# METS Data Model

The following data model corresponds roughly to the current METS XML Schema. In the following sections a description will be provided for each of the primary entities. In some cases how the entities are related to elements in the current XML schema will be described.



Figure 1 UML Class Diagram

## InformationPackage

The InformationPackage represents a complete METS document. An InformationPackage must contain one or more StructuralDivisions and may contain zero or more InformationEntities. An InformationPackage can be described by zero or more MetadataEntities. [This may be how we want to handle the metsHdr in the future as a special kind of MetadataEntity.]

An InformationPackage is also a kind of Manifestation, allowing InformationPackages to be nested as representations of specific StructuralDivisions in other InformationPackages. This accommodates the <mptr> element in a <structMap> <div> elements.

In relation to the current XML Schema, InformationPackage represents the <mets> and <metsHdr> elements and their attributes.

## InformationEntity

The InformationEntity represents a digital object that is referenced by the InformationPackage or embedded in the InformationPackage. There are two kinds of InformationEntites: MetadataEntities and FileStreamEntities. InformationEntities can be described by zero or more MetadataEntities.

In relation to the current XML Schema, InformationEntity is a representation of all the METS object types whose data is either linked from or embedded in the METS document. These include <dmdSec>, <techMD>, <rightsMD>, <sourceMD>, <digiprovMD>, and <file> elements.

## InformationGrouping

InformationEntities can be grouped in various arbitrary ways. An InformationGrouping can contain one or more InformationEntities, and InformationEntities can be contained in zero or more InformationGroupings. InformationGroupings can also contain zero or more other InformationGroupings. InformationGroupings can be described by zero or more MetadataEntities.

In relation to the XML Schema, InformationGroupings accommodates both the hierarchy of <fileGrp> elements in the <fileSec> and the GROUPID attribute of the <file>, <dmdSec>, <techMD>, <rightsMD>, <sourceMD>, <digiprovMD> elements.

## MetadataEntity

A MetadataEntity is a kind of InformationEntity. A couple things distinguished a MetadataEntity from its sibling the FileStreamEntity. One is that a MetadataEntity can be the object of an “is described by” relationship with various other classes of objects in this model. Another is that a MetadataEntity cannot be a Manifestation of a StructuralDivision.

In relation to the XML Schema, MetadataEntity accommodates the <dmdSec>, <techMD>, <rightsMD>, <sourceMD>, <digiprovMD> elements.

## FileStreamEntity

A FileStreamEntity is a kind of InformationEntity. FileStreamEntities make up the primary content of an InformationPackage. There are two kinds of FileStreamEntities: FileEntities and StreamEntities. A FileStreamEntity is a kind of Manifestation, and a FileStreamEntity can be part of a FileArea.

In relation to the XML Schema, FileStreamEntity accommodates the <file> and <stream> elements.

## FileEntity

A FileEntity is a kind of FileStreamEntity and it represents one of the content files contained in the InformationPackage. A FileEntity may contain zero or more other FileEntities, for example a ZIP file which contains other files. A FileEntity may also contain zero or more StreamEntities, such as the different audio streams contained in an MPEG4 file.

In relation to the XML Schema, FileEntity accommodates the <file> element.

## StreamEntity

A StreamEntity is a kind of FileStreamEntity and it represents individual component streams of a parent FileEntity, such as the audio streams of an MPEG4 file.

In relation to the XML Schema, FileEntity accommodates the <stream> element.

## StructuralDivision

A StructuralDivision allows the contents of the InformationPackage to be organized in various ways. An InformationPackage must have one or more StructuralDivisions. A StructuralDivision can be described by one or more MetadataEntities.

A StructuralDivision can be related to zero or more other StructureDivisions. This allows StructuralDivisions to be arranged as hierarchies or as arbitrary graphs of nodes. The relationships are represented by the DivisionRelationship association class, allowing relationships to have their own properties such as relationship type or label. In relation to the XML Schema, the DivisionRelationship association class accommodates both the hierarchy of <div> elements in the <structMap>and the <structLink> XLinks.

StructuralDivisions can have zero or more Manifestations. If a StructuralDivisions has more than one Manifestation, they should be considered alternative Manifestations.

## Manifestation

A Manifestation ties a StructuralDivision to a specific representation of the division. A StructuralDivision can be represented by five different kinds of Manifestations: InformationPackage, FileStreamEntity, FileArea, ParallelFiles, and SequentialFiles.

## FileArea

A FileArea is a kind of Manifestation and it general represents just a portion or area of a FileStreamEntity. However in some cases, such as when referenced from a ParallelFiles class or a SequentialFiles class, it could represent the entire FileEntity or StreamEntity. A FileArea is part of a single FileStreamEntity. A FileArea can be described by zero or more MetadataEntities.

## ParallelFiles

ParallelFiles is a kind of Manifestation that represents an aggregation of FileAreas and SequentialFiles that must be rendered simultaneously in order to manifest a particular StructuralDivision. ParallelFiles can contain zero or more FileAreas and zero or more SequentialFiles.

## SequentialFiles

SequentialFiles is a kind of Manifestation that represents an aggregation of FileAreas and ParallelFiles that must be rendered sequentially in order to manifest a particular StructuralDivision. SequentialFiles can contain zero or more FileAreas and zero or more ParallelFiles.