Oliver Streiter
NUK, National University of Kaohsiung
Leonhard Voltmer, Isabella Ties and Verena Lyding
Eurac research, European Academy Bolzano

STRUCTURING TERMINOLOGICAL DATA: THE BISTRO PROPOSAL

Abstract

For more than 15 years the computer is employed in terminology work. In the beginning terminology systems have been simple programs for inserting and modifying data stored in a static format. During the last couple of years the potentialities of computer assistance became obvious. These days systems that offer the possibility to flexibly access, link and display information are much asked for. In this paper we will present the BISTRO approach for a dynamic terminology system. We will point out requirements that current implementations have to tackle, provide a comparison with alternative approaches and discuss the strengths of our implementation.

1 INTRODUCTION

When managing complex terminological data, the limitations of term-banks become apparent soon. The list of requirements is long: Terminological data should be unambiguous, economical, non redundant, modular and formalized. They have to be transparent and reflect relevant differences in meaning or usage. They should serve different users with different information needs, different cultural and linguistic backgrounds. The data should be suited for on-line consultation with different protocols, the creation of CD-ROMs and paper dictionaries and should support the process of document creation, translation and management as good as possible. Term-banks should be embedded into a multi-user environment and support simultaneous consultation and updating. A term-bank failing on one or more of these criteria will sooner or later lead the terminologist into severe difficulties.

While older term-banks clearly cannot cope with these requirements, a closer look behind the scene of modern systems reveals that one of the main hindrances which impede progress in the field has not been addressed. Terminological data are still fragmented in so-called entries and emulate a file card system which dates from the early days of human writing. Although this model has obvious strong points as its modularity and its clear arrangement, further reflection will highlight its inadequacy to fulfill the above mentioned requirements. As shown in Streiter and Voltmer (2003a) and Streiter et al. (to appear), this model is not suited for the storage and insertion of data. It complicates or even prohibits their management and it makes a user-adaptive term presentation or presentation in different media, protocols or devices intolerably cumbersome.

Modern term-banks embrace XML, as XML promised to separate the storage and presentation of data and to provide a solution to the aforementioned requirements. At the same time, however, they stick to tree-like data structures, a design which reflects a fundamental misconception of human cognition and communication. Conceptual structures are circular networks. They are communicated however in tree-like structures, leaving it to the receiver to recompute the reentrances with the help of principles and conventions. This wisdom is embodied in almost all syntactic theories. They assume a non-tree-like underlying (conceptual) structure and map it on a tree-like surface structure (cf. Meaning <=> Text Theory, LFG, Relational Grammar, GB and its successors).

What may seem a theoretical subtleness may become dramatic when complex and valuable data are at stake. Using tree-like structures to store knowledge is like using a natural language expression for the storage and to hand over the task of understanding to the computer. But how can computers correctly recalculate the relations from a complex sentence like: "I gave Mary the flowers which I had bought 3 days before the birthday of her son, so that she could give them to him for me in time."?

Net structures have a promising future. Some the leading projects in the periphery of terminology, let's name WordNet, EuroNet, FrameNet, MindNet and RDF, have abandoned tree-like structures already a decade ago, c.f. Powers (2003). Tree-structured terminology deemed them a dead-end road.

Under these circumstances, EURAC investigated and implemented an alternative data design to overcome the limitations of entry-based terminology. In our BISTRO, the Juridical Terminological Information System Bolzano, project we handle legal terminology for Austria, Germany, Switzerland and Italy, with special focus on the legal and linguistic situation in South Tyrol, an autonomous province with 3 languages and legally binding translation relations between them.

The remainder of this paper is structured as follows. In Section 2 we will present our abstract data model. In Section 3 we will motivate the actual implementation and discuss the respective roles we assigned to the relational database and XML. Section 4 is dedicated to so-called term tools (term extraction, term recognition, text acquisition through agents, text classification, KWIC) and the status they have in our model. Section 5, finally will discuss matters of term presentation and how we think to orient the user in a richer and more informative environment. In the final conclusions we will come back to our initial claim and show that the break-up of an entry-based term model indeed can solve many of the problems current terminological work is struggling with.

2 DATA MODEL FOR TERMINOLOGY WITHOUT ENTRIES

The model is designed to flexibly make statements about entities which are relevant for a specific terminology. Statements are formalized as predicates. Unary predicates identify entities. E.g. <tt>term(213)</tt> states that <em>there is a term called 213</em>. Binary predicates state something about an entity. <tt>denomination(553,'legge provinciale')</tt> states that <em>there is a denomination 553 which is written as 'legge provinciale'</em>. In the same way predicates of higher order define entities and establish relations between different entity types.

Following these principles we can model a small fragment of a terminology by defining three predicates: <tt>grammar(665,'Nf')</tt>, <tt>denomination(553,'legge provinciale',665)</tt>, <tt>term(213,553)</tt>. The binary predicate <tt>grammar(665,'Nf')</tt> identifies a grammatical entity called 665 that describes a female noun. The trinary predicate <tt>denomination(553,'legge provinciale',665)</tt> thus states that <em>the denomination 553, which is spelled as 'legge provinciale', is a noun of the grammatical gender female</em>. In virtue of this <tt>term(213,553)</tt> states that<em>the term 213, spelled as 'legge provinciale', is a female noun</em>.

Using upper case characters (e.g. TERM, DENOMINATION, GRAMMAR) as variables for entity names and lower case characters for defining attribute values, we may formalize this simple model as shown below. This model can handle queries like '<em>Find all terms belonging to a specific grammatical class</em>' or vice versa <em>Find all grammatical classes belonging to a specific term</em>'.

Model I

```
term(TERM,DENOMINATION).
denomination(DENOMINATION,denomination,GRAMMAR).
grammar(GRAMMAR,grammar).
```

While such queries are within the scope of tree-like entry structures, queries like '<em>Find all terms which are assigned a legal domain which is different from that assigned to the text which contains the definition of the term.'</em> may already be beyond the expressive power of tree-like entry structures and require the intervention of a student programmer. In our systems, such queries are very much within the system and do not require other tools than a simple term search. Of course, many more queries may be handled by this model.

Model II

```
term(TERM,DENOMINATION,DEFINITION,LEGAL_DOMAIN).
legal_domain(LEGAL_DOMAIN,legal_domain).
denomination(DENOMINATION,denomination,GRAMMAR).
```

```
grammar(GRAMMAR,grammar).
definition(DEFINITION,CORPUS_SEGMENT).
corpus_segment(CORPUS_SEGMENT,LEGAL_DOCUMENT).
legal_document(LEGAL_DOCUMENT,LEGAL_DOMAIN).
```

An almost complete sketch of the model designed for BISTRO is reproduced as Model III. A graphical representation of it in terms of arcs and nodes is shown in Figure 1.

Model III

```
term(TERM,DENOMINATION,DEFINITION,LEGAL_DOMAIN, CONTEXT,LEGAL_SYSTEM).
term_relation_type(TERM_RELATION_TYPE, term_relation_type).
term_relation(TERM_RELATION,TERM1, TERM_RELATION_TYPE,TERM2).
translation(TRANSLATION,TERM1, NORMATION,TERM2).
normation(NORMATION,normation).
legal_domain(LEGAL_DOMAIN,legal_domain).
legal_system(LEGAL_SYSTEM,legal_system).
legal_hierarchy(LEGAL_HIERARCHY,legal_hierarchy).
legal_quality(LEGAL_QUALITY,legal_quality).
author(AUTHOR,author).
publishing_house(PUBLISHING_HOUSE, publishing_house).
publishing_place(PUBLISHING_PLACE, publishing_place).
title(TITLE,title)
denomination(DENOMINATION,denomination, GRAMMAR,LANGUAGE).
grammar(GRAMMAR,grammar).
language(LANGUAGE,language).
definition(DEFINITION,CORPUS_SEGMENT).
context(CONTEXT,CORPUS_SEGMENT).
corpus_segment(CORPUS_SEGMENT,LEGAL_DOCUMENT, paragraph,corpus_paragraph).
bi-lingual_corpus_segment(BI-LING_CORPUS_SEGMENT,
LEGAL_DOCUMENT1,paragraph1,corpus_paragraph1,LEGAL_DOCUMENT2,
paragraph2,corpus_paragraph2).
tri-lingual_corpus_segment(TRI-LING_ CORPUS_SEGMENT,LEGAL_DOCUMENT1,paragraph1,
corpus_paragraph1, LEGAL_DOCUMENT2,paragraph2,corpus_paragraph2,
LEGAL_DOCUMENT3,paragraph3,corpus_paragraph3).
legal_document(LEGAL_DOCUMENT,TITLE,url,
publication_date,passing_date,LANGUAGE,LEGAL_DOMAIN, LEGAL_SYSTEM,
LEGAL_HIERARCHY,LEGAL_QUALITY,AUTHOR,PUBLISHING_HOUSE, PUBLISHING_PLACE).
```

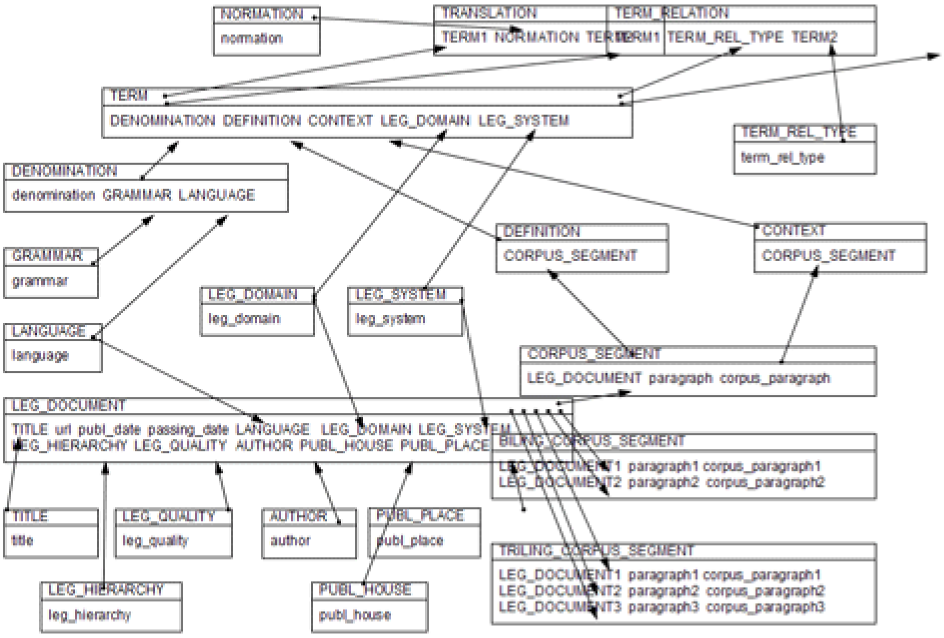

NORMATION
normation
TRANSLATION
TERM1 NORMATION TERM2
TERM_RELATION
TERM1 TERM_REL_TYPE TERM2
TERM
DENOMINATION DEFINITION CONTEXT LEG_DOMAIN LEG_SYSTEM
TERM_REL_TYPE
term_rel_type
DENOMINATION
denomination GRAMMAR LANGUAGE
DEFINITION
CORPUS_SEGMENT
CONTEXT
CORPUS_SEGMENT
GRAMMAR
grammar
LEG_DOMAIN
leg_domain
LEG_SYSTEM
leg_system
LANGUAGE
language
CORPUS_SEGMENT
LEG_DOCUMENT paragraph corpus_paragraph
LEG_DOCUMENT
TITLE url publ_date passing_date LANGUAGE LEG_DOMAIN LEG_SYSTEM
LEG_HIERARCHY LEG_QUALITY AUTHOR PUBL_HOUSE PUBL_PLACE
BILING_CORPUS_SEGMENT
LEG_DOCUMENT1 paragraph1 corpus_paragraph1
LEG_DOCUMENT2 paragraph2 corpus_paragraph2
TITLE
title
LEG_QUALITY
leg_quality
AUTHOR
author
PUBL_PLACE
publ_place
TRILING_CORPUS_SEGMENT
LEG_DOCUMENT1 paragraph1 corpus_paragraph1
LEG_DOCUMENT2 paragraph2 corpus_paragraph2
LEG_DOCUMENT3 paragraph3 corpus_paragraph3
LEG_HIERARCHY
leg_hierarchy
PUBL_HOUSE
publ_house

Figure 1: Modell III

Central to our model are terms. They are linked to a word or expression (DENOMINATION). This denomination bears grammatical information and language information. The term belongs to a legal system (e.g. Italy) and a legal domain (e.g. family law). These two attributes are also used to describe legal documents in a corpus. Legal documents are also described with respect to their legal hierarchy (international, national, regional, etc) and their legal quality (law, jurisdiction, treaty, text-book, etc). In addition, legal documents share a great number of bibliographic data, such as title, url, publication date etc. Segments of a legal document may serve as definition or as context for a term. The term and its context and definition should be consistent with respect to the language, the legal system and the legal domain.

Terms are linked to other terms via a relation like antonymy, synonymy, hyperonymy and meronymy on the one side and equivalence and partial equivalence on the other. While the first set of relations connects terms of the same legal system only (e.g. only Italian terms), the latter set is used to connect terms of different legal systems (e.g. Italian and Austrian terms, but also Austrian and German terms). As for the difficult case of legal systems including each other (EU includes Italy which includes the Autonomous Province of Bolzano) we had two options for coding the term relations. According to the first we assume an implicit equivalence and have no explicit connection of terms belonging to different hierarchies. The second option, which we finally followed, consists of marking the equivalence and non-equivalence of all terms explicitly. In fact, although as a general pattern terms from a lower hierarchy level are equivalent to the higher level, this may not be the case where the lower level enjoys a certain degree of autonomy. Terms of different languages and the same legal system (e.g. Swiss German and Swiss French) are also marked as equivalent or non-equivalent (**Figure 1**).

Without going too much into the details of this model, we notice a number of important differences between our model and tree-like entry structures. First, in our model we can cut out an arbitrarily large part of the network as answer to a query. Second, we can take any node of the network as starting point. We thus can do statements not only on terms, but on corpus segments, languages, definitions, normative statements, denominations, legal domains etc. Third, the combination of these two features allows for a flexible investigation and presentation of data.

For example, we can visualise and thus contrast the context citation of two terms which have a specific term relation, e.g. two synonyms, antonyms or hypo- and hyperonyms, compare the definitions of two equivalent terms in different legal systems, list the terms related to a legal domain, identify documents which support the normative statements etc, list false friends, and many more. All these specific queries are predefined in rules (e.g. two false friends have the same denomination but are not equivalent terms). The application of rules is triggered from hyperlinks (e.g. a "false friend"-button) and the output is rendered by specific style sheets, the "false friend"-style sheet.

One important rule identifies one term as a translation of the other. This is used for dictionary-like views on the term-bank. This rule selects all 'equivalent' terms belonging to different languages. It produces a set of descriptively adequate translations. To this set we join the normative translations as stipulated by normative bodies for the South-Tyrolean German and the two variants of Ladin, a Rhaeto-Romance language spoken in the Dolomites, in South-Tyrol.

Still other rules construct mono-, bi- and tri-lingual corpora, including the CATEX corpus of Italian legislation with its German translation (Gamper 1999) and the CLE corpus, an Italian, German and Ladin parallel corpus of regional and local legislation and administration (Streiter et al. 2004). Through the close linkage of terms and corpus, we do not only assist the descriptive and normative terminographical work, but prepare new perspectives on term-presentation. Thus, in near future we may have more than one contextual description of a term, each featuring a different aspect of the term or a dynamic linking of term and context, provided we will identify good heuristics for the ranking of contexts.

3 THE PROPER PLACE OF XML FOR TERM MANAGEMENT

Although XML and relational databases are equally suited to formalise the above model, there are a number of reasons to prefer relational databases, at least when data are written to or read from the hard disk.

Relational databases, be they commercial or free, safely handle simultaneous updates and retrievals, following the consistency requirements specified with the database. It is thus perfectly legitimate to update a term, for which another author creates a definition, while term, context and definition are seen by an external user (cf. Ahmad, K. & Holmes-Higgin, P. 1996). XML as a text-based medium is organized in large files. They can be opened for writing by one author only at a time.

Validation of data is more immediate with relational databases than it can be with text-based XML. With XML, data are first entered and than validated against a DTD. Although DTD-aware XML-editors might help, the validation of data then is left over to a device at the periphery of the system. All updates which do not go through the editor remain unvalidated until they shall eventually be visualized.

The notion of transaction which assures completed inserts or updates in relational databases is basically absent in pure XML. It has to be added in the form of a commercial transaction server, which takes over the managing of data in a system as complex as a relational database. With relational databases however we have free reliable resources with additional features like user management, network support, and fast retrieval with configurable indices.

The place we assign to XML is another. We use it as middle-ware between the database and the output. Thus while our data are stored in a network of relations, a query returns a table. Reentrances in the network have been pruned off (they are not included in the answer) or have been dissolved by multiplication (e.g. each term has its proper grammar field although derived from one single node originally). The columns in a table can be shown in different orders. We call the leftmost column the TOPIC and it is the TOPIC which determines the sorting of the data. The second column is called the FOCUS and it serves as secondary sorting key.

RELATIONAL NETWORK

```
term(1,5,9,7).
term(2,5,10,7).
term(3,5,11,8).
term(4,5,12,8).
legal_system(7,'AT').
legal_system(8,'DE').
denomination(5,'Kind',6).
grammar(6,N_m).
definition(9,14).
definition(10,15).
definition(11,16).
definition(12,17).
corpus_segment(14,'Ein Kind ...').
corpus_segment(15,'Das Kind ...').
corpus_segment(16,'Ein Kind im ...').
corpus_segment(17,'Kinder im ...').
```

IS ORGANIZED IN A NUMBER OF TABLES:

term id	denomination	definition	legal_system
1	5	9	7
2	5	10	7
3	5	11	8
4	5	12	8

legal_system

id	object
7	AT
8	DE

denomination

id	object	grammar
5	Kind	6

etc...

RELATIONAL DATABASE OUTPUT:

```
denomination legal_system  definition
========================================
Kind         AT            Ein Kind...
Kind         AT            Das Kind...
Kind         DE            Ein Kind im ...
Kind         DE            Kinder im ...
```

The tables returned by the relational database are converted into trees, collecting information belonging to the same TOPIC and FOCUS under the same node. An explicit annotation of the first and second column as topic and focus provides the further visualization of the data.

XML-STRUCTURE

```
  <terms>
    <denomination function='theme'>

      <object>Kind</object>
      <legal_system function='focus'>
         <object>AT</object>
         <definition function='rheme'>
            <object>Ein Kind...</object>

         </legal_system>
         <definition function='rheme'>
            <object>Das Kind...</object>
         </legal_system>
      </legal_system>
      <legal_system function='focus'>

         <object>DE</object>
         <definition function='rheme'>
            <object>Ein Kind im ...</object>
         </legal_system>
         <definition function='rheme'>

            <object>Kinder im ...</object>
         </legal_system>
      </legal_system>
    </denomination>
  </terms>
```

XHTML AFTER XSLT TRANSFORMATION

```
  <h1>TERMS</h1>
  <table>
   <tr>
    <td class='highlight'>

      <h2>DENOMINATION</h2>
                <ul>
          <li>Kind</li>
                </ul>
    </td>

    <td class='focus'>
      <table>
        <tr>
          <td class='focus'>
            <h3>LEGAL SYSTEM</h3>
```

```
              <ul>
                <li>AT</li>
                      </ul>
                  </td>
          <td class='rheme'>
            <h3>DEFINITION</h3>

                      <ul>
                <li>Ein Kind ...</li>
                <li>Das Kind ...</li>
                      </ul>
          </td>

        </tr>
        <tr>
          <td class='focus'>
            <h3>LEGAL SYSTEM</h3>
                      <ul>

                <li>DE</li>
                      </ul>
                  </td>
          <td class='rheme'>
            <h3>DEFINITION</h3>

                    <ul>
              <li>Ein Kind im ...</li>
              <li>Kinder im ...</li>
                    </ul>
          </td>

        </tr>
      </table>
     </td>
   </tr>
 </table>
```

Arbitrarily large parts of the network may be queried with pre-defined views which connect tables in a PROLOG-like syntax. A basic view, which features a term, a context and a definition with the bibliographic indications looks as follows:

```
view_term(TERM,DENOMINATION,LEGAL_DOMAIN,LEGAL_SYSTEM,
CORPUS_SEGMENT,TITLE,URL,PUBLICATION_DATE,PASSING_DATE,
PARAGRAPH,CORPUS_SEGMENT2,TITLE2,URL2,PUBLICATION_ DATE2,PASSING_DATE2,PARAGRAPH2):-

term(TERM,DENOMINATION,DEFINITION,LEGAL_DOMAIN, CONTEXT,LEGAL_SYSTEM),
definition(DEFINITION,CORPUS_SEGMENT),

corpus_segment(CORPUS_SEGMENT,LEGAL_DOCUMENT, PARAGRAPH,CORPUS_PARAGRAPH),

legal_document(LEGAL_DOCUMENT,TITLE,LEGAL_DOMAIN,
url,publication_date,passing_date,LANGUAGE,      LEGAL_SYSTEM,LEGAL_HIERARCHY,LEGAL_QUALITY,
AUTHOR,PUBLISHING_HOUSE,PUBLISHING_PLACE),
context(CONTEXT,CORPUS_SEGMENT2),

corpus_segment(CORPUS_SEGMENT2,LEGAL_DOCUMENT2, PARAGRAPH2,CORPUS_PARAGRAPH2),

legal_document(LEGAL_DOCUMENT2,LEGAL_DOMAIN2,
URL2,PUBLICATION_DATE2,PASSING_DATE2,LANGUAGE2,      LEGAL_SYSTEM2,LEGAL_HIERARCHY2,
LEGAL_QUALITY2, AUTHOR2,PUBLISHING_HOUSE2,PUBLISHING_PLACE2).
```

A classical terminological entry, composed of a term, its synonyms and equivalents is emulated as:

```
view_entry( ):-
view_term(TERM,DENOMINATION,LEGAL_DOMAIN,                    LEGAL_SYSTEM,CORPUS_SEGMENT,
TITLE,URL,PUBLICATION_DATE,PASSING_           DATE,PARAGRAPH,CORPUS_SEGMENT2,TITLE2,URL2,
PUBLICATION_DATE2, PASSING_DATE,PARAGRAPH2),
term_relation_types(TERM_RELATION_TYPE2,synonym),
term_relation(TERM_RELATION,TERM,TERM_RELATION_ TYPE2,TERM2),
view_term(TERM2,DENOMINATION2,LEGAL_DOMAIN2,                              LEGAL_SYSTEM2,
CORPUS_SEGMENT2,TITLE2,URL2,                 PUBLICATION_DATE2,PASSING_DATE2,PARAGRAPH2,
CORPUS_SEGMENT22,TITLE22,URL22,PUBLICATION_DATE22, PASSING_DATE22,PARAGRAPH22),
term_relation_types(TERM_RELATION_TYPE3,equivalent),
term_relation(TERM_RELATION,TERM,TERM_ RELATION_TYPE3,TERM3),
view_term(TERM3,DENOMINATION3,LEGAL_DOMAIN3,
LEGAL_SYSTEM3,CORPUS_SEGMENT3,TITLE3,URL3,
PUBLICATION_DATE3,PASSING_DATE3,PARAGRAPH3,            CORPUS_SEGMENT32,TITLE32,URL32,
PUBLICATION_DATE32,PASSING_DATE32,PARAGRAPH32).
```

4 BISTRO'S TOOLS FOR TERMINOGRAPHY

BISTRO provides a number of term-tools to insert and update data. Term tools are program-modules, the power of which goes beyond that of deductive SQL-statements in a given data-model, however, similar to the SQL-statements they start from a set of entities of the data model and arrive (possibly with human intervention) at another set of entities. The difference actually resides in the type of reasoning involved. While SQL-statements are purely deductive, term-tools allow for knowledge acquisition through induction and abduction. The purpose of term tools is thus to extend the terminological knowledge base on the basis of the available data in the model, new external data, e.g. in URL-located documents and plausibility statements.

In traditional terminography humans initiate and undertake every step of knowledge extension. With electronic term tools knowledge extension becomes much more effective, more consistent and data-driven, but it needs to be well controlled as induction and abduction are not necessarily reliable reasoning modes.

We will present here term extraction, term recognition, text classification and term acquisition through agents.

1. Term extraction
Term extraction identifies terms in text. The quality of the chosen terms prejudices the quality of the produced terminology. While humans usually consider one text at a time, computers can process large amounts of text and add information on absolute and relative frequency, structural resemblances between already described and not yet described terms, and find variants. Our term extraction starts from terms in the data-base, distills models from the terms and applies those models to text. An in-depth description of this tool is provided in "Example-based Term Extraction for Minority Languages: A case-study on Ladin" (Streiter et al. under http://dev.eurac.edu:8080/ autoren/publs/termex5.pdf). The term candidates are then ranked. At this point the quality has to be controlled: Either terminographers assess the automatically produced list of term candidates in context, or other control mechanisms are installed, e.g. only the best term candidate is processed further.

2. Term annotation or term recognition
Term recognition starts from terms and texts and tries to identify texts as contexts of terms or terms as contained in texts. This is a useful application of the terminology, because users can immediately see where a text uses described terminology. They do not loose time triggering searches, sometimes on the wrong lemma and sometimes without results. On the other hand term recognition is also a tool for terminographers, because they can immediately perceive how many terms of the text are already described. If a text contains a lot of already described terms, it is a good text for intensifying the description of a subject field. If a text contains few previously described terms, it is rather a text for extending terminography to other subject fields (which might not be in the scope of the project).

Furthermore, term recognition provides a ranking criterion for term descriptive contexts: A context that contains a lot of described terms is often more precise and gives a more typical context, but its comprehension also demands more domain specific expertise.

3. Text classification

A text classification tool starts from texts and a grid of meta-tags, e.g. language, subject domain, contents and tries to establish the relation between text and grid. Such labeling is necessary for a weighted corpus, for a more targeted searching in texts and helps automatic tools to improve in quality. Our text classification approach and first results are described in Streiter et al. (2003) and Voltmer et al. (2003). For maintaining quality and have nevertheless very early computer assistance in the classification task, we start out with manual classification. As soon as some documents have been classified, the system uses these classified documents to propose a classification of a new document. Humans only need to confirm or modify. Manual classification of data however is not necessary when a term-bank is available. As shown in Voltmer (2005), when combining the contexts and definitions of terms into pseudo-corpora, almost perfect automatic classification results can be obtained based on these pseudo-corpora.

Nevertheless, the classification task in legal terminology is particularly difficult. Lawyers need to know if a text is a valid norm, if it is a law or a court decision and if it is emanated by the Austrian or the German legal system. Even experts could hardly rely merely on the text for such classifications, e.g. humans rely on the authority of an editor, and computers count e.g. on the reliability of official Internet sites. Text taken from the site http://www.bundesverfassungsgericht.de containing the string "entscheidungen" in the URL will most probably contain a valid court decision of the German legal system. For this reason, automatic classification tools in legal terminology must take into account not only the content of the document as input parameter, but those data which describe the publication of the document. In our experience there is no loss in quality when the classification relies on the classification of "external corpora" (here: official legal databases) rather than on the classification of a single text.

In accordance with copyright, these corpora can even be imported to build an internal corpus. Such a classification allows a meta-search on external resources: A search string is sent to the appropriate selection of Internet sites (e.g. an Italian string only to sites with text in Italian) and the results of all relevant sites are reported. A user can then avoid searching in sites where the string is not present or too frequent and he also gets a first feedback on the frequency of the search-string and whether it belongs rather to normative or judicial language.

4. Term acquisition through agents

Given that programs can classify texts from the Internet through text classification, given that another program can recognize the terminological value of a text through term recognition, and given that a term extraction program finds new term candidates, the tools 1, 2 and 3 combine to an agent. This agent starts from certain criteria (language, legal system etc.) and returns with appropriate texts and terms. Agents can even find contexts and definitions. A context search is a full text search restricted to a class of documents (e.g. of the desired language and subject domain). Term recognition on the results ranks them by relevance or even subject domain relevance.

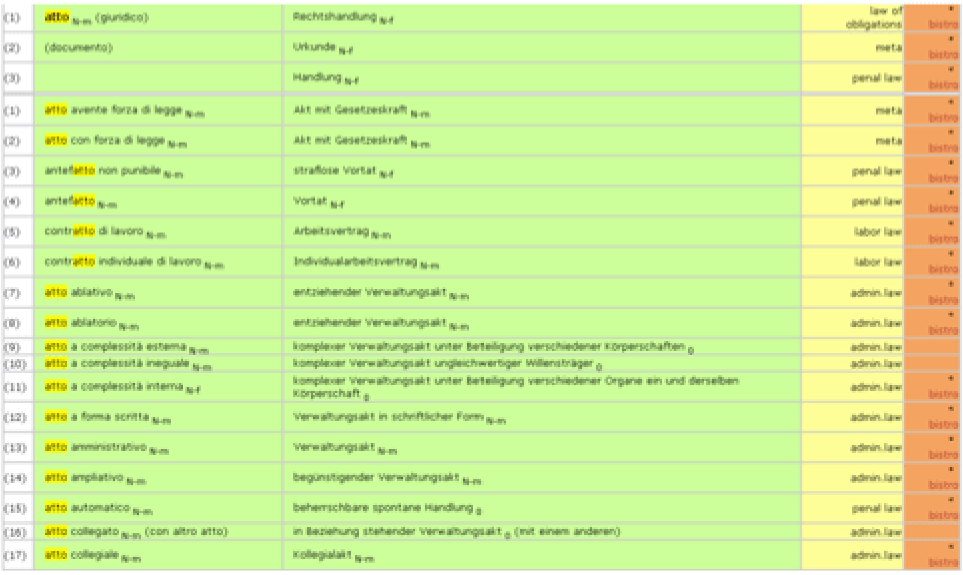
A definition search (implemented also by GOOGLE: "http://www.google.com/search?q=define:XXX") over legal resources is a context search with further restrictions. The search string is included in definition-indicating context. The term "A" would be searched with "The definition of A is", "A is defined as", "A is a", "A is the", and so on. Such a definition search does not allow for automatic collection of definitions, but according to our experience it is an extremely valuable tool for terminographers, because they can start out from various definition-similar contexts. Their attention is immediately drawn to polysemy, competing definitions in different contexts (legal systems) and to the different starting points for the definition (inside subject domain, inside LSP, definition with common language).

The term-tools can operate on their own output. They can be used to intensify the description of one subject domain, to extend the description to other domains and even to control the quality of the existing terminology. The described tools can indicate the contexts which have a low occurrence of domain specific terms and propose contexts with a higher occurrence of LSP-terms.

5 TERM PRESENTATION

This section will offer an overview of how we designed the interface and term presentation in order to guarantee a user-friendly work-platform and an intuitive presentation of instruments, tools and data. In an idealized description we could characterize the interface of BISTRO as a browser to the described data model and its extension through term tools. The user thus can approach the same data from different points of view, arriving after each hyperlink at a new node in the data model from which the data can be continued to be explored, zoomed in or seen with a different focus. In order to reduce the unexpected complexity, some of the possible paths have been pruned off and some 'standard' views are provided, e.g. the entrance page. At the top bar BISTRO's first page offers a series of all the tools that might be used: term search, corpus search, term tools and book search. Before starting the search, the term search and its presentation offer the possibility to choose between searching through all the terms normed by a terminology commission (button TERKOM) or to see all the data (button BISTRO). The search can be done in all three languages Italian, German and Ladin (in two variants).

The search returns a hit list, containing the searched terms and similar terms. That means that looking up the Italian word "atto" results in a list featuring in first column the searched term “atto”, followed by compound terms like "atto amministrativo", "atto collegato" etc. The second column contains the equivalents in target languages, in this case, German and Ladin. Clicking here checks the back-translation for translation validation. The third column refers to to the legal domain of the term. Clicking here sets the focus on normative documents or the legal domain or allows to zoom in on terms from a specific normative document or legal domain. The last column contains a number of links which elaborate the term in various directions (zoom on one terminological entry or validation/exploration with WWW-search and corpus queries) (figure 2). The zoom to the terminological entry returns a description of a term in three parts: the first containing domain (e.g. administrative law) and sub domain, registration number and responsible terminographers, the second containing the term, its definition and context including the relative bibliographical data. The third part finally contains the normative data which specify the status of the term, e.g. normed, refused or waiting for decision and in case of normation the information on the normative document issued by the region Trentino-South Tyrol. The entry is hyperlinked according to our data model, e.g. providing links to the bibliographical data to every single context or definition.



(1)	atto $_{\text{N-m}}$ (giuridico)	Rechtshandlung $_{\text{N-f}}$	law of obligations	* bistro
(2)	(documento)	Urkunde $_{\text{N-f}}$	meta	* bistro
(3)		Handlung $_{\text{N-f}}$	penal law	* bistro
(1)	atto avente forza di legge $_{\text{N-m}}$	Akt mit Gesetzeskraft $_{\text{N-m}}$	meta	* bistro
(2)	atto con forza di legge $_{\text{N-m}}$	Akt mit Gesetzeskraft $_{\text{N-m}}$	meta	* bistro
(3)	antefatto non punibile $_{\text{N-m}}$	straflose Vortat $_{\text{N-f}}$	penal law	* bistro
(4)	antefatto $_{\text{N-m}}$	Vortat $_{\text{N-f}}$	penal law	* bistro
(5)	contratto di lavoro $_{\text{N-m}}$	Arbeitsvertrag $_{\text{N-m}}$	labor law	* bistro
(6)	contratto individuale di lavoro $_{\text{N-m}}$	Individualarbeitsvertrag $_{\text{N-m}}$	labor law	* bistro
(7)	atto ablativo $_{\text{N-m}}$	entziehender Verwaltungsakt $_{\text{N-m}}$	admin.law	* bistro
(8)	atto ablatorio $_{\text{N-m}}$	entziehender Verwaltungsakt $_{\text{N-m}}$	admin.law	* bistro
(9)	atto a complessità esterna $_{\text{N-m}}$	komplexer Verwaltungsakt unter Beteiligung verschiedener Körperschaften $_{\text{g}}$	admin.law	
(10)	atto a complessità ineguale $_{\text{N-m}}$	komplexer Verwaltungsakt ungleichwertiger Willensträger $_{\text{g}}$	admin.law	
(11)	atto a complessità interna $_{\text{N-f}}$	komplexer Verwaltungsakt unter Beteiligung verschiedener Organe ein und derselben Körperschaft $_{\text{g}}$	admin.law	* bistro
(12)	atto a forma scritta $_{\text{N-m}}$	Verwaltungsakt in schriftlicher Form $_{\text{N-m}}$	admin.law	* bistro
(13)	atto amministrativo $_{\text{N-m}}$	Verwaltungsakt $_{\text{N-m}}$	admin.law	* bistro
(14)	atto ampliativo $_{\text{N-m}}$	begünstigender Verwaltungsakt $_{\text{N-m}}$	admin.law	* bistro
(15)	atto automatico $_{\text{N-m}}$	beherrschbare spontane Handlung $_{\text{g}}$	penal law	* bistro
(16)	atto collegato $_{\text{N-m}}$ (con altro atto)	in Beziehung stehender Verwaltungsakt $_{\text{g}}$ (mit einem anderen)	admin.law	
(17)	atto collegiale $_{\text{N-m}}$	Kollegialakt $_{\text{N-m}}$	admin.law	* bistro

Figure 2

The WWW-search of terms allows meta-search selected official internet sites on legal issues in Italy, Germany, Austria and Switzerland. Web-sites which have recently been used for the documentation of terms are preferred. Terminographers classify the sites to guide the WWW-search. By searching a term in CATEX (the bilingual Italian-German corpus) or in CLE (the trilingual Italian-German-Ladin corpus), BISTRO offers a special search mask that again permits the user to link his search, deciding to look up terms contained in documents edited before or after 2001, in specific types of documents or a specific legal system. The number of hits can be chosen, too. The searched terms are highlighted in the contexts and variants found. Finally Bistro also contains a KWIC tool for all three languages (Italian, German and Ladin).

The idea behind BISTRO is thus to grant the user maximal autonomy with a proper perspective and focus. The approach could be termed equally 'constructivist' as users have access to the same term tools which terminologist used for the elaboration of the terminology. At the moment the user can decide to select between normed and not normed terms, choose the target language, the domain, to set a focus and to advance through the data through hyperlinks and term tools e.g. from term to context to bibliographical data, to similar bibliographical data, to associated corpora, to term extraction on the associated corpora and a KWIC of the extracted terms etc.

6 CONCLUSION

In this paper we have presented a new approach to managing terminological data, which is based on a structured database model. By describing the different subparts of the system, we have discussed its strengths in respect to features crucial to terminological systems like data consistency, powerful querying facilities and user adaptiveness. Further we have presented so called term tools for semi-automatic terminology work. In the end we have addressed the question of data representation and provided examples of how data is currently displayed.

In the BISTRO system all relevant information like term entries, definitions, bibliographical information and corpus segments are stored as a network of interlinked entities. The data kept in a relational database is displayed in the form of ordinary tables comprising different cutouts of information relevant to the user. The transformation from the data level to the graphical representation is completed by the help of XML.

Presenting our abstract data model we could show that a well designed net structure with a normalized set of atomic entities creates a framework for development and storage of a valid set of formalized, unambiguous and non redundant terminological data. By providing a facility for structured and modular data organization the proposed system overcomes a main shortcoming of previous models. As we emphasized before, modularity and coherency of data are a fundamental prerequisite for long term maintainability of data.

At the same time organizing data as a network enables us to autonomously access different parts of information content without the need to always consider a complete set of information. Addressing the question of flexible data access we demonstrated how our model is capable of resolving complex queries concerning relations between different parts of the network. We showed why powerful querying facilities are important for both efficient information access and user adapted data presentation.

Further we discussed the advantages of an architecture combining a relational database with XML representations over approaches solely based on XML. Relational databases are designed to ensure data validity at all times. Therefore one of its strengths is to provide mechanisms that handle simultaneous data access and modification automatically, whereas XML serves well for the flexible presentation of data to the user. We could show that the BISTRO system integrates these two techniques to fully exploit the benefits of each one. The BISTRO system is freely accessible online, where the user can choose between different search masks and data displays.

Overall we demonstrated that the proposed approach integrates different ideas and technologies to overcome well known problems that all terminological information systems have to struggle with. By doing that we could show the strengths of the BISTRO system in respect to the main requirements for a terminological system.

REFERENCES

AHMAD, K. & HOLMES-HIGGIN, P. (1996). Is your Terminology in Safe Hands? Data Analysis, Data Modelling and Term Banks, Terminology and Knowledge Engineering. In GALINSKI, C., SCHMITZ, K.D. (eds.) Proceedings of the 4th International Congress on Terminology and Knowledge Engineering, Vienna 1996.

GAMPER, J. (1999). Construction of a Parallel Text Corpus Encoding Primary Data. In: Academia, 18 1999. Bozen, 32-33.

POWERS, S. (2003). Practical RDF, Solving Problems with the Resource Description Framework, O'Reilly, Sebastopol CA.

STREITER, O. & VOLTMER, L. (2003). A Model for Dynamic Term Presentation. In: Terminologie et Intelligence Artificielle TIA-2003, Strasbourg 2003.

STREITER, O. & VOLTMER, L. (2003). Document Classification for Corpus-based Legal Terminology. In: VRABIE, G., Turi, J.G. (eds.) The theory and the practice of linguistic policies in the world. 24-26 May 2002, Iasi 2003.

STREITER, O., ZIELINSKI, D., TIES, I. & VOLTMER, L. (2003). Term Extraction for Ladin: An Example-based Approach. In: Actes de la Dixième Conférence Nationale sur le Traitemen Automatique des Langues Naturelles (TALN 2003), Batz-sur la Mer 2003.

STREITER, O., STUFLESSER, M. & TIES, I. (2004). CLE, an aligned Tri-lingual Ladin-Italian-German Corpus. Corpus Design and Interface. In: Workshop on "First Steps for Language Documentation of Minority Languages: Computational Linguistic Tools for Morphology, Lexicon and Corpus Compilation", Lisbon 2004.

STREITER, O., TIES, I., RALLI, N. & VOLTMER, L. (to appear). BISTRO, the online platform for terminology management: structuring terminology without entry structures. In: TEMMERMAN, R. & KNOPS, U. (eds.) Linguistica Antverpiensia, Antwerp.

STREITER, O., ZIELINSKI, D., TIES, I. & VOLTMER, L. (to appear). Example-based Term Extraction for Minority Languages: A case-study on Ladin. In: Soziolinguistica y Language Planning, Ortisei 2002.

VOLTMER, L., STREITER, O., ZIELINSKI, D. & TIES, I. (2003). Termextraktion durch Beispielterme Ansätze und Versuchsergebnisse für eine Minderheitensprache, Eurac Workingpaper October 2003.

VOLTMER, L. & STREITER, O., Klassifizierung von Korpora für die Rechtsterminologie, Eurac Workingpaper October 2003.

VOLTMER, L. (2005). Werkzeuge für Rechtsdatenbanken - Über computerlinguistische Verfahren zur Untersuchung, Speicherung und Kommunikation rechtlichen Wissens, Diss. LMU München 2005.