Schéma XML pro vytvoření datového balíčku SIP

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- METS: Metadata Encoding and Transmission Standard -->
<!-- Copyright © 2001, 2002, 2003, 2004, 2005, 2006, 2008, 2009, 2010, 2012 Digital Library
Federation -->
<!-- Prepared for the Digital Library Federation by Jerome McDonough, New York University,
with the assistance of Michael Alexander (British Library), Joachim Bauer (Content Conversion
Specialists, Germany), Rick Beaubien (University of California), Terry Catapano (Columbia
University), Morgan Cundiff (Library of Congress), Susan Dahl (University of Alberta), Markus
Enders (State and University Library, Göttingen/British Library),  Richard Gartner (Bodleian
Library at Oxford/King's College, London), Thomas Habing (University of Illinois at Urbana-
Champaign), Nancy Hoebelheinrich (Stanford University/Knowledge Motifs LLC), Arwen Hutt (U.C.
San Diego), Mark Kornbluh (Michigan State University), Cecilia Preston (Preston & Lynch),
Merrilee Proffitt (Research Libraries Group), Clay Redding (Library of Congress), Jenn Riley
(Indiana University), Richard Rinehart (Berkeley Art Museum/Pacific Film Archive), Mackenzie
Smith (Massachusetts Institute of Technology), Tobias Steinke (German National Library),
Taylor Surface (OCLC), Brian Tingle (California Digital Library) and Robin Wendler (Harvard
University), Robert Wolfe (Massachusetts Institute of Technology), Patrick Yott (Brown
University).
-->
<!-- March, 2012 -->
<!-- Version 1.9.1 -->
<!-- Change History -->
<!-- April 23, 2001: Alpha Draft completed -->
<!-- June 7, 2001: Beta completed -->
<!-- 6/7/2001 Beta Changes:
  1. add 'Time' as a possible time code value, as well as TCF.
  2. Make dmdSec ID attribute required; make ID attribute optional on MDRef/MDWrap.
  3. Add 'Label' attribute to StructMap, along with 'Type'.
  4. Add DDI and FGDC as potential metadata schemes to enumeration.
  5. Enable an "otherMDtype" attribute for MDWrap/MDRef and any other element where
     there's an 'other' in the enumerated possibilities.
  6. Add a "profile" attribute to METS element.
  7. Revised mptr declaration so that it's like FLocat/MDRef (and not like XLink)
  8. Extend internal documentation of <area> attributes.
  9. Add "other" to the possible set of LOCTYPEs.
  10. Change ADMIDS to ADMID on FileGrp.
  11. Change "N" to "Order" on <div> element.
  12. Change "Number" to "order label" on <div> element
  13. Add createdate and lastmoddate attributes to mets element.
  14. Allow <div> and <area> elements to link to administrative metadata sections.
  15. Normalize attribute pointing facilities for file element and mdRef.
  16. Provide a LOCTYPE of "other" and an "otherloctype" attribute for pointing to external
files.
  17. Drop PDI from enumeration of LOCTYPES.
  18. Make MDTYPE required in mdRef and mdWrap.
  19. Rename preservationMD to digiprovMD.
  20. Add optional CHECKSUM attribute to FContent element.
  21. Modularize declarations of fileGrpType and mdSecType attributes and enumerations to
     simplify maintenance.
  22. Add TYPE attribute to structMap.
  23. Declare structMap element using structMapType rather than direct declaration.
  24. Add area element as possible subelement to <div>, along with par and seq.
  25. Change mdSec model to ALL, to enable differing order of mdRef/mdWrap elements.
  26. Extend documentation on <par> and <seq> elements.
 -->
<!-- October 22, 2001: Gamma completed -->
<!-- 10/22/2001 Gamma changes:
  1. Added optional fileSec element beneath METS root element to contain fileGrps.
  2. Created subsidiary schema file xlink.xsd for XLink attributes, restored XLink attributes
  to mptr element, and added XLink support to mdRef and FLocat.
  3. Created new element metsHdr to handle metadata regarding METS document
  itself (analogous to TEI Header).  Moved CREATEDATE and LASTMODDATE attributes
  to metsHdr, and added new RECORDSTATUS attribute.  Added new subsidiary elements
  agent and altRecordID to metsHdr.
  4. Made CREATEDATE and LASTMODDATE attributes type xsd:dateTime to allow more precise
  recording of when work was done.
  5. Changed all attributes using data type of xsd:binary to xsd:base64Binary to conform to
final
```

```
      W3C schema recommendations.
     6. Cleaned up annotations/documentation.
  -->
<!-- December 19, 2001: Epsilon and PROTOFINAL completed-->
<!-- 12/19/2001 Epsilon changes:
    1. Changed sequence operator for StructMap so that only 1 root div element is permitted.
    2. Add new roles to agent element's role attribute and support for extensible 'other' role.
    3. Add support for extensible 'other' type attribute on agent element.
    4. Yet more documentation clean up.
    5. Relocate CHECKSUM attribute from FContent to File element.
    6. Change the file element's CREATED attribute and fileGroup's VERSDATE attribute to
    a type of xsd:dateTime
    7. Change attribute name DMD for div element to DMDID for consistency's sake.
    8. Added new behaviorSec for support of referencing executable code from METS object
  -->
<!-- February 8, 2002: Zeta bug fix to final -->
<!-- 2/8/2002 Zeta changes:

    1. Eliminated redundant VRA in metadata type enumeration.
    2. Changed mdWrap content model, adding xmlData element to eliminate
       ambiguous content model
  -->
<!-- June 3, 2002: Version 1.1 -->
<!-- 6/3/2002 v1.1 changes:

     1. Add new structLink section for recording hyperlinks between media represented by
structMap nodes.
    2. Allow a <par> element to
    contain a <seq> -->
<!-- Dec. 27, 2002: Version 1.2 -->
<!-- 12/27/2002 v1.2 changes:
1. Add "USE" attribute to FileGrp, File, FLocat and FContent;
2. Make FLocat repeatable;
3. Have FContent mimic mdWrap in using separate binData/xmlData sections;
4. Copyright statement added;
5. Allow both FLocat and Fcontent in single file element;
6. Allow behaviorSec elements to group through GROUPID attribute;
7. allow descriptive and administrative metadata sections to be grouped through GROUPID
attribute;
8. allow <file> element to point to descriptive metadata via DMDID attribute;
9. allow descriptive metadata and all forms of administrative metadata to point to
administrative metadata via ADMID attribute;
10. CREATED and STATUS attributes added to all desc. and adm. metadata sections; and
11. clean up documentation in elements to reflect reality.
-->
<!-- May 8, 2003: Version 1.3 -->
<!-- 05/05/2003 v1.3 changes:

1. Change "2. OBJID: a primary identifier assigned to the original source document" to "2.
OBJID: a primary identifier assigned to the METS object."
2. Add MODS to MDTYPEs.
3. Modify <file> attributes so that instead of just CHECKSUM we have CHECKSUM and
CHECKSUMTYPE, where CHECKSUMTYPE is a controlled vocabulary as follows:
     HAVAL, MD5, SHA-1, SHA-256, SHA-384, SHA-512, TIGER, WHIRLPOOL
4.Alter BehaviorSec to make it recursive, and add a new behavior element to wrap mechanism and
interfaceDef elements.
-->
<!-- May 1, 2004: Version 1.4 -->
<!-- 05/01/2003 v1.4 changes:

1. Moved attribute documentation out of element documentation
(thank you, Brian Tingle).
2. New CONTENTIDS attribute (and URIs simpleType) added to div, fptr,
mptr and area elements for mapping MPEG21 DII Identifier values
3. XLink namespace URI changed to conform with XLink recommendation.
4. ID Attribute added to FContent.
5. ID Attribute addedt to structLink.
6. ID Attribute added to smLink.
7. "LOM" added as metadata type.
 -->
 <!-- April 12, 2005: Version 1.5 -->
 <!-- 04/12/2005 v1.5 changes:

 1. Made file element recursive to deal with PREMIS Onion Layer model and
 support XFDU-ish unpacking specification.
 2. Add <stream> element beneath <file> to allow linking of metadata to
 subfile structures.
```

```
   3. Modify structLink TO and FROM attributes to put them in XLink namespace.
   4. Make processContents "lax" for all xsd:any elements.
   -->
   <!-- October 18, 2006: Version 1.6 -->
   <!-- 10/18/2006 v1.6 changes:

   1. add ID to stream and transformFile
   2. add ADMID to metsHdr
   3. make smLink/@xlink:to and smLink/@xlink:from required
   -->
<!-- October 16, 2007/ Jan 20, 2008: Version 1.7 -->
<!-- 10/16/2007 01/30/2008  v 1.7 changes:

1. create parType complex type to allow a seq to contain a par
2. create FILECORE attribute group with MIMETYPE, SIZE, CHECKSUM, CHECKSUMTYPE;
     change fileType, mdWrapType and mdRefType use the attribute group, so mdType and mdRef
end
     up with new SIZE, CHECKSUM, and CHECKSUMTYPE attributes (file does not change)
20080130
2a. CREATED added to FILECORE
3. PREMIS:OBJECT PREMIS:AGENT PREMIS:RIGHTS PREMIS:EVENT added to MDTYPE value enumeration
-->
<!-- April 2009: Version 1.8 -->
<!-- Version 1.8 changes:
   1. Add CRC32, Adler-32, MNP to the enumerated values constraining CHECKSUMTYPE to align with
MIX messageDigestAlgorithm constraints.
   2. Add TEXTMD and METSRIGHTS to the enumeration values constraining MDTYPE.
   3. Add an MDTYPEVERSION attribute as a companion to the MDTYPE attribute in the mdRef and
mdWrap elements.
   4. ID and STRUCTID attributes on the behavior element made optional.  Depending on whether
the behavior applies to a transformFile element or div elements in the structMap, only one or
the other of the attributes would pertain.
   5. Documentation aligned with the METS Primer, and corrected.
   6. xml:lang="en" atttribute value added to every <documentation> element
   7. xlink:extendedLink support added to the <structLink> element by means of a new
<smLinkGrp> element, and its child <smLocatorLink> and <smArcLink> elements.
-->
<!--February 2010: Version 1.9-->
<!--Version 1.9 Changes:
   1. Added a <metsDocumentID> element to the <metsHdr> for recording a unique identifier for
the METS document itself where this is different from the OBJID, the identifier for the entire
digital object represented by the METS document.
   2. Added "ISO 19115:2003 NAP" to the enumerated values for the MDTYPE attribute in the
METADATA attribute group.
   3. Added "XPTR" to the enumerated values for the BETYPE attribute on the areaType data type
   4. Added BEGIN, END and BETYPE attributes to the <file> and <stream> elements for specifying
the location of a nested file or a stream within it's parent file.
-->
<!-- March 2012: Version 1.9.1 -->
<!-- Version 1.9.1 Changes::
   1.  Added 'EAC-CPF' as potential metadata scheme to MDTYPE enumeration
     EAC-CPF = Encoded Archival Context - Corporate Bodies, Persons, and Families
     http://eac.staatsbibliothek-berlin.de/eac-cpf-schema.html
-->
<xsd:schema targetNamespace="http://www.loc.gov/METS/" xmlns="http://www.loc.gov/METS/"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xsd:import namespace="http://www.w3.org/1999/xlink"
schemaLocation="http://www.loc.gov/standards/xlink/xlink.xsd"/>

  <xsd:element name="mets">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">METS: Metadata Encoding and Transmission Standard.
        METS is intended to provide a standardized XML format for transmission of complex
digital library objects between systems.  As such, it can be seen as filling a role similar to
that defined for the Submission Information Package (SIP), Archival Information Package (AIP)
and Dissemination Information Package (DIP) in the Reference Model for an Open Archival
Information System. The root element &lt;mets&gt; establishes the container for the
information being stored and/or transmitted by the standard.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="metsType"/>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>
```

```xml
<xsd:complexType name="metsType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">metsType: Complex Type for METS Sections
    A METS document consists of seven possible subsidiary sections: metsHdr (METS document
header), dmdSec (descriptive metadata section), amdSec (administrative metadata section),
fileGrp (file inventory group), structLink (structural map linking), structMap (structural
map) and behaviorSec (behaviors section).
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="metsHdr" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
        The mets header element &lt;metsHdr&gt; captures metadata about the METS document
itself, not the digital object the METS document encodes. Although it records a more limited
set of metadata, it is very similar in function and purpose to the headers employed in other
schema such as the Text Encoding Initiative (TEI) or in the Encoded Archival Description
(EAD).
    </xsd:documentation>
      </xsd:annotation>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="agent" minOccurs="0" maxOccurs="unbounded">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">agent:
              The agent element &lt;agent&gt; provides for various parties and their roles
with respect to the METS record to be documented.
              </xsd:documentation>
            </xsd:annotation>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name="name" type="xsd:string">
                  <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                    The element &lt;name&gt; can be used to record the full name of the
document agent.
                    </xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="note" type="xsd:string" minOccurs="0"
maxOccurs="unbounded">
                  <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                    The &lt;note&gt; element can be used to record any additional
information regarding the agent's activities with respect to the METS document.
                    </xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
              <xsd:attribute name="ID" type="xsd:ID" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely
identifies the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="ROLE" use="required">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">ROLE (string/R): Specifies the function
of the agent with respect to the METS record. The allowed values are:
CREATOR: The person(s) or institution(s) responsible for the METS document.
EDITOR: The person(s) or institution(s) that prepares the metadata for encoding.
ARCHIVIST: The person(s) or institution(s) responsible for the document/collection.
PRESERVATION: The person(s) or institution(s) responsible for preservation functions.
DISSEMINATOR: The person(s) or institution(s) responsible for dissemination functions.
CUSTODIAN: The person(s) or institution(s) charged with the oversight of a
document/collection.
IPOWNER: Intellectual Property Owner: The person(s) or institution holding copyright, trade or
service marks or other intellectual property rights for the object.
OTHER: Use OTHER if none of the preceding values pertains and clarify the type and location
specifier being used in the OTHERROLE attribute (see below).
                  </xsd:documentation>
                </xsd:annotation>
                <xsd:simpleType>
                  <xsd:restriction base="xsd:string">
```

```xml
                    <xsd:enumeration value="CREATOR"/>
                    <xsd:enumeration value="EDITOR"/>
                    <xsd:enumeration value="ARCHIVIST"/>
                    <xsd:enumeration value="PRESERVATION"/>
                    <xsd:enumeration value="DISSEMINATOR"/>
                    <xsd:enumeration value="CUSTODIAN"/>
                    <xsd:enumeration value="IPOWNER"/>
                    <xsd:enumeration value="OTHER"/>
                  </xsd:restriction>
                </xsd:simpleType>
              </xsd:attribute>
              <xsd:attribute name="OTHERROLE" type="xsd:string" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">OTHERROLE (string/O): Denotes a role not
contained in the allowed values set if OTHER is indicated in the ROLE attribute.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="TYPE" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">TYPE (string/O): is used to specify the
type of AGENT. It must be one of the following values:
INDIVIDUAL: Use if an individual has served as the agent.
ORGANIZATION: Use if an institution, corporate body, association, non-profit enterprise,
government, religious body, etc. has served as the agent.
OTHER: Use OTHER if none of the preceding values pertain and clarify the type of agent
specifier being used in the OTHERTYPE attribute
                  </xsd:documentation>
                </xsd:annotation>
                <xsd:simpleType>
                  <xsd:restriction base="xsd:string">
                    <xsd:enumeration value="INDIVIDUAL"/>
                    <xsd:enumeration value="ORGANIZATION"/>
                    <xsd:enumeration value="OTHER"/>
                  </xsd:restriction>
                </xsd:simpleType>
              </xsd:attribute>
              <xsd:attribute name="OTHERTYPE" type="xsd:string" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">OTHERTYPE (string/O): Specifies the type
of agent when the value OTHER is indicated in the TYPE attribute.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
            </xsd:complexType>
          </xsd:element>
          <xsd:element name="altRecordID" minOccurs="0" maxOccurs="unbounded">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">
                The alternative record identifier element &lt;altRecordID&gt; allows one to
use alternative record identifier values for the digital object represented by the METS
document; the primary record identifier is stored in the OBJID attribute in the root
&lt;mets&gt; element.
              </xsd:documentation>
            </xsd:annotation>
            <xsd:complexType>
              <xsd:simpleContent>
                <xsd:extension base="xsd:string">
                  <xsd:attribute name="ID" type="xsd:ID" use="optional">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely
identifies the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                      </xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
                  <xsd:attribute name="TYPE" type="xsd:string" use="optional">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">TYPE (string/O): A description of the
identifier type (e.g., OCLC record number, LCCN, etc.).
                      </xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
                </xsd:extension>
              </xsd:simpleContent>
            </xsd:complexType>
```

```
          </xsd:element>
          <xsd:element name="metsDocumentID" minOccurs="0">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">
                The metsDocument identifier element &lt;metsDocumentID&gt; allows a unique
identifier to be assigned to the METS document itself.  This may be different from the OBJID
attribute value in the root &lt;mets&gt; element, which uniquely identifies the entire digital
object represented by the METS document.
              </xsd:documentation>
            </xsd:annotation>
            <xsd:complexType>
              <xsd:simpleContent>
                <xsd:extension base="xsd:string">
                  <xsd:attribute name="ID" type="xsd:ID" use="optional">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely
identifies the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                      </xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
                  <xsd:attribute name="TYPE" type="xsd:string" use="optional">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">TYPE (string/O): A description of the
identifier type.
                      </xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
                </xsd:extension>
              </xsd:simpleContent>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
        <xsd:attribute name="ID" type="xsd:ID" use="optional">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:attribute>
        <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">ADMID (IDREFS/O): Contains the ID attribute
values of the &lt;techMD&gt;, &lt;sourceMD&gt;, &lt;rightsMD&gt; and/or &lt;digiprovMD&gt;
elements within the &lt;amdSec&gt; of the METS document that contain administrative metadata
pertaining to the METS document itself.  For more information on using METS IDREFS and IDREF
type attributes for internal linking, see Chapter 4 of the METS Primer.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:attribute>
        <xsd:attribute name="CREATEDATE" type="xsd:dateTime" use="optional">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">CREATEDATE (dateTime/O): Records the date/time
the METS document was created.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:attribute>
        <xsd:attribute name="LASTMODDATE" type="xsd:dateTime" use="optional">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">LASTMODDATE (dateTime/O): Is used to indicate
the date/time the METS document was last modified.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:attribute>
        <xsd:attribute name="RECORDSTATUS" type="xsd:string" use="optional">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">RECORDSTATUS (string/O): Specifies the status
of the METS document. It is used for internal processing purposes.
            </xsd:documentation>
          </xsd:annotation>
        </xsd:attribute>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="dmdSec" type="mdSecType" minOccurs="0" maxOccurs="unbounded">
```

```
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            A descriptive metadata section &lt;dmdSec&gt; records descriptive metadata
pertaining to the METS object as a whole or one of its components. The &lt;dmdSec&gt; element
conforms to same generic datatype as the &lt;techMD&gt;, &lt;rightsMD&gt;, &lt;sourceMD&gt;
and &lt;digiprovMD&gt; elements, and supports the same sub-elements and attributes. A
descriptive metadata element can either wrap the metadata  (mdWrap) or reference it in an
external location (mdRef) or both.  METS allows multiple &lt;dmdSec&gt; elements; and
descriptive metadata can be associated with any METS element that supports a DMDID attribute.
Descriptive metadata can be expressed according to many current description standards (i.e.,
MARC, MODS, Dublin Core, TEI Header, EAD, VRA, FGDC, DDI) or a locally produced XML schema.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="amdSec" type="amdSecType" minOccurs="0" maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The administrative metadata section &lt;amdSec&gt; contains the administrative
metadata pertaining to the digital object, its components and any original source material
from which the digital object is derived. The &lt;amdSec&gt; is separated into four sub-
sections that accommodate technical metadata (techMD), intellectual property rights
(rightsMD), analog/digital source metadata (sourceMD), and digital provenance metadata
(digiprovMD). Each of these subsections can either wrap the metadata  (mdWrap) or reference it
in an external location (mdRef) or both. Multiple instances of the &lt;amdSec&gt; element can
occur within a METS document and multiple instances of its subsections can occur in one
&lt;amdSec&gt; element. This allows considerable flexibility in the structuring of the
administrative metadata. METS does not define a vocabulary or syntax for encoding
administrative metadata. Administrative metadata can be expressed within the amdSec sub-
elements according to many current community defined standards, or locally produced XML
schemas. </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="fileSec" minOccurs="0">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The overall purpose of the content file section element &lt;fileSec&gt; is to
provide an inventory of and the location for the content files that comprise the digital
object being described in the METS document.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="fileGrp" maxOccurs="unbounded">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  A sequence of file group elements &lt;fileGrp&gt; can be used group the
digital files comprising the content of a METS object either into a flat arrangement or,
because each file group element can itself contain one or more  file group elements,  into a
nested (hierarchical) arrangement. In the case where the content files are images of different
formats and resolutions, for example, one could group the image content files by format and
create a separate &lt;fileGrp&gt; for each image format/resolution such as:
-- one &lt;fileGrp&gt; for the thumbnails of the images
-- one &lt;fileGrp&gt; for the higher resolution JPEGs of the image
-- one &lt;fileGrp&gt; for the master archival TIFFs of the images
For a text resource with a variety of content file types one might group the content files at
the highest level by type,  and then use the &lt;fileGrp&gt; element's nesting capabilities to
subdivide a &lt;fileGrp&gt; by format within the type, such as:
-- one &lt;fileGrp&gt; for all of the page images with nested &lt;fileGrp&gt; elements for
each image format/resolution (tiff, jpeg, gif)
-- one &lt;fileGrp&gt; for a PDF version of all the pages of the document
-- one &lt;fileGrp&gt; for  a TEI encoded XML version of the entire document or each of its
pages.
A &lt;fileGrp&gt; may contain zero or more &lt;fileGrp&gt; elements and or &lt;file&gt;
elements.
                </xsd:documentation>
              </xsd:annotation>
              <xsd:complexType>
                <xsd:complexContent>
                  <xsd:extension base="fileGrpType"/>
                </xsd:complexContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
          <xsd:attribute name="ID" type="xsd:ID" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
```

```
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                    </xsd:documentation>
                </xsd:annotation>
            </xsd:attribute>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="structMap" type="structMapType" maxOccurs="unbounded">
        <xsd:annotation>
            <xsd:documentation xml:lang="en">
                The structural map section &lt;structMap&gt; is the heart of a METS document. It
provides a means for organizing the digital content represented by the &lt;file&gt; elements
in the &lt;fileSec&gt; of the METS document into a coherent hierarchical structure. Such a
hierarchical structure can be presented to users to facilitate their comprehension and
navigation of the digital content. It can further be applied to any purpose requiring an
understanding of the structural relationship of the content files or parts of the content
files. The organization may be specified to any level of granularity (intellectual and or
physical) that is desired. Since the &lt;structMap&gt; element is repeatable, more than one
organization can be applied to the digital content represented by the METS document.  The
hierarchical structure specified by a &lt;structMap&gt; is encoded as a tree of nested
&lt;div&gt; elements. A &lt;div&gt; element may directly point to content via child file
pointer &lt;fptr&gt; elements (if the content is represented in the &lt;fileSec&lt;) or child
METS pointer &lt;mptr&gt; elements (if the content is represented by an external METS
document). The &lt;fptr&gt; element may point to a single whole &lt;file&gt; element that
manifests its parent &lt;div&lt;, or to part of a &lt;file&gt; that manifests its &lt;div&lt;.
It can also point to multiple files or parts of files that must be played/displayed either in
sequence or in parallel to reveal its structural division. In addition to providing a means
for organizing content, the &lt;structMap&gt; provides a mechanism for linking content at any
hierarchical level with relevant descriptive and administrative metadata.
                    </xsd:documentation>
                </xsd:annotation>
            </xsd:element>
            <xsd:element name="structLink" minOccurs="0">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                        The structural link section element &lt;structLink&gt; allows for the
specification of hyperlinks between the different components of a METS structure that are
delineated in a structural map. This element is a container for a single, repeatable element,
&lt;smLink&gt; which indicates a hyperlink between two nodes in the structural map. The
&lt;structLink&gt; section in the METS document is identified using its XML ID attributes.
                    </xsd:documentation>
                </xsd:annotation>
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="structLinkType"/>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="behaviorSec" type="behaviorSecType" minOccurs="0"
maxOccurs="unbounded">
                <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                        A behavior section element &lt;behaviorSec&gt; associates executable behaviors
with content in the METS document by means of a repeatable behavior &lt;behavior&gt; element.
This element has an interface definition &lt;interfaceDef&gt; element that represents an
abstract definition of the set of behaviors represented by a particular behavior section. A
&lt;behavior&gt; element also has a &lt;mechanism&gt; element which is used to point to a
module of executable code that implements and runs the behavior defined by the interface
definition. The &lt;behaviorSec&gt; element, which is repeatable as well as nestable, can be
used to group individual behaviors within the structure of the METS document. Such grouping
can be useful for organizing families of behaviors together or to indicate other relationships
between particular behaviors.</xsd:documentation>
                </xsd:annotation>
            </xsd:element>
        </xsd:sequence>
        <xsd:attribute name="ID" type="xsd:ID" use="optional">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:attribute>
        <xsd:attribute name="OBJID" type="xsd:string" use="optional">
            <xsd:annotation>
```

```
      <xsd:documentation xml:lang="en">OBJID (string/O): Is the primary identifier assigned
to the METS object as a whole. Although this attribute is not required, it is strongly
recommended. This identifier is used to tag the entire METS object to external systems, in
contrast with the ID identifier.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="LABEL" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">LABEL (string/O): Is a simple title string used to
identify the object/entity being described in the METS document for the user.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="TYPE" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">TYPE (string/O): Specifies the class or type of the
object, e.g.: book, journal, stereograph, dataset, video, etc.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="PROFILE" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">PROFILE (string/O): Indicates to which of the
registered profile(s) the METS document conforms. For additional information about PROFILES
see Chapter 5 of the METS Primer.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>
<xsd:complexType name="amdSecType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">amdSecType: Complex Type for Administrative Metadata
Sections
    The administrative metadata section consists of four possible subsidiary sections:
techMD (technical metadata for text/image/audio/video files), rightsMD (intellectual property
rights metadata), sourceMD (analog/digital source metadata), and digiprovMD (digital
provenance metadata, that is, the history of migrations/translations performed on a digital
library object from it's original digital capture/encoding).
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="techMD" type="mdSecType" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
        A technical metadata element &lt;techMD&gt; records technical metadata about a
component of the METS object, such as a digital content file. The &lt;techMD&gt; element
conforms to same generic datatype as the &lt;dmdSec&gt;, &lt;rightsMD&gt;, &lt;sourceMD&gt;
and &lt;digiprovMD&gt; elements, and supports the same sub-elements and attributes.  A
technical metadata element can either wrap the metadata  (mdWrap) or reference it in an
external location (mdRef) or both.  METS allows multiple &lt;techMD&gt; elements; and
technical metadata can be associated with any METS element that supports an ADMID attribute.
Technical metadata can be expressed according to many current technical description standards
(such as MIX and textMD) or a locally produced XML schema.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="rightsMD" type="mdSecType" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
        An intellectual property rights metadata element &lt;rightsMD&gt; records
information about copyright and licensing pertaining to a component of the METS object. The
&lt;rightsMD&gt; element conforms to same generic datatype as the &lt;dmdSec&gt;, &lt;techMD&gt;,
&lt;sourceMD&gt; and &lt;digiprovMD&gt; elements, and supports the same sub-elements and
attributes. A rights metadata element can either wrap the metadata  (mdWrap) or reference it
in an external location (mdRef) or both.  METS allows multiple &lt;rightsMD&gt; elements; and
rights metadata can be associated with any METS element that supports an ADMID attribute.
Rights metadata can be expressed according current rights description standards (such as
CopyrightMD and rightsDeclarationMD) or a locally produced XML schema.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="sourceMD" type="mdSecType" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
        A source metadata element &lt;sourceMD&gt; records descriptive and administrative
metadata about the source format or media of a component of the METS object such as a digital
```

content file. It is often used for discovery, data administration or preservation of the digital object. The <sourceMD> element conforms to same generic datatype as the <dmdSec>, <techMD>, <rightsMD>, and <digiprovMD> elements, and supports the same sub-elements and attributes. A source metadata element can either wrap the metadata (mdWrap) or reference it in an external location (mdRef) or both. METS allows multiple <sourceMD> elements; and source metadata can be associated with any METS element that supports an ADMID attribute. Source metadata can be expressed according to current source description standards (such as PREMIS) or a locally produced XML schema.
                 </xsd:documentation>
             </xsd:annotation>
         </xsd:element>
         <xsd:element name="digiprovMD" type="mdSecType" minOccurs="0" maxOccurs="unbounded">
             <xsd:annotation>
                 <xsd:documentation xml:lang="en">
                     A digital provenance metadata element <digiprovMD> can be used to record any preservation-related actions taken on the various files which comprise a digital object (e.g., those subsequent to the initial digitization of the files such as transformation or migrations) or, in the case of born digital materials, the files' creation. In short, digital provenance should be used to record information that allows both archival/library staff and scholars to understand what modifications have been made to a digital object and/or its constituent parts during its life cycle. This information can then be used to judge how those processes might have altered or corrupted the object's ability to accurately represent the original item. One might, for example, record master derivative relationships and the process by which those derivations have been created. Or the <digiprovMD> element could contain information regarding the migration/transformation of a file from its original digitization (e.g., OCR, TEI, etc.,)to its current incarnation as a digital object (e.g., JPEG2000). The <digiprovMD> element conforms to same generic datatype as the <dmdSec>, <techMD>, <rightsMD>, and <sourceMD> elements, and supports the same sub-elements and attributes. A digital provenance metadata element can either wrap the metadata (mdWrap) or reference it in an external location (mdRef) or both. METS allows multiple <digiprovMD> elements; and digital provenance metadata can be associated with any METS element that supports an ADMID attribute. Digital provenance metadata can be expressed according to current digital provenance description standards (such as PREMIS) or a locally produced XML schema.
                 </xsd:documentation>
             </xsd:annotation>
         </xsd:element>
     </xsd:sequence>
     <xsd:attribute name="ID" type="xsd:ID" use="optional">
         <xsd:annotation>
             <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the element within the METS document, and would allow the element to be referenced unambiguously from another element or document via an IDREF or an XPTR. For more information on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
             </xsd:documentation>
         </xsd:annotation>
     </xsd:attribute>
 </xsd:complexType>
 <xsd:complexType name="fileGrpType">
     <xsd:annotation>
         <xsd:documentation xml:lang="en">fileGrpType: Complex Type for File Groups
             The file group is used to cluster all of the digital files composing a digital library object in a hierarchical arrangement (fileGrp is recursively defined to enable the creation of the hierarchy). Any file group may contain zero or more file elements. File elements in turn can contain one or more FLocat elements (a pointer to a file containing content for this object) and/or a FContent element (the contents of the file, in either XML or Base64 encoding).
         </xsd:documentation>
     </xsd:annotation>
     <xsd:choice>
         <xsd:element name="fileGrp" type="fileGrpType" minOccurs="0" maxOccurs="unbounded"/>
         <xsd:element name="file" minOccurs="0" maxOccurs="unbounded" type="fileType" >
             <xsd:annotation>
                 <xsd:documentation xml:lang="en">
                     The file element <file> provides access to the content files for the digital object being described by the METS document. A <file> element may contain one or more <FLocat> elements which provide pointers to a content file and/or a <FContent> element which wraps an encoded version of the file. Embedding files using <FContent> can be a valuable feature for exchanging digital objects between repositories or for archiving versions of digital objects for off-site storage. All <FLocat> and <FContent> elements should identify and/or contain identical copies of a single file. The <file> element is recursive, thus allowing sub-files or component files of a larger file to be listed in the inventory. Alternatively, by using the <stream> element, a smaller component of a file or of a related file can be placed within a <file> element. Finally, by using the <transformFile> element, it is possible to include within a <file> element a different version of a file that has undergone a transformation for some reason, such as format migration.

```
          </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:choice>
    <xsd:attribute name="ID" type="xsd:ID" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="VERSDATE" type="xsd:dateTime" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">VERSDATE (dateTime/O): An optional dateTime attribute
specifying the date this version/fileGrp of the digital object was created.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ADMID (IDREF/O): Contains the ID attribute values of
the &lt;techMD&gt;, &lt;sourceMD&gt;, &lt;rightsMD&gt; and/or &lt;digiprovMD&gt; elements
within the &lt;amdSec&gt; of the METS document applicable to all of the files in a particular
file group. For more information on using METS IDREFS and IDREF type attributes for internal
linking, see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="USE" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">USE (string/O): A tagging attribute to indicate the
intended use of files within this file group (e.g., master, reference, thumbnails for image
files). A USE attribute can be expressed at the&lt;fileGrp&gt; level, the &lt;file&gt; level,
the &lt;FLocat&gt; level and/or the &lt;FContent&gt; level.  A USE attribute value at the
&lt;fileGrp&gt; level should pertain to all of the files in the &lt;fileGrp&gt;.  A USE
attribute at the &lt;file&gt; level should pertain to all copies of the file as represented by
subsidiary &lt;FLocat&gt; and/or &lt;FContent&gt; elements.  A USE attribute at the
&lt;FLocat&gt; or &lt;FContent&gt; level pertains to the particular copy of the file that is
either referenced (&lt;FLocat&gt;) or wrapped (&lt;FContent&gt;).
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="structMapType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">structMapType: Complex Type for Structural Maps
    The structural map (structMap) outlines a hierarchical structure for the original object
being encoded, using a series of nested div elements.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="div" type="divType">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The structural divisions of the hierarchical organization provided by a
&lt;structMap&gt; are represented by division &lt;div&gt; elements, which can be nested to any
depth. Each &lt;div&gt; element can represent either an intellectual (logical) division or a
physical division. Every &lt;div&gt; node in the structural map hierarchy may be connected
(via subsidiary &lt;mptr&gt; or &lt;fptr&gt; elements) to content files which represent that
div's portion of the whole document.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="ID" type="xsd:ID" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="TYPE" type="xsd:string" use="optional">
      <xsd:annotation>
```

```
          <xsd:documentation xml:lang="en">TYPE (string/O): Identifies the type of structure
represented by the &lt;structMap&gt;. For example, a &lt;structMap&gt; that represented a
purely logical or intellectual structure could be assigned a TYPE value of "logical" whereas a
&lt;structMap&gt; that represented a purely physical structure could be assigned a TYPE value
of "physical". However, the METS schema neither defines nor requires a common vocabulary for
this attribute. A METS profile, however, may well constrain the values for the
&lt;structMap&gt; TYPE.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="LABEL" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">LABEL (string/O): Describes the &lt;structMap&gt; to
viewers of the METS document. This would be useful primarily where more than one
&lt;structMap&gt; is provided for a single object. A descriptive LABEL value, in that case,
could clarify to users the purpose of each of the available structMaps.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="divType">

    <xsd:annotation>
      <xsd:documentation xml:lang="en">divType: Complex Type for Divisions
        The METS standard represents a document structurally as a series of nested div
elements, that is, as a hierarchy (e.g., a book, which is composed of chapters, which are
composed of subchapters, which are composed of text).  Every div node in the structural map
hierarchy may be connected (via subsidiary mptr or fptr elements) to content files which
represent that div's portion of the whole document.

SPECIAL NOTE REGARDING DIV ATTRIBUTE VALUES:
to clarify the differences between the ORDER, ORDERLABEL, and LABEL attributes for the
&lt;div&gt; element, imagine a text with 10 roman numbered pages followed by 10 arabic
numbered pages. Page iii would have an ORDER of &quot;3&quot;, an ORDERLABEL of
&quot;iii&quot; and a LABEL of &quot;Page iii&quot;, while page 3 would have an ORDER of
&quot;13&quot;, an ORDERLABEL of &quot;3&quot; and a LABEL of &quot;Page 3&quot;.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="mptr" minOccurs="0" maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            Like the &lt;fptr&gt; element, the METS pointer element &lt;mptr&gt; represents
digital content that manifests its parent &lt;div&gt; element. Unlike the &lt;fptr&gt;, which
either directly or indirectly points to content represented in the &lt;fileSec&gt; of the
parent METS document, the &lt;mptr&gt; element points to content represented by an external
METS document. Thus, this element allows multiple discrete and separate METS documents to be
organized at a higher level by a separate METS document. For example, METS documents
representing the individual issues in the series of a journal could be grouped together and
organized by a higher level METS document that represents the entire journal series. Each of
the &lt;div&gt; elements in the &lt;structMap&gt; of the METS document representing the
journal series would point to a METS document representing an issue.  It would do so via a
child &lt;mptr&gt; element. Thus the &lt;mptr&gt; element gives METS users considerable
flexibility in managing the depth of the &lt;structMap&gt; hierarchy of individual METS
documents. The &lt;mptr&gt; element points to an external METS document by means of an
xlink:href attribute and associated XLink attributes.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:attribute name="ID" type="xsd:ID" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
            </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
          <xsd:attributeGroup ref="LOCATION"/>
          <xsd:attributeGroup ref="xlink:simpleLink"/>
          <xsd:attribute name="CONTENTIDS" type="URIs" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">CONTENTIDS (URI/O): Content IDs for the
content represented by the &lt;mptr&gt; (equivalent to DIDL DII or Digital Item Identifier, a
unique external ID).
                </xsd:documentation>
              </xsd:annotation>
```

```
                    </xsd:attribute>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="fptr" minOccurs="0" maxOccurs="unbounded">
            <xsd:annotation>
                <xsd:documentation xml:lang="en">
                    The &lt;fptr&gt; or file pointer element represents digital content that manifests
its parent &lt;div&gt; element. The content represented by an &lt;fptr&gt; element must
consist of integral files or parts of files that are represented by &lt;file&gt; elements in
the &lt;fileSec&gt;. Via its FILEID attribute,  an &lt;fptr&gt; may point directly to a single
integral &lt;file&gt; element that manifests a structural division. However, an &lt;fptr&gt;
element may also govern an &lt;area&gt; element,  a &lt;par&gt;,  or  a &lt;seq&gt;  which in
turn would point to the relevant file or files. A child &lt;area&gt; element can point to part
of a &lt;file&gt; that manifests a division, while the &lt;par&gt; and &lt;seq&gt; elements
can point to multiple files or parts of files that together manifest a division. More than one
&lt;fptr&gt; element can be associated with a &lt;div&gt; element. Typically sibling
&lt;fptr&gt; elements represent alternative versions, or manifestations, of the same content
                </xsd:documentation>
            </xsd:annotation>
            <xsd:complexType>
                <xsd:choice>
                    <xsd:element name="par" type="parType" minOccurs="0">
                        <xsd:annotation>
                            <xsd:documentation xml:lang="en">
                                The &lt;par&gt; or parallel files element aggregates pointers to files,
parts of files, and/or sequences of files or parts of files that must be played or displayed
simultaneously to manifest a block of digital content represented by an &lt;fptr&gt; element.
This might be the case, for example, with multi-media content, where a still image might have
an accompanying audio track that comments on the still image. In this case, a &lt;par&gt;
element would aggregate two &lt;area&gt; elements, one of which pointed to the image file and
one of which pointed to the audio file that must be played in conjunction with the image. The
&lt;area&gt; element associated with the image could be further qualified with SHAPE and
COORDS attributes if only a portion of the image file was pertinent and the &lt;area&gt;
element associated with the audio file could be further qualified with BETYPE, BEGIN, EXTTYPE,
and EXTENT attributes if only a portion of the associated audio file should be played in
conjunction with the image.
                            </xsd:documentation>
                        </xsd:annotation>
                    </xsd:element>
                    <xsd:element name="seq" type="seqType" minOccurs="0">
                        <xsd:annotation>
                            <xsd:documentation xml:lang="en">
                                The sequence of files element &lt;seq&gt; aggregates pointers to files,
parts of files and/or parallel sets of files or parts of files  that must be played or
displayed sequentially to manifest a block of digital content. This might be the case, for
example, if the parent &lt;div&gt; element represented a logical division, such as a diary
entry, that spanned multiple pages of a diary and, hence, multiple page image files. In this
case, a &lt;seq&gt; element would aggregate multiple, sequentially arranged &lt;area&gt;
elements, each of which pointed to one of the image files that must be presented sequentially
to manifest the entire diary entry. If the diary entry started in the middle of a page, then
the first &lt;area&gt; element (representing the page on which the diary entry starts) might
be further qualified, via its SHAPE and COORDS attributes, to specify the specific, pertinent
area of the associated image file.
                            </xsd:documentation>
                        </xsd:annotation>
                    </xsd:element>
                    <xsd:element name="area" type="areaType" minOccurs="0">
                        <xsd:annotation>
                            <xsd:documentation xml:lang="en">
                                The area element &lt;area&gt; typically points to content consisting of just
a portion or area of a file represented by a &lt;file&gt; element in the &lt;fileSec&gt;. In
some contexts, however, the &lt;area&gt; element can also point to content represented by an
integral file. A single &lt;area&gt; element would appear as the direct child of a
&lt;fptr&gt; element when only a portion of a &lt;file&gt;, rather than an integral
&lt;file&gt;, manifested the digital content represented by the &lt;fptr&gt;. Multiple
&lt;area&gt; elements would appear as the direct children of a &lt;par&gt; element or a
&lt;seq&gt; element when multiple files or parts of files manifested the digital content
represented by an &lt;fptr&gt; element. When used in the context of a &lt;par&gt; or
&lt;seq&gt; element an &lt;area&gt; element can point either to an integral file or to a
segment of a file as necessary.
                            </xsd:documentation>
                        </xsd:annotation>
                    </xsd:element>
                </xsd:choice>
                <xsd:attribute name="ID" type="xsd:ID" use="optional">
                    <xsd:annotation>
```

```
                  <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="FILEID" type="xsd:IDREF" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">FILEID (IDREF/O): An optional attribute that
provides the XML ID identifying the &lt;file&gt; element that links to and/or contains the
digital content represented by the &lt;fptr&gt;. A &lt;fptr&gt; element should only have a
FILEID attribute value if it does not have a child &lt;area&gt;, &lt;par&gt; or &lt;seq&gt;
element. If it has a child element, then the responsibility for pointing to the relevant
content falls to this child element or its descendants.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="CONTENTIDS" type="URIs" use="optional">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">CONTENTIDS (URI/O): Content IDs for the
content represented by the &lt;fptr&gt; (equivalent to DIDL DII or Digital Item Identifier, a
unique external ID).
                      </xsd:documentation>
                    </xsd:annotation>
                     </xsd:attribute>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="div" type="divType" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="ID" type="xsd:ID" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="ORDER" type="xsd:integer" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">ORDER (integer/O): A representation of the div's
order among its siblings (e.g., its absolute, numeric sequence). For an example, and
clarification of the distinction between ORDER and ORDERLABEL, see the description of the
ORDERLABEL attribute.
        </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="ORDERLABEL" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">ORDERLABEL (string/O): A representation of the div's
order among its siblings (e.g., "xii"), or of any non-integer native numbering system. It is
presumed that this value will still be machine actionable (e.g., it would support 'go to page
___' function), and it should not be used as a replacement/substitute for the LABEL attribute.
To understand the differences between ORDER, ORDERLABEL and LABEL, imagine a text with 10
roman numbered pages followed by 10 arabic numbered pages. Page iii would have an ORDER of
"3", an ORDERLABEL of "iii" and a LABEL of "Page iii", while page 3 would have an ORDER of
"13", an ORDERLABEL of "3" and a LABEL of "Page 3".
        </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="LABEL" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">LABEL (string/O): An attribute used, for example, to
identify a &lt;div&gt; to an end user viewing the document. Thus a hierarchical arrangement of
the &lt;div&gt; LABEL values could provide a table of contents to the digital content
represented by a METS document and facilitate the users' navigation of the digital object.
Note that a &lt;div&gt; LABEL should be specific to its level in the structural map. In the
case of a book with chapters, the book &lt;div&gt; LABEL should have the book title and the
chapter &lt;div&gt;; LABELs should have the individual chapter titles, rather than having the
chapter &lt;div&gt; LABELs combine both book title and chapter title . For further of the
distinction between LABEL and ORDERLABEL see the description of the ORDERLABEL attribute.
        </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="DMDID" type="xsd:IDREFS" use="optional">
        <xsd:annotation>
```

```
        <xsd:documentation xml:lang="en">DMDID (IDREFS/O): Contains the ID attribute values
identifying the &lt;dmdSec&gt;, elements in the METS document that contain or link to
descriptive metadata pertaining to the structural division represented by the current
&lt;div&gt; element.  For more information on using METS IDREFS and IDREF type attributes for
internal linking, see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ADMID (IDREFS/O): Contains the ID attribute values
identifying the &lt;rightsMD&gt;, &lt;sourceMD&gt;, &lt;techMD&gt; and/or &lt;digiprovMD&gt;
elements within the &lt;amdSec&gt; of the METS document that contain or link to administrative
metadata pertaining to the structural division represented by the &lt;div&gt; element.
Typically the &lt;div&gt; ADMID attribute would be used to identify the &lt;rightsMD&gt;
element or elements that pertain to the &lt;div&gt;, but it could be used anytime there was a
need to link a &lt;div&gt; with pertinent administrative metadata. For more information on
using METS IDREFS and IDREF type attributes for internal linking, see Chapter 4 of the METS
Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="TYPE" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">TYPE (string/O): An attribute that specifies the type
of structural division that the &lt;div&gt; element represents. Possible &lt;div&gt; TYPE
attribute values include: chapter, article, page, track, segment, section etc. METS places no
constraints on the possible TYPE values. Suggestions for controlled vocabularies for TYPE may
be found on the METS website.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="CONTENTIDS" type="URIs" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">CONTENTIDS (URI/O): Content IDs for the content
represented by the &lt;div&gt; (equivalent to DIDL DII or Digital Item Identifier, a unique
external ID).
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute ref="xlink:label">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">xlink:label - an xlink label to be referred to by an
smLink element</xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="parType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">parType: Complex Type for Parallel Files
      The &lt;par&gt; or parallel files element aggregates pointers to files, parts of
files, and/or sequences of files or parts of files that must be played or displayed
simultaneously to manifest a block of digital content represented by an &lt;fptr&gt; element.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="area" type="areaType" minOccurs="0"/>
      <xsd:element name="seq" type="seqType" minOccurs="0"/>
    </xsd:choice>
    <xsd:attribute name="ID" type="xsd:ID" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="seqType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">seqType: Complex Type for Sequences of Files
        The seq element should be used to link a div to a set of content files when those
files should be played/displayed sequentially to deliver content to a user.  Individual
&lt;area&gt; subelements within the seq element provide the links to the files or portions
thereof.
        </xsd:documentation>
```

```
    </xsd:annotation>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="area" type="areaType" minOccurs="0"/>
      <xsd:element name="par" type="parType" minOccurs="0"/>
    </xsd:choice>
    <xsd:attribute name="ID" type="xsd:ID" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="areaType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">areaType: Complex Type for Area Linking
        The area element provides for more sophisticated linking between a div element and
content files representing that div, be they text, image, audio, or video files.  An area
element can link a div to a point within a file, to a one-dimension segment of a file (e.g.,
text segment, image line, audio/video clip), or a two-dimensional section of a file   (e.g,
subsection of an image, or a subsection of the  video display of a video file.  The area
element has no content; all information is recorded within its various attributes.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:attribute name="ID" type="xsd:ID" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="FILEID" type="xsd:IDREF" use="required">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">FILEID (IDREF/R): An attribute which provides the XML
ID value that identifies the &lt;file&gt; element in the &lt;fileSec&gt; that then points to
and/or contains the digital content represented by the &lt;area&gt; element. It must contain
an ID value represented in an ID attribute associated with a &lt;file&gt; element in the
&lt;fileSec&gt; element in the same METS document.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="SHAPE" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">SHAPE (string/O): An attribute that can be used as in
HTML to define the shape of the relevant area within the content file pointed to by the
&lt;area&gt; element. Typically this would be used with image content (still image or video
frame) when only a portion of an integal image map pertains. If SHAPE is specified then COORDS
must also be present. SHAPE should be used in conjunction with COORDS in the manner defined
for the shape and coords attributes on an HTML4 &lt;area&gt; element. SHAPE must contain one
of the following values:
RECT
CIRCLE
POLY
        </xsd:documentation>
      </xsd:annotation>
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="RECT"/>
          <xsd:enumeration value="CIRCLE"/>
          <xsd:enumeration value="POLY"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="COORDS" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">COORDS (string/O): Specifies the coordinates in an
image map for the shape of the pertinent area as specified in the SHAPE attribute. While
technically optional, SHAPE and COORDS must both appear together to define the relevant area
of image content. COORDS should be used in conjunction with SHAPE in the manner defined for
the COORDs and SHAPE attributes on an HTML4 &lt;area&gt; element. COORDS must be a comma
delimited string of integer value pairs representing coordinates (plus radius in the case of
CIRCLE) within an image map. Number of coordinates pairs depends on shape: RECT: x1, y1, x2,
y2; CIRC: x1, y1; POLY: x1, y1, x2, y2, x3, y3 . . .
```

```
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="BEGIN" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">BEGIN (string/O): An attribute that specifies the
point in the content file where the relevant section of content begins. It can be used in
conjunction with either the END attribute or the EXTENT attribute as a means of defining the
relevant portion of the referenced file precisely. It can only be interpreted meaningfully in
conjunction with the BETYPE or EXTTYPE, which specify the kind of beginning/ending point
values or beginning/extent values that are being used. The BEGIN attribute can be used with or
without a companion END or EXTENT element. In this case, the end of the content file is
assumed to be the end point.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="END" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">END (string/O): An attribute that specifies the point
in the content file where the relevant section of content ends. It can only be interpreted
meaningfully in conjunction with the BETYPE, which specifies the kind of ending point values
being used. Typically the END attribute would only appear in conjunction with a BEGIN element.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="BETYPE" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">BETYPE: Begin/End Type.
          BETYPE (string/O): An attribute that specifies the kind of BEGIN and/or END values
that are being used. For example, if BYTE is specified, then the BEGIN and END point values
represent the byte offsets into a file. If IDREF is specified, then the BEGIN element
specifies the ID value that identifies the element in a structured text file where the
relevant section of the file begins; and the END value (if present) would specify the ID value
that identifies the element with which the relevant section of the file ends. Must be one of
the following values:
BYTE
IDREF
SMIL
MIDI
SMPTE-25
SMPTE-24
SMPTE-DF30
SMPTE-NDF30
SMPTE-DF29.97
SMPTE-NDF29.97
TIME
TCF
XPTR
          </xsd:documentation>
        </xsd:annotation>
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="BYTE"/>
            <xsd:enumeration value="IDREF"/>
            <xsd:enumeration value="SMIL"/>
            <xsd:enumeration value="MIDI"/>
            <xsd:enumeration value="SMPTE-25"/>
            <xsd:enumeration value="SMPTE-24"/>
            <xsd:enumeration value="SMPTE-DF30"/>
            <xsd:enumeration value="SMPTE-NDF30"/>
            <xsd:enumeration value="SMPTE-DF29.97"/>
            <xsd:enumeration value="SMPTE-NDF29.97"/>
            <xsd:enumeration value="TIME"/>
            <xsd:enumeration value="TCF"/>
            <xsd:enumeration value="XPTR"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="EXTENT" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">EXTENT (string/O): An attribute that specifies the
extent of the relevant section of the content file. Can only be interpreted meaningfully in
conjunction with the EXTTYPE which specifies the kind of value that is being used. Typically
the EXTENT attribute would only appear in conjunction with a BEGIN element and would not be
used if the BEGIN point represents an IDREF.
          </xsd:documentation>
        </xsd:annotation>
```

```
      </xsd:attribute>
      <xsd:attribute name="EXTTYPE" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">EXTTYPE (string/O): An attribute that specifies the
kind of EXTENT values that are being used. For example if BYTE is specified then EXTENT would
represent a byte count. If TIME is specified the EXTENT would represent a duration of time.
EXTTYPE must be one of the following values:
BYTE
SMIL
MIDI
SMPTE-25
SMPTE-24
SMPTE-DF30
SMPTE-NDF30
SMPTE-DF29.97
SMPTE-NDF29.97
TIME
TCF.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="BYTE"/>
            <xsd:enumeration value="SMIL"/>
            <xsd:enumeration value="MIDI"/>
            <xsd:enumeration value="SMPTE-25"/>
            <xsd:enumeration value="SMPTE-24"/>
            <xsd:enumeration value="SMPTE-DF30"/>
            <xsd:enumeration value="SMPTE-NDF30"/>
            <xsd:enumeration value="SMPTE-DF29.97"/>
            <xsd:enumeration value="SMPTE-NDF29.97"/>
            <xsd:enumeration value="TIME"/>
            <xsd:enumeration value="TCF"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">ADMID (IDREFS/O): Contains the ID attribute values
identifying the &lt;rightsMD&gt;, &lt;sourceMD&gt;, &lt;techMD&gt; and/or &lt;digiprovMD&gt;
elements within the &lt;amdSec&gt; of the METS document that contain or link to administrative
metadata pertaining to the content represented by the &lt;area&gt; element. Typically the
&lt;area&gt; ADMID attribute would be used to identify the &lt;rightsMD&gt; element or
elements that pertain to the &lt;area&gt;, but it could be used anytime there was a need to
link an &lt;area&gt; with pertinent administrative metadata. For more information on using
METS IDREFS and IDREF type attributes for internal linking, see Chapter 4 of the METS Primer
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="CONTENTIDS" type="URIs" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">CONTENTIDS (URI/O): Content IDs for the content
represented by the &lt;area&gt; (equivalent to DIDL DII or Digital Item Identifier, a unique
external ID).
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="structLinkType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">structLinkType: Complex Type for Structural Map Linking
        The Structural Map Linking section allows for the specification of hyperlinks between
different components of a METS structure delineated in a structural map.  structLink contains
a single, repeatable element, smLink.  Each smLink element indicates a hyperlink between two
nodes in the structMap.  The structMap nodes recorded in smLink are identified using their XML
ID attribute  values.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element name="smLink">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The Structural Map Link element &lt;smLink&gt; identifies a hyperlink between two
nodes in the structural map. You would use &lt;smLink&gt;, for instance, to note the existence
of hypertext links between web pages, if you wished to record those links within METS. NOTE:
&lt;smLink&gt; is an empty element. The location of the &lt;smLink&gt; element to which the
&lt;smLink&gt; element is pointing MUST be stored in the xlink:href attribute.
```

18

```
            </xsd:documentation>
          </xsd:annotation>
          <xsd:complexType>
            <xsd:attribute name="ID" type="xsd:ID" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute ref="xlink:arcrole" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  xlink:arcrole - the role of the link, as per the xlink specification.  See
http://www.w3.org/TR/xlink/
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute ref="xlink:title" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  xlink:title - a title for the link (if needed), as per the xlink
specification.  See http://www.w3.org/TR/xlink/
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute ref="xlink:show" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  xlink:show - see the xlink specification at http://www.w3.org/TR/xlink/
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute ref="xlink:actuate" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  xlink:actuate - see the xlink specification at http://www.w3.org/TR/xlink/
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute ref="xlink:to" use="required">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  xlink:to - the value of the label for the element in the structMap you are
linking to.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute ref="xlink:from" use="required">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  xlink:from - the value of the label for the element in the structMap you are
linking from.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="smLinkGrp">
          <xsd:annotation>
            <xsd:documentation xml:lang="en">
              The structMap link group element &lt;smLinkGrp&gt; provides an implementation of
xlink:extendLink, and provides xlink compliant mechanisms for establishing xlink:arcLink type
links between 2 or more &lt;div&gt; elements in &lt;structMap&gt; element(s) occurring within
the same METS document or different METS documents.  The smLinkGrp could be used as an
alternative to the &lt;smLink&gt; element to establish a one-to-one link between &lt;div&gt;
elements in the same METS document in a fully xlink compliant manner.  However, it can also be
used to establish one-to-many or many-to-many links between &lt;div&gt; elements. For example,
if a METS document contains two &lt;structMap&gt; elements, one of which represents a purely
logical structure and one of which represents a purely physical structure, the
&lt;smLinkGrp&gt; element would provide a means of mapping a &lt;div&gt; representing a
logical entity (for example, a newspaper article) with multiple &lt;div&gt; elements in the
physical &lt;structMap&gt; representing the physical areas that  together comprise the logical
entity (for example, the &lt;div&gt; elements representing the page areas that together
comprise the newspaper article).
```

```
                </xsd:documentation>
            </xsd:annotation>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name="smLocatorLink" minOccurs="2" maxOccurs="unbounded" >
                  <xsd:annotation>
                    <xsd:documentation xml:lang="en">
                        The structMap locator link element &lt;smLocatorLink&gt; is of xlink:type
&quot;locator&quot;.  It provides a means of identifying a &lt;div&gt; element that will
participate in one or more of the links specified by means of &lt;smArcLink&gt; elements
within the same &lt;smLinkGrp&gt;. The participating &lt;div&gt; element that is represented
by the &lt;smLocatorLink&gt; is identified by means of a URI in the associate xlink:href
attribute.  The lowest level of this xlink:href URI value should be a fragment identifier that
references the ID value that identifies the relevant &lt;div&gt; element.  For example,
&quot;xlink:href=&apos;#div20&apos;&quot; where &quot;div20&quot; is the ID value that
identifies the pertinent &lt;div&gt; in the current METS document. Although not required by
the xlink specification, an &lt;smLocatorLink&gt; element will typically include an
xlink:label attribute in this context, as the &lt;smArcLink&gt; elements will reference these
labels to establish the from and to sides of each arc link.
                    </xsd:documentation>
                  </xsd:annotation>
                  <xsd:complexType>
                    <xsd:attribute name="ID" type="xsd:ID">
                      <xsd:annotation>
                        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely
identifies the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS
Primer.</xsd:documentation>
                      </xsd:annotation>
                    </xsd:attribute>
                    <xsd:attributeGroup ref="xlink:locatorLink"/>
                  </xsd:complexType>
                </xsd:element>
                <xsd:element name="smArcLink" minOccurs="1" maxOccurs="unbounded">
                  <xsd:complexType>
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">
                        The structMap arc link element &lt;smArcLink&gt; is of xlink:type
&quot;arc&quot; It can be used to establish a traversal link between two &lt;div&gt; elements
as identified by &lt;smLocatorLink&gt; elements within the same smLinkGrp element. The
associated xlink:from and xlink:to attributes identify the from and to sides of the arc link
by referencing the xlink:label attribute values on the participating smLocatorLink elements.
                      </xsd:documentation>
                    </xsd:annotation>
                    <xsd:attribute name="ID" type="xsd:ID">
                      <xsd:annotation>
                        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely
identifies the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS
Primer.</xsd:documentation>
                      </xsd:annotation>
                    </xsd:attribute>
                    <xsd:attributeGroup ref="xlink:arcLink"/>
                    <xsd:attribute name="ARCTYPE" type="xsd:string">
                      <xsd:annotation>
                        <xsd:documentation xml:lang="en">ARCTYPE (string/O):The ARCTYPE attribute
provides a means of specifying the relationship between the &lt;div&gt; elements participating
in the arc link, and hence the purpose or role of the link.  While it can be considered
analogous to the xlink:arcrole attribute, its type is a simple string, rather than anyURI.
ARCTYPE has no xlink specified meaning, and the xlink:arcrole attribute should be used instead
of or in addition to the ARCTYPE attribute when full xlink compliance is desired with respect
to specifying the role or purpose of the arc link.
                        </xsd:documentation>
                      </xsd:annotation>
                    </xsd:attribute>
                    <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
                      <xsd:annotation>
                        <xsd:documentation xml:lang="en">ADMID (IDREFS/O): Contains the ID
attribute values identifying the &lt;sourceMD&gt;, &lt;techMD&gt;, &lt;digiprovMD&gt; and/or
&lt;rightsMD&gt; elements within the &lt;amdSec&gt; of the METS document that contain or link
to administrative metadata pertaining to &lt;smArcLink&gt;. Typically the &lt;smArcLink&gt;
ADMID attribute would be used to identify one or more &lt;sourceMD&gt; and/or &lt;techMD&gt;
elements that refine or clarify the relationship between the xlink:from and xlink:to sides of
the arc. For more information on using METS IDREFS and IDREF type attributes for internal
linking, see Chapter 4 of the METS Primer.
```

```
                 </xsd:documentation>
               </xsd:annotation>
             </xsd:attribute>
           </xsd:complexType>
         </xsd:element>
       </xsd:sequence>
       <xsd:attribute name="ID" type="xsd:ID"/>
       <xsd:attribute name="ARCLINKORDER" default="unordered">
         <xsd:annotation>
           <xsd:documentation xml:lang="en">ARCLINKORDER (enumerated string/O):
ARCLINKORDER is used to indicate whether the order of the smArcLink elements aggregated by the
smLinkGrp element is significant. If the order is significant, then a value of
&quot;ordered&quot; should be supplied.  Value defaults to &quot;unordered&quot; Note that the
ARLINKORDER attribute has no xlink specified meaning.</xsd:documentation>
         </xsd:annotation>
         <xsd:simpleType>
           <xsd:restriction base="xsd:string">
             <xsd:enumeration value="ordered"/>
             <xsd:enumeration value="unordered"/>
           </xsd:restriction>
         </xsd:simpleType>
       </xsd:attribute>
       <xsd:attributeGroup ref="xlink:extendedLink"/>
     </xsd:complexType>
   </xsd:element>
 </xsd:choice>
 <xsd:attribute name="ID" type="xsd:ID" use="optional">
   <xsd:annotation>
     <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
     </xsd:documentation>
   </xsd:annotation>
 </xsd:attribute>
</xsd:complexType>
<xsd:complexType name="behaviorSecType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">behaviorSecType: Complex Type for Behavior Sections
    Behaviors are executable code which can be associated with parts of a METS object.  The
behaviorSec element is used to group individual behaviors within a hierarchical structure.
Such grouping can be useful to organize families of behaviors together or to indicate other
relationships between particular behaviors.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="behaviorSec" type="behaviorSecType" minOccurs="0"
maxOccurs="unbounded"/>
    <xsd:element name="behavior" type="behaviorType" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          A behavior element &lt;behavior&gt; can be used to associate executable behaviors
with content in the METS document. This element has an interface definition
&lt;interfaceDef&gt; element that represents an abstract definition of a set of behaviors
represented by a particular behavior. A &lt;behavior&gt; element also has a behavior mechanism
&lt;mechanism&gt; element, a module of executable code that implements and runs the behavior
defined abstractly by the interface definition.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="ID" type="xsd:ID" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="CREATED" type="xsd:dateTime" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">CREATED (dateTime/O): Specifies the date and time of
creation for the &lt;behaviorSec&gt;
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
```

```
    <xsd:attribute name="LABEL" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">LABEL (string/O): A text description of the behavior
section.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="behaviorType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">behaviorType: Complex Type for Behaviors
       A behavior can be used to associate executable behaviors with content in the METS
object.  A behavior element has an interface definition element that represents an abstract
definition  of the set  of behaviors represented by a particular behavior.  A behavior element
also has an behavior  mechanism which is a module of executable code that implements and runs
the behavior defined abstractly by the interface definition.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="interfaceDef" type="objectType" minOccurs="0">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The interface definition &lt;interfaceDef&gt; element contains a pointer to an
abstract definition of a single behavior or a set of related behaviors that are associated
with the content of a METS object. The interface definition object to which the
&lt;interfaceDef&gt; element points using xlink:href could be another digital object, or some
other entity, such as a text file which describes the interface or a Web Services Description
Language (WSDL) file. Ideally, an interface definition object contains metadata that describes
a set of behaviors or methods. It may also contain files that describe the intended usage of
the behaviors, and possibly files that represent different expressions of the interface
definition.
      </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="mechanism" type="objectType">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
          A mechanism element &lt;mechanism&gt; contains a pointer to an executable code
module that implements a set of behaviors defined by an interface definition. The
&lt;mechanism&gt; element will be a pointer to another object (a mechanism object). A
mechanism object could be another METS object, or some other entity (e.g., a WSDL file). A
mechanism object should contain executable code, pointers to executable code, or
specifications for binding to network services (e.g., web services).
          </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="ID" type="xsd:ID" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. In the case of a &lt;behavior&gt;
element that applies to a &lt;transformFile&gt; element, the ID value must be present and
would be referenced from the transformFile/@TRANSFORMBEHAVIOR attribute. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
      </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="STRUCTID" type="xsd:IDREFS" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">STRUCTID (IDREFS/O): An XML IDREFS attribute used to
link a &lt;behavior&gt;  to one or more &lt;div&gt; elements within a &lt;structMap&gt; in the
METS document. The content to which the STRUCTID points is considered input to the executable
behavior mechanism defined for the behavior.  If the &lt;behavior&gt; applies to one or more
&lt;div&gt; elements, then the STRUCTID attribute must be present.
      </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="BTYPE" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">BTYPE (string/O): The behavior type provides a means
of categorizing the related behavior.</xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="CREATED" type="xsd:dateTime" use="optional">
      <xsd:annotation>
```

```
        <xsd:documentation xml:lang="en">CREATED (dateTime/O): The dateTime of creation for
the behavior.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="LABEL" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">LABEL (string/O): A text description of the behavior.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="GROUPID" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">GROUPID (string/O): An identifier that establishes a
correspondence between the given behavior and other behaviors, typically used to facilitate
versions of behaviors.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ADMID (IDREFS/O): An optional attribute listing the
XML ID values of administrative metadata sections within the METS document pertaining to this
behavior.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="objectType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">objectType: complexType for interfaceDef and mechanism
elements
        The mechanism and behavior elements point to external objects--an interface definition
object or an executable code object respectively--which together constitute a behavior that
can be applied to one or more &lt;div&gt; elements in a &lt;structMap&gt;.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:attribute name="ID" type="xsd:ID" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. For more information on using ID
attributes for internal and external linking see Chapter 4 of the METS Primer.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="LABEL" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">LABEL (string/O): A text description of the entity
represented.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attributeGroup ref="LOCATION"/>
    <xsd:attributeGroup ref="xlink:simpleLink"/>
  </xsd:complexType>
  <xsd:complexType name="mdSecType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">mdSecType: Complex Type for Metadata Sections
    A generic framework for pointing to/including metadata within a METS document, a la
Warwick Framework.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:all>
      <xsd:element name="mdRef" minOccurs="0">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The metadata reference element &lt;mdRef&gt; element is a generic element used
throughout the METS schema to provide a pointer to metadata which resides outside the METS
document.  NB: &lt;mdRef&gt; is an empty element.  The location of the metadata must be
recorded in the xlink:href attribute, supplemented by the XPTR attribute as needed.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:attribute name="ID" type="xsd:ID" use="optional">
            <xsd:annotation>
```

```
              <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
          <xsd:attributeGroup ref="LOCATION"/>
          <xsd:attributeGroup ref="xlink:simpleLink"/>
          <xsd:attributeGroup ref="METADATA"/>
          <xsd:attributeGroup ref="FILECORE"/>
          <xsd:attribute name="LABEL" type="xsd:string" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">LABEL (string/O): Provides a label to display
to the viewer of the METS document that identifies the associated metadata.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
          <xsd:attribute name="XPTR" type="xsd:string" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">XPTR (string/O): Locates the point within a
file to which the &lt;mdRef&gt; element refers, if applicable.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="mdWrap" minOccurs="0">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
          A metadata wrapper element &lt;mdWrap&gt; provides a wrapper around metadata
embedded within a METS document. The element is repeatable. Such metadata can be in one of two
forms: 1) XML-encoded metadata, with the XML-encoding identifying itself as belonging to a
namespace other than the METS document namespace. 2) Any arbitrary binary or textual form,
PROVIDED that the metadata is Base64 encoded and wrapped in a &lt;binData&gt; element within
the internal descriptive metadata element.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:choice>
            <xsd:element name="binData" type="xsd:base64Binary" minOccurs="0">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                The binary data wrapper element &lt;binData&gt; is used to contain Base64
encoded metadata.                          </xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="xmlData" minOccurs="0">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                The xml data wrapper element &lt;xmlData&gt; is used to contain XML encoded
metadata. The content of an &lt;xmlData&gt; element can be in any namespace or in no
namespace. As permitted by the XML Schema Standard, the processContents attribute value for
the metadata in an &lt;xmlData&gt; is set to "lax". Therefore, if the source schema and its
location are identified by means of an XML schemaLocation attribute, then an XML processor
will validate the elements for which it can find declarations. If a source schema is not
identified, or cannot be found at the specified schemaLocation, then an XML validator will
check for well-formedness, but otherwise skip over the elements appearing in the
&lt;xmlData&gt; element.
                </xsd:documentation>
              </xsd:annotation>
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:any namespace="##any" maxOccurs="unbounded" processContents="lax"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:choice>
          <xsd:attribute name="ID" type="xsd:ID" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
```

```
      <xsd:attributeGroup ref="METADATA"/>
      <xsd:attributeGroup ref="FILECORE"/>
      <xsd:attribute name="LABEL" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">LABEL: an optional string attribute providing a
label to display to the viewer of the METS document identifying the metadata.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>
</xsd:all>
<xsd:attribute name="ID" type="xsd:ID" use="required">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">ID (ID/R): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. The ID attribute on the
&lt;dmdSec&gt;, &lt;techMD&gt;, &lt;sourceMD&gt;, &lt;rightsMD&gt; and &lt;digiprovMD&gt;
elements (which are all of mdSecType) is required, and its value should be referenced from one
or more DMDID attributes (when the ID identifies a &lt;dmdSec&gt; element) or ADMID attributes
(when the ID identifies a &lt;techMD&gt;, &lt;sourceMD&gt;, &lt;rightsMD&gt; or
&lt;digiprovMD&gt; element) that are associated with other elements in the METS document. The
following elements support references to a &lt;dmdSec&gt; via a DMDID attribute: &lt;file&gt;,
&lt;stream&gt;, &lt;div&gt;.  The following elements support references to &lt;techMD&gt;,
&lt;sourceMD&gt;, &lt;rightsMD&gt; and &lt;digiprovMD&gt; elements via an ADMID attribute:
&lt;metsHdr&gt;, &lt;dmdSec&gt;, &lt;techMD&gt;, &lt;sourceMD&gt;, &lt;rightsMD&gt;,
&lt;digiprovMD&gt;, &lt;fileGrp&gt;, &lt;file&gt;, &lt;stream&gt;, &lt;div&gt;, &lt;area&gt;,
&lt;behavior&gt;. For more information on using ID attributes for internal and external
linking see Chapter 4 of the METS Primer.
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="GROUPID" type="xsd:string" use="optional">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">GROUPID (string/O): This identifier is used to
indicate that different metadata sections may be considered as part of a group. Two metadata
sections with the same GROUPID value are to be considered part of the same group. For example
this facility might be used to group changed versions of the same metadata if previous
versions are maintained in a file for tracking purposes.
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">ADMID (IDREFS/O): Contains the ID attribute values of
the &lt;digiprovMD&gt;, &lt;techMD&gt;, &lt;sourceMD&gt; and/or &lt;rightsMD&gt; elements
within the &lt;amdSec&gt; of the METS document that contain administrative metadata pertaining
to the current mdSecType element. Typically used in this context to reference preservation
metadata (digiprovMD) which applies to the current metadata. For more information on using
METS IDREFS and IDREF type attributes for internal linking, see Chapter 4 of the METS Primer.
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="CREATED" type="xsd:dateTime" use="optional">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">CREATED (dateTime/O): Specifies the date and time of
creation for the metadata.
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="STATUS" type="xsd:string" use="optional">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">STATUS (string/O): Indicates the status of this
metadata (e.g., superseded, current, etc.).
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
<xsd:complexType name="fileType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">fileType: Complex Type for Files
      The file element provides access to content files for a METS object.  A file element
may contain one or more FLocat elements, which provide pointers to a content file, and/or an
FContent element, which wraps an encoded version of the file. Note that ALL FLocat and
FContent elements underneath a single file element should identify/contain identical copies of
a single file.
    </xsd:documentation>
```

```
      </xsd:annotation>

    <xsd:sequence>
      <xsd:element name="FLocat" minOccurs="0" maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The file location element &lt;FLocat&gt; provides a pointer to the location of a
content file. It uses the XLink reference syntax to provide linking information indicating the
actual location of the content file, along with other attributes specifying additional linking
information. NOTE: &lt;FLocat&gt; is an empty element. The location of the resource pointed to
MUST be stored in the xlink:href attribute.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:attribute name="ID" type="xsd:ID" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
          <xsd:attributeGroup ref="LOCATION"/>
          <xsd:attribute name="USE" type="xsd:string" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">USE (string/O): A tagging attribute to indicate
the intended use of the specific copy of the file  represented by the &lt;FLocat&gt; element
(e.g., service master, archive master). A USE attribute can be expressed at the&lt;fileGrp&gt;
level, the &lt;file&gt; level, the &lt;FLocat&gt; level and/or the &lt;FContent&gt; level.  A
USE attribute value at the &lt;fileGrp&gt; level should pertain to all of the files in the
&lt;fileGrp&gt;.  A USE attribute at the &lt;file&gt; level should pertain to all copies of
the file as represented by subsidiary &lt;FLocat&gt; and/or &lt;FContent&gt; elements.  A USE
attribute at the &lt;FLocat&gt; or &lt;FContent&gt; level pertains to the particular copy of
the file that is either referenced (&lt;FLocat&gt;) or wrapped (&lt;FContent&gt;).
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
          <xsd:attributeGroup ref="xlink:simpleLink"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="FContent" minOccurs="0">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            The file content element &lt;FContent&gt; is used to identify a content file
contained internally within a METS document. The content file must be either Base64 encoded
and contained within the subsidiary &lt;binData&gt; wrapper element, or consist of XML
information and be contained within the subsidiary &lt;xmlData&gt; wrapper element.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:choice>
            <xsd:element name="binData" type="xsd:base64Binary" minOccurs="0">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  A binary data wrapper element &lt;binData&gt; is used to contain a Base64
encoded file.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="xmlData" minOccurs="0">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">
                  An xml data wrapper element &lt;xmlData&gt; is used to contain  an XML
encoded file. The content of an &lt;xmlData&gt; element can be in any namespace or in no
namespace. As permitted by the XML Schema Standard, the processContents attribute value for
the metadata in an &lt;xmlData&gt; element is set to "lax". Therefore, if the source schema
and its location are identified by means of an xsi:schemaLocation attribute, then an XML
processor will validate the elements for which it can find declarations. If a source schema is
not identified, or cannot be found at the specified schemaLocation, then an XML validator will
check for well-formedness, but otherwise skip over the elements appearing in the
&lt;xmlData&gt; element.
                </xsd:documentation>
              </xsd:annotation>
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:any namespace="##any" maxOccurs="unbounded" processContents="lax"/>
```

```
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:choice>
          <xsd:attribute name="ID" type="xsd:ID" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely identifies
the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
          <xsd:attribute name="USE" type="xsd:string" use="optional">
            <xsd:annotation>
              <xsd:documentation xml:lang="en">USE (string/O): A tagging attribute to indicate
the intended use of the specific copy of the file represented by the &lt;FContent&gt; element
(e.g., service master, archive master). A USE attribute can be expressed at the&lt;fileGrp&gt;
level, the &lt;file&gt; level, the &lt;FLocat&gt; level and/or the &lt;FContent&gt; level.  A
USE attribute value at the &lt;fileGrp&gt; level should pertain to all of the files in the
&lt;fileGrp&gt;.  A USE attribute at the &lt;file&gt; level should pertain to all copies of
the file as represented by subsidiary &lt;FLocat&gt; and/or &lt;FContent&gt; elements.  A USE
attribute at the &lt;FLocat&gt; or &lt;FContent&gt; level pertains to the particular copy of
the file that is either referenced (&lt;FLocat&gt;) or wrapped (&lt;FContent&gt;).
              </xsd:documentation>
            </xsd:annotation>
          </xsd:attribute>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="stream" minOccurs="0" maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            A component byte stream element &lt;stream&gt; may be composed of one or more
subsidiary streams. An MPEG4 file, for example, might contain separate audio and video
streams, each of which is associated with technical metadata. The repeatable &lt;stream&gt;
element provides a mechanism to record the existence of separate data streams within a
particular file, and the opportunity to associate &lt;dmdSec&gt; and &lt;amdSec&gt; with those
subsidiary data streams if desired. </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:restriction base="xsd:anyType">
              <xsd:attribute name="ID" type="xsd:ID" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely
identifies the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="streamType" type="xsd:string" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">streamType (string/O): The IANA MIME media
type for the bytestream.</xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="OWNERID" type="xsd:string" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">OWNERID (string/O): Used to provide a
unique identifier (which could include a URI) assigned to the file. This identifier may differ
from the URI used to retrieve the file.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
                <xsd:annotation>
                  <xsd:documentation xml:lang="en">ADMID (IDREFS/O): Contains the ID attribute
values of the &lt;techMD&gt;, &lt;sourceMD&gt;, &lt;rightsMD&gt; and/or &lt;digiprovMD&gt;
elements within the &lt;amdSec&gt; of the METS document that contain administrative metadata
pertaining to the bytestream. For more information on using METS IDREFS and IDREF type
attributes for internal linking, see Chapter 4 of the METS Primer.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:attribute>
              <xsd:attribute name="DMDID" type="xsd:IDREFS" use="optional">
                <xsd:annotation>
```

```
                <xsd:documentation xml:lang="en">DMDID (IDREFS/O): Contains the ID attribute
values identifying the &lt;dmdSec&gt;, elements in the METS document that contain or link to
descriptive metadata pertaining to the content file stream represented by the current
&lt;stream&gt; element.  For more information on using METS IDREFS and IDREF type attributes
for internal linking, see Chapter 4 of the METS Primer.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute name="BEGIN" type="xsd:string" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">BEGIN (string/O): An attribute that
specifies the point in the parent &lt;file&gt; where the current &lt;stream&gt; begins. It can
be used in conjunction with the END attribute as a means of defining the location of the
stream within its parent file. However, the BEGIN attribute can be used with or without a
companion END attribute. When no END attribute is specified, the end of the parent file is
assumed also to be the end point of the stream. The BEGIN and END attributes can only be
interpreted meaningfully in conjunction with a BETYPE attribute, which specifies the kind of
beginning/ending point values that are being used.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute name="END" type="xsd:string" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">END (string/O): An attribute that specifies
the point in the parent &lt;file&gt; where the &lt;stream&gt; ends. It can only be interpreted
meaningfully in conjunction with the BETYPE, which specifies the kind of ending point values
being used. Typically the END attribute would only appear in conjunction with a BEGIN
attribute.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute name="BETYPE" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">BETYPE: Begin/End Type.
                  BETYPE (string/O): An attribute that specifies the kind of BEGIN and/or
END values that are being used. Currently BYTE is the only valid value that can be used in
conjunction with nested &lt;file&gt; or &lt;stream&gt; elements.
                </xsd:documentation>
              </xsd:annotation>
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="BYTE"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:attribute>
          </xsd:restriction>
        </xsd:complexContent>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="transformFile" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">
          The transform file element &lt;transformFile&gt; provides a means to access any
subsidiary files listed below a &lt;file&gt; element by indicating the steps required to
"unpack" or transform the subsidiary files. This element is repeatable and might provide a
link to a &lt;behavior&gt; in the &lt;behaviorSec&gt; that performs the
transformation.</xsd:documentation>
      </xsd:annotation>
      <xsd:complexType>
        <xsd:complexContent>
          <xsd:restriction base="xsd:anyType">
            <xsd:attribute name="ID" type="xsd:ID" use="optional">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">ID (ID/O): This attribute uniquely
identifies the element within the METS document, and would allow the element to be referenced
unambiguously from another element or document via an IDREF or an XPTR. For more information
on using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute name="TRANSFORMTYPE" use="required">
              <xsd:annotation>
                <xsd:documentation xml:lang="en">TRANSFORMTYPE (string/R): Is used to
indicate the type of transformation needed to render content of a file accessible. This may
include unpacking a file into subsidiary files/streams. The controlled value constraints for
this XML string include "decompression" and "decryption". Decompression is defined as the
action of reversing data compression, i.e., the process of encoding information using fewer
```

bits than an unencoded representation would use by means of specific encoding schemas.
Decryption is defined as the process of restoring data that has been obscured to make it
unreadable without special knowledge (encrypted data) to its original form.
</xsd:documentation>
                    </xsd:annotation>
                    <xsd:simpleType>
                      <xsd:restriction base="xsd:string">
                        <xsd:enumeration value="decompression"></xsd:enumeration>
                        <xsd:enumeration value="decryption"></xsd:enumeration>
                      </xsd:restriction>
                    </xsd:simpleType>
                  </xsd:attribute>
                  <xsd:attribute name="TRANSFORMALGORITHM" type="xsd:string" use="required">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">TRANSFORM-ALGORITHM (string/R): Specifies
the decompression or decryption routine used to access the contents of the file. Algorithms
for compression can be either loss-less or lossy.</xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
                  <xsd:attribute name="TRANSFORMKEY" type="xsd:string" use="optional">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">TRANSFORMKEY (string/O): A key to be used
with the transform algorithm for accessing the file's contents.</xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
                  <xsd:attribute name="TRANSFORMBEHAVIOR" type="xsd:IDREF" use="optional">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">TRANSFORMBEHAVIOR (string/O): An IDREF to a
behavior element for this transformation.</xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
                  <xsd:attribute name="TRANSFORMORDER" type="xsd:positiveInteger" use="required">
                    <xsd:annotation>
                      <xsd:documentation xml:lang="en">TRANSFORMORDER (postive-integer/R): The
order in which the instructions must be followed in order to unpack or transform the container
file.</xsd:documentation>
                    </xsd:annotation>
                  </xsd:attribute>
              </xsd:restriction>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="file" type="fileType" minOccurs="0"
maxOccurs="unbounded"></xsd:element>
    </xsd:sequence>
    <xsd:attribute name="ID" type="xsd:ID" use="required">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">ID (ID/R): This attribute uniquely identifies the
element within the METS document, and would allow the element to be referenced unambiguously
from another element or document via an IDREF or an XPTR. Typically, the ID attribute value on
a <file> element would be referenced from one or more FILEID attributes (which are of
type IDREF) on <fptr>and/or <area> elements within the <structMap>. Such
references establish links between structural divisions (<div> elements) and the
specific content files or parts of content files that manifest them. For more information on
using ID attributes for internal and external linking see Chapter 4 of the METS Primer.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="SEQ" type="xsd:int" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">SEQ (integer/O): Indicates the sequence of this
<file> relative to the others in its <fileGrp>.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attributeGroup ref="FILECORE"></xsd:attributeGroup>
  <xsd:attribute name="OWNERID" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">OWNERID (string/O): A unique identifier assigned to
the file by its owner. This may be a URI which differs from the URI used to retrieve the
file.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="ADMID" type="xsd:IDREFS" use="optional">
    <xsd:annotation>

```
      <xsd:documentation xml:lang="en">ADMID (IDREFS/O): Contains the ID attribute values of
the &lt;techMD&gt;, &lt;sourceMD&gt;, &lt;rightsMD&gt; and/or &lt;digiprovMD&gt; elements
within the &lt;amdSec&gt; of the METS document that contain administrative metadata pertaining
to the file. For more information on using METS IDREFS and IDREF type attributes for internal
linking, see Chapter 4 of the METS Primer.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="DMDID" type="xsd:IDREFS" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">DMDID (IDREFS/O): Contains the ID attribute values
identifying the &lt;dmdSec&gt;, elements in the METS document that contain or link to
descriptive metadata pertaining to the content file represented by the current &lt;file&gt;
element.  For more information on using METS IDREFS and IDREF type attributes for internal
linking, see Chapter 4 of the METS Primer.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="GROUPID" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">GROUPID (string/O): An identifier that establishes a
correspondence between this file and files in other file groups. Typically, this will be used
to associate a master file in one file group with the derivative files made from it in other
file groups.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="USE" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">USE (string/O): A tagging attribute to indicate the
intended use of all copies of the file aggregated by the &lt;file&gt; element (e.g., master,
reference, thumbnails for image files). A USE attribute can be expressed at the&lt;fileGrp&gt;
level, the &lt;file&gt; level, the &lt;FLocat&gt; level and/or the &lt;FContent&gt; level.  A
USE attribute value at the &lt;fileGrp&gt; level should pertain to all of the files in the
&lt;fileGrp&gt;.  A USE attribute at the &lt;file&gt; level should pertain to all copies of
the file as represented by subsidiary &lt;FLocat&gt; and/or &lt;FContent&gt; elements.  A USE
attribute at the &lt;FLocat&gt; or &lt;FContent&gt; level pertains to the particular copy of
the file that is either referenced (&lt;FLocat&gt;) or wrapped (&lt;FContent&gt;).
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="BEGIN" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">BEGIN (string/O): An attribute that specifies the
point in the parent &lt;file&gt; where the current &lt;file&gt; begins.  When used in
conjunction with a &lt;file&gt; element, this attribute is only meaningful when this element
is nested, and its parent &lt;file&gt; element represents a container file. It can be used in
conjunction with the END attribute as a means of defining the location of the current file
within its parent file. However, the BEGIN attribute can be used with or without a companion
END attribute. When no END attribute is specified, the end of the parent file is assumed also
to be the end point of the current file. The BEGIN and END attributes can only be interpreted
meaningfully in conjunction with a BETYPE attribute, which specifies the kind of
beginning/ending point values that are being used.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="END" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">END (string/O): An attribute that specifies the point
in the parent &lt;file&gt; where the current, nested &lt;file&gt; ends. It can only be
interpreted meaningfully in conjunction with the BETYPE, which specifies the kind of ending
point values being used. Typically the END attribute would only appear in conjunction with a
BEGIN attribute.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="BETYPE" use="optional">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">BETYPE: Begin/End Type.
        BETYPE (string/O): An attribute that specifies the kind of BEGIN and/or END values
that are being used. Currently BYTE is the only valid value that can be used in conjunction
with nested &lt;file&gt; or &lt;stream&gt; elements.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="BYTE"/>
```

```
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
  </xsd:complexType>

  <xsd:simpleType name="URIs">
      <xsd:list itemType="xsd:anyURI"/>
  </xsd:simpleType>

  <xsd:attributeGroup name="METADATA">
    <xsd:attribute name="MDTYPE" use="required">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">MDTYPE (string/R): Is used to indicate the type of
the associated metadata. It must have one of the following values:
MARC: any form of MARC record
MODS: metadata in the Library of Congress MODS format
EAD: Encoded Archival Description finding aid
DC: Dublin Core
NISOIMG: NISO Technical Metadata for Digital Still Images
LC-AV: technical metadata specified in the Library of Congress A/V prototyping project
VRA: Visual Resources Association Core
TEIHDR: Text Encoding Initiative Header
DDI: Data Documentation Initiative
FGDC: Federal Geographic Data Committee metadata
LOM: Learning Object Model
PREMIS:  PREservation Metadata: Implementation Strategies
PREMIS:OBJECT: PREMIS Object entiry
PREMIS:AGENT: PREMIS Agent entity
PREMIS:RIGHTS: PREMIS Rights entity
PREMIS:EVENT: PREMIS Event entity
TEXTMD: textMD Technical metadata for text
METSRIGHTS: Rights Declaration Schema
ISO 19115:2003 NAP: North American Profile of ISO 19115:2003 descriptive metadata
EAC-CPF: Encoded Archival Context - Corporate Bodies, Persons, and Families
OTHER: metadata in a format not specified above
        </xsd:documentation>
      </xsd:annotation>
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="MARC"/>
          <xsd:enumeration value="MODS"/>
          <xsd:enumeration value="EAD"/>
          <xsd:enumeration value="DC"/>
          <xsd:enumeration value="NISOIMG"/>
          <xsd:enumeration value="LC-AV"/>
          <xsd:enumeration value="VRA"/>
          <xsd:enumeration value="TEIHDR"/>
          <xsd:enumeration value="DDI"/>
          <xsd:enumeration value="FGDC"/>
            <xsd:enumeration value="LOM"/>
          <xsd:enumeration value="PREMIS"/>
          <xsd:enumeration value="PREMIS:OBJECT"/>
          <xsd:enumeration value="PREMIS:AGENT"/>
          <xsd:enumeration value="PREMIS:RIGHTS"/>
          <xsd:enumeration value="PREMIS:EVENT"/>
          <xsd:enumeration value="TEXTMD"/>
          <xsd:enumeration value="METSRIGHTS"/>
          <xsd:enumeration value="ISO 19115:2003 NAP"/>
          <xsd:enumeration value="EAC-CPF"></xsd:enumeration>
          <xsd:enumeration value="OTHER"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="OTHERMDTYPE" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">OTHERMDTYPE (string/O): Specifies the form of
metadata in use when the value OTHER is indicated in the MDTYPE attribute.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="MDTYPEVERSION" type="xsd:string" use="optional">
      <xsd:annotation>
        <xsd:documentation xml:lang="en">MDTYPEVERSION(string/O): Provides a means for
recording the version of the type of metadata (as recorded in the MDTYPE or OTHERMDTYPE
attribute) that is being used.  This may represent the version of the underlying data
dictionary or metadata model rather than a schema version. </xsd:documentation>
      </xsd:annotation>
```

```
      </xsd:attribute>
    </xsd:attributeGroup>
    <xsd:attributeGroup name="LOCATION">
      <xsd:attribute name="LOCTYPE" use="required">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">LOCTYPE (string/R): Specifies the locator type used
in the xlink:href attribute. Valid values for LOCTYPE are:
            ARK
            URN
            URL
            PURL
            HANDLE
            DOI
            OTHER
          </xsd:documentation>
        </xsd:annotation>
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="ARK"/>
            <xsd:enumeration value="URN"/>
            <xsd:enumeration value="URL"/>
            <xsd:enumeration value="PURL"/>
            <xsd:enumeration value="HANDLE"/>
            <xsd:enumeration value="DOI"/>
            <xsd:enumeration value="OTHER"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="OTHERLOCTYPE" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">OTHERLOCTYPE (string/O): Specifies the locator type
when the value OTHER is used in the LOCTYPE attribute. Although optional, it is strongly
recommended when OTHER is used.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:attributeGroup>
    <xsd:attributeGroup name="FILECORE">
      <xsd:attribute name="MIMETYPE" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">MIMETYPE (string/O): The IANA MIME media type for the
associated file or wrapped content. Some values for this attribute can be found on the IANA
website.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="SIZE" type="xsd:long" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">SIZE (long/O): Specifies the size in bytes of the
associated file or wrapped content.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="CREATED" type="xsd:dateTime" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">CREATED (dateTime/O): Specifies the date and time of
creation for the associated file or wrapped content.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="CHECKSUM" type="xsd:string" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">CHECKSUM (string/O): Provides a checksum value for
the associated file or wrapped content.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
      <xsd:attribute name="CHECKSUMTYPE" use="optional">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">CHECKSUMTYPE (enumerated string/O): Specifies the
checksum algorithm used to produce the value contained in the CHECKSUM attribute.
CHECKSUMTYPE must contain one of the following values:
            Adler-32
            CRC32
            HAVAL
            MD5
            MNP
```

```
              SHA-1
              SHA-256
              SHA-384
              SHA-512
              TIGER
              WHIRLPOOL
          </xsd:documentation>
        </xsd:annotation>
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Adler-32"/>
            <xsd:enumeration value="CRC32"/>
            <xsd:enumeration value="HAVAL"/>
            <xsd:enumeration value="MD5"/>
            <xsd:enumeration value="MNP"/>
            <xsd:enumeration value="SHA-1"/>
            <xsd:enumeration value="SHA-256"/>
            <xsd:enumeration value="SHA-384"/>
            <xsd:enumeration value="SHA-512"/>
            <xsd:enumeration value="TIGER"/>
            <xsd:enumeration value="WHIRLPOOL"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
    </xsd:attributeGroup>
</xsd:schema>
```

Popis prvků schématu XML za účelem vytvoření datového balíčku SIP

Kořenový element METS

Element definuje globální atributy.

Element	<mets:mets>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE
Atributy	**xsi:schemaLocation** zaznamenává dvojici údajů, která spojuje adresu (URI) deklarovaného jmenného prostoru s umístěním příslušného schématu XML. Výchozí hodnota atributu je "http://www.loc.gov/METS/ http://www.loc.gov/standards/mets/mets.xsd http://www.mvcr.cz/nsesss/v2 http://www.mvcr.cz/nsesss/v2/nsesss.xsd".
	OBJID identifikuje balíček SIP. Jde o jedinečný identifikátor balíčku v rámci původce. Jeho konstrukce není předepsána.
	LABEL uvádí popis použití dokumentu XML. Výchozí hodnota atributu je "Datový balíček pro předávání dokumentů a jejich metadat do archivu – Submission Information Package (SIP)".
	xmlns:xsi zaznamenává adresu (URI) jmenného prostoru schématu XML. Tento jmenný prostor je identifikován prostřednictvím URL. Výchozí hodnota je "http://www.w3.org/2001/XMLSchema-instance".
	xmlns:nsesss zaznamenává adresu (URI) jmenného prostoru schématu NSESSS verze 2.0. Výchozí hodnota je "http://www.mvcr.cz/nsesss/v2".
	xmlns:mets zaznamenává adresu (URI) jmenného prostoru schématu METS. Výchozí hodnota je "http://www.loc.gov/METS/".

Záhlaví METS

Element definuje subjekty, pro které je dokument XML určen, a informace o vytvoření a pozdějších úpravách dokumentu.

Element	<mets:metsHdr>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE
Atributy	**LASTMODDATE** zaznamenává datum poslední úpravy dokumentu XML ve formě, která je dána normou ISO 8601.
	CREATEDATE zaznamenává datum vytvoření dokumentu XML ve formě, která je dána normou ISO 8601.

Subjekt

Element zaznamenává subjekt, který dokument XML vytvořil nebo kterému je adresován. Uveden je původce i fyzická osoba odpovědná za tvorbu balíčku SIP stejně jako digitální archiv a fyzická osoba odpovědná za příjem balíčku.

Element	**\<mets:agent>**
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	ANO
Atributy	**TYPE** definuje, o jaký typ subjektu jde. Povolené hodnoty jsou "ORGANIZATION" (korporace) a "INDIVIDUAL" (fyzická osoba).
	ROLE definuje, jakou roli příslušný subjekt plní. Povolené hodnoty jsou "CREATOR" (původce) a "ARCHIVES" (archiv).
	ID identifikuje subjekt. Jde o jedinečný identifikátor subjektu. Jeho konstrukce není předepsána.

Název/jméno subjektu

Element definuje název nebo jméno subjektu.

Element	**\<mets:name>**
Typ	jednoduchý datový typ
Povinnost	ANO
Opakovatelnost	NE

Poznámka subjektu

Element definuje další libovolnou charakteristiku subjektu.

Element	**\<mets:note>**
Typ	jednoduchý datový typ
Povinnost	NE
Opakovatelnost	ANO

Sekce popisných metadat

Element definuje část dokumentu XML, která je určena pro vkládání popisných metadat. Ta jsou definována schématem NSESSS.

Element	<mets:dmdSec>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE
Atributy	**ID** identifikuje část dokumentu XML. Jde o jedinečný identifikátor části v celém dokumentu. Jeho konstrukce není předepsána.

Vložená metadata

Element zaznamenává vložená popisná metadata. Ta jsou definována schématem NSESSS.

Element	<mets:mdWrap>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE
Atributy	**LABEL** uvádí popis schématu metadat. Výchozí hodnota atributu je "Schéma XML pro předávání dokumentů a jejich metadat do archivu".
	xmlns:nsesss zaznamenává adresu (URI) jmenného prostoru schématu NSESSS verze 2.0. Výchozí hodnota je "http://www.mvcr.cz/nsesss/v2".
	MDTYPEVERSION zaznamenává verzi schématu NSESSS. Výchozí hodnota je "2.0".
	OTHERMDTYPE zaznamenává název schématu XML. Výchozí hodnota je "NSESSS".
	MDTYPE zaznamenává název schématu XML z číselníku známých schémat. Výchozí hodnota je "OTHER".
	MIMETYPE zaznamenává určení typu a souborového formátu metadat podle internetového standardu MIME. Výchozí hodnota je "text/xml".

Data XML

Element obsahuje vložená popisná metadata. Ta jsou definována schématem NSESSS. Do elementu jsou vkládány kořenové elementy schématu NSESSS s prefixem nsesss. V případě, že je základní entita NSESSS (tj. díl, dokument nebo spis) nebo její podřízené či nadřízené entity spojeny s jinou entitou prostřednictvím křížového odkazu, je tato entita rovněž vložena do elementu jako základní.

Element	<mets:xmlData>
Typ	smíšený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE

Sekce souborů (komponent)

Element definuje část dokumentu XML, která je určena pro soubory (komponenty).

Element	<mets:fileSec>
Typ	složený datový typ (kontejner)
Povinnost	NE
Opakovatelnost	NE

Skupina souborů (komponent)

Element zaznamenává soubory (komponenty).

Element	<mets:fileGrp>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE

Soubor (komponenta)

Element zaznamenává jednotlivý soubor (komponentu).

Element	<mets:file>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	ANO
Atributy	**ID** identifikuje komponentu. Jde o jedinečný identifikátor komponenty v rámci dokumentu XML. Jeho konstrukce není předepsána.
	MIMETYPE zaznamenává určení typu a souborového formátu metadat podle internetového standardu MIME.
	OWNERID identifikuje vlastníka komponenty. Jde o jedinečný identifikátor v rámci dokumentu XML. Jeho konstrukce není předepsána.
	CHECKSUMTYPE zaznamenává šifrovací algoritmus pro tvorbu otisku (hash) komponenty.
	CHECKSUM zaznamenává otisk (hash) komponenty.
	SIZE zaznamenává velikost komponenty v bytech.
	CREATED zaznamenává datum vytvoření komponenty ve formě, která je dána normou ISO 8601.

Obsah souboru (komponenty)

Element zaznamenává obsah souboru (komponenty).

Element	<mets:FContent>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE

Binární data souboru (komponenty)

Element obsahuje binární obsah souboru (komponenty) v kódování Base64.

Element	<mets:binData>
Typ	smíšený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE

Strukturální mapa

Element definuje strukturu objektů a entit podle schématu NSESSS v hierarchické struktuře od nejvýše umístěného spisového plánu až po nejníže umístěný dokument.

Element	<mets:structMap>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	NE

Objekt/entita

Element zaznamenává jednotlivý objekt nebo entitu podle schématu NSESSS.

Element	<mets:div>
Typ	složený datový typ (kontejner)
Povinnost	ANO
Opakovatelnost	ANO
Atributy	**TYPE** zaznamenává typu objektu nebo entity. Nejvyšší entitou je "spisový plán", dále "věcná skupina", "typový spis", "součást", "díl", "spis" a nejnižší entitou je "dokument". **DMID** identifikuje objekt nebo entitu. Jde o jedinečný identifikátor v rámci dokumentu XML. Jeho konstrukce není předepsána.

Odkaz na soubor (komponentu)

Element zaznamenává odkaz na soubor (komponentu) přiřazenou k příslušnému dokumentu. Uvádí se jen u dokumentů, které obsahují komponenty.

Element	**<mets:fptr>**
Typ	složený datový typ (kontejner)
Povinnost	NE
Opakovatelnost	ANO
Atributy	**FILEID** identifikuje komponentu. Jde o jedinečný identifikátor v rámci dokumentu XML. Jeho konstrukce není předepsána.

Poznámka

XML dokument reprezentující datový balíček SIP je zpravidla opatřen uznávaným elektronickým podpisem nebo uznávanou elektronickou značkou. Datový balíček může být rovněž opatřen kvalifikovaným časovým razítkem. Užití uvedených autentizačních prvků si pro předání dokumentů a jejich metadat zpravidla sjedná předávající organizace a příslušný archiv.

Pokud organizace užije uznávaný elektronický podpis, uznávanou elektronickou značku nebo kvalifikované časové razítko, jsou tyto autentizační prvky zpracovány v XML podobě a rovněž jsou vloženy do datového balíčku. Specifikace autentizačních prvků v XML podobě je stanovena standardem *XML Signature Syntax and Processing (Second Edition)*, který je uveden na URL <http://www.w3.org/TR/xmldsig-core>, nebo standardem *XML Advanced Electronic Signatures (XAdES)*, který je uveden na URL <http://www.w3.org/TR/XAdES>.