enumeration value="OR"/>
               <enumeration value="MinimumRelevance"/>
             </restriction>
            </simpleType>
         </element>
         <element name="ProfileScope" type="mpeg7:SpatioTemporalLocatorType" minOccurs="0"
            maxOccurs="unbounded"/>

         <element name="RelevanceLevel" minOccurs="0">
          <simpleType>
            <restriction base="integer">
             <minInclusive value="0"/>
             <maxInclusive value="10"/>
            </restriction>
          </simpleType>
         </element>
       </sequence>
     </extension>
   </complexContent>
</complexType>

<!-- Definition of QCValue -->
<complexType name="QCValueType">
   <simpleContent>
    <extension base="string">
      <attribute name="name" type="string" use="required" />
      <attribute name="unit" type="string" use="optional" />
      <attribute name="track" use="optional">
        <simpleType>
         <union memberTypes="string nonNegativeInteger" />
        </simpleType>
      </attribute>
      <attribute name="valueRange" type="string" use="optional" />
      <attribute name="type" type="string" use="optional" />
      <attribute name="representation" type="string" use="optional" />
    </extension>
   </simpleContent>
</complexType>

<!-- Definition of SegmentQCValue -->
<complexType name="SegmentQCValueType">
   <simpleContent>
    <extension base="mpeg7:QCValueType">
      <attribute name="start" type="mpeg7:mediaTimePointType" />
      <attribute name="duration" type="mpeg7:mediaDurationType" />
    </extension>
   </simpleContent>
</complexType>
```