2006 E-MELD Workshop on Digital Language Documentation

Wayne State University - Eastern Michigan University

Tools and Standards:
The State of the Art

June 20-22, in conjunction with the 2006 LSA Summer Meeting

Electronic Metastructure for EMELD Endangered Languages Data

A UNIVERSAL DATA MODEL FOR LINGUISTIC ANNOTATION TOOLS

By

Scott Farrar

Paper presented at

2006 E-MELD Workshop on Digital Language Documentation
Lansing, MI.
June 20-22, 2006

Please cite this paper as:

Farrar, S. (2006), A universal data model for linguistic annotation tools, *in* ‘Proceedings of the EMELD’06 Workshop on Digital Language Documentation: Tools and Standards: The State of the Art’. Lansing, MI. June 20-22, 2006.

A universal data model for linguistic annotation tools

Scott Farrar
University of Arizona

Abstract

An important goal of the E-MELD enterprise is to recommend best-practice standards and resources in support of the digitization of endangered language data. As a result, there have been several proposals put forth at E-MELD sponsored events for best-practice data models, in particular, for dictionaries, paradigms, and interlinear text. While these proposals emphasize structural and encoding interoperability, by using XML and Unicode respectively, the resulting data models are not necessarily interoperable with respect to content. One suggestion for going beyond mere structural and encoding interoperability was to include in the various data models a reference to a common markup ontology, such as the General Ontology for Linguistic Description. As of yet, however, there have been few suggestions on how to implement such a model that emphasizes content interoperability via an ontology. This paper attempts to fill the gap by describing a common data exchange format to be useful for a variety of data digitization tools.

1 Introduction

Since its inception, the E-MELD project has advocated the use of **best practice** in the digitization and markup of language data. *Best practice* means creating resources that are "…longlasting, accessible, and re-usable by other linguists and speakers" [1]. One of the most well articulated calls for best practice was presented by Bird & Simons (2003), and subsequently adopted by the E-MELD community. Focusing directly on the tasks of language documentation and language description, they emphasize seven steps towards the best practice of digital data, seven steps that they refer to as the "portability of language data". One of the steps, what Bird & Simons (2003) term *format*, concerns the best practice of encoding and markup. Under this general rubric, there have been several data models proposed, in particular, for dictionaries, paradigms, and interlinear text. While these proposals emphasize structural and encoding compatibility, mostly through the recommendation of using XML and Unicode respectively, the resulting data models are not necessarily interoperable with respect to *content*. One suggestion proposed by Lewis, Farrar & Langendoen (2001) and Bird & Simons (2003) was to go beyond mere structural and encoding interoperability and relate the elements of the various data models to a common markup ontology, such as the General Ontology for Linguistic Description (GOLD) (Farrar & Langendoen 2003). As of yet, however, there have been few suggestions on how to tie together data *format* with data *content*, specifically concerning the relationship to an ontology. This paper attempts to fill the

[1] See http://emeld.org/school/what.html.

gap by describing a data model that has the potential to facilitate migration of XML to richer structures.

The data model advocated here has as its impetus the need for sharing the same data among annotation tools with very different purposes, such that different aspects of the same data can be manipulated by each kind of tool. For example, consider a lexicon creation tool such as FIELD (Aristar 2003) whose output is a highly structured lexicon. If the data were structured according to a universally recognized model, then the results could then be loaded into another kind of tool, for instance, one that produces interlinear text based on the lexicon, or one that adds detailed phonetic annotation to each entry. The most important requirement is that the data exchange format accommodate the fundamental linguistic data types, both of the traditional print variety (e.g., dictionary entries and interlinear text) and of a more technical nature, such as those used in natural language processing applications (e.g., treebanks and computational lexicons). In § 2, we examine these data types to create an inventory of basic elements that serve as the basis for a mapping to an ontology. Another important requirement of the model is that it be conversion "friendly", not only to ensure that the data can be displayed in a human-readable format, but mostly importantly, to ensure that the data is compatible with various tools, that is, migratable to a semantically interoperable form. Thus, the main design issues surrounding display- versus content-oriented data structures are discussed in detail in § 4. Third, the role of current markup standards is discussed and how they can be leveraged to create a more structured data exchange format. Therefore in § 5, we discuss the use of the Resource Description Framework (RDF) and the Web Ontology Language (OWL) as a means of adding more structure to vanilla XML. Once these desiderata are established, we present our data model in § 6.

2 Fundamental data types

Descriptive linguistics is a discipline that is—in no small way—driven by tradition. This can be seen in the **data types** that linguists generally use to present analyses of language data. A linguistic data type is any structured entity that acts as a container for annotated data and the elements of annotation, often referred to as the *analysis*. For example, the tradition of using interlinear glossed text (IGT) is particularly salient in print journals. In various descriptive grammars, on the other hand, there is the tradition of using phonological and morphological paradigms, essentially multidimensional tables showing feature systems of a language. Of course there is the lexicography tradition that focuses largely on how to display or organize lexical entries in a format that is maximally beneficial for the human user in a print environment. There are also traditions of using tree diagrams for morphological, syntactic, and phonological structure. Not to be left out is the tradition of using phonological and syntactic rules in, for example, a deeper grammatical analysis. More recently, however, some branches of linguistics (specifically computational linguistics and natural language processing) have begun to place an emphasis on more formal data structures, used in language resources, specifically tailored for machine readability. For example, there are the treebank data structures that provide a means of representing structural descriptions in an efficient format. Furthermore, there are some very successful electronic dictionaries that encode lexical structures and relations to be read by a computer. Whether print-based or electronic, all these entities can be considered as linguistic data types. The following section presents a discussion of the fundamental data types[2] and summarizes the explicit and implicit content most often associated with each type.

2.1 Interlinear glossed text

The first data type to consider is interlinear glossed text (IGT) which is characterized by a tabular presentation of morphemes and their labels, usually aligned vertically on the page. Bow, Hughes & Bird (2003, Sec. 3) cite the association of a morpheme with a label as the most consistent feature of IGT. An instance of IGT starts with a segment of text. Usually, the text is presented in some recognized orthography. But as noted by Peterson (2000), there can be several layers of transcription, including a phonetic or phonemic transcription, but also transcriptions in a second orthography. The text is segmented somehow showing morpheme, word, and possibly phrase boundaries. Then, there are the glosses of the morphemes that compose the text (either in the form of lexical items from the language of description or abbreviations for grammatical or semantic categories (e.g., 3PL, PAST, ANIM). This is followed by a free translation in the language of description or some other language of scholarship.

This is essentially the explicit information given in an instance of IGT. There is, on the other hand, implicit information that adds to the basic entities discussed above. For instance, consider the Leipzig Glossing Rules that recommend structures rich in information on morpheme type (Bickel, Comrie & Haspelmath 2004, pp. 2–7). First of all, clitic boundaries (and hence the existence of clitics) may be indicated with an equals sign between morphemes. The characterization of some morpheme as a portmanteau morpheme may also be present, indicated by a period in the gloss line. Forms such as stems are indicated using a backslash to separate them from the inflectional or derivational material. Also noted in the Glossing Rules are the agent-like and patient-like arguments of a verb. There are "bipartite elements" such as infixes and circumfixes, marked in a number ways, and morpho-phonological information such as reduplication, indicated with a tilde.

2.2 Paradigms

Next, we turn to paradigms. For our discussion , we simply highlight a few observations already made Penton, Bow, Bird & Hughes (2004). According to

[2] Though we do not discuss phonetic and phonological annotations, we consider these equally important alongside other data types. For a survey of such types of annotation, see Bird & Liberman (2000).

this work, paradigms are perhaps the most pervasive linguistic data type found in the literature. The underlying model for any paradigm includes an ordered set of forms that show some contrast or systematic variation. This is summed up in the following working definition from Bird (1999) and extended by (Penton et al. 2004):

> "...'a paradigm (broadly construed) is any kind of rational tabulation of words or phrases to illustrate contrasts and systematic variation.' This definition needs to be extended to include content below the level of the word, such as phones or morphs" (Penton et al. 2004, p. 1)

Penton et al. (2004, p. 6) also point out that "...linguistic paradigms simply represent an association between linguistic forms and linguistic categories." Important in paradigms, then, is the listing of specific features, construction types, or meanings with which to order the illustrative forms. The generalization that is put forward is as follows:

> "...let $D_0 \ldots D_n$ be a set of linguistic properties (or domains). Then a paradigm is a function: $f : D_1 \times \ldots \times D_n \longrightarrow D_0$" (Penton et al. 2004, p. 6).

That is, while paradigms are usually presented in tabular format in print materials, Penton et al. (2004) propose the above underlying structure, meant to describe the information in most paradigms surveyed in the literature.

2.3 Dictionaries, lexicons, etc.

We now consider the most widespread type of linguistic resource: the dictionary. Dictionaries and their accompanying entries are perhaps the most codified of the data types under discussion, considering that there are fields dedicated to their study, namely lexicography and lexicology. Though the contents of dictionary entries vary widely, there are some general consistencies that can be identified. In their survey of various print dictionaries, for example, Bell & Bird (2000) show that a general model for a dictionary entry can be achieved. The body of an entry contains: pronunciation information, usually in the form of a phonetic transcription; morphosyntactic information (syntactic categories, features, etc.); sense information in the form of a definition, semantic realm, or semantic features; mapping information that provide ordering to the set of lexemes; and finally, optional miscellaneous information concerning, for example, "etymology, obsolescence, cross-references, register, informant identity ...". Here we simplify the results from Bell & Bird (2000) for the body of dictionary entry:

$$Body = \{Pron, MSI, Sense, Mapping, (Aux)\}$$

It is clear even from this survey of print resources that a dictionary entry can contain informtation of a much more varied and open-ended type as compared to the other data types reviewed thus far.

Expanding the discussion now to include a broader collection of resources, we cite Calzolari, Grishman & Palmer (2001) who have conducted a survey of existing electronic lexical resources including, among other things, machine-readable dictionaries and computational lexicons. Machine readable dictionaries are essentially electronic versions of print dictionaries, but "...lack an explicit representation of linguistic information such as inflectional class, obligatory complements, alternations, regular polysemy, etc" (Calzolari et al. 2001, p. 229). Computational lexicons on the other hand contain "...explicit morphosyntactic, syntactic and semantic knowledge, partly through an extensive work of extraction from corpora" and are mostly monolingual, though "founded on well-established theoretical frameworks" (Calzolari et al. 2001, p. 229). What perhaps has the potential to set these electronic resources apart from their print counterparts is (1) the inclusion of rich semantic information, for example, "Reference to an ontology of types which are used to classify word senses ..." and "[d]ifferent types of relations (e.g. synonymy, antonymy, meronymy, hypernymy, Qualia Roles, etc.) between word senses, etc." (Calzolari et al. 2001, p. 18). This research confirms the findings of Bell & Bird (2000) but also shows that these electronic resources may go beyond even the most complex print dictionaries.

2.4 Treebanks

Treebanks are data structures containing rich syntactic information. For instance, the Penn Treebank (Marcus, Santorini & Marcinkiewicz 1994) contains information on tokenization, part of speech, constituency, and syntactic function. Furthermore, other, more subtle syntactic information can be encoded such as trace information produced by movement operations (Cotton & Bird 2002, p. 2). Some treebanks are designed to show dependency relations among syntactic elements, e.g., the Prague Dependency Treebank (Hajič, Böhmová, Hajičová & Vidová-Hladká 2000). Beyond syntactic information, a number of treebanks also include information on morphological categories. For instance, various HPSG-based treebanks show explicit information concerning morphological and syntactic features, e.g., in the BulTreeBank (Simov, Popova & Osenova 2001) for Bulgarian. Also, treebanks may be enriched with semantic information, such as topic and focus in the Prague Dependency Treebank (Hajič et al. 2000, p. 15) or predicate-argument structure and semantic role information in the Susanne corpus (Sampson 1995). Finally, aimed at providing deeper syntactic and semantic annotation of the Penn Treebank, the PropBank project (Kingsbury & Palmer 2002) also contains predicate-argument information, but adds specific semantic markup of verb modifiers, e.g., directional, locative, or manner elements.

3 Accommodating the fundamental data types

Now that we have reviewed the fundamental linguistic data types, it should be clear that the data types overlap significantly with one another in terms of their information content. For instance, dictionaries may contain substantial morphological information on the headword, for instance its syntactic category (cf. Treebanks) or its morphological features (cf. IGT). On the other hand, morphological markup, in the form of IGT, contains a significant amount of lexical information—enough, perhaps, to create a dictionary, provided there were an adequate number of lexical item represented in the IGT instances. Then, there are morphosyntactic paradigms that contain morphosyntactic feature names and values, the information content of which overlaps with that of IGT, namely, feature values. Furthermore, throughout all of these data types, the most basic entity that shows up again and again is a transcription of linguistic form. Form comes in the guise of the headword in a dictionary entry, the contents of the cell in a paradigm, and the elements in the first line of IGT. Because of this overlap, it seems quite reasonable to reuse as much material as possible to arrive at an underlying, general model.

Discovering the generalities expressed by the data types requires being very specific about the type of linguistic object that is being represented: This is precisely what the developers of GOLD have intended by creating a markup ontology. Thus, identifying the linguistic object in each data type is an ontological issue. But instead of delving into an in-depth ontological discussion, we take a more practical approach in developing the model. Our aim, then, is similar to that of some computer scientists who model linguistic data:

> "In particular, we think that there should exist some features common for all the linguistic objects, and this set of features should determine the base object linguistic object hierarchy. This abstract object should not belong to any of the traditional linguistic levels but instead should organically unify them" Sidorov & Gelbukh (1999, p. 2).

From an ontological standpoint, one of the most basic questions to ask is whether an element of annotation is relational. A phonetic transcription, for example, is not considered relational: it is a first order representation of the segmental aspect of raw data. Consider, though, a headword in a bilingual dictionary entry and the associated translation. There is an implied relation between the headword and the translation. Morphological annotations and treebanks, by definition, contain implied morphological and syntactic constituency relations between explicitly represented grammatical elements. Once the basic distinction between relational and non-relational elements is made clear, it is also important to keep in mind the classification of other entity types. For instance, the parts of speech (noun, verb, adjective, etc.) are not the same kinds of entities as grammatical categories (case, tense, number, etc.). But even more fundamentally, there should be a strict delineation between, for example, semantic concepts and grammatical concepts.

Essentially, we need a way to combine the variety of data objects represented in the fundamental data types. It is tempting to create an arbitrarily complex data type whose contents subsume all the elements of the fundamental types. However, any general model should be compatible with linguistic theory and not be an *ad hoc* collection of data objects—in as much as this is possible. A solution is to use the notion of the **linguistic sign** (de Saussure 1959/1915, Hjelmslev 1953) as the basis of our data model. Though direct discussion of the linguistic sign is not usually considered a current topic in linguistics, the nature of the sign is still somewhat controversial, cf. Hervey (1979). Therefore, we include here a brief discussion of the basics of our approach to the sign. A linguistic sign is a 3-tuple $\langle F, M, G \rangle$ consisting of a form component F, a meaning component M, and a grammatical component G. For each linguistic sign, there must be some language L to which the sign belongs. We define linguistic form F as any annotation entity that represents the phonetic, phonological, orthographic, or otherwise physical manifestation of the sign (e.g., transcription of hand shape for a sign language). As for the meaning component M, this represents the concept which the signs signifies. By meaning component, we refer specifically to semantic units or features of semantic units, e.g., the concept DOG or the feature [+Animate]. We do not include in M annotation entities such as the definitions of lexical items or the translations of headwords. While definitions and translations do provide additional semantic annotation, they are essentially shortcuts that rely on form components of other signs. We consider such information as auxiliary to the sign. If the meaning component is annotated, as it is sometimes in dictionaries or instances of IGT, then the units come from an ontology of (possibly language independent) concepts. Finally, the grammatical component G refers to the morphological or syntactic characteristics of the sign. Included here are categories such as the part of speech and morphosyntactic features and values. As an example of a possible XML serialization of this model, consider the following:

```
<sign lang="esp">
   <form trans="phon">casa</form>
   <grammar>
      <feature name="gender" value="feminine"/>
   </grammar>
   <meaning category="house">
      <feature name="animacy" value"inanimate"/>
   </meaning>

   <translation>
      <sign lang="eng">
         <form trans="ortho">house; a dwelling</form>
      </sign>
   </translation>

</sign>
```

Notice that the translation element introduces a relation between two signs. Whereas the form component of the sign nearly always gets included in annotations, it is possible, and quite likely, that one or more of the other components will be missing from a given annotation. The meaning component, as we have defined it, rarely gets annotated. Even in instances of IGT, the gloss line contains linguistic forms from a language scholarship. Dictionary entries may not contain any morphosyntactic information. Thus, while we consider it best practice to include all aspects of the sign, it is not always possible in reality, especially considering that most descriptive projects are partial and incomplete.

Turning to the opposite problem, there is usually more information in annotated data than just the linguistic sign. We have already mentioned translations and etymology, but the list is quite open-ended. Consider that a dictionary entry is one of the most heterogeneous of the data types. It may contain additional information such as semantic realm (e.g., botany), register (e.g., colloquial), and information regarding the speaker (e.g., age=35). We recommend not requiring such information be present with the sign, as with the *translation* element in the above example. Instead, we suggest creating relations for such auxiliary information which may be linked to the sign. Note, we are focusing on content only, trying to delineate pure linguistic from auxiliary information. In the next section, we present a more specific discussion of content.

4 Display- versus content-oriented markup

In the survey of data types presented in § 2, we emphasized how a general model for annotated data must make certain commitments as to **content**. By content, we simply refer to all elements that can be considered linguistic data or annotation. We contrast elements of content with those of **display**, or those entities that pertain to how data and annotation is to appear on the page. To illustrate the difference, consider two types of markup elements in HTML. The first type includes tags for unordered lists *ul*, list items *li*, and table data *td*. The second includes tags for italics *i*, bold *b*, and for line breaks *br*. Whereas the tags in the first group act more like containers for structuring data, the tags in the second control how the data is displayed on the page. Of course the first group also determines how the data are to be displayed, but the second is solely for display.

Consider XML markup, our central concern, which provides as a very general (tree-like) structure for encoding all kinds of data. It provides the ability to specify type and token information and various relationships among data. As such, XML is not intended to be a display-centric format; rather, it is a format that also allows explicit structure. It is tempting to use XML for encoding display information. However, little is actually gained by encoding display concepts at the level of abstraction which XML was intended. For instance, consider a hypothetical markup scheme for IGT.

```
<igt>
   <line type="transcription">
   ...
   </line>
   <line type="gloss">
   ...
   </line>
   <line type="translation">
   ...
   </line>
</igt>
```

While syntactically well-formed, explicitly encoding something like *line* merely adds to the complexity of the XML, complexity that could be moved to an accompanying stylesheet. It is actually the implicit information indicated by having lines that is important, namely, the relations holding among the transcription and the elements of the gloss. Some of the projects discussed in § 2 can be said to be at least partly based on print models. In Bell & Bird (2000), for instance, the notions of *head* and *body* make up the data model for lexical entries. But these notions are clearly a legacy from our time-honored print models. The proposal for interlinear text is somewhere in the middle, where the notions of *text*, *phrase*, *word* and *morpheme* "...are to be interpreted with reference to the common forms of interlinear display" (Bow et al. 2003, Sec 4.2). The work of Penton et al. (2004), on the other hand, seems to get away totally from a display-oriented model, as in the extract below (Penton et al. 2004, p. 14):

```
<paradigm>
   <form>
      <attribute name="caste" value="Brahmin"/>
      <attribute name="town" value="Dharwar"/>
      <attribute name="morpheme" value="it is"/>
      <attribute name="content" value="ede"/>
   </form>
      ...
```

Rather than a strict display/non-display-oriented split, data models tend to lie on a kind of continuum between very display-oriented and very content-oriented. The inclusion of markup elements such as *cell* or *line* suggest that the data model tends towards a display orientation. The importance of this distinction may be better appreciated when comparing tradition dictionary models to computational lexicons. While traditional print dictionaries, and their electronic counterparts, are intended be accessed via the headword, lexicons such as WordNet (Fellbaum & Miller 1998) were designed to be accessed either according to the form of the entry or to its associated **synset**, or its semantic classification. In modern applications, it may be quite useful to be able to sort according to *any* criterion, not just orthographic form. Thus, from the same data, it would be possible to exact either a traditional dictionary or a thesaurus sorted by meaning.

5 Adding more structure

The E-MELD and OLAC communities have set out to address the larger issues concerning digitization: accounting for authorship, data provenance, language identification, just to name a few. We think these issues have largely been solved, namely by advocating systems of metadata to be embedded within each document instance. In terms of advocating specific markup schemes, the issue is more complex. As has been argued at many E-MELD sponsored events, XML is a useful markup language for linguistic annotation because, among other reasons, it offers a more structured syntax than do other alternatives, for example, HTML or Shoebox code. One reason to have more structure is to facilitate migration which requires the interpretation of markup perhaps orthogonal to, or even at odds with, its original purpose, as summed up here:

> "When data are reused or processed outside their original context, however, the regularities exploited by the designer of the vocabulary may no longer exist, and the notation will accordingly seem hard to understand and arbitrary in its meanings. Translation to a common reference model like RDF serves to make at least some of the implicit assumptions embedded in colloquial XML vocabularies more explicit, and to make the data more easily reusable in new applications and more easily comprehensible to larger communities" (Sperberg-McQueen & Miller 2004).

In this section we turn to the specific issue of structure and XML and advocate some additions to take advantage of recent developments in markup languages, in particular, the use of RDF (Lassila & Swick 1999) and OWL (McGuinness & van Harmelen 2004).

The main advantage of using XML, rather than less structured markup languages, is that the XML may be manipulated, e.g., via XSL transformations (W3C 2001), and thus migrated to other formats suitable for specific tasks like human-oriented display, database applications, manipulation by specific programming languages, an observation summarized by Sperberg-McQueen & Miller (2004). We argued in §4 for a content-based data model over one that is display-based. It will likely turn out that creating an interoperable data model renderable in a variety of display formats is relatively straightforward, even for content-centric formats such as the one for paradigms described by Penton et al. (2004, p. 1): "The range of presentations possible for the same data set indicate that the underlying structure of the paradigm can be rendered into a variety of visual formats." The idea is that once an adequate data model is established for content interoperability, the difficult work is done, and various stylesheets can be constructed for displaying the data in a variety of ways. We now turn to the more complex issue of designing a model for migration to a semantically interoperable format.

Consider the work of Simons (2003) and Simons (2004), which we consider an excellent test case for such a migration task. Simons developed the Semantic Interpretation Language (SIL) to transform semi-structured data in XML to highly-structured data in RDF serialized as XML. The SIL is a generalized framework implemented using XML and XSL that formally maps the elements and attributes of best practice XML resources to a common **semantic schema**, vis-à-vis an ontology. The strength of the SIL is that it provides the means to manipulate the original XML at both the syntactic and the semantic level, once the semantics of the markup is defined according to a **metaschema** (Simons 2003). The metaschema is a document consisting of a set of directives in the SIL language that instructs the processor on how to interpret the original markup elements according to the concepts of semantic schema. Furthermore, the metaschema formally interprets the original markup structure by declaring what the dominance and linking relations in the XML document structure represent. We have demonstrated in Simons, Fitzsimons, Langendoen, Lewis, Farrar, Lanham, Basham & Gonzalez (2004) and Simons, Lewis, Farrar, Langendoen, Fitzsimons & Gonzalez (2004) that the migration process can be successfully implemented in a scalable, systematic fashion. However, the creation of a metaschema document is not at all straightforward. A particular challenge is determining the meaning of relationships within the document tree. For example, whereas the actual XML document tree consists of constituency relations specific to the Document Object Model (DOM), authors of XML documents often give these relations an implicit meaning. This suggests that methods such as using the SIL language can be made more transparent if such relations are encoded directly. The first structural design principle, then, is to explicitly encode the relations in the XML, and encode them as elements. Consider, for example, the following XML code from Bow et al. (2003) representing a partial instance of IGT:

```
<interlinear-text>
<item type="title">SE Text</item>
<item type="media">kalsrap.mov</item>
<item type="comment">Story from tape 20001bx ...</item>
<phrases>
    <phrase>
        <item type="gls1">We all know that place ...</item>
        <item type="gls2">Yumi evriwan isave ples ia ...</item>
        <words>
        ...
```

The problem with such a model is that it does not explicitly indicate relations so as to allow an automated tool to see them as $rel(A, B)$. For example, it may not be immediately clear as to what kind of relation holds between the South Efate text and the English text, given that there is no mention of *translation*. Consider the following revised XML:

```
<igt>
   <sign lang="erk">
      <form>Yumi evriwan isave ples ia ...</form>
      <free-translation>
         <sign lang="eng">
            <form>We all know that place ...</form>
         </sign>
      </free-translation>
   </sign>
   ...
</igt>
```

Thus, the specific translation relation is indicated and could be easily read by an annotation tool, or for that matter an RDF processor that recognizes only subject-predicate-object schemes.

But even more basic perhaps is the challenge of interpreting non-relational markup tags. Bird & Simons (2003) advocate using tags that are compatible with elements in an ontology, e.g., GOLD. In other words, "[m]ake sure that every element comes from a specific namespace," and insure that the namespace is from a recognized ontology, rather than "making up your own URIs" (DuCharme & Cowan 2002). For instance, to simplify matters, the default namespace for the XML instance document could be the ontology itself.

```
<igt xmlns:gold="http://www.linguistics-ontology.org/ns/gold/0.3/gold.owl#">
   <gold:sign lang="erk">
      <gold:form>Yumi evriwan isave ples ia ...</form>
      ...
      <gold:free-translation>
         <gold:sign lang="eng">
            <gold:form>We all know that place ...</form>
         </gold:sign>
      </gold:free-translation>
   </gold:sign>
   ...
</igt>
```

There are other recommendations that go along with the use of RDF. For instance, prefer the use of *rdf:ID* attributes to *ID* attributes common in DTDs. Whereas these attributes may seem to serve the same purpose, current RDF processors are designed only to interpret the former (DuCharme & Cowan 2002). Furthermore, there are other elements already present in RDF that a schema designer may leverage because of their recognized semantics. For instance, consider the various container elements: *rdf:Bag*, *rdf:Seq*, and *rdf:Alt*. Respectively, these elements are interpreted as unordered list, ordered list (or sequence), and alternative. After DuCharme & Cowan (2002) we propose to make use of these, for example, *rdf:Seq* to keep order sensitive entities explicit, or *rdf:Bag* to indicate that order in a collection of texts is not significant:

```
<text xmlns:gold="http://www.linguistics-ontology.org/ns/gold/0.3/gold.owl#">
   <rdf:Bag>
      <gold:sign lang="erk">
      ...
      </gold:sign>
      ...
      <gold:sign lang="erk">
      ...
      </gold:sign>
   </rdf:Bag>
</text>
```

6 Summary and future work

To summarize, we have discussed best-practice markup for language resources not only in terms of format but also in terms of content. We have argued that by including tighter control over the content of markup, migration to semantically interoperable formats can be facilitated. Furthermore we have discussed the need for such a content-based model in the design of annotation tools. To arrive at recommendations for content, we have surveyed various best-practice approaches for the fundamental data types, including linguistic paradigms, interlinear glossed text, dictionaries and lexicons, and treebanks. We then turned to a discussion of the virtues of content- over display-oriented data models. Finally, we gave a few recommendations on how to add even more structure to existing XML models by using constructs from RDF and OWL. The overall recommendations for the model are summarized here:

1. Use elements that accommodate the fundamental linguistic data types.
2. Use the linguistic sign as the most basic data concept.
3. Prefer content-oriented rather than display-oriented markup.
4. Use the existing standards that facilitate data migration and mapping to an ontology.
5. Prefer elements that are oriented towards content, rather than those oriented towards display.
6. Tie elements of XML markup to a recognized ontology by taking advantage of the namespace construct.
7. Ensure that the implied relations among elements are justified as linguistic relations.
8. Take advantage of various semantically anchored RDF constructs.

References

Aristar, A. (2003), FIELDL, Technical report, presented at the Workshop on Digitizing and Annotating Texts and Field Recordings, LSA Institute.

Bell, J. & Bird, S. (2000), 'A preliminary study of the structure of lexicon entries', Presented at the Workshop on Web-Based Language Documentation and Description.

Bickel, B., Comrie, B. & Haspelmath, M. (2004), The Leipzig Glossing Rules: Conventions for interlinear morpheme-by-morpheme glosses, Technical report, Department of Linguistics of the Max Planck Institute for Evolutionary Anthropology and the Department of Linguistics of the University of Leipzig. Revised version.

Bird, S. (1999), Multidimensional exploration of online linguistic field data, *in* P. Tamanji, M. Hirotani & N. Hall, eds, 'Proceedings of the 29th Annual Meeting of the Northeast Linguistics Society', University of Massachusetts at Amherst Press, pp. 33–47.

Bird, S. & Liberman, M. (2000), A formal framework for linguistic annotation, Technical Report MS-CIS-99-01, Computer and Information Science, University of Pennsylvania.

Bird, S. & Simons, G. (2003), 'Seven dimensions of portability for language documentation and description', *Language* **79**.

Bow, C., Hughes, B. & Bird, S. (2003), Towards a general model of interlinear text, *in* 'Proceedings of the E-MELD Language Digitization Project: Workshop on Digitizing and Annotating Texts and Field Recordings', University of Michigan.

Calzolari, N., Grishman, R. & Palmer, M. (2001), Survey of major approaches towards bilingual/multilingual lexicons, ISLE Deliverable D2.1-D3.1, ISLE Computational Lexicons Working Group, Pisa.

Cotton, S. & Bird, S. (2002), 'An integrated framework for treebanks and multilayer annotations', *CoRR* **cs.CL/0204007**.

de Saussure, F. (1959/1915), *Course in General Linguistics*, Peter Owen Ltd., London. (translated by W.Baskin).

DuCharme, B. & Cowan, J. (2002), 'Make your XML RDF-friendly', electronic publication. Available on-line at http://www.xml.com/pub/a/2002/10/30/rdf-friendly.html.

Farrar, S. & Langendoen, D. T. (2003), 'A linguistic ontology for the Semantic Web', *GLOT International* **7**(3), 97–100. www.u.arizona.edu/∼farrar/papers/FarLang03b.pdf.

Fellbaum, C. & Miller, G., eds (1998), *WordNet*, The MIT Press.

Hajič, J., Böhmová, A., Hajičová, E. & Vidová-Hladká, B. (2000), The prague dependency treebank: A three-level annotation scenario, *in* 'Treebanks: Building and Using Parsed Corpora', Amsterdam:Kluwer, pp. 103–127.

Hervey, S. (1979), *Axiomatic Semantics: A Theory of Linguistic Semantics*, Scottish Academic Press, Edinburgh.

Hjelmslev, L. (1953), *Prolegomena to a theory of language*, Indiana University Publications in Anthropology and Linguistics, Bloomington, Indiana. Translated by Francis J. Whitfield.

Kingsbury, P. & Palmer, M. (2002), From treebank to propbank, *in* 'P. Kingsbury and M. Palmer. From Treebank to PropBank. In Proceedings of the LREC, Las Palmas, Canary Islands, Spain, 2002'.

Lassila, O. & Swick, R. R. (1999), Resource description framework (rdf) model and syntax specification, Recommendation, W3C. Available at http://www.w3.org/TR/REC-rdf-syntax/.

Lewis, W. D., Farrar, S. & Langendoen, D. T. (2001), Building a knowledge base of morphosyntactic terminology, *in* 'Proceedings of the IRCS Workshop on Linguistic Databases', University of Pennsylvania, pp. 150–156. www.u.arizona.edu/∼farrar/papers/LewFarLang01.pdf.

Marcus, M. P., Santorini, B. & Marcinkiewicz, M. A. (1994), 'Building a large annotated corpus of english: The penn treebank', *Computational Linguistics* **19**(2), 313–330.

McGuinness, D. L. & van Harmelen, F. (2004), OWL web ontology language: Overview, Recommendation, W3C. Available at http://www.w3.org/TR/owl-features/.

Penton, D., Bow, C., Bird, S. & Hughes, B. (2004), Towards a general model for linguistic paradigms, *in* 'Proceedings of the EMELD'04 workshop on databases and best practice'.

Peterson, J. (2000), Cross-reference grammar project 2.0. final report, Technical report, Ludwig-Maximilians-University, Munich.

Sampson, G. R. (1995), *English for the Computer: The SUSANNE Corpus and Analytic Scheme.*, Clarendon Press.

Sidorov, G. & Gelbukh, A. (1999), A hierarchy of linguistic programming objects, *in* 'Proc. ENC'99, Segundo Encuentro de Computación', Pachuca, Hidalgo.

Simons, G. F. (2003), Developing a metaschema language to support interoperation among XML resources with different markup schemas, *in* 'Proceedings of the ACH/ALLC conference', Athens, GA.

Simons, G. F. (2004), A metaschema language for the semantic interpretation of XML markup in documents, Technical report, SIL. http://www.sil.org/∼simonsg/metaschema/sil.htm.

Simons, G. F., Fitzsimons, B., Langendoen, D. T., Lewis, W. D., Farrar, S. O., Lanham, A., Basham, R. & Gonzalez, H. (2004), A model for interoperability: XML documents as an RDF database, *in* 'Proceedings of the EMELD Workshop on Databases', Detroit, MI. www.u.arizona.edu/∼farrar/papers/Sim-etal04a.pdf.

Simons, G. F., Lewis, W. D., Farrar, S. O., Langendoen, D. T., Fitzsimons, B. & Gonzalez, H. (2004), The semantics of markup: Mapping legacy markup schemas to a common semantics, *in* 'Proceedings of the 4th workshop on NLP and XML (NLPXML-2004): held in cooperation with ACL-04', Barcelona, Spain, pp. 25–32. www.u.arizona.edu/∼farrar/papers/Sim-etal04b.pdf.

Simov, K., Popova, G. & Osenova, P. (2001), Hpsg-based syntactic treebank of bulgarian (BulTreeBank), *in* 'Proceedings of the Corpus Linguistics 2001 Conference', p. 561.

Sperberg-McQueen, C. M. & Miller, E. (2004), On mapping from colloquial XML to RDF using XSLT, *in* 'Proceedings of Extreme Markup Languages'. Available online at http://www.mulberrytech.com/Extreme/Proceedings/html/2004/Sperberg-McQueen01/EML2004Sperberg-McQueen01.html.

W3C (2001), Extensible Stylesheet Language (XSL) version 1.0, Recommendation, W3C. Available at http://www.w3.org/TR/2001/REC-xsl-20011015/.