

# LexEVS 6.1 Design Document - Detailed Design - Loader - HL7 MIF

## Contents of This Page

- [Use Cases](#)
- [Mapping from the .coremif Files to LexEVS](#)
  - [Core MIF source](#)
  - [RIM Core MIF](#)
    - [A Superimposed Hierarchy with Connections to the Vocabulary MIF?](#)
    - [What are Entities in the RIM Core MIF?](#)
    - [The Purpose of Entrypoints in the RIM core.](#)
  - [Vocabulary Core MIF](#)
    - [Relationships](#)
    - [Entities](#)
    - [Value Sets](#)
  - [To Be Determined](#)
- [Implementation Design](#)

## Document Information

**Author:** Bauer, Scott  
**Email:** Bauer.Scott@mayo.edu  
**Team:** LexEVS  
**Contract:** ST12-1106  
**Client:** NCI CBIIT  
National Institutes of Health  
US Department of Health and Human Services

## Revision History

Version	Date	Description of Changes	Author
1.0	2013/03/05	Initial Version	Bauer, Scott

## Use Cases

Previous implementations of the HL7 loaded into LexEVS have made use of artifacts in the RIM database. The current requirements call for loading data directly from the Model Interchange format XML documents also known as MIF. These sources expand the set of likely entities and relationships as well as add some assertions about concept domains and data types defined in the expanded definitions of the RIM available in the .coremif documents. This creates some problems around the unique identifier used for entities defined in the current HL7 load implementation as no correlation exists for the uniquely defined values in those portions of the .coremif documents where correlations to entities from the MIF to LexGrid will require further definition. What follows is a proposed implementation of the load of the MIF, with some questions as to how to deal with additional values and some interpretations of the MIF xml structure in the context of it's mapping to the LexGrid Data Model.

## Mapping from the .coremif Files to LexEVS

### Core MIF source

Two separate source files exist with the file extension coremif in the RIM distribution package, "DEF=UV=RIM=<version>" and "DEF=UV=VO=<version>." The first defines a kind of super hierarchy to the RIM and the second defines the values and relationships that are consistent with the current load of the HL7 RIM to LexEVS. Opportunities for expressing entities and assertions that define entity relationships exist for the "DEF=UV=RIM" or RIM core MIF that should be outlined here. Also connections from this MIF file to the DEF=UV=VO, or vocabulary, file can further define relationships to the values that are currently expressed in the old LexEVS mapping.

### RIM Core MIF

#### A Superimposed Hierarchy with Connections to the Vocabulary MIF?

The RIM core MIF expresses a number of tags under the root static model that are defined as subject area packages. We know from the defining xsd that these are sub-packages and we can see that one sub-package can be nested within another, implying that one package is a subclass of another. Furthermore some sub-packages define contained class tags implying yet another area of sub classification. We can deduce relationships from these that will allow us to map to LexEVS associations.

```

<mif:subjectAreaPackage name="MessageCommunicationsControl" sortKey="02">
  <mif:annotations>
    <mif:documentation>
      <mif:description>
        <mif:text>
          <mif:p>A collection of classes related to the technical definition and control of
        </mif:text>
      </mif:description>
    </mif:documentation>
  </mif:annotations>
</mif:subjectAreaPackage name="MessageControl" sortKey="01">

```

```

<mif:subjectAreaPackage name="CoreInfrastructure" sortKey="01">
  <mif:annotations>
    <mif:documentation>
      <mif:description>
        <mif:text>
          <mif:p>This subject area contains those classes that provide foundation elements for
        </mif:text>
      </mif:description>
    </mif:documentation>
  </mif:annotations>
  <mif:containedClass name="ActHeir"/>
  <mif:containedClass name="EntityHeir"/>
  <mif:containedClass name="InfrastructureRoot"/>
  <mif:containedClass name="RoleHeir"/>
</mif:subjectAreaPackage>

```

When the contained classes are elsewhere fully defined in the RIM core MIF, attributes and annotations are declared that normally might map to Entities and Entity Properties respectively in LexEVS. However, the attribute tagged values will sometimes bear references to a data type tag or a concept domain tag, both of which are defined in the vocabulary core MIF and correlate to the the code systems also defined there. We could pre-correlate this relationship by persisting this as a LexEVS association making the attribute a first class entity, or we could leave it as a property and post-correlate the relationship for interested users through a series of LexEVS calls.

```

<mif:containedClass>
  <mif:class name="AcknowledgementDetail" isAbstract="false">
    <mif:annotations>
      <mif:documentation>
        <mif:definition>
          <mif:text>
            <mif:p>A message that provides information about the communication, parsing or formal (non
          </mif:text>
        </mif:definition>
      </mif:documentation>
    </mif:annotations>
    <mif:attribute name="typeCode" sortKey="005" isImmutable="true" minimumMultiplicity="1" maximumMultiplicit
    <mif:annotations>

```

```

  <mif:type name="CS"/>
  <mif:vocabulary>
    <mif:conceptDomain name="AcknowledgementDetailType"/>
  </mif:vocabulary>
</mif:attribute>

```

Some instances of contained classes are otherwise defined as concepts in code systems designated by the definingVocabulary tag in the RIM core MIF. This is another relationship between files that will need consideration for definition in LexEVS.

```

<mif:containedClass>
  <mif:class name="Account" isAbstract="false">
    <mif:definingVocabulary>
      <mif:code codeSystemName="ActClass" code="ACCT"/>
    </mif:definingVocabulary>
  </mif:class>
</mif:containedClass>

```

## What are Entities in the RIM Core MIF?

If it's decided that subjectAreaPackage definitions and containedClasses should be represented as Entities in LexEVS then additional tasks will be creating unique entity identifiers for them. Defining tags and attributes in these elements can be mapped to presentations, definitions and other property types as necessary.

## The Purpose of Entrypoints in the RIM core.

Some few values are expressed in the RIM core MIF as attributes in an entryPoint tag, if these should be expressed in LexEVS, some determination will have to be made.

## Vocabulary Core MIF

### Relationships

The vocabulary core MIF will map in a manner similar to the previous load, with some exceptions. Relationships will be defined with names noted in the concept relationship tag attribute "relationshipName" where previously all relationships were named "hasSubtype." Otherwise, relationships with the code system will be created based on containment and can maintain the hasSubtype designation. Most hierarchies defined in LexEVS provide a leaf-to-root resolve direction. The HL7 load was an exception. This should be revisited in this design.

### Entities

We will otherwise map the definition tag to a definition type property in the LexGrid and concept property tag attributes will define the numerical portion of the unique identifier where the concept property name is "internalId." The code tag will define the other portion of the unique identifier using the code attribute to concatenate to the end of the numerical identifier preceded by a colon. These should complete conformance, to a certain extent, with old LexEVS load of the RIM from the MSAccess database.

### Value Sets

Value sets are expressed in the vocabulary core MIF that are not currently being loaded to LexEVS. It appears that these value sets are tied to code systems defined in the MIF and that other values from other coding schemes are sometimes combined with them when applied using the unionWithContent tag. An evaluation of these will need to be made to determine how and whether to load these as value sets in LexEVS.

## To Be Determined

1. How value sets are defined in the vocabulary core MIF and how and whether they will be mapped to LexEVS.
2. How much of the RIM core MIF should be mapped to LexEVS, if any, and whether relationships implied in the definingCodeSystem, dataType and conceptDomain tags should be expressed in LexEVS.
3. whether attributes in the RIM core MIF under contained classes be first class entities.
4. whether entryPoint designations should somehow be expressed in LexEVS (These are rather opaquely defined in the static-base xsd. "Identifies a class within the model that may be used as the initial class in a serializable representation of the model" for the purposes of defining this as an entry to a vocabulary representation in LexEVS).
5. verification of all mapping proposals with stakeholders.

## Implementation Design

The loading application will stream and parse the XML elements into coding scheme, entity and association elements where they can be tied together as a coding scheme object and loaded to the LexGrid data base.

# HL7 MIF Loader

