

# Searchable Metaspaces

Steve Berman, Stefan Evert, Ulrich Heid  
IMS, University of Stuttgart  
{berman,evert,heid}@ims.uni-stuttgart.de

Draft for  
the EAGLES/ISLE Workshop  
Athens, 29/30 May 2000

## Abstract

The purpose of this presentation is to start a discussion about methodological and operational requirements for developing tools for internet browsing and/or querying of meta-descriptions of language resources, in particular multimodal corpora. Among the most important requirements are: delimiting the relationship both between meta-descriptions and the resources they apply to, and between browsing and querying over the internet; establishing a standard for representing meta-descriptions; administering the web-based availability of language resource and their accessibility via meta-descriptions; and establishing user support for query editing and data interchange. We attempt to stake out positions regarding these requirements, addressing both their advantages and disadvantages. We base our positions on the EAGLES/ISLE proposal for a meta-description standard for language resources ([1]). Our views are also influenced by work on the development of query languages for linguistic resources, such as CQP<sup>1</sup>, the MATE query language Q4M<sup>2</sup>, and the TIGER query language<sup>3</sup> for syntactic tree annotations.

## 1 Overview of Objectives

With the increasing development and use of multi-modal language resources, there is a growing need for suitable tools to access and query these resources. To facilitate these tasks, it is common—and essential—for the resources to be associated with meta-descriptions of their content (including object data annotations). Two perspectives are possible and relevant in this context:

- The local or site perspective: an institution has a (number of) multi-modal resource(s), and somebody wants to identify parts of these resources that satisfy certain meta-descriptions. Possibly one even wants to retrieve, from such resource(s), certain subsets (say turns, sentences, whole dialogues), according to a combination of metadata and linguistic (or other modality-specific) criteria annotated in the resource. The resources are accessed locally and the search is also carried out on site.
- The global or web perspective: somebody wants to know about the existence of resources of a certain kind (i.e. satisfying certain conditions in terms of meta-descriptions); if a web search engine accepts the required meta-descriptions, then the resources can be located by browsing, and possibly even accessed and queried over the web.

Although these perspectives make different demands on implementation, we will see that, from the point of view of the language resource user, they complement rather than compete with each other; hence, resource owners should accommodate both perspectives. Following the EAGLES/ISLE

---

<sup>1</sup><http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/>

<sup>2</sup><http://www.cogsci.ed.ac.uk/~dmck/MateCode/>

<sup>3</sup><http://www.coli.uni-sb.de/cl/projects/tiger/>

proposal for a meta-description standard for language resources ([1]), we concentrate on the web perspective, though for comparison we also refer to the site perspective.

The development of tools for implementing the web perspective entails a number of methodological and operational requirements. Among the most important requirements are: delimiting the relationship both between meta-descriptions and the resources they apply to, and between browsing and querying over the internet; establishing a standard for representing meta-descriptions; administering the web-based availability of language resources and their accessibility via meta-descriptions; and establishing user support for query editing and data interchange. We discuss these in turn.

## 2 Meta-Descriptions vs. Object Resources

### 2.1 The Scope of Meta-Descriptions

As [1] points out, in the context of language resources, meta-descriptions are usually distinguished from resource content by including them in the header of the resource file, or storing them in separate header files, which are linked to the resource. This division presupposes a clearcut differentiation between metadata annotations and object data annotations, or briefly, between metadata and annotations; however, in practice, this is often not the case. For example, a corpus might include tags for metalinguistic information such as speaker or background noise alongside tags for object data information such as part of speech or intonation contour.<sup>4</sup> Given this, the creation of a meta-description involves extracting those annotations from the language resource that constitute metadata.

There are two ways of associating meta-descriptions with a language resource:

- All meta-descriptions are (always and only) attached to resources as a whole.
- Certain meta-descriptions are annotated to regions in a resource (individual dialogues, turns, sentences, etc.).

On the first alternative, all meta-descriptions are in effect global, i.e. valid for the entire resource, while the second alternative admits strictly local meta-descriptions, which are valid for only a proper part of the resource. From the web perspective, it is clear that global meta-descriptions can be more efficiently browsed than strictly local meta-descriptions. In particular, global meta-descriptions can be readily stored in a header file separate from the resource (i.e. from the corpus files). However, it seems that it is in most cases necessary, or at least useful, to have local meta-descriptions. For example, in a multilingual corpus, a metadata attribute `language` will have different values at different locations in the corpus; in speech corpora, there will usually be one or more attributes identifying the current speaker. To make these efficiently browsable, it would be necessary to create a summary of them, with some loss of information. What is at stake here is the issue of an efficient representation for meta-descriptions, which we return to below. First, however, it is useful to clarify the relationship between browsing and querying as applied to metadata.

### 2.2 Browsing vs. Querying

From the web perspective, ‘browse’ is usually taken to mean ‘navigate (i.e. explore) a hierarchical structure of document links’. If we think of LR metadescriptions as instantiating such links, this implies that a browsable metaspace forms such a hierarchical structure. Querying is more generally the (attempted) extraction of information from a set of data, whether hierarchically structured or not. When the effect of structure on the (machine as well as user) efficiency of browsing and querying over the web is considered, the following generalizations represent the current state of technology:

- Flat, independent structures can be efficiently queried over the web, e.g. by specifying logical combinations of attribute/value pairs. Flat structures cannot express hierarchical or recursive

---

<sup>4</sup>Cf. for instance the MATE annotation guidelines: <http://www.ims.uni-stuttgart.de/projekte/mate/mdag/>

relationships (which is the bread-and-butter of syntactic annotation<sup>5</sup>), but most, if not all, meta-descriptions are neither recursive, nor inherently hierarchical.

- A single hierarchy can be efficiently browsed over the web; but the information it contains is restricted to non-overlapping annotations. An attempt to merge the multitude of meta-descriptions required by various sub-communities into a single hierarchy would invariably lead to a highly artificial and complex design.
- Several, possibly overlapping, hierarchies cannot easily be browsed over the web. Even when the hierarchies are connected by hyperlinks, it does not seem possible to explore meta-descriptions from different hierarchies simultaneously. It is possible to search multiple hierarchies using an appropriate query language, but the computational efficiency of the query processor is a major issue.

In short, the arguments presented here favour an approach using queries on flat structures or multiple hierarchies. Flat structures would allow an easy implementation and efficient query execution.

## 2.3 Storing Meta-Descriptions

The choice of storage method for meta-descriptions is primarily determined by the scope of the descriptions as discussed in section 2.1. If metadata are annotated to regions within a resource, they are inextricably linked with the object data. Such meta-annotations will be stored in the same file(s) as the object data and annotations, and share their (often proprietary) encoding. From a strict site perspective (as defined in section 1), this allows queries that access meta-annotations and object annotations simultaneously. However, from a web perspective the following problems arise:

- Since most corpora use proprietary annotation and/or encoding formats, queries to resource-bound metadata would have to be delegated to various proprietary query processors. Each of those would require a mapping from the general, standardised query language to the syntax used by the query processor. Some query processors may not support the full expressive power of the standardised query language.
- It would be prohibitively expensive to search the full amount of corpus data available on the web for the requested combination of meta-annotations, rather than just scan a short header file for each corpus.
- Many linguistic resources are not public available (usually for copyright reasons). From the web perspective, it would still be desirable to locate such resources by their meta-annotations. Users would then seek to obtain permission to access the resources which meet their needs.

If, on the other hand, all meta-descriptions are attached to resources as a whole, they can be stored in separate header files. Such header files could be made accessible to the public even where access to the full data is restricted. Header files could easily be converted to a standardised format. The comparatively small size of the files would allow direct queries from a central ‘search engine’ using a standardised query language.

## 3 Representing Meta-Descriptions

### 3.1 The Structure of Meta-Descriptions

An increasingly standard markup language for language resource annotation is XML,<sup>6</sup> which is also the EAGLES/ISLE proposed standard markup language for meta-descriptions. Given the strictly hierarchical structure of XML, the question arises whether meta-descriptions in fact need such

---

<sup>5</sup>E.g. a noun phrase embedded in a prepositional phrase which again is part of a larger noun phrase.

<sup>6</sup><http://www.w3.org/XML/>

a structure.<sup>7</sup> On the ‘grove’ view of XML documents, regions (which are delimited by start and end tags) correspond to the nodes of a tree, which are annotated with attribute-value pairs. The annotations of each node have scope over all descendants of this node. Object data is connected to terminal nodes (the ‘leaves’ of the tree), possibly through hyperlinks. Thus, the question is whether the scopes of meta-descriptions form a small number of deep hierarchies (ideally, just one), or whether they form a fairly large number of shallow hierarchies (maximally, just the values themselves). The former configuration implies that the meta-descriptions are highly interrelated, while on the latter setup they are largely independent of each other.

Our impression is that existing meta-descriptions do not consist of highly structured annotation hierarchies, but rather that there are many independent attributes relevant for metadata (e.g. text type, situation, speakers’ age, social status, etc.), and that each of these attributes has a small or even trivial hierarchy of values, i.e. a list of values. Of course, one could forcibly bind the attributes into a (more or less) strict hierarchy, creating a highly structured XML tree; but there does not seem to be any inherent need for this imposed by the characteristics of meta-description annotations.<sup>8</sup> Our position is supported by the EAGLES/ISLE proposal for a meta-description standard, which points out that there are LR “sub-communities . . . who would . . . structure their data-bases around . . . references which are of little or no interest to other sub-communities” ([1, p. 3]). This observation implies that there is little benefit to be gained from imposing a single hierarchical structure on the set of metadata.

It is still possible, and in our opinion desirable, to use XML as an *encoding* for the flat metadata annotations. The hierarchical structure of XML might then be used to represent semantically motivated type hierarchies (cf. footnote 8).

## 3.2 The Representation of Global and Local Meta-Descriptions

If we adopt the view that meta-descriptions form a large number of independent, shallow hierarchies, it is obvious that the inherent hierarchical structure of XML documents is not well suited to the representation of metadata. The shallow hierarchies that do exist for some attributes may be ignored without much loss of information. Thus, we see meta-descriptions as a large number of unrelated, flat structures on the object data.

At this point let us return to the distinction between global and local meta-descriptions. Global meta-descriptions could simply be annotated as sets of attribute-value pairs to individual resources (since their values are fixed for the entire resource). As noted above, for meta-descriptions to be browsable or searchable over the web, it is most efficient for them to be stored in separate header files. Local meta-descriptions, on the other hand, could have different values associated with a given attribute at different locations of the resource.

As pointed out in section 2.3, it is desirable for various reasons to store meta-descriptions in separate header files, each of which is valid for a linguistic resource as a whole. Hence, local meta-annotations need to be extracted from the corpus and collected in what we call a meta-summary. Since we consider meta-descriptions to be completely independent of each other, the most detailed meta-summary would list all combinations of meta-annotation values that appear in the resource, e.g.

```
{gender=female, language=english, age=28}  
{gender=male, language=german, age=31}  
{gender=male, language=german, age=56}.
```

However, when there are many fine-grained, and possibly overlapping meta-annotations, this approach would produce a prohibitively large amount of data for the meta-summary. The opposite extreme would list all occurring values for each of the attributes independently:

---

<sup>7</sup>This property of XML also poses problems for object data annotations: “the strictly hierarchical nature of XML is at odds with certain aspects of linguistic (particularly speech) data. In multi-speaker dialogues, speech may overlap, and different annotation hierarchies coded on a corpus may overlap, for example prosody and syntax . . . One way to indicate this non-hierarchical structure in XML is by the use of standoff annotation . . . (where hyperlinks can refer to elements in the same or a different file)” ([2, pp. 3–4]).

<sup>8</sup>This is not to deny that they can be organised according to semantically motivated type hierarchies. For instance, it makes sense to group all attributes defining a speaker together in a type hierarchy (such as `speaker` → `language` → `dialect`), but this does not imply that `<language>` annotations need to be parent nodes of `<dialect>` annotations in an XML tree.

```
gender: {male, female}
language: {english, german}
age: {28, 31, 56},
```

losing much of the more detailed information about the resource.<sup>9</sup> This least detailed meta-summary would allow highly efficient queries and minimise the amount of data that has to be transmitted over the web.

A more flexible solution may be provided by a feature-logic approach similar to the TIGER description language ([3]). On this approach, meta-summaries would consist of Boolean expressions over feature-value pairs. Both the finest and the coarsest level of detail can be expressed in this formalism:

```
((gender=female) & (language=english) & (age=28))
| ((gender=male) & (language=german) & (age=31))
| ((gender=male) & (language=german) & (age=56))
vs.
((gender=male) | (gender=female))
& ((language=english) | (language=german))
& ((age=28) | (age=31) | (age=56)).
```

In addition, combinations of meta-attribute values which share certain features may be ‘packed’:

```
((gender=female) & (language=english) & (age=28))
| ((gender=male) & (language=german) & ((age=31) | (age=56))),
```

reducing the size of the meta-summary files. It is even possible to ‘abstract’ over some of the attributes. For instance, if the exact combinations of gender and language are deemed important, but age is seen rather as an independent attribute, we would obtain an intermediate level of detail in the meta-summary:

```
( ((gender=female) & (language=english))
| ((gender=male) & (language=german)) )
& ((age=28) | (age=31) | (age=56)).
```

However, the technical side of generating and querying such packed or ‘abstracted’ meta-summaries raises complex problems and would have to be investigated thoroughly ([3] might provide a useful starting point).

### 3.3 Accommodating the Web and the Site Perspectives

We have argued in the preceding sections that meta-descriptions cannot conveniently be represented in a single hierarchy, and that there should be a separate, moderately sized, and publicly accessible meta-summary header file for each resource. However, since some types of meta-description are local (in the sense defined in section 2.1), it is necessary from the web perspective to generate (global) meta-summaries of the local metadata.

From the site perspective, however, it is desirable to have access to the local meta-descriptions and to be able to formulate integrated queries using both object annotations and meta-annotations. Ideally, the web perspective and the site perspective should be integrated into a common query language working on both the meta-annotation and the object annotation level, with web-based access to the full resource data (possibly password-protected). A similar task was tackled by the ELAN project,<sup>10</sup> but the difficulties encountered in the design of a ‘smallest common denominator’ query language suggest that there is little hope to reconcile the much more varied demands covered by the ISLE proposal.

To accommodate both perspectives, then, the following model seems plausible to us:

---

<sup>9</sup>For instance, a user looking for text from female German speakers would be presented with the above resource, even though it contains text from male German and female English speakers only.

<sup>10</sup><http://solaris3.ids-mannheim.de/elan/>

- Resources are annotated at the site-specific level of granularity with site-specific annotation schemes. Encodings may be proprietary, and the resources can be searched using site-specific tools. If intended by the resource developer, the site-specific query tool may allow integrated queries including constraints on both object annotations and meta-descriptions.

This would satisfy the site perspective.

- The same resources would in addition be described in terms of the relevant meta-description criteria, in separate meta-summary files attached to the resources. The summaries would have to adhere to a standardised annotation scheme, and use a standardised encoding format, allowing web-based search or browsing. Summaries include meta-annotations extracted from the proprietary annotation schemes and/or from existing header files, and mapped to the proposed meta-description standard.

This would satisfy the web perspective.

In such a scenario, we see the objectives of the ISLE/NIMM work regarding the representation of meta-description annotations as follows:

- to design a common meta-description scheme for multi-modal resources, which can serve as a smallest common denominator for many proprietary schemes; i.e. to design a meta-description standard
- to design the annotation scheme in terms of attribute/value lists for web-browsable meta-summaries, which are stored in separate files linked with the actual object resources.

A core/extension design as suggested in [1, sec. 6.3] seems to be appropriate, consisting of

- a core set of required attributes and values;
- an extension for each sub-community involved, defined by a committee appointed by the sub-community concerned;
- further site-specific, non-standardised extensions covering the specific demands of individual sites or resources.

## 4 Operational and Administrative Considerations

Users interested in multi-modal resources would have to operate in two steps:

1. Use a specialised web search engine, which accepts queries on meta-descriptions, to identify the subset of available resources world-wide which satisfy their specific needs in terms of meta-descriptions. This corresponds to the web perspective.
2. a) If necessary, obtain permission from site-owners to access non-public resources.  
 b) Extract the relevant object data from the resources selected in step 1 by formulating queries using site-specific, proprietary tools. This would either have to be done on-site, or through a web-based interface provided by the resource owner. Those queries will usually access object annotations and meta-annotations simultaneously. This corresponds to the site perspective.

Since this is a sequential procedure, we see that the two perspectives are complementary, rather than in competition. Thus, it is incumbent on resource owners, if they want to provide users with the most convenient and efficient service, and at the same time make their resources maximally accessible, to implement both perspectives for their resources, and, if possible, to allow remote access to site-specific query tools.

Administrative considerations that will have to be addressed include the following:

- Mappings from proprietary annotation schemes to the core standard and the extended sub-community standards have to be defined.

- Mainly for efficiency reasons, web-based search or browsing should be limited to relevant sites. As is the case in the field of legal information extraction, a specialised search engine could be used which has access to a list of registered sites that provide multimodal resources together with meta-summaries in the standardised format. Upon registration, all meta-summaries at a given site could be checked for conformance to the ISLE standard.

## 5 User Support

### 5.1 Generating and Editing Meta-Summaries

Meta-summaries should be encoded in a human-readable and easily manageable format such as XML. Since most existing language resources will already contain some amount of meta-information, configurable tools should be provided for mapping proprietary header files and annotation schemes to the standard format. It has to be investigated whether such tools could be based on existing software (such as XSL,<sup>11</sup> OML's XML Metadata Interchange,<sup>12</sup> or perhaps the Atlas Interchange Format<sup>13</sup>).

It is unlikely that a general 'abstracting' tool can be designed which generates meta-summaries from the multitude of proprietary encoding formats used. However, site owners could be assisted by tools that compile lists of attribute value combinations into meta-summaries and map site-specific attributes to the meta-descriptions allowed by the standard. Resource providers would then only have to extract a list of meta-attribute value combinations from their resources and define a mapping from their site-specific annotation schemes to the (core or extended) standard.

Tools for editing meta-description header files or meta-summaries are needed as well, both for manual corrections and to fill in attributes required by the core standard. Header files for future resources may be written in the standard format using this editor tool.

### 5.2 Queries

Tools should be provided which help users to formulate (syntactically and semantically) consistent queries and locate resources matching the specified conditions. The query syntax could use Boolean expressions over conditions on attribute values. One possible scenario provides one or more public search engine(s) that can be accessed over the web. Alternatively, a query tool could be installed locally and scan meta-summary files from a list of registered resource provider sites.

Some kind of resource 'browsing' might be achieved with incremental queries. Starting from very general selection criteria, lists of matching resources are shown and can be refined by adding further conditions. A specialised 'browser' tool might automatically suggest constraints that quickly cull the result list.

## References

- [1] P. Wittenburg, D. Broeder, B. Sloman. *EAGLES/ISLE. A Proposal for a Meta Description Standard for Language Resources*. White Paper, 8. draft version. Nijmegen: MPI, 2000.
- [2] A. Isard, D. McKelvie, A. Mengel, M. B. Møller, M. Grosse, M. V. Olsen. *Data Structures and APIs for the MATE Workbench*. MATE Deliverable D3.2, 2000.
- [3] W. Lezius, E. König. *The TIGER Language. A Description Language for Syntax Graphs*. Manuscript, IMS Stuttgart, 2000.  
<http://www.ims.uni-stuttgart.de/~esther/Papers.html>

---

<sup>11</sup><http://www.w3.org/Style/XSL/>

<sup>12</sup>See <http://www.omg.org>.

<sup>13</sup><ftp://ftp.cis.upenn.edu/pub/sb/papers/lrec00-atlas/lrec00-atlas.html>