Björn Hansen / Petr Karlík (eds.)

Modality in Slavonic Languages

New Perspectives

VERLAG OTTO SAGNER · MÜNCHEN 2005
The Regensburg Diachronic Corpus of Russian: A New Source for Linguistic Research (Not Only) on Modality*

Roland Meyer, Regensburg
roland.meyer@sprachlit.uni-regensburg.de

1. Introduction

In diachronic linguistics, sparseness of data is the rule rather than the exception, which makes every bit of authentic material extremely valuable and worthy of consideration. Truly empirical diachronic research has therefore always been a long and painstaking process, involving not only philological expertise, but also many recurrent and time-consuming tasks, such as searching, sorting and counting of results. Many of the latter could nowadays be fulfilled more quickly and reliably by a computer, if only data sources were digitized in a uniform format and made accessible in a user-friendly way. At the same time, philological knowledge about a given text has to be represented consistently and should be retrievable by the user. In order to contribute to the filling of these gaps, a research group at the University of Regensburg has commenced work on an electronic diachronic corpus of Russian. The main design decisions have been made, and the growing corpus is now gradually being made available to students and researchers over the internet.

This article serves a dual purpose: Firstly, it presents some central considerations governing the design of this corpus, so that potential users can estimate its usefulness for their own research. Secondly, and as a demonstration of the first point, it shows how selected modal predicates can be searched in the material included so far, making a very modest contribution to research on the development of modality markers in Russian.

* I wish to thank S. Birzer, F. Čermák, E. Hansack, B. Hansen, K. Kučera, Vl. Petkevič, K. Ribarov and R. v. Waldenfels for discussion of various problems of diachronic corpora. C. Grillborzer and R. v. Waldenfels have scanned and prepared the electronic version of Šestodnev, and E. Hansack has contributed the corrected electronic version of Flavius’ Judeiskaja vojna. All errors are my own.
2. An Electronic Diachronic Corpus of Russian

2.1 General Considerations

McEnery & Wilson (2003) provide four guiding criteria for the design of corpora in the modern sense of the word:

- sampling and representativeness
- finite size
- machine-readable form
- a standard reference

With respect to diachronic corpora, representativeness is hardly attainable, since the genres and registers of the available texts are fixed by historical accident. Nevertheless, one can keep the goal in mind, and at the same time warn the user about uncritical over-interpretation of seemingly objective search results (cf. Rissanen 1989). In his design considerations for the diachronic part of the Czech National Corpus, Kůčera (2002) proposes to include as many documents as possible only for the very early periods, working towards the sampling scheme of the synchronic part as more and more original sources are preserved. A similar sampling approach has been chosen for the Helsinki diachronic corpus of English. Although we intend to follow this line in the future, the present stage of our collection is clearly opportunistic (in the sense of McEnery & Wilson 2003), for practical reasons. As soon as a considerable amount of text has been prepared, this will have to change. The same holds for the second and fourth criterion: The Regensburg collection is in the process of growth rather than being a standard reference of finite size. So far, progress has mainly been made on the technical side of the enterprise. For these reasons, the label “corpus” in the sense of modern corpus linguistics rather describes the goal than the present stage of development. Incidentally, the same could be said, to various degrees, about all publicly available synchronic corpora of Russian of which we are aware.

A corpus must be designed with the intended user in mind; in the present case, we aim at building a corpus genuinely for linguistic and – to some extent – for philological research. It seems that development of electronic encodings of historical sources has been lead by at least two different objectives: First, it is expected to preserve valuable cultural heritage in the most accurate and authentic way possible, providing a highly diplomatic rendition of the sources in a very elaborate encoding. Within Slavonic corpus linguistics, this has e.g. been the goal of the Repertorium project (Birnbaum 1996; cf. also Dipper et al. 2004 on plans for a versatile German diachronic corpus). Second, it may provide only those aspects of the source text which can reasonably be expected to be linguistically relevant. This is probably the more common method in present-day diachronic corpus linguistics (cf. Kůčera 2002 on the
Czech National Corpus and Kytö 1996 on the Helsinki corpus). While only new electronic editions of the most authentic sources seem sensible under the former approach, this is not mandatory under the latter – here, good standard editions could already form a useful source for research. The TITUS project (cf. http://titus.uni-frankfurt.de) is an example of a reduced encoding approach which relies for the most part on previous editions, whereas the ACT project (http://prometheus.ms.mff.cuni.cz/act/www/) offers a reduced encoding of previously unedited manuscripts in a technically mature corpus.

For several reasons, we have decided to focus on a reduced encoding of materials which have mostly already been edited and published in print. Corpus preparation is far more laborious with diachronic texts than with synchronic ones, and resources are restricted. Since our corpus is intended to be used in linguistic research, it seems only logical to include the best available non-electronic sources which have provided a common standard for linguistic analysis in the past. Needless to say, previously unedited manuscripts are treasures to be included in the corpus whenever possible; but there is a certain trade-off between diplomaticity and linguistic usefulness. It would already be extremely useful to bundle forces and have a comfortable online system on the basis of reliable editions. Early manuscripts of Russian are rare and hardly accessible outside Russia; the mere size of the task of editing them calls for a cooperative effort. A formal cooperation with the Institute of the Russian language of the Russian Academy of Sciences is under way, and if the corpus proves helpful, Slavic philologists might consider presenting their collections under this common roof.

The goal of the Regensburg diachronic corpus of Russian could be summarized as follows:

- Providing an empirical basis for research on linguistically relevant diachronic changes in Russian
- in a standardized format, conforming to the Guidelines of the Text Encoding Initiative (TEI)
- ensuring compatibility across texts and times of origin
- containing text-structural and linguistic annotation
- with easy and comfortable online access over the internet

At the present stage, we have set up the basic specifications of the project, encoded two larger texts and included them in a database with a lightweight query interface for online access using a run-of-the-mill web browser. The sources currently available are Flavius Josephus’ *Judejskaja Vojna* (in a revised version of Meščerskij’s edition, cf. Hansack 1999) and the *Šestodnev* (cf. Barankova 1998), both Old Russian
versions of earlier texts, written down around the 15th century. More data will be included and functions added to the interface sequentially.

2.2 Design

2.2.1 Data Structures at the Token Level

Textual data to be included in a corpus may be inherently structured to various degrees. It may contain a relatively small set of non-crossing structural units, such as headings, paragraphs, spoken paragraphs, sentences, and tokens (word forms); each word form may be annotated with lemma, morphosyntactic and possibly further information. Such a format would be clear enough to be directly transformed into a relational database scheme, and it has proven efficient and successful for many modern corpora, notably those based on the Corpus Workbench of IMS Stuttgart (CWB, originally CQP, cf. Christ 1994). Among the Slavonic corpora, this data format is used in the Oslo Bosnian Corpus, the Tübingen Russian Corpora, the Croatian, Czech and Slovak National Corpora², among others.

On the other hand, edited textual data can also be structured in far less predictable ways: It may contain comments, additions and corrections at various levels, e.g. regarding sentences, word forms, or graphemes; there may be various possible regularizations for some original forms, but none for others; some, but not all of the original data may be aligned with parallel texts in other languages, sentence by sentence or by larger paragraph units. Data of this kind is called semi-structured, hinting at the incompleteness and partial unpredictability of its structuring. These are only a few of the problems typically occurring with historical corpora. When we turn to manuscripts, many more decisions have to be made; as the following picture from the Flavius manuscript illustrates, there may be unclear writing, edge glosses, and even graphic material which the encoder might decide to include.

Diagram 1. Graphic material in the Flavius manuscript.

---

1 Thanks are due to the editors for permission to include their texts in the electronic corpus.
2 Not all of these use the IMS Corpus Workbench (CWB); the Croatian, Czech, and Slovak National Corpora rely on the Manatee/Bonito system (Rychlí 2004), which has the same data format as the CWB.
Since all information of potential linguistic relevance must be encoded, it would be undesirable to skip details just because they do not fit our initial database scheme. Rather, we should face the semi-structuredness of our data and design an annotation scheme which is strict enough in the parts accessible by simple standard queries, and at the same time flexible and extensional enough for the less clearly structured, unpredictable parts of the annotation.3

The currently most widely-used format for the annotation of semi-structured data is XML, an international standard which is mainly intended as a storage and exchange format, rather than as a base for processing. The XML metalanguage follows a restrictive syntax, but it is not assigned any semantics to begin with. However, the TEI has proposed annotation techniques for XML-based text encoding, together with prescriptive rules not only for the combinatorial syntax, but also for the meaning and usage of XML elements and attributes (McQueen & Burnard 2005). Furthermore, following an international standard such as TEI guarantees easy reusability of the data and better compatibility with new software. TEI-P5 is an extremely rich annotation system; the choice of the appropriate subset of elements and attributes is intentionally left to the corpus designer. A mechanism for the comfortable introduction of additional structures and the redefinition of existing ones is being developed in the latest version of the TEI guidelines (P5). In fact, modifications of the TEI scheme are necessary even for a relatively modest corpus as the Regensburg one, since the standard remains somewhat unspecific on linguistic annotation. This was the reason for other annotation standards – as e.g. TUSNELDA, defined for the Tübingen research program on “Linguistic Data Structures” (Kallmeyer, Meyer & Wagner 2001), and the annotation of the IPI PAN Polish corpus (Przepiórkowski 2004) – to borrow and modify parts of the Corpus Encoding Standard (XCES) of the Expert Advisory Group on Corpus Encoding Standards (EAGLES). By their very goal, the TEI guidelines are rather tailored towards semi-structured textual data than towards linguistic corpora, which are usually more strictly structured at least at the token level.

When designing an annotation system for linguistic corpora, one should always keep in mind that this enterprise serves a practical purpose; standards not only arise by a priori decisions, but by consensus on practical needs. This point becomes virulent when databases and query tools for the XML-annotated corpora have to be written. An XML-annotated document may be many times as large as the original; direct searching of single documents by XPath or XQuery processors is both inefficient and extremely uncomfortable for the corpus user. There are various ways of transferring semi-

---

3 In the first version of the Regensburg corpus, we used CWB with its data structure. For the lack of Unicode support (which has been remedied in the meantime) and for the reasons given above, this seemed no longer sufficient; thus, we switched to a TEI-(XML-)based encoding.
structured data into a relational or object-relational database (see Klettke et al. 2003 for an overview); the choice has strong implications for the speed and flexibility of possible queries. A very promising approach in this respect has been taken by the designers of the ACT system (Ribarov et al. 2004). Their annotation for Old Church Slavonic manuscripts, albeit impoverished in comparison to the TEI proposals, was designed with an implementation as a relational database in mind. The system comes with an efficient relational database scheme and the respective encoding tools.

To give a concrete example⁴, let us review the annotation of a typical ambiguous form according to TEI and according to the ACT system. The critical token in (2), оуби(л), is rendered as the preterite l-participle оубил in Meščerskiy’s edition of Flavius Josephus’ Judeiskaja Vojna. Close inspection of the manuscript has revealed that the final л was added as a later superscript gloss, while the original scribe had apparently written the aorist form оуби. Beyond doubt, this piece of information is important e.g. for linguists who conduct research on the development of tense forms in Russian. Intuitively, the annotation should separate the two rendered forms at some level, assigning both of them analyses of their own; but it should also, at another level, keep them together as two representations of the same token. Following the TEI-P5 guidelines straightforwardly, we could annotate the respective line as follows:⁵

(2) града Модеина, поемъ от своих с многих оуби³

(3) <line n="27"> града Модеина, поемъ от своих

   <expan abbr="e">с мѣни</expan>

   <choice>
       <orig>оуби</orig><add hand="later"><hi rend="superscript">л</hi><add></add></orig>
       <reg type="without_gloss">оуби</reg>
       <reg type="with_gloss">оубил</reg>
   </choice>

</line>

The whole line is embedded into a <line> element with a number attribute n. Tokens which do not require any special markup are left as running text. Only оуби³ gives rise to a <choice> element with subelements for the original form (<orig>) and its regularization(s) (<reg>). Since markup is added only where necessary, this type of

---

⁴ E. Hansack pointed out this problem to me in his edition of the Judeiskaja vojna.

⁵ There are many different choices which would still conform to TEI; e.g., every token could be enclosed in a word (<w>) element, or line breaks could be marked up differently (regarding the latter, see section 2.2.2).
encoding is clearly semi-structured in the above sense. The superscript in the original form is analyzed as an addition by a later scribe (<add hand="later"/>), written in a highlighted form (<hi/>).

If a search tool had to look for instances of оубил, it would have to check through running text as well as text deeply embedded in <choice> elements. If the output should only include original graphic forms, it would have to find a mixture of text embedded in <orig> elements and unannotated text. A second issue is where to add linguistic information like lemmata or morphosyntactic categories. In TEI, these are usually encoded as attributes of a word element (<w>), which has to be put around every (linguistically analyzed) token. But since regularizations differ as to their morphosyntax, we are left with two options: Either <w> is introduced as a daughter of <reg>, or <reg> is also marked up with the respective attributes. The first option seems quite counterintuitive (an original may yield many “words”), and the second one again makes us search the same information in two very different places (<w> and <reg>, respectively). This is not impossible, but queries become unnecessarily complicated and time-consuming.

In the ACT system, the above line would be partly represented as follows:\(^6\)

\[
\begin{align*}
(4) & \quad \textit{<originalform row="27" positioninrow="1" page="1"/>}\textit{града</text>} \\
& \quad \textit{<renderedform/>}\textit{града</text>}</renderedform> \\
& \quad \textit{</originalform>} \\
& \quad \ldots \\
& \textit{<originalform row="27" positioninrow="6" page="1"/>}\textit{снвъ</text>} \\
& \quad \textit{<renderedform/>}\textit{с<ы>н<о>въ</text>}</renderedform> \\
& \quad \textit{</originalform>} \\
& \textit{<originalform row="27" positioninrow="7" page="1"/>}\textit{e</text>} \\
& \quad \textit{<renderedform/>}\textit{e</text>}</renderedform> \\
& \quad \textit{</originalform>} \\
& \textit{<originalform row="27" positioninrow="8" page="1"/>}\textit{оубил</text>} \\
& \quad \textit{<renderedform variantnumber="1"/>}\textit{оуби</text>} \\
& \quad \textit{<morphology/>}\textit{aorist</morphology>}</renderedform> \\
& \quad \textit{<renderedform variantnumber="2"/>}\textit{оубил</text>} \\
& \quad \textit{<morphology/>}\textit{1-participle</morphology>}</renderedform> \\
& \quad \textit{</originalform>}
\]

\(^6\) Notably, this is not the format stored \textit{internally} in the database tables, but an ACT-XML representation of it. The morphosyntactic pseudo-“tags” \textit{aorist} and \textit{l-participle} are given here only for illustration and would definitely be encoded differently.
Every token of the corpus is encoded in an `<originalform>` which obligatorily contains `<text>` and optionally embeds an arbitrary number of `<renderedform>` elements. This setup avoids both problems mentioned above: If text has to be searched or displayed, it can always be found easily in a `<text>` daughter, either of the regularization or of the originalform. Linguistic annotations are generally ascribed to rendered forms, rather than to corpus tokens. Both analyzed and non-analyzed tokens are thus represented by the same data structure. 7 For the initial annotation of the documents in the Regensburg corpus, we independently chose an annotation scheme very similar to ACT, using, however, the element and attribute names of the TEI standard. Revised following the new TEI-P5 guidelines, it would look roughly like this:

(5) `<seg type="token" id="1.27.1" n="226">\<orig>града</orig>\<reg n="1" type="graphemic">града</reg></seg>`

As in ACT, every token is encoded in an element (`<seg type="token">`) and numbered according to its page, line, and position in the line. A token always contains an `<orig>` and at least one `<reg>` daughter(s). As proposed in TEI-P5, we indicate lemmata and morphosyntactic analyses as attributes on the `<reg>` element. The morphosyntactic features `#lpart` and `#aor` are only illustrative; tags will eventually be formulated in a positional tag system and are understood as pointers to explicit feature structures (see TEI-P5). A further necessary step will be the inclusion of so-called `hyperlemmata` (Dipper et al. 2004), i.e. a representation of lemmata which is unified across the texts in the database. Since we are aiming at a *diachronic* corpus, it has to be

---

7 The same happens internally in the corpus processor Xaira which is currently under development under the auspices of TEI (see [http://www.xaira.org](http://www.xaira.org)): Running text is automatically tokenized and put into `<w>` elements.
ensured that lemmata which “belong together” historically can be searched together. In this way, one will eventually be able to follow the diachronic development of a lexical unit, even if it changed in surface form across time.

Other than the prospective German Diachronic Corpus DDD (cf. Dipper et al. 2004), we will not base our encoding on the grapheme as a minimal unit, forming words only on an abstract level by links to linear sequences of graphemes. While this is conceptually reasonable, it generates an enormous overhead during corpus processing. Most of our texts will be previously tokenized; in the rare cases of ambiguous tokenization, we intend to follow an idea of the ACT designers: the whole ambiguous stretch is included in one large <orig> element whose <reg> daughters represent the tokens of each of the disambiguated sequences; they are numbered according to the disambiguous sequence to which they belong and according to their position in this sequence (cf. Bubník 2003, 16).

A major decision at the token level concerns the rendition of orthographic variants. Obviously, not all of these are relevant for morphosyntactic analysis or lemmatization; in fact, they often lead to the erroneous assumption of two lemmata where one would be more appropriate. Birnbaum (1996) proposes to conflate graphic differences at some level, according to philological knowledge about the meaningful distinctions at a given state of the language. For Old Russian, the graphic /i/-variants – ѣ, і, ḳ, Ṭ – can be rendered as one grapheme у, the spellings of /u/ – оў, ϵ, χ, ё – as one grapheme у, the e-variants – е, е, и, е, among others – as one grapheme е, and the /o/-variants – ӧ, ѡ, ɬ – as one grapheme о (the latter have to be distinguished, however, for number signs). In our annotation, all these normalizations of spelling apply at the rendition level, i.e. the <orig> element is spelt as closely as possible to the original glyphs, while the <reg> elements are coded in normalized orthography. It does not always make sense to code all the graphic distinctions in a source text as separate Unicode graphs (so-called glyphs) at the <orig> level; a major problem being that not all of the necessary codepoints exist. For the superscript characters prevalent in early texts, we distinguish two cases: If the edition is uninformative about the exact position of characters above the baseline, we use the annotation proposed by TEI (see (5) above): the superscript part is included in a <hi> (i.e. “highlighted”) element. If the positions of characters above each other can be made out exactly, we follow a convenient notation based on the so-called “HIP standard” used in the Russian orthodox internet community (cf. http://www.pechatnyj-dvor.su/docs.html): {абв’dе} denotes “дё above abc”, {дё} denotes “дё above the baseline”. Cascaded Style Sheets (CSS) then take care of the appropriate rendering by web browsers or in pdf documents. This leads to a more or less acceptable output, as illustrated below.
As is obvious from (5), we code the expansion of abbreviations like сйво in the regularized form (<reg>). The bracket notation с(ы)н(о)въ, commonly used in editions, may be included as a further regularization of type edited, but has no status in the system apart from preserving the editor’s idiosyncratic coding.

2.2.2 Data Structure at Higher Levels

TEI-P5 introduces a huge selection of XML codes for markup beyond the level of single tokens. ACT takes a very different route and postulates only a generic element <complex>, which points to groups of tokens or other complexes. In principle, this construction can mimic any given TEI structure; the TEI element names would have to be supplied as attributes of the <complex> elements. A drawback of this aspect of the ACT encoding may be that restrictions on the elements’ syntax can hardly be formulated in a Document Type Definition (DTD), since they all formally share the same element name. A restriction of the kind “element a with attribute X must have element a with attribute Y as a daughter” cannot be expressed in DTDs. One would also like to be able to see and search the standard TEI elements at some level; in this case, the TEI markup has to be restored by processing. One way or the other – using “dummy” elements like <complex> or the more explicit TEI elements –, it has to be decided which structural units of the text are to be annotated, and how.

Texts may be structured logically – e.g. into parts, chapters, paragraphs, and sentences – and physically – into folia, pages, lines etc. Obviously, we need both aspects: the logical one for linguistic analysis, the physical one for proper citations and display. The main problem in this domain consists in elements from the linguistic level which cut across elements from the physical level and vice versa. These so-called concurrent hierarchies are generally forbidden in XML, and constitute a major problem when devising an annotation scheme for corpora. TEI proposes several strategies for circumventing the issue; we will mainly make use of so-called milestones: Instead of enclosing structural units such as lines and pages in <line> and <page> elements and thereby taking the risk of cutting across e.g. sentence boundaries, we introduce line
breaks and page breaks as numbered empty elements without content. Cf. the following example from Flavius’ *Judeiskaja vojna.*

(6)  
<lb n="26"/>[…] и то место ровом глубоком. Храмы же(е) ставиша о"мти-
<pb n="153"/>
<lb n="1"/>ноуд части в’нутрь града. и клад(я) създаша конець

Since every token is internally numbered for page and line position (see 2.1.1), it is easy to address the context of a hit (e.g., five words or two lines around it), and line units are unnecessary. For simplicity, we encode tokens which are split up across lines and pages (e.g. ο"μιουδ in (6)) as such only at the <orig> level; they are represented as an unsplit whole in the <reg> daughter of the first part.

The DDD corpus (Dipper et al. 2004) envisages a more radical solution, namely, a strict stand-off annotation, in which the logical and physical (and possibly further) structures of a document are stored separately and linked to the respective stretches of text by pointers. Concurrent hierarchies can then easily be evaded. While this approach is certainly conceptually superior, it also generates extreme overhead during encoding: whenever part of the original text is changed, all annotation files have to be updated consistently. Since we do not expect too many cases of overlap in the Regensburg XML annotation, we would rather stick with the more traditional technique (see Wagner (2005) for a similar view with respect to the TUSNELDA corpora).

TEI-P5 provides useful solutions for many important issues in the edition of primary sources: Original text which is viewed as erroneous by the editor may be marked up in a <sic>/<corr> group; text which has been added by the author, a later scribe or corrector can be included in an <add> element; text deleted by the author, a later scribe or corrector is put into a <del> element; text deleted by the author, a later scribe or corrector is put into a <del> element; text deleted by the author, a later scribe or corrector is put into a <del> element; text supplied by the editor should be marked by a <supplied> element; gaps in the original may be indicated using <gap> elements (cf. TEI-P5 for details). All of these elements may either concern (groups of) tokens, i.e. complete <seg> elements, or reach down into the <orig> level. We seem to need the latter possibility, i.e. embed elements like <sic>, <corr>, <add>, <del>, <supplied> and <gap> inside an original form (cf. (7), (8) below). This is probably our strongest deviation from the ACT proposal, in which the text within a <originalform> cannot be internally structured.10 (7) shows an example of overlap between an <add> element and word boundaries. Curly brackets indicate glossed material added by a later scribe:

---

9 For reasons of clarity, markup at the token level is left out here (but see section 2.2.1).
10 However, K. Ribarov (p.c.) suggests a way to circumvent this restriction by encoding sub-token annotation in a rendered form (= written-out or regularized form in our sense), which would be allowed in ACT-XML.
Such cutting across elements is syntactically ill-formed in XML: However, in the case at hand, we do not see a problem in splitting the element in two and respect token boundaries, i.e. as two separate added glosses:

(8) \{\text{"no}\} \delta u \{\text{"\} ux

This concludes our short overview of the reasoning behind the XML annotation of the Regensburg corpora. Let us now turn to issues of implementation.

2.2.3 Implementation

Although our project has commenced only recently, it has already seen two very different test implementations. The first one, presented at the Regensburg workshop on modality in November 2004, relies on the Corpus Workbench CWB, which was developed at the University of Stuttgart. A screenshot of the web interface is shown below (diagram 3). This version has been quite efficient and unproblematic to set up, but it suffers from the drawback of not being fully XML-aware. CWB cannot generally handle semi-structured data, but rather requires a strictly structured, tabular data format for its input. Since the main rationale of our software setup could be described by the slogans flexibility and conformity to standards, this was a strong reason to commence work on a second implementation and explore a purely XML-based approach.

Flexibility is important for us, because we expect details of the annotation to change due to practical needs as more and more texts become encoded. As long as our corpora conform to the rather liberal TEI conventions, we want to be able to switch to a revised annotation structure easily. As far as standards are concerned, Unicode is

---

11 To be sure, XML markup can be included in CWB data to some extent, but it basically only provides boundaries for the context of a hit. It cannot extend below the token level, and it cannot be searched properly.
indispensable for our character encodings, and TEI-XML forms the basis of our markup. Independently of the ultimate internal format for storage and processing, it has to be ensured that the XML structure of our texts is available for output in a fully restored form. In this way, we hope to guarantee compatibility with future software for corpus processing, which will most probably be based on some XML format.

However, bare XML is extremely cumbersome to read. Instead of having to understand XML, the user should be able to choose a converted textual format with freely definable context size. Although advanced users may know the internal structure of our documents, we cannot expect the same from the casual visitor. The query interface has to serve both groups: Advanced users should be able to fully exploit the database by searching via XPath or XQuery expressions (see Lausen & May 2003 for an overview and the specifications at http://www.w3.org/XML/Query for details). On the other hand, users who wish to ignore technicalities should be supported by easy-to-use, predefined query types which grant convenient access to the database. The new query form therefore requires the user to decide on the level of analysis at which the query input is to be evaluated. Currently, five levels are available (see diagram 4).
Searching, e.g., at the level of graphemic rendering will only return results which match the query string graphemically (within `<reg type="graphemic">`); in the case of “бог*”, this includes any abbreviated forms from the original source (бъ and the like). As the query string itself is truncated by an asterisk, it will match any graphemic renderings which begin with бог, i.e. all case forms of богъ, but also all forms of богатъ, богатство etc. Notably, the query at the graphemic level will also catch tokens which match the query string in one of their variants. Thus, e.g. searches for “уби” or “убил” in example (4) would both return the original оубил.

Queries at the level of the editor’s rendering will output the spelling which the editor of the source chose for encoding; by searching for “б(о)ж*” in the “Jewish War”, it would, e.g., hit all instances of abbreviation which begin with бъ. Obviously, this requires knowledge about the editor’s idiosyncratic encoding scheme; the rationale behind having this level is mainly that we want to be able to restore the original input document from the database without loss also if it was an edited source. Unsurprisingly, searches for all renderings will find their match in any of the renderings. Querying the original form makes it possible to restrict searches to the exact graphic realization in the source (e.g. for analyses of orthography); in this way, one may still search explicitly for one special type of ı, ı, ı, ı or оыв, ё, ю, ю. The last two options in diagram 4 require more knowledge about the internal XML structure of the tokens and the database, respectively, and mainly serve advanced and internal purposes. Essentially, this feature allows making use of the corpus as a fully-fledged database rather than a text. Just to give an idea, diagram 5 shows an example of an XQuery request used to output a list of
all different tokens in a document which match certain criteria, rather than the lines in which a form occurs.

As indicated by the above version numbering, the interface is still at a preliminary stage of development. The list of search levels will obviously be extended as more levels of linguistic analysis become encoded. In the near future, we plan to include at least a lemmatization of the texts, together with the respective hyperlemmata.

The output has to be postprocessed in order to be readable as usual text. We currently provide retrieval by lines, together with the option of seeing more context in an in-line format or in XML. Conversion to the in-line format is effected dynamically by XSLT transformations. As shown in the screenshot (diagram 6), the user is standardly presented with the data in an in-line, so-called K(ey) W(ord) I(n) C(ontext) format. The area in the middle of the window displays the single lines containing hits, or rather, the textual daughters of their <orig> parts. By clicking on the button to the left, a larger context may be requested, which then shows up in the bottom part. Needless to say, much could still to be done on the optical details. For the time being, hits are highlighted by special background colour, and the source is indicated to the left of the lines.

Diagram 6. The search interface, version 0.2.
By default, we only display the original forms from the source, not the regularizations or other parts of the XML tree (although searching these is supported, as mentioned above), but a raw XML view is also possible. Diagram 7 contains an overview of the whole query system.

The native XML database with its XQuery search engine forms the core of the system. Input documents undergo preprocessing by using Perl scripts and XSLT transformations, until they are valid with respect to our version of the TEI-DTD. The web interface (written in PHP) composes XQuery requests and sends them to the database via a standard protocol. The query output is sent back to the web server, stored intermediately for later access and partly postprocessed by XSLT transformations. At the present stage, we are using the open source database eXist (http://www.exist-db.org) for storage and retrieval. It has to be stressed that it is not at all our goal to develop competitive corpus processing software, although we do expect to sequentially improve the interface with further functions. We simply need a convenient, lightweight and highly configurable tool for accessing our collections over the internet, and at the same time keep our texts in a standard XML format compatible with more advanced XML-based software like e.g. TEI’s Xaira (see http://www.xaira.org), which is currently under heavy development.

Wagner (2005) – whose goals are similar, although more geared towards complex linguistic annotation – introduces a system based on the (commercial) Tamino XML database server, which would in principle seem suitable also for the Regensburg corpora. Employing a native XML database rather than a relational one (as e.g. in the
ACT system) has the advantage that arbitrary XML documents may be included and restored correctly, using general import tools which are provided by the database; the disadvantage is a much lower performance. In this respect, we will have to explore faster implementations in the future, and work on the optimization of the indexing of documents and their parts within the database. Since diachronic corpora will not nearly reach the size of modern synchronic corpora for a long time, this does not seem too problematic. If XML and XQuery functionality is not needed, one may still use the much faster first version of the CWB-based corpus.

3. A Use Case: Modals in the Regensburg Corpus

Let us now turn to modal predicates, which we will use as a first test case for the (admittedly still rather small) corpus presented in the previous sections. Our purpose is mainly to illustrate the advantages of search options as provided in the XML database, rather than an exhaustive linguistic discussion of the query results; nevertheless, we want to put some findings from these two texts into the perspective emerging from recent research on modality and grammaticalization.

3.1 Background

As Hansen (2004) discusses in detail, several diachronic shifts in meaning relating to general principles of grammaticalization (Lehmann 1995) have occurred within the category of modals in Russian. To name just a few, ДОЛЖЕН almost exclusively denoted ethic-religious obligation and combined with nouns referring to humans until 14th century (cf. also Pallasová this vol., 271); by a process of semantic bleaching, it developed into a rather general expression of “objective necessity”, and later into “high probability”. Several modals from the semantic field of “necessity” were dropped along the way (АЛІПО, НЕЛІПО, ДОСТОІНО, НЕДОСТОІНО); the function of the remaining central expression of “necessity”, ПОДОБАТИ, was later taken over by НАДОЕТ – cf. Vaulina (this vol.). НЕЛЬЗЯ / НЕЛЬЗЖ expanded from the meaning of “objective possibility” to “permission”, while its non-negated counterpart НЛѢТ fell into disuse early (Vaulina this vol.). МОЖЕТО lost its old lexical meaning “powerful” and specialized into a modal (Hansen 2004, 259).

The two larger texts currently included in the Regensburg database are potentially interesting with respect to the meaning of modality markers, because they both represent Old Russian (14th-15th century) versions of south-Slavonic originals and might preserve some archaic features (cf. Hansack 1999 and Barankova 1998 for details). According to their time of origin and the above generalizations, however, we could already expect to find ДОЛЖЕН with an emerging non-ethical meaning; furthermore, НАДОЕТ should already occur, while АЛІПО, НЕЛІПО, ДОСТОІНО,
should be rare, and подобати might already be peripheral. Пользу should occur in the meaning of “permission”, and можно/можно/можно should clearly be a specialized modal.

3.2 Searching for Modals

Most of the modals mentioned above (except for должен) have the advantage of occurring in a fixed morphological form, so we can do without a proper lemmatization. However, we expect quite divergent orthographic realizations. As alluded to already in diagrams 5 and 6, the modal должен was found in two different surface realizations and its cognate должны in four (out of only seven examples altogether): должен, долженъ, долженъ, долженъыч, должении, долженъы (some of which also differ morphologically). In a similar vein, we find можно along with можно by and можно by and можно etc. These cases pose a serious problem for a purely surface-oriented search. Some of them can be handled by using placeholders and regular expressions on word forms: thus, all forms of должен can be found by searching for “А.?а.?ж.?н.*” in version 1, and for “А.?а.?ж.?н*” in version 2, respectively. The first variant corresponds to so-called (Perl) regular expressions, in which the question mark and the asterisk quantify over the immediately preceding expression, in this case the dot, which itself stands for any character. The question mark indicates 0 or 1 occurrences, and the asterisk stands for any number of occurrences (including 0). The syntax of variant 2 is typical for database queries; here, the question mark stands for any single character and the asterisk for any number of characters. The outcome is the same in both cases, namely, должен with possibly further characters inserted between the consonants and an ending of arbitrary length. Although these expressions are rather easy to learn, it would certainly be more convenient and less error-prone to have a proper lemmatization rather than think about all possible variants, including the almost unpredictable ones (cf. less above).

It should be noted that modals, like many other interesting word classes, are very unlikely to occur in the word index of an authoritative edition (as e.g. Barankova 1998), since these indices usually rather include content words of cultural or philological interest. Therefore, the linguist has to work through the text manually for every new phenomenon investigated. This is precisely where electronic corpora come in handy. With more and more linguistic information beyond the simple orthographic regularization being annotated, their usefulness will become ever more obvious.

As a first step, let us now check the absolute frequencies of the modals mentioned above in the two texts. In an electronic corpus, this is an extremely easy exercise. Version 1 of our query interface (actually, the Stuttgart corpus workbench CWB) provides functions for alphabetic sorting and frequency counts.
Obviously, several of the characteristic shifts around the 14\textsuperscript{th}/15\textsuperscript{th} century reported in Vaulina (this volume) have not yet happened in Flavius and Šestodnev: Among the modals expressing obligation and necessity,  долженъ is still very rare compared to the older льтъ and достоино, and the later central надо is occurring only once. Подобати clearly forms the most frequent proponent of this category. (Ne)лыдъ and (Ne)моно have not yet been differentiated; all four occur with similar frequency.

The semantic meanings of the single modals mirror this rather “early” character of the two texts. Thus,  долженъ predominantly marks ethic or religious obligation, cf.

\begin{quote}
9 \textit{дя азъ д(ов)шо свою предам о п(о)зъ, иако есъ долженъ} ‘so that I (would) offer my soul to god, as I am obliged to’ (Flavius, 164. 14)
\end{quote}

\begin{quote}
долженъ ли еси. ‘you owe me (something)’ (Šestodnev, 445.7)
\end{quote}

\begin{quote}
Моно has already clearly developed into a modal and lost every trace of its early lexical meaning, cf.
\end{quote}

\begin{quote}
10 \textit{. д(к)къ Римляномъ, обходящий городъ, не моно пъкъ приступитьти.} ‘[for five] days it was impossible for the Romans, who were moving around the city, to enter it.’ (Flavius, 119.10)
\end{quote}

On the other hand, льдъ never expresses “permission”, but only “objective possibility” in the examples found in the two texts – see e.g.
4. Conclusion

The Regensburg diachronic corpus of Russian is intended as a tool which facilitates linguistic analysis rather than as a depository for the diplomatic preservation of sources. It therefore has to be considered carefully what pieces of information should be included because of their potential linguistic and philological importance, and how this information should be structured. We have argued for a specific XML encoding, conformant to the standard of the Text Encoding Initiative (TEI). Since we need access to the full XML structure of our documents and want to be able flexibly adjust the system, we decided to have a native XML database as the core engine. Using common techniques for internet programming, a preliminary query interface has been set up. For quicker searching, a system based on the Stuttgart CWB tool is also still available. In the final section of the paper, we demonstrated how the database can be put to use for a sample analysis of modals and their meanings. The results indicate that both Flavius and Šestodnev, despite originating from the 14th/15th century, represent a rather early stage in the development of modals in Russian.

References


Vaulina, S. S. (this volume): Diachroničeskie aspekty issledovanija modal’nosti v russkom jazyke.