LREC 2008 Workshop

Sustainability of Language Resources and Tools for Natural Language Processing

PROCEEDINGS

Edited by

Andreas Witt, Georg Rehm, Thomas Schmidt,
Khalid Choukri, Lou Burnard

May 31, 2008

LREC 2008 Workshop

Proceedings of the LREC 2008 Workshop
"Sustainability of Language Resources and Tools for Natural Language Processing"

Edited by Andreas Witt (Tübingen, Germany), Georg Rehm (Tübingen, Germany), Thomas Schmidt (Hamburg, Germany), Khalid Choukri (Paris, France), Lou Burnard (Oxford, UK)

May 31, 2008

Organisers

- Lou Burnard, Oxford University
- Khalid Choukri, ELRA/ELDA
- Georg Rehm, University of Tübingen
- Thomas Schmidt, University of Hamburg
- Andreas Witt, University of Tübingen

Programme Committee

- Helen Aristar-Dry, Eastern Michigan University, USA
- Jeannine Beeken, Instituut voor Nederlandse Lexicologie, The Netherlands
- Jean Carletta, University of Edinburgh, School of Informatics, UK
- Dan Cristea, University of Iasi, Romania
- Stefanie Dipper, Bochum University, Germany
- Jost Gippert, Johann-Wolfgang-Goethe-Universität Frankfurt, Germany
- Erhard Hinrichs, Tübingen University, Germany
- Marc Kupietz, Institut für Deutsche Sprache Mannheim, Germany
- Sandra Kübler, Indiana University, USA
- D. Terence Langendoen, NSF, USA
- Joakim Nivre, Växjö University & Uppsala University, Sweden
- Massimo Poesio, University of Trento, Italy
- Kiril Ribarov, Charles University Prague, Czech Republic
- Laurent Romary, Max-Planck Digital Library, Germany
- Hinrich Schuetze, Stuttgart University, Germany
- Serge Sharoff, University of Leeds, UK
- Gary F. Simons, SIL International, USA
- Manfred Stede, Potsdam University, Germany
- Simone Teufel, University of Cambridge, Computer Laboratory, UK
- Peter Wittenburg, MPI for Psycholinguistics, Nijmegen, The Netherlands
- Martin Wynne, Oxford Text Archive, UK
- Heike Zinsmeister, University of Konstanz, Germany

Preface

One of the problems in Natural Language Processing and related fields is that the sustainability of language resources such as, for example, corpora, and language technology tools (e. g., annotation or query tools) are neglected on a regular basis. This results in, for example, tools whose algorithms and data structures are poorly documented and whose area of application is evident only to the people who built the software. Similar issues arise with regard to language resources: often, these are tailored to the needs of an individual application or to a project with a very specific research question. When the project is finished it becomes next to impossible (especially for third parties) to gain access to the resource that may have taken several months or even years to create.

The very complex question of how to ensure or maybe even guarantee sustainability is related to several key issues spanning a broad spectrum across several closely related fields: in the area of language documentation, seven dimensions of portability (content, format, discovery, access, citation, preservation, rights) have been suggested. Another area of research is primarily concerned with annotation technology, especially the problem of building generic annotation frameworks as well as representing several different layers of linguistic annotation referring to one specific set of primary data by means of standoff annotation. Closely related work deals with the standardisation of annotation frameworks, especially with regard to the level of impact a specific linguistic theory has on vocabularies and markup grammars. Another area is concerned with providing sustainability primarily through specific software engineering processes for Computational Linguistics and NLP tools, applications and resources.

Increased sustainability for linguistic tools and language resources becomes more and more important for the research community. Meanwhile, even funding organisations recognise this fact and the underlying problems – they often encourage research projects to make sure that language resources will be accessible and (re-)usable in ten, 15, or 20 years time.

The challenge of ensuring sustainability is a multi-faceted one and depends on several subtasks. This workshop is the first that is especially devoted to the "sustainability of language resources and tools for Natural Language Processing" – it addresses some of the abovementioned subtasks.

In addition to the papers presented in these proceedings we invited several researchers to report on their ongoing work. As a consequence, not all of the presentations listed in the programme could be included in these proceedings. Additional materials related to the workshop are available online.

A. Witt, G. Rehm, T. Schmidt, K. Choukri, L. Burnard May 2008

Programme

09.00 – 09.10 Introduction

09.10 – 09.50 Peter Wittenburg
Data Preservation of Linguistic Resources from the 90ies (invited talk)

09.50 – 10.30 Lou Burnard
TEI P5: Conformance and sustainability issues (invited talk)

11.00 – 11.45 Dan Cristea, Ionut Pistol
Managing language resources and tools using a hierarchy of annotation schemas

11.45 – 12.30 Menzo Windhouwera, Alexis Dimitriadisa
Sustainable operability: Keeping complex resources alive

12.30 – 12.45 Khalid Choukri
Sustainability at ELRA (invited report)

12.45 – 13.00 Erhard Hinrichs
Sustainability: Bringing researchers and funding agencies together (invited report)

14.30 – 15.15 Marie-Hélène Lay, Marie-Luce Demonet
Sustainability and sharability of the Humanist Virtual Library
(Bibliothèques Virtuelles Humanistes, BVH): experiment feed-back

15.15 – 16.00 Jan-Philipp Soehn, Heike Zinsmeister, Georg Rehm
Requirements of a User-Friendly, General-Purpose Corpus Query Interface

16.30 – 17.15 Maik Stührenberg, Michael Beißwenger, Kai-Uwe Kühnberger, Harald Lüngen,
Alexander Mehler, Dieter Metzing, Uwe Mönnich
Sustainability of Text-Technological Resources

17.15 Discussion

Table of Contents

Managing Language Resources and Tools Using a Hierarchy of Annotation Schemas
Dan Cristea, Ionut Pistol 1

Sustainable Operability: Keeping Complex Resources Alive
Menzo Windhouwera, Alexis Dimitriadisa 9

Sustainability and Sharability of the Humanist Virtual Library (BVH): Experiment Feed-back
Marie-Hélène Lay, Marie-Luce Demonet 19

Requirements of a User-Friendly, General-Purpose Corpus Query Interface
Jan-Philipp Soehn, Heike Zinsmeister, Georg Rehm 27

Sustainability of Text-Technological Resources
Maik Stührenberg, Michael Beißwenger, Kai-Uwe Kühnberger,
Harald Lüngen, Alexander Mehler, Dieter Metzing, Uwe Mönnich 33

Managing Language Resources and Tools using a Hierarchy of Annotation Schemas

Dan Cristea
Faculty of Computer Science, University "Al. I. Cuza" of Iaşi, Romania
Institute for Computer Science, Romanian Academy, Iaşi, Romania
dcristea@info.uaic.ro

Ionut Cristian Pistol
Faculty of Computer Science, University "Al. I. Cuza" of Iaşi, Romania
ipistol@info.uaic.ro

Abstract

This paper describes the concept and usage of ALPE (Automated Linguistic Processing Environment) a system designed to facilitate the management and deployment of large and dynamic collections of linguistic resources and tools. ALPE can build linguistic processing chains involving the annotation formats and the tools integrated into a hierarchical structure. The particularities and advantages of integrating ALPE in a project involving the development and usage of multiple linguistic resources are the main topics of this paper.

1. Introduction

Making sure that corpora, resources and tools are reusable in different contexts than that of the originating project is one of the recent main topics of interest in the Natural Language Processing community. Re-using a resource initially developed for a specific project usually fails for one of two reasons: either the resource is not enough documented (the format is not known to the re-user), or the resource is not directly accessible (the location of the resource is not known to the re-user). Making sure a project's results are well organized and accessible ensures a better impact and a longer lasting significance, as more people will be able to use the developed resources and tools.

One of the latest developments in NLP, and one which promises to have a significant impact for future linguistic processing systems, is the emerging of linguistic annotation meta-systems, which make use of existing processing tools and implement some sort of processing architecture, pipelined or otherwise.

In this paper we describe ALPE, a system offering a new perspective to the task of exploiting NLP meta-systems, by helping a community of users to have an integrated look at a whole range of tools that are able to communicate on the basis of common formats.

For annotated linguistic resources several standardization efforts have been made, such as XCES [1] and TEI [2]. However, the proposed standardizations are not universally accepted, most research projects developing resources according to their own described formats. More recent developments, such as GOLD[3], propose unification methods for the various annotation formats. Due to such methods one can easily transform the name space of a corpus in order to make it compatible to her/his own targets. Several systems tried to facilitate the access to existing processing tools and to ease their usage. The more prominent ones are GATE [4] and UIMA [5]. Both systems make easier the access to a set of independently developed NLP tools which are already parts of an environment offering means to create and use processing chains intended to add linguistic metadata to an input corpus. GATE (Cunningham et al., 2002, Cunningham et al., 2003) is a versatile environment for building and deploying NLP software and resources, allowing for the integration of a large amount of built-ins in new processing pipelines that receive as input a single document or corpus. UIMA (Ferrucci and Lally, 2004) offers the same general functionalities as GATE, but once a processing module is integrated in UIMA it can be used in any further chains without any modifications (GATE requires wrappers to be written to allow two new modules to be connected in a chain). Since the appearance of UIMA, the GATE developers have made available a module that allows GATE and UIMA processing modules to be interchangeable, basically merging the "pool" of modules available.

ALPE, a new NLP meta-system still in development, allows a user, even with very limited programming capabilities, to automatically exploit already walked-on processing paths or to configure new ones on-the-spot, by exploiting the annotation schemas at intermediate steps. ALPE is based on the hierarchy of annotation schemas described in (Cristea and Butnariu, 2004). In this model, XML annotation schemas are nodes in a directed acyclic graph, and the hierarchical links are subsumption relations between schemas. In (Cristea et al., 2006) is described how the graph may be augmented with processing power by marking edges linking parent nodes to daughter nodes with processors, each realising an elementary NLP step.

Section two of this paper presents the theory behind the ALPE system, and section three describes the significant features of ALPE, relevant in the context of a large scale research project, employing multiple layers of annotation schemas and various tools. Section four makes a brief comparison between ALPE and the two most prominent NLP meta-systems (GATE and UIMA). The conclusions, as well as the further planned developments are described in section five.

2. The Underlying Model

2.1 Linguistic Metadata Organised in a Hierarchy

We base our model on the direct acyclic graph (DAG) described in (Cristea and Butnariu, 2002), which

[1] www.xml-ces.org/
[2] www.tei-c.org/
[3] http://www.linguistics-ontology.org/gold.html
[4] http://www.gate.ac.uk/
[5] www.research.ibm.com/UIMA/

configures the metadata of linguistic annotation in a hierarchy of XML schemas. Nodes of the graph are distinct XML annotation schemas, while edges are hierarchical relations between schemas. By interacting with the graph, a user can modify it from an initial trivial shape, which includes just one empty annotation schema, up to a huge graph accommodating a diversity of annotation and processing needs. If there is an oriented edge linking a node A with a node B in the hierarchy (we will say also that A subsumes B or that B is a descendant of A) then the following conditions hold simultaneously:

- any tag-name of A is also in B;
- any attribute in the list of attributes of a tag-name in A is also in the list of attributes of the same tag-name of B.

As such, a hierarchical relation between a node A and one descendant B describes B as an annotation schema which is more informative than A. In general, either B has at least one tag-name which is not in A, and/or there is at least one tag-name in B such that at least one attribute in its list of attributes is not in the list of attributes of the homonymous tag-name in A. We will agree to use the term *path* in this DAG with its meaning from the support graph, i.e. a path between the nodes A and B in the graph is the sequence of adjacent edges, irrespective of their orientation, which links nodes A and B. As we will see later, the way this graph is being built triggers its property of being connected. This means that, if edges are seen undirected, there is always at least one path linking any two nodes.

2.2 The Hierarchy Augmented with Processing Power

In NLP, the needs for reusability of modules and the language and application independence impose the reuse of specific modules in configurable architectures. In order for the modules to be interconnectable, their inputs and outputs must observe the constraints expressed as XML schemas.

When processes are placed on the edges of the graph of linguistic metadata, the hierarchy of annotation schemas becomes a graph of interconnecting modules. More precisely, if a node A is placed above a node B in the hierarchy, there should be a process which takes as input a file observing the restrictions imposed by the schema A and produces as output a file observing the restrictions imposed by the schema B.

In (Cristea et al., 2006) a graph (or hierarchy) of annotation schemas on which processing modules have been marked on edges is called augmented with processing power (or simply, augmented). The null process, marked Ø, is a module that leaves an input file unmodified.

2.3 Building the Hierarchy

Three hierarchy building operations are introduced in (Cristea et al., 2006): initialize-graph, classify-file and integrate-process. In this section we briefly present them.

The *initialize-hierarchy* operation receives no input and outputs a trivial hierarchy formed by a ROOT node (representing the empty annotation schema). Once the graph is initialised, its nodes and edges are contributed by classifying documents in the hierarchy or manually.

The *classify-file* operation takes an existing hierarchy and a document marked with an XML metadata and classifies the schema of the document within the hierarchy. The operation results in a (possibly) updated hierarchy and the location of the input schema as a node of the hierarchy. If the input document fully complies with a schema described by a node of the hierarchy, the latter remains unchanged and the output indicates this found node; otherwise a new node, corresponding to the annotation schema of the input document, is inserted in the proper place within the hierarchy.

Integrate-process is an operation aiming to properly attach processes to the edges of a hierarchy of annotation schemas, mainly by labelling edges with processors, but also by adding nodes and edges and labelling the connecting edges.

Apart from these basic operations that allow building a hierarchy from scratch or modifying an existing one by exploiting the annotation incorporated in files, a graphical interface allows the user to also define new nodes manually, which ALPE will place at proper places in the hierarchy automatically. But building a hierarchy can be made independent of any explicit interaction with the system by a user. It is still not unusual that an interaction results also in an augmentation of an existing hierarchy with nodes, corresponding to user's input and/or output file. Through multiple interactions, an initial minimal hierarchy which is accessed by a community of users can thus be developed.

2.4 Operations on the Augmented Graph

Three main operations can be supported by the Cristea et al. (Cristea et al., 2006) model.

If an edge linking a node *A* to a node *B* (therefore *B* being a descendant of *A*) is marked with a process *p*, it is said that ***A* pipelines to *B* by *p***. Equally, when a file corresponding to the schema *A* is pipelined to *B* by *p*, it will be transformed by the process *p* onto a file that corresponds to the restrictions imposed by the schema *B*. This arises in augmenting the annotation of the input file (observing the restrictions of the schema *A*) with new information, as described by schema *B*.

For any two nodes *A* and *B* of the graph, such that *B* is a descendant of *A*, it is said that ***B* can be simplified to *A***. When a file corresponding to the schema *B* is simplified to *A*, it will lose all annotations except those imposed by the schema *A*. Practically, a simplification is the opposite of a (series of) pipeline(s) operation(s).

The **merge** operation can be defined in nodes pointed by more than one edge on the hierarchical graph. It is not unusual that the edges pointing to the same node are labelled by empty processors. The merge operation applied to files corresponding to parent nodes combines the different annotations contributed by these nodes onto one single file corresponding to the schema of the emerging node.

With these operations, the graph augmented with processing power is useful in two ways: for goal-driven, dynamic configuration of processing architectures and for transforming metadata attached to documents. Automatic configuration of a processing architecture is a result of a navigation process within the augmented graph between a start node and a destination node, the resulted processes being combinations of branching pipelines (serial simplifications, processing and merges). In terms of processing, the difference with respect to GATE and UIMA, both allowing only pipeline processing in which the whole output of the preceding processor is given as

input to the next processor, is that in the described model the required processing may result in a combination of branching pipelines. This is due to the introduction of the merge operation which is able to combine two different annotations on the same file. Once the process is computed, then it can be applied on an input file displaying a certain metadata in order to produce an output file with the metadata changed as intended. These two files comply with the restrictions encoded by the start node and, respectively, the destination node of the hierarchy.
Since the graph is connected, there should always be at least one path connecting these two nodes. The paths found are made up of oriented edges and, depending on whether the orientation of the edges is the same as that of the path or not, we will have pipeline operations or simplification operations. A **flow** is a combination of paths between the start and the destination node that configures the processing which transforms any file observing the specifications of the start node (schema) onto a file observing the specifications of the destination node (schema).

Once the entry and exit points in the hierarchy have been determined and processing flows (combination of paths in the graph) have been devised, all the rest is done by the hierarchy augmented with the processing power in the manner described above. This way, the processing needed to arrive from the input to the output is computed by the hierarchy as sequences of serial and parallel processing steps, each of them supported in the hierarchy by means of specialized modules. Then the process itself is launched on the input file.

2.5 ALPE

ALPE is a system implementing the described model. Besides implementing all the previously described features, ALPE brings several additions.

The core modules
ALPE includes 11 core modules, used in any ALPE hierarchy (the hierarchy augmented with processing power, as described) but not attached to any edge. These core modules perform built-in tasks such as language

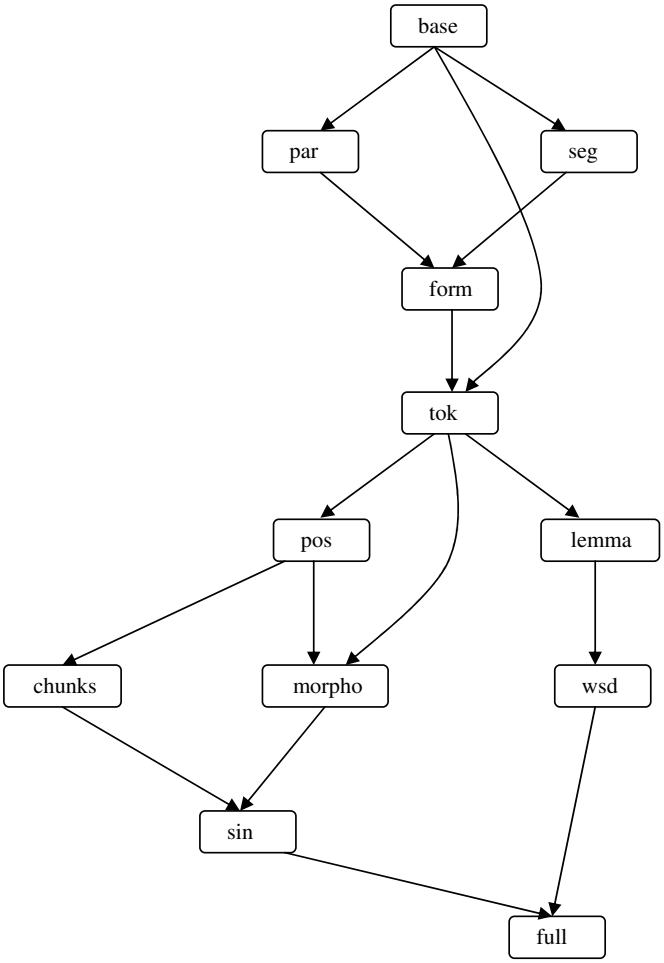

Figure 1: The ALPE core hierarchy

identification, but also implement the basic operations in the hierarchy (among others, flow computation, merging and simplifying). These core modules are used in any ALPE hierarchy and are not replaceable by user tools. They ensure that any ALPE hierarchy implements the basic behaviour, as described in this paper.

The core hierarchy

One of the main problems in developing a new NLP system is selecting a relevant and useful annotation format for the developed resources. Establishing a hierarchy of generally used XML metadata is not one of ALPE's main purposes, but having most annotated documents adhere to some common format brings obvious benefits both to the developer of new NLP software and to the user who would have an easier time finding the tools required for a particular annotation task.

As base for any new ALPE hierarchy is offered a core hierarchy, with 12 annotation schemas ranging from basic XML format to a full XCES (Ide et al., 2000) linguistic annotation specification[6]. The intermediate formats are designed to conform to specific requirements for document annotation, such as tokenization, POS-tagging, NP-chunking, etc. as well as combination of these markings. Figure 1 shows the ALPE core hierarchy. All nodes are subsets of the XCES standard for annotated data, and the subsumption relation is observed between all pairs of nodes linked through an edge.

The 12 nodes in figure 1 correspond to XML annotation schemas as follows:

- *base*: subset of XCESAna including just *cesAna* tags – corresponding to a basic XML format;
- *par*: adds the *par* tag to the parent node – corresponding to an XML with marked paragraphs;
- *seg*: adds the *s* tag to the parent node – corresponding to an XML with marked sentences;
- *form*: a merge of the subsuming formats – corresponding to an XML with marked formatting (paragraphs and sentences) information;
- *tok*: adds the *tok* and *orth* tags to the parent node – corresponding to a tokenized text;
- *pos*: adds the *ctag* tag to the parent node – corresponding to a pos-tagged text;
- *lemma*: adds the *base* tag to the parent node – corresponding to a lemmatized text;
- *chunks*: adds the *chunk* and *chunklist* tags to the parent node – corresponding to a (Noun/Verb) phrase-chunked text;
- *morpho*: adds the *msd* tag to the parent node – corresponding to an XML displaying morphological metadata;
- *wsd*: adds a *wsd* tag for semantic disambiguation;
- *sin*: merges the parent nodes – corresponding to an XML displaying full syntactic information;
- *full:* merges all parent nodes.

The purpose of the core hierarchy is to offer both a starting point to any new hierarchy as well as anchors for any new linguistic annotation formats that a user would like to include. When the XML formats of the user's input and output files are not identical with schemas belonging to the hierarchy (for instance, due to differences in the tags name space or to configurations of attributes that convey in different ways the same information) then the user has to provide convertors (wrappers) able to accommodate his notations with those corresponding to nodes of the hierarchy.

The user's needs and the selection of flows

The ALPE augmented hierarchy can be used in many ways. Suppose a user wants to process an XML file from one input format to some output format. In principle, any such processing task involves a transformation by some module capable to receive the input format and to output the required final format. The ALPE philosophy details such a processing task in relation with the pair of input-output schemas by establishing the way these schemas interrelate from the point of view of the subsumption relation. Two cases can be evidenced: either the two schemas do observe a subsumption relation or not. When they do, then the node corresponding to the input file can be connected through a direct descending or ascending edge to the one corresponding to the output file. It will be descending if the output schema results from the input schema through some adds, and it will be ascending if in order to obtain the output, simplification applied to the input are required. When the two schemas are not in a subsumption relation, then there should be a node such that either both are subsumed by it, or both subsume it.

ALPE comes with a core hierarchy whose nodes act as a grid of fixed bench-marks with respect to which the locations of the input and output schemas are set out. When the pair of users' schemas matches two nodes of the core hierarchy, then processing can be drawn in terms of known (built-in) interconnected modules. When a match (modulo, as noticed above, the XML elements name space and/or differences in configurations of attributes still conveying the same information) of one or even both of user's schemas against nodes of the hierarchy is not possible, then the non-matching schemas should be seen as new nodes of the hierarchy. In this case it is the user's responsibility to locate also the processes which will be assigned to the new edges which will interconnect the new nodes onto the hierarchy.

ALPE designs a solution to the user's problem by first computing all possible chains of edges which link the input schema to the output schema and, if needed, executing them.

Each computed flow is characterized by a set of features. These features include properties such as: flow length (defined as number of processing steps involved), cost (for instance, if processing involving one or more modules presupposes financial costs), the estimated precision of execution, and the estimated time of execution. The user can then select and run the flow most suitable to his needs.

3. Features

In this section we will describe a set of features implemented in ALPE often wished for in environments working with linguistic resources and tools. We will see how these features emerge from the model described above. Many of these features are key elements of the

[6] http://www.cs.vassar.edu/XCES/dtd/xcesAna.dtd

future European linguistic infrastructure, as seen by CLARIN[7].

Multilinguality

In modern NLP, algorithms are separated from linguistic details. This way, a module designed to perform a specific task can be put to work on any language if fuelled with appropriate language resources. This is the case, for instance, with POS-taggers (see, for instance, TNT (Brants, 2000)), which are powered by specific language models (frequency of n-grams of POS tags). A syntactic parser should be powered by the grammar of a language to be effective in parsing sentences of that language. A shallow parser, which usually implements an abstract automata machinery, could recognize noun phases of one language if powered by a resource consisting of a set of regular expressions specific to that language.

To implement multilinguality within the proposed model means to map the edges of the augmented graph on a collection of repositories of configuring resources (language models, sets of grammar rules, regular expressions, etc.) which are specific to different languages. This can be achieved if the edges of the graph labelled with processes are indexed with indices corresponding to languages. This way, to each particular language an instance of the graph can be generated, in which all edges keep one and the same index – the one corresponding to that particular language. This means that all processors of that particular language should access the configuring resources specific to that language in order for the hierarchy to work properly. For instance, in the graph instance of language *Lx*, the edge corresponding to a POS-tagger has as index *Lx*, meaning that it accesses a configuring resource file that is specific to language *Lx* (that language model).

It is a fact that different languages have different sets of processing tools developed, English being perhaps the richer, presently. Ideally, the blame for the lack of a tool in a specific language should be put on the lack of the corresponding configuring resource, once a language independent processing module is available for that task. It is also the case that differences exist in processing chains among languages. For instance one language could have a combined POS-tagger and lemmatizer while another one realizes these operations independently, pipelining a POS-tagger with a lemmatization module. These differences are reflected in particular instances of sections of the graph, which, although reproduce the same set of nodes, do not allow but for certain edges linking them. The missing edges inhibit pipelining operations

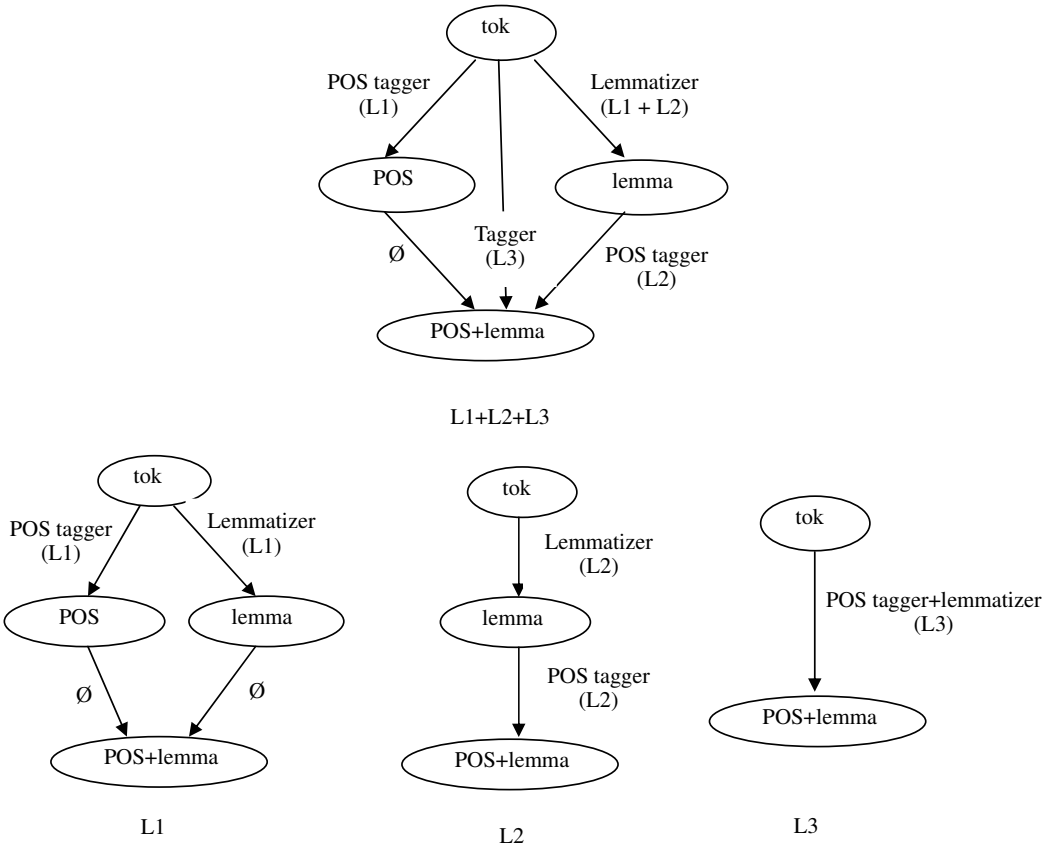

Figure 2: Computation of different flows for specific languages

[7] http://www.clarin.eu

along them, but are suited for simplification operations.
In figure 2 is given a simple example of how ALPE handles multiple languages integrated in the same hierarchy. The first hierarchy (marked as L1+L2+L3 in the figure) has four nodes (annotation schemas):

- *tok*: XML which marks lexical tokens;
- *POS*: XML marking tokens and their part-of-speech;
- *lemma*: XML marking tokens and their lemmas;
- *POS+lemma*: XML with tokens, POS and lemma information.

These four nodes correspond to simple processing stages for linguistically annotated documents. The ALPE hierarchy fragment representation (shown on the L1+L2+L3 section of Figure 2) indicates the subsuming relations between the respective nodes and the attached tools. For each tool, in parenthesis, it is indicated the languages for which the tool is available. In the sections marked L1, L2 and L3, respectively, of Figure 2 are sketched the corresponding instantiations of this sub-hierarchy for the three languages.
The user can provide an input document (XML with marked lexical tokens) and specify the required output format as being the final node (suppose *POS+lemma*). ALPE determines the language of the input document (as being L1, L2 or L3). If the input document belongs to the language L1, the computed flow will include only tools available for that language. Thus the only possible flow will use the *POS tagger* and the *Lemmatizer* tools, then merge their results into the output format. For the second language the flow will use a different *POS tagger* tool, one that requires as input a file corresponding to the *lemma* node. So the computed flow will run first the *Lemmatizer*, then the *POS tagger* on the result. For the third language, a tool is available that can directly annotate an input file in the *tok* format up to the required output.
We can look at the ALPE hierarchy as having three layers, one for each language. The three language specific hierarchies can look completely different for each language, but are still able to compute and run the same flows as the combining hierarchy. The three layers are aligned by nodes which display the same XML structure.

Manual versus automatic annotation
We have seen how automatic annotation is supported by the augmented graph. But how can manual annotation be accommodated within this approach?
Usually, in order to train processing modules in NLP, developers use manually annotated corpora. To create such corpora, they make use of annotation tools configured to help placing XML elements over a text, and to decorate them with attributes and values. As such, if annotation tools do, although in a different way, the same jobs which can be performed by processing modules, it is most convenient to associate them with edges in the graph in the same way in which processing modules are associated with these edges.
Meanwhile, it is clear that manual annotation cannot be chained in complex processing architectures in the same way in which automatic annotation can. In order to differentiate between automatic and manual processes, as encumbered by pairs of schemas observing the subsumtion relation, it results that edges should have facets, for instance AUT and MAN. Under the AUT facet of a POS-tagging edge, for instance, the automatic POS-tagger should be placed, while under the MAN facet – the POS-tagging annotation tool should be placed.
The configuration files of these tools can usually be separated from the tools themselves. We can say that the corresponding configuration files particularise the annotation tools, which label edges of the graph, in the same way in which language specific resources particularise processing modules.

IPR and cost issues
Intellectual property rights can be attached to documents and modules as access rights. Only a user whose profile corresponds to the IPR profile of a resource/tool can have access to that file/service. As a result, while computation of processing chains within the hierarchy is open to anybody, the actual access to the dynamically computed architectures could be banned to users which do not correspond to certain IPR profiles of certain component modules or resources they need.
More than that, some price policies can be easily implemented within the model. For instance, one can imagine that the computation of a flow results also in a computation of a price, depending on particular fees the chained Web servers charge for their services.
Out of this, it is also imaginable the graph as including more than one edge between the same two nodes in the hierarchy. This can happen when different modules performing the same task are reported by different contributors. When these modules charge fees for their services, it is foreseeable also an optimization calculus with respect to the overall price over the set of paths that can be computed for a required processing.

Facing the diversity of annotation styles
It is a fact that, today, a huge diversity of annotation variants circulates and is being used in diverse research communities. It is far from us to belief that a Procustean Bed policy could ever be imposed in the CL or NLP community, that would aim for a strict adoption of standards for the annotated resources. On the other hand, it is also true that efforts towards standardization are continually being made (see the TEI, XCES, ISLE, etc. initiatives). Moreover, Semantic Web, with its tremendous need for interconnection and integration of resources and applications on communicating environments, boosts vividly the appeal for standardization. It is therefore foreseeable that more and more designers will adopt recognized standards, in order to allow easy interoperability of their applications. A realistic view on the matter would bring into the focus the standards while also providing means for users to interact with the system even if they do not rigorously comply with the standards.
We have seen already that, by classification, any schema could be placed in the hierarchy. Of course, classification could increase in an uncontrollable way the number of nodes of the hierarchy. The proliferation could be caused not so much by the semantic diversity of the annotations, as by the differences in name spaces (names of tags and attributes).
Technically, this can be achieved by temporarily creating links between the new schema classified by the hierarchy, as a new node, and its corresponding schema in the hierarchy. Processing along such a link is different than

the usual behaviour associated to the edges of the graph and is specific to wrappers. It describes a translation process, in which the annotation is not enriched, but rather names of XML elements and attributes are changed. Ideally, the processing abilities of the hierarchy should include also the capability to automatically discover wrapping procedures. This task is not trivial since it would require that the hierarchy "understands" the intentions hidden behind the annotation, displaying, this way, some kind of semantic processing capabilities which is not easy to implement. However, recent initiatives as GOLD make us believe that significant steps forward in this direction are near us.

4. Evaluation

4.1 ALPE vs. GATE and UIMA

In this section we will compare functionalities of ALPE with those of GATE and UIMA, systems which can give very similar results with our.

First of all, ALPE is intended primarily to facilitate the user's interaction with the system, allowing for an programming non-expert to integrate resources and tools. As a standalone linguistic processing environment, the user is presented with a visual representation of a hierarchy of annotation formats and has basically three main choices: s/he can add a new resource to the hierarchy (for example enabling an already integrated processing module to work for another language by adding a corresponding language model), add a new processing tool (attached to an existing edge, or attached to a newly created edge) or compute and use a processing chain (providing the input file and selecting the output format). GATE offers a user interface adequate for creating and using processing chains. Chains have to be built manually and presuppose an intimate knowledge of the system. UIMA is even more oriented to the NLP professional, offering little in terms of visual user interaction. A direct comparison that would put on stage quantitative evaluations is difficult to be made for these kinds of systems. Perhaps a better prospect would be a qualitative comparison performed by a significant pool of users, providers as well as consumers of language resources and tools. In the following, we make just an estimative comparison, but a qualitative evaluation versus human performance is planned.

Every one of the three main functionalities (adding a new resource, adding a new tool, and computing and using a processing chain) is easier to perform in ALPE. Both UIMA and GATE require some formal description to be written for each new resource integrated into the system, while ALPE generates these formal descriptions automatically. When adding a new processing tool, ALPE has much more permissive restrictions with regard to what tool can be integrated: it basically has to be either a webservice or a command line, executable under Windows or Linux. GATE allows the user to integrate at least Java and Perl based tools, and this is done by writing some dedicated code, a task which is however above the capabilities of some users. UIMA is even more restrictive, allowing only C++ based tools to be integrated, and only after significant implementations and changes to the original code. However, an extension allowing modified Perl, Python and TCL modules to be integrated is available.

An evident advantage of ALPE over both GATE and UIMA is that the processing chains in ALPE are automatically computed, therefore requiring no human intervention. Moreover, they can be created between any two formats defined in the hierarchy (providing the modules decorating the connecting edges are available, otherwise there are signalled as missing). ALPE deals with multilinguality, thanks to its core module that performs language identification for each input file, then selects to corresponding tools and language resources, if available. GATE and UIMA are mainly focused on English (GATE incorporating also modules dedicated to some other languages), but the user has to make sure to select the proper modules when designing a processing chain for a document in other language than English.

Let us consider the example of a use-case in which the user has two processing tools s/he wants to use on the same input file and to merge the results in an output file. Using ALPE, this user has to specify the input/output formats of the modules, then let the system integrate the tools as arches linking the corresponding nodes in the hierarchy (in the case when one of both of these formats are not currently part of the hierarchy, they will become as such), then input the file and specify the required output format (node). Using GATE, the user has to implement the integration of the tools to make them available to the processing chain building interface, then build and run two processing chains, one for each tool, then merge the results outside GATE (since it does not allow parallel processing and merging of annotations). UIMA performs this task basically in the same way as GATE, requiring even more implementation when integrating the new tools, but allows annotation merging.

4.2 Qualitative evaluation

In order to evaluate ALPE versus human computational linguistic specialists, we have developed an ALPE augmented hierarchy configured for a current research project involving documents in 9 European languages (Bulgarian, Czech, English, German, Dutch, Maltese, Polish, Portuguese and Romanian) and using a significant number of language processing tools[8]. All documents have to be annotated according to 6 main annotation formats (and 8 optional ones), resulting a significant hierarchy of standards. This hierarchy is already implemented and serves as a management and access facility for the collected documents.

At the time of writing this paper, an ALPE core hierarchy specific to the mentioned project is implemented for English and Romanian.

5. Conclusions

We think that the model we propose and its first implementation, as the ALPE system, encapsulate different organisational, standardisation and processing features which make it interesting for the goals of a project like CLARIN.

In this proposal we have been concerned with the following features of functionality, also identified as of

[8] LT4eL – an FP6 project (www.lt4el.eu)

primary importance in CLARIN[9]:

- **unique access gate and distributivity**: although distributed in different places, LR and LT could be, in the vision described in this paper, identified through a single access gate;
- **metadata policy**: primary text and speech documents should be given the possibility to be accompanied by metadata describing human and/or automatic annotation over them. The ALPE conventions allow for the metadata to have a form which make it easily removable when the primary raw documents are needed of being recuperated;
- **independence of representation**: it is clear that the XML representation adopted by ALPE allows for LR to be manipulated in such a way as to benefit of the same treatment irrespective of the particular metadata conventions;
- **quick access**: ALPE comes very close to the objective that CLARIN LR and LT be accessed instantaneously from all over Europe;
- **conversion services**: the ALPE approach incorporates features that allows easy conversion operations from and onto different representations;
- **processing services**: the ALPE portal provides processing services for enrichment and or simplification of metadata attached to LR;
- **versioning**: the portal allows manipulation of different versions of data as well as of the metadata accompanying the texts;
- **multilinguality**: the structure allows uniform treatment of documents in different languages, as well as of parallel texts;
- **IPR issues**: the structure provides means of dealing with IPR.

In this paper we have described a model of dynamical building of processing architectures based on a hierarchy of XML schemas and an implementation – called ALPE. We have argued that ALPE brings some advantages over other known systems with similar objectives, mainly coming from a plus in manoeuvrability and complete automation of the configuring tasks. It is also shown how ALPE, has brought already significant advantages in the context of a multilingual research project. In this context ALPE has automatically configured complex processing chains involving several modules and documents in different languages. The features brought by the addition of an ALPE type hierarchy into a complex project contribute significantly to acquire multilinguality, distributivity, versioning of language resources, automatic and manual annotation, management of IPR and cost issues, as well as managing diversity of annotation styles, features that the CLARIN project considers of extreme importance.

One important further development of ALPE will be a web-service allowing users to build, configure and use ALPE hierarchies on the web, either as a limited password-protected resource or a global linguistic resources collection. This type of hierarchy is able to manage multilingual resources as well as resources which require a fee to be paid before usage. Each user will be able to contribute its own tools and annotated resources, as well as using processing chains adapted to its specifications, both in terms of input and output formats and cost and performance issues.

[9] We foresee that other requirements, as, for instance, discovery of resources and tools, preservation of resources, archiving services, content discovery, distribution, authentication and authorization, could also be designed around the structure we propose.

Acknowledgments

Part of the work for the paper was supported by the ROTEL (CEEX project) AMCSIT contract no. 29/03.10.2005, the CLARIN INFRA-2007-2.2.1.2 project, and the FP6 LT4eL project.

References

T. Brants (2000): TnT: a statistical part-of-speech tagger. In Proceedings of the sixth conference on Applied Natural Language Processing, Seattle, Washington, pag: 224 – 231.

D. Cristea, C. Butnariu (2004): Hierarchical XML representation for heavily annotated corpora. In *Proceedings of the LREC 2004 Workshop on XML-Based Richly Annotated Corpora*, Lisbon, Portugal.

D. Cristea, C. Forăscu, I. Pistol. (2006): Requirements-Driven Automatic Configuration of Natural Language Applications. In Bernadette Sharp (Ed.): *Proceedings of the 3rd International Workshop on Natural Language Understanding and Cognitive Science - NLUCS 2006*, in conjunction with ICEIS 2006, Cyprus, Paphos, May 2006. INSTICC Press, Portugal. ISBN: 972-8865-50-3.

H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan. (2002): GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the ACL (ACL'02)*. Philadelphia, US.

H. Cunningham, V. Tablan, K. Bontcheva, M. Dimitrov. (2003): Language engineering tools for collaborative corpus annotation. *Proceedings of Corpus Linguistics* 2003, Lancaster, UK.

D. Ferrucci and A. Lally. (2004): UIMA: an architectural approach to unstructured information processing in the corporate research environment, *Natural Language Engineering* 10, No. 3-4, 327-348.

N.Ide, Bonhomme P., Romary L. (2000) : XCES: An XML-based Encoding Standard for Linguistic Corpora, *Proceedings of the Second International Language Resources and Evaluation Conference*. Paris: European Language Resources Association

Sustainable operability: Keeping complex resources alive

Menzo Windhouwer[a], **Alexis Dimitriadis**[a,b]

[a]University of Amsterdam, [b]Utrecht institute of Linguistics OTS
M.A.Windhouwer@uva.nl, alexis.dimitriadis@let.uu.nl

Abstract

The data contained in a typological database are difficult or impossible to use on their own. Sustainability must include not only preservation of the data, but also of the interface designed to present them—or a reasonable substitute. The *Typological Database System* project (TDS), which originated as a way to address issues of fragmentation and interoperability of typological databases, also points the way to a model of sustainability beyond the lifetime of a database's host application.

1. Introduction: Obstacles to the sustainability of complex resources

While the sustainability of language resources such as corpora and dictionaries can be largely safeguarded by relying on documented, standard formats for their encoding, the approach does not scale well for resources with more complex internal structure, for which no general standard can be sufficient. Such complex resources have the characteristic that they require a certain software tool for their proper utilization; and that this software tool is not generic (e.g., an audio player, text editor, or linguistic annotation tool that supports the storage format of the resource), but is made specifically for the resource in question: Databases, in particular, are typically accessed through a custom-made user interface. A second, interacting problem is that much of the information needed to properly navigate and interpret such data is encoded in its user interface, not with the data itself. We consider the case of typological databases, and describe our approach to their integration and long-term sustainability.

Consider, as a concrete example, a typological database consisting of several linked tables, accessible over the internet through a web interface comprising several forms. Numerous such databases exist today, and more are being created at a rapid pace.[1] Once they are completed, such databases are subject to the usual perils afflicting electronic linguistic resources: Gradual obsolescence of their encoding formats or host software; sudden disappearance due to incompatible software updates, retirement of a "legacy" server, or as URLs change and links fail to be updated; gradual fall into unusability with the dissipation of the insider knowledge often needed to usefully operate a poorly documented resource; etc.

To render such a database sustainable, it is not enough to export its tables in some format guaranteed to be readable (e.g., tab-separated files in a Unicode encoding, or even an SQL dump in some portable format). Doing so is insufficient in two important respects:

a. The meaning of the table contents, and their interrelationships, are not explicitly given in the data tables; this is the familiar problem of documentation for a resource, but exacerbated (compared to corpora or dictionaries) by the complexity and variability of database structures, and by the relatively abstract level of linguistic description involved.

b. Even if accompanied by full documentation, a static collection of data is difficult, tedious, or even impossible to utilize without a suitable software tool. The forms and menus created by the original developers to operate a database are essential to its use, but they cannot be exported along with the data. We will term this consideration, which has not received as much explicit attention as issues of format and access, as the problem of sustainable *operability*.

To appreciate the scale of the operability problem, consider the difficulty of using a general-purpose table browser (a spreadsheet application, for example) to navigate the contents of a database consisting of several tables. Table columns (attributes) typically contain numeric values expressing different properties (whose meaning is, at best, explained in a separate document).[2] The tables are linked to each other by means of numeric keys with no intrinsic meaning. The process of navigating such data is tedious and error-prone, and likely to deter all but the most motivated researchers.

Lack of operability also has a detrimental impact on resource discovery: Summary metadata can only give an approximate indication of the utility of a resource for any particular task. A future researcher who will need to evaluate a large number of potentially useful resources will be hindered by the inability to inspect their contents without a large investment of effort.

1.1. The limits of data-only formats

The vast majority of existing typological databases are stored in relational database management systems. The

[1]Web-accessible databases include the Graz Database on Reduplication, at *http://reduplication.uni-graz.at/*; the databases of the Surrey Morphology Group, at *http://www.smg.surrey.ac.uk/*; the Typological Database of Intensifiers and Reflexives, at *http://userpage.fu-berlin.de/~gast/tdir/*; the Stress Typology Database, at *http://stresstyp.leidenuniv.nl/*; the Berlin-Utrecht Reciprocals Survey, at *http://languagelink.let.uu.nl/burs/*; etc.

[2]In proper relational design, numeric values can be indices into a separate table that matches numeric codes to a text equivalent. In practice, however, the meaning of numeric values is often embedded in the user interface; and prose documentation can be nonexistent or out of date.

relational structure itself is a sort of encoding standard, and would seem to provide a basis for standardization: While SQL implementations are too variable for database dumps in SQL format to be themselves portable, some version or extension of standard SQL could conceivably be chosen as the standard for data archiving. Even if the obstacles to unifying the many extant flavors of SQL could be overcome, however, the result would allow implementation-independent data storage but would still not render databases operable. The SQL schema of a database is insufficient in the same respects already mentioned:

First, it is an incomplete description of the database, since it does not include those parts of the database logic that are encoded in the user interface: Documentation and instructions to the user, business rules (explicit or implicit), and, in many cases, the text equivalents of values and menu options that are stored as small integers in the database. In the language of the OAIS Reference Model (ISO 14721, 2003), an SQL dump of a typological database is rarely "independently understandable."[3]

Second, general-purpose browsers for relational databases are too low-level; they allow viewing of one table at a time, but do not automatically perform appropriate joins or aggregations of records in one view—and, even with knowledge of foreign key declarations, have no way of determining which joins or aggregations are "appropriate." Simply put, the user interface of a database is underdetermined by its relational schema.

We doubt that these problems are restricted to relational databases. Similar issues doubtless arise with other complex resources developed with their own interface, and with other data models besides relational databases.

1.2. Toward a solution

The difficulty of achieving sustainable operability can be summarized as follows: Complex resources require ad hoc software that cannot be maintained over the long term; so operability can only be ensured by relying on generic software that can be maintained, and periodically replaced, in a cost-effective manner. But traditional data archiving practices do not provide enough information for generic software (or even human specialists in many cases) to reconstruct the proper structuring and presentation of the data.

It can be seen now that to fully meet the goal of sustainable operability, the archived data must first be "independently understandable." We can distinguish here between user-oriented metadata (documentation), which helps users interpret the data when it is presented, and formal, system-oriented metadata that is machine-understandable and can describe not only the encoding and relational structure (narrowly considered), but also appropriate ways of managing and presenting the data to the user.

[3]The OAIS Reference Model charges conforming archives with ensuring that archived information be "independently understandable" by its designated target community, i.e., interpretable without recourse to hard-to-access resources, including the individuals who created it. This is considered necessary for long-term data preservation. We thank an anonymous reviewer for calling this point to our attention.

What is needed, minimally, is a software platform that provides operability of typological databases with diverse structures. While no tool could probably be fully generic and at the same time achieve operability without a prohibitive amount of configuration, the problem is not intractable when restricted to one application domain at a time—in our case, to the data models applicable to typological databases. But no software platform can make up for the lack of information that is essential to managing or understanding a resource; this problem must be addressed by ensuring that the required information is collected, and is suitably utilized by the software platform in question.

Sustainable operability, in short, requires two things: sufficiently rich metadata and documentation for the data to be not only "independently understandable" by its end-users, but also for automatically determining appropriate ways of rendering it; and a software tool, or a series of software tools over a long period of time, that utilize this information to provide the actual operability.

To provide operability of an open-ended collection of resources in a practical way, there must be a way for a generic application (or several) to be used with all of them. Because the native storage formats (usually relational) are insufficient to describe typological databases to a degree that allows operability via a generic tool, we adopt a hierarchical, semi-structured data model that combines the data itself with rich documentation of database contents and of the linguistic properties being described. We will term this the *Integrated Data and Documentation Format* (IDDF). Sustained operability is then a matter of mapping resources to the IDDF format at the time of archiving, and maintaining a generic tool, or tools, that support searching and browsing over IDDF resources. This approarch accomplishes operability of the databases in the narrow sense, and also provides access to the documentation needed by the end user to properly interpret the available data.

Eventually, even the generic software will approach obsolescence due to changes in web technology, host operating systems, and the like. At that point it will need to be replaced by new IDDF-aware software with analogous functionality. The self-describing nature of IDDF documents is meant to support their migration to new access tools (or even the addition of new tools next to existing ones) without any changes to the resources themselves.

But long-term operability is more than a matter of keeping the software running. A proper solution should also support other considerations of sustainability. In particular, it should be positioned within a scenario involving data archiving and its complement, resource discovery.

The Typological Database System (TDS), described in more detail in section 2., is a working implementation of such an architecture. The TDS provides integrated access to a collection of independently developed typological databases through a single, generic web interface. Databases are imported into the system through a process that combines rich documentation of all aspects of the data with automated transformation the data itself into a common, hierarchical data space. The result is a unified data structure (the IDDF data tree) that can be searched or

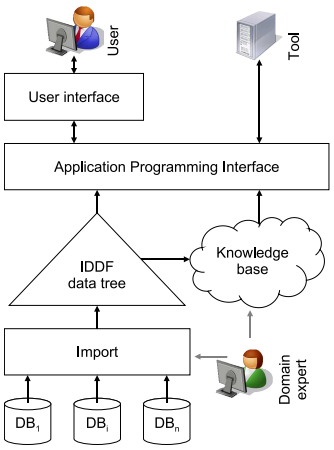

Figure 1: The TDS architecture

browsed over the web through the TDS webserver.[4]
While the process can easily be performed on each database separately, the approach has the added benefit of allowing the integration of multiple databases into a unified resource. (This is in fact the primary goal of the TDS). Arguably, integration is not essential for sustained operability of the resources; but it greatly enhances their usefulness, efficiency of utilization, and ease of resource discovery.
Data archival inadvertently exacerbates the problem of operability, because archives cannot commit to long-term hosting and maintaining a caleidoscope of diverse database applications; rather than wait for obsolescence of the software or hardware, operability threatens to be lost at the moment of uploading the static content of a resource to a digital archive. From our perspective, this can be seen as a blessing in disguise: Sustainability problems can be addressed while the original technical infrastructure is still operational, and the custodians of a resource still possess the required knowledge (either in their heads or as offline documentation).

2. The Typological Database System

The Typological Database System is a web-based service that provides integrated search access to a collection of independently developed typological databases. The system consists of a data integration module and a web server that provides access to the integrated data.[5] At the intersection of the two parts is the IDDF, a hierarchical data model that integrates data and metadata from multiple databases into a unified data space.
Figure 1 shows the TDS architecture. The primary data input to the system comes from the component databases. A domain expert creates an import schema that includes a mapping of each database into a unified hierarchy, enriched by documentation of the data and its relationship to the common TDS knowledge base. On the basis of this schema, data and documentation from multiple databases are integrated into a single hierarchical structure, the *IDDF data tree.* A separate component of the system, the TDS webserver, supports querying, browsing, and resource-discovery functions over the collected data.
The entire system is XML-based and relies on a number of (commercial) open source or freely available libraries. It is written largely in Java, XSLT, XQuery and a XML pipelining language specific to the application server 1060 NetKernel.[6]
With around a dozen databases currently in the TDS, the total number of parameters in the system is well over a thousand; hence the system follows a two-stage access model, consisting of resource discovery followed by query formulation and execution. During the resource discovery stage, users search or browse the combined metadata to discover database fields of interest. The user interface supports integrated search, display and navigation of the metadata, presenting users with the information necessary to assess both the relevance and the correct interpretation of a field. Selected fields are accumulated using a shopping basket model. In the second stage, the user constructs and executes a query on the basis of the fields in the query basket.

2.1. The integration process

The import schema is defined in a special-purpose language developed by the TDS project, the *Data Transformation Language* (DTL). [7] The TDS import engine interprets the DTL specifications, and uses an appropriate software plug-in to extract data from a copy of the original database (which can be in a variety of database formats) and transform it into the IDDF tree.
Typically, the documentation provided with a database is insufficient to make its semantics and logical structure fully explicit, and the creation of the DTL specification involves repeated interaction between the TDS domain expert and the creators of the database. The required metadata, which often lives only in the heads of the database's creators, is in this way elicited and recorded. The process is non-trivial but necessary for the sustainability of the data. Because the developers of the component databases have devoted much time and effort to collecting information in their databases, each component database represents a valuable resource; and therefore the time investment is justified.
In any event the process is reusable: Once the transformation schema has been defined, new data added to the database can be imported with minimal human intervention. In this way a database can be mapped to the IDDF before the data collection is finished and its data frozen.

[4] *http://languagelink.let.uu.nl/tds/*.

[5] The TDS is a project of the Netherlands Graduate School of Linguistics (LOT). It is supported by a grant from the Netherlands Organization for Scientific Research (NWO), and by funds from the participating universities (University of Amsterdam, Utrecht University and Leiden University). For more information on the TDS, see (Saulwick et al., 2005; Dimitriadis et al., 2005; Dimitriadis et al., 2008).

[6] *http://www.1060.org/*.

[7] The DTL is a non-procedural language that allows an IDDF schema to be specified and annotated, and the resulting data tree to be populated from the database contents. It was designed for use by linguists with no special technical background. See (Saulwick et al., 2005; Dimitriadis et al., 2008).

Only if the database schema is modified is it necessary to modify the transformation schema.
It should be added here that while it is necessary to have a working understanding of a database's semantics in order to integrate it into the TDS, much of the documentation collected and recorded into the IDDF tree is not explicitly encoding-related, but intended for the benefit of the end-user. For example, a TDS component database gives the number of basic color terms in some languages as "4.5". As a matter of encoding it is enough to know, as its documentation explains, that color term counts can be fractional numbers, and that 4.5 means "between four and five". But what does "between four and five" *mean*? It might indicate a dialectal split, inconsistencies between speakers, the presence of a marginal or dubious color term, uncertainty about the facts, or all of the above. The answer is of interest to potential users of the database, and only its creators can provide it.
Conceptually, the DTL is just one means of carrying out this transformation;[8] what is important from our present perspective is that the DTL, or an equivalent, defines a mapping of a data resource into an IDDF tree; and that the result comprises a combination of data and relevant documentation. Our vision of the IDDF is as an open format, which can be generated and manipulated by other tools. Section 3. gives more details on its structure, and on the way other components of the TDS architecture can be generalized.

2.2. What is transformed

Independently created data resources differ in a variety of ways, which need to be addressed during the integration process. The TDS makes an important distinction between differences in encoding (in the broad sense) and differences stemming from deeper theoretical or practical considerations. The former include variation in font encodings or notational conventions such as interlinear gloss labels, codes for Boolean values (*true/false* vs. *0/1*, etc), the organization of information into fields and tables, etc. The deeper differences are ultimately differences in meaning (semantics): They stem from considerations such as the theoretical commitments of a research group (including the associated terminology), the specific classificatory categories and coding decisions adopted during the construction of a database, etc.
While standardization efforts might one day lead to more uniformity in structure and encoding among databases, they will have no effect on the divergence of theoretical viewpoints and research traditions that constitutes the most intractable source of heterogeneity. These diverse viewpoints are not only dearly held by their practitioners: They are the subject matter and outcome of linguistic analysis, and cannot (should not) be replaced by any uniform, agreed-upon framework. While it might seem like a good idea to transform data into some "standard" terminology, the abstract nature of typological data collections makes this impossible. First of all, two theoretical terms are rarely if ever exactly co-extensive; even if they were, the terminology

[8] One could, for example, convert data into XML and transform it by means of hand-written XSLT, as the TDS did during the pilot phase of the project.

```
<iddf:warehouse
 xmlns:iddf="http://.../ns/iddf">
  <iddf:meta>
    <iddf:scope id="tds" type="warehouse">
      ...
    </iddf:scope>
    <iddf:notion id="n1" name="language"
     scope="tds" type="root"
     key-datatype="enum">
      <iddf:label>Language</iddf:label>
      <iddf:description>
         One of the world's languages
      </iddf:description>
      ...
    </iddf:notion>
    ...
  </iddf:meta>
  <iddf:data xmlns:tds="..." ...>
    <tds:language iddf:notion="n1"
     key="...">
      ...
    </tds:language>
    ...
  </iddf:data>
</iddf:warehouse>
```

Figure 2: The top-level structure of the IDDF.

adopted by a researcher is often the result of a deliberate process, and can be felt to be as much a part of a linguistic analysis as its empirical claims. To substitute terminology under such circumstances would be a form of misrepresentation.
Accordingly, the TDS approach is to compensate for encoding differences wherever possible, by transforming the source data to adhere to, or at least be relatable to, a uniform design ("object model"); but semantic divergences are maintained, and are made explicit by suitable documentation and careful construction of relationships between various levels of metadata.
Because the various component databases each have their own schema and focus, i.e., they are heterogeneous, the aggregated IDDF data is semi-structured. To assist in the process of resource discovery by end-users, the TDS metadata includes links to a unified knowledge base, consisting of an ontology of linguistic terms and several taxonomies that provide quick domain-oriented entry points.

3. Sustainable operability with the IDDF

At the heart of the TDS, and of our vision for sustainable database operability, is the IDDF data tree. It organizes data and metadata into a unified structure that provides sufficient information for generic resource discovery, query operations, and interactive browsing tools.

3.1. The IDDF data structure

The IDDF data structure consists of two parts, a metadata schema and a data part. The metadata part defines and annotates the schema to which the data part conforms.[9] We use the term "data tree" to refer to the entire structure, since the

[9] The IDDF can be conceptually considered as the concatenation of two documents. The document as a whole is valid XML,

two parts are closely interrelated. An abbreviated example is shown in figure 2. (A detailed example is given in the Appendix).
Figure 3 provides an informal overview of the conceptual structure of the IDDF data tree. It can be informally understood as a hierarchy of nodes (called Notions), which serve a variety of functions.

(namespace)
Grouping Notion ← (documentation)
(more groups)
Data Notion ← (documentation)
(Data) ← (value documentation)

Figure 3: Conceptual organization of the IDDF data model.

At the leaves of the tree are *Field Notions,* which correspond to fields of the component databases.[10] When the tree is built ("instantiated") by importing the databases, these Notions are populated with the data. (Note that the documentation remains in the schema portion of the IDDF, as shown above).
There are also *Grouping Notions*, which contain other Notions (either of the data or the grouping kind) and thus define the hierarchical structure of the IDDF data tree. Fields from several databases can be mapped to the same part of the tree, even the same Notion; for example, the attribute *Language Name* is a single Notion used for all databases. (The TDS organizes data according to topic, regardles of its database of origin; one could easily adopt a different policy, and map each database into a dedicated part of the hierarchy).
To facilitate management of all this data from diverse sources, Notion definitions are overlaid with a system of namespaces, which can be nested; Notions defined in a particular namespace can only be used within its scope. For example, the TDS project defines a top-level "tds scope" that provides the upper levels of semantic context, such as *clause-level phenomena.* The component databases can then define database-scoped Notions as descendants of appropriate points in the global hierarchy.
Besides its content, each Notion is associated with documentation and format information (which are stored in the schema part of the IDDF, as detailed below). Grouping Notions can be associated with a description of the kind of data they dominate, including summaries of the linguistic theory and terminology of the data providers; Field Notions can be associated with a description as well as an enumeration of possible values, which can themselves have associated documentation.

validated against a Relax NG schema that essentially ignores the data section. Validation as an IDDF document requires two passes: After the initial minimal validation, an XSLT 1.0 stylesheet is run on the metadata section to generate a complete Relax NG schema. This is then used to validate the entire IDDF document.

A sample IDDF document, and the required schema and stylesheet, are available at *http://languagelink.let.uu.nl/tds/iddf/*.

[10]The relationship to the original database fields is not one-to-one. Some Notions are in fact created by splitting up or combining several database fields.

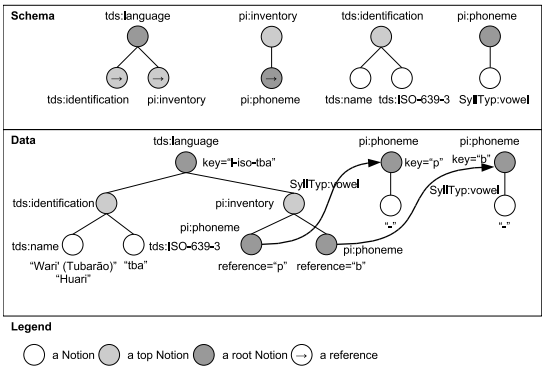

Figure 4: Graphical representation of the example IDDF schema and data tree

In many cases, a database uses a number of fields for information that belongs together and should be considered as a whole. For example, geographic latitude and longitude together make up geographic coordinates, and these together with language name, ISO code, and other essential information make up the *Language Identification* group. Each such group of fields is mapped to a subtree of the IDDF, which is identified as a *semantic context* by means of a special label assigned to its root Notion. These *Top Notions,* as they are called, are treated specially by the TDS search and browsing interface.
Larger hierarchies can be built by reusing these semantic contexts and nesting Notions inside each other. There can be multiple separate hierarchies, each with its own top-level root (called a *Root Notion*). Hierarchies can be linked to each other by establishing a primary/foreign key relationship between a Root Notion and another Root Notion. The role of Root Notions in the IDDF data model can be compared with tables in the relation model.
Figure 4 shows how the hierarchical definitions in the schema, `tds:language`, `tds:identification`, `pi:inventory` and `pi:phoneme`, are utilized during the instantiation process of the data tree. The reference leaves indicate the valid ways of linking these hierarchies together; e.g., `pi:inventory` is nested in `tds:language`; through the `pi:phoneme` reference in `pi:inventory`, the hierarchies `tds:language` and `pi:phoneme` are related.
Each of these building blocks, i.e., Notions, scopes and values, can be extensively described in the metadata. The metadata part of the IDDF document shown in the Appendix starts with describing four scopes: `tds`, `pi`, `SyllTyp` and `UPSID`. Due to space limitations, we do not discuss scopes further. A Notion schema can contain the following information:

1. an identifier;
2. a scope;
3. (optional) a label;
4. (optional) a description, possibly formatted using XHTML;
5. (optional) one or more typed links to the knowledge base;
6. (optional) one or more links to other Notions;

7. (optional) semantic data type;
8. (optional) semantic value data type and/or key data type
9. (optional) an enumeration, possibly partial, of the possible values or key values; and for each (key) value:
   (a) the literal (key) value as it appears in the data;
   (b) (optional) a label;
   (c) (optional) a description;
   (d) (optional) one or more links to the knowledge base;
   (e) (optional) one or more links to other Notions.

The example in the Appendix includes several Notions that illustrate some of these documentation units: `tds:ISO-639-3` has a description marked up with XHTML to include a link to the standards website; `pi:phoneme` and `SyllTyp:vowel` have links to concepts in the ontology, such as *segment* and *vowel*; `pi:inventory` has the semantic data type UPPC (Universal Phoneme Positioning Chart, see (Dimitriadis et al., 2008)); the metadata of Root Notions `tds:language` and `pi:phoneme` contain enumerations of their possible key values, while `SyllTyp:vowel` contains an enumeration of its values (see Figure 5).

```
<iddf:notion id="n7" name="vowel"
  scope="SyllTyp">
 <iddf:label>Vowel</iddf:label>
 <iddf:description>
  Is the segment a vowel?
 </iddf:description>
 <iddf:link type="concept" rel="as"
   href="http://...owl#vowel"/>
 <iddf:link type="concept" rel="to"
   href="http://...owl#vocalicFeatureNode"/>
 <iddf:values datatype="enum">
  <iddf:value>
   <iddf:literal>+</iddf:literal>
   <iddf:description>
    The segment is a vowel.
   </iddf:description>
  </iddf:value>
  <iddf:value>
   <iddf:literal>-</iddf:literal>
   <iddf:description>
    The segment is not a vowel.
   </iddf:description>
  </iddf:value>
 </iddf:values>
</iddf:notion>
```

Figure 5: Example of IDDF metadata associated to a notion.

3.2. The data

Since there are multiple top-level Root Notions, the data tree is actually a forest of trees, each of them an instantiation of a hierarchy dominated by a Root Notion. These trees are linked to each other using the `key` and `ref` attributes (see the Appendix). As Notions (with the exception of Top and Root Notions) can't be uniquely identified by just the combination of the scope and the identifier, each node in the tree also specifies which Notion is being instantiated, using the `iddf:notion` attribute.

Each instantiation is based on data from at least one component database. The source of a node in a tree is indicated by the `iddf:srcs` attribute. When data loaded from various databases are in agreement, they are instantiated as a single node and this attribute lists all these database scopes. But databases may also disagree. For example the Syllable Typology Database uses the name "Wari' (Tubarão)" for a certain language, while UPSID uses "Huari." Both names are stored in the IDDF document, but each in its own `iddf:value` node with a `srcs` attribute indicating its origin.[11]

3.3. The IDDF surroundings

3.3.1. The metadata and data source

The IDDF, as already mentioned, is an ordinary XML format. There are no barriers to creating valid IDDF documents with tools other than the DTL engine; one might wish, for example, to design a description language with a different syntax and primitives, perhaps for resource types that are very different than the typological databases we have been working with. Another possibility might be for a (complex) database application to directly support IDDF as an export format, without the intervention of a description language. In this case, the descriptive metadata might still need to be manually supplemented. This indicates that there could be a need for specific IDDF metadata editors. It is easy to visualize the use of a specific GUI to annotate Notions, and perhaps even to create the semantic hierarchies (contexts).

3.3.2. Links to external semantic resources

As figure 1 shows, the IDDF document can be linked to a knowledge base. In the case of the TDS this consists of an OWL ontology, developed during the course of the project, and a number of SKOS taxonomies. This allows the TDS to semantically extend queries by following the formal relationships in the ontology. The taxonomies provide alternative organizations of entry points into the data schema. Other forms of encoding knowledge, e.g. in the form of a tag cloud, could also be associated with the IDDF schema. In the TDS project, developing the metadata and the knowledge base went hand in hand. In applications of IDDF where the metadata is readily available one could also extract the knowledge base, or part of it, by mining the metadata (Feldman and Sanger, 2006; Cimiano, 2006). To get enough input for the mining algorithms one might use other related inputs, e.g., in the case of scientific databases the articles written on the basis of the data. One could also bootstrap the mining process by manually creating an initial domain-specific knowledge base.

3.3.3. Standards

Because the data in typological databases is overwhelmingly about languages, data aggregation depends crucially on reliably identifying the language that data is about. The TDS protocol relies on ISO 639-3 language codes (ISO

[11] Note that the IDDF could have also allowed each database to be mapped to a separate hierarchy, avoiding any chance of an overlap or clash.

639-3, 2007), internally and externally, to identify the language described and carry out data integration. ISO language codes are used internally as part of the key, and they are always utilized for data integration, if available. For databases or records that do not provide them, the TDS domain experts attempt to add them (by means of the DTL script), on the basis of language names and the assistance of the database creators. Again, this is a labor-intensive process but is justified in view of the value of the data, and unavoidable if the language described is to be unambiguously identified. (Once again the result is enrichment of the original data through the transformation process). In alternative application domains where cross-database integration of records is not a goal, such issues are less of a concern.

To control the proper handling of the various kinds of integrated data, the IDDF tracks the data type of each variable; the primitive types *free text* and *enumeration* can be overlaid with an open set of other (semantic) types, which are defined dynamically in the IDDF schema (that is, through the DTL) and typically apply to a group of related Notions rather than to a single one. The TDS web interface, for example, has special renderer modules for the semantic types *interlinear glossed text* (consisting of aligned morphemic tiers, a translation, etc.) and *phoneme inventory*[12]

To fully exploit this approach, it should be possible for Notions (atomic or complex) to be associated with standard data types or controlled vocabularies. Thus the ISO language code can be linked to the namespace of the appropriate authority, which provides a controlled vocabulary shared by other tools; fields conforming to other controlled vocabularies can be linked to the appropriate "data category" registered in the future ISO Data Category Registry (ISO 12620, 2008; Kemps-Snijders et al., 2008), etc. Other encoding types such as MIME types, complex structures like interlinear glossed text, etc., should similarly be reported in a standard way, and/or linked to an appropriate URI to allow their identification.

In effect, this approach extends the notion of standard data types beyond simple numeric, text and enumerated types, to more complex aggregations of data. There still work to be done in the domain of registering such resource types (the ISO Data Category Registry is designed to cover only unitary data types, not hierarchies), but the IDDF can be positioned to utilize such advances when they occur.

4. The generic user interface

The rich structure of the IDDF has made it possible to develop a generic data browser service for the typological database domain, available through the TDS server.

The TDS server is divided (somewhat imperfectly, at the moment) into an Application Programming Interface (API) and a web interface. While the web interface is closely tied to the state of today's web browsers and associated technology (including JavaScript support, etc.), the API is considerably more stable. By untangling these two better, an API can be created that provides services to multiple generations of other tools.

[12]The phoneme inventory type triggers a specific table-based rendering of a full or partial phoneme inventory.

The data browser is generic, in the sense that it does not incorporate schema or data information about any of the component databases; all such information resides in the IDDF. The browser is limited, however, by the kind of data models and displayable objects one expects to find in typological databases. Much of the data in typological databases can be displayed as tables of short values, and therefore such tables are prominent in the browser interface. There are special provisions for presenting interlinear glossed text and tables of phonemic inventories, and a mapping module for displaying data values at the geographic location of the language in question. On the other hand, there is currently no provision for displaying video streams, or (more importantly) any provision for managing data aligned to particular portions of a video stream.

While more such display modules can be developed as necessary, the browser remains generic only in the limited context of the intended application domain. For very different kinds of resources (such as experimental measurements, corpora, annotated multimedia data, etc.), one can imagine a completely different data browser that is suited to the structure of that application domain. The IDDF itself can encapsulate a wide variety of such formats.

The structure of the IDDF also makes partial compliance possible: An IDDF-aware tool, for example, could extract and manipulate interlinear glossed text from a larger resource whose full structure is not supported by the tool.

Finally, it must be acknowledged that the TDS interface (and probably any conceivable generic equivalent) is not as effective in presenting data as the best custom-built typological database interfaces; but it is more than sufficient for providing operability of the data, and other generic browsers over the IDDF data could undoubtedly do even better. In any event, several of the component databases of the TDS had no autonomous interface at all, or only a very primitive one; and the TDS interface is immensely more effective than these.

5. The IDDF in broader context

The issues we have discussed are not new, of course. We have already mentioned OAIS, the Open Archival Information System Reference Model (ISO 14721, 2003), which provides definitions of terms related to data archiving and defines roles and responsibilities in the contect of a functional model. The OAIS document discusses in some detail the requirement that archived resources should be *independently understandable* to their target community of users, and also acknowledges the issue of operability, mentioning that the native user interface sometimes encodes information essential for its understandability and noting that "maintaining Content Information-specific software over the Long Term has not yet been proven cost effective due to the narrow application of such software." In this context, our approach can be seen as a way to achieve an economy of scale, by transferring the burden of operability to domain-wide generic tools which manage the generic IDDF format. This will reduce the burden of maintaining operability in a very scalable way, and will *hopefully* prove to be acceptably cost-effective. Whether this expectation will be realized can only be determined in the long term.

The OAIS also devotes attention to issues of archiving for the *Long Term,* defined as a period long enough to raise issues of adapting to new technology or a changing user community. The latter issue, of a changing user community, is not one we address directly; our user-oriented documentation is intended to make data independently understandable to present-day linguists, not future ones. However, there is sufficient creativity and variation in today's linguistic theories that even for understandability by contemporary linguists, they must be documented in some detail. Thus the documentation that is necessary today will serve as a good basis for understandability in the future.

Mapping a database to IDDF format requires manual enrichment of the resource with metadata that cannot be automatically automatically computed from its schema. Typically, the creators or maintainers of the original resource are asked to provide supplementary information (concerning both formal and user-oriented metadata) that is not embedded in the native data dump. While this is necessary if the resource is to be independently understandable (and is therefore indispensable to real data preservation), it means that the approach is applicable only to data of sufficient value to merit this sort of intervention. For very large-scale data collection projects, this kind of attention to each incoming resource might well be impossible. In such cases, the IDDF architecture can still support operability at a lower level, comparable with that provided by present-day solutions: The resource, along with whatever documentation is available, is imported in a form that simply mirrors the relational structure of the original database. Such data cannot be rendered in the most appropriate way, but can be browsed and manipulated at the relational table level by suitable generic software. This gives a level of functionality equivalent to viewing a database with a DBMS administration tool.

For large-scale data integration, then, the IDDF "dumbs down" to a level of functionality comparable to that provided by some existing large-scale archiving solutions. For example, (Heuscher et al., 2004) addresses the task of archiving the records of the Swiss Federal Administration, which are reported to be growing at a rate of some twenty terabytes per year. The SIARD project achieves "software-invariant" archiving of relational databases via transformation, at time of import, to a consensus SQL model (SQL-3). "On principle, functionality (i.e. software, hardware) is not archived" (Heuscher et al., 2004, p. 1). Archived data can be browsed at the relational table level by reloading into a conforming DBMS. The Chronos system (Brandl and Keller-Marxer, 2007) maintains data in its original dump format and provides low-level user access, again at the level of browsing the relational structure and tables, by supporting "on-the-fly migration" from an ever-growing collection of dump formats. This approach, while allowing archives to be maintained on a very large scale, does not provide high-level operability, especially for complex data of the type we have been concerned with. The IDDF architecture allows higher levels of operability to be achieved where this is practical, but can be (under)utilized to yield low-level operability for large volumes of complex data.

Roles and responsibilities

The architecture described relies on software support at two levels: On the input side, there must be tools to support the creation of IDDF documents. On the access side, there must be a generic data browser for any supported application domain. The two levels of tools have different maintenance requirements:

Once a resource has been mapped to the IDDF format, input-side software is not needed for its continued operability (unless, of course, the original resource changes and needs to be re-imported). An archive that stores resources in IDDF format need only ensure the continuous availability of appropriate data browsers on the access side. As such browsers become outdated or unmaintainable, they must be replaced by new IDDF-aware browsers with analogous functionality.

For IDDF-based archiving to be practical, however, suitable conversion tools are necessary. In the TDS architecture, IDDF generation is carried out by the TDS import engine, which is driven by DTL schemas and relies on plugins that grant it access to various database and dump formats.[13]

In principle, responsibility for maintaining IDDF generation tools (or using them) need not rest with the archive. A resource provider can arrange to export their data in IDDF format, perhaps via a DTL-like transformation module or in some other way. If the format should become widespread, one could even expect general-purpose DBMS applications to support such conversions. For the meantime, however, archives relying on the IDDF architecture must also address the problem of bringing data to IDDF form.

6. Conclusions

As we have seen, the problem of sustained operability of complex resources is ultimately traceable to the limitations of common storage and interchange formats, which do not provide sufficient information for generic navigation. By focusing on the particular (but broad) domain of typological databases, we have shown that the rich IDDF architecture can integrate sufficient information for a generic data browser adapted to the types of data common in typological databases. The approach is extensible and suitable for alternative application domains, as long as there is some homogeneity in the kind of data that is being collected (regardless of how each resource has chosen to present it). In effect, the idea of storing resources in a standard format that can be managed with generic tools is extended to families of complex formats that represent similar data collections.

A notable aspect of the TDS is its focus not only on metadata pertaining to encoding formats and operability, but also on documentation intended for the end-user. Because of the abstract nature of linguistic analysis, such user-oriented

[13] Note that while a diverse collection of such formats must be specifically supported, there is no need to support long-obsolete formats. When an archive no longer plans to archive databases stored on eighty-column punched cards, there will be no need to maintain support for this format (or the associated hardware). Once a resource is converted to IDDF, the original format is irrelevant to operability.

documentation is essential for the proper interpretation of high-level resources like typological databases.

More generally, by collecting and centralizing metadata and documentation, the TDS archival procedure safeguards the interpretability (and therefore true operability) of the archived data.

In the context of an archival environment, the IDDF architecture also solves the problem of versioning and citeability of evolving resources: Instead requiring resource creators to maintain multiple versions of their database, an archive can simply host multiple versions of a resource, and make them available (and operable) as if they were separate databases. Hence the archive can provide a versioned, operable mirror of the database without the need for any versioning provisions in the database schema itself.

In short, the rich IDDF format can support sustainable operability of complex resources, by allowing a critical mass of such resources to be managed through generic (but domain-specific) tools.

7. References

Stefan Brandl and Peter Keller-Marxer. 2007. Long-term archiving of relational databases with Chronos. In *First International Workshop on Database preservation (PresDB '07)*.

Philipp Cimiano. 2006. *Ontology Learning and Population from Text*. Springer-Verlag, Berlin.

A. Dimitriadis, A. Saulwick, and M. Windhouwer. 2005. Semantic relations in ontology mediated linguistic data integration. In *Proceedings of the E-MELD Workshop on Morphosyntactic Annotation and Terminology: Linguistic Ontologies and Data Categories for Linguistic Resources*, Cambridge, Massachusetts, July.

A. Dimitriadis, M. Windhouwer, A. Saulwick, R. Goedemans, and T. Bíró. 2008. How to integrate databases without starting a typology war: The Typological Database System. In S. Musgrave and M. Everaert, editors, *The Use of Databases in Cross-Linguistic Studies*. Mouton de Gruyter. To appear.

Ronen Feldman and James Sanger. 2006. *The Text Mining Handbook*. Cambridge University Press.

Stephan Heuscher, Stephan Järmann, Peter Keller-Marxer, and Frank Möhle. 2004. Providing authentic long-term archival access to complex relational data. In *European Space Agency Symposium "Ensuring Long-Term Preservation and Adding Value to Scientific and Technical Data"*.

ISO 12620. 2008. Terminology and other language resources – Data categories – Specification of data categories and management of a data category registry for language resources. To appear.

ISO 14721. 2003. Space data and information transfer systems – Open archival information system – Reference model.

ISO 639-3. 2007. Codes for the representation of names of languages – Part 3: Alpha–3 code for comprehensive coverage of languages.

M. Kemps-Snijders, M. Windhouwer, P. Wittenburg, and S.E. Wright. 2008. ISOcat: Corralling data categories in the wild. In *Proceedings of the International Conference on Language Resources and Evaluation*, Marrakech, Morocco, May.

A. Saulwick, M. Windhouwer, A. Dimitriadis, and R. Goedemans. 2005. Distributed tasking in ontology mediated integration of typological databases for linguistic research. In *Proceedings of the International Workshop on Data Integration and the Semantic Web*, Porto, Portugal, June.

Appendix: A longer IDDF example

We include here a sample IDDF structure. The first part (`<meta/>`) integrates data schema and documentation, while the `<data/>` element contains the sparse data.

```
<iddf:warehouse
   xmlns:iddf="http://.../ns/iddf">
 <iddf:meta>
  <iddf:datatype id="UPPC"/>
  <iddf:scope id="tds" type="warehouse">
   <iddf:label>
    Typological Database System
   </iddf:label>
   <iddf:scope id="pi">
    <iddf:label>
     Phoneme Inventories
    </iddf:label>
    <iddf:scope id="SyllTyp" type="database">
     <iddf:label>
      Syllable Typology Database
     </iddf:label>
    </iddf:scope>
    <iddf:scope id="UPSID" type="database">
     <iddf:label>
      UCLA Phonological Segment
      Inventory Database
     </iddf:label>
    </iddf:scope>
   </iddf:scope>
  </iddf:scope>
  <iddf:notion id="n1" name="language"
     scope="tds" type="root">
   <iddf:label>Language</iddf:label>
   <iddf:description>
    One of the world's languages
   </iddf:description>
   <iddf:keys datatype="enum">
    <iddf:key>
     <iddf:literal>
      l-iso-tba
     </iddf:literal>
     <iddf:label>Aikan&#227;</iddf:label>
    </iddf:key>
    ...
   </iddf:keys>
   <iddf:notion ref="n2"/>
   <iddf:notion ref="n5"/>
  </iddf:notion>
  <iddf:notion id="n2" name="identification"
     scope="tds" type="top">
   <iddf:label>
    Language identification
   </iddf:label>
   <iddf:notion id="n3" name="name"
```

```
    scope="tds">
   <iddf:label>Name</iddf:label>
   <iddf:values datatype="free"/>
  </iddf:notion>
  <iddf:notion id="n4" name="ISO-639-3"
     scope="tds" >
   <iddf:label>ISO 639-3 code</iddf:label>
   <iddf:description
      xmlns:xhtml="http://...">
    The code as assigned to the
    language in the
    <xhtml:a href="http://...">
     ISO 639-3 standard
    </xhtml:a>.
   </iddf:description>
   <iddf:values datatype="enum">
    <iddf:value>
     <iddf:literal>tba</iddf:literal>
    </iddf:value>
    ...
   </iddf:values>
  </iddf:notion>
 </iddf:notion>
 <iddf:notion id="n5" name="inventory"
    scope="pi" type="top"
    datatype="UPPC">
  <iddf:label>
   Phoneme inventory
  </iddf:label>
  <iddf:notion ref="n6"/>
 </iddf:notion>
 <iddf:notion id="n6" name="phoneme"
    scope="pi" type="root">
  <iddf:label>Phoneme</iddf:label>
  <iddf:link type="phoneme" rel="as"
     href="http://...owl#segment"/>
  <iddf:keys datatype="enum">
   <iddf:key>
    <iddf:literal>p</iddf:literal>
   </iddf:key>
   <iddf:key>
    <iddf:literal>b</iddf:literal>
   </iddf:key>
   ...
  </iddf:keys>
  <iddf:notion id="n7" name="vowel"
     scope="SyllTyp">
   <iddf:label>Vowel</iddf:label>
   <iddf:description>
    Is the segment a vowel?
   </iddf:description>
   <iddf:link type="concept" rel="as"
      href="http://...owl#vowel"/>
   <iddf:link type="concept" rel="to"
      href="http://...owl#vocalicFeatureNode"/>
   <iddf:values datatype="enum">
    <iddf:value>
     <iddf:literal>+</iddf:literal>
     <iddf:description>
      The segment is a vowel.
     </iddf:description>
    </iddf:value>
    <iddf:value>
     <iddf:literal>-</iddf:literal>
     <iddf:description>
      The segment is not a vowel.
     </iddf:description>
    </iddf:value>
   </iddf:values>
  </iddf:notion>
 </iddf:notion>
</iddf:meta>
<iddf:data
  xmlns:tds="http://.../ns/iddf/tds"
  xmlns:pi="http://.../ns/iddf/pi"
  xmlns:SyllTyp="http://.../ns/iddf/SyllTyp"
>
 <tds:language iddf:notion="n1"
    key="l-iso-tba"
    iddf:srcs="SyllTyp UPSID">
  <tds:identification iddf:notion="n2"
     iddf:srcs="SyllTyp UPSID">
   <tds:name iddf:notion="n3"
      iddf:srcs="SyllTyp UPSID">
      <iddf:value srcs="SyllTyp">
       Wari' (Tubar&#227;o)
      </iddf:value>
      <iddf:value srcs="UPSID">
       Huari
      </iddf:value>
   </tds:name>
   <tds:ISO-639-3 iddf:notion="n4"
      iddf:srcs="SyllTyp UPSID">
      <iddf:value
         srcs="SyllTyp UPSID">
       tba
      </iddf:value>
   </tds:ISO-639-3>
  </tds:identification>
  <pi:inventory iddf:notion="n5"
     iddf:srcs="SyllTyp UPSID">
   <pi:phoneme iddf:notion="n6" ref="p"
      iddf:srcs="SyllTyp UPSID"/>
   <pi:phoneme iddf:notion="n6" ref="b"
      iddf:srcs="UPSID"/>
   ...
  </pi:inventory>
 </tds:language>
 ...
 <pi:phoneme iddf:notion="n6" key="p"
    iddf:srcs="SyllTyp UPSID">
  <SyllTyp:vowel iddf:notion="n7"
     iddf:srcs="SyllTyp">
   <iddf:value srcs="SyllTyp">
    -
   </iddf:value>
  </SyllTyp:vowel>
 </pi:phoneme>
 <pi:phoneme iddf:notion="n6" key="b"
    iddf:srcs="SyllTyp UPSID">
  <SyllTyp:vowel iddf:notion="n7"
     iddf:srcs="SyllTyp">
   <iddf:value srcs="SyllTyp">
    -
   </iddf:value>
  </SyllTyp:vowel>
 </pi:phoneme>
 ...
</iddf:data>
</iddf:warehouse>
```

Sustainability and sharability of the Humanist Virtual Library (Bibliothèques Virtuelles Humanistes, BVH): experiment feed-back

Marie-Hélène Lay[1], Marie-Luce Demonet[2]
1. ICAR3, CNRS, Ecole Normale Supérieure lsh Lyon
2. Centre d'Etudes sur la Renaissance, CNRS, Tours

E-mail: marie-helene.lay@ens-lsh.fr, marie-luce.demonet@univ-tours.fr

Abstract

While the linguistic resources multiplied during these last decades, the question of their sustainability is constantly referred to. Many projects, settled in space (such as an individual, a group), and in time (such as a thesis, a four-year research agreement), choose to create ad hoc data, « tailored for a very specific research », without bothering about the sustainability of the resources that have been built up. Yet this phase is scarcely an objective by itself: it is felt most of the time as time and energy consuming, little interesting in itself, but incontrovertible step, except if we are able to re-use existing data, in order to go faster and further. The point is how to identify, locate and use the data which would be usefull. Use of standards should be helpful, in different ways : "light standard encoding" of primary data, "pivot language" for built data, and structural annotations allowing interoperability. But without efficient tools to manage and transfer the standardized data, their use will stay as a wish.

1. Produce and use Linguistic Resources

1.1 Some elements of reflection about no-sustainability and no-sharibility from LR.

While the linguistic resources multiplied during these last decades, the question of their sustainability is constantly referred to. Many projects, settled in space (such as an individual, a group), and in time (such as a thesis, a four-year research agreement), choose to create ad hoc data, « tailored for a very specific research », without bothering about the sustainability of the resources that have been built up. Yet this phase is scarcely an objective by itself: it is felt most of the time as time and energy consuming, little interesting in itself, but incontrovertible step, except if we are able to re-use existing data, in order to go faster and further. This need coincides a priori with the request for sustainability of the resources developed in the frame of « NLP », be they corpora, dictionaries, annotation tool, and so on.

From that point, it seems interesting to ask WHY data sustainability seems to be « neglected on a regular basis ». The point of view we adopt her is one-sided: it mirrors 15 years experiment of « linguistic-and-humanities researchers » or of « humanities-and-linguistic researchers », involved in this process of creating and sustaining linguistic resources, corpora and dictionaries, with the constitution of a database containing Renaissance texts (www.cesr.univ-tours.fr/Epistemon/), with the conception and achievement of generic and/ or applied dictionaries (Lay, 1992; Lay, 1994), with the annotation of heterographic texts (Lay&Demonet, 2000). Indeed, this activity cannot be thought out of proper tools, but we choose here not to reach this aspect of the problem.

1.2 Corpora: from papyrus to hypertext

One of the first aspects of corpora transmission looks like an editorial practice, and a rather classical one (Demonet, 1999). Yet the communities of linguistic resources creators often built in proficiency in language description, or in constitution of text resources dedicated to one or another kind of publication. Very clearly, it is the case for our research groups, whereas a good understanding of environments, standards and computing tools is obviously less noticeable.

The Holy Scriptures and their exegesis have gone through the centuries (Vandendorpe, 1999). Their analysis is founded on the study by people who, along the centuries, have built indexes and concordances… by hand. This method of fact collecting is also the basis for the creation of dictionaries. No doubt that part of the job has faded away, part of know-how has been lost, but the data are mostly available, and the expertise has been transmitted.

The community of Humanities is used to create many text data, to update and impart them: classics are passed on generations, accompanied with a critical apparatus that mostly grows as time goes by. When they are by-products of editions and reprints, cheap editions, the same resources are utilized in various shapes, popular or luxury publications, commentaries, translations: through manual copy, lead fonts, computer assisted publishing, digitized and today cms editions, this community is ready to adopt all the tools, while taking into account the maturity necessary to these tools and the tutorial associated to each technical innovation accompanied with a social innovation (Demonet, 1995). The BVH (Bibliothèques Virtuelles Humanistes) are a good example and witness of this experiment (Demonet & Lay, 2008).

2. Renaissance corpus and digital library

The « Bibliothèques Virtuelles Humanistes » (Virtual Humanistic Libraries, BVH) are a program and a website in which (in the sub-database Epistemon, www.bvh.univ-tours.fr/) texts are published, and book images are digitized. Although the present topic deals only with text databases, the choice of showing simultaneously the image and the corresponding text is an important condition of standardization (Demonet, 2006): to think the text structure according to both modes of representation enables the easy transfer of models from the one to the other.

The BVH have been built depending on the requests of large communities, composed of disciplines with a variety of requirements: historians, art historians, specialists of literature, philosophy, languages, historians of sciences, a demand that can be dispatched in four directions:

- Archive (content-document oriented)
- Book history (form-document oriented)
- Linguistics (language oriented)
- Style (aesthetics-oriented)

We do not have to choose, but the linguistic dimension has to be considered as the basis for the three others. The questions we ask within the document description depend on heterogeneous targets, that we try to make compatible by the development of a specific notion, that of patrimonial and generic edition for the Renaissance (while watching out the extension to other periods). The objective is to preserve, but also to offer and to represent the documents through the highlighting by means of digital media: displaying digitized documents in image mode, documents that are browsable, downloadable or not, searching through the document by the hidden text access (owing to mass OCR), but also transcriptions of very old documents (epigraphic, manuscripts), and a highly specialized dissemination of documents that can be visualized in a diplomatic or quasi diplomatic form, owing to a DTD adapted to this type of document (see the Ecole des Chartes editions, http://www.cn-telma.fr/).).

The BVH group in Tours has planned to digitize 2000 1500-1650 books, 200-250 of them being tei-encoded texts. They are held by French libraries (mainly in France, Région Centre, Lyons, Troyes, Paris, Poitiers), but the program does not exclude any other collaboration (encouraging contacts have been held with the Wolfenbüttel Bibliothek, the University of Virginia Library for example). 212 books are online (March 2008) in image mode, 170 others (already digitized) are now being processed in order to be published at the end of the year. 17 transcripts have been published on the Epistemon website, 5 others are used as tests for encoding, before we encode 33 more (in French) in 2008.

Whoever wants to publish online patrimonial documents coming from archive repositories or libraries, is helped by XML language and TEI recommendations (http://www.c-tei.org), that constitute a satisfying frame so that the encoding norms be shared by as many communities as possible. However, these recommendations, known for a long time (Burnard, 1995) were of little use, because of the lack of convenient, or automatic encoding procedures, or of request tools that would not require encoding again, and tools that would not change with every document. While waiting for better tools, the publication of Renaissance texts on the Epistemon website (Poitiers, 1998-2001, then Tours since 2001) has been carried out of converting documents from word processing into html files. These texts have been published online for ten years (fourteen years in the case of the Rabelais database managed at the University of Nice), and, even without XML encoding, they were, and still are, useful. An intermediary solution, located in Toronto University[2], offered word search into this small corpus owing to Tactweb. Now it is time to generalize the XML/TEI encoding in order to manage requests through software such as Philologic (www.artamene.org/philologic.php) or Weblex (http://weblex.ens-lsh.fr), and to build the document structure following these principles.

2.1 « TEI-Renaissance » encoding

In the general aim of the BVH, we wish to suggest to all the users some structure patterns taken out of TEI encoding procedures, specifically with a « TEI-Renaissance » subset[1], so that they can be shared out, harvested and requested together. The main corpus is the whole collection of text data aggregated by the CESR in his program: it concerns, mostly, a corpus of printed documents going from the end of the sixteenth century up to the mid-seventeenth century: the *terminus ad quem* does not signify a major change of media that would necessitate another type of encoding, but a reasonable size, according to the material and human means we can afford[2].

The TEI-Renaissance application (using the P5 guidelines) enables to integrate the manuscripts with the printed material and to restore their linguistic plurality. In fact, the document produced during the second half of the fifteenth century, when handwritten copies imitated by incunabula coexist with prints, and with the same specifications and structure, can be linked to the historical collection of three centuries of printing. It is not a specific encoding for French literature,and no pre-definite subset could be used, because of the variety of genres: the observation of other online publications[3] shows the possibility of multilingual usage of structure description. The universality of tags allows the compatibility of analyses, up to a certain degree of description: headers, text segmentation and division, speakers, proper names. Mixing languages being very frequent inside the texts (with quotations in Latin, Greek, even in Hebrew and Arabic, and about twenty idioms

[1] List of main attributes used in the TEI-Renaissance :
ana, calendar, cols, corresp, date, hand, ident, n, part, place, reason, ref, rend, resp, role, rows, scribe, target, type, url, value, who, whole, xml:base, xml:id,

[2]

used by Rabelais…), the consideration of multilingualism is an important element in the plurilingual restitution of the document. The digitized sources of the BVH program (that started in 2003) come from all kinds of encyclopedic fields, written in various languages: their publication offers therefore more general objectives than those which prevail in the communities studying a particular state of an idiom (old or middle French, English, German…) and build specific tools adapted only to literary documents.

2.2 A « patrimonial » encoding

This « patrimonial » encoding owing to this Renaissance adapted TEI is the procedure that offers a state of the text digitally represented and that is the smallest common denominator for all communities. It excludes therefore, in principle, every specialized annotation. It offers a basic encoding that respects at best the status of the source, this one being displayed ideally according to several choices and being enriched by addition of other label sets without resuming every operation. The manual (about to be) displays generic recommendations, and options that give the possibility of adapting them to such or such database.

This encoding process is said « patrimonial », first, because the reference given is that of the repository location (libraries, archives, museums, private collections), and the « headers » are made compatible with the cataloging norms in effect: we are waiting for precise instructions from the Bibliothèque nationale de France, to include book or manuscript metadata currently in use within the community of librarians. Already, the mutual harvesting of repositories requires an OAI protocol and the standardization of catalogs (http://www.oai.org/). It has been already effective with the French National Library (2006), the BIUM (Library of Medicine, Paris) and the Library of Troyes (2007). Afterwards, the patrimonial encoding presumes that we respect at best the elements that signify the historical nature of the document and that we do not blur the traces of this diachrony. To keep page and line numbering is part of this preservation, as interesting for the history of the document as for that o the language in which it has been written.

This point allows us to evoke one of the difficulties in the applications of standards: the TEI encoding enables, in principle, this segmentation, but we had to find an economic solution to keep both the end of line (marked with a hyphen) and the automatic treatment of the whole word; we had tocreate a new element, (<caes>)[3] that had to be validated by the TEI Consortium. Without this element, the subsequent sentence, taken from the *Euvres* by Louise Labé (1555)

<p>Qui est cette fole qui me pous-
<lb/>se

would have prevented the automatic treatment of the whole word “pousse”. With <caes>, it is possible to take into account the division of the syllable at the end of the line. This was necessary, in order to tag also the words that are divided without any hyphen, as it is often the case in many Renaissance prints or manuscripts.

The new encoding appears in this form:

<caes whole="pousse">pou-</caes><lb/>
<caes part="F">se</caes>

which imposes to process the end-of-lines with a macro, to create the routines.

Depending on communities, keeping this original layout offers a major or minor interest, and its enforcement had to be estimated in terms of human costs. Questions of this kind are asked about modernization of ij and uv, the development of abbreviations and the processing of ligatures. Renaissance and seventeenth century typography often ignores the distinction between u and v at the phonological level (the position alone is important): avoir is typed “auoir”, un is “vn”, and it is the contrary for capitals; very few “j” are used before the end of the sixteenth century, except for numbers. For the comfort of the modern reader, and the linguistic processing, we need to modernize; but to build the dictionaries that are necessary to improve the OCR, we need to use a version with the exact spelling. The possibility of offering several versions of the text owing to versioning systems modulates this ground transcription, according to choices operated by the end-user. Abbreviations and ligatures can be easily encoded, or not, with <abbr> and <expan>, or directly converted when using an OCR with training. For our purpose, we give priority to readability.

2.3 The Text-Image Link

At last, we want to underline that the major care has been applied to the text document structure, because it commands also the image document structure. Actually, the software Agora (Ramel &al., 2006) (developed by the Computing Laboratory in Tours) segments semi-automatically the components of the page (titles, bands, margins, illustrations, ornamented letters) and generates XML files labeled according to the text divisions settled with the TEI encoding. The names of the “types” within the elements of the tags are given according to the typographic terminology. OCR outputs are able to generate xml files compatible with the METS/ ALTO standards, and to format online publication of structured documents.

This correspondence between text and image publications has lead to the idea of associating a thesaurus of topoi, already in use for iconographic indexation. The advantage is to overcome the linguistic barrier and to search through the text in a way that associates the image to the text owing to an alphanumerical number. The Iconclass thesaurus, that inventories through a conventional tree-structure the diverse motives in use to describe images, is also useful for texts and can be encoded according to the TEI. This code manages also the links between the image of the book, and allows the request in any language of the

[3] <caesura> already existed, to mark for verse encoding.

thesaurus (English, French, Italian, German), and partly in 22 other languages. This topic is an important milestone in the standardization of indexing through keywords for documents quite heterogeneous: the same keywords can also be used to link databases of objects that are not books, such as collections of museums, sculptures, architectural elements, etc. This thesaurus is often discussed in the community of art historians (particularly in France), but up to now no other multilingual thesaurus exists. The standardization operates also through the adoption of conventional interlinguistic descriptors.

However, experience shows that even a thesaurus as steady as Iconclass is not secured against main technological changes: the installation of a new platform provoked lately the interruption during several months of accessibility and harvesting system of several simultaneous and distant databases. More generally, the projects of portals that flourish everywhere, and seem to solve the problems of accessibility and interoperability, are likely to generate other instabilities between regional portals, as well as institutional, private, international and national portals…. The running of a steady website is also dependent on the political decisions that overtake the main options.

3. Enrichment of text access

Beyond the availability at the best state of the art, we wish to build with the BVH an original library, by offering to its visitors text access based on tools that come from the NLP. Indeed, the evolution that corpus edition acknowledges is increased by the evolution of annotation tools. The availability of annotated texts enables the enrichment of the traditional access to texts and broades the lexicometric practice (lexical statistics on text forms) on the textual and linguistic levels. The documentary treatment of annotated texts can be done while articulating search engines and lexicometric tools that equip (among others) the old practice of indexes and concordances (Demonet, 1996). To this purpose, we establish for texts linguistically annotated editions. And for this, annotated resources imply the same question : how could it be possible to ensure their sustainability, which means the sharability and the reusability of such a complex resource, involving different other resources in a particular way.

3.1 The need to access the primary resources

In fact, the enriched, annotated, corpus is the meeting point of textual and annotational resources (manifold types of thesauri, dictionaries, and grammars) that do not imply the same constraints. To study the texts, our ambition is, for a defined objective, to enrich the corpus with all the relevant and available information. In a typical way, we wish to combine a wide coverage of linguistic resources (as much information as possible) with an efficient validation (i.e. choosing among information and selecting what is relevant), in terms of corpus coverage in terms of deepness of the descriptions.

The richer the available annotational resources are, the more important the validation of the annotated corpus is: the number of choices to operate and to validate at each stage increases with the quantity of constructed annotation (it does not imply inevitably that it is more difficult). This shows the narrow dependence between the quality (i.e. the bundle of properties) of resources in use, be they textual or linguistic, and the quality of the annotated text. The evaluation of this quality remains a problem in itself. Validation (or any other treatment of annotated texts) implies to be able to have access to the annotational resources as much as to the textual resources.

The annotations are in fact produced from these different sources and their interpretations need their contextualization, as much on the syntagmatic level of the observed text, as on the level of the preferred paradigmatic organization (i.e. the underlying lexicological model, the grammar of components, the thesaurus of warfare, etc.).In the field of specific literary or linguistic research, the use of available resources is a shared wish; but the need is to exactly know about the encoding choices and their foundations, to undestand how to modify it.To carry out the available resources tightly depends upon the fixed objectives. The same phenomenon (linguistic or not, such as the distinction between the name of an actual/ fictive character) can be seen from different points of view according to the selected description model. The model itself is prone to various interpretations; at last, many readings are possible for the same segment of the text. And if the use of existing resources, or the possibility of sharing constructed resources leads to overcost, the choice is to operate « from scratch »: even if the result is not absolutely satisfying, every stage of the description process in under control, not depending n somebody elses. And so is highlighted one of the major problems of the reuse of this kind of resources: the possibility of offering several valid annotations, even, the necessity of multi-annotation.

3.2 What about generic data ?

Every observation made during the study can lead to redefine part of or all the data in use, to construct sub-corpora, to aggregate dictionaries of different origins, to tailor them again: resources are living objects.

In our case, the wish of deal with several dictionaries (generic and dedicated dictionaries) corresponds to our research projects about the works or the authors of sixteenth century : studies of proper names linked to a particular work or a period, studies on spellings/OCR project, study of graphic variation for graphematic studies, or attribution, studies on word evolution and semantic constellation.To think about this different uses of the corpora and dictionnaries explain why : « often, the language resources are tailored to the needs of an individual application or of a project with a very specific research question. » To say the truth, the first characteristic of a corpus is not to be reusable as such

(Meyer, 2002) : a priori, it has been thought, and customized, described and annotated so that it represents at best one given problem. Corpora are not natural objects (if one such thing may exist), but constructed objects. Then, the question is: how can we achieve the transformation of the constructed data in «sustainable data », made available since they are neutralized in order to be manipulated and reconfigured.

The creation of such resources, said « generic », is a constant preoccupation since the 1980s, with the launching of research programs in frames such as ESPRIT or EUREKA. Evaluation campaigns (that ground quality criteria for data) have similar effetcs : exchange formats must be defined, allowing comparison between the data from each competitor. The observations we can find, for example, in the introduction of the Genelex report (GenericLexicon) (Lay, 1994), insist on the necessity of making the difference between lexicographic data, lexicological models of description for these data (lexicon-grammar, trees-adjuncts, traditional lexicological models), the formalism used to express these models (i.e the generic model making them compatible), and the representations linked to the implementation choices. Finally, there is a critical need for (1) information about data and and models, (2) dedicated extraction and management tools ; The two of them in order to use the "sustainable data" in a specific context.

In each aspect, recommendations that could become standards can be offered ; in such a way, standards can be useful: they utter very precisely a milestone that enables for an object or part of if to be situated. By taking this milestone into account, one can benefit the available data and all the tools that come around and allow the use of the standards.

Yet the dealing with lexicographic data emphasises that « sustainability is a multi-faceted task which depends on different individual subtasks », one of these subtasks (and not the least) is to work on the problem of data intricacy and overlap, on interoperability between data and between manipulation tools, etc (Heiden, 2006). The applications and researches requiring annotated corpora work with complex data. Yet by the tokenization task, we have to deal with the use of specific information. We need to determine what status to give to the ' - . to specify in which context they are separators or lexical units (l'=le , aujourd'hui, S.N.C.F, mRNA) ; we can also decide to appeal to accentuation rules, to identify multiword expressions, etc. The management of available resources is, in fact, only a part of the problem: they are in use in a succession of complex operations. Each operation need choices, generic ones and many micro-choices, depending on the defined goal, interacting with all the different stages. This fact explains why « tools whose algorithms and data structures are poorly documented ».

On the way, research leads to create new data, but these data are inserted in an experimental process. Nothing says that all the data of an experiment are reusable. Some of them are intermediary objects, not to be kept. Others are to be stored, but they are not inevitably transferable. Some of them, at last, should be integrated in generic data repositories. But this will not be done,if this task leads to overcost

4. Sharing annotated data : which validation for which annotation level?

One of the questions implied by the reuse of data (sustainable data that would be never used are of little interest), is the insurance of quality and reliability of data. Insofar, to determine in itself what is a good annotation remains a difficult endeavor: the relevance of an annotation depends very often on the decided objective of usage.

According to contexts, information that costs too much to be verified can be discarded, if the purpose of the study does not require a complete and homogeneous labeling. We can judge that all the items do not require to be described with the same level of precision (partial disambiguation), on the syntagmatic level (there is nothing to say about an occurrence) or on the paradigmatic level (a precise disambiguation can be irrelevant, f.e it is enough to identify a verb, without going further in the description of the inflexion). For a described unit, these consideration will determine the choice (or the non choice), of validating a sub-set of possible labels, excluding others. Thus, in a forthcoming study about the texts by Rabelais, we have decided to neutralize all the information that has no relation with nouns and adjectives. A research about the personal pronouns does not require the same type of annotation as a study about a semantic field.

Consequently, one should wonder about the transmissibility of partial labeling, of aggregation and/ or reversibility of such annotations, echoing this assertion (Habert,2005): [The automatic language processing] « obliges to provide the attested data with fine and multiples annotations, allowing an improvement towards underlying regularities. In order to survey the coherence of the data, we must, a minima, be able to encode explicitly a partial labeling, to signify that all the information about interpretation for a given category has not been encoded. »

therefore, to enable the treatment of the available data and the capitalization of information that have been constructed during this treatment, obliges to get out of stabilized vision of resources. We must have, for a basic corpus, a set of annotational resources and one model of data, used for annotation; we must report the diverse possibilities of treatment. Given a particular application, we wish to define a particular instance of the model, that goes sometimes up to redefine the category initially designed. The definition of categories is settled at the level of data modelization and enables the description, materialized by labels, of units observable in a corpus, following a description system external to any application. If the labels are understood as designations

associated to categories within a model, the practice of their assignation needs anyway associated information, in the form of recommendations more or less formalized, and more or less implementable-implemented.

The conditions of assignation describe, for a given list of labels, the way by which these labels will be used during the enrichment of the text. They can be integrated in two places: (1) into the label set ; that is, the relevance conditions are encoded in the instance of the model), (2) into the labeling tool ; that is, the relevance conditions are joined to the assignation rules.

The conditions of assignation belong to these different levels : (1) level of the system, (2) level of the instructions for use, (3) level of the contractualisation of the targeted objective. Indeed, these conditions are the formalization of syntagmatic contexts on the paradigmatic level: rules formulation (system) to be used (instructions) when labelling, in order to reach the expected result (contractualisation). It may involve the formulation of local constraints, to take the elements of the intratextual syntagmatic context into account : for instance, « le » is an article before a noun, but a pronoun before a verb; a "proper noun", if associated with a verb of movement, can signify a place, but precedes by a formulation of « titles », it may signify a named entity…). The contextual constraints may involve the text as a whole, intra or inter corpus : they can require several levels of information such as the metadata of a work, like genre, domain, date, etc. In fact, the meaningfull interpretation may imply the explicit access to an enlarged context.

5. The need for mutable corpora and "reversible multi-annotation" tools.

The « sustainability » of the annotated corpora seems to rely tightly on formalism and tools allowing to understand the annotated text as a scalable object, on which annotations can be aggregated, or suppressed on a coherent basis (Loiseau, 2007). Making resources sustainable is often understood as a stage according to which data is stored in a "generic" repository, that insure their availability and accessibility; but does that mean that the data are sharable? Nevertheless, this is done in a fixed maner, and doesn't take the need of flexibility into account. If a resource can claim to be sharable, it is not as a finite object, untouchable and availble for ever, but as a mobile configuration, as a dynamic space, towards which converge a quaint bundle of informations, whatever the level of description may be, from the tokenisation (that is the first annotation level) to the most encyclopedic annotation.

6. References

Antoni-Lay, MH., (1992). DIOGENE, dictionnaire IBM-Genelex. Research Report n°68

Antoni-Lay, MH., et al., (1994). A Generic Model for Reuseable Lexicons: The Genelex Project. *Literary & Linguistic Computing*. 9:pp. 47-54

Antoni-Lay, MH., Demonet, ML. (2000) Adaptation d'un lemmatiseur au corpus rabelaisien : naissance d'Humanistica. Jadt2000

Bird S. & Liberman M. (2001). « A formal framework for linguistic annotation », Speech Communication, 1/2-33, pp. 23–60.

Burnard L. (1995) « Text Encoding for Information Interchange – An Introduction to the Text Encoding Initiative », Proceedings of the Second Language Engineering Conference.

Bonnin, E., Dallo, A., (2003) Hyperbase et Lexico 3, outils lexicométriques pour l'historien, Histoire &Mesure, XVIII, n°3/4

Demonet, ML. (1995), Pour une edition hypertextuelle de La briefve declaration de Rabelais. Wooldridge ed, CA

Demonet, ML. (1996), Pronostiquer avec Hyperbase, Mots chiffrés et déchiffrés, Slatkine, pp455-471.

Demonet, ML. (1999), Les oeuvres Romanesques de François Rabelais, édition en fac dissimilé, Poitiers, La Licorne, 449 p.

Demonet, ML. (2006), Les Bibliothèques Virtuelles Humanistes (BVH) au Centre de la Renaissance de Tours : numériser en région pour l'Europe, 10°journée des pôles associés, bnf.

Demonet, ML., Lay MH.(2008), Digitizing European Renaissance prints: a 3-year experiment on image-and-text retrieval, Kolkata, *International Workshop on Digital Preservation of Heritage* (IWDPH07)

Flores, S., Vachey, J. (1995) Generating a lexicon for syntactic LFG processor from a French generic electronic dictionary encoded in the GENELEX model

Habert B. & Zweigenbaum (2002). « Régler les règles », TAL, 43-3, pp. 83-105.

Habert B. (2005).Instruments et ressources électroniques pour le français, Paris : Ophrys.

Heiden S. (2006). « Un modèle de données pour la textométrie : contribution à une interopérabilité entre outils », Actes des 8ème Journées internationales d'Analyse Statistique des Données Textuelles, Besançon : Presses Universitaires de Franche-Comté.

Hunston, S., (2002). Corpora in Applied Linguistics. Cambridge. Cambridge University Press.

Kraif O., Chen B. (2004) Combining clues for lexical level aligning using the Null hypothesis approach, in Proceedings of Coling 2004, Geneva, August 2004, pp. 1261-1264.

Loiseau, S, (2007) CorpusReader : un dispositif de codage pour articuler une pluralité d'interprétations, Revue Corpus n°6

Meyer, CF., (2002). English Corpus Linguistics - An introduction. Cambridge. Cambridge University Press

Ramel, JY., Busson, S., Demonet, ML (2006) AGORA: the interactive document image analysis tool of the BVH project, DIAL, Digital Image Analysis for Library, Lyon.

Vandendorpe, C., (1999) Du papyrus à l'hypertexte. Essai sur les mutations du texte et de la lecture. Éditions de la Découverte

www.cesr.univ-tours.fr/Epistemon/
www.bvh.univ-tours.fr/
http://www.oai.org/
http://www.c-tei.org
www.artamene.org/philologic.php
http://weblex.ens-lsh.fr
http://www.perseus.tufts.edu
http://www.bnf.fr/pages/infopro/journeespro/pdf/poles_pdf/poles2006_pdf/Demonet.pdf

List of main attributes used in the TEI-Renaissance :
ana, calendar, cols, corresp, date, hand, ident, n, part, place, reason, ref, rend, resp, role, rows, scribe, target, type, url, value, who, whole, xml:base, xml:id,

Requirements of a User-Friendly, General-Purpose Corpus Query Interface

Jan-Philipp Soehn[1], **Heike Zinsmeister**[2], **Georg Rehm**[1]

[1]Tübingen University
Sonderforschungsbereich 441
Nauklerstraße 35
72074 Tübingen, Germany

[2]Konstanz University
Department of Linguistics
Fach D 185
78457 Konstanz, Germany

Abstract

This article reports on a survey that was conducted among 16 projects of a collaborative research centre to learn about the requirements of a web-based corpus query interface. This interface is to be created for a collection of corpora that are heterogeneous with respect to their languages, levels of annotations, and their users' research interests. Based on the survey and a comparison of three existing corpus query interfaces we compiled a set of requirements. In the context of sustainable strategies of corpus storage and accessibility we point out how to design an interface that is general enough to cover multiple corpora and at the same time suitable for a wide range of users.

1. Introduction

Immense amounts of corpus data have been created in recent years. The process of building a language resource is expensive, time-consuming, and it includes aspects such as corpus sampling and linguistic annotation on multiple levels. There is an urgent need to ensure that researchers are able to access data collections such as these beyond the lifetime of the project that created the resource. Issues of sustainability and preservation are increasingly important to the community; see, for example, Bird and Simons (2003), Trilsbeek and Wittenburg (2006), Dipper et al. (2006) as well as efforts such as OLAC (`http://www.language-archives.org`), E-MELD (`http://emeld.org`), and metadata aggregators such as the Digital Repository Infrastructure for European Research (`http://www.driver-repository.eu`).

One major aspect of sustainability is perpetuating access to corpora independently of project duration, availability of the researchers who built the resource, and development cycles of operating systems, tools, and applications. There is a great danger of a language resource turning into an expensive data graveyard if the tools for accessing, displaying, and searching the resource become obsolete or if there is no proper documentation available for the respective data collection (Bird and Simons, 2003; Schmidt et al., 2006).

A straightforward way out of this problem is to adhere to a particular annotation and encoding standard so that only one common interface needs to be supported for accessing a whole range of resources (Lehmberg and Wörner, In print; Rehm et al., 2007; Rehm et al., 2008a; Rehm et al., 2008b; Witt et al., 2007; Zinsmeister et al., In print). The availability of such an interface would lead to two new challenges. First, due to the diversity of information that needs to be accessed, the interface must be general enough to cover multiple corpora with heterogeneous annotation and it must be specific enough to enable users to find the information they are looking for. Second, due to the diversity of potential users, the query interface has to be designed to favour high acceptability. Such a user interface should assist users who cannot be expected to be experts in composing queries in, for example, a formal query language that is based on first-order logic. At the same time the interface should do justice to the experienced user and support efficient data access. Thus, alternative approaches have to be explored to facilitate accessing and querying linguistic resources for a heterogeneous group of users.

The goal of this article is twofold. On the one hand we outline a set of general requirements for a sustainable corpus query interface, on the other we report on ongoing work of implementing such a general-purpose linguistic query interface for a set of heterogeneous corpus resources. Both efforts build upon a survey conducted among 16 projects of the German collaborative research centre 441 at Tübingen University supplemented by a qualitative analysis of three existing corpus interfaces which we take to be prototypical representatives of specific types of corpus interfaces. It is worth pointing out that we do not discuss query languages as such but take it for granted that a user-friedly interface is independent of the underlying query language. For surveys on the expressiveness of query languages see, for example, Lai and Bird (2004) or Dipper et al. (2007).

This article is structured as follows: In Section 2 the survey is reported. We present the results by aggregating the answers given to us by the project staff. Section 3 presents three existing corpus query interfaces, comparing and summarising their respective functions. In Section 4, we outline some of the requirements for the query interface that we collected based on the survey as well as from our analyses of the query interfaces. Section 5 gives a detailed overview of a corpus query interface that is currently under development. Its design is guided by the results of our studies from Sections 2 and 3. Finally, Section 6 rounds off this paper with a conclusion and an outlook on future work.

2. Survey of Requirements

This contribution reports on a survey we conducted to learn about the requirements of a web-based corpus query interface. This interface is to be created for a collection of corpora that are diverse with respect to their languages, levels of annotations, and research interests of the users, who, furthermore, come from several communities, each with their own standards and traditions (Witt et al., 2007; Rehm et al., 2007; Rehm et al., 2008a). Based on a questionnaire (Lehmberg et al., 2007, describe a related approach), we interviewed the research staff of 16 projects based in the

The respondents' areas of expertise	Computational Linguistics	6	30%
	German Language	3	15%
	Romance Languages	3	15%
	Slavic Languages	3	15%
	General Linguistics	2	10%
	English Language	1	5%
	Psycholinguistics	1	5%
	Tibetan Language	1	5%
	Among them with a specialisation in Language Acquisition: 2 and Semantics: 1		
Programming skills	yes: 45%		no: 55%
Data creation	involved: 75%		not involved: 25%
Age	<30: 30%	30–40: 45%	>40: 25%
Sex	female: 65%		male: 35%

Table 1: Demographic characteristics of the questionnaire respondents

collaborative research centre SFB 441 at Tübingen University concerning the question of how users are supposed to query the corpora they have created and what their suggestions for a query interface are. In total, twenty subjects answered the questionnaire. Table 1 contains demographics and lists a summary of the subjects' special fields, whether they have programming skills (in the sense of having the expertise to write scripts for data access on their own), and it also notes whether they were involved in compiling and annotating linguistic data themselves. Corpora created in these projects involve a collection of bilingual language acquisition data (Dieser, 2007), a collection of diachronic Romance corpora, a collection of Russian corpora including the Uppsala Corpus of Modern Russian, a collection of Bosnian, Serbian and Croatian data including the Novosadski Corpus of Spoken Language, a Tibetan Corpus (Wagner and Zeisler, 2004), a treebank of suboptimal structures (Sternefeld, 2004), and the German treebank TüBa-D/Z (Hinrichs et al., 2004). Some of the projects do not create their own data but use corpora either provided by other projects of the research centre or independently available resources such as corpora from the child language data exchange system CHILDES (MacWhinney, 1995) or the German treebanks TIGER (Brants et al., 2003) and TüBa-D/Z (Hinrichs et al., 2004).

We distinguished three functional areas in the questionnaire: *search*, *visualisation*, and *export of query results*. Concerning these areas the following open-ended questions were posed:

1. What kind of information will be requested by the user of your corpus (please give examples)?
2. Please give examples of frequent queries.
3. What is the input format of the query (text, XML, specialised query language, ...)?
4. What are your requirements on a query form (beyond a simple text-field and a search button)? Are there any online tools you consider suitable?
5. What will be the format in which search results are displayed? Are there existing websites that use this format?

The respondents took two dimensions into account. First, they referred to the specific annotation, metadata and requirements of the corpora created in their respective projects. Most of them did not generalise with regard to the questions on adequate formats of search results or the query interface. Second, they considered their research interests and their formal background as well as their computer literacy. The answers to the survey are extremely heterogeneous, ranging from rather short to very detailed answers. To illustrate their broad range consider, for example, the following two answers to question 2 on example queries. On the one hand we got

> FSQ-query for subject wh-movement: (E y (& (cat y D) (E z (& (cat z W-Pron) (≫ y z))) (E x (& (cat x Trace) (mor x nom) (move x y)))))

and on the other

> Find all accentuated adjectives!
> Find an activity verb in stative passive!

Table 2 contains a summary of the answers we received.

3. Existing Corpus Query Interfaces

In addition to the questionnaire we compare and summarise the functions of three corpus query interfaces that have been mentioned by respondents as suitable tools. In this way we can identify their features and components. These features were integrated into a requirements document (Rehm and Schonefeld, 2008) that specifies properties and functional areas of the query interface that is currently under development in the project Sustainability of Linguistic Data, a joint initiative of the Universities of Hamburg, Potsdam and Tübingen. The query interfaces that we examined as a complement to the questionnaire are COSMAS II, TIGERSearch, and ELAN, that can be conceptualised as three different types of corpus user interface. COSMAS II represents the general interface to query large amounts of textual data which takes into account positional, i. e., word-based annotation only. Other instances of this kind of interface are, for example, the web interface of the Corpus del Español (Davies, 2005), XSara the search tool accompanying the British National Corpus (`http://www.oucs.ox.ac.uk/rts/xaira/`), or the WordSmith tool (Scott, 2004). TIGERSearch goes beyond positional information and allows the user to query and display hierarchical annotation and distributional relations. Other examples of this kind of interface include the fsq tool (Kepser, 2003) and the Linguist's Search Engine (Resnik and Elkiss, 2005). ELAN is taken as a prototypical interface to multiple-layered annotated corpora which are organised according to a reference line. Related interfaces are provided by EXAKT (`http://www.exmaralda.org/exakt.html`) the search tool of EXMARaLDA (Schmidt, 2004).

The three example interfaces are all parts of highly accepted and widely used tools in their respective research communities. Only COSMAS II is implemented as a genuine online

1.	Information requested by the user	Words/lemmas, strings, patterns (regular expressions), part-of-speech tags, morphological/prosodic annotation, syntactic structures, metadata (about source, date, etc.), specific elements and attributes in the XML structure
2.	Examples of frequently used queries	Only project-specific responses were given ranging from structural dependencies ("cat1 dominates (word1 & pos1)") over regular expressions ("[zZ]avod[aueoy]m?i?") to very abstract natural-language queries ("find an activity verb in stative passive")
3.	Input format of the query	Text, graphical query interface (cf. TIGERSearch), macros or example queries as templates, FSQ
4 a.	Requirements on a query form	Display frequent queries, features: save and name queries, drop-down menus of all categories that can be searched for (this feature should be hideable)
4 b.	Existing online tools	Examples: COSMAS II (`http://www.ids-mannheim.de/cosmas2/`), CQP-Online (`http://www.ims.uni-stuttgart.de/projekte/CQPDemos/Bundestag/frames-cqp.html`), Corpus del Español (`http://www.corpusdelespanol.org`)
5.	Display format of search results	The following options should be available: text (with links to tree graphs or audio files), KWIC with hideable/adjustable context, syntactic structure (constituents in brackets), cross-sentence discourse structure, search history, structured text (XML, spreadsheet), export to HTML, etc.

Table 2: Summary of the answers to the questionnaire

interface, while TIGERSearch and ELAN require local installation. We do not intend to compare the interfaces in a contrastive way and to measure their pros and cons. This would not do justice to them because they are too heterogeneous in the features they offer. Instead we document how they deal with the three functional dimensions of *search*, *visualisation*, and *export of query results*, and take a user perspective in our presentation.

- *Interface I:* A user of COSMAS II (developed by the Mannheim Institute for German Language, `http://www.ids-mannheim.de/cosmas2/`) can confine his search on subcorpora guided by metadata. He can retrieve corpus data that contain target words or expressions. A client for MS Windows allows the user to create his queries in a graphical interface. A query is then composed by selecting graphical representations of search primitives (operators such as AND, PROXIMITY, etc.) and by specifying parameters. Alternatively, text can be used for the query, assisted by a help function and a wide range of parameters. The system documents the search history and allows the user to re-use previous queries easily. Hits are presented in KWIC format. The user gets information on type-token ratio including different options of time-based distribution and can retrieve statistics on collocations. Results can be re-used for a new search, for a co-occurrence analysis and they can be exported as RTF or ASCII.
- *Interface II:* The user of TIGERSearch (Lezius, 2002) is interested in syntactic structures realised in a treebank. In TIGERSearch only a single corpus is queryable at a given time. A corpus-specific info pane informs about its metadata. Just as in COSMAS II the user can choose between graphical or textual input. In the graphical interface the user can draw partial trees by clicking nodes and relations and choosing features from drop-down menus. Search queries are not stored automatically but can be saved by the user in a bookmark function. Results are displayed graphically with optional re-use for co-occurrence frequency listings. TIGERSearch offers various export options including XSLT filters and graphical export formats.
- *Interface III:* ELAN, the Linguistic Annotator developed in the European Distributed Corpora Project (`http://www.mpi.nl/world/tg/lapp/eudico/eudico.html`), can be used to annotate, to query and to visualise audio or video resources (`http://www.lat-mpi.eu/tools/elan/`). ELAN's search tool supports, among others, queries on multiple annotation layers, regular expressions, the specification of ranges, and a query history. ELAN visualises sound files in waveform format and provides export to CHAT, Praat, Tiger XML, HTML, CSV, interlinear text, and subtitles text.

4. General Requirements

In the following subsections, we outline requirements for a general query interface based on our findings on the questionnaire (Section 2) as well as from our analyses of existing query interfaces (Section 3).

4.1. Input Options

For the search function a text-field should be provided that supports Unicode encoding, given the need to accommodate non-Latin (e. g., Russian or Tibetan) scripts. Alternatively, it would be advantageous if the user interface contained a graphical tool to assemble a query based on predefined graphical objects that represent linguistic concepts. These building blocks should range, for example, from part-of-speech categories such as different types of nouns ("proper name", "inanimate object"), verbs ("ditransitive verb"), and prepositions, to grammatical functions ("genitive object"), or simply terminal and non-terminal nodes of a hierarchical structure, as well as to relations such as

dominance and precedence. This requisite is reported by our informants in their answers 1 and 3 in Table 2. Users of TIGERSearch and COSMAS II are used to this twofold way of formulating queries; which of the two modes is most appropriate depends on the user's preferences as well as on the type of query that is conducted.

4.2. Search Functions

The search function should be able to address primary data, multiple levels of annotation, and metadata. Frequent queries should be available as examples, represented both in a graphical and textual way, so that users who are not familiar with corpus query languages can use and modify them in order to explore the system capabilities as well as to arrive quickly at queries that are useful for their own research questions. This is further supported by a mapping of graphical queries into the textual query language syntax.
In addition, a query form would be desirable for experienced users who would like to edit the underlying query formula directly. Though the interface is independent of a specific query language, we suggest to use XQuery, a language for finding and extracting elements and attributes from XML data, analogous to what SQL is for relational databases. XQuery is built on XPath expressions and standardised by the World Wide Web Consortium. It is rather easy to learn for an XML-experienced user and deployable in a broader range of applications. Moreover, the possibility to manipulate XQuery queries most directly meets the requirement to search for specific elements and attributes in the XML structure. Thus, XQuery is the obvious choice when it comes to picking a query formalism for XML-based linguistic resources.
Furthermore, a search history and a function to save and load queries (i. e., a kind of bookmark function) should be available just as in TIGERSearch (see row 4 a in Table 2). Lastly, a summary of all available search criteria and constraints, displayed via drop-down menus or similar means would help the user in composing a query. For example, in COSMAS II, search operators with an intuitive description are displayed prominently within the search window and allow users to drag them into the search pane.

4.3. Visualisation

The query interface should cover linguistic patterns in a large and heterogeneous set of language resources. For the purpose of querying and visualising a corpus, all resources should be mapped onto abstract corpus types for type-specific query and visualisation methods. For example, the results for one specific corpus type are displayed as hyperlinked matches in a KWIC format, for another type as matrix of annotation layers, or as hierarchical tree structures. There should be functions that allow the user to include or exclude several layers of information in the display, such as complete sentences, information on words, or cross-sentence discourse annotation. In addition, the amount of visible context to the left and to the right should be customisable and there should be an option of enlarging the match up to a whole paragraph with cross-sentence annotation. Detailed tree structures that provide clickable nodes, and secondary/tertiary edges should be available where appropriate in suitable formats (e. g., SVG). Appropriate export formats (ODF, Excel, TXT, XML, HTML, etc.) are demanded by the researchers who participated in our questionnaire, both for the query results and for user-specified subsets of a corpus. An ID list of hits would be a useful feature to locate a particular result quickly. Statistical functions (frequencies, co-occurrences, mean utterance length, type-token ratio) analogous to COSMAS II complete the desirable functionality of the query interface.
This concludes our overview of the basic requirements. Certainly, we did not do justice to all of the features of COSMAS II, TIGERSearch, and ELAN but focused on the main properties relevant for a general query interface.

5. Our Corpus Query Interface

We are currently developing a corpus query interface for a sustainability web platform (see Section 1). The development process is completely guided by and based upon requirements that we collected in a survey (Section 2) and that we extracted from the feature sets of several existing and widely used corpus query tools (Section 3). Initially we made a design decision and introduced a basic distinction that separates between querying for *corpus metadata* and querying for *corpus data*, i. e., corpus contents, so that we can tailor and fine-tune the respective functions.
A user has to login first. From here, the user can either go to the saved queries area or explore the available metadata records. There are several different options how the metadata can be displayed, sorted, and searched (for example, by corpus type, by organisation or project, by properties such as number of tokens, or by the respective research question a corpus was created for). The implementation of this part of the interface is based on Java Server Pages and operates on a relational database due to performance and security considerations (Rehm et al., 2008b).
As soon as the user has decided upon one or more resources, the corpus contents of these collections can be queried using an intuitive graphical query interface that generalises as much as possible from the underlying data structures and querying methods actually used. The system employs Ajax technologies (Asynchronous JavaScript and XML) so that a dynamic, interactive, drag-and-drop-enabled query interface can be provided. An ontology of linguistic annotations (Rehm et al., 2008a) enables us to provide abstract representations of linguistic concepts (e. g., *noun*, *verb*, *preposition* etc.) that may have a specific set of features; operands can be used to glue together the linguistic concepts by dragging and dropping these graphical representations onto a specific area of the screen, building a query step by step. We also provide several output and visualisation modules for query results, e. g., queried corpus subsets that contain syntactic trees can be visualised as trees, and data that is modelled using a timeline-based approach is displayed in a tabular fashion.
Among other functions, the interface provides a graphical tree fragment query editor that allows the user to submit complex queries for retrieving those particular syntactic structures from the currently selected resources that match the tree fragment query. Queries are interpreted and translated into XQuery internally. When the interface is in tree

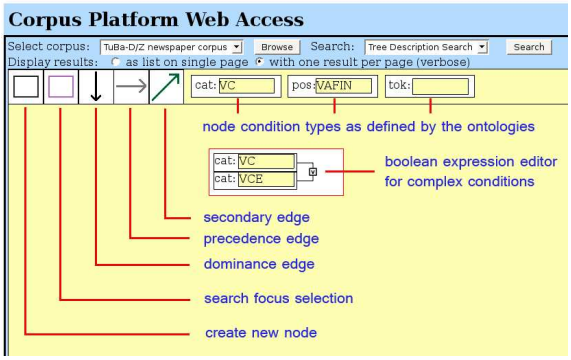

Figure 1: The tree fragment query editor

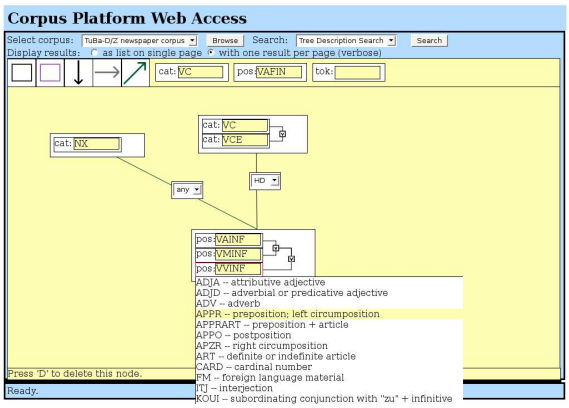

Figure 2: The tree fragment query editor

fragment query mode (see Figures 1 and 2), the user can drag and drop components of a query onto an assembly pane, so that queries can be constructed in a step-by-step fashion. Currently, nodes can be combined by dominance, precedence, and secondary edge relations. The structures defined by these graphs mirror the structures to be found

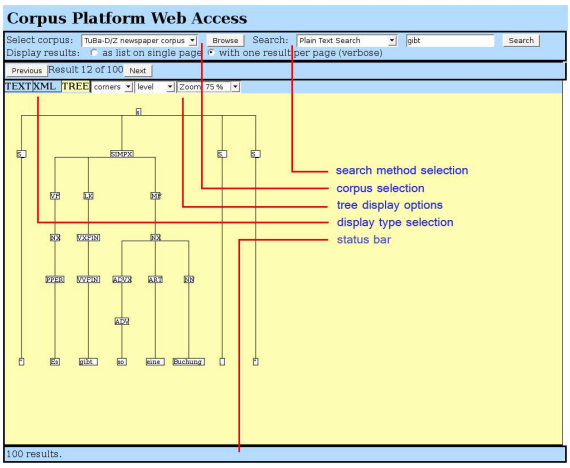

Figure 3: The front-end in tree display mode

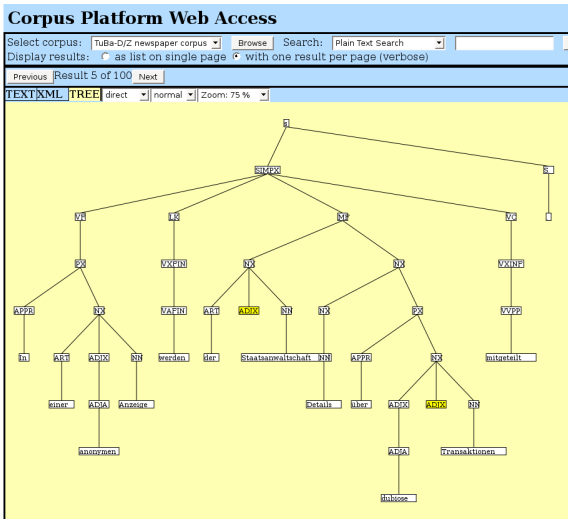

Figure 4: Browsing a corpus (yellow nodes are collapsed)

by the XQuery engine of the native XML database that we use. A node may contain one or more conditions linked by boolean connectives that help to refine the node classes a specific query is supposed to match. Tree fragment queries are not the only type of queries allowed by the front-end. It also supports plain text and regular expression queries. Experienced users can formulate their queries in XQuery directly, or they can fine-tune queries initially generated graphically. Our aim is to give the user a variety of options for viewing and exploring results. Four different major display modes are already implemented: plain text view, XML view, graphical tree view and timeline view (see Figures 3 and 4). It should be noted that figures 1 to 4 do not represent the final look of the graphical query interface. The environment is still work in progress – its design will be finalised in the autumn of 2008. Rehm et al. (2008a) provide a detailed description of the corpus query interface and several related components such as the interaction between the XQuery engine and the ontology.

6. Concluding Remarks and Future Work

In this article, we presented requirements of a corpus query interface which have been compiled based on two sources: a survey among linguists that regularly consult corpora and also create corpora themselves and an analysis of existing applications for corpus querying. This approach turned out to be a suitable and effective way to accumulate a number of important and useful requirements for our own query interface. We consider it an additional advantage that users of established software will recognise some popular features in our interface and will not be confronted with entirely new paradigms and metaphors.

The survey and analysis presented here is associated with the project "Sustainability of Linguistic Data" which is still work in progress. We want to highlight some of the aspects that we plan to put into effect by the end of 2008. In addition to the ongoing corpus normalisation and metadata transformation work (Rehm et al., 2008b), most rele-

vant for the results of our survey is the continuous implementation of the metadata exploration interface and of the graphical visualisation and querying front-end (Rehm et al., 2008a). We plan to upgrade and enhance several aspects of the GUI. Next to a substantial design overhaul of the interface in order to improve its usability, we will integrate graphical query templates and saved searches that act like bookmarks in a web browser. For their representation we will use an XML-based format to store all necessary data in one place. Moreover, we will integrate functions for multi-layer querying as well as for the visualisation of multi-layer annotations, and we will finalise the ontology-based query expansion component. We plan to finish work on the GUI as well as on the whole platform by September.

Acknowledgments

The research presented in this paper was supported by a grant from *Deutsche Forschungsgemeinschaft* within the project *Nachhaltigkeit linguistischer Daten*. The authors would like to thank Hanan Bechara and Johannes Dellert (Tübingen University) for implementing significant parts of the user interface and Lucas Ogden for proofreading.

7. References

S. Bird and G. Simons. 2003. Seven dimensions of portability for language documentation and description. *Language*, 79:557–582.

S. Brants, S. Dipper, P. Eisenberg, S. Hansen-Schirra, E. König, W. Lezius, C. Rohrer, G. Smith, and H. Uszkoreit. 2003. TIGER: Linguistic Interpretation of a German Corpus. *Journal of Language and Computation*.

M. Davies. 2005. Advanced research on syntactic and semantic change with the Corpus del Español. In et al. C. Pusch, editor, *Romance Corpus Linguistics II: Corpora and Diachronic Linguistics*, pages 203–214. Narr, Tübingen.

E. Dieser. 2007. Early language separation: A longitudinal study of a Russian-German bilingual child. In S. Featherston and W. Sternefeld, editors, *Roots: Linguistics in Search of its Evidential Base*, pages 133–160. Mouton de Gruyter.

S. Dipper, E. Hinrichs, T. Schmidt, A. Wagner, and A. Witt. 2006. Sustainability of Linguistic Resources. In *Proc. of the LREC 2006 Satellite Workshop Merging and Layering Linguistic Information*, pages 48–54, Genoa, Italy, May.

S. Dipper, M. Götze, U. Küssner, and M. Stede. 2007. Representing and Querying Standoff XML. In G. Rehm, A. Witt, and L. Lemnitzer, editors, *Data Structures for Linguistic Resources and Applications*, pages 337–346. Narr, Tübingen.

E. Hinrichs, S. Kübler, K. Naumann, H. Telljohann, and J. Trushkina. 2004. Recent developments of Linguistic Annotations of the TüBa-D/Z Treebank. In *Proc. of TLT*.

S. Kepser. 2003. Finite Structure Query - A Tool for Querying Syntactically Annotated Corpora. In *Proc. of the EACL 2003*, pages 179–186, Budapest, Hungary.

C. Lai and S. Bird. 2004. Querying and updating treebanks: A critical survey and requirements analysis. In *Proc. of the Australasian Language Technology Workshop*, pages 139–146, Sydney, Australia.

T. Lehmberg and K. Wörner. In print. Annotation Standards. In A. Lüdeling and M. Kytö, editors, *Corpus Linguistics*, Handbücher zur Sprach- und Kommunikationswissenschaft (HSK). de Gruyter, Berlin, New York.

T. Lehmberg, C. Chiarcos, E. Hinrichs, G. Rehm, and A. Witt. 2007. Collecting Legally Relevant Metadata by Means of a Decision-Tree-Based Questionnaire System. In *Digital Humanities 2007*, pages 164–166, Urbana-Champaign, IL, USA, June. ACH, ALLC, Graduate School of Library and Information Science, University of Illinois, Urbana-Champaign.

W. Lezius. 2002. *Ein Suchwerkzeug für syntaktisch annotierte Textkorpora*. Ph.D. thesis, University of Stuttgart.

B. MacWhinney. 1995. *The CHILDES-Project: Tools for Analyzing Talk*. Erlbaum, Hillsdale, NJ, 2 edition.

G. Rehm and O. Schonefeld. 2008. Specification of the Sustainability Platform. Internal Specification and Technical Report. SFB 441, University of Tübingen.

G. Rehm, R. Eckart, and C. Chiarcos. 2007. An OWL- and XQuery-Based Mechanism for the Retrieval of Linguistic Patterns from XML-Corpora. In *Int. Conf. Recent Advances in Natural Language Processing (RANLP 2007)*, pages 510–514, Borovets, Bulgaria, September.

G. Rehm, R. Eckart, C. Chiarcos, and J. Dellert. 2008a. Ontology-Based XQuery'ing of XML-Encoded Language Resources on Multiple Annotation Layers. In *Proc. of the 6th Language Resources and Evaluation Conf. (LREC 2008)*, Marrakech, Morocco, May.

G. Rehm, O. Schonefeld, A. Witt, T. Lehmberg, C. Chiarcos, H. Bechara, F. Eishold, K. Evang, M. Leshtanska, A. Savkov, and M. Stark. 2008b. The Metadata-Database of a Next Generation Sustainability Web-Platform for Language Resources. In *Proc. of the 6th Language Resources and Evaluation Conf. (LREC 2008)*, Marrakech, Morocco, May.

P. Resnik and A. Elkiss. 2005. The Linguist's Search Engine: An Overview. In *Proc. of the ACL Interactive Poster and Demonstration Sessions 2005*, University of Michigan, USA.

T. Schmidt, C. Chiarcos, T. Lehmberg, G. Rehm, A. Witt, and E. Hinrichs. 2006. Avoiding Data Graveyards: From Heterogeneous Data Collected in Multiple Research Projects to Sustainable Linguistic Resources. In *Proc. of the E-MELD 2006 Workshop on Digital Language Documentation: Tools and Standards – The State of the Art*, East Lansing, Michigan, June.

T. Schmidt. 2004. Transcribing and annotating spoken language with EXMARaLDA. In *Proc. of the LREC-Workshop on XML based richly annotated corpora, Lisbon 2004*, Paris. ELRA.

M. Scott. 2004. *WordSmith Tools*. Oxford University Press, Oxford.

W. Sternefeld. 2004. Stylebook for the German Treebank of the A3 Project. Technical report, Universität Tübingen. `http://tusnelda.sfb.uni-tuebingen.de/sinbad/Stylebook/stylebooknew.pdf`.

P. Trilsbeek and P. Wittenburg. 2006. Archiving Challenges. In J. Gippert, N. P. Himmelmann, and U. Mosel, editors, *Essentials of Language Documentation*, pages 311–335. Mouton de Gruyter, Berlin, New York.

A. Wagner and B. Zeisler. 2004. A syntactically annotated corpus of Tibetan. In *Proc. of LREC 2004*, pages 1141–1144, Lisbon, Portugal, May.

A. Witt, O. Schonefeld, G. Rehm, J. Khoo, and K. Evang. 2007. On the Lossless Transformation of Single-File, Multi-Layer Annotations into Multi-Rooted Trees. In B. T. Usdin, editor, *Proc. of Extreme Markup Languages 2007*, Montréal, Canada, August.

H. Zinsmeister, A. Witt, S. Kübler, and E. Hinrichs. In print. Linguistically Annotated Corpora: Quality Assurance, Reusability and Sustainability. In A. Lüdeling and M. Kytö, editors, *Corpus Linguistics*, Handbücher zur Sprach- und Kommunikationswissenschaft (HSK). de Gruyter, Berlin, New York.

Sustainability of Text-Technological Resources

**Maik Stührenberg[1], Michael Beißwenger[2], Kai-Uwe Kühnberger[3], Harald Lüngen[4]
Alexander Mehler[1], Dieter Metzing[1], Uwe Mönnich[5]**

[1]Universität Bielefeld, [2]Technische Universität Dortmund, [3]Universität Osnabrück,
[4]Justus-Liebig-Universität Gießen, [5]Universität Tübingen

Abstract

We consider that there are obvious relationships between research on sustainability of language and linguistic resources on the one hand and work undertaken in the Research Unit "Text-Technological Modelling of Information" on the other. Currently the main focus in sustainability research is concerned with archiving methods of textual resources, i.e. methods for sustainability of primary and secondary data; these aspects are addressed in our work as well. However, we believe that there are additional certain aspects of sustainability on which new light is shed on by procedures, algorithms and dynamic processes undertaken in our Research Unit.

The Research Unit "Text-Technological Modelling of Information"

The Research Unit 437 "Text-Technological Modelling of Information" is funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) and consists of five projects.[1] The funding started in 2002, since October 2005 the Group is in its second period lasting until late 2008. Before we will go into detail about the observations we made about relationships between research on sustainability of language and linguistic resources and the work carried out in our Research Unit, we will first introduce the projects that take part in it, followed by some preliminary distinctions regarding certain aspects of sustainability (Section 1.).

A2: Sekimo The topic of the project Sekimo is the integration of heterogenous linguistic resources which can be divided into annotated textual documents on the one hand and grammars, parsers, lexicons or ontologies – amongst others – on the other hand. The project focuses on the application and integration of the latter ones on raw texts or pre-annotated documents. For accomplishing this task two architectures have been developed: a Prolog fact based approach, developed in the first period of the project and described in detail in Witt (2004), and an XML-based approach which makes use of the Sekimo Generic Format (SGF) in conjunction with a native XML database system (Stührenberg and Goecke, 2008). Both architectures can be used to examine relationships between modelling units derived from different annotation layers without the need for markup unification – which could leed to overlapping problems. The exemplifying application of these architectures focuses the analyses of anaphoric relations which are of high relevance for projects in the inner context of the research group (e.g. when measuring the value of anaphoric relations as cues for rhetorical relations) and for external cooperations. The corpus under investigation is based on the corpus of German scientific articles of the C1 (SemDok) project and was extended with German newspaper texts. All texts are annotated with multiple annotation layers, including a document structure layer (developed by the projects B1 and C1), a morphological and syntactic layer (provided by the commercial tagger software Machinese Syntax from Connexor Oy, which is used in other projects of the Research Group as well), a discourse entity layer (automatically generated), and the cohesive layer containing the semantic relations. For the latter task the web-based annotation tool Serengeti has been developed (Stührenberg et al., 2007), which allows for both high quality and quantity annotation of texts carried out by a large number of users. In addition the comparison of annotations on the same text made by different users is possible as well. Ongoing work on Serengeti includes the support of the SFG and the possibility of user-defined annotation schemas.

A4: Indogram The topic of the project Indogram is the automatic induction of probabilistic document grammars as models of hypertext types or web genres, respectively. It develops an algorithm for learning the internal structure of web documents as instances of web genres (e.g. conference websites, personal academic home pages or weblogs). The project starts from the idea that an appropriate web genre model gets its validity to the degree to which it clarifies the relation of *explicit* (*visible*) or *manifesting website* structure and *implicit* (*hidden*) or *manifested web genre* structure. Thus, beyond tagging genre labels to websites or pages the learning of genre-related web document structures is a major project goal. A central observation which makes this kind of structure mining a challenge beyond classical approaches to text mining is that hypertext graphs of the same type are distributed according to a multidimensional power-law (Mehler et al., 2007b). Thus, there are no typical page-based web genre structures so that classical approaches to structure learning cannot be applied. In order to solve this task various structure-related classifiers were developed which operate on the level of textual (Mehler et al., 2007a) and hypertextual (Mehler, 2008) structures. They show

[1]More information can be observed at `http://www.text-technology.de/`

that classifiers of structure come into reach with a very low space and a moderate time complexity. Further, A4 has developed a two stage model of hypertext types which combines an SVM tagger of genre constituents with a HMM of their networking. For this task, A4 explores *structural* features (on the level of the logical document structure), *lexical* features (including named entities) and HTML features. Thus, A4 has built a classifier which utilizes heterogeneous linguistic resources. Further, in order to master the dynamics and semantic diversity of websites, A4 has developed an approach to topic labeling based on social ontologies and, thus, combines content and structure mining. This model has been utilized for implementing a prototype for hypertext zoning, that is, an algorithm which delimits websites no longer in terms of physical (URL-related) features, but in terms of their content and function.

B1: HyTex On the basis of a corpus of documents from two scientific domains (hypertext and text technology)[2], project B1 develops and evaluates innovative strategies for handling conceptual problems of the so-called text-to-hypertext conversion. The approach is coherence-based (Kuhlen, 1991) and aims at generating hypertext views which provide the selective reader instant access to all textual units that he or she may need for a proper understanding of the current hypertext node and, thus, make selective reading and browsing more efficient and more convenient than would be possible with print media (Lenz and Storrer, 2006; Storrer, 2008). The strategies developed in the project process markup information from different annotation layers. XML document grammars have been developed for:

- the document structure layer (applying an annotation scheme derived from DocBook, which was developed in cooperation with project C1; cf. Lenz and Lüngen (2004));
- the terms and definitions layer, on which occurrences as well as definitions of technical terms in the documents are annotated (Storrer and Wellinghoff, 2006; Lenz et al., 2006; Wellinghoff, 2006);
- the thematic structure layer (applying an annotation scheme based on the typology of thematic progression according to Zifonun et al. (1997) §C6; cf. Lenz and Storrer (2003));
- the cohesion layer, on which various types of text-grammatical information are annotated (e.g. co-reference, connectives, text-deictic expressions; Holler (2003a; Holler (2003b; Holler et al. (2004)).

Additional linguistic information was provided by morphosyntactic annotations automatically assigned by the KaRoPars technology (Müller, 2004).
Besides coherence-based strategies for segmentation and linking on the document level, the HyTex approach also comprises strategies for providing hypertext users with navigation devices that support the reconstruction of domain-specific knowledge as well as thematical orientation while browsing the hypertext version of a scientific document:

- On the one hand and with special respect to the needs of users which are "semi-experts" in a certain domain, the project built up TermNet, a WordNet-style semantic net which describes the technical terminology specific to the respective domains (Beißwenger, 2008). On the presentation level, TermNet is used for generating glossary views that are linked to the term occurrences in the corpus.
- On the other hand, the project develops topic-based linking strategies which use a GermaNet-based lexical chaining approach as a resource (cf. (Cramer and Finthammer, 2008)) and which aim at generating topic views – thematic indices based on text-grammatical information constructed of a selection of topic items – as an additional navigation and orientation device.

C1: SemDok The goal of the SemDok project is to develop a text parser (discourse parser) for a complex text type in the framework of Rhetorical Structure Theory (RST, Mann and Thompson (1988); Marcu (2000)). The linguistic features for discourse interpretation of scientific research articles are firstly derived from a discourse marker lexicon with about 100 entries encoding features (e.g. induced relation, directionality, discourse segment level, frequency, and others) of lexical discourse markers (i.e. conjunctions and discourse adverbs). Secondly, a development corpus of German research journal articles was compiled and subsequently annotated on several levels of linguistic analysis corresponding to the output of pre-processing components for logical document structure analysis (Lenz and Lüngen, 2004), text type structure analysis (Bärenfänger et al., 2006), initial discourse segmentation (Lüngen et al., 2006a), and lexical discourse marker annotation. Each level was added as a separate XML annotation layer in the framework of XML-based multi-layer annotation (Witt, 2004), with document grammars formulated as XML schemas. Several articles were also annotated according to RST-HP, which is the XML application that serves as the target structure of the SemDok parser (Lüngen et al., 2006b). It utilises the XML document tree to represent an RST discourse tree. The SemDok hierarchy of rhetorical relations called "RRSet" is an adaptation of previously suggested relation taxonomies to the analysis of scientific research articles. It is formalised in the web ontology language OWL, as an extension of the work described in (Goecke et al., 2005), cf. Bärenfänger et al. (2007). The perl program rs3tohp converts manual annotations of RST structure built with the RSTTool by O'Donnell (2000)

[2]The corpus documents (in their "raw" and annotated versions) are freely available at `http://www.hytex.info/030_ergebnisse/020_korpus`.

into the RST-HP format. Additionally, morphological and syntactic annotations are provided using the already mentioned commercial tagger MACHINESE SYNTAX. The development corpus and its annotations will be made available as soon as the relevant legal issues are clarified with the publisher of the research articles.

C2: Ontologies Text technologically based information modelling is confronted with two main problems for which there are no solutions in formal linguistics. On the one hand, there is the phenomenon of markup-structures which are despite of their character similar to trees not presentable in classical techniques of tree grammars. This lack of presentability is due to the possibility of unbounded branchings and the appearance of secondary relations. On the other hand, the dynamic aspect of web-oriented ontologies is a challenge which can not be refuted by the means of methods of dynamic logics and their linguistic incarnation. The C2 project attempts to extract ontological knowledge from syntactically given information (coded in annotation graphs), in order to expand ontologically coded information. On the syntactic side, a major goal of the project is the logical and complexity theoretic characterization of certain types of annotation graphs, such as the well-known Bird-Liberman graphs (Bird and Liberman, 2001). In Michaelis and Mönnich (2007), the authors present results for characterizing a large class of annotation trees, namely, single time line, multiple tiers (STMT) models, which constitute a subclass of annotation graphs in the sense of Bird and Liberman, and from which multi-rooted trees can be constructed. On the semantic side, it is a matter of fact that automatic as well as semi-automatic procedures for the expansion of ontologies (triggered by information coded in annotation graphs) contain errors of different types. These errors can range from structural and user-defined inconsistencies to logical contradictions (Haase and Stojanovic, 2005). In a series of papers, members of the C2 project provide algorithmic solutions for resolving automatically certain types of logical inconsistencies in ontology design relative to various types of description logics (cf. (Ovchinnikova and Kühnberger, 2006a; Ovchinnikova et al., 2007; Ovchinnikova and Kühnberger, 2007).

1. Overview and Preliminary Distinctions

We consider that there are obvious relationships between research on sustainability of language and linguistic resources on the one hand and work undertaken in the Research Unit "Text-Technological Modelling of Information" on the other. Furthermore, we see new relationships that merit to be explored in more detail. An important aspect of sustainability research is the long-term availability of resources, either for basis research or for applications. Sustainability research has many facets, the following aspects may be distinguished: aspects related to primary data, secondary data and category systems (cf. Section 2.); aspects related to procedures for these data and category systems (cf. Section 3.); aspects related to process properties in a long-term perspective cf. Section 4.); and aspects related to a community of experts and non-experts agreeing to work with shared standards (cf. Section 5.). Work undertaken in the Research Unit is concerned with these aspects with the exception of aspects related to process properties, which presupposes an organizational framework in order to guarantee sustainability, for example the constant actualization of the process organization and the continuous adjustment to changed goals and basic conditions.

2. Data and Category Systems

2.1. Sustainability of Primary and Secondary Data

The building and usage of corpora are important aspects of text-technological research. Examples of corpora that were built in the Research Unit and which are partially available online were provided by all projects. For assuring sustainability of primary and secondary data, the usage of XML (in favour of proprietary formats) can be considered as key issue. XML-based modelling of information has been the base line of the Research Unit. This is reflected both by the explicit usage of XML in the A2 (Sekimo) project and the usage of XML representations for coding document and discourse structure and metadata in other projects. The explicit usage of XML in the Sekimo project consists of several format descriptions for annotation schemas (both in XML DTDs and XML Schema Descriptions, XSD) and results in the development and implementation of the generic representation format SGF (SEKIMO GENERIC FORMAT) as basis for a generic architecture (in conjunction with a database backend, either native XML or relational, cf. Stührenberg et al. (2006)). The XSD-based SGF consists of a base layer that uses a standoff approach (Thompson and McKelvie, 1997) for storing multiple annotated data. An arbitrary number of annotation layers, separated via distinct namespaces, can be imported into the base layer (Stührenberg and Goecke, 2008). In contrast to similar approaches such as the pivot format of the LAF (Linguistic Annotation Framework, cf. Ide et al. (2003)) or PAULA (Potsdamer Austauschformat für linguistische Annotation, cf. Dipper (2005)), SGF uses only a single file to store multiple annotations on a single or even multiple files and imports all sorts of annotation schemas (including graph based). The format is designed to stick as accurately as possible to the imported annotation layer – including the possibility of validating its content – and uses standard XQuery (instead of introducing extensions to already established standards, cf. Alink et al. (2006)) for analyzing relationships between elements derived from different layers. A description of possible relations is given by Witt et al. (2005).

Further, XML-based formats for poly-hierarchical hypertext structures as well as document networks (Mehler and Gleim, 2006) were developed in the Indogram project, including text internal structures down to the level of dependency trees (Pustylnikov and Mehler, 2008; Pustylnikov et al., 2008). A basic requirement

of structure-oriented annotations is the flexibility and adaptability of the annotation format in use. This requirement has been met by further implementing a graph theory-related format in conjunction with a text-technological database operating thereon in the Indogram project (Gleim et al., 2007a), and – in the Sekimo project – by the before mentioned generic representation format SGF and its employment in conjunction with native XML databases.
Project C2 chose to represent automatically extracted lexical-semantic patterns in a standard description logic format (which can be considered as a syntactic variant of OWL).
Standards for metadata are considered to be a necessity for text technological applications, this applies to the research carried out in our Research Unit, too. Recommendations regarding the use of metadata standards (Dublin Core, cf. DCMI Usage Board (2006) OLAC, cf. Simons and Bird (2003) and IMDI, cf. IMDI (ISLE Metadata Initiative) (2003) the latter one for multimodal corpora) were given in an examination of several annotation standards for structuring and representing textual corpora (DocBook, TEI, CES and XCES) carried out in the Sekimo project (Stührenberg, 2007). OLAC metadata is used both in the projects Sekimo (exclusively, imported into the SGF) and SemDok (in addition to newly developed metadata sets). Options guaranteeing an open access to our corpora (e.g. Open Access[3] or Creative Commons[4]) are still under discussion, however, as stated in the description of the projects, some corpora are available to the public at present.

2.2. Sustainability of Category Systems

Category systems (like ontologies) can be used as a mediator between different sets of annotation (Schmidt et al., 2006). However, one has to assure that the category system is sustainable as well. One way to guarantee sustainability of category systems is the introduction of standardized formal ontologies specifying the categories and, hopefully, mediating between different category systems. The overall (and long-term) goal of this mediation is interoperability between different ontological resources. Among the standards that are used to ensure sustainability of linguistic and text-technological resources, ontologies and the Web Ontology Language OWL (W3C Web Ontology Working Group, 2004) play an important role. OWL is a W3C recommendation, and as an XML application, it offers a standardised formalism. As an ontology language it allows for a formal description of the semantics of XML tag sets. There are two ways in which ontologies are employed to ensure aspects of sustainability:

1. Reference ontologies as instruments to ensure the sustainability of linguistic tag sets.
   Linguistic or text-technological resources (corpus annotations) are made interoperable by mapping the category sets employed in them (annotation schemes) onto a formal ontology that has been introduced as a proposal for standardisation in the domain. An example of such an ontology is GOLD (Farrar and Langendoen, 2003). Moreover, the reference ontology developed in the SFB 441[5] is designed to link domain ontologies that represent the syntax and morphology annotation schemes of three different research projects (Chiarcos, 2007).
2. Construction of ontological resources.
   Linguistic resources other than annotations, especially lexical-semantic resources are represented (pro-actively, or by retroactive conversion) in a standardised ontology formalism, such as the OWL versions of the Princeton WordNet (van Assem et al., 2006).

With respect to these two directions, several domain-specific ontologies have been developed in the research unit: Regarding the first point, existing ontological standards for linguistic domains such as GOLD currently provide universal concepts for morphological and syntactic categories. Those categories however, for which resources were constructed in the research unit, are mostly found on the textual levels of linguistic analysis, i.e. above syntax and morphology, e.g. logical text objects, discourse entities, discourse relations, co-reference, lexical chains, topic chains, and text type structure categories. Presently, no ontological standardisations of discourse categories are available, but within the research unit, a proposal for a an ontology of discourse units and relations (rhetorical relations and anaphoric relations) as an extension of the GOLD approach has been put forward (Goecke et al., 2005).
In order to research ways of making general-language and domain-specific wordnets interoperable, different aspects of modelling wordnets in OWL have been researched in a cooperation of the C1 (SemDok) and the B1 (HyTex) project. In doing so, several resources have been constructed in OWL: Firstly, the terminological wordnet TermNet, in which terminology from domains of hypertext research and text technology is represented on the basis of an wordnet model that has been extended for terminologies (Beißwenger, 2008; Selzam, 2008). Secondly, the GermaNet resource; thirdly, an integrated version of TermNet and a subset of GermaNet that have been connected in OWL via so-called plug-in relations (Lüngen and Storrer, 2007; Kunze et al., 2007; Lüngen et al., 2008).[6] Other domain-specific ontologies that have been developed in the Research Unit are a framework for integrating lexical ontologies as a resource of semantic annotation of documents (Goecke et al., 2007b; Mehler et al., 2007c) and a lexical-semantic ontology based on automatically extracted patterns from heterogeneous resources, where a special focus concerns the integration of primary data into one

[3] http://www.open-access.net/
[4] http://creativecommons.org/
[5] http://www.sfb441.uni-tuebingen.de/c2/
[6] All OWL resources from this B1/C1/GermaNet cooperation have been made available on the web under

homogeneous database of hypotheses (Krumnack et al., 2007). A remaining challenge is the development of procedures for adapting and merging different ontologies (cf. Section 3.).

2.3. Availability of Methods and Tools

Often a sustainable use of methods and tools is prevented by the fact that documentation or source code is not made available to the public. In case of our Research Unit, documentations of markup specifications developed for diverse layers of linguistic annotation are already available online: e.g. in terms of thematic structure (Lenz and Storrer, 2003), coreference phenomena (Holler, 2003a; Holler, 2003b; Holler et al., 2004), term definitions in text (Storrer and Wellinghoff, 2006; Lenz et al., 2006; Wellinghoff, 2006), and an annotation schema for annotating anaphoric relations (Goecke et al., 2007a). The sources of the web-based annotation tool SERENGETI[7] (Stührenberg et al., 2007) will be made publicly available online under the GPL (GNU Public License) before the end of the Research Unit together with the corresponding documentation. Further, the ARIADNE SYSTEM for the development, maintenance and statistical analysis of large-scale multimodal corpora[8] (Gleim et al., 2007a), the SCIENTIFIC DESKTOP for the semantic analysis of web documents[9] (Waltinger et al., 2008) and the WEBCEP SYSTEM[10] (Gleim et al., 2007b) for the development and maintenance of web genre corpora are available online.

3. Procedural Aspects

Procedural aspects of sustainability are based on the idea that long-term archiving of text-technological resources cannot be reduced to a static saving of data.

3.1. Learning and Induction of Ontological Systems

The hand-coded development of ontologies, relevant for the interoperability of category systems, is a tedious, time-consuming, and expensive task. Therefore automatic procedures for the extraction of knowledge and the learning of ontologies play a central role now and in the future. Inductive methods from machine learning for explorative data analysis and the build-up of ontologies and text technological resources were developed (Mehler et al., 2007c; Waltinger et al., 2008). Architectures for the integration of heterogenous linguistic resources and partial solutions for the automatic extraction of heterogeneous data sources and the transformation of these data in a format that allows uniform querying were developed as well (Krumnack et al., 2007; Goecke et al., 2007b).

http://www.wordnets-in-owl.de.

[7] http://coli.lili.uni-bielefeld.de/serengeti/

[8] http://varda.coli.uni-bielefeld.de:8080/Ariadne/

[9] http://www.scientific-workplace.org

[10] http://ariadne.coli.uni-bielefeld.de:8080/WikiCEP/

3.2. Dynamics and Consistency Preservation of Ontological Resources

Due to the fact that sustainability of textual data requires the possibility of extensions of ontological resources, consistency problems of such resources can be considered as a central problem. Several types of problems can be distinguished and were partially solved: overgeneralizations of concepts in case that nonmonotonic extensions of the underlying data basis are necessary (Ovchinnikova and Kühnberger, 2006a), undergeneralizations of concepts (Ovchinnikova and Kühnberger, 2006b), and polysemy problems (Ovchinnikova and Kühnberger, 2007). These results were tested on example ontologies (Ovchinnikova et al., 2007). Following the state-of-the-art to code ontological knowledge in description logics results in the task to develop different resolution algorithms relative to the chosen description logic. Due to the fact that different description logics have different expressive strengths and different constructors can be used to form new concepts, there is no easy way to expand a known algorithm for a certain DL to another more expressive DL. Resolution strategies for certain types of inconsistencies can be provided for mild extensions of the attributive language family.

3.3. Formal Properties of Data Structures

Only a formally precise characterization of the underlying data structures of repositories ensures that algorithmic solutions can be found for sustainability questions and the comparability of different data formats. Purely structural properties of trees and graphs can be learnt in the framework of quantitative structure analysis and graph kernel methods which are successfully used to classify document types (Dehmer and Mehler, 2007; Mehler et al., 2007a). A logical characterization of annotation graphs as well as a constructive procedure in order to algorithmically transform the annotation graphs developed by Bird & Liberman (Bird and Liberman, 2001) into multi-rooted trees was developed by (Michaelis and Mönnich, 2007). In Mönnich and Kühnberger (2007), this approach is embedded into a broader context of text technological research. Connected with the formal specification of the underlying coding formats with respect to their complexity theoretic properties are import-export interfaces for standards and representation formalisms like RDF, OWL (in its different versions), SKOS etc.

4. Process Perspective

In a process perspective of sustainability the interrelationships between different actors or institutions may be focussed upon. This may be illustrated by four actors and their relationships introduced by Gary Simons in his talk at the conference "Processing Text-Technological Resources" at Bielefeld (Germany) in March this year.[11] First, there is the *creator* who

[11] The short abstract is available at http://coli.lili.uni-bielefeld.de/Texttechnologie/Forschergruppe/PTTR/abstracts/Abstract-Simons.pdf

brings a resource into existence, preferably according to aspects introduced in Section 2.. Second, there is the *curator* or *archiver* who takes on the responsibility to sustain the necessary conditions for use, preferably according to aspects introduced in Section 3., especially interoperability. Third, there is the *aggregator* who takes care of the web-accessibility of resources archived at different places, preferably according to advanced search procedures, i.e. according to aspects introduced in Section 3.. Forth, there is the *user* expecting that resources of interest to him are and will be discoverable; or there is a *community* of users, introduced in Section 5., that may influence the process of sustainability at different stages.

5. Community of Experts and Non-Experts

A central aspect of sustainability is the existence of a community and a complex network of valuable cooperation, i.e. a community agreeing to work with shared standards as well as with procedures of interoperability, accepting the web-based collaboration with experts and non-experts and exploiting the web as a field of global cooperation (Simons, 2007). An example in this sense is the cooperation of the Sekimo project with the international projects "Anaphoric Bank"[12] and "AnaWiki" (Poesio and Kruschwitz, 2008) contributing to its corpora, representation format and the architecture of the before mentioned web-based annotation tool SERENGETI. A special wiki that supports scientific collaboration with respect to the exchange and maintenance of treebanks[13] was implemented in the Indogram project (Pustylnikov and Mehler, 2008), and by means of a scientific desktop[14] parts of the procedural output of our research were made available.

6. Conclusion

As stated by Simons (2007) and in Section 1. the concept of sustainability has many facets. The work undertaken in the Research Unit "Text-Technological Modelling of Information" account for most of them and proposes additional aspects of sustainability regarding procedures, algorithms and dynamic processes. In addition, the cooperation between different projects in the Research Unit, the use of shared corpora, methods and tools has prevented multiple implementations and reduced the overall amount of work in these fields. Apart from the projects that take part in the Research Unit the publicly available access to most of the documentation and tools can help other interested projects in the same way and can contribute to a community building process.

7. References

Wouter Alink, Raoul Bhoedjang, Arjen P. de Vries, and Peter A. Boncz:. 2006. Efficient XQuery Support for Stand-Off Annotation. In *Proceedings of the 3rd International Workshop on XQuery Implementation, Experience and Perspectives, in cooperation with ACM SIGMOD*, Chicago, USA, Juni.

Maja Bärenfänger, Mirco Hilbert, Henning Lobin, Harald Lüngen, and Csilla Puskàs. 2006. Cues and constraints for the relational discourse analysis of complex text types – the role of logical and generic document structure. In Candy Sidner, John Harpur, Anton Benz, and Peter Kühnlein, editors, *Proceedings of the Workshop on Constraints in Discourse*, pages 27–34. National University of Ireland, Maynooth, Ireland.

Maja Bärenfänger, Henning Lobin, Harald Lüngen, and Mirco Hilbert. 2007. Using OWL ontologies in discourse parsing. In Kai-Uwe Kühnberger and Uwe Mönnich, editors, *OTT'06 - Ontologies in Text technology: Approaches to Extract Semantic Knowledge from Structured Information. Series Publications of the institute of Cognitive Science (PICS) 1*, Osnabrück.

Michael Beißwenger. 2008. TERMNET — ein terminologisches Wortnetz im Stile des Princeton Wordnet. Technical report of the B1 Project. `http://www.hytex.info/`.

Steven Bird and Mark Liberman. 2001. A Formal Framework for Linguistic Annotation. *Speech Communication*, 33(1–2):23–60.

Christian Chiarcos. 2007. An Ontology of Linguistic Annotation: Word Classes and Morphology. In *Proceedings of DIALOG 2007, Bekasovo/Moscow.*

Irene Cramer and Marc Finthammer. 2008. An Evaluation Procedure for Word Net Based Lexical Chaining: Methods and Issues. In *Proceedings of the Global WordNet Conference 2008, Szeged, Hungary.* `http://www.inf.u-szeged.hu/projectdirs/gwc2008/`.

DCMI Usage Board. 2006. DCMI Metadata Terms. DCMI Recommendation, Dublin Core Metadata Initiative.

Matthias Dehmer and Alexander Mehler. 2007. A New Method of Measuring the Similarity for a Special Class of Directed Graphs. *Tatra Mountains Mathematical Publications*, 36:39–59.

Stefanie Dipper. 2005. XML-based Stand-off Representation and Exploitation of Multi-Level Linguistic Annotation. In *Proceedings of Berliner XML Tage 2005 (BXML 2005)*, pages 39–50, Berlin, Deutschland.

Scott Farrar and D. Terence Langendoen. 2003. A Linguistic Ontology for the Semantic Web. *GLOT International*, 7(3):97–100.

Rüdiger Gleim, Alexander Mehler, and Hans-Jürgen Eikmeyer. 2007a. Representing and Maintaining Large Corpora. In *Proceedings of the Corpus Linguistics 2007 Conference, Birmingham (UK).*

Rüdiger Gleim, Alexander Mehler, Matthias Dehmer, and Olga Pustylnikov. 2007b. Aisles through the Category Forest – Utilising the Wikipedia Category System for Corpus Building in Machine Learning. In Joaquim Filipe, José Cordeiro, Bruno Encarnação, and Vitor Pedrosa, editors, *3rd International Conference on Web Information Systems and Technologies (WEBIST '07), March 3-6, 2007, Barcelona*, pages 142–149, Barcelona.

Daniela Goecke, Harald Lüngen, Felix Sasaki, Andreas Witt, and Scott Farrar. 2005. GOLD and Discourse: Domain- and Community-Specific Extensions. In *Proceedings of the E-MELD Workshop on Morphosyntactic Annotation and Terminology: Linguistic Ontologies and Data Categories for Language Resources*, Cambridge, Massachusetts.

[12] `http://www.anaphoricbank.org`

[13] `http://ariadne.coli.uni-bielefeld.de/wikis/treebankwiki/index.php5/Main_Page`

[14] `http://www.scientific-workplace.org`

Daniela Goecke, Anke Holler, and Maik Stührenberg. 2007a. Koreferenz, Kospezifikation und Bridging: Annotationsschema. Technical report of the A2 Project.

Daniela Goecke, Maik Stührenberg, and Tonio Wandmacher. 2007b. Extraction and Representation of Semantic Relations for Resolving Definite Descriptions. Extended Abstract. In Uwe Mönnich and Kai-Uwe Kühnberger, editors, *OTT'06. Ontologies in Text Technology: Approaches to Extract Semantic Knowledge from Structured Information*, volume 1-2007 of *Publications of the Institute of Cognitive Science (PICS)*, pages 27–32. Institute of Cognitive Science, Osnabrück, January.

P. Haase and L. Stojanovic. 2005. Consistent evolution of OWL ontologies. In *Proceedings of the Second European Semantic Web Conference*, pages 182–197, Lissabon.

Anke Holler, Jan-Frederik Maas, and Angelika Storrer. 2004. Exploiting Coreference Annotations for Text-to-Hypertext Conversion. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 651–654, Lissabon.

Anke Holler. 2003a. Koreferenz in Hypertexten: Anforderungen an die Annotation. *Osnabrücker Beiträge zur Sprachwissenschaft*, 68:9–29.

Anke Holler. 2003b. Spezifikation für ein Annotationsschema für Koreferenzphänomene im Hinblick auf Hypertextualisierungsstrategien. Technical report of the B1 Project. `http://www.hytex.info/`.

Nancy Ide, Laurent Romary, and Eric de la Clergerie. 2003. International Standard for a Linguistic Annotation Framework. In *Proceedings of HLT-NAACL'03 Workshop on The Software Engineering and Architecture of Language Technology*, Edmunton.

IMDI (ISLE Metadata Initiative). 2003. Metadata Elements for Session Descriptions. version 3.0.4. Reference Document, MPI, Nijmegen, October.

Ulf Krumnack, Ekaterina Ovchinnikova, and Tonio Wandmacher. 2007. LexO – constructing a lexical ontology from heterogenous resources. In *Proceedings of the OntoLex'07 Workshop at ISWC*, Busan, Korea.

Rainer Kuhlen. 1991. *Hypertext. Ein nicht-lineares Medium zwischen Buch und Wissensbank.* Springer, Berlin.

Claudia Kunze, Lothar Lemnitzer, Harald Lüngen, and Angelika Storrer. 2007. Repräsentation und Verknüpfung allgemeinsprachlicher und terminologischer Wortnetze in OWL. *Zeitschrift für Sprachwissenschaft*, 26:267–290.

Eva Anna Lenz and Harald Lüngen. 2004. Dokumentation: Annotationsschicht: Logische Dokumentstruktur. Technical report of the B1 Project. `http://www.hytex.info/`.

Eva Anna Lenz and Angelika Storrer. 2003. Annotationsschicht: Thematische Strukturen/Themenentwicklung. Technical report of the B1 Project. `http://www.hytex.info/`.

Eva Anna Lenz and Angelika Storrer. 2006. Generating hypertext views to support selective reading. In *Digital Humanities 2006. Conference Abstracts. Paris-Sorbonne, 5–9 Juli 2006*, pages 320–323. `http://www.hytex.info/050_publikationen/LenzStorrer87.pdf`.

Eva Anna Lenz, Michael Beißwenger, and Sandra Wellinghoff. 2006. Annotationsschicht: Definitionen und Termverwendungsinstanzen. Technical report of the B1 Project. `http://www.hytex.info/`.

Harald Lüngen and Angelika Storrer. 2007. Domain ontologies and wordnets in OWL: Modelling options. *LDV Forum*, 22(2):1–19.

Harald Lüngen, Henning Lobin, Maja Bärenfänger, Mirco Hilbert, and Csilla Puskàs. 2006a. Text Parsing of a Complex Genre. In *Proceedings of the Conference on Electronic Publishing (ELPUB)*, pages 247–256, Bansko, Bulgaria.

Harald Lüngen, Csilla Puskàs, Maja Bärenfänger, Mirco Hilbert, and Henning Lobin. 2006b. Discourse Segmentation of German Written Text. In *Proceedings of the 5th International Conference on Natural Language Processing (FinTAL 2006)*, pages 245–256, Åbo, Finland. Springer.

Harald Lüngen, Claudia Kunze, Lothar Lemnitzer, and Angelika Storrer. 2008. Towards an Integrated OWL Model for Domain-specific and General Language WordNets. In *Proceedings of the 4th Global Wordnet Conference (GWC 2008)*, pages 281–296.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical Structure Theory: Toward a functional theory of text organisation. *Text*, 8(3):243–281.

Daniel Marcu. 2000. *The Theory and Practice of Discourse Parsing and Summariszation.* MIT Press, Cambridge MA.

Alexander Mehler and Rüdiger Gleim. 2006. The Net for the Graphs – Towards Webgenre Representation for Corpus Linguistic Studies. In Marco Baroni and Silvia Bernardini, editors, *WaCky! Working Papers on the Web as Corpus*, pages 191–224. Gedit, Bologna.

Alexander Mehler, Peter Geibel, and Olga Pustylnikov. 2007a. Structural Classifiers of Text Types: Towards a Novel Model of Text Representation. *LDV Forum*, 22(2):51–66.

Alexander Mehler, Rüdiger Gleim, and Armin Wegner. 2007b. Structural Uncertainty of Hypertext Types. An Empirical Study. In *Proceedings of the Workshop "Towards Genre-Enabled Search Engines: The Impact of NLP", September, 30, 2007, in conjunction with RANLP 2007, Borovets, Bulgaria*, pages 13–19.

Alexander Mehler, Ulli Waltinger, and Armin Wegner. 2007c. A Formal Text Representation Model Based on Lexical Chaining. In Peter Geibel and B. J. Jain, editors, *Proceedings of the KI 2007 Workshop on Learning from Non-Vectorial Data (LNVD 2007) September 10, Osnabrück*, pages 17–26, Osnabrück. Universität Osnabrück.

Alexander Mehler. 2008. Structural Similarities of Complex Networks: A Computational Model by Example of Wiki Graphs. *Appears in: Applied Artificial Intelligence.*

Jens Michaelis and Uwe Mönnich. 2007. Towards a Logical Description of Trees in Annotation Graphs. *LDV Forum*, 22(2):68–83.

Uwe Mönnich and Kai-Uwe Kühnberger, editors. 2007. *OTT'06. Ontologies in Text Technology: Approaches to Extract Semantic Knowledge from Structured Information*, volume 1-2007, Osnabrück, January. Institute of Cognitive Science.

Frank Henrik Müller. 2004. Stylebook for the Tübingen Partially Parsed Corpus of Written German (TüPP-D/Z). Technical report. `http://www.sfs.uni-tuebingen.de/tupp/dz/stylebook.pdf`.

Michael O'Donnell. 2000. RSTTool 2.4 – A Markup Tool for Rhetorical Structure Theory. In *Proceedings of the International Natural Language Generation Conference (INLG'2000)*, pages 253 – 256, Mitzpe Ramon, Israel.

Ekaterina Ovchinnikova and Kai-Uwe Kühnberger. 2006a. Adaptive ALE-TBox for Extending Terminological Knowledge. In A. Sattar and B. H. Kang, editors, *AI 2006. Proceedings of the 19th ACS Australian Joint Con-*

ference on Artificial Intelligence (LNAI 4304), Lecture Notes in Artificial Intelligence, pages 1111–1115. Springer.

Ekaterina Ovchinnikova and Kai-Uwe Kühnberger. 2006b. Aspects of Automatic Ontology Extension: Adapting and Regeneralizing Dynamic Updates. In M. Orgun, editor, *Advances in Ontologies. Proceedings of the Australasian Ontology Workshop (AOW 2006), Conferences in Research and Practice in Information Technology*, volume 72, pages 52–60.

Ekaterina Ovchinnikova and Kai-Uwe Kühnberger. 2007. Automatic Ontology Extension: Resolving Inconsistencies. *LDV Forum*, 22(2):19–33.

Ekaterina Ovchinnikova, Tonio Wandmacher, and Kai-Uwe Kühnberger. 2007. Solving Terminological Inconsistency Problems in Ontology Design. *International Journal of Interoperability in Business Information Systems (IBIS)*, (4):65–80.

Massimo Poesio and Udo Kruschwitz. 2008. Anawiki: Creating anaphorically annotated resources through web cooperation. Submitted to LREC 2008.

Olga Pustylnikov and Alexander Mehler. 2008. Towards a Uniform Representation of Treebanks: Providing Interoperability for Dependency Tree Data. In *Proceedings of First International Conference on Global Interoperability for Language Resources (ICGL 2008), Hong Kong SAR, January 9-11*.

Olga Pustylnikov, Alexander Mehler, and Rüdiger Gleim. 2008. A unified database of dependency treebanks. integrating, quantifying & evaluating dependency data. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC 2008)*, Marrakech (Morocco).

Thomas Schmidt, Christian Chiarcos, Timm Lehmberg, Georg Rehm, Andreas Witt, and Erhard Hinrichs. 2006. Avoiding Data Graveyards: From Heterogeneous Data Collected in Multiple Research Projects to Sustainable Linguistic Resources. In *Proceedings of the EMELD'06 Workshop on Digital Language Documentation: Tools and Standards: The State of the Art*, Lansing, Michigan.

Bianca Selzam. 2008. Modellierung des TermNet in OWL. Technical report of the B1 Project. `http://www.hytex.info/`.

Gary Simons and Steven Bird, 2003. *OLAC Metadata*. OLAC: Open Language Archives Community.

Gary Simons. 2007. Doing linguistics in the 21st century: Interoperation and the quest for the global riches of knowledge. In *Proceedings of the "Toward the Interoperability of Language Resources" Workshop, LSA Summer Institute*, Stanford University, July.

Angelika Storrer and Sandra Wellinghoff. 2006. Automated detection and annotation of term definitions in german text corpora. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, Genoa, Italy.

Angelika Storrer. 2008. Mark-up driven strategies for text-to-hypertext conversion. In *Linguistic Modelling of Information and Markup Languages*. Spinger, Dordrecht.

Maik Stührenberg and Daniela Goecke. 2008. Integrated Linguistic Annotation Models and their Application in the Domain of Antecedent Detection. *Submitted to Balisage 2008*.

Maik Stührenberg, Andreas Witt, Daniela Goecke, Dieter Metzing, and Oliver Schonefeld. 2006. Multidimensional Markup and Heterogeneous Linguistic Resources. In David Ahn, Erik Tjong Kim Sang, and Graham Wilcock, editors, *Proceedings of the 5th EACL Workshop on NLP and XML (NLPXML-2006): Multi-Dimensional Markup in Natural Language Processing*, pages 85–88, Trento. EACL.

Maik Stührenberg, Daniela Goecke, Nils Diewald, Irene Cramer, and Alexander Mehler. 2007. Web-based Annotation of Anaphoric Relations and Lexical Chains. In *Proceedings of the Linguistic Annotation Workshop (LAW)*, pages 140–147, Prag, Juni. Association for Computational Linguistics. `http://acl.ldc.upenn.edu/W/W07/W07-1523.pdf`.

Maik Stührenberg. 2007. Texttechnological Standards – An Overview. In Georg Rehm, Andreas Witt, and Lothar Lemnitzer, editors, *Datenstrukturen für linguistische Ressourcen und ihre Anwendungen. Data Structures for Linguistic Resources and Applications. Proceedings of the Biennial GLDV Conference 2007*, pages 157–166, Tübingen. Gunter Narr Verlag.

Henry S. Thompson and David McKelvie. 1997. Hyperlink semantics for standoff markup of read-only documents. In *Proceedings of SGML Europe '97: The next decade – Pushing the Envelope*, pages 227–229, Barcelona.

Marc van Assem, Aldo Gangemi, and Guus Schreiber. 2006. Conversion of WordNet to a standard RDF/OWL representation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06), Genoa, Italy*.

W3C Web Ontology Working Group. 2004. OWL Web Ontology Language. Set of seven specifications. W3C Recommendation, World Wide Web Consortium.

Ulli Waltinger, Alexander Mehler, and Gerhard Heyer. 2008. Towards automatic content tagging: Enhanced web services in digital libraries using lexical chaining. In *4rd International Conference on Web Information Systems and Technologies (WEBIST '08), 4-7 May, Funchal, Portugal*. Barcelona.

Sandra Wellinghoff. 2006. Manuelle Annotation definitorischer Textsegmente incl. Guidelines Phase i und ii. Technical report of the B1 Project. `http://www.hytex.info/`.

Andreas Witt, Daniela Goecke, Felix Sasaki, and Harald Lüngen. 2005. Unification of XML Documents with Concurrent Markup. *Literary and Lingustic Computing*, 20(1):103–116.

Andreas Witt. 2004. Multiple Hierarchies: New Aspects of an Old Solution. In *Proceedings of Extreme Markup Languages*.

Gisela Zifonun, Ludger Hoffmann, and Bruno Strecker. 1997. *Grammatik der deutschen Sprache*. deGruyter, Berlin/New York.

List of Authors

Michael Beißwenger 33
Dan Cristea 1
Marie-Luce Demonet 19
Alexis Dimitriadisa 9
Kai-Uwe Kühnberger 33
Marie-Hélène Lay 19
Harald Lüngen 33
Alexander Mehler 33
Dieter Metzing 33
Uwe Mönnich 33
Ionut Pistol 1
Georg Rehm 27
Jan-Philipp Soehn 27
Maik Stührenberg 33
Menzo Windhouwera 9
Heike Zinsmeister 27