A Model for Linguistic Resource Description

Nancy Ide
Department of Computer Science
Vassar College
Poughkeepsie, New York, USA
ide@cs.vassar.edu

Keith Suderman
Department of Computer Science
Vassar College
Poughkeepsie, New York, USA
suderman@anc.org

Abstract

This paper describes a comprehensive standard for resource description developed within ISO TC37 SC4). The standard is instantiated in a system of XML headers that accompany data and annotation documents represented using the the Linguistic Annotation Framework's Graph Annotation Format (GrAF) (Ide and Suderman, 2007; Ide and Suderman, Submitted). It provides mechanisms for describing the organization of the resource, documenting the conventions used in the resource, associating data and annotation documents, and defining and selecting defined portions of the resource and its annotations. It has been designed to accommodate the use of XML technologies for processing, including XPath, XSLT, and, by virtue of the system's linkage strategy, RDF/OWL, and to accommodate linkage to web-based ontologies and data category registries such as the OLiA ontologies (Chiarcos, 2012) and ISOCat (Marc Kemps-Snijders and Wright, 2008).

1 Introduction

While substantial effort has gone into defining standardized representation formats for linguistically annotated language resources, very little attention has been paid to standardizing the metadata and documentation practices associated with these resources (see, for example, (Ide and Pustejovsky, 2010)). Multiple techniques have been proposed to represent resource provenance, and a W3C Working Group[1] has recently been convened to devise means to enable provenance information to be exchanged, in particular for data originating from and/or distributed over the web. Beyond this, there exist some standard practices for resource publication through established data distribution centers such as the Linguistic Data Consortium (LDC)[2] and ELRA[3], but they are not completely consistent among different centers, and they are not comprehensive. Whether a resource is distributed from a data center or via the web, detailed information about methodology, annotation schemes, etc. is often sparse. However, users need this kind of information to not only use but also assess the quality of a resource, replicate processes and results, and deal with idiosyncrasies or documented errors.

Another area that has received virtually no attention involves standardized strategies for formally describing the structure and organization of a resource. Information about directory structure and relations among files is typically provided in accompanying README files that provide no means to ensure that the requisite components are in place or perform systematic processing without developing customized scripts. Formalized description of resource organization would enable automatic validation as well as enhanced processing capabilities.

This paper describes a comprehensive standard for resource description developed within ISO TC37 SC4[4]. The standard is instantiated in a system of XML headers that accompany data and annotation documents represented using the the Linguis-

[1] http://www.w3.org/2011/01/prov-wg-charter.html

[2] http://www.ldc.upenn.edu

[3] http://www.elra.info

[4] http://www.tc37sc4.org

tic Annotation Framework's Graph Annotation Format (GrAF) (Ide and Suderman, 2007; Ide and Suderman, Submitted). It provides mechanisms for describing the organization of the resource, documenting the conventions used in the resource, associating data and annotation documents, and defining and selecting defined portions of the resource and its annotations. It has been designed to accommodate the use of XML technologies for processing, including XPath, XSLT, and, by virtue of the system's linkage strategy, RDF/OWL, and to accommodate linkage to web-based ontologies and data category registries such as the OLiA ontologies (Chiarcos, 2012) and ISOCat (Marc Kemps-Snijders and Wright, 2008). We first describe the general architecture of resources rendered in GrAF, followed by a description of the headers that instantiate the resource description standard.

2 GrAF Overview

GrAF has been developed with ISO TC37 SC4 to provide a general framework for representing linguistically annotated resources. Its design has been informed by previous and current approaches and tools, including but not limited to UIMA CAS(Ferrucci and Lally, 2004), GATE (Cunningham et al., 2002), ANVIL (Kipp, Forthcoming), ELAN (Auer et al., 2010), and the NLP Interchange Format (NIF)[5] under development within the Linked Open Data (LOD) effort[6]. The approach has been to develop a *lingua franca* or "pivot" format into and out of which other models may be translated in order to enable exchange among systems.[7] In order to serve this purpose, the GrAF data model was designed to capture the *relevant structural generalization* underlying best practices for linguistic annotation, which is the directed (acyclic) graph.

The overall architecture of a linguistically-annotated resource rendered in GrAF consists of the following:

- One or more *primary data documents*, in any medium;
- One or more documents defining a set of regions over each primary data document, each of which may serve as a *base segmentation* for annotations;
- Any number of *annotation documents* containing feature structures associated with nodes and/or edges in a directed graph; all nodes reference either a base segmentation document (in which case the node is a 0-degree node with no outgoing edges) or are connected to other nodes in the same or other annotation documents via outgoing edges;
- *Header documents* associated with each primary data document and annotation document, and a resource header that provides information about the resource as whole.

We describe the GrAF headers below, followed by a brief overview of how header elements are used in primary data, segmentation, and annotation documents. Note that the full description of GrAF, including GrAF schemas and a description of all components, elements, and attributes, appears in the LAF ISO Candidate Draft; similar GrAF documentation together with schemas in a variety of formats are available at `http://www.anc.org/graf`.

3 The GrAF Headers

In GrAF, all primary data, segmentation, and annotation documents, as well as the resource as a whole, require a header to provide a formal description of the various properties of the resource component. All of the headers have been designed with the aim of facilitating the automatic processing and validation of the resource content and structure.

3.1 Resource header

The GrAF resource header provides metadata for the resource by establishing resource-wide definitions and relations among files, datatypes, and annotations that support automatic validation of the resource file structure and contents. The resource header is based on the XML Corpus Encoding Standard (XCES

[5] http://blog.aksw.org/2011/nlp-interchange-format-nif-1-0-spec-demo-and-reference-implementation/

[6] http://linkeddata.org/

[7] This approach that has been widely adopted in the standardization field as the most pragmatic way to provide interoperability among tools, systems, and descriptive information such as metadata and linguistic annotations.

)header[8], omitting the information that is relevant only to single documents. A `resourceDesc` (resource description) element is added that describes the resource's characteristics and provides pointers to supporting documentation. The relevant elements in the resource description are as follows:

fileStruct: Provides the file structure of the resource, including the directory structure and the contents of each directory (additional directories and individual files). A set of fileType declarations describe the data files in the resource. Each is associated via attributes with a medium (content type), a set of annotation types, an optional name suffix, an indication of whether or not the file type is required to be present for each primary data document in the resource, and a list of one or more file types required by this filetype for processing.

annotationSpaces: Provides a set of one or more annotation spaces, which are used in a way similar to XML namespaces. AnnotationSpaces are needed especially when multiple annotations of the same data are merged, to provide context and resolve name conflicts.

annotationDecls: A set of one or more annotation declarations, which provide information about each annotation type included in the resource, including the annotation space it belongs to, a prose description, URI for the responsible party (creator), the method of creation (automatic, manual, etc.), URI for external documentation, and an optional URI for a schema or schemas providing a formal specification of the annotation scheme.

media: Provides a set of one or more medium types that files may contain, the type, encoding (e.g., utf-8), and the file extension used on files containing data of this type.

anchorTypes: a set of one or more types of anchors used to ground annotations in primary data (e.g., character-anchor, time-stamp, line-segment, etc.), the medium with which these anchor types are used, and a URI for a formal specification of the anchor type.[9] Via this mechanism, different anchor types have different semantics, but all GrAF anchors are represented in the same way so that a processor can transform the representation without consulting the definition or having to know the semantics of the representation, which is provided externally by the formal specification.

groups: Definition of one or more groups of annotations that are to be regarded as a logical unit for any purpose. The most common use of groups is to associate annotations that represent a "layer" or "tier"[10], such as a morpho-syntactic or syntactic layer. However, grouping can be applied to virtually any set of annotations. GrAF provides five types of grouping mechanisms:

1. *annotation*: annotations with specific values for their labels (as given on the @LABEL attribute of an `a` element in an annotation document) and/or annotation space. Wildcards may be used to select sets of annotations with common labels or annotation spaces, e.g., `*:tok` selects all annotations with label *tok*, in any annotation space (designated with "*:"), `xces:*` selects all annotations in the xces annotation space.

2. *type*: annotations of a specific type or types, by referencing the id of an annotation declaration defined in the resource header;

3. *file*: annotations appearing in a specific file type or types, by referring to the id of a file type defined in the resource header;

4. *enumeration*: an enumerated list of annotation ids appearing in a specified annotation document;

5. *expression*: an xPath-like expression that can navigate through annotations–for example, the expression @SPEAKER='ALICE' would choose all annotations with a feature named *speaker* that has the value *Alice*;

[8] http://www.cs.vassar.edu/CES/CES1-3.html

[9] Note that all anchor types are associated with one or more media, but a medium is not necessarily associated with an anchor type–in particular, media types associated with documents other than primary data documents (notably, annotation documents) are not associated with an anchor type.

[10] Groupings into layers/tiers are frequently defined in speech systems such as ELAN and ANVIL.

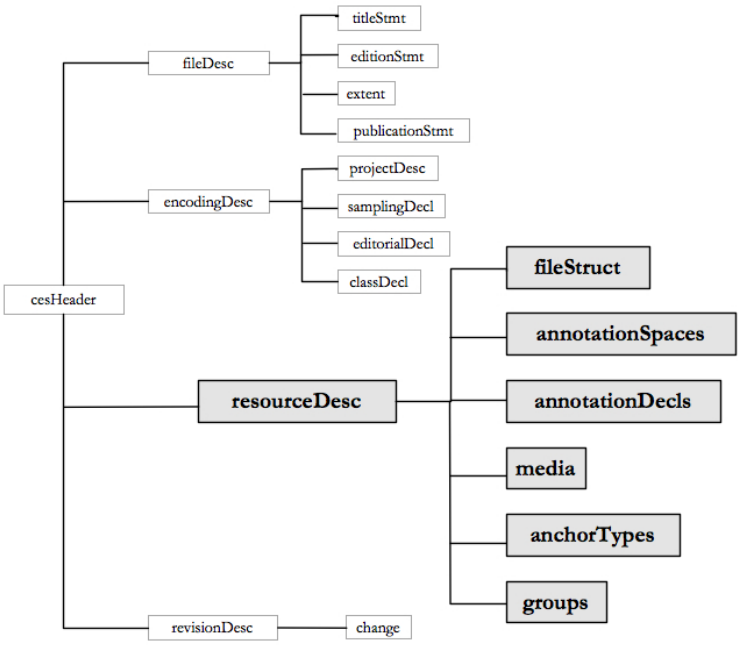

Figure 1: Main elements of the resourceDesc element in the GrAF resource header.

6. *group*: another group or set of groups. This can be used, for example, to group several enumeration groups in order to group enumerated annotation ids in multiple annotation documents.

All files, annotation spaces, annotations, media, anchors, and groups have an @xml:id attribute, which is used to relate object definitions where applicable. Figure 3 provides an example of a groups definition illustrating the different grouping mechanisms as well as the use of ids for cross-reference among objects defined in the header. It assumes declarations of the form shown in Figure 2 elsewhere in the resource header. The dependencies for several of these elements are shown graphically in Figure 4, which also shows the use of the @SUFFIX attribute for file types and the @EXTENSION attribute for media in a sample file name.

3.2 Primary data document header

The primary document header is stored in a separate XML document with root element `documentHeader`. The document header contains TEI-like elements for describing the primary data document, including its title, author, size, source of the original, language and encoding used in the document, etc., as well as a `textClass` element that provides genre/domain information by referring to classes defined in the resource header. Additional elements provide the locations of the primary data document and all associated annotation documents, using either a path relative to the root (declared on a `directory` element in the resource header) or a URI or persistent identifier (PID).

3.3 Annotation document header

Annotation documents contain both a header and the graph of feature structures comprising the annotation. The annotation document header is brief; it provides four pieces of information:

1. a list of the annotation labels used in the document and their frequencies;

2. a list of documents required to process the annotations, which will include a segmentation document and/or any annotation documents directly referenced in the document;

3. a list of annotationSpaces referenced in the document, one of which may be designated as a default for annotations in the document;

```
<fileType xml:id="f.entities" suffix="ne" a.ids="a.ne"
       medium="xml" requires="f.ptbtok"/>
...
<annotationSpace xml:id="xces" pid="http://www.xces.org/schema/2003"/>
...
<annotationDecl xml:id="a.ne" as="xces">
   <a.desc>named entities</a.desc>
   <a.resp lnk:href="http://www.anc.org">ANC project</a.resp>
   <a.method type="automatic-validated"/>
   <a.doc lnk:href="https://www.anc.org/wiki/wiki/NamedEntities"/>
</annotationDecl>
...
 <medium xml:id="text" type="text/plain" encoding="utf-8" extension="txt"/>
 <medium xml:id="xml" type="text/xml" encoding="utf-8" extension="xml"/>
...
 <anchorType medium="text" default="true"
        lnk:href="http://www.xces.org/ns/GrAF/1.0/#character-anchor"/>
```

Figure 2: Definitions in the GrAF resource header

```
<groups>
   <group xml:id="g.token">
      <!-- all annotations in any annotation space with label "tok" -->
      <g.member value="*:tok" type="annotation"/>
   </group>
   <group xml:id="g.example">
            <!-- all annotations of type logical -->
            <g.member value="a.logical" type="type"/>
            <!-- all files of containing entity annotations -->
            <g.member value="f.entities" type="file"/>
            <!-- all annotations with a feature "speaker" with value "Alice" -->
            <g.member value="@speaker='alice'" type="expression"/>
            <!-- annotations with ids "id_1" to "id_n" in file "myfile.xml"-->
            <g.member xml:base="myfile.xml" value="id1 id2 ... idN"
                     type="enumeration"/>
            <!-- the annotations included in group g.token, as defined earlier -->
            <g.member value="g.token" type="group"/>
   </group>
</groups>
```

Figure 3: Group definitions in the GrAF resource header

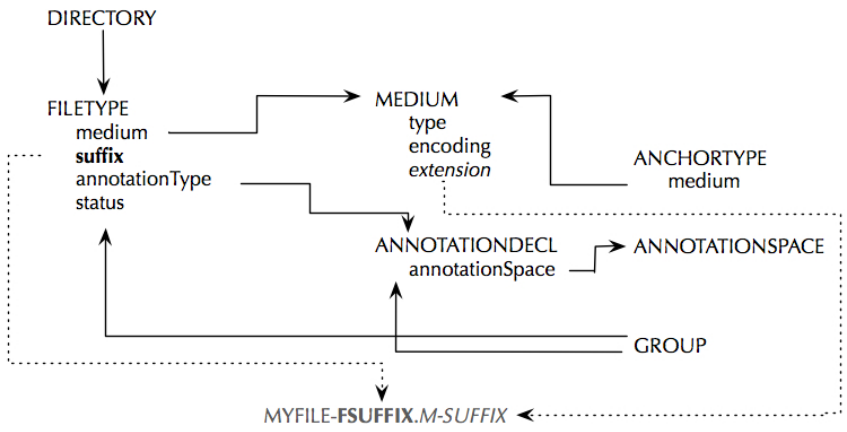

Figure 4: Dependencies among objects in the resource header

4. (optional) The root node(s) in the graph, when the graph contains one or more graphs that comprise a well-formed tree.

Information about references to other documents is intended for use by processing software, to both validate the resource (ensure all required documents are present) and facilitate the loading of required documents for proper processing. Information about annotation spaces provides a reference to required information in the resource header. When there is more than one tree in a graph, specification of their root nodes is required for proper processing. An example annotation document header is shown in Figure 5.

Following the header, annotation documents contain a graph or graphs and their associated annotations. LAF recommends that each annotation type or layer be placed in a separate annotation document, although in the absence of a standard definition of layers it is likely that there will be considerable variation in how this is implemented in practice. A newly-proposed ISO work item will address this and other organization principles in the near future.

4 Using Resource Header Elements

4.1 Primary data documents

Primary data in a LAF-compliant resource is frozen as read-only to preserve the integrity of references to locations within the document or documents. This, a primary data document will contain only the data that is being annotated. Corrections and modifications to the primary data are treated as annotations and stored in a separate annotation document.

In the general case, primary data does not contain markup of any kind. If markup appears in primary data (e.g., HTML or XML tags), it is treated as a part of the data stream by referring annotations; no distinction is made between markup and other characters in the data when referring to locations in the document. Although LAF does not recommend anchoring annotations in primary data by referencing markup, when necessary, XML elements in a document that is valid XML may be referenced by defining a medium type as XML and defining the associated anchor type as an XPath expression. References to locations within these XML elements (i.e., XML element content) can be made using standard offsets, which will be computed by including the markup as part of the data stream; in this case, two media types would be associated with the primary document's file type, as shown in Figure 6.

4.2 Segmentation: regions and anchors

Segmentation information is specified by defining *regions* over primary data. Regions are defined in terms of *anchors* that directly reference locations in primary data. All anchors are typed; anchor types used in the resource are each defined with an `anchorType` element in the resource header (see Section 3.1). The type of the anchor determines its semantics and therefore how it should be processed by an application. Figure 8 shows a set of region definitions and the associated anchor type and medium definitions from the resource header.[11]

Anchors are first-class objects the LAF data model (see Figure 7) along with regions, nodes, edges, and links. The anchor is the only object in the model that may be represented in two alternative ways in the GrAF serialization: as a the value of an @ANCHORS attribute on the `region` element, or with an `anchor` element. When anchors are represented with the `anchor` element, the `region` element will include a @REFS attribute (and must not include an @ANCHORS attribute) providing the ids of the associated anchors. For example, an alternative representation for region "r2" in Figure 8 is given in Figure 9.

In general, the design of GrAF follows the principle of orthogonality, wherein there is a single means to represent a given phenomenon. The primary reason for allowing alternative representations for anchors is that the proliferation of `anchor` elements in a segmentation document is space-consuming and potentially error-prone. As shown in Figure 8, the attribute representation can accommodate most references into text, video, and audio; the only situation in which use of an `anchor` element may be necessary is one where a given location in a document needs to be interpreted in two or more ways, as, for example, a part of two regions that should not be considered to have a common border point. In this case, multiple `anchor` elements can be de-

[11] Note that the @TYPE attribute on the `region` element specifies the anchor type and not the region type.

```
<graph xmlns="http://www.xces.org/ns/GrAF/1.0/">
 <header>
   <labelsDecl>
     <labelUsage label="Target" occurs="171"/>
     <labelUsage label="FE" occurs="372"/>
     <labelUsage label="sentence" occurs="32"/>
     <labelUsage label="NamedEntity" occurs="32"/>
   </labelsDecl>
   <dependencies>
     <dependsOn file_type.id="fntok"/>
   </dependencies>
   <annotationSpaces>
     <annotationSpace as.id="FrameNet" default="true"/>
   </annotationSpaces>
 </header>
...
```

Figure 5: Annotation document header

```
<fileType xml:id = "f.primary" medium="text xml"/>
<medium xml:id = "text" type="text/plain" encoding = "utf-8" extension = "txt"/>
<medium xml:id = "xml" type = "xml" encoding = "utf-8" extension = "xml"/>
<anchorType medium = "xml" default = "true"
            lnk:href = "http://www.w3.org/TR/xpath20/"/>
<anchorType medium = "text"
            lnk:href = "http://www.xces.org/ns/GrAF/1.0/#character-anchor"/>
```

Figure 6: Referencing XML elements in primary data

```
<anchor xml:id="a1" value="10,59"/>
<anchor xml:id="a2" value="10,173"/>
<anchor xml:id="a3" value="149,173"/>
<anchor xml:id="a4" value="149,59"/>

<region xml:id="r2" refs="a1 a2 a3 a4"
   anchor_type="image-point"/>
```

Figure 9: Region and anchor definitions

fined that reference the same location, and each anchor may then be uniquely referenced. Because of its brevity and in the interests of orthogonality, the attribute representation is recommended in LAF.

4.2.1 Segmentation documents

An annotation document is called a *segmentation document* if it contains only segmentation information–i.e., only `region` and `anchor` elements. Although regions and anchors may also be defined in an annotation document containing the graph of annotations over the data, LAF strongly recommends that when a segmentation is referenced from more than one annotation document, it appears in an independent document in order to avoid a potentially complex jungle of references among annotation documents.

A *base segmentation* for primary data is one that defines minimally granular regions to be used by different annotations, usually annotations of the same type. For example, it is not uncommon that different annotations of the same text–especially annotations created by different projects–are based on different tokenizations. A base segmentation can define a set of regions that include the smallest character span isolated by any of the alternative tokenizations–e.g., for a string such as “three-fold”, regions spanning “three”, ”-”, and ”fold” may be included; a tokenization that regards “three-fold” as a single token can reference all three regions in the @TARGETS attribute on a `link` element associated with the node with which the token annotation is attached, as shown in Figure 10.

Multiple segmentation documents may be associated with a given primary data document. This is useful when annotations reference very different

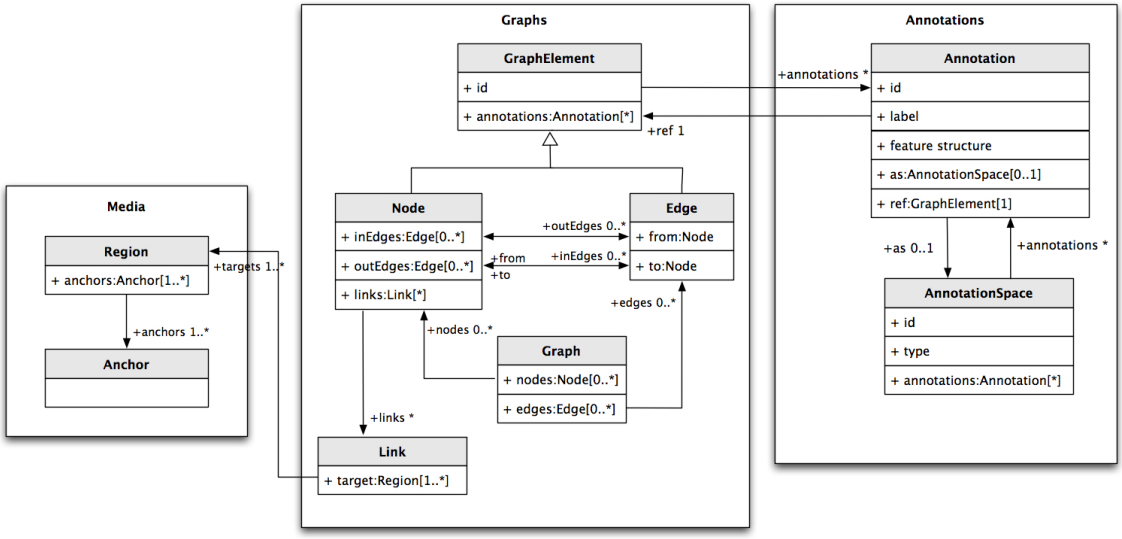

Figure 7: LAF model

```
<!-- Definitions in the resource header -->
<medium xml:id="text" type="text/plain" encoding="utf-8" extension="txt"/>
<medium xml:id="audio" type="audio" encoding="MP4" extension="mpg"/>
<medium xml:id="video" type="video" encoding="Cinepak" extension="mov"/>
<medium xml:id="video" type="image" encoding="jpeg" extension="jpg"/>
...
<anchorType xml:id="text-anchor" medium="text" default="true"
        lnk:href="http://www.xces.org/ns/GrAF/1.0/#character-anchor"/>
<anchorType xml:id="time-slot" medium="audio"
        lnk:href="http://www.xces.org/ns/GrAF/1.0/#audio-anchor"/>
<anchorType xml:id="video-anchor" medium="video"
        lnk:href="http://www.xces.org/ns/GrAF/1.0/#video-anchor"/>
<anchorType xml:id="image-point" medium="image"
        lnk:href="http://www.xces.org/ns/GrAF/1.0/#image-point"/>

<!-- Regions in the segmentation document -->
<region xml:id="r1" anchor_type="time-slot" anchors="980 983"/>
<region xml:id="r2" anchor_type="image-point" anchors="10,59 10,173 149,173 149,59"/>
<region xml:id="r3" anchor_type="video-anchor" anchors="fr1(10,59) fr2(59,85) fr3(85,102)"/>
<region xml:id="r4" anchor_type="text-anchor" anchors="34 42"/>
```

Figure 8: Region and anchor definitions

```
<region xml:id="seg-r770" anchors="211 216"/>
<region xml:id="seg-r771" anchors="216 217"/>
<region xml:id="seg-r772" anchors="217 221"/>

<node xml:id="n1019">
  <link targets="seg-r770 seg-r771 seg-r772"/>
</node>
<a label="tok" ref="n1019" as="xces">
  <fs>
      <f name="msd" value="JJ"/>
  </fs>
</a>
```

Figure 10: Referencing multiple regions

regions of the data; for example, in addition to the base segmentation document containing the minimal character spans that is partially shown in Figure 10, there may also be a segmentation based on sentences, which may in turn be referenced by annotations for which this unit of reference is more appropriate.[12] Alternative segmentations for different granularities, such as phonetic units, may also be useful for some purposes.

4.3 Annotation documents

In addition to the header, annotation documents contain a graph consisting of nodes and edges, either of which may be associated with an annotation. Annotations associated with a node or edge are represented with `a` elements that have a @REF attribute that provides the id of the associated node. The @LABEL attribute on an `a` element gives the main category of the annotation; this may be the string used to identify the annotation as described by the annotation documentation referenced in the annotation type declaration in the resource header, a category identifier from a data category registry such as ISOCat, an identifier from a feature structure library, or any PID reference to an external annotation specification. Each annotation is also associated with an *annotation space*, as defined in the resource header, which is referenced in the annotation document header. Figure 11 shows an example of an annotation for FrameNet that includes the annotation

```
<node xml:id="fn-n2"/>
<a label="FE" ref="fn-n2"  as="FrameNet">
 <fs>
  <f name="name" value="Recipient"/>
  <f name="GF" value="Obj"/>
  <f name="PT" value="NP"/>
 </fs>
</a>
```

Figure 11: Node with associated annotation

space in the AS attribute of the `a` element.[13]

5 Conclusion

We provide here a general overview of a system for formal description of a linguistically annotated resource, designed to allow automatic validation and processing of the resource. It provides means to define the file structure of a resource and specify inter-file requirements and dependencies so that the integrity of the resource can be automatically checked. The scheme also provides links to metadata as well as annotation semantics, which may exist externally to the resource itself in a database or ontology, and provides mechanisms for defining grouping of selected annotations or files based on a wide range of criteria.

Although some of these mechanisms for resource documentation have been implemented in other schemes or systems, to our knowledge this is the first attempt at a comprehensive documentation system for linguistically annotated resources. It addresses a number of requirements for resource documentation and description that have been identified but never implemented formally, such as documentation of annotation scheme provenance, means of production, and resource organization and dependencies. Many of these requirements were first outlined in the Sustainable Interoperability for Language Technology (SILT) project[14], funded by the U.S. National Science Foundation, which drew input from the community at large.

Similar to the graph representation for annotations, the GrAF documentation system is designed to be easily integrated with or mappable to other

[12] Sentences may also be represented as annotations defined over tokens, but for some purposes it is less desirable to consider a sentence as an ordered set of tokens than as a single span of characters.

[13] Note that if the annotation document header in Figure 5 were used, no AS attribute would be needed to specify the FrameNet annotation space, since it is designated as the default.

[14] http://www.anc.org/SILT/

schemes, especially those relying on Semantic Web technologies such as RDF/OWL. However, it should be noted that GrAF is equally suitable for resources that are not primarily web-based (i.e., do not link to information elsewhere on the web) and therefore do not require the often heavy mechanisms required for Semantic Web-based representations.

Due to space constraints, many details of the GrAF scheme are omitted or mentioned only briefly. The MASC corpus (Ide et al., 2008; Ide et al., 2010), freely downloadable from `http://www.anc.org/MASC`, provides an extensive example of a GrAF-encoded resource, including multiple annotation types as well as the resource header and other headers. Other examples of GrAF annotation, including annotation for multi-media, are provided in (Ide and Suderman, Submitted).

Acknowledgments

This work was supported by National Science Foundation grant INT-0753069.

References

Eric Auer, Albert Russel, Han Sloetjes, Peter Wittenburg, Oliver Schreer, S. Masnieri, Daniel Schneider, and Sebastian Tschpel. 2010. Elan as flexible annotation framework for sound and image processing detectors. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Christian Chiarcos. 2012. Ontologies of linguistic annotation: Survey and perspectives. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC)*.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. 2002. GATE: A framework and graphical development environment for robust nlp tools and applications. In *Proceedings of ACL'02*.

David Ferrucci and Adam Lally. 2004. UIMA: An architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348.

Nancy Ide and James Pustejovsky. 2010. What does interoperability mean, anyway? Toward an operational definition of interoperability. In *ICGL 2010: Proceedings of the Second International Conference on Global Interoperability for Language Resources*, Hong Kong, China.

Nancy Ide and Keith Suderman. 2007. GrAF: A graph-based format for linguistic annotations. In *Proceedings of the First Linguistic Annotation Workshop*, pages 1–8, Prague.

Nancy Ide and Keith Suderman. Submitted. The Linguistic Annotation Framework: A Standard for Annotation Interchange and Merging. *Language Resources and Evaluation*.

Nancy Ide, Collin Baker, Christiane Fellbaum, Charles Fillmore, and Rebecca Passonneau. 2008. MASC: The Manually Annotated Sub-Corpus of American English. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco.

Nancy Ide, Collin Baker, Christiane Fellbaum, and Rebecca Passonneau. 2010. The Manually Annotated Sub-Corpus : A community resource for and by the people. In *Proceedings of the The 48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden.

Michael Kipp. Forthcoming. Anvil: A universal video research tool. In G. Kristofferson J. Durand, U. Gut, editor, *Handbook of Corpus Phonology*. Oxford University Press.

Peter Wittenburg Marc Kemps-Snijders, Menzo Windhouwer and Sue Ellen Wright. 2008. Isocat: Corralling data categories in the wild. In et al. Nicoletta Calzolari, editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).