The Czech Academic Corpus 2.0 Guide

Barbora Vidová Hladká
Jan Hajič
Jirka Hana
Jaroslava Hlaváčová
Jiří Mírovský
Jan Raab

The Czech Academic Corpus 2.0 Guide

by Barbora Vidová Hladká, Jan Hajič, Jirka Hana, Jaroslava Hlaváčová, Jiří Mírovský, and Jan Raab

Table of Contents

List of Figures

List of Tables

Chapter 1. Preface

The Prague family of annotated corpora has a new member – the Czech Academic Corpus version 2.0 (CAC 2.0) – a morphologically and syntactically manually annotated corpus of the Czech language. The precise formulation of the CAC 2.0 would be *new and old* member, as there was only one version preceding the current one. The first version contained "only" morphological annotations; it was published a year ago, therefore it can be understood as outdated. The new phenomenon brought about by the CAC 2.0 is syntactical annotation – therefore we can characterise our corpus by another Praguian attribute – *dependency*.

The CAC 2.0 Guide is a guide to the CD-ROM, just like the previous CAC 1.0 Guide. The contents of the Guide provide all the necessary information about the project; however the user does not need to be familiar with the CAC 1.0 Guide. The CAC 1.0 Guide can be referred to for the details of the CAC project's history and its preparation details. Nevertheless, if you are already familiar with the CAC 1.0 Guide, navigating it will be easy, as we have maintained its chapters' organisation into three main units.

The first unit, Chapter 2, describes the main characteristics of the Czech Academic Corpus 2.0, the structure of its annotations and the documentation of the partial steps of the syntactical annotations.

The second unit, Chapters 3 through 6, contain the CD-ROM information and the documentation of the data component, tools, bonus material and tutorials. Part 3.2 introduces the corpus as a data file with an inner representation. A considerable amount of information concerns the corpus viewing tools – Bonito (part 3.3.1) and Netgraph (part 3.3.4), annotation editors – LAW (part 3.3.2) and TrEd (part 3.3.3) and tools for morpho-syntactical processing of texts (part 3.3.5). Chapter 4 is decorated with two bonuses; these are the STYX Czech electronic exercise book (part 4.1) and the TrEdVoice module for the voice control of the TrEd (part 4.2). All the tools provided and their graphical interfaces are documented and equipped with tutorials in the form of demos – see Chapter 5 for the complete list. Chapter 6 contains the installation instructions for the CD-ROM components. Chapter 7 summarises the information on the distribution of the CD-ROM.

Chapters 8 and 9 form the third unit of the Guide. They cover the personal and financial aspects of the project. You will find five annexes: Appendix A enumerates the sources of corpus' texts; Appendix B describes the structure of lemmas for the simple orientation in the morphological annotations; Appendix C describes the structure of a morphological tag; Appendix D guides the user through syntactical annotations; Appendix E completes the Guide with web links.

This CD is being published in the final year of the project *Resources and Tools for Information Systems*, No. 1ET101120413, financed by the Grant Agency of the Academy of Sciences of the Czech Republic. The CD completes the comprehensive results presentation of the five years of work on the project.

Chapter 2. Introduction

2.1. Introducing the Czech Academic Corpus (CAC) 2.0

The Czech Academic Corpus 2.0 is a morphologically and syntactically annotated corpus of 650,000 words.

The Czech Academic Corpus (CAC) was created by a team from the Institute of the Czech Language, of the ASCR, led by Marie Těšitelová [11] from 1971 till 1985.[1]The original purpose of the corpus was to build a frequency dictionary of the Czech language and the original name of the corpus was "Korpus věcného stylu" (*Practical corpus*). The corpus has been morphologically and syntactically annotated manually. Independent from the CAC, an annotation of the Prague Dependency Treebank (PDT) was launched in 1996. The idea of transferring the internal format and annotation scheme of the CAC into the PDT emerged during the work on the PDT's second version [16]. The main goal was to make the CAC and the PDT fully compatible and thus enable the integration of the CAC into the PDT. After converting the inner format and morphological annotation scheme, we have published the first version of the CAC (Vidová Hladká a kol., 2007). The second version presented here enriches the CAC 1.0 by adding the surface syntax annotation; in the terminology of the PDT we call this annotation an "analytical layer".

While creating the CAC 1.0, the omitted words and numerical expressions were manually replaced by wildcard symbols ("#" and "?") – these corrections and the reasons why those changes were deemed necessary are described in detail in the CAC 1.0 Guide (Vidová Hladká a kol., 2007). These wildcard symbols were not further processed during the phase of CAC 2.0's creation.

The CAC 2.0 offers:

- For linguists: Language material reflecting the real usage of the language,
- For computational linguists: The tools and a considerable amount of data that could help amend applications working with natural language and are not feasible without morphological and syntactical text processing,
- For TrEd annotation tool users: The possibility to use voice control for the tool,
- For teachers and their students: An interesting didactic tool for practising Czech language morphology and syntax.

2.2. Sources of the texts

The CAC contains mostly unabridged articles taken from a wide range of media. These articles include newspapers, magazines, and transcripts of spoken language from radio and TV programs covering administration, journalism and scientific fields. The texts are taken from the 70s and 80s of the 20th century and thus, the selection of texts is influenced by the political and cultural climate of this time period. A complete list of resources can be found in Appendix A.

2.3. Annotation layers

We cannot call a corpus "annotated" without specifying what kind of annotation the corpus contains. In other words, from the linguistic theory viewpoint, one must first characterise the so-called layers

[1]This text contains both bibliographic references (e.g. Vidová Hladká a kol., 2007) and Internet references in the form of a number in brackets (e.g. [1]) referring to the list of internet URLs in Appendix E).

of annotation. The annotation of the CAC 2.0 covers two layers: morphological and analytical. To be absolutely accurate, we must add that we also operate on another layer: the layer of words. In fact, the word layer is not a layer for annotation as it consists of the original text divided into word tokens (words, numbers written in digits and punctuation). However, for the sake of convenience, we will refer to the word layer as an annotation layer. Henceforth, we will refer to the word, morphological and analytical layer as the w-layer, m-layer and a-layer, respectively.

A morphological layer of annotation provides the word tokens with further data (annotation), which characterises the morphological properties of the word tokens (as apparent in the lemma which is the canonical form of a lexeme), the part of speech, and morphological categories (case, number, tense, person, etc.). Formally, part of speech classes combine together with values of morphological categories to represent morphological tags (or, simply, tags). In the CAC 2.0, tags are designed according to the PDT as strings of definite length (15 positions) where each position corresponds to a single category. Appendix C contains the complete list of these morphological positional tags and their detailed description.

Example: The word form *Prahu* (a form of "Prague") is analysed as an affirmative (11th position) noun (1st and 2nd position), feminine (3rd position), singular (4th position), and accusative (5th position). All of the other positions are correctly filled with the symbol "-" that represents the irrelevance of the morphological category towards the part of speech. For example, one does not determine a person and tense with nouns (8th and 9th position).

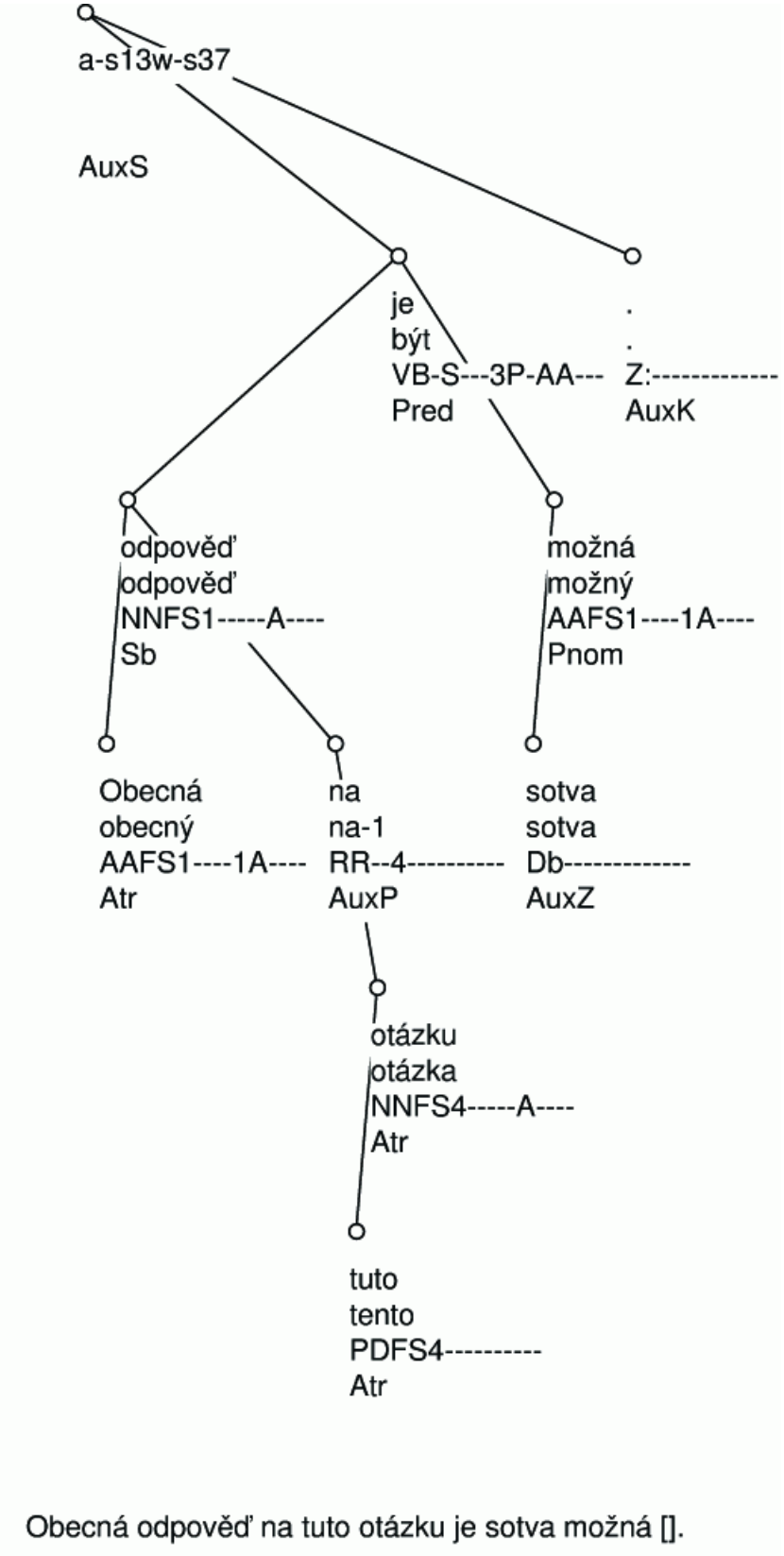
Table 2.1. Examples of lemmas and tags of particular word forms

Word token	Lemma	Tag	Description
Prahu	Praha	`NNFS4-----A----`	Noun, feminine, singular, accusative, affirmative
123	123	`C=-------------`	Digit token
)	)	`Z:-------------`	Punctuation mark (right parenthesis)

An a-layer annotation assigns each word unit the corresponding data characterising the syntactical features of the unit and therefore its relation to the other sentence elements along with its sentence function. Formally, the sentence relations are represented by a dependency tree. The word unit functions in the sentence are represented by so-called analytic functions, which are listed and described in Appendix D.

Example: Figure 2.1 shows the syntactical annotation of the sentence *Obecná odpověď na tuto otázku je sotva možná.*(Lit.: *A general response to this question is hardly possible.*) Each word unit (word, number, punctuation mark) is represented by a single node in the resulting tree. Note that due to technical reasons each tree is rooted by one extra node – the tree in our example therefore consists of 9 nodes. The annotation approach builds on the tradition of the Prague linguistic school, where the predicate (usually verb) is understood to be the centre of the sentence. Therefore the predicate *is* placed as a direct daughter of the root. The final punctuation is also placed as a daughter of the root node. Two constituents of the sentence *are* dependent on the predicate – *odpověď* (*answer*) and *možná* (*possible*). Please note that each node in the tree is annotated with the word form, lemma, morphological tag and analytic function. Looking at the node representing the word *odpověď* (*answer*), we can see its form is a feminine noun in nominative singular and that this unit stands in the role of subject of the sentence, which is expressed by the analytic function `Subj`.

Figure 2.1. Example of an a-layer annotation



The conception of the main internal format of the CAC 2.0 (in PML format – see Chapter 3.2.1) treats the annotation layers separately where each layer of annotation in the document corresponds to one file. (In the case of the CSTS format, all layers of annotation are contained in one file.) This relationship

in the CAC 2.0 means that there are three instances (files) for every document, one for the w-layer, one for the m-layer and a third one for the a-layer. However, the distinction between layers does not restrict interconnection between groups for particular layers of annotation. In fact, the opposite is true as will be demonstrated later in this section.

The word layer does not reflect the segmentation of the text into sentences; this segmentation occurs on the m-layer. This means that unlike the w-layer, the m-layer contains final punctuation. Additionally, the number of word tokens in both layers may differ. The differences originate from the concatenation of the incorrectly split word into one word, or reversely, from the division of incorrectly connected words into more units. The correctly written text should be contained in the m-layer.

Example: The three following figures illustrate the w-layer and m-layer interconnection. Also the interconnection of the files in the sense of the number of word units is captured and denoted by arrows. All three examples were chosen from the CAC 2.0 deliberately so that the user can directly view the instances; the name of the document and number of the sentence is provided for every sentence. Figure 2.2 serves to illustrate the 1:1 ratio of the layers. The layers do not differ except for the final punctuation. Figure 2.3 exemplifies the situation where a word token is inserted into the text – the year information was clearly missing. Since it is almost impossible for the corrector to add the missing year, the symbol "#" is used as this symbol has no counterpart on the w-layer. In contrast, Figure 2.4 illustrates the situation where more m-layer units corresponds to the same w-layer unit – the word unit *pedagogicko-psychologické* (E: "*psychological-pedagogical*") has been divided into three separate units.

Figure 2.2. Technical interconnection of the w-layer and m-layer: No changes other than the final-sentence punctuation

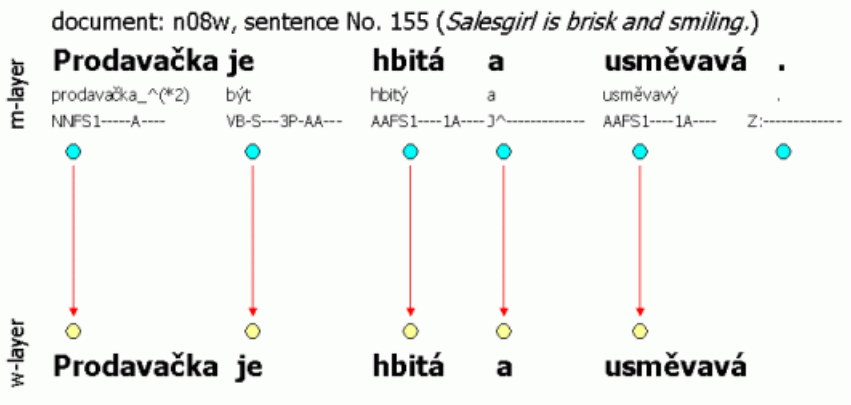


Figure 2.3. Technical interconnection of the w-layer and m-layer: The insertion of a word token

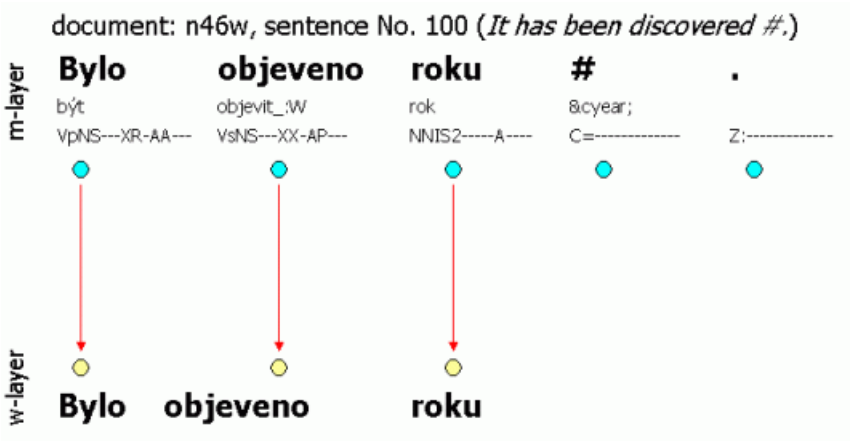
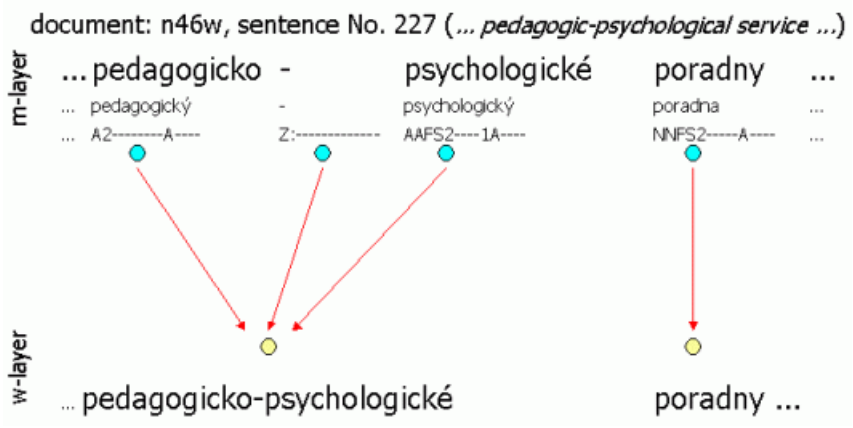


Figure 2.4. Technical interconnection of the w-layer and m-layer: The division of a word token



The interconnection between the a-layer and m-layer means that each m-layer word unit corresponds exactly to one node of the dependency tree on the a-layer, and vice versa. The only exception is the technical root, which has no counterpart on the m-layer. Figure 2.1 illustrates the interconnection described above.

2.4. The project's progress

The project of the Czech Academic Corpus comes down to us the centuries, as we have described in detail in the article (Hladká, Králík, 2006). We will not address the long journey of the CAC leading to its first version published here. The CAC 1.0 Guide (Vidová Hladká a kol., 2007) contains all of that information. Here, we would like to summarise the process of building up the layers of the second version of the CAC.

2.4.1. On the road to the CAC 2.0: Morphological annotation

The data preparation of the CAC 2.0 involved further semi-automatic checks of the morphological annotation; extensive semi-automatic checks have been already run during the CAC 1.0 preparations. These checks have been motivated by the similar processes during the building of the Prague Dependency Treebank 2.0. Detailed descriptions can be found in the CAC 1.0 Guide.

The automatic scripts verifying the data went through the corpus and marked suspicious positions; the annotators then checked the marked sentences and corrected them if needed. The main point of this work was to ensure that the morphological categories of the original tag in the CAC and of the positional morphological tag in the CAC 1.0 matched. For example, as for the noun's case category, the scripts have marked 1,258 suspicious tags; the annotator found 332 of them to be wrong and corrected them. There have been 177 suspicious instances of adjective's case and the annotator corrected 41 of them.

All of the verifications conformed to the rules of the PDT morphological annotation [17].

2.4.2. On the road to the CAC 2.0: Syntactical annotation

The analytical annotation of the corpus has raised the question of how to map the original annotation to the Prague Dependency Treebank style of annotations. Based on the experiences from the morphological annotation, we have split this question into three sub-questions: *Automatically? Semi-automatically? Manually?* The article by Ribarov, Bémová, Hladká, 2006 describes our search for the answers in detail. The authors have reached a possibly surprising conclusion: They have decided to ignore the original annotation completely and process the manually morphologically annotated texts

of the CAC 1.0 by an automatic procedure (parser). This procedure assigns a dependency tree to each sentence and an analytical function to each node. These automatically assigned trees have been manually verified (annotated). The *maximum spanning tree* parser (MST parser) described below has been used. For details see 3.3.5.

Professional linguists conducted the analytic annotation of Prague Dependency Corpus. Two annotators from the PDT group became the main arbiter for our project. Among the other annotators were one Czech student of philology and three Slovak annotators experienced in annotating the Slovak National Corpus [21] under the leadership of Prague linguists trained in the PDT annotations. Therefore the CAC annotation had two phases: annotation, arbitration. In the beginning, each document was annotated by two annotators, the annotators worked in parallel. The two annotations were automatically compared and the result proceeded to the arbiter. As soon as the arbiter agreed that the work of the annotators was fluent enough, each document was annotated only once. During the second stage of annotations, the arbiter reviewed the complete documents, not only the differences in parallel annotations. The documents were then processed by the automatic scripts verifying the different phenomena between the annotation stages.

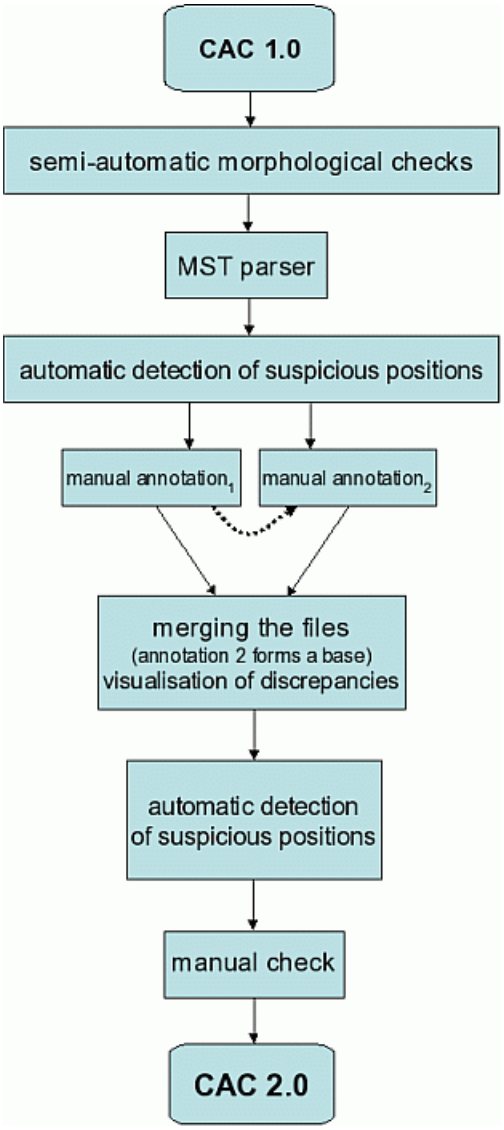
The automatic scripts verification was inspired by the scripts used in the PDT 2.0 preparations, similarly to the analytical annotations. The scripts marked suspicious positions in the data. The relations of the nodes on the analytical layer have been checked for their grammatical permissibility, and the possible combinations of the morphological tag and analytical function of each node has been checked. In the next stage the marked suspicious positions were highlighted and a brief description of the possible problem was displayed on the annotator's screen. The problem could occur either in the morphological or in the analytical annotation.

All of the verifications conformed to the rules of PDT morphological annotation [18].

As an example of the analytically-morphological verifying script, we will describe the script as it checks the annotation of the word form *se*. The script checked the following condition for each node for the word form "se": Each node for the word form *se* is either a reflexive pronoun with the analytical function `AuxT` or `AuxR`, or it is a vocalised preposition with the analytical function `AuxP`. Other scripts reviewed the agreement of morphological tag categories or the permissibility of the combination of the governing and dependent nodes' analytical functions (e.g. the preposition and its dependent noun or the permissibility of the position of a node marked as subject `Subj`).

Figure 2.5 illustrates operations on the data since the CAC 1.0 release up until the CAC 2.0 release.

Figure 2.5. CAC 2.0 preparation – data processing



Chapter 3. The Czech Academic Corpus 2.0 CD-ROM

3.1. Directory structure

This section describes the visual representation of the directory structure contained in the CD-ROM up to its second, or third tier (see Table 3.1). Any references made regarding the content of the CD-ROM that resides deeper within the tree structure notes the full path to the file.

Table 3.1. CAC 2.0 CD-ROM – Directory structure

`index.html`	# CAC 2.0 Guide in Czech (html)
`index-en.html`	# CAC 2.0 Guide in English (html)
`Install-on-Linux.pl`	# Install script for Linux (English)
`Install-on-Windows.exe`	# Installation program for MS Windows (English)
`Instaluj-na-Linuxu.pl`	# Installation script for Linux (Czech)
`Instaluj-na-Windows.exe`	# Installation program for MS Windows (Czech)
`bonus-tracks/`	# Bonus material
`STYX/`	# Electronic exercise book of Czech language
`data/`	# Data component
`csts/`	# CAC 2.0 in CSTS format (files `[ans][0-9][0-9][sw].csts`)
`pml/`	# CAC 2.0 in PML format (files `[ans][0-9][0-9][sw].[amw]`)
`schemas/`	# PML schemes and dtd of CSTS format
`doc`	# Documentation
`cac-guide/`	# CAC 2.0 Guide in Czech and English (pdf)
`tools/`	# Tools
`Bonito/`	# Corpus manager
`Java/`	# Java Runtime Environment 6 Update 3 for Linux and MS Windows
`LAW/`	# Editor of morphological annotations
`TrEd/`	# Editor of syntactical annotations, including the TrEdVoice module for voice control
`Netgraph/`	# Corpus viewing and searching tool
`tool_chain/`	# Tools for the automatic processing of Czech texts
`tool_chain`	# Script running the tokenisation and/or morphological analysis and/or tagging and/or parsing
`...`	
`tutorials/`	# Tutorials for the data and the tools

3.2. Data

This section describes the inner representation of the files itself, the rules used to name the files, and the organisation of the CAC 2.0 corpus into files.

3.2.1. Data formats

We used the *Prague Markup Language* (PML) as the main data format. The PML is a generic XML-based [31] data format designed for the representation of the rich linguistic annotation of text. Each of the annotation layers is represented by a single PML instance. The PML was developed in concurrence with the annotation of the PDT 2.0.

A secondary data format used in the CAC 2.0 is a format named *CSTS*. This is an SGML-based [20] format used in the PDT 1.0 annotation and also in the Czech National Corpus [14]. The reason why we use a secondary format for the CAC 2.0 is its more efficient human readability, the ease of its processing by simple tools and also the fact that some of the tools developed for the CAC 2.0 are only able to work with the CSTS format. A conversion tool for these two formats is also available.

In the following section you will find a summary of the main characteristics of the PML format; detailed information has been published in a technical report (Pajas, Štěpánek, 2005). The next section contains a summary of the main characteristics of the CSTS format. For more detailed information see the PDT 2.0 documentation [13].

The PML format

The layers of annotation can overlap or be linked together in the PML as well as with other data sources in a consistent way. Each layer of annotation is described in a *PML schema* file, which can be seen as the formalisation of an abstract annotation scheme for the particular layer of annotation. The PML schema file describes which elements occur in that layer, how they are nested and structured, what the attribute types are for the corresponding values, and what role they play in the annotation scheme (this *PML-role* information can also be used by applications to determine an adequate way to present a PML instance to the user). New schemata can be automatically generated out of the PML scheme, e.g. Relax NG [19]. This means that data consistence can be checked by common XML tools. Both versions of the schemata are available in the directory `data/schemas/`. An example of the w-layer part of the PML schema of the CAC can be found in Table 3.2 (`data/schemas/wdata_schema.xml`). In the illustrated example, the paragraph (type `para`, the whole document in the case of the CAC 2.0) consists of an array of `w-node.type` elements. This type is closely defined as a structure also containing obligatory elements: `id` (unambiguous identifier with the role of `#ID`) and `token` (word unit).

Table 3.2. The PML schema of the w-layer in the CAC 2.0

```
<type name="w-para.type">
    <sequence>
        <element name="w" type="w-node.type"/>
        ...
    </sequence>
</type>
<type name="w-node.type">
    <structure name="w-node">
        <member as_attribute="1" name="id" role="#ID" required="1">
          <cdata format="ID"/></member>
        <member name="token" required="1">
          <cdata format="any"/>
        </member>
        <member name="no_space_after" type="bool.type"/>
    </structure>
</type>
...
```

Every PML instance begins with a header referring to the PML schema. The header contains references to all external sources that are being referred to from this instance, together with some additional information necessary for the correct link resolving. The rest of the instance is dedicated to the annotation itself. Table 3.3 provides an example of the head of an m-layer instance (`n01w.m`) with a reference to a PML schema (`mdata_schema.xml`) and the appropriate instance within the w-layer (`n01w.w`).

Table 3.3. Part of the header of the m-layer instance `n01w.m`

```
<head>
    <schema href="mdata_schema.xml" />
    <references>
        <reffile id="w" href="n01w.w" name="wdata" />
    </references>
</head>
...
```

Table 3.4 similarly shows the referential part of the header of the instance of the a-layer (`n01w.a`), referring to the PML-schema of that instance (`adata_schema.xml`) and the corresponding m-layer instance (`n01w.m`) and w-layer instance (`n01w.w`).

Table 3.4. Part of the header of the a-layer instance `n01w.a`

```
<head>
    <schema href="adata_schema.xml" />
    <references>
        <reffile id="m" href="n01w.m" name="mdata" />
        <reffile id="w" href="n01w.w" name="wdata" />
    </references>
</head>
...
```

The annotation is expressed using XML elements and attributes named and used according to their corresponding PML schema. Table 3.5 illustrates an example of the morphological annotation of a part of the sentence *Váš boj je i naším bojem* (Lit.: *Your fight is our fight too*). The opening tag of the element `s` contains an identifier of the whole sentence followed by the opening tag of the element `m`,

which contains identifiers to the annotation corresponding to the token of the w-layer that are being referred to from the element `w.rf`. Other elements contain the form (`form`), morphological tag (`tag`) and `src.rf` provides the source of the annotation, in this case a manual annotation.

Table 3.5. An example of sentence m-layer annotation in the PML format

```
<s id="m-n01w-s14">
    <m id="m-n01w-s14W1">
        <src.rf>manual</src.rf>
        <w.rf>w#w-n01w-s14W1</w.rf>
        <form>Váš</form>
        <lemma>tvůj_^(přivlast.)</lemma>
        <tag>PSYS1-P2-------</tag>
    </m>
    <m id="m-n01w-s14W2">
        <src.rf>manual</src.rf>
        <w.rf>w#w-n01w-s14W2</w.rf>
        <form>boj</form>
        <lemma>boj</lemma>
        <tag>NNIS1-----A----</tag>
    </m>
    <m id="m-n01w-s14W3">
        <src.rf>manual</src.rf>
        <w.rf>w#w-n01w-s14W3</w.rf>
        <form>je</form>
        <lemma>být</lemma>
        <tag>VB-S---3P-AA---</tag>
    </m>
    ...

    <m id="m-n01w-s14W7">
        <src.rf>manual</src.rf>
        <form_change>insert</form_change>
        <form>.</form>
        <lemma>.</lemma>
        <tag>Z:-------------</tag>
    </m>
</s>
```

Table 3.6 shows an example of the analytic annotation of a sentence *Váš boj je i naším bojem.* (Lit.: *Your fight is our fight too.*) The less important elements have been left out to make the example more transparent. The dependency structure of the sentence is represented by structured nested elements. Daughter nodes are enveloped by the element `children`. Furthermore, each node is enveloped in the element `LM` with the identifier of this node as an attribute; lists of single nodes are the only exception, as this element can be omitted for them. The identifier of the node becomes an attribute of the element `children`. The element `m.rf` links to the corresponding element of the lower layer containing the particular word form. The element `afun` contains the analytical function of the node. The element `ord` contains the sequential number of the node in the tree in left-to-right order. This number is equal to the word order in the sentence.

Table 3.6. An example of sentence a-layer annotation in the PML format

```
<LM id="a-n01w-s14">
    <s.rf>m#m-n01w-s14</s.rf>
    <afun>AuxS</afun>
    <ord>0</ord>
    <children>
        <LM id="a-n01w-s14W3">
            <afun>Pred</afun>
            <m.rf>m#m-n01w-s14W3</m.rf>
            <ord>3</ord>
            <children>
                <LM id="a-n01w-s14W2">
                    <afun>Sb</afun>
                    <m.rf>m#m-n01w-s14W2</m.rf>
                    <ord>2</ord>
                    <children id="a-n01w-s14W1">
                        <afun>Atr</afun>
                        <m.rf>m#m-n01w-s14W1</m.rf>
                        <ord>1</ord>
                    </children>
                </LM>
                <LM id="a-n01w-s14W6">
                    <afun>Pnom</afun>
                    <m.rf>m#m-n01w-s14W6</m.rf>
                    <ord>6</ord>
                    <children id="a-n01w-s14W5">
                        <afun>Atr</afun>
                        <m.rf>m#m-n01w-s14W5</m.rf>
                        <ord>5</ord>
                        <children id="a-n01w-s14W4">
                            <afun>AuxZ</afun>
                            <m.rf>m#m-n01w-s14W4</m.rf>
                            <ord>4</ord>
                        </children>
                    </children>
                </LM>
            </children>
        </LM>
        <LM id="a-n01w-s14W7">
            <afun>AuxK</afun>
            <m.rf>m#m-n01w-s14W7</m.rf>
            <ord>7</ord>
        </LM>
    </children>
</LM>
```

XML elements of a PML instance occupy a dedicated namespace: `http://ufal.mff.cuni.cz/pdt/pml/` (this is not a real link, it is just a name of the namespace). The PML format offers unified representations for the most common annotation constructs, such as attribute-value structures, lists of alternative values of a certain type (either atomic or further structured), references within a PML instance, links among various PML instances (used in the CAC 2.0 to create links across layers), and links to other external XML-based resources.

CSTS format

A single file in CSTS format can contain all layers of annotation.

A CSTS format file opens with a (facultative) header (element `h`) followed by at least one `doc` element. The element `doc` consists of a header (element `a`) and contents (element `c`). The element `c` is then formed by a sequence of paragraphs (element `p`) and sentences of those paragraphs (element `s`).

Each word token of the sentence is placed on a separate line in the file (element `f` or `d` for punctuation). The line continues with the annotations of this word token on all layers. The element `l` is filled with the lemma, the element `t` contains its morphological tag. The element `A` is filled with the analytical function of the word token. The unique identifier of the word token in the sentence is stored in the element `r`. The element `g` contains a link to the governing node of the word in the form of an identifier of that governing node.

See Table 3.7 for an example of the complete annotation of the sentence *Váš boj je i naším bojem.* (Lit.: *Your fight is our fight too.*) in CSTS format.

Table 3.7. An example of sentence annotation in CSTS format

```
<s id=n01w-s14>
<f id=n01w-s14W1>Váš<l>tvůj_^(přivlast.)<t>PSYS1-P2-------
    <r>1<g>2<A>Atr
<f id=n01w-s14W2>boj<l>boj<t>NNIS1-----A----<r>2<g>3<A>Sb
<f id=n01w-s14W3>je<l>být<t>VB-S---3P-AA---<r>3<g>0<A>Pred
<f id=n01w-s14W4>i<l>i<t>J^-------------<r>4<g>5<A>AuxZ
<f id=n01w-s14W5>naším<l>můj_^(přivlast.)<t>PSZS7-P1-------
    <r>5<g>6<A>Atr
<f id=n01w-s14W6>bojem<l>boj<t>NNIS7-----A----<r>6<g>3<A>Pnom
<D>
<d id=n01w-s14W7>.<l>.<t>Z:-------------<r>7<g>0<A>AuxK
```

The DTD file for CSTS format can be found in the directory `data/schemas/`. For more detailed information on this format see the PDT 2.0 documentation [13].

Directories `tools/tool_chain/csts2pml/` and `tools/tool_chain/pml2csts/` provide conversion scripts for the two formats.

3.2.2. File naming conventions

Each data file used in the CAC 2.0 relates to one annotated document. The base of the file name contains a single letter that classifies the subject of the text contained in the file. Namely `n` indicates newspaper articles, `s` marks scientific texts, and `a` denotes administrative texts. Next, the file name specifies a two-digit ordinal number of the document within a group of documents of the same style. Following this two-digit number, a letter indicates if the text is derived from a written text (letter `w`) or if it is a transcript of spoken language (letter `s`). The file names of the documents are included as the identifiers of sentences and elements in these sentences, e.g. `<m id="m-n01w-s1W1">` in table 3.5. See Appendix A for file names of each document.

Example: Instances noted according to template `a[0-9][0-9]s*` contain transcripts of the spoken language in an administrative style.

In PML format, the file extension embodies the layer of the document's annotation. The extension of w-layer files is `.w`, `.m` denotes m-layer and `.a` denotes a-layer. Then they will be referred to as w-files, m-files and a-files. Each a-file exactly corresponds to one m-file and one w-file. Each a-file contains links to the corresponding m-file and w-file, and each m-file contains links to the corresponding w-file (see above). Due to this dependency, it is critical that files not be renamed. There are no links from w-files to m-files (or a-files), as well as there are no links from m-files into a-files.

In CSTS format, there is the "csts" extension for all the files.

Example: The code `s17w.a` defines a PML instance containing the a-layer annotations of a document written in a scientific style. The file links to `s17w.m` and `s17w.w` files, file `s17w.m` links to `s17w.w` file. The code `s17w.csts` defines a CSTS file containing all layers (w-layer, m-layer, a-layer) annotation of a document written in a scientific style.

3.2.3. Data size

The CAC 2.0 is composed of 180 manually annotated documents containing 31,707 sentences and 652,131 tokens as calculated from the m-files. Tokens without punctuation total 570,760 and tokens without punctuation and digit tokens reach 565,910. Table 3.8 states the sizes of the individual parts of the data according to its style and form.

Table 3.8. Size of the CAC 2.0 parts according to style and form

Style	Form	Number of docs	Number of sentences	Number of word tokens	Number of word tokens w/o punctuation	Number of word tokens w/o punctuation and digit tokens
Journalism	Written	52	10 234	189 435	165 469	163 693
Journalism	Transcription	8	1433	28 737	24 864	24 859
Scientific	Written	68	11 113	245 174	216 280	214 127
Scientific	Transcription	32	4576	115 853	100 281	100 272
Administrative	Written	16	3362	58 697	51 431	50 524
Administrative	Transcription	4	989	14 235	12 435	12 435
Total	**Written**	**136**	**24 709**	**493 306**	**433 180**	**428 344**
Total	**Transcription**	**44**	**6998**	**158 825**	**137 580**	**137 566**
Total	**Written and transcription**	**180**	**31 707**	**652 131**	**570 760**	**565 910**

Table 3.9 contains separate quantitative data for the characters “#” and “?” that were manually inserted into the CAC to replace missing words and numbers written as digits.

Table 3.9. Quantitative characteristics of the CAC 2.0 – replacement characters “#” and “?”

Style	Form	Number of “#” characters (in a specified number of sentences)	Number of “?” (in a specified number of sentences)	Number of “#” or “?” (in a specified number of sentences)	Number of sentences not containing replacement symbols
Journalism	Written	1,776 (1,187)	925 (680)	2,701 (1,563)	8,671
Journalism	Transcription	5 (5)	25 (25)	30 (30)	1,403
Scientific	Written	2,153 (1,224)	2,230 (1,418)	4,383 (2,031)	9,082
Scientific	Transcription	9 (9)	1,31 (108)	140 (113)	4,463
Administrative	Written	907 (616)	635 (476)	1,542 (919)	2,443
Administrative	Transcription	0 (0)	16 (15)	16 (15)	974

Every experiment conducted on the CAC 2.0 data made public should contain information about the data that was used to obtain the derived results.

The Annotation of the CAC 2.0 is divided into three layers: the w-layer (word layer), m-layer (morphological layer) and a-layer (analytical layer). Each of these layers includes its own PML schema located in the directory structure (`data/schemas/` files `wdata_schema.xml`, `mdata_schema.xml`, `adata_schema.xml`). The directory structure `data/pml/` is composed of a total of 496 files: 180 w-files, 180 m-files and 136 a-files. Transcriptions have not been annotated on the a-layer. It is impossible to apply the guidelines for the syntactical annotation of the written texts to the annotation of the spoken texts.
The directory `data/csts/` contains 180 files of this same data in CSTS format: 136 consist of morphological and syntactical annotations and 44 only morphological annotations.

With regards to target to integrate the CAC into the PDT, we present Table 3.10 that compares the basics of both corpora. We only mention the characteristics common to both corpora. The CAC 2.0 will be integrated into the PDT when the next version of the PDT is published.

Table 3.10. A comparison of the CAC 2.0 and the PDT 2.0

Characteristics	PDT 2.0		CAC 2.0	
	Number of words (thousands)	Number of sentences (thousands)	Number of words (thousands)	Number of sentences (thousands)
Morphological annotation	2,000	116	652	32
Analytical annotation	1,500	88	493	25
Written form	2,000	116	493	25
Transcriptions	--	--	159	7
Journalistic style	1,620	94	218	12
Administrative style	--	--	73	4
Scientific style	380	22	361	16

3.3. Tools

We provide the whole range of tools for data annotations, annotation corrections, searching within the annotated data and automatic data processing. Considering the fact that the CAC 2.0 is annotated on the m-layer and a-layer, we provide the tools for working with the CAC (and other) data on these two layers. Table 3.11 helps the user to orient himself to the tools contained on this CD-ROM. Each tool is described by its main features and its appointed kind of use. The following sections describes the tools in more detail.

Table 3.11. Tools – outline

Tool	Description	Purpose
Bonito	Corpus manager	• Searching within CAC 2.0 texts • Searching within the morphological annotations of the CAC 2.0 • Searching within the analytical functions assigned to words in the CAC 2.0 as a part of the a-layer • Basic statistics on the CAC 2.0
LAW	Morphological annotations editor	• Morphological annotation (manual disambiguation of morphological analysis results)
TrEd	Syntactical annotations editor	• Syntactical annotations (assigning analytical functions and syntactical dependencies)
Netgraph	Corpus viewer	• Searching within the trees in the CAC 2.0
tool_chain	Automatic procedure processing Czech texts	• Tokenisation • Morphological analysis • Tagging (automatic disambiguation of morphological analysis results) • Parsing (automatic syntactical analysis with analytical functions assignment)

3.3.1. Corpus manager Bonito

The graphic tool Bonito [32] simplifies tasks commonly associated with language corpora, especially searching within them and calculating basic statistics on the search results. Bonito is a graphical interface to the corpus manager Manatee, which conducts various operations on corpus data. A detailed documentation for the Bonito tool is included in the application itself and can be launched from the main `Help` menu.

Figure 3.1 illustrates the Bonito main screen. The command of the tool is demonstrated in the following examples.

Figure 3.1. Bonito: Main screen

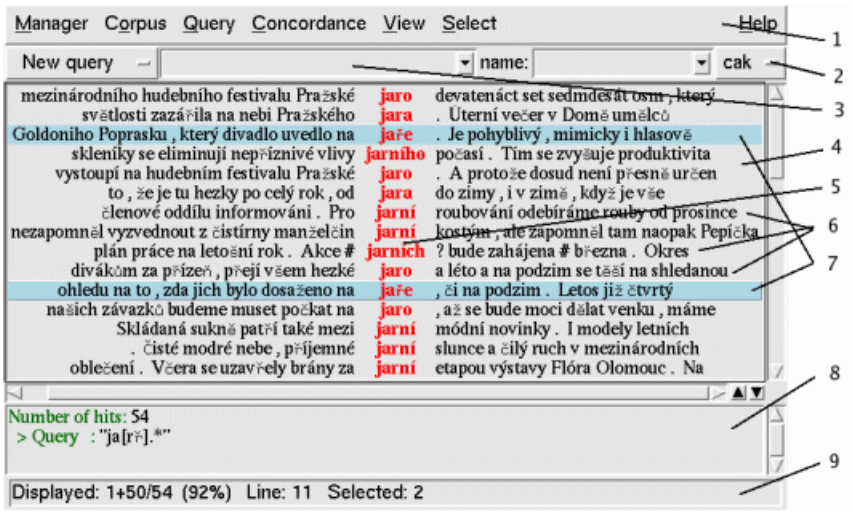
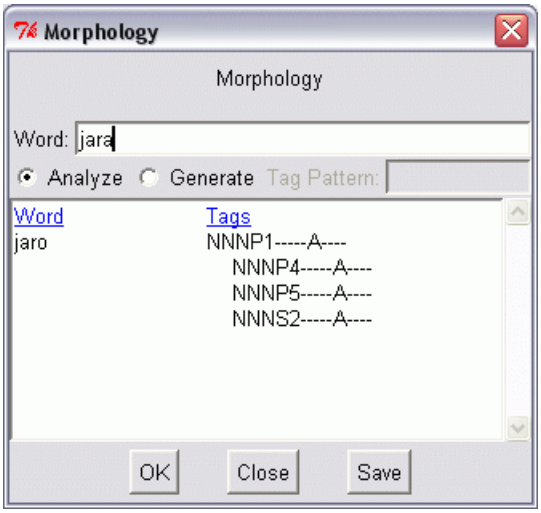


Figure 3.1 description

- 1 Main menu
- 2 Corpus selection button
- 3 Query line
- 4 Main window displaying query results
- 5 Column of the query results
- 6 Concordance lines
- 7 Selected concordance lines
- 8 Window for displaying query history and broader context
- 9 Status line

Bonito makes it possible to run the Czech morphological analyser directly through the menu `Manager | Morphology`. This command opens a new window; the user can keep this window open while working with the corpus tool. It can be used to run morphological analysis or synthesis (generating). The morphological analysis of a given word lists all possible lemmas and tags corresponding to the entered word form. In case a synthesis is selected, the tool generates all possible word forms that can be generated from the given lemma and the corresponding tags. See Figure 3.2.

Figure 3.2. Bonito: Running the morphological analyser



The tutorial [/tutorials/bonito-text_en.htm] contains more detailed information how to master Bonito.

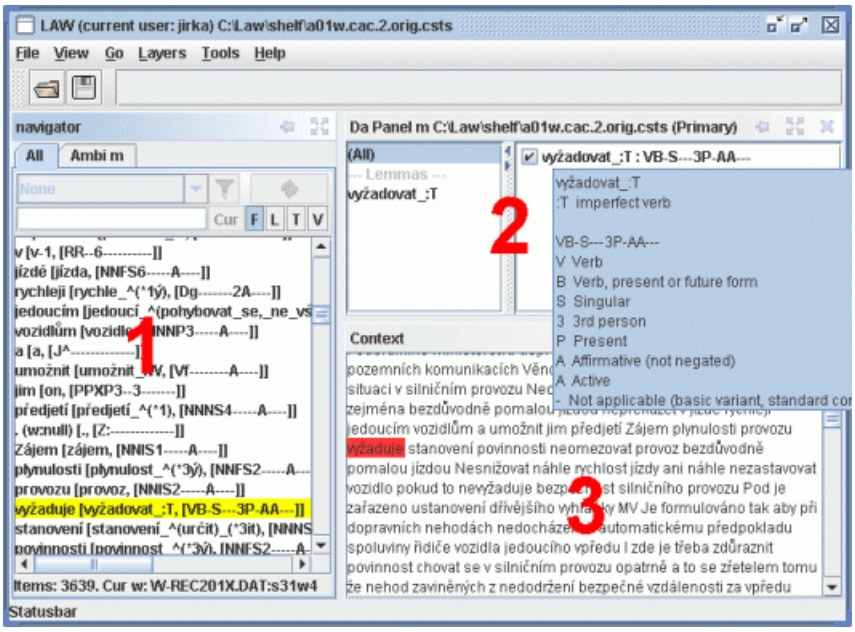
3.3.2. LAW – Editor for morphological annotation

The Lexical Annotation Workbench (LAW, [33]) is an integrated environment for morphological annotation. It supports simple morphological annotation (assigning a lemma and tag to a word), the comparison of different annotations of the same text, and searching for a particular word, tag etc. The workbench runs on all operating systems supporting Java, including Windows and Linux. It is an open system extensible via external modules – e.g. for different data views, import/export filters, assistants. The LAW editor supports PML [15], CSTS [13] and TNT [38] formats.

Major components

The application consists of three major components as shown in Figure 3.3.

Figure 3.3. LAW: Main screen



1. *Navigator* – For navigating through words of the document that have been filtered by different criteria and the selection of words for disambiguation.

2. *Da Panels* – For displaying and disambiguating morphological information (lemmas, tags) of a word. The panel consists of two windows – a grouping list and a list of items. The latter displays all the lemma-tag pairs associated with the current word (on the particular m-layer). The former makes it possible to restrict the items to a particular group, e.g., items with a particular lemma, detailed pos or gender. One of the panels is always defined as *primary* – certain actions apply to that panel only (e.g. **Ctrl-T** activates the list of lemmas and tags in the main panel).

3. *Context Windows* – Contain various context information, e.g. plain text of the document, syntactic structures, etc.

The usual workflow

The usual annotation work proceeds as follows:

1. Open the desired m-file: `File | Open` (**Ctrl-O**). The associated w-file opens automatically.

2. Switch to the ambi-list (Ambi+ name of m-file) in the Navigator that is displaying the ambiguous words (words with more than one result of the morphological analysis) and select the first word.

3. Press `Enter`. The cursor moves to the primary Da Panel. Select the correct lemma and tag and press `Enter` again. The cursor will move to the next ambiguous word.

   In case you make a mistake, switch to the list of all entries in the Navigator (All), find the word you want to review and select it. The Da Panel will display the corresponding annotation. You can now select the correct lemma and tag and then switch back to the Ambi X list.

4. Save the annotations: `File | Save` (**Ctrl-S**).

3.3.3. TrEd – Editor for syntactical annotation

The Tree Editor (TrEd, [37]) is a fully integrated environment primarily designed for the syntactical annotations of tree structures assigned to sentences. The editor can also be used for data viewing and searching with the help of several kinds of search functions.

The TrEd supports the PML and CSTS formats of input and output. More details on these formats can be found in 3.2.1. The TrEd system is highly modular, which means support for other formats can be easily plugged in.

The TrEd offers various possibilities of custom settings. User-defined macros in the Perl language can extend its functionality. Macros are called upon from menus or through the assigned hotkeys.

Users oriented with programming will certainly be able utilise the TrEd version without graphical user interface – called “btred” – for batch data processing (the Batch-mode Tree Editor). The NTrEd tool is another add-on to the editor. It brings with it the possibility to parallelise the “btred” processes and to distribute them on more computing machines.

To open the files in the TrEd use the menu command `File | Open`. Choose a file with the extension *.a or *.csts. The file opens in the TrEd and the first sentence of the file displays on the screen.

Figure 3.4 shows a typical TrEd screen. The sentence *Problémy motivace jsou tak staré jako lidstvo.* (E.: *The motivational problems are as old as the human race.*) Please find the explanatory notes below.

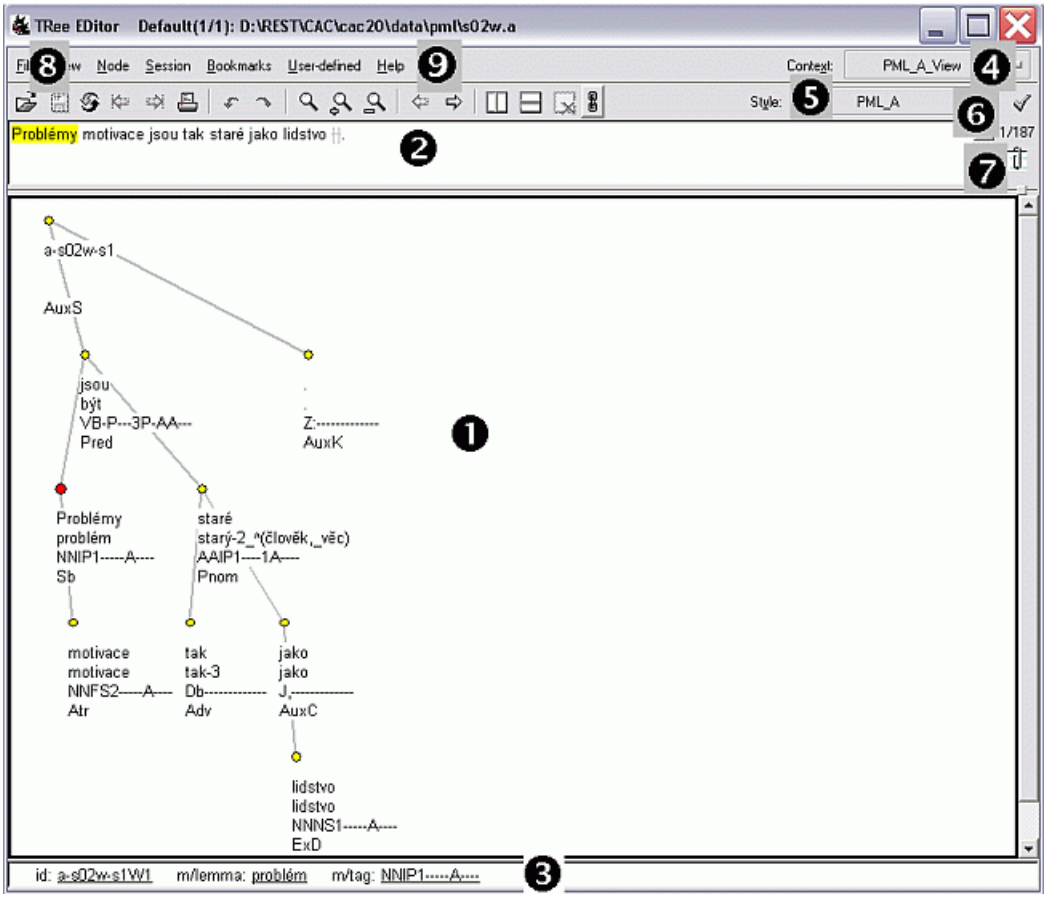
- 1. A window shows the tree representing the syntactical annotation of the sentence.

- 2. The represented sentence.

- 3. Status line: The status line shows various information on the selected word (the highlighted node, in our case *Problémy*). In our example the ID number of the node, its lemma and tag are displayed.

- 4. Current context. The environment for working with the annotations is called the context. There is a context which only allows the user to view the annotations (e.g. the PML_A_View context serves for viewing the syntactical annotations), another context might enable changing the annotations (e.g. the PML_A_Edit context allows for editing the annotations). To change the context, click on the current context name and choose another context from the pop-up list.
- 5. Current display style. The display style can be changed in the same way as the context.
- 6. Editing the display style.
- 7. Viewing the list of all sentences in the open file.
- 8. Buttons for opening, saving and re-opening a file.
- 9. Buttons for moving to the previous or following tree in the open file and for window management.

Figure 3.4. TrEd: Main screen



The CAC 2.0 files open in the PML_A_View context by default. In this context the user can view the trees and the editing is disabled. In case you wish to edit the trees, switch to the PML_A_Edit context. Both contexts offer only a single display style – PML_A. To view the list of all defined macros and the hotkeys assigned to them for any currently used context choose `View | List of Named Macros` from the menu.

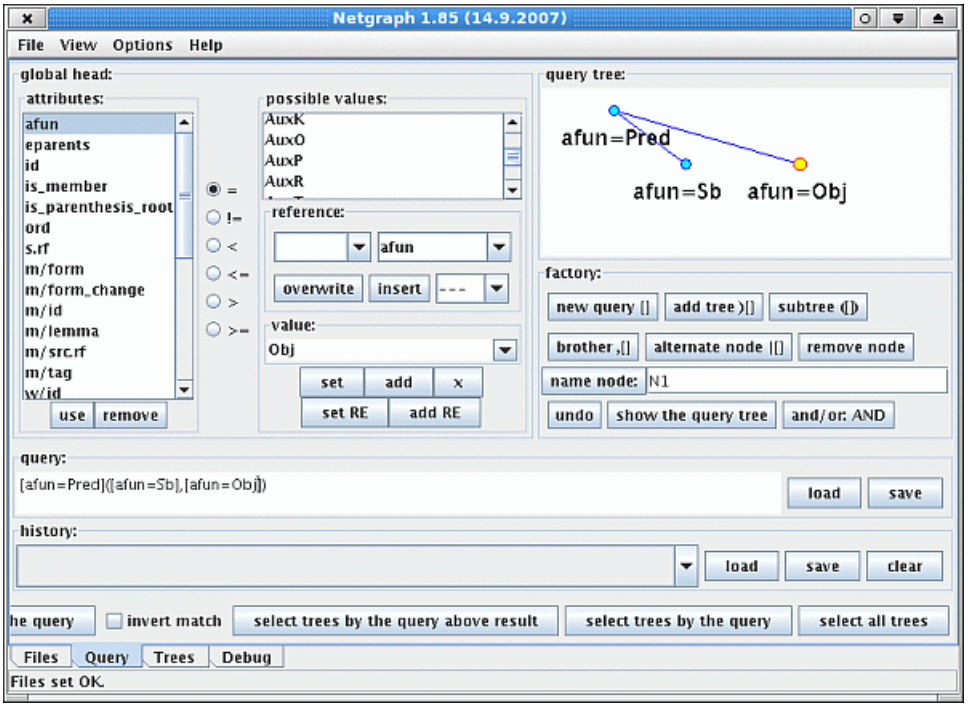
3.3.4. Corpus viewer Netgraph

Netgraph [35] is a client-server application for searching through and viewing the CAC 2.0. Several users can view the corpus online at the same time. The Netgraph has been designed for simple and intuitive searching while maintaining the high search power of the query language (Mírovský, 2008).

A query in Netgraph is formulated as a node or tree with defined characteristics that should match the required trees in the corpus. Therefore, searching the corpus means searching for sentences (annotated into the form of trees) containing the given node or tree. The user's queries can range from the very simple (e.g. searching for all trees in the corpus containing a desired word) to the more advanced queries (e.g. searching for all sentences containing a verb with a dependent object, where the object is not in dative, and there is at least one dependent adverbial, etc.). So called *meta attributes* enable searching for even more complex structures.

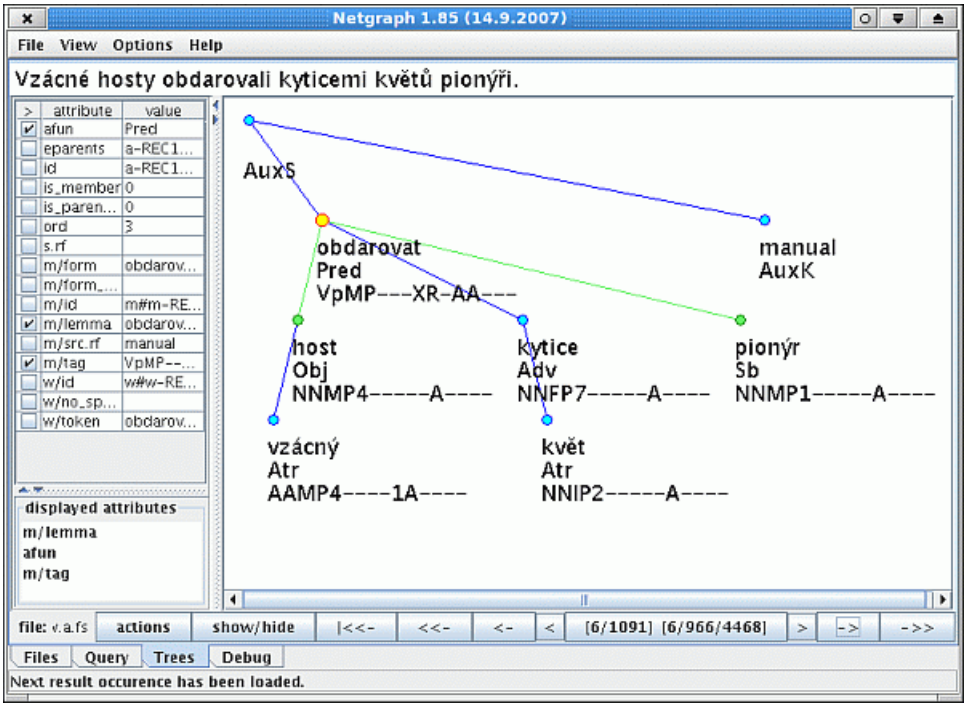
The Netgraph tool offers a user friendly graphical interface for query formulation. See Figure 3.5 as an example. This simple query searches for all the trees containing a node marked as the predicate that has at least two dependent nodes marked as subject and object. The order of these dependent nodes is not specified in the query.

Figure 3.5. Netgraph: Query formulation



The tree in Figure 3.6 could be one of the results the server returns.

Figure 3.6. Netgraph: Query result



Users always use the client side of the Netgraph application. The client connects to the public server `quest.ms.mff.cuni.cz` through the `2001` port . Another possibility for the user is to install the server part of the application and then search the corpus offline.

3.3.5. The automatic processing of texts

The data and applications for the morphological and syntactical analysis of the Czech texts were developed simultaneously. The CD-ROM contains two fundamental morphological applications – **morphological analysis** and **tagging** – and one syntactical application – **parsing**. Also, the procedure for **tokenisation** is included.

Tokenistion is the process of splitting the given text into word tokens. Its result is so-called “vertical” which means it is a file containing each word or punctuation on a separate line. The term tokenization is often used for both splitting the text into words and segmentation, i.e. marking sentence and paragraph boundaries. Our tokenisation procedure also segments the text.

However we understand tokenisation even more broadly – the procedure vertically converts into the CSTS format (see Section 3.2.1). This conversion includes: adding the file header to the beginning of the vertical column and marking each word with a simple tag distinguishing the word properties that are clear straight from the orthographic form of the word. Punctuation, digits or words containing digits are especially marked. The upper case words and words beginning with upper case letters are marked with special tags, too. The resulting vertical column in the CSTS format serves as the input for further processing.

The morphological analysis evaluates individual word forms and determines lemmas as well as possible morphological interpretations for the word form.

The morphological analysis is based on the morphological dictionary containing part of speech information on Czech word forms. Each word form is assigned a morphological tag describing the morphological characteristics of the word form. The morphological dictionary used for the analysis contains additional information for many lemmas – style, semantics or derivational information. The lemmas of abbreviations are often enriched by comments referring to the explanatory text in Attachment B.

Due to the high homonymy of the Czech language, most word forms can be assigned more morphological tags or even more lemmas. For example, the word form *pekla* has two lemmas – noun *peklo* (*hell*) and verb *péci* (*to bake*). Both lemmas generate several tags for the given word form. The morphological analysis compares the possible word forms from the whole corpus to the word forms contained in the morphological dictionary. The corresponding lemmas and tags are assigned to the given word form in case they match. Therefore a set of pairs "lemma – morphological tag" is the result of the morphological analysis for each word form.

The morphological analysis is followed by tagging (also called disambiguation). In this phase the right combination of the lemma and tag for the given context is selected from the set of all possible lemmas and tags. Regarding the character of the task, it is impossible to generate a method of tagging that would function with 100 percent accuracy. The program carrying out the tagging is called *tagger*. The tagger application included on the CD-ROM is based on the *Hidden Markov Model* (HMM) and implements the use of the *averaged perceptron* statistical method (Collins, 2002): The method is statistically based. A text that contains the set of all possible morphological tags and lemmas for every word (the output from the morphological analysis) is the input for the tagger. In the output, the tagger defines this dataset with an unambiguously determined tag and its corresponding lemma. The tagger was trained on data in the PDT 2.0.

After tagging the next step of text processing is parsing. The parsing procedure assigns each word in the sentence its syntactical dependency on another word along with its analytical function. The program carrying out the parsing is called *parser*. The parser included in the CD-ROM is based on the same methodology as the tagger. The input of the parser is a text consisting of words labelled by a single pair lemma-tag. The output is a tree structure labelled by analytical functions for each sentence. The parser has been trained on the PDT 2.0 training data.

The script `tool_chain` is provided for the user's convenience. This script uses basic switches to run the needed tool. For the switches documentation see Table 3.12. Concatenating more switches enables running more tools in sequence.

Example: The following command morphologically analyses raw text: `tool_chain -tA`

Note: When working with files in the PML format, the directory containing the input file of the `tool_chain` script must contain all files linked from the processed file. In case the m-file serves as input, it has to be "accompanied" by the corresponding w-file.

Table 3.12. Script `tool_chain`

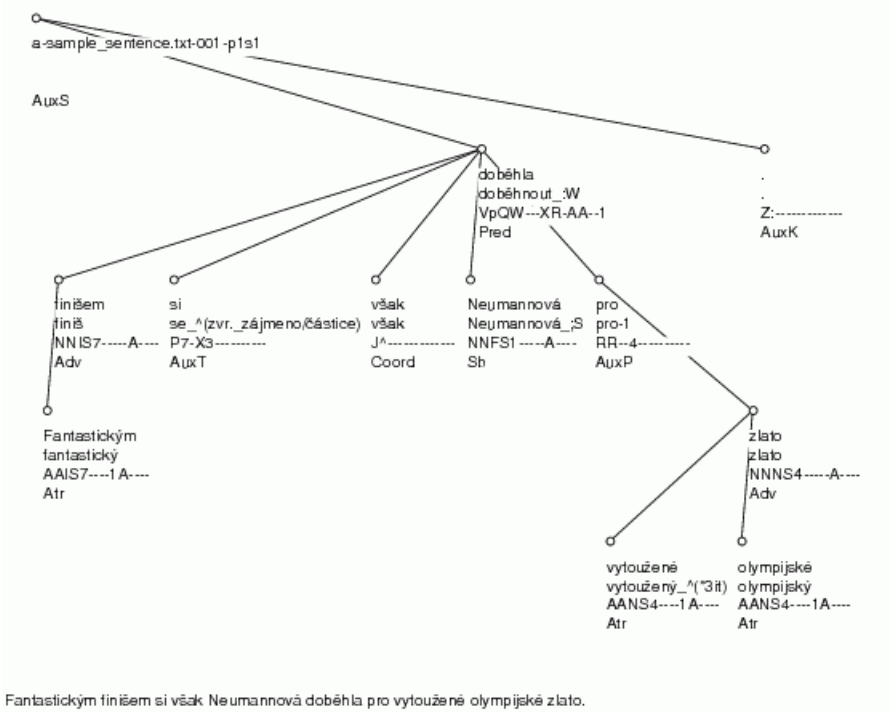
Parameter	Processing type	Input file format	Output file format
`-t`	Tokenisation	Raw text	CSTS
`-A`	Morphological analysis	CSTS	PML m-file, CSTS
`-T`	Tagging	PML m-file, CSTS (morphological analysis output)	PML m-file, CSTS
`-P`	Parsing	PML m-file, CSTS	PML a-file, CSTS

Example: Let´s have a look at the analysis of *Fantastickým finišem si však Neumannová doběhla pro vytoužené olympijské zlato* (E.: *Neumannova powered down the final straight to win the longed-for gold*). The results of the morphological analysis (run by the command `tool_chain -tA`) and tagging (run by the command `tool_chain -T`) is summarized Table 3.13. In case more possible lemmas exist for the given word form (e. g. the word form *si* is analysed either as the verb *být* (to be) or as the reflexive particle *se*) the word form possibilities are separated with the pipe symbol "|". To spare the reader from searching for errors the tagger itself made, we confirm that there are no errors in this output. Figure 3.7 shows the parsing result (parsing run by the command `tool_chain -P`). Each node of the tree displays a word form, disambiguated lemma, disambiguated morphological tag and analytic function. To spare the reader from searching for errors the parser has made, we confirm that there are no errors in this output.

Table 3.13. An example of text treated with morphological analysis and tagging

Text	Morphological analysis	Tagging
Fantastickým	*fantastický* AAFP3----1A---- AAIP3----1A---- AAIS6----1A---7 AAIS7----1A---- AAMP3----1A---- AAMS6----1A---7 AAMS7----1A---- AANP3----1A---- AANS6----1A---7 AANS7----1A----	*fantastický* AAIS7----1A----
finišem	*finiš* NNIS7-----A----	*finiš* NNIS7-----A----
si	*být* VB-S---2P-AA--7| *se_ ^(zvr._zájmeno/částice)* P7-X3----------	*se_^(zvr._zájmeno/částice)* P7-X3----------
však	*však* J^----------	*však* J^----------
Neumannová	*Neumannová_;S* NNFS1-----A---- NNFS5-----A----	*N e u m a n n o v á _ ; S* NNFS1-----A----
doběhla	*doběhnout_:W* VpQW---XR-AA--1	*d o b ě h n o u t _ : W* VpQW---XR-AA--1
pro	*pro-1* RR--4----------	*pro-1* RR--4----------
vytoužené	*vytoužený_^(*3it)* AAFP1----1A---- AAFP4----1A---- AAFP5----1A---- AAFS2----1A---- AAFS3----1A---- AAFS6----1A---- AAIP1----1A---- AAIP4----1A---- AAIP5----1A---- AAMP4----1A---- AANS1----1A---- AANS4----1A---- AANS5----1A----	*v y t o u ž e n ý _ ^ (* 3 i t)* AANS4----1A----
olympijské	*olympijský* AAFP1----1A---- AAFP4----1A---- AAFP5----1A---- AAFS2----1A---- AAFS3----1A---- AAFS6----1A---- AAIP1----1A---- AAIP4----1A---- AAIP5----1A---- AAMP4----1A---- AANS1----1A---- AANS4----1A---- AANS5----1A----	*olympijský* A ANS4----1A----
zlato	*zlato* NNNS1-----A---- NNNS4-----A---- NNNS5-----A----	*zlato* NNNS4-----A----
.	. Z:----------	. Z:----------

Figure 3.7. An example of sentence parsing



We recommend the users to test the tools by running the script `tool_chain -tA` on an arbitrary Czech text. The results of the script can be opened in the LAW tool, which also enables the disambiguation of the assigned tags.

Run the script `tool_chain -P` on the manually disambiguated file. The result of the script can be opened in the TrEd tool, which also enables correcting the dependencies and analytic functions.

Chapter 4. Bonus material

4.1. The STYX electronic exercise book

The bonus material is aimed at advanced students in primary and high schools and their respective teachers. The bonus material section labelled STYX [36] presents the user with an electronic exercise book for practising Czech morphology and syntax. The most noteworthy feature of this material is the number of sentences offered: More than 11,000 sentences have been compiled along with the corresponding annotations in the PDT to facilitate effective training. In addition to this large vocabulary, the application provides immediate verification of user's parsing accuracy. It is important to stress that the academic notion of Czech syntax (presented in the PDT 2.0) differs in some ways from the concepts traditionally taught in the school system. These differences are closely documented (Kučera, 2006). Each exercise processes an arbitrary number of sentences according to Czech syntax: Each word in the sentence will be morphologically analysed and the entire sentence will be parsed including determining the constituents of the sentence. Only a small subset of the 11,000 sentences is available on the CD-ROM to avoid overloading the user – 50 sentences (see `bonus-tracks/STYX/sample.styx`).

The steps for using STYX are clearly illustrated in Figure 4.1. First, the user selects the part of speech associated with each word and then (s)he determines the morphological analysis and appropriate morphological categories (upper part of the right window). The word nodes are juxtaposed together at the beginning of the parsing and each node is removed when it has been successfully parsed. The next step leads to determining the constituents of the sentence including the basic clause elements (predicate and subject). Figure 4.2 demonstrates the parsing evaluation process. The user in our example morphologically analysed the word *předměty* (E: *subjects*) correctly; also the syntax and analytical functions analysis is correct (the top tree has been constructed by the user, the lower tree serves for evaluation purposes).

Figure 4.1. STYX: Exercises

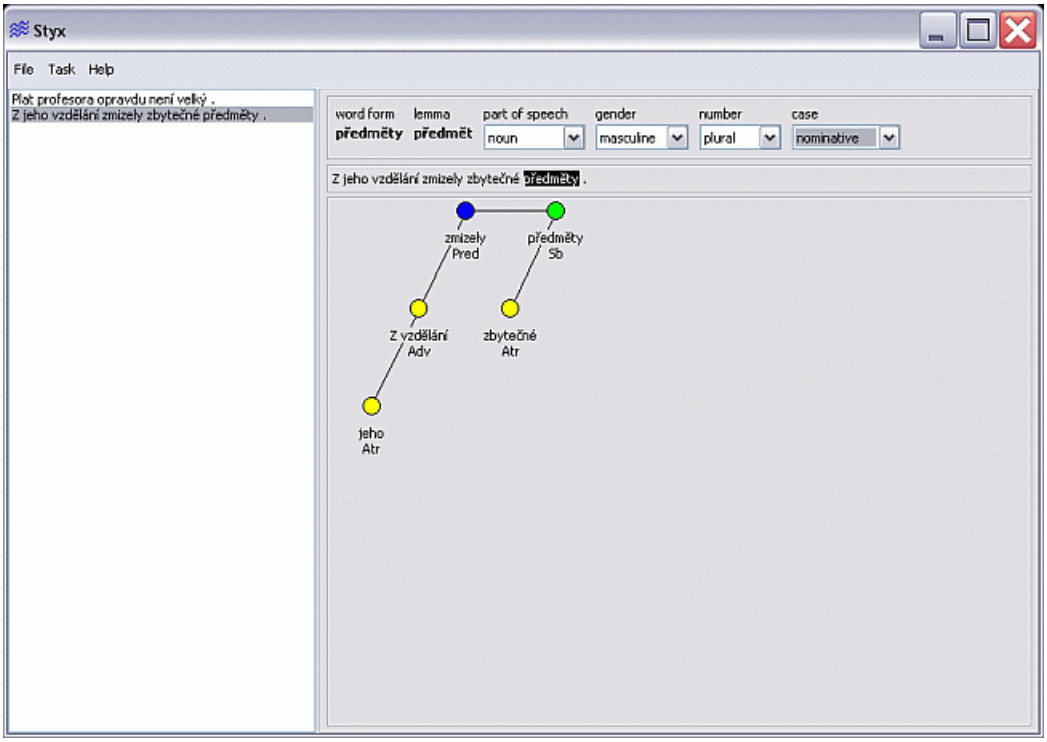
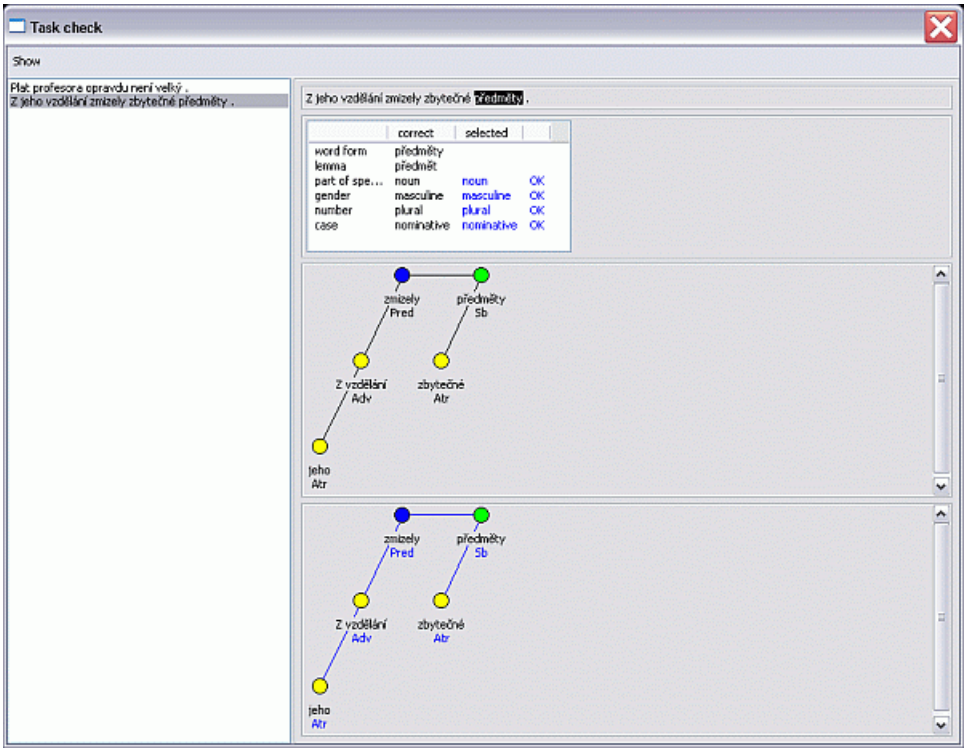


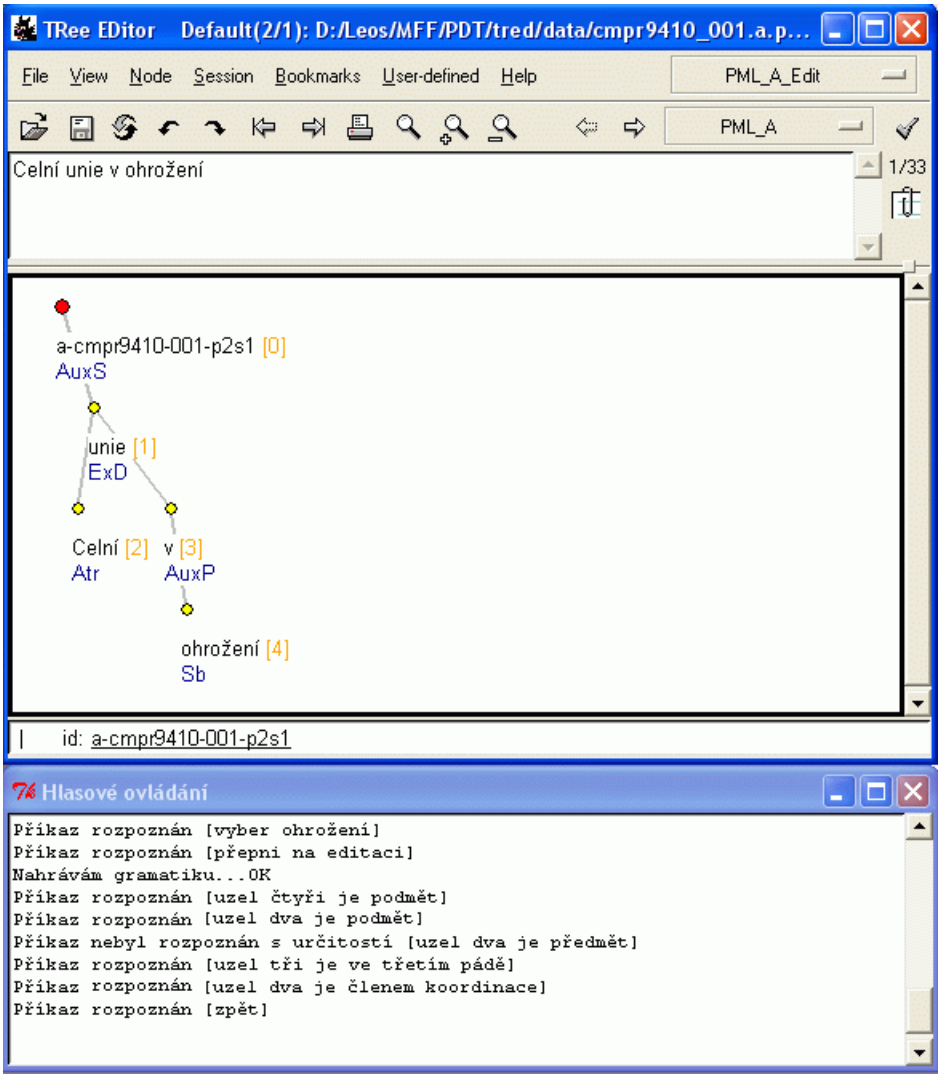
Figure 4.2. STYX: Exercise evaluation



4.2. Voice control of the TrEd editor via the TrEdVoice module

The TrEd annotation editor is the essential annotation tool used to annotate the CAC 2.0 on the analytical layer (see Chapter 3.3.3). From the very beginning the TrEd was equipped with many complex functions and macros, and their number even increased over time. Most of the functions are assigned hotkeys, as it would be extremely time consuming to call upon all the functions from the menu system each time. Nevertheless, the system that consists of a large number of hotkeys is also complicated for the user's memory. One of the ways of how to rid the user from these complications is the voice control system, which is quite rarely used for application programs. That was why we have developed the TrEdVoice module (Přikryl, 2007). This module's purpose was not to create a complete voice control of all TrEd functions and enable its full control without using the keyboard and mouse. However, it is a useful accessory extending the original control possibilities (menus, hotkeys and mouse). Figure 4.3 shows the main TrEd screen with voice control enabled. The automatic speech recognition module (so-called ASR module) created by the Department of Cybernetics of the University of West Bohemia in Plzen's team [6] (Müller, Psutka, Šmídl, 2000) is used for voice commands recognition. The ASR module is not embodied into the TrEdVoice, it runs independently as the ASR server and the TCP/IP network protocol is used to communicate with the TrEdVoice. The ASR module is based on statistics and it is speaker-independent, which means it can recognise an arbitrary speaker's voice. For more details on voice recognition see (Psutka, Müller, Matoušek, Radová, 2006).

Figure 4.3. The TrEd editor screen with the TrEdVoice module enabled



Chapter 5. Tutorials

We provide two kinds of tutorials to simplify introducing the data and the tools to the user. Mainly, there are videos and handouts of the lectures given at the tutorial on the PDT (Prague Treebanking for Everyone: A two-day tutorial [28]) held in the autumn of 2006. The videos and text documents provided are in English. The second kind of tutorials are the demos guiding the user through the graphical interface controls of the provided tools. The demos are placed directly on the CD-ROM, while the videos are linked from an external source. Table 5.1 lists all tutorials (videos) concerning the data: the tutorials on annotation layers (m-layer, a-layer) and the tutorial on the inner data representation (PML format). Table 5.2 lists all tutorials (videos, demos and texts) concerning the tools.

Table 5.1. Data tutorials

Video clip
m-layer [23]
a-layer [22]
PML [27]

Table 5.2. Tool tutorials

Video clip	Demo	Text
Bonito [24]	Bonito [/tutorials/bonito_en.htm]	Bonito [/tutorials/bonito-text_en.htm]
LAW [25]	LAW [/tutorials/law_en.htm]	---
TrEd [30]	TrEd [/tutorials/tred_en.htm]	bTrEd [12]
Netgraph [26]	Netgraph [/tutorials/netgraph_en.htm]	---
STYX [29]	STYX [/tutorials/styx_en.htm]	---
---	TrEdVoice [/tutorials/tredVoice_cs.htm]	---

Chapter 6. Installation

To streamline your work with the CAC 2.0 we provide “installation” programs for Linux and MS Windows operation systems. Please note that in both operating systems **the components of the CD-ROM are copied to the hard drive, not installed**. Users must install the selected tools themselves – the `README_EN.txt` file with the installation instructions is available for every tool in its home directory within the CD directory. This file contains the system requirements, documentation references and installation instructions. Most parts of the CAC 2.0 can also be used directly from the distributed CD-ROM or its copies. Table 6.1 summarises all tools contained on the CD-ROM and the possibility to run them in Linux and MS Windows operating systems.

Table 6.1. Tools compatibility with Linux and MS Windows operating systems

Tool	Linux	MS Windows
Bonito	yes	yes
LAW	yes	yes
STYX	yes	yes
TrEd	yes	yes
TrEdVoice	no	yes
Netgraph	yes	yes
tool_chain	yes	no

Use the following commands to run the “Installation“:

- **Installation in Linux OS.** Run the program `Install-on-Linux.pl` from the root directory of the CD-ROM.
- **Installation in MS Windows.** Launch the installation program by double-clicking the `Install-on-Windows.exe` icon in the root directory of the distribution.

The installation process starts with one of these two types of installation. The user is then prompted to enter the destination folder (the structure of the destination folder will follow the directory structure of the CD-ROM):

- **Basic** – Copies of the documentation, tutorials and installation packages of Bonito, TrEd (including the TrEdVoice module for voice control in MS Windows) and STYX tools.
- **Custom** – Copies all components selected by the user from the CD-ROM.

Warning for CD-ROM CAC 1.0 users: The installation programs contained on the CD-ROM CAC 2.0 are independent of CAC 1.0 installation. We recommend installing all the tools that were part of the CAC 1.0 installation again from the CAC 2.0 CD-ROM. The CAC 2.0 distribution contains updated versions of the tools.

Warning for Bonito tool users: To search within the CAC 2.0 using the Bonito tool **it is not necessary** to copy the CAC 2.0 in `XML` format from the `data/pml` directory.

Warning for TrEd and TrEdVoice tool users: The TrEdVoice module for the voice control of the TrEd tool can only be used in MS Windows OS. Installing the TrEd in MS Windows using the installation package distributed with the CAC 2.0 (`tools/TrEd/tred_wininst_en.zip`) also installs the TrEdVoice tool. Please note that even though the TrEdVoice is offered as bonus material, its user manual is placed in the directory `tools/TrEd/docs/` (not in `bonus-tracks/`) due to the TrEdVoice’s close interconnection with the TrEd.

Chapter 7. Distribution and license information

The full distribution of the CAC 2.0 CD-ROM can be ordered from the Linguistic Data Consortium [10] publishing house; during the ordering process you will be redirected to the license agreement web page (see the license agreement text at http://ufal.mff.cuni.cz/corp-lic/cac20-reg-en.html [http://ufal.mff.cuni.cz/corp-lic/cac20-reg-en.html]). To complete the order, the user must fill in the license agreement form.

Some of the distributed tools are covered by the GPL License (GNU Public License). This fact is always explicitly stated in the `README_EN.txt` file of the tool, which is placed in the home directory of the tool on the CAC 2.0 CD-ROM. In these cases the GPL takes precedence over the CAC 2.0 license.

Chapter 8. Project VIPs

All the people who contributed to the CAC 2.0 are introduced by name.

- **Czech Academic Corpus version 2.0**
  - **Morphological annotations checking:** Jiří Mírovský
  - **Syntactical annotations checking**: Alla Bémová, Katarína Gajdošová, Katarína Kandračová, Ivana Klímová, Kiril Ribarov, Zdeňka Urešová, Miroslav Zumrík
- **Tools**
  - **Bonito:** Pavel Rychlý, Oldřich Krůza
  - **LAW:** Jirka Hana
  - **TrEd:** Petr Pajas
  - **Netgraph:** Jiří Mírovský
  - **Segmentation and tokenization of Czech texts:** Jan Hajič, Michal Křen
  - **Czech morphological analyser:** Jan Hajič, Jaroslava Hlaváčová, David Kolovratník, Pavel Květoň
  - **Tagger:** Jan Raab
  - **Parser:** Ryan McDonald, Václav Novák, Kiril Ribarov
  - **Automatic morphological and syntactical processing of Czech texts:** Michal Kebrt
- **Bonus material**
  - **STYX:** Ondřej Kučera
  - **TrEdVoice:** Leoš Přikryl
- **CD-ROM, Web page**
  - **Installation script:** Ondřej Bojar
  - **CD booklet, web page:** Michal Šotkovský
- **CAC Guide**
  - **Technical editor:** Jan Raab
  - **Czech language corrections:** Magda Ševčíková
  - **English translation:** Alena Chrastová
  - **Proofreading:** Sezin Rajandran

Chapter 9. Financial support

The development of the Czech Academic Corpus, version 2.0, has been supported by the following organizations and projects:

- *Grant Agency of Czech Academy of Sciences*, grants no. 1ET101120413, 1ET101120503,
- *Grant Agency of the Charles University*, grant no. 207-10/257559,
- *Ministry of Education, Youth and Sports*, grant no. MSM0021620838,
- *Faculty of Mathematics and Physics of the Charles University in Prague*,
- *Charles University in Prague.*

Chapter 10. Bibliography

[Collins, 2002] Michael Collins: *Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms*. Proceedings of EMNLP´2002, University of Pennsylvania, Philadelphia, USA, 2002.

[Čermák, Blatná, 2005] František Čermák, Renata Blatná: *Jak využívat Český národní korpus*. Nakladatelství Lidové noviny, Praha, 2005.

[Hajič, 2004] Jan Hajič: *Disambiguation of Rich Inflection (Computational Morphology of Czech)*. Karolinum, Praha, 2004.

[Hajič et al., 2004] Jan Hajič, Jarmila Panevová, Eva Buráňová, Alevtina Bémová, Jan Štěpánek, Petr Pajas, Jiří Kárník: *Anotace na analytické rovině. Návod pro anotátory*. Institute of Formal and Applied Linguistics, MFF UK, Prague, Czech Republic, 2004.

[Hana, Zeman, 2005] Jiří Hana, Daniel Zeman, Jan Hajič, Hana Hanová, Barbora Hladká, Emil Jeřábek: *Manual for Morphological Annotation*. TR-2005-27, Institute of Formal and Applied Linguistics, MFF UK, Prague, Czech Republic. 2005.

[Hladká, Králík, 2006] Barbora Hladká, Jan Králík: *Proměny Českého akademického korpusu*. Slovo a slovesnost, 67: pp. 179–194, 2006.

[Jelínek, Bečka, Těšitelová, 1961] Jaroslav Jelínek, Josef Václav Bečka, Marie Těšitelová: *Frekvence slov, slovních druhů a tvarů v českém jazyce (FSSDTČJ)*. SPN, Praha, 1961.

[Kopřivová, Kocek, 2000] Marie Kopřivová, Jan Kocek: *Český národní korpus, úvod a příručka uživatele*. FF UK, Prague, Czech Republic, 2000.

[Kučera, 2006] Ondřej Kučera: *Pražský závislostní korpus jako cvičebnice jazyka českého [Prague Dependency Treebank as an Exercise Book of Czech]*. Master thesis, MFF UK, Prague, Czech Republic, 2006.

[McDonald, Pereira, Ribarov, Hajič, 2005] Ryan McDonald, Fernando Pereira, Kiril Ribarov, Jan Hajič: *Non-projective Dependency Parsing using Spanning Tree Algorithms*. Proceedings of HLT/EMNLP´2005, pp. 523 – 530, Vancouver, Canada, 2005.

[Mikulová a kol., 2006] Marie Mikulová, Alevtina Bémová, Jan Hajič, Eva Hajičová, Jiří Havelka, Veronika Kolářová, Lucie Kučová, Markéta Lopatková, Petr Pajas, Jarmila Panevová, Magda Razímová, Petr Sgall, Jan Štěpánek, Zdeňka Urešová, Kateřina Veselá, Zdeněk Žabokrtský: *Anotace na tektogramatické rovině Pražského závislostního korpusu. Anotátorská příručka*. TR-2005-28, Institute of Formal and Applied Linguistics, MFF UK, Prague, Czech Republic, 2005.

[Mírovský, 2008] Jiří Mírovský: *Netgraph - Making Searching in Treebanks Easy*. Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP 2008), Hyderabad, India, pp. 945 – 950, 2008.

[Müller, Psutka, Šmídl, 2000] Luděk Müller, Josef Psutka, Luboš Šmídl: *Design of Speech Recognition Engine*. TSD 2000, Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin-Heidelberg, pp. 259 – 264, 2000.

[Pajas, Štěpánek, 2005] Petr Pajas, Jan Štěpánek: *A Generic XML-based Format for Structured Linguistic Annotation and its Application to the Prague Dependency Treebank 2.0*. TR-2005-29, Institute of Formal and Applied Linguistics, MFF UK, Prague, Czech Republic. 2005.

[Přikryl, 2007] Leoš Přikryl: *Rozhraní v mluveném jazyce pro korpusové anotační nástroje*. Diplomová práce, MFF UK, Praha, 2007.

[Psutka, Müller, Matoušek, Radová, 2006] Josef Psutka, Luděk Müller, Jindřich Matoušek, Vlasta Radová: *Mluvíme s počítačem česky*. Academia, Praha, 2006.

[Ribarov, 2004] Kiril Ribarov: *Automatic Building of a Dependency Tree – The Rule-Based Approach and Beyond*. Doktorská práce, MFF UK, Praha, 2004.

[Ribarov, Bémová, Hladká, 2006] Kiril Ribarov, Alla Bémová, Barbora Hladká: *When a statistically oriented parser was more efficient than a linguist: A case of treebank conversion*. Prague Bulletin of Mathematical Linguistics 86, pp. 21 – 38, 2006.

[Savický, Hlaváčová, 2002] Petr Savický, Jaroslava Hlaváčová: *Measures of Word Commonness*. Journal of Quantitative Linguistics. Swets & Zeitlinger, Vol. 9, No. 3, pp. 215 – 231. 2002.

[Šmilauer, 1972] Vladimír Šmilauer: *Nauka o českém jazyku*. Praha, 1972.

[Vidová Hladká a kol., 2007] Barbora Vidová Hladká, Jan Hajič, Jiří Hana, Jaroslava Hlaváčová, Jiří Mírovský, Jan Votrubec: *Průvodce Českým akademickým korpusem 1.0*. Karolinum, Praha, 2007.

[Votrubec, 2005] Jan Votrubec: *Volba vhodné sady rysů pro morfologické značkování češtiny [Selecting an Optimal Set of Features for the Morphological Tagging of Czech]*. Master thesis, MFF UK, Prague, Czech Republic, 2005.

Appendix A. Sources of the texts

Table A.1. Administrative documents

File	Written form		File	Transcription
a01w	Vyhláška č. 100		a16s	Zelená vlna
a02w	Hospodaření s domovním bytovým majetkem		a17s	Zprávy o počasí
a03w	Pracovní řád		a18s	Přehled rozhlasových pořadů
a04w	Národní pojištění 12/1977		a19s	Hlášení v metru
a05w	Kolektivní smlouvy – TIBA			
a06w	Materiál – TIBA			
a07w	Zpráva o činnosti Ústavu pro jazyk český			
a08w	Metodické pokyny			
a09w	Zápisy z porad			
a10w	Závazky			
a11w	Zápisy ze schůzí			
a12w	Pokyny SÚRPMO			
a13w	Pracovní návody, pokyny			
a14w	Oběžníky Ústavu pro jazyk český			
a15w	Zpráva o činnosti oddělení matematické lingvistiky			
a20w	Hlášení v obchodním domě			

Table A.2. Documents covering journalism

File	Written form		File	Transcription
n01w	Rudé právo		n53s	Rozhlasové reportáže a rozhovory
n02w	Svět práce		n54s	Televizní komentáře
n03w	Práce		n55s	Zprávy čs. rozhlasu
n04w	Československý rozhlas I.		n56s	Televizní diskuse
n05w	Mladá fronta		n57s	Televizní zprávy a reportáže
n06w	Československý rozhlas II.		n58s	Rozhlasová diskuse
n07w	Večerní Praha		n59s	Televizní zprávy a lekce
n08w	Československý sport		n60s	Televizní diskuse a komentáře
n09w	Svobodné slovo			
n10w	Lidová demokracie			
n11w	Obrana lidu			
n12w	Týdeník aktualit			
n13w	Zemědělské noviny			
n14w	Gramorevue G 73			
n15w	Tribuna			
n16w	Záběr			
n17w	Úder			
n18w	Svoboda			
n19w	Služba lidu			
n20w	Zpravodaj TIBY			
n21w	Nové Hradecko			
n22w	Pochodeň			
n23w	Technický týdeník			
n24w	Horník a energetik			
n25w	Sázavan			
n26w	Čelákovický zpravodaj			
n27w	Nové Klatovsko			
n28w	Pravda			
n29w	Průboj			
n30w	Zpravodaj TIBY			
n31w	Krkonošská pravda			
n32w	Školství a věda			
n33w	Stráž lidu			
n34w	Zbrojovák			
n35w	Nová svoboda			
n36w	Vlasta			
n37w	Mladý svět			
n38w	Naše rodina			
n39w	Ahoj na sobotu			
n40w	Květy			

File	Written form		File	Transcription
n41w	Signál			
n42w	Zahrádkář			
n43w	Film a doba			
n44w	Melodie			
n45w	Stadion			
n46w	Věda a technika mládeži			
n47w	Haló sobota			
n48w	Svět socialismu			
n49w	Zahradnické listy			
n50w	Kino			
n51w	Chovatel			
n52w	Zápisník Z’73			

Table A.3. Documents covering the scientific field

File	Written form		File	Transcription
s01w	Dějiny české hudební kultury		s69s	Divadelní přehlídka
s02w	Motivace lidského chování		s70s	Výklad Zákoníku práce
s03w	Škola – opora socialismu		s71s	Opera o Bratrech Karamazových (prof. dr. Václav Holzknecht)
s04w	Jak rozumíme chemickým vzorcům a rovnicím		s72s	Zpráva o cestě do Belgie (PhDr. Marie Těšitelová, DrSc.)
s05w	Konflikty mezi lidmi		s73s	Obecné otázky jazykové kultury
s06w	Škoda 1000		s74s	Provozní kontrola potrubí
s07w	Pražský vodovod		s75s	Modelování diod
s08w	Nauka o materiálu		s76s	Přenosové parametry
s09w	Tranzistory řízené elektrickým polem		s77s	O počtu koster jednoho grafu
s10w	Pro půvab a eleganci		s78s	Streptokoky
s11w	Tisíciletý vývoj architektury		s79s	Statické zajištění domu U Rytířů
s12w	Polovodičová technika		s80s	Problémy aerodynamiky závodních vozů
s13w	Plazma, čtvrté skupenství hmoty		s81s	Schůze vědecké rady ČSTV
s14w	Nadhodnota a její formy		s82s	Plenární schůze ROH / Pauzy váhání
s15w	Určování efektivnosti za socialismu		s83s	Seminář o houbách
s16w	Stažlivost myokardu		s84s	Česká filharmonie hraje a hovoří (Václav Neumann)
s17w	K biologickým a psychologickým zřetelům výchovy		s85s	Seminář o fotografii
s18w	Poetika		s86s	Působení hromadných sdělovacích prostředků
s19w	Slovo a slovesnost 4/1973		s87s	Ochrany v průmyslových závodech
s20w	Sociologický časopis 3/1973		s88s	Práce se čtenářem
s21w	Teorie a empirie		s89s	Dlouhodobé skladování masa
s22w	Česká literatura		s90s	Personalistika
s23w	Československá informatika		s91s	Archeologické nálezy v Toušeni (Jaroslav Špaček)
s24w	Národopisné aktuality		s92s	Přednáška o geografii
s25w	Vlastivědný sborník moravský		s93s	Úvod do dějin feudalismu
s26w	Český lid		s94s	Filosofie fyziky (RNDr. Jiří Mrázek, CSc.)
s27w	Otázky lexikální statistiky		s95s	O vývoji knihovnictví
s28w	Památková péče 4/1974		s96s	Základní podmínky pro pěstování zeleniny
s29w	Základní a rekreační tělesná výchova 10/1974		s97s	O výchově socialistické inteligence
s30w	Společenské vědy ve škole 2/1974		s98s	Petrologie sedimentů a reziduálních hornin
s31w	Hospodářské právo		s99s	Organizace a řízení vnitřního obchodu
s32w	Sociální jistoty včera a dnes		s00s	Rozbor situace v JZD

File	Written form		File	Transcription
s33w	Arbitrážní praxe			
s34w	Filosofický časopis 5/1974			
s35w	Československá psychologie			
s36w	Společenská struktura a revoluce			
s37w	Humanismus v naší filosofické tradici			
s38w	Společnost – vzdělání – jedinec			
s39w	Rozvoj osobnosti a slovesné umění			
s40w	Ke kritice buržoasních teorií společnosti			
s41w	Spisovný jazyk v současné komunikaci			
s42w	Přirozený jazyk v informačních systémech			
s43w	Česká literatura			
s44w	NA			
s45w	Vědeckotechnická revoluce a socialismus			
s46w	Zesilovače se zpětnou vazbou			
s47w	Teorie a počítače v geofyzice			
s48w	Výzkum hlubinné geologické stavby Československa			
s49w	Podstata hypnózy a spánek			
s50w	Nukleární medicína			
s51w	Hutnictví a strojírenství			
s52w	Záruční lhůty potravinářských výrobků			
s53w	Mineralogie			
s54w	Ptáci			
s55w	Elektronický obzor 6/1974			
s56w	Teplárenství			
s57w	Vědecko-technický rozvoj za socialismu			
s58w	Jak na práce se stavebninami			
s59w	NA			
s60w	Obkládáme interiéry a fasády			
s61w	Alpinkářův svět			
s62w	Opravujeme a modernizujeme rodinný domek			
s63w	Jak na práce s kovem			
s64w	Astronomie			
s65w	Pokroky matematiky, fyziky a astronomie			
s66w	Elektrotechnický obzor			
s67w	Hvězdářská ročenka			
s68w	Lékařská fyzika			

Appendix B. Description of lemmas

In the CAC 2.0, lemma has a form of string *lemma_:P1_;P2_,P3_^(K)* where *lemma* is the lemma proper and *P1, P2, P3, K* stand for the optional additional info; *lemma* has a form of string *LemmaProper-[0-9]** where the optional string "-[0-9]*" helps to distinguish several senses of a homonymous base form.

Table B.1. Additional information of the lemmas

Labelling	Separator	Description	Notes
P1	:	morpho-syntactic flag	part of speech or its detailed specification
P2	;	semantic flag	common semantic clasification
P3	,	style flag	stylistical classification
K	^	comment	explanatory note, derivational comments, other comments

Table B.2. Morpho-syntactic flags of the lemmas

Value	Description
B	abbreviation
T	imperfect verb
W	perfect verb

Table B.3. Semantic flags of the lemmas

Value	Description
E	member of a particular nation, inhabitant of a particular territory
G	geographical name
H	chemistry
K	company, organization, institution
L	natural sciences
R	product
S	surname (family name)
U	medicine
Y	given name
b	economy, finances
c	computers and electronics
g	technology in general
j	justice
m	other proper name
o	color indication
p	politcs, government, military
u	culture, education, arts, other sciences
w	sports
y	hobby, leisure, travelling
z	ecology, environment

Table B.4. Style flags of the lemmas

Value	Description
a	archaic
e	expressive
h	colloquial
l	slang, argot
n	dialect
s	bookish
t	foreign word
v	vulgar
x	outdated spellimg or misspeling

Table B.5. Examples of lemmas

Lemma	Additional info	Description
Abchaz (*Abkhazian*)	_;E	member of a particular nation
Agned	_;Y_,t	given name|foreign word
dobromysl (*oregano*)	_;L	natural sciences
dementi	_,t	foreign word
FFUK (*Faculty of Arts, Charles University*)	_ : B _ ; K _ ; u-^(Filozof._fakulta_Univerzity_Karlovy)	abbreviation|institution|culture, education|abbreviation description
líně (*lazy*)	_^(*1ý)	derivation: remove one character from the end (i.e. "ě"), add character "ý": "líný"

Appendix C. Description of tags

Table C.1. Part of speech

Value	Description
A	Adjective
C	Numeral
D	Adverb
I	Interjection
J	Conjunction
N	Noun
P	Pronoun
V	Verb
R	Preposition
T	Particle
X	Unknown, Not Determined, Unclassifiable
Z	Punctuation (also used for the Sentence Boundary token)

Table C.2. Sub-part of speech

Value	Description	POS
#	Sentence boundary	Z – punctuation
%	Author's signature, e.g. `haš-99_:B_;S`	N – noun
*	Word krát (lit.: "times")	C – numeral
,	Conjunction subordinate (incl. "aby", "kdyby" in all forms)	J – conjuction
}	Numeral, written using Roman numerals (XIV)	C – numeral
:	Punctuation (except for the virtual sentence boundary word ###, which uses the C.2 #)	Z – punctuation
=	Number written using digits	C – numeral
?	Numeral "kolik" (lit. "how many"/"how much")	C – numeral
@	Unrecognized word form	X – unknown
^	Conjunction (connecting main clauses, not subordinate)	J – conjunction
4	Relative/interrogative pronoun with adjectival declension of both types (soft and hard) ("jaký", "který", "čí", ..., lit. "what", "which", "whose", ...)	P – pronoun
5	The pronoun he in forms requested after any preposition (with prefix n-: "něj", "něho", ..., lit. "him" in various cases)	P – pronoun
6	Reflexive pronoun se in long forms ("sebe", "sobě", "sebou", lit. "myself" / "yourself" / "herself" / "himself" in various cases; "se" is personless)	P – pronoun
7	Reflexive pronouns "se" (C.5 = 4), "si" (C.5 = 3), plus the same two forms with contracted -s: "ses", "sis" (distinguished by C.8 = 2; also number is singular only) This should be done somehow more consistently, virtually any word can have this contracted -s ("cos", "polívkus", ...)	P – pronoun
8	Possessive reflexive pronoun "svůj" (lit. "my"/"your"/"her"/"his" when the possessor is the subject of the sentence)	P – pronoun
9	Relative pronoun "jenž", "již", ... after a preposition (n-: "něhož", "niž", ..., lit. "who")	P – pronoun
A	Adjective, general	A – adjective
B	Verb, present or future form	V – verb
C	Adjective, nominal (short, participial) form "rád", "schopen", ...	A – adjective
D	Pronoun, demonstrative ("ten", "onen", ..., lit. "this", "that", "that", ... "over there", ...)	P – pronoun
E	Relative pronoun "což" (corresponding to English which in subordinate clauses referring to a part of the preceding text)	P – pronoun
F	Preposition, part of; never appears isolated, always in a phrase ("nehledě (na)", "vzhledem (k)", ..., lit. "regardless", "because of")	R – preposition
G	Adjective derived from present transgressive form of a verb	A – adjective
H	Personal pronoun, clitical (short) form ("mě", "mi", "ti", "mu", ...); these forms are used in the second position in a clause (lit. "me", "you", "her", "him"), even though some of them ("mě") might be regularly used anywhere as well	P – pronoun
I	Interjections	I – interjection
J	Relative pronoun "jenž", "již", ... not after a preposition (lit. "who", "whom")	P – pronoun
K	Relative/interrogative pronoun "kdo" (lit. "who"), incl. forms with affixes -ž and -s (affixes are distinguished by the category C.15 (for -ž) and C.8 (for -s))	P – pronoun

Value	Description	POS
L	Pronoun, indefinite “všechen”, “sám” (lit. “all”, “alone”)	P – pronoun
M	Adjective derived from verbal past transgressive form	A – adjective
N	Noun (general)	N – noun
O	Pronoun “svůj”, “nesvůj”, “tentam” alone (lit. “own self”, “not-in-mood”, “gone”)	P – pronoun
P	Personal pronoun “já”, “ty”, “on” (lit. “I”, “you”, “he”) (incl. forms with the enclitic -s, e.g. “tys”, lit. “you’re”); gender position is used for third person to distinguish “on”/“ona”/“ono” (lit. “he”/“she”/“it”), and number for all three persons	P – pronoun
Q	Pronoun relative/interrogative “co”, “copak”, “cožpak” (lit. “what”, “isn’t-it-true-that”)	P – pronoun
R	Preposition (general, without vocalization)	R – preposition
S	Pronoun possessive “můj”, “tvůj”, “jeho” (lit. “my”, “your”, “his”); gender position used for third person to distinguish “jeho”, “její”, “jeho” (lit. “his”, “her”, “its”), and number for all three pronouns	P – pronoun
T	Particle	T – particle
U	Adjective possessive (with the masculine ending -ův as well as feminine -in)	A – adjective
V	Preposition (with vocalization -e or -u): (“ve”, “pode”, “ku”, ..., lit. “in”, “under”, “to”)	R – preposition
W	Pronoun negative (“nic”, “nikdo”, “nijaký”, “žádný”, ..., lit. “nothing”, “nobody”, “not-worth-mentioning”, “no”/“none”)	P – pronoun
X	(temporary) Word form recognized, but tag is missing in dictionary due to delays in (asynchronous) dictionary creation	
Y	Pronoun relative/interrogative co as an enclitic (after a preposition) (“oč”, “nač”, “zač”, lit. “about what”, “on”/“onto” “what”, “after”/“for what”)	P – pronoun
Z	Pronoun indefinite (“nějaký”, “některý”, “číkoli”, “cosi”, ..., lit. “some”, “some”, “anybody’s”, “something”)	P – pronoun
a	Numeral, indefinite (“mnoho”, “málo”, “tolik”, “několik”, “kdovíkolik”, ..., lit. “much”/“many”, “little”/“few”, “that much”/“many”, “some” (“number of”), “who-knows-how-much/many”)	C – numeral
b	Adverb (without a possibility to form negation and degrees of comparison, e.g. “pozadu”, “naplocho”, ..., lit. “behind”, “flatly”); i.e. both the C.11 as well as the C.10 attributes in the same tag are marked by – (Not applicable)	D – adverb
c	Conditional (of the verb “být” (lit. “to be”) only) (“by”, “bych”, “bys”, “bychom”, “byste”, lit. “would”)	V – verb
d	Numeral, generic with adjectival declension (“dvojí”, “desaterý”, ..., lit. “two-kinds”/..., “ten-...”)	C – numeral
e	Verb, transgressive present (endings -e/-ě, -íc, -íce)	V – verb
f	Verb, infinitive	V – verb
g	Adverb (forming negation (C.11 set to A/N) and degrees of comparison C.10 set to 1/2/3 (comparative/superlative), e.g. “velký”, “za\-jí\-ma\-vý”, ..., lit. “big”, “interesting”	
h	Numeral, generic: only “jedny” and “nejedny” (lit. “one-kind”/“sort-of”, “not-only-one-kind”/“sort-of”)	C – numeral
i	Verb, imperative form	V – verb
j	Numeral, generic greater than or equal to 4 used as a syntactic noun (“čtvero”, “desatero”, ..., lit. “four-kinds”/“sorts-of”, “ten-...”)	C – numeral

Value	Description	POS
k	Numeral, generic greater than or equal to 4 used as a syntactic adjective, short form (“čtvery”, ..., lit. “four-kinds”/“sorts-of”)	C – numeral
l	Numeral, cardinal “jeden”, “dva”, “tři”, “čtyři”, “půl”, ... (lit. “one”, “two”, “three”, “four”); also “sto” and “tisíc” (lit. “hundred”, “thousand”) if noun declension is not used	C – numeral
m	Verb, past transgressive; also archaic present transgressive of perfective verbs (ex.: “udělav”, lit. “(he-)having-done”; arch. also “udělaje” (C.15 = 4), lit. “(he-)having-done)”	V – verb
n	Numeral, cardinal greater than or equal to 5	C – numeral
o	Numeral, multiplicative indefinite (“-krát”, lit. (“times”): “mnohokrát”, “tolikrát”, ..., lit. “many times”, “that many times”)	C – numeral
p	Verb, past participle, active (including forms with the enclitic - s, lit. ’re (“are”))	V – verb
q	Verb, past participle, active, with the enclitic -ť, lit. (“perhaps”) - “could-you-imagine-that?” or “but-because-” (both archaic)	V – verb
r	Numeral, ordinal (adjective declension without degrees of comparison)	C – numeral
s	Verb, past participle, passive (including forms with the enclitic -s, lit. ’re (“are”))	V – verb
t	Verb, present or future tense, with the enclitic -ť, lit. (“perhaps”) “-could-you-imagine-that?” or “but-because-” (both archaic)	V – verb
u	Numeral, interrogative “kolikrát”, lit. “how many times?”	C – numeral
v	Numeral, multiplicative, definite (-krát, lit. “times”: “pětkrát”, ..., lit. “five times”)	C – numeral
w	Numeral, indefinite, adjectival declension (“nejeden”, “tolikátý”, ..., lit. “not-only-one”, “so-many-times-repeated”)	C – numeral
y	Numeral, fraction ending at -ina; used as a noun (“pětina”, lit. “one-fifth”)	C – numeral
z	Numeral, interrogative “kolikátý”, lit. “what” (“at-what-position-place-in-a-sequence”)	C – numeral

Table C.3. Gender

Value	Description
F	Feminine
H	{F, N} – Feminine or Neuter
I	Masculine inanimate
M	Masculine animate
N	Neuter
Q	Feminine (with singular only) or Neuter (with plural only); used only with participles and nominal forms of adjectives
T	Masculine inanimate or Feminine (plural only); used only with participles and nominal forms of adjectives
X	Any
Y	{M, I} – Masculine (either animate or inanimate)
Z	{M, I, N} – Not fenimine (i.e., Masculine animate/inanimate or Neuter); only for (some) pronoun forms and certain numerals

Table C.4. Number

Value	Description
D	Dual , e.g. “nohama"
P	Plural, e.g. “nohami”
S	Singular, e.g. “noha”
W	Singular for feminine gender, plural with neuter; can only appear in participle or nominal adjective form with gender value Q
X	Any

Table C.5. Case

Value	Description
1	Nominative, e.g. “žena”
2	Genitive, e.g. “ženy”
3	Dative, e.g. “ženě”
4	Accusative, e.g. “ženu”
5	Vocative, e.g. “ženo”
6	Locative, e.g. “ženě”
7	Instrumental, e.g. “ženou”
X	Any

Table C.6. Possessive gender

Value	Description
F	Feminine, e.g. “matčin”, “její”
M	Masculine animate (adjectives only), e.g. “otců”
X	Any
Z	{M, I, N} – Not feminine, e.g. “jeho”

Table C.7. Possessive number

Value	Description
P	Plural, e.g. “náš”
S	Singular, e.g. “můj”
X	Any, e.g. “your”

Table C.8. Person

Value	Description
1	1st person, e.g. “píšu”, “píšeme”
2	2nd person, e.g. “píšeš”, “píšete”
3	3rd person, e.g. “píše”, “píšou”
X	Any person

Table C.9. Tense

Value	Description
F	Future
H	{R, P} – Past or Present
P	Present
R	Past
X	Any

Table C.10. Grade

Value	Description
1	Positive, e.g. “velký”
2	Comparative, e.g. “větší”
3	Superlative, e.g. “největší”

Table C.11. Negation

Value	Description
A	Affirmative (not negated), e.g. “možný”
N	Negated, e.g. “nemožný”

Table C.12. Voice

Value	Description
A	Active, e.g. “píšící”
P	Passive, e.g. “psaný”

Table C.13. Reserve 1

Value	Description
-	not applicable

Table C.14. Reserve 2

Value	Description
-	not applicable

Table C.15. Variant

Value	Description
-	Basic variant, standard contemporary style; also used for standard forms allowed for use in writing by the Czech Standard Orthography Rules despite being marked there as colloquial
1	Variant, second most used (less frequent), still standard
2	Variant, rarely used, bookish, or archaic
3	Very archaic, also archaic + colloquial
4	Very archaic or bookish, but standard at the time
5	Colloquial, but (almost) tolerated even in public
6	Colloquial (standard in spoken Czech)
7	Colloquial (standard in spoken Czech), less frequent variant
8	Abbreviations
9	Special uses, e.g. personal pronouns after prepositions etc.

Appendix D. Analytical function description

Table D.1. Analytical functions (AF) in the CAC 2.0

AF	Description	AF	Description	AF	Description	AF	Description
Pred	predicate, a node not depending on another node; depends on #	Pnom	nominal predicate, or nom. part of predicate with copula be	AuxC	conjunction (subord.)	AuxK	terminal punctuation of a sentence
Sb	subject	AuxV	auxiliary verb be	AuxO	redundant or emotional item, “coreferential” pronoun	ExD	a technical value for a deleted item; also for the main element of a sentence without predicate (externally-dependent)
Obj	object	Coord	coord. node	AuxZ	emphasizing word	AtrAtr	an attribute of any several preceding (syntactic) nouns
Adv	adverbial	Apos	apposition (main node)	AuxX	comma (not serving as a coordinating conjunction)	AtrAdv	structural ambiguity between adverbial and adnominal (hung on a name/noun) dependency without a semantic difference
Atv	complement (so-called determining) technically hung on a non-verbal element	AuxT	reflexive tantum	AuxG	other graphic symbols, not terminal	AdvAtr	dtto with reverse preference
AtvV	complement (so-called determining) hung on a verb, no 2nd gov. node	AuxR	passive reflexive	AuxY	adverbs, particles not classed elsewhere	AtrObj	structural ambiguity between object and adnominal dependency without a semantic difference
Atr	attribute	AuxP	primary preposition, parts of a secondary preposition	AuxS	root of the tree (#)	ObjAtr	dtto with reverse preference

Appendix E. World Wide Web links

	Name (description) **Location**
	PROJECTS
1.	Resources and tools for information systems http://ufal.mff.cuni.cz/rest
2.	Morphological tagging of Czech (a complete guide) http://ufal.mff.cuni.cz/czech-tagging
3.	Parsing of Czech (a complete guide) http://ufal.mff.cuni.cz/czech-parsing
	INSTITUTIONS
4.	Academy of Sciences of the Czech Republic http://www.cas.cz
5.	Grant Agency of the Academy of Sciences of the Czech Republic http://www.gaav.cz
6.	Department of Cybernetics of the University of West Bohemia in Plzen, Czech Republic http://www.kky.zcu.cz
7.	Ministry of Education, Youth and Sports of the Czech Republic http://www.msmt.cz
8.	Charles University in Prague, Czech Republic http://www.cuni.cz
9.	Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics, Charles University in Prague, Czech Republic http://ufal.mff.cuni.cz
10.	Linguistic Data Consortium, Philadelphia, PA, USA http://www.ldc.upenn.edu
11.	Institute of Czech Language, Academy of Sciences of the Czech Republic http://www.ujc.cas.cz
	DATA, RESOURCES, GUIDELINES, TUTORIALS
12.	bTrEd and nTrEd tutorial (tutorial on bTrEd and nTrEd) http://ufal.mff.cuni.cz/pdt2.0/doc/tools/tred/bn-tutorial.html
13.	csts DTD (an internal data format based on SGML) http://ufal.mff.cuni.cz/pdt2.0/doc/pdt-guide/en/html/ch03.html#a-data-formats-csts

	Name (description) **Location**
14.	Czech National Corpus http://ucnk.ff.cuni.cz
15.	Prague Markup Language (an internal data format based on XML) http://ufal.mff.cuni.cz/jazz/pml
16.	Prague Dependency Treebank http://ufal.mff.cuni.cz/pdt
17.	Manual for Morphological Annotation of PDT http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/m-layer/html/index.html
18.	Manual for Analytical Annotation of PDT http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/a-layer/html/index.html
19.	Relax NG (XML scheme) http://www.relaxng.org
20.	SGML http://www.w3.org/MarkUp/SGML/
21.	Slovak National Corpus http://korpus.juls.savba.sk/index.en.html
22.	Tutorial on the a-layer http://lectures.ms.mff.cuni.cz/video/recordshow/index/17/29
23.	Tutorial on the m-layer http://lectures.ms.mff.cuni.cz/video/recordshow/index/17/28
24.	Tutorial on Bonito http://lectures.ms.mff.cuni.cz/video/recordshow/index/2/24
25.	Tutorial on LAW http://lectures.ms.mff.cuni.cz/video/recordshow/index/2/22
26.	Tutorial on Netgraph http://lectures.ms.mff.cuni.cz/video/recordshow/index/2/25
27.	Tutorial on PML format http://lectures.ms.mff.cuni.cz/video/recordshow/index/17/34
28.	Tutorial on the Prague Dependency Treebanks: Prague Treebanking for Everyone http://lectures.ms.mff.cuni.cz/video/categoryshow/index/1
29.	Tutorial on STYX http://lectures.ms.mff.cuni.cz/video/recordshow/index/2/27
30.	Tutorial on TrEd http://lectures.ms.mff.cuni.cz/video/recordshow/index/2/23

	Name (description) **Location**
31.	XML http://www.w3.org/XML
	TOOLS
32.	Bonito (graphical user interface of the Manatee corpus manager) http://nlp.fi.muni.cz/projekty/bonito/
33.	LAW (morphological annotation editor) http://www.ling.ohio-state.edu/~hana/law.html
34.	Morče (morphological tagger of Czech) http://ufal.mff.cuni.cz/morce
35.	Netgraph (tool for searching dependency corpora) http://quest.ms.mff.cuni.cz/netgraph
36.	STYX (electronic exercise book of Czech based on PDT) http://ufal.mff.cuni.cz/styx
37.	TrEd (syntactical annotation editor) http://ufal.mff.cuni.cz/~pajas/tred
38.	TNT (Trigrams'n'Tags tagger) http://www.coli.uni-saarland.de/~thorsten/tnt/