Appendix A: PBCore 2.1 XML Schema and Process History

<?xml version="1.0" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
         xmlns="http://www.pbcore.org/PBCore/PBCoreNamespace.html"
         targetNamespace="http://www.pbcore.org/PBCore/PBCoreNamespace.html"
         elementFormDefault="qualified" version="2.1draft3">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">This is the PBCore version 2.1draft3 XML schema. All element descriptions can be found at http://www.pbcore.org</xsd:documentation>
  </xsd:annotation>
  <!-- Change Log: 20150717 -->
  - Added the 'source, ref, version, annotation' collection of attributes to all elements where they are not yet currently available.
  - Added supplemental attribute groups 'titleTypeSource, titleTypeRef, titleTypeVersion, titleTypeAnnotation'; 'subjectTypeSource, subjectTypeRef, subjectTypeVersion, subjectTypeAnnotation'; 'descriptionTypeSource, descriptionTypeRef, descriptionTypeVersion, descriptionTypeAnnotation'; 'segmentTypeSource, segmentTypeRef, segmentTypeVersion, segmentTypeAnnotation'; 'affiliationSource, affiliationRef, affiliationVersion, affiliationAnnotation'; and 'partTypeSource, partTypeRef, partTypeVersion, partTypeAnnotation' to allow for the sourcing of information in the 'titleType,' 'subjectType,' 'descriptionType,' 'segmentType,' 'affiliation' and 'partType' attributes.
  - Updated descriptions for all elements and attributes.
  -->
  <!-- the pbcoreCollection root element -->
  <xsd:element name="pbcoreCollection" type="pbcoreCollectionType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">Definition: The pbcoreCollection element groups multiple pbcoreDescriptionDocument XML into one container element to allow for a serialized output. Uses might include API returns or other web service output.</xsd:documentation>
      <xsd:documentation xml:lang="en">Best practice: This element is not intended to be equivalent to the archive/library concept of a 'collection.' Please see pbcoreAssetType for information on how PBCore can be used to express information about collections. The element is only applicable to XML expressions of PBCore. This container enables a similar function to RSS; pbcoreCollection would be similar to rss:channel and pbcoreDescription document to rss:item.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <!-- the pbcoreDescriptionDocument root element -->
  <xsd:element name="pbcoreDescriptionDocument" type="pbcoreDescriptionDocumentType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">Definition: the pbcoreDescriptionDocument element is a root XML element for the expression of an individual PBCore record. pbcoreDescriptionDocument can be used to express intellectual content only (e.g. a series or collection level record with no associated instantiations), or intellectual content with one or more instantiations (e.g. an episode of a program with copies/instantiations on
videotape and digital file). This element is only applicable to XML
expressions of PBCore.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<!-- the pbcoreInstantiationDocument root element -->
<xsd:element name="pbcoreInstantiationDocument" type="instantiationType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcoreInstantiation element is the equivalent of the instantiation element, but used for the expression of an instantiation record at the root of an XML document. This is most commonly used when referenced from other schemas, or if you want to create and express a single, stand-alone instantiation.</xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:complexType>
</xsd:sequence>
</xsd:complexType>
</xsd:annotation>
</xsd:element>
<!-- the pbcoreCollectionType -->
<xsd:complexType name="pbcoreCollectionType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcoreCollectionType schema type allows the addition of attributes that describe the PBCoreCollection. The attributes define the title, the description, the source, the reference and the date of the collection.</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="1" ref="pbcoreDescriptionDocument">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The pbcoreDescriptionDocument element assembles together all of PBCore knowledge items into a single data record organized in a hierarchical structure. For PBCore these knowledge items are metadata descriptions of media, including all the knowledge items and metadata terms and values associated with its content and containers.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="collectionTitle" type="xsd:string">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">Definition: The collectionTitle attribute is a title or label for the group of individual serialized XML records contained within one pbcoreCollection element.</xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="collectionDescription" type="xsd:string">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">Definition: The collectionDescription attribute is a description group of individual serialized XML records contained within one pbcoreCollection element.</xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>
</xsd:annotation>
<xsd:attribute name="collectionSource" type="xsd:string">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The collectionSource attribute indicates an organization, application, or individual for group of individual XML records contained within a pbcoreCollection element.</xsd:documentation>
  </xsd:annotation>
</xsd:attribute>

<xsd:attribute name="collectionRef" type="xsd:string">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The collectionRef attribute provides a URL for the source organization, application, or individual for a group of XML records contained within a pbcoreCollection element.</xsd:documentation>
  </xsd:annotation>
</xsd:attribute>

<xsd:attribute name="collectionDate" type="xsd:string">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The collectionDate attribute provides the date of creation for a pbcoreCollection XML document.</xsd:documentation>
  </xsd:annotation>
</xsd:attribute>

<xsd:attributeGroup ref="sourceVersionGroup"/>
</xsd:complexType>
Definition: The pbcoreAssetDate element is intended to reflect dates associated with the Intellectual Content.

Best practice: By contrast, instantiationDate is intended to reflect date information for the specific instance. For example, if you have a VHS copy of Gone With The Wind, the pbcoreAssetDate would be 1939, while the instantiationDate of the VHS copy could be 1985. pbcoreAssetDate may also be used to reflect availability dates, etc. Date types should be specified using the @dateType attribute. Dates or time-based events related to the content of the asset, on the other hand, would be described in the 'coverage' element -- so, while the storyline of Gone with the Wind takes place in the nineteenth century, this information should be noted in the Coverage field, not the assetDate field. Best practice is to use ISO 8601 or some other date/time standard if possible.

Definition: The pbcoreIdentifier element provides an identifier that can apply to the asset. This identifier should not be limited to a specific instantiation, but rather is shared by or common to all instantiations of an asset. It can also hold a URL or URI that points to the asset.

Best practice: Identify the asset by means of a string or number corresponding to an established or formal identification system if one exists. Otherwise, use an identification method that is in use within your agency, station, production company, office, or institution.

Definition: The pbcoreTitle element is a name or label relevant to the asset.

Best practice: An asset may have many types of titles, an asset may have, such as a series title, episode title, segment title, or project title; therefore the element is repeatable.

Definition: The pbcoreSubject element is intended to reflect dates associated with the Intellectual Content.
<xsd:element maxOccurs="unbounded" minOccurs="0"
name="pbcoreSubject"
  type="subjectStringType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcoreSubject element is used to assign topic headings or keywords that portray the intellectual content of the asset. A subject is expressed by keywords, key phrases, or even specific classification codes. Controlled vocabularies, authorities, formal classification codes, as well as folksonomies and user-generated tags, may be employed when assigning descriptive subject terms.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

<!-- the pbcore description - this element may occur as many times as desired, however if it does occur, then a description tag is required. optionally, the description type may appear - but it has a limited vocabulary -->
<xsd:element maxOccurs="unbounded" minOccurs="1"
name="pbcoreDescription"
  type="descriptionStringType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcoreDescription element uses free-form text or a narrative to report general notes, abstracts, or summaries about the intellectual content of an asset. The information may be in the form of an individual program description, anecdotal interpretations, or brief content reviews. The description may also consist of outlines, lists, bullet points, rundowns, edit decision lists, indexes, or tables of content.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

<!-- the pbcore genre - this element may occur as many times as desired. -->
<xsd:element maxOccurs="unbounded" minOccurs="0"
name="pbcoreGenre"
  type="sourceVersionStartEndStringType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcoreGenre element describes the Genre of the asset, which can be defined as a categorical description informed by the topical nature or a particular style or form of the content.</xsd:documentation>
    <xsd:documentation xml:lang="en">Best practice: Genre refers to the intellectual content of the asset, whereas the element pbcoreAssetType defines a broader structural category; i.e. an asset might have the Asset Type of Segment, with a Genre of News, together defining a news segment.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

<!-- the pbcore relation - this element may occur as many times as desired. -->
<xsd:element maxOccurs="unbounded" minOccurs="0"
name="pbcoreRelation">
  <xsd:annotation>
Definition: The pbcoreRelation element contains the pbcoreRelationType and pbcoreRelationIdentifier elements. In order to properly use these two elements they must be nested with the pbcoreRelation element, and pbcoreRelation must contain both pbcoreRelationType and pbcoreRelationIdentifier if it is included.

Definition: The pbcoreRelationType element describes the relationship between the asset being describe by the pbcore document and any other asset. Ideally it would contain text from a controlled vocabulary for describing relationships. There is some depth to what a relationship could be. The assets can be related as different episodes in a series, different tapes in a box set, or different versions of an original, among others.

Best practice: The assets may be related in that they are different discrete parts of a single intellectual unit, one may be a derivative of another, or they may be different versions that are distinct enough to be described as separate assets.

Definition: The pbcoreRelationIdentifier element contains the identifier of the related asset. In the case that the related asset has a PBCore record, this identifier should correspond with the pbcoreIdentifier of the related asset. However, it is possible to use this element with a record that isn’t in PBCore, in which case the source attribute should identify the source of the identifier.

 coverageType -->

Definition: The pbcoreCoverage element is a container for sub-elements 'coverage' and 'coverageType'.
The coverage element refers to either the geographic location or the time period covered by the asset's intellectual content. For geographic locations ('spatial' descriptors), it is expressed by keywords such as place names (e.g. 'Alaska' or 'Washington, DC'), numeric coordinates or geo-spatial data. For time-based events ('temporal' descriptors), it is expressed by using a date, period, era, or time-based event that is portrayed or covered in the intellectual content (e.g. '2007' or 'Victorian Era'). The PBCore metadata element coverage houses the actual spatial or temporal keywords. The companion element coverageType is used to identify the type of keywords that are being used.

The coverageType element is used to identify the actual type of keywords that are being used by its companion metadata element coverage. coverageType provides a picklist of two possible types - spatial or temporal - because coverage in intellectual content may be expressed spatially by geographic location or it may also be expressed temporally by a date, period, era, or time-based event.

The pbcoreAudienceLevel element identifies a type of audience, viewer, or listener for whom the media item is primarily designed or educationally useful.

The pbcoreAudienceRating element identifies a type of audience, viewer, or listener for whom the media item is primarily designed or educationally useful.

Definition: The pbcoreAudienceLevel element identifies a type of audience, viewer, or listener for whom the media item is primarily designed or educationally useful.

Definition: The pbcoreAudienceRating element identifies a type of audience, viewer, or listener for whom the media item is primarily designed or educationally useful.
The pbcoreAudienceRating element designates the type of users for whom the intellectual content of a media item is intended or judged appropriate. This element differs from the element pbcoreAudienceLevel in that it utilizes standard ratings that have been crafted by the broadcast television and film industries and that are used as flags for audience or age-appropriate materials.

---

The pbcoreCreator element is a container for sub-elements 'creator' and 'creatorRole'.

---

The creator element identifies the primary person, people, or organization(s) responsible for creating the asset. Note that non-primary names and roles should be included within the pbcoreContributor container. Best practice: We recommend providing a consistent internal standard for entering proper names and organizational names, such as 'Last name, First name, Middle name,' or 'Main group, subdivision.' We also recommend supplying separate pbcoreCreator containers for each creator to be named for a resource.

---

The creatorRole element is used to identify the role played by the person, people or organization(s) identified in the companion descriptor creator. The PBCore schema allows for creatorRole to be repeated in the pbcoreCreator container element. This can be useful when a single person or organization is associated with multiple roles in an asset.
The contributor element identifies a person, people, or organization that has made substantial creative contributions to the asset. This contribution is considered to be secondary to the primary author(s) (person or organization) identified in the descriptor creator. Best practice: We recommend providing a consistent internal standard for entering proper names and organizational names, such as 'Last name, First name, Middle name,' or 'Main group, subdivision.' We also recommend supplying separate pbcoreCreator containers for each creator to be named for a resource.

The pbcorePublisher element - this follows the same guidelines as the contributor element. This may exist as many times as we wish, but inside it there must be a publisher tag. A publisherRole tag is optional.

The pbcorePublisher element is a container for sub-elements 'publisher' and 'publisherRole.'
The publisher element identifies a person, people, or organization primarily responsible for distributing or making the asset available to others. The publisher may be a person, a business, organization, group, project or service. Best practice: We recommend providing a consistent internal standard for entering proper names and organizational names, such as 'Last name, First name, Middle name,' or 'Main group, subdivision.' We also recommend supplying separate pbcoreCreator containers for each creator to be named for a resource.

The publisherRole element is used to identify the role played by the specific publisher or publishing entity identified in the companion descriptor publisher. The PBCore schema allows for publisherRole to be repeated in the pbcorePublisher container element. This can be useful when a single person or organization is associated with multiple roles in an asset.

The pbcoreRightsSummary element is a container for sub-elements 'rightsSummary', 'rightsLink', and 'rightsEmbedded' used to describe Rights for the asset.

The instantiationType element contains sub-elements that describe a single instantiation of an asset. The definition is malleable but it should be thought of as any discreet and tangible unit that typically (though not always) comprises a whole representation of the asset. For example, an original master videotape, a preservation master video file, and a low-bitrate access copy would all be considered Instantiations of a single video program. All of the sub-elements held by this element are used to describe the instantiation specifically, not necessarily the asset as a whole.
<xsd:element maxOccurs="unbounded" minOccurs="0" name="pbcoreAnnotation" type="annotationStringType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcoreAnnotation element allows the addition of any supplementary information about the metadata used to describe the PBCore record. pbcoreAnnotation clarifies element values, terms, descriptors, and vocabularies that may not be otherwise sufficiently understood.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

<xsd:element maxOccurs="unbounded" minOccurs="0" name="pbcorePart" type="pbcorePartType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcorePart element may be used to split up a single asset so as to enable the use of all available elements at the pbcoreDescriptionDocument level to describe the intellectual content of individual segments of an asset. Best practice: Splitting up an asset in this way allows for defining and describing segments, stories, episodes or other divisions within the asset, such as individual films in a compilation reel, or distinct segments of a news show when each may have their own titles, creators, publishers, or other specific intellectual content information that does not apply across the whole asset.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

<xsd:element maxOccurs="unbounded" minOccurs="0" name="pbcoreExtension" type="extensionType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The pbcoreExtension element can be used as either a wrapper containing a specific element from another standard OR embedded xml containing the extension. Best practice: Use it to supplement other metadata sub-elements of the PBCore description document in which it appears.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

</xsd:complexType>
</xsd:attributeGroup ref="sourceVersionGroup"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:complexType>
single instantiation or multiple instantiations within a pbcoreDocumentDescription.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<!-- the pbcore instantiationIdentifier -->
<xsd:element maxOccurs="unbounded" minOccurs="1"
name="instantiationIdentifier"
type="requiredSourceVersionStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The instantiationIdentifier element contains an unambiguous reference or identifier for a particular instantiation of an asset.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<!-- the pbcore instantiationDate -->
<xsd:element maxOccurs="unbounded" minOccurs="0"
name="instantiationDate"
type="dateStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The instantiationDate element is a date associated with an instantiation.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<!-- the pbcore instantiationDimensions -->
<xsd:element maxOccurs="unbounded" minOccurs="0"
name="instantiationDimensions"
type="technicalStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The instantiationDimensions element specifies either the dimensions of a physical instantiation, or the high-level visual dimensions of a digital instantiation.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<!-- the pbcore instantiationPhysical -->
<xsd:element maxOccurs="1" minOccurs="0"
name="instantiationPhysical"
type="sourceVersionStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The instantiationPhysical element is used to identify the format of a particular instantiation as it exists in a physical form that occupies physical space.
(e.g., a tape on a shelf). This includes physical digital media, such as a DV tape, audio CD or authored DVD, as well as analog media.

Best practice: PBCore provides a controlled vocabulary for media objects, though any controlled vocabulary can be used as long as it is referenced. For digital storage carriers that contain portable file-based media, such as data CDs, LTO tapes or hard drives, use instantiationDigital to convey the MIME type of the file instead of describing the carrier.

Definition: The instantiationDigital element is used to identify the format of a particular instantiation of an asset as it exists as a digital file on a server, hard drive, or other digital storage medium. Digital instantiations should be expressed as a formal Internet MIME types.

Best practice: instantiationDigital should only be used to describe the MIME type of the digital file itself. There are multiple options to convey more information about the storage medium or location of the digital file, which are discussed in more detail on the PBCore site.

Definition: The instantiationStandard element can be used, if the instantiation is a physical item, to refer to the broadcast standard of the video signal (e.g. NTSC, PAL), or the audio encoding (e.g. Dolby A, vertical cut). If the instantiation is a digital item, instantiationStandard should be used to express the container format of the digital file (e.g. MXF).

Best practice: While the usage described in the definition is best practice for 2.1, this usage is likely to change if new elements are added for PBCore 3.0.

Definition: The instantiationLocation element may contain information about a specific location for an instantiation, such as an organization's name, departmental name, shelf ID and contact information. The instantiationLocation for a digital file should include domain, path or URI to the file.
Best practice: For digital files, instantiationLocation should always include a path or URI to the file. There are multiple ways to convey additional information about the location of a carrier or storage medium of the digital file, which are expressed on the PBCore site.

Definition: The instantiationMediaType element identifies the general, high level nature of the content of an instantiation. It uses categories that show how content is presented to an observer, e.g., as a sound, text or moving image.

Definition: The instantiationGenerations element identifies the use type and provenance of the instantiation. The generation of a video tape may be an "Original Master" or "Dub", the generation of a film reel may be an "Original Negative" or "Composite Positive", an audiotape may be a "Master" or "Mix Element", an image may be a "Photograph" or a "Photocopy.

Definition: The instantiationFileSize element indicates the file size of a digital instantiation. It should contain only numerical values. As a standard, express the file size in bytes. Units of Measure should be declared in the unitsOfMeasure attribute.

Definition: The instantiationTimeStart element describes the point at which playback begins for a time-based instantiation. It is likely that the content on a tape may begin an arbitrary amount of time after the beginning of the instantiation. Best practice is to use a timestamp format such as HH:MM:SS[[:;]FF or HH:MM:SS.mmm or S.mmm.
The instantiationDuration element provides a timestamp for the overall length or duration of a time-based media item. It represents the playback time. Best practice is to use a timestamp format such as HH:MM:SS[;|;]FF or HH:MM:SS.mmm or S.mmm.

The instantiationDataRate element expresses the amount of data in a digital media file that is encoded, delivered or distributed, for every second of time. This should be expressed as numerical data, with the units of measure declared in the unitsOfMeasure attribute. For example, if the audio file is 56 kilobits/second, then 56 should be the value of instantiationDataRate and the attribute unitsOfMeasure should be kilobits/second.

The instantiationColors element indicates the overall color, grayscale, or black and white nature of the presentation of an instantiation, as a single occurrence or combination of occurrences in or throughout the instantiation.

The instantiationTracks element is simply intended to indicate the number and type of tracks that are found in a media item, whether it is analog or digital. (e.g. 1 video track, 2 audio tracks, 1 text track, 1 sprite track, etc.) Other configuration information specific to these identified tracks should be described using instantiationChannelConfiguration.

Best practice: Best practices is to use essenceTracks, as this element has been deprecated.
<xsd:element maxOccurs="1" minOccurs="0"
name="instantiationChannelConfiguration"
  type="sourceVersionStringType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The instantiationChannelConfiguration element is designed to indicate, at a general narrative level, the arrangement or configuration of specific channels or layers of information within an instantiation's tracks. Examples are 2-track mono, 8-track stereo, or video track with alpha channel.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

<!-- the pbcore instantiationLanguage -->
<xsd:element name="instantiationLanguage"
  type="threeLetterStringType"
  maxOccurs="unbounded" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The instantiationLanguage element identifies the primary language of the tracks' audio or text. Languages must be indicated using 3-letter codes standardized in ISO 639-2 or 639-3. If an instantiation includes more than one language, the element can be repeated. Alternately, both languages can be expressed in one element by separating two three-letter codes with a semicolon, i.e. <instantiationLanguage>eng;fre</instantiationLanguage>. + Best practice: Alternative audio or text tracks and their associated languages should be identified using the element instantiationAlternativeModes.</xsd:documentation>
  </xsd:annotation>
</xsd:element>

<!-- the pbcore instantiationAlternativeModes -->
<xsd:element maxOccurs="0"
name="instantiationAlternativeModes"
  type="sourceVersionStringType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The instantiationAlternativeModes element is a catch-all metadata element that identifies equivalent alternatives to the primary visual, sound or textual information that exists in an instantiation. These are modes that offer alternative ways to see, hear, and read the content of an instantiation. Examples include DVI (Descriptive Video Information), SAP (Supplementary Audio Program), ClosedCaptions, OpenCaptions, Subtitles, Language Dubs, and Transcripts. For each instance of available alternativeModes, the mode and its associated language should be identified together, if applicable. Examples include 'SAP in English,' 'SAP in Spanish,' 'Subtitle in French,' 'OpenCaption in Arabic.' </xsd:documentation>
  </xsd:annotation>
</xsd:element>

<!-- the pbcore instantiationEssenceTrack -->
<xsd:element maxOccurs="unbounded" minOccurs="0"
name="instantiationEssenceTrack"
  type="essenceTrackType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The instantiationEssenceTrack element is an XML container element that allows for grouping of related essenceTrack elements and their repeated use. Use instantiationEssenceTrack element to describe the individual streams that
comprise an instantiation, such as audio, video, timecode, etc.

Best practice: Essence tracks can exist in either the digital or physical realm. In the digital realm, they may refer to the separate audio and video tracks within a digital file. In the physical realm, they may refer to the video and audio tracks contained on a single video tape.

Definition: The instantiationRelation element is a container for sub-elements instantiationRelationType and instantiationRelationIdentifier to describe relationships to other instantiations.

Best practice: Use to express relationships between instantiations, for example to note that they are different discrete parts of a single intellectual unit, generationally related, derivative of another, or different versions.

Definition: The instantiationRelationIdentifier element is used to provide a name, locator, accession, identification number or ID where the related item can be obtained or found.

Best practice: We recommend using a unique identifier or global unique ID in this element.
is a multi-part instantiation and time notation is important.
</xsd:documentation>
</xsd:annotation>
</xsd:attributeGroup>
<xsd:attributeGroup ref="sourceVersionGroup">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The instantiation level attribute group sourceVersionGroup may be used when there is a multi-part instantiation and notation is important. </xsd:documentation>
</xsd:annotation>
</xsd:attributeGroup>
</xsd:complexType>
<!-- the pbcore instantiation essenceTrackType -->
<xsd:complexType name="essenceTrackType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The essenceTrackType schema type uses a common structure to allow for grouping of the essence related elements and their repeated use. </xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<!-- the pbcore instantiation essenceTrackType -->
<xsd:element maxOccurs="1" minOccurs="0" name="essenceTrackType" type="sourceVersionStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The essenceTrackType element refers to the media type of the decoded data. Tracks may possibly be of these types: video, audio, caption, metadata, image, etc. </xsd:documentation>
</xsd:annotation>
</xsd:element>
<!-- the pbcore instantiation essenceTrackIdentifier -->
<xsd:element maxOccurs="unbounded" minOccurs="0" name="essenceTrackIdentifier" type="sourceVersionStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The essenceTrackIdentifier element is an identifier of the track. Several audiovisual containers include such identifier schema to identify each track, such as MPEG2 PIDs or QuickTime Track IDs. </xsd:documentation>
</xsd:annotation>
</xsd:element>
<!-- the pbcore instantiation essenceTrackStandard -->
<xsd:element maxOccurs="1" minOccurs="0" name="essenceTrackStandard" type="sourceVersionStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The essenceTrackStandard element should be be used with file-based instantiations to describe the broadcast standard of the video signal (e.g. NTSC, PAL) or to further clarify the standard of the essenceTrackEncoding format. </xsd:documentation>
</xsd:annotation>
</xsd:element>
<!-- the pbcore instantiation essenceTrackEncoding -->
<xsd:element maxOccurs="1" minOccurs="0" name="essenceTrackEncoding" type="sourceVersionStringType"/>
Definition: The essenceTrackEncoding element identifies how the actual information in an instantiation is compressed, interpreted, or formulated using a particular scheme. Identifying the encoding used is beneficial for a number of reasons, including as a way to achieve reversible compression; for the construction of document indices to facilitate searching and access; or for efficient distribution of the information across data networks with differing bandwidths or pipeline capacities. Human-readable encoding value should be placed here. Use @ref to identify the codec ID.

Best practice: Use @source to describe the type of encoding reference used, such as fourcc. In @ref, use a URI/URL from the source to identify the codec utilized by its container format.

Definition: The essenceTrackDataRate element measures the amount of data used per time interval for encoded data. The data rate can be calculated by dividing the total data size of the track's encoded data by a time unit. By default use bytes per second.

Best practice: Example: 1920x1080.

Definition: The essenceTrackFrameRate element is relevant to tracks of video track type only. The frame rate is calculated by dividing the total number of frames by the duration of the video track. By default measure frame rate in frames per second expressed as fps as a unit of measure. e.g., 24 fps.

Best practice: Example: 1280x720.

Definition: The essenceTrackPlaybackSpeed element specifies the rate of units against time at which the media track should be rendered for human consumption. e.g., 15ips (inches per second).
**essenceTrackSamplingRate** element measures how often data is sampled when information from the audio portion from an instantiation is digitized. For a digital audio signal, the sampling rate is measured in kilohertz and is an indicator of the perceived playback quality of the media item (the higher the sampling rate, the greater the fidelity).

**essenceTrackBitDepth** element specifies how much data is sampled when information is digitized, encoded, or converted for an instantiation (specifically, audio, video, or image). Bit depth is measured in bits and generally implies an arbitrary perception of quality during playback of an instantiation (the higher the bit depth, the greater the fidelity).

**essenceTrackFrameSize** element measures the width and height of the encoded video or image track. The frame size refers to the size of the encoded pixels and not the size of the displayed image. It may be expressed as a combination of pixels measured horizontally vs. the number of pixels of image/resolution data stacked vertically (interlaced and progressive scan).

**essenceTrackAspectRatio** element indicates the ratio of horizontal to vertical proportions in the display of a static image or moving image.

**essenceTrackTimeStart** element provides a time stamp for the beginning point of playback for a time-based essence track. It is likely that the content on
A tape may begin an arbitrary amount of time after the beginning of the instantiation.

Best practice: Use in combination with essenceTrackDuration to identify a sequence or segment of an essence track that has a fixed start time and end time. Best practice is to use a timestamp format such as HH:MM:SS[:;|:]FF or HH:MM:SS.mmm or S.mmm.

Definition: The essenceTrackDuration element provides a timestamp for the overall length or duration of a track. It represents the track playback time. Best practice is to use a timestamp format such as HH:MM:SS[:;|:]FF or HH:MM:SS.mmm or S.mmm.

Definition: The essenceTrackLanguage element identifies the primary language of the tracks' audio or text.

Best practice: Alternative audio or text tracks and their associated languages should be identified using the element alternativeModes.

Definition: The essenceTrackAnnotation element can store any supplementary information about a track or the metadata used to describe it. It clarifies element values, terms, descriptors, and vocabularies that may not be otherwise sufficiently understood.

Definition: The essenceTrackExtension element can be used as either a wrapper containing a specific element from another standard or embedded XML containing the extension. The essenceTrackExtension element is a container to accommodate track-level metadata from external systems. Use it to supplement other
metadata sub-elements of instantiationEssenceTrack in which it appears.</xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
<xsd:attributeGroup ref="sourceVersionGroup"/>
</xsd:complexType>
<!-- extensionType -->
<xsd:complexType name="extensionType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">Definition: The extensionType
schema type uses a common structure to allow for the use of multiple,
qualified extensions at the asset, instantiation and essence
levels.</xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:element maxOccurs="unbounded" minOccurs="1"
      name="extensionWrap">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The
extensionWrap element serves as a container for the elements
extensionElement, extensionValue, and
extensionAuthorityUsed.</xsd:documentation>
      </xsd:annotation>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element maxOccurs="1" minOccurs="1"
            name="extensionElement"
            type="xsd:string">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">Definition: The
extensionElement element should contain the name of an element used from
another metadata standard, in the case that an element from another metadata
standard is used. While we recommend the usage of an existing standard, this
element can also be used to define local elements that may not be part of an
existing standard. "</xsd:documentation>
            </xsd:annotation>
          </xsd:element>
          <xsd:element maxOccurs="1" minOccurs="1"
            name="extensionValue">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">Definition: The
extensionValue element is used to express the data value of the label
indicated by extensionElement.</xsd:documentation>
            </xsd:annotation>
          </xsd:element>
          <xsd:element maxOccurs="0"
            name="extensionAuthorityUsed">
            <xsd:annotation>
            </xsd:annotation>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:choice>
</xsd:complexType>
The extensionAuthorityUsed element identifies the authority used for the extensionElement.

Best practice: If metadata extensions to PBCore are assigned to a media item with the element extensionElement, and the terms used are derived from a specific authority or metadata scheme, use extensionAuthorityUsed to identify whose metadata extensions are being used.

-- pbcorePartType --

Definition: The pbcorePartType schema type uses a common structure to allow for the repeating of descriptive sub-documents to define different segments, episodes etc., just as super-element 'pbcoreDescriptionDocument' can be collected and used to describe higher-level media programs.
<xsd:documentation>Best practice: This might be the name of a controlled vocabulary, namespace or authority list, such as the official PBCore vocabulary. We recommend a consistent and human readable use.</xsd:documentation>

<xsd:attribute name="partTypeRef" type="xsd:string">
    <xsd:documentation>Definition: The partTypeRef attribute is used to supply a source's URI for the value of the attribute titleTypeSource.</xsd:documentation>
    <xsd:documentation>Best practice: The partTypeRef attribute can be used to point to a term in a controlled vocabulary, or a URI associated with a source.</xsd:documentation>
</xsd:attribute>

<xsd:attribute name="titleTypeVersion" type="xsd:string">
    <xsd:documentation>Definition: The partTypeVersion attribute identifies any version information about the authority or convention used to express data of this element.</xsd:documentation>
</xsd:attribute>

<xsd:attribute name="titleTypeAnnotation" type="xsd:string">
    <xsd:documentation>Definition: The partTypeAnnotation attribute includes narrative information intended to clarify the nature of data used in the element.</xsd:documentation>
    <xsd:documentation>Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.</xsd:documentation>
</xsd:attribute>

<!dateStringType-->
<xsd:complexType name="dateStringType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The dateStringType schema type allows for the addition of the dateTime attribute.</xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="dateTime" type="xsd:string">
                <xsd:documentation xml:lang="en">Definition: The dateTime attribute classifies by named type the date-related data of the element e.g., created, broadcast, dateAvailableStart.</xsd:documentation>
                <xsd:documentation xml:lang="en">Best practice: Used to clarify how the date is related to the asset or instantiation. Date Created may be the most common, but the element could also be used to describe the Date Accessioned or Date Deaccessioned, for example.</xsd:documentation>
            </xsd:attribute>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>
<xsd:attributeGroup ref="sourceVersionGroup"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
<!-- sourceVersionStringType -->
<xsd:complexType name="sourceVersionStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The sourceVersionStringType schema type is used with a number of elements to allow the attachment of the attributes: source, ref, version and annotation.</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attributeGroup ref="sourceVersionGroup"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
<!-- requiredSourceVersionStringType -->
<xsd:complexType name="requiredSourceVersionStringType">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The requiredSourceVersionStringType schema type type is the same as sourceVersionStringType with the addition that the source attribute is required instead of optional.</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="source" type="xsd:string" use="required">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The source attribute provides the name of the authority used to declare the value of the element.</xsd:documentation>
<xsd:documentation xml:lang="en">Best practice: Different elements will use the source attribute slightly differently. For example, identifier source (required) should be the name of the organization, institution, system or namespace that the identifier came from, such as "PBS NOLA Code" or an institutional database identifier. For other elements, this might be the name of a controlled vocabulary, namespace or authority list, such as Library of Congress Subject Headings. We recommend a consistent and human readable use.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="ref" type="xsd:string">
<xsd:annotation>
<xsd:documentation xml:lang="en">Definition: The ref attribute is used to supply a source's URI for the value of the element.</xsd:documentation>
<xsd:documentation xml:lang="en">Best practice: Attribute ref can be used to point to a term in a controlled vocabulary, or a URI associated with a source.</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="version" type="xsd:string">
<xsd:annotation>
</xsd:attribute>
Definition: The version attribute identifies any version information about the authority or convention used to express data of this element.

Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.

Definition: The annotation attribute includes narrative information intended to clarify the nature of data used in the element.

Definition: The titleStringType schema type allows for the addition of a titleType attribute as well as the standard sourceVersionGroup attributes and a startEndTimeGroup or attributes.

Definition: The titleType attribute is used to indicate the type of title being assigned to the asset, such as series title, episode title or project title.

Definition: The titleTypeSource attribute is used to provide the name of the authority used to declare data value of the titleType attribute.

Best practice: This might be the name of a controlled vocabulary, namespace or authority list, such as the official PBCore vocabulary. We recommend a consistent and human readable use.

Definition: The titleTypeRef attribute is used to supply a source's URI for the value of the attribute titleTypeSource.

Best practice: Attribute titleTypeRef can be used to point to a term in a controlled vocabulary, or a URI associated with a source.
<xsd:attribute name="titleTypeVersion" type="xsd:string">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The titleTypeVersion attribute identifies any version information about the authority or convention used to express data of this element.</xsd:documentation>
    </xsd:annotation>
</xsd:attribute>

<xsd:attribute name="titleTypeAnnotation" type="xsd:string">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The titleTypeAnnotation attribute includes narrative information intended to clarify the nature of data used in the element.</xsd:documentation>
        <xsd:documentation xml:lang="en">Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.</xsd:documentation>
    </xsd:annotation>
</xsd:attribute>

<xsd:attributeGroup ref="sourceVersionGroup"/>
<xsd:attributeGroup ref="startEndTimeGroup"/>

<!-- subjectStringType -->
<xsd:complexType name="subjectStringType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The subjectStringType schema type allows for the addition of a subjectType attribute as well as the standard sourceVersionGroup attributes and a startEndTimeGroup or attributes.</xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="subjectType" type="xsd:string">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">Definition: The subjectType attribute is used to indicate the type of subject being assigned to the attribute subjectType, such as 'topic,' 'personal name,' or 'keyword'.</xsd:documentation>
                </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute name="subjectTypeSource" type="xsd:string">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">Definition: The subjectTypeSource attribute provides the name of the authority used to declare the value of the attribute subjectType.</xsd:documentation>
                    <xsd:documentation xml:lang="en">Best practice: This might be the name of a controlled vocabulary, namespace or authority list, such as the official PBCore vocabulary. We recommend a consistent and human readable use.</xsd:documentation>
                </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute name="subjectTypeRef" type="xsd:string">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">Definition: The subjectTypeRef attribute is used to supply a source's URI for the value of the attribute subjectType.</xsd:documentation>
                </xsd:annotation>
            </xsd:attribute>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>
Best practice: Attribute subjectTypeRef can be used to point to a term in a controlled vocabulary, or a URI associated with a source.

Definition: The subjectTypeVersion attribute identifies any version information about the authority or convention used to express data of the attribute subjectType.

Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.

Definition: The subjectTypeAnnotation attribute includes narrative information intended to clarify the nature of data used in the attribute subjectType.

Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.

Definition: The descriptionType attribute is used to indicate the type of description being assigned to the element, such as 'abstract,' 'summary,' or 'physical description.'
such as the official PBCore recommended vocabulary. We recommend a consistent and human readable use.

```xml
<xs:attribute name="descriptionTypeRef" type="xsd:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">Definition: The descriptionTypeRef attribute is used to supply a source's URI for the value of the attribute descriptionType.</xs:documentation>
    <xs:documentation xml:lang="en">Best practice: The descriptionTypeRef attribute can be used to point to a term in a controlled vocabulary, or a URI associated with a source.</xs:documentation>
  </xs:annotation>
</xs:attribute>

<xs:attribute name="descriptionTypeVersion" type="xsd:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">Definition: The descriptionTypeVersion attribute identifies any version information about the authority or convention used to express data of the attribute descriptionType.</xs:documentation>
    <xs:documentation xml:lang="en">Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.</xs:documentation>
  </xs:annotation>
</xs:attribute>

<xs:attribute name="descriptionTypeAnnotation" type="xsd:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">Definition: The descriptionTypeAnnotation attribute includes narrative information intended to clarify the nature of data used in the element.</xs:documentation>
    <xs:documentation xml:lang="en">Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.</xs:documentation>
  </xs:annotation>
</xs:attribute>

<xs:attribute name="segmentType" type="xsd:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">Definition: The segmentType attribute is used to define the type of content contained in a segment.</xs:documentation>
    <xs:documentation xml:lang="en">Best practice: We recommend using description and descriptionType instead of segmentType.</xs:documentation>
  </xs:annotation>
</xs:attribute>

<xs:attribute name="segmentTypeSource" type="xsd:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">Definition: The segmentTypeSource attribute provides the name of the authority used to declare data value of the attribute segmentType.</xs:documentation>
    <xs:documentation xml:lang="en">Best practice: This might be the name of a controlled vocabulary, namespace or authority list, such as the official PBCore recommended vocabulary.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```
Definition: The segmentTypeRef attribute is used to supply a source's URI for the value of the attribute segmentType.

Best practice: Attribute segmentTypeRef can be used to point to a term in a controlled vocabulary, or a URI associated with a source.

Definition: The segmentTypeVersion attribute identifies any version information about the authority or convention used to express data of the attribute segmentType.

Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.

Definition: The segmentTypeAnnotation attribute includes narrative information intended to clarify the nature of data used in the attribute segmentType.

Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.

SourceVersionStartEndStringType

Definition: The sourceVersionStartEndStringType adds attributes that define the source of the string with the option of time related attributes.

affiliatedStringType

Definition: The affiliatedStringType adds attributes of affiliation and time relevance.

affiliation type="xsd:string"
Definition: The affiliation attribute is used to indicate the organization with which an agent is associated or affiliated.

Definition: The affiliationSource attribute provides the name of the authority used to declare the value of the attribute affiliation.

Best practice: This might be the name of a controlled vocabulary, namespace or authority list, such as the official PBCore recommended vocabulary.

Definition: The affiliationRef attribute is used to supply a source's URI for the value of the attribute affiliation.

Best practice: Attribute affiliationRef can be used to point to a term in a controlled vocabulary, or a URI associated with a source.

Definition: The affiliationVersion attribute identifies any version information about the authority or convention used to express data of the attribute affiliation.

Definition: The affiliationAnnotation attribute includes narrative information intended to clarify the nature of data used in the attribute affiliation.

Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.

Definition: The contributorString helps define the portrayal role as well as the general source and version group attributes.
<xsd:attribute name="portrayal" type="xsd:string">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The portrayal attribute identifies any roles or characters performed by a contributor.</xsd:documentation>
    </xsd:annotation>
</xsd:attribute>
</xsd:attributeGroup ref="sourceVersionGroup"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
<!-- technicalStringType -->
<xsd:complexType name="technicalStringType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The technicalStringType schema type adds the attributes of unitsOfMeasure and annotation.</xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="unitsOfMeasure" type="xsd:string">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">Definition: The unitsOfMeasure attribute defines the unit used in the containing element, e.g. pixels, GB, Mb/s, ips, fps, kHz, inches, lines, dpi.</xsd:documentation>
                    <xsd:documentation xml:lang="en">Best practice: We recommend standardizing the notation that is most widely recognized in your institution and using with consistency.</xsd:documentation>
                </xsd:annotation>
            </xsd:attribute>
            <xsd:attributeGroup ref="sourceVersionGroup"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>
<!-- instantiationDigitalStringType -->
<xsd:complexType name="instantiationStandardStringType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">Definition: The instantiationStandardStringType schema type allows for the addition of a profile attribute along with the sourceVersionGroup.</xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="profile" type="xsd:string">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">Definition: The profile attribute is used to further quantify the profile of the container format (e.g. Op1a).</xsd:documentation>
                    <xsd:documentation xml:lang="en">Best practice: This attribute can be used as a notes field to include any additional information about the element or associated attributes.</xsd:documentation>
                </xsd:annotation>
            </xsd:attribute>
            <xsd:attributeGroup ref="sourceVersionGroup"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>
<xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
<!-- annotationStringType -->
<xsd:complexType name="annotationStringType">
<xsd:documentation xml:lang="en">Definition: The stringType schema type added an annotationType attribute and a reference.</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
   <xsd:attribute name="annotationType" type="xsd:string">
      <xsd:documentation xml:lang="en">Definition: Use the attribute annotationType to indicate the type of annotation being assigned to the asset, such as a comment, clarification, or cataloging note.</xsd:documentation>
   </xsd:attribute>
   <xsd:attributeGroup ref="sourceVersionGroup"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
<!-- rightsSummaryType -->
<xsd:complexType name="rightsSummaryType">
<xsd:documentation xml:lang="en">Definition: The rightsSummaryType schema type allows the use of rights at the asset level and the instantiation level. The rights can be expressed as a summary or a link or an embedded XML record. These can also contain time relations.</xsd:documentation>
</xsd:annotation>
<xsd:choice>
   <xsd:element maxOccurs="1" minOccurs="0" name="rightsSummary" type="sourceVersionStringType">
      <xsd:documentation xml:lang="en">Definition: The rightsSummary element is used as a general free-text element to identify information about copyrights and property rights held in and over an asset or instantiation, whether they are open access or restricted in some way. If dates, times and availability periods are associated with a right, include them. End user permissions, constraints and obligations may also be identified as needed.</xsd:documentation>
      <xsd:annotation>
         <xsd:documentation xml:lang="en">Best practice: For rights information that applies to the asset as a whole, use this element within the container pbcoreRightsSummary. For rights information that is specific to an instantiation of an asset, use it within the container instantiationRights.</xsd:documentation>
      </xsd:annotation>
   </xsd:element>
   <xsd:element maxOccurs="1" minOccurs="0" name="rightsLink" type="rightsLinkType">
      <xsd:documentation xml:lang="en">Definition: The rightsLink element is a URI pointing to a declaration of rights.</xsd:documentation>
      <xsd:annotation>
         <xsd:documentation xml:lang="en">Definition: The rightsLink element is a URI pointing to a declaration of rights.</xsd:documentation>
      </xsd:annotation>
   </xsd:element>
</xsd:choice>
</xsd:complexType>
<xsd:documentation xml:lang="en">Definition: This algorithm controls the language element to insure the use of three letter codes.</xsd:documentation>

<xsd:restriction base="xsd:string">
  <xsd:pattern value="([a-z]{3}((;[a-z]{3}))?)*"/>
<!-- allows for null -->
</xsd:restriction>
</xsd:simpleType>

<xsd:attributeGroup name="sourceVersionGroup">
<xsd:annotation>
</xsd:annotation>
<xsd:attribute name="source" type="xsd:string" use="optional">
  <xsd:documentation xml:lang="en">Definition: The source attribute provides the name of the authority used to declare the value of the element.</xsd:documentation>
</xsd:attribute>
<xsd:attribute name="ref" type="xsd:string">
  <xsd:documentation xml:lang="en">Definition: The ref attribute is used to supply a source's URI for the value of the element.</xsd:documentation>
</xsd:attribute>
<xsd:attribute name="version" type="xsd:string">
  <xsd:documentation xml:lang="en">Definition: The version attribute identifies any version information about the authority or convention used to express data of this element.</xsd:documentation>
</xsd:attribute>
<xsd:attribute name="annotation" type="xsd:string">
  <xsd:documentation xml:lang="en">Definition: The annotation attribute includes narrative information intended to clarify the nature of data used in the element.</xsd:documentation>
</xsd:attribute>
</xsd:attributeGroup>
Update to the Public Broadcasting Metadata Dictionary project

After a long process of review, we are excited to announce the updated PBCore 2.1 schema!

In deciding what changes to implement for PBCore 2.1, the PBCore Schema Team considered the following criteria:
What problems and challenges with the PBCore 2.0 schema were brought up during our open call for PBCore users to submit issues on GitHub as of September 30, 2014? What issues required a change to the schema, and what issues could be resolved by improving the documentation around PBCore elements and attributes to clarify their usage? What changes would allow the 2.1 schema to remain backwards compatible with PBCore 2.0, so that current users could continue to validate their metadata? Keep in mind that PBCore 2.1 is an incremental version, not a major release.

After balancing these considerations, we decided to implement the following schema changes for PBCore 2.1:

In 2.0, the collection of attributes that includes `@source, @ref, @version, @annotation` -- which is designed to allow catalogers to provide accurate information about the source of their metadata -- was available to most elements, but not all of them.

The updated schema provides the option to include `@source, @ref, @version, @annotation` information to all elements. This change affects:

- pbcoreDescription
- pbcoreAssetDate
- creator
- contributor
- publisher
- instantiationLocation
- instantiationDimensions
- instantiationDataRate
- instantiationFileSize
- instantiationTimeStart
- instantiationDuration
- instantiationDate
- instantiationTracks
- instantiationChannelConfiguration
- instantiationAlternativeModes
- essenceTrackType
- essenceTrackDataRate
- essenceTrackFrameRate
- essenceTrackPlaybackSpeed
- essenceTrackSamplingRate
- essenceTrackBitDepth
- essenceTrackTimeStart
- essenceTrackDuration

In all of these cases, these attributes are optional, but they will allow users to document their metadata in greater detail if they so choose. The increased ability to provide URIs for PBCore XML data elements will benefit users who wish to convert their PB Core XML records to Linked
Data. Discussions are ongoing with EBU Core to provide a common RDF ontology for this purpose.

Several PBCore elements include attributes -- specifically, the @titleType attribute (for pbcoreTitle), the @subjectType attribute (for pbcoreSubject), and the @affiliation attribute (for pbcoreCreator, pbcoreContributor, and pbcorePublisher) -- for which users also requested the ability to provide the source of the value used to express the type. In future releases of PBCore, the schema could be altered such that these attributes become elements in their own right. However, in order to comply with goal of keeping PBCore 2.1 backwards compatible, this was not possible for 2.1. Therefore, we created several new optional attribute groups for inclusion with the following elements:

for pbcoreTitle:
@titleTypeSource
@titleTypeRef
@titleTypeVersion
@titleTypeAnnotation

for pbcoreSubject:
@subjectTypeSource
@subjectTypeRef
@subjectTypeVersion
@subjectTypeAnnotation

for pbcorePart:
@partType
@partTypeSource
@partTypeRef
@partTypeVersion
@partTypeAnnotation

for creator, contributor and publisher:
@affiliationSource
@affiliationRef
@affiliationVersion
@affiliationAnnotation

In PBCore 2.0, the element essenceTrackBitDepth did not include the option to add a @unitofMeasure attribute. PBcore 2.1 now includes this optional attribute.

In PBCore 2.0, the elements instantiationLanguage and essenceTrackLanguage are not repeatable. This required that if an instantiation or essence track contains multiple languages, both of them would have to be entered in the same data field as three-letter language codes separated by a semicolon, e.g.

<instantiationLanguage>eng;fre</instantiationLanguage>.
While this form of entering data is still valid, we have made those fields repeatable in 2.1 to allow for the option of entering language information separately, e.g.

<instantiationLanguage>eng</instantiationLanguage>
<instantiationLanguage>fre</instantiationLanguage>

This allows for more specificity and searchability in entering metadata.

In order to provide more flexibility in accommodating local metadata elements and values (e.g. from an in-house database), the requirement to use extensionAuthorityUsed when using the container extensionWrap has been removed. However, we still highly recommend using this element whenever possible to document the source system or schema of the element.

One newly suggested element, to define asset version, was approved by the Schema Team. However, it was not explicitly added to the schema at this time due to the ongoing work to merge some efforts between PBCore and EBUCore (currently limited to a common RDF ontology). This element does exist in EBUCore; therefore, the team suggests that this (and other similar elements) be considered for future releases of PBCore and/or a future merger with EBUCore. In the meantime, it should be expressed in PBCore using extensions, with the EBUCore element as the extensionElement and EBUCore as extensionAuthorityUsed, as follows:

version - The purpose of this element is to express the version of the intellectual content of the asset being described. In this case, version is specific to content, not to the instantiations of that content (e.g. UK edit, Hulu version, etc.). Use the EBUCore element version to express this information. In a PBCore extension, this could look like:

    <extensionWrap>
        <extensionElement>version</extensionElement>
        <extensionValue>Hulu Version</extensionValue>
        <extensionAuthorityUsed>EBUCore</extensionAuthorityUsed>
    </extensionWrap>

The schema team found that several of the issues raised on GitHub were caused by confusion over the definition or usage of an element or attribute. Many of these were addressed by changes to the documentation, specifically the element and attribute definitions, which have been completely revised. Best practice guidelines for nearly all elements have also been added, and will appear on the website alongside definitions. Longer explanations addressing common use cases (e.g. when and how to use extensions) will be provided in blog posts on the updated PBCore website.

Several other changes were suggested over the course of this process. Many would require changes that may be implemented for the eventual release of PBCore 3.0, which will provide a broader revision of the PBCore data model. Please also note that this release does not include changes to the PBCore vocabularies. Suggested changes for these are forthcoming.

We welcome your questions and comments about PBCore 2.1!