
**The Prague Bulletin of Mathematical Linguistics**
**NUMBER 89 JUNE 2008**

## EDITORIAL BOARD



Published twice a year by Charles University in Prague

Editorial office and subscription inquiries:
ÚFAL MFF UK, Malostranské náměstí 25, 118 00, Prague 1, Czech Republic
E-mail: pbml@ufal.mff.cuni.cz

ISSN 0032-6585

PBML

**The Prague Bulletin of Mathematical Linguistics**

**NUMBER 89 JUNE 2008**

**CONTENTS**

## Articles



## Reviews

PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 89 JUNE 2008 5–22

# Two Languages – One Annotation Scenario? Experience from the Prague Dependency Treebank

Silvie Cinková, Eva Hajičová, Jarmila Panevová, Petr Sgall

**Abstract**

This paper compares the two FGD-based annotation scenarios for Czech and for English, with the Czech as the basis. We discuss the secondary predication expressed by infinitive and its functions in Czech and English, respectively. We give a few examples of English constructions that do not have direct counterparts in Czech (e.g., tough movement and causative constructions with *make*, *get*, and *have*), as well as some phenomena central in English but much less employed in Czech (object raising or control in adjectives as nominal predicates), and, last, structures more or less parallel both in their function and distribution, whose respective annotation differs due to significant differences in the respective linguistic traditions (verbs of perception).

## 1. Introductory Remarks

### 1.1. The current tasks of corpus linguistics

The expansion of the use of computers for linguistic studies based on very large empirical language material led to the appearance of an allegedly new domain, corpus linguistics. One can then ask what the position of corpus linguistics is with regard to computational linguistics. And also what its relation to "real" linguistics is. It is no doubt that the intersection of the two former domains is very large and also that there is no reason to distinguish between corpus and "real" linguistics. There is no descriptive framework universally accepted since there is a diversity of many different trends in linguistics. A discussion on theoretical characterization of linguistic phenomena and the computerized checking of the adequacy of descriptive frameworks belong to fundamental goals in linguistics, and a highly effective collaboration of researchers in all the relevant fields is needed. This implies also the necessity of a systematic, intrinsic collaboration (if not a symbiosis) of corpus oriented and computational linguistics with linguistic theory.


Please cite this article as: Silvie Cinková, Eva Hajičová, Jarmila Panevová, Petr Sgall, Two Languages – One Annotation Scenario? Experience from the Prague Dependency Treebank. The Prague Bulletin of Mathematical Linguistics No. 89, 2008, 5–22.

In our opinion, the following aims of the use of corpora in theoretical linguistic studies can be pointed out:

(i) to offer new conditions for most diverse kinds of research in linguistics itself as well as in neighbouring domains,
(ii) to check existing descriptive frameworks or their parts: for improvements of their consistency, their enrichment or, in the negative case, the abandonment of falsified hypotheses;
(iii) on the basis of aligned corpora to compare descriptions of two or more languages, attempting at a formulation of procedures that would serve as sources for transfer components of translation systems;
(iv) the search for suitable combinations of structural and statistically based procedures of most different kinds and levels, starting from an adequate linguistic background of a POS system with disambiguation.

It is no longer possible to see the centre of all appropriate uses of computers in corpus linguistics in gathering large corpora with searching procedures. A qualified choice between the existing theoretical approaches (or their parts and ingredients) is necessary to make it possible to use corpora effectively for the aims of theoretical linguistics, as well as of frameworks oriented towards pedagogical and other applications.

### 1.2. The objective of the present paper

The present paper is intended as a contribution towards the aim listed as (iii) above. In particular, we want to illustrate how the description of underlying structures carried out in annotating Czech texts (Sect. 2) may be used as a basis for comparison with a more or less parallel description of English. Specific attention is given to several points in which there are differences between the two languages that concern not only their surface or outer form, but (possibly) also their underlying structures, first of all the so-called secondary predication (Sect. 3). In Section 4, we discuss the representations of these constructions in the PDT of Czech as compared with the corresponding annotation in the scenario of a treebank of English (PEDT), being developed in Prague as an English counterpart of PDT (Šindlerová et al., 2007, Bojar et al., 2007).

## 2. Tectogrammatics

In the Functional Generative Description (see Sgall et al., 1986, Hajičová et al., 1998), tectogrammatics is the interface level connecting the system of language (cf. the notions of *langue,* linguistic competence, I-language) with the cognitive layer, which is not directly mirrored by natural languages. Language is understood as a system of oppositions, with the distinction between their prototypical (primary) and peripheral (secondary, marked) members. We assume that the tectogrammatical representations (TRs) of sentences can be captured as dependency based structures the core of which is determined by the valency of the verb and of other parts of speech. Syntactic dependency is handled as a set of relations between head words and their modifications (arguments and adjuncts). However, there are also the relations of coordination

(conjunction, disjunction and other) and of apposition, which we understand as relations of a further dimension. Thus, the TRs are more complex than mere dependency trees.

The TRs also reflect the topic-focus articulation (information structure) of sentences with a scale of communicative dynamism (underlying word order) and the dichotomy of contextually bound (CB) and non-bound (NB) items, which belong primarily to the topic and the focus, respectively. The scale is rendered in the TRs by the left-to-right order of the nodes, although in the surface the most dynamic item, i.e., focus proper, is indicated by a specific (falling) pitch.

In a theoretical description of language, the TRs are seen in a direct relationship to morphemic (surface) structures. This relationship is complicated by many cases of asymmetry – ambiguity, synonymy, irregularities, including the differences between communicative dynamism and surface word order (the latter belonging to the level of morphemics).

The core of a TR is a dependency tree the root of which is the main verb. Its direct dependents are arguments, i.e., Actor, Objective (Patient), Addressee, Origin and Effect, and adjuncts (of location and direction, time, cause, manner, and so on). Actor primarily corresponds to a cognitive (intentional) Agentive, in other cases to an Experiencer (Bearer) of a state or process. If the valency frame of a verb contains only a single participant, then this participant is its Actor, even though (in marked cases) it corresponds to a cognitive item that primarily is expressed by Objective (see (1)).

(1) The book (Actor) appeared.

If the the valency frame of a verb contains just two participants, these are Actor and Objective, which primarily correspond to Agentive and Objective, although the Objective may also express a cognitive item that primarily corresponds to another argument (see (2)).

(2) The chairman (Actor) addressed the audience (Objective).

If the frame contains more than two items, then it is to be distinguished whether the "third" of them is Addressee, Origin, or Effect (cf. the difference between e.g., (3) and (4).

(3) Jim (Actor) gave Mary (Addressee) a book (Objective).
(4) Jim (Actor) changed the firm (Objective) from a small shop (Origin) into a big company (Effect).

In a TR, there are no nodes corresponding to the function words (or to grammatical morphs). Correlates of these items (especially of prepositions and function verbs) are present in the TRs only as indices of node labels: the syntactic functions of the nodes (arguments and adjuncts) are rendered here as functors, and the values of their morphological categories (tense, number, and so on) have the forms of grammatemes. Functors and grammatemes can be understood as indices of lexical items.

In annotating texts from the Czech National Corpus in the frame of the project of the Prague Dependency Treebank (PDT) (Hajič et al., 2006), we work with several specific deviations from theoretically conceived TRs described above. The most important of these deviations is that the tectogrammatical tree structures (TGTSs) we work with in PDT differ from TRs in that they have the form of trees even in cases of coordination; this is made possible by the coordi-

nating conjunctions being handled as specific nodes (with a specific index, here the subscript *coord*, distinguishing between the coordinated items and an item depending on the coordination construction as a whole). Thus, the (primary) TGTS of the sentence (5), with many simplifications, is the tree presented in Figure 1:

(5) Mary and Tom, who are our neighbours, have two children.

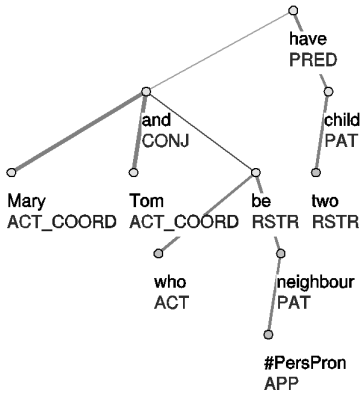



*Figure 1.*

More details are presented in a linearized form of the corresponding TR in (5‘); note that (i) every dependent item (or a string of coordinated items) is embedded in its own pair of parentheses, and the functors are present here as subscripts of the parenthesis oriented towards the head, and (ii) the left-to-right order of the nodes, corresponding to the communicative dynamism, differs from the surface word order of the numeral *two*, which is contextually non-bound and is more dynamic than its head noun. Most of the grammatemes are left out.

(5‘) ((Mary Tom)Conj (Rstr be (Obj neighbour.Plur (App we))))Actor have (Obj child.Plur (Rstr two))

*Rstr* indicates here a restrictive adjunct, *App* one of Appurtenance (broader than possession), the other abbreviations being self-explaining.

Dependency trees are projective; i.e., for every pair of nodes in which *a* is a rightside (left-side) daughter of *b*, every node *c* that is less (more) dynamic than *a* and more (less) dynamic than *b* depends directly or indirectly on *b* (where *indirectly* refers to the transitive closure of *depend*). This strong condition together with similar conditions holding for the relationship between dependency, coordination and apposition, makes it possible to represent the TRs in a linearized way, as illustrated by (5‘) above. Projective trees thus come relatively close to linear strings; they belong to the simplest kinds of patterning.

## 3. Selected English Syntactic Constructions for Comparison

### 3.1. Introduction

A general assumption common to any postulation of a deep (underlying) layer of syntactic description is the belief that languages are closer to each other on that level than in their surface shapes. This idea is very attractive both from the theoretical aspects as well as from the point of view of possible applications in the domain of natural language processing: for example, a level of language description considered to be "common" (at least in some basic features) to several (even if typologically different) languages might serve as a kind of a "pivot" language in which the analysis of the source and the synthesis of the target languages of an automatic translation system may meet (see Vauquois' known "triangle" of analysis – pivot language – synthesis, Vauquois, 1975).

With this idea in mind, it is then interesting (again, both from the theoretical and the applied points of view) to design an annotation scheme by means of which parallel text corpora can be annotated in an identical or at least easily comparable way. It goes without saying, of course, that the question to which extent a certain annotation scenario designed originally for one language is transferrable to annotation of texts of another language is interesting in general, not just for parallel corpora.

It is well known from classical linguistic studies (let us mention here – from the context of English-Czech contrastive studies – the writings of Czech anglicists Vilém Mathesius, Josef Vachek and Libuše Dušková) that one of the main differences between English and Czech concerns the degree of condensation of the sentence structure following from the differences in the repertoire of means of expression in these languages: while in English this system is richer (including also the forms of gerund) and more developed (the English nominal forms may express not only verbal voice but temporal relations as well), in Czech, the more frequent (and sometimes the only possible) means expressing the so called second predication is a dependent clause (see Dušková et al., 1994, p. 542 ff.).

It is no wonder then that in our project, secondary predication has appeared as one of the most troublesome issues. In the present section, we devote our attention to one typical nominal form serving for the expression of secondary predication in English, namely infinitive (Section 3.2), and look for its adequate representation on the tectogrammatical layer of PDT. The leading idea of our analysis is that we should aim at a representation that would make it possible to capture synonymous constructions in a unified way (i.e., to assign to them the same TGTS, both in the same language and across languages) and to appropriately distinguish different meanings by the assignment of different TGTSs.

The considerations included in the present section of our contribution resulted from our work on a project in which the PDT scenario (characterized above in Section 2) was applied to English texts in order to find out if such a task is feasible and if the results may be used for a build-up of a machine translation system (or other multilingual systems); see Šindlerová et al. (2007) and Bojar et al. (2007). This English counterpart of PDT (PEDT) comprises approx. 50,000 dependency trees, which have been obtained by an automatic conversion of

the original Penn Treebank II constituency trees into the PDT-compliant a-layer trees (i.e., trees representing the surface shape of sentences). These a-layer trees have been automatically converted into t-layer trees.

### 3.2. Secondary Predication Expressed by Infinitive

Two classes of constructions are often distinguished: equi-NP deletion and raising. The distinction between the two classes of verbs was already mentioned by Chomsky (1965, pp. 22-23) who illustrated it on the examples (6) and (7):

(6) They expected the doctor to examine John.
(7) They persuaded the doctor to examine John.

Referring to Rosenbaum (1967), Stockwell et al. (1973), p. 521ff., discuss the distinction between *expect* and *require* (which is even clearer than Rosenbaum's distinction between *expect* and *persuade*) and point out that a test involving passivization may help to distinguish the two classes: while (8) and (9) with an equi-verb are synonymous (if their information structure is not considered), (10) and (11) with a raising verb are not:

(8) They expected the doctor to examine John.
(9) They expected John to be examined by the doctor.
(10) They required the doctor to examine John.
(11) They required John to be examined by the doctor.

The authors propose a deep structure indicated by (12) for *expect* (*hate* or *prefer*) and a deep structure that includes an animate object in addition to a sentential object for *require* and *persuade* (see (13)) while it is not important that this NP is then rewritten as S)

(12) They – AUX – VP [V(expect) NP (the doctor examine John)]
(13) They – AUX – VP [V(require) – NP (the doctor) – NP (the doctor examine John)]

Such a treatment of structures with equi verbs implies that there must be a position in the deep structure which is phonologically null (empty category PRO) and which is coreferential with one of the complementations of the equi verb; in our examples above, it is the object in (7). In theoretical linguistics, this issue is referred to as the relation of control (Chomsky, 1981; see also a detailed cross-linguistic study by Růžička, 1999; for Czech, see Panevová, 1986; 1996). More recently, a detailed categorization of the control relation (in a broader sense of the term, i.e. not only with infinitives as objects) has been proposed by Landau (2000); see also the contributions in Davis and Dubinsky, eds. (2007). The following types (not necessarily disjunctive) are distinguished: obligatory, non-obligatory, exhaustive, partial, split, arbitrary, and implicit. The classification is mostly based on the extra-linguistic relation between the controller and the controllee: thus with an arbitrary control in (14) the controller is fully identical with the controllee (the chair both manager and gathers), with a partial control in (15) the controller is a part of the (group of) controllee(s) (the chair is one of those who gather), with a split control in (16) the controller and the controllee form a "joint object" (John and his song together) and

with an arbitrary control in (17) the controlee may be any "object".

(14) The chair managed to gather the committee at 6. (Landau's ex. 8a, p.5)
(15) The chair preferred to gather at 6. (Landau's ex. 9a, p. 5)
(16) John promised his son to go to the movies together (Landau's ex. 11a, p. 31)
(17) It is dangerous for babies to smoke around them. (Landau's ex. 18a, p. 34)

It is a matter of discussion what is the background of such distinctions: they seem to be based on considerations that go beyond grammatical criteria and can be explained on the basis of the lexical meanings of the verbs concerned (if somebody manages to do something s/he also does it, while if somebody prefers to gather (it is understood: with somebody), s/he is part of the gathered group) or on the basis of the linguistic or extra-linguistic context (in (c): John and his son go together) or the preferred reading can be derived from a prototypical situation (babies do not smoke).

The different behaviour of verbs in the structures verb plus infinitive is discussed also in traditional grammars of English. Quirk et al. (2004) observe a certain gradience in the analysis of three superficially identical structures, namely N1 V N2 *to*-V N3 (see their Table 16.64a, p. 1216 reproduced below) illustrated by sentences (18), (19) and (20); in the Table below, these classes belong to the columns 1, 3, and 4, respectively), each of which conforms to this pattern:

(18) We asked the students to attend a lecture.
(19) They expected James to win the race.
(20) We like all parents to visit the school.
(21) James was expected to win the race.

The authors claim that there is a strong reason to see a clear distinction between (18) and (20): in (18) the N2 should be analyzed as the object of the main clause while in (20) they postulate a structure in which N2 functions as the subject of the infinitival clause. However, according to the authors, (19) partakes in both these descriptions: from the semantic point of view, the same analysis as that of (20) would be appropriate; from the structural viewpoint, the analysis similar to that of (18) is preferable. This is supported by the fact that N2 may become the subject of the passive sentence (21). With this analysis, N2 behaves like an object in relation to the verb of the main clause and like a subject in relation to the infinitival clause. The authors use the term raised object to characterize this situation, and they support their analysis by several criteria, which we briefly summarize here as a commentary to their Table 16.64a, p. 1216) reproduced below:

With the structures including the verbs of the class exemplified by ex. 18 above and summarized in the column 1 in the Table below the following criteria apply:

(i) *to*-V N3 can be replaced by a pronoun, an NP or a finite clause (eg. *We asked the students something*),
(ii) *to*-V N3 can be the answer to a wh-question (*What did you ask the students?*),
(iii) when the sequence N2 *to*-V N3 is turned to passive the meaning is always changed: (or it would be even absurd to change *They asked the students to attend the lectures* into *They*

*asked a lecture to be attended by the students*).

(iv) *to*-V N3 can only marginally become the focus of a pseudo-cleft sentence.

With the structures including the verbs of the class exemplified by ex. 20 above and summarized in the column 4 in the Table below the following criteria apply:

(i) the N2 can be replaced by a pronoun referring to the whole clause, e.g. *We like it*;
(ii) the N2 can be an answer to a *what*-question (e.g., *What do you like best?*),
(iii) in some dialects of English the N2 may be preceded by ‚*for*',
(iv) N2 can be the focus of a pseudocleft sentence (e.g., *What we like best is for all patients to visit …*),
(v) when the sentence is turned into the passive form there is no change of meaning: (*We like the school to be visited by all parents*).

The gradience of the analysis of the superficially identical structures N1 V N2 *to*-V N3 is best illustrated by the following Table (reproduced from Quirk et al. 2004, p. 1216)

| **Verb class** | **(1)** | **(2)** | **(3)** | **(4)** |
|---|---|---|---|---|
| **criteria** | ***ask, tell*** | ***elect, allow*** | ***attend, expect*** | ***want,like*** |
| V-inf can be replaced by a finite clause | + | - | - | - |
| change of meaning in passive | + | + | - | - |
| $N_2$ can become subject of passive | + | + | + | - |

The authors emphasize that this is only a rough classification and that it is possible to break these categories further into subcategories between which the differences are small.

To make the picture complete, it should be noted that the relation of control can be postulated also for objects expressed by other nominalised forms, such as the *–ing* participle in *John hates missing the train* and *John hates her missing the train*. The choice between the infinitive and the participle is often guided by extra-linguistic factors: Quirk et al. (2004, Sect. 16.40, p. 1192) mention a mere potentiality expressed by the infinitive (*She hoped to learn English*) vs. a sense of the actual performance of the action itself expressed by the participle (*She enjoyed learning French*), or a difference between an attempt which was not crowned by an achieved act (*Sheila tried to bribe the jailor* = attempted but did not manage it) and a realized attempt without achieving the desired effect (*Sheila tried bribing the jailor* = She actually did bribe the jailor but without (necessarily) achieving what she wanted).

It is interesting to notice that in the two very detailed discussions devoted to nominalizations in English, namely Rosenbaum (1967) and Stockwell, Schachter and Partee (1973), most of the attention is devoted to the derivation of nominalizations while the question of synonymy/non-synonymy of nominalizations with the corresponding finite verbal *that*-clauses is left aside. However, it should be noticed that in their detailed treatment of different aspects of ambiguity (as compared with underspecification, or vagueness), Zwicky and Sadock (1975, esp. pp.16f.) consider the issue of "meaning-changing" transformations and illustrate the com-

plexity of this issue on sentences *We expected that the psychosemanticist would examine George* (his 55) and *We expected the psychosemanticist to examine George* (his 56). The difference between the meaning of the two sentences lies – according to Zwicky and Sadock – in the fact that (55) has two understandings, namely who is the object of our expectations, (i) the psychosemanticist or (ii) George, while (56) has only the understanding (ii). The question is how to account for this distinction. The authors have no definite conclusion: in their opinion, there are two possibilities: either (55) is ambiguous and has two distinctive syntactic structures corresponding to (i) and to (ii), and the raising transformation is applied only to one of them, or (55) has a somewhat 'simpler' syntactic structure (it is underspecified) than (56), and the difference in structure conditions the possibility of raising in (56).

In large contemporary grammars of English the issue of the possibly semantic difference between the nominalization and the that-clause is mentioned rather marginally. E.g., in Quirk et al. (2004), only in the section on the so-called raised object (and in Sect. 16.64) the authors remark that in contrast to the *that*-clause, the infinitival construction is a more formal expression (*The police reported that the traffic was heavy* vs. a formal structure *The police reported the traffic to be heavy*).

## 4. Solutions Proposed

### 4.1. Subject Raising

In the scenario of PEDT (the Prague English Dependency Treebank), the distinction between the structures with the so-called raising verbs and control verbs is preserved. The sentence (22) (see Figure 2) is a typical example for the subject raising construction in English, see also a possibility of (22a) in English:

(22) John seems to understand everything.
(22a) It seems that John understands everything.

However, its Czech counterpart *zdát se* is connected with certain constraints: this verb must be determined by verbo-nominal (or only nominal) complement, see ex. (23). With verbo-nominal complement it has an analogical structure to the English example in Figure 2, see Figure 3. These constraints, however, eliminate this verb from the "pure" raising constructions; see also the unacceptability of (24) in Czech:

(23) Jan se zdá (být) smutný.
Lit. John Refl. he-seems (to-be) sad.
(24) * Jan se zdá rozumět.
Lit. John Refl. he-seems to-understand

In English, the modal and phase verbs are considered as belonging to the class of subject raising verbs. In the PDT scenario (as well as in the theoretical framework for it, FGD) most of these verbs are treated as auxiliaries, and their modal meanings are described by morphological grammatemes assigned to the autosemantic verb. As for modal verbs, this approach is adopted

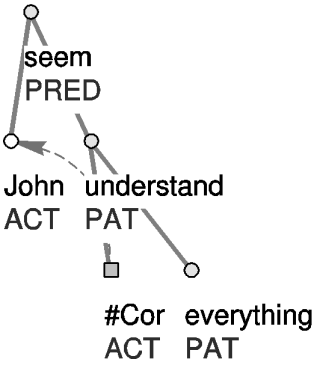



*Figure 2.*

for PEDT as well (see Cinková et al., 2006, p. 88f.). This approach is planned for the treatment of phase verbs, too (*Jan začal pracovat* [John started to work], *Jan začínal pracovat* [John was going to start to work] could be described as multi verbal predicates).

The underlying structure proposed for subject raising constructions in Czech as well as in English is, however, identical to the control verb constructions, where ACT (i.e., the first argument of the control verb) controls Sb (subject) of the infinitive clause (see Section 4.3).

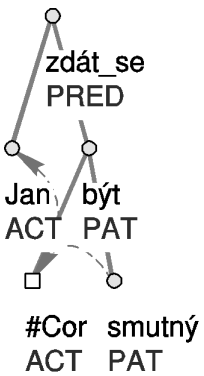



*Figure 3.*

### 4.2. Object Raising

The English verbs used as clear examples of object raising verbs have no Czech counterparts with infinitive constructions; cf. (25) and Figure 4 for English:

(25) John expects Mary to leave.

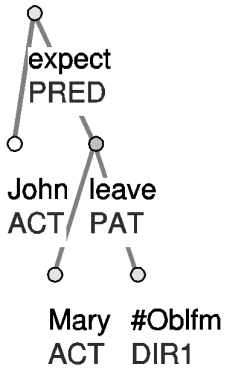



*Figure 4.*

However, the subclass of verbs displaying this operation, called sometimes ECM (exceptional case marking), share this behaviour with Czech constructions of *accusativus cum infinitivo* (AccI in sequel). It concerns the verbs of perception (see (26a) and Figure 5 for English and (26b) and Figure 6 for Czech):

(26a) John hears Mary cry/crying.
(26b) Jan slyší Marii plakat.

There are two possible ways to reflect the underlying structures of these sentences:
The approach (A) is influenced by the English tradition: The verbs of perception proper (such as *to see, to hear*) are understood in English as two-argument structures; if their second argument is expressed by secondary predication, the first argument of the secondary predication is raised up and it receives ("exceptionally") the Accusative form. The structure given in Figure 5 would yield the surface structure (26a) as well as the surface structure (26c):

(26c) John hears that Mary cries.
(26d) Jan slyší, že Marie pláče.

However, the synonymy illustrated by (26a) and (26c) does not hold in all contexts, see (27a), (27b), (27c) and (27d), and also (28a) and (28b):

(27a) Jan slyšel, že Carmen zpívá Dagmar Pecková.

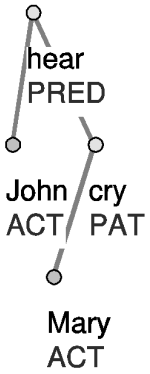



*Figure 5.*

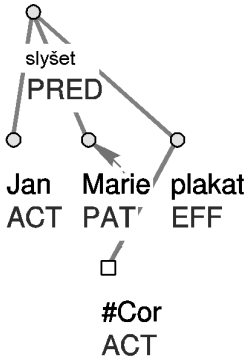



*Figure 6.*

Lit. Jan heard that Carmen-Acc sings Dagmar Pecková
(27b) Jan slyšel, že Dagmar Pecková zpívá Carmen.
Lit. Jan heard that Dagmar Pecková sings Carmen
(27c) Jan slyšel Dagmar Peckovou zpívat Carmen.
Lit. Jan heard Dagmar Pecková to-sing Carmen
(27d) ?Jan slyšel Carmen zpívat Dagmar Peckovou.
Lit. Jan heard Carmen-Acc to-sing Dagmar Peckova-Acc
(28a) Jan slyšel tu skladbu hrát kapelu Olympic.
Lit. Jan heard the piece-Acc to-play the band Olympic-Acc
(28b) Jan slyšel, že/jak tu skladbu hraje kapela Olympic.

Lit. Jan heard that/how the piece-Acc plays the band Olympic-Nom

In the pairs (27a), (27b) vs. (27c), (27d) the difference between the meanings of the polysemic verb *slyšet* [to hear] is reflected: while in (27a) and (27b) Jan is either the direct hearer of the singing or he may be only told about the singing, in (27c) and (27d), if it is possible at all, he must be a direct listener. Moreover, the possible pre-posing of the object of the dependent clause (see (27a) and (28a) for Czech) has no counterpart in English.

In the approach (B) reflecting the situation in Czech the verbs of perception are understood as three-argument structures with the underlying structure given in Figure 6 corresponding to the sentence (26d), which differs from the underlying structure of ex. (26c) given in Figure 5.

Under the approach (A), the formulation of the conditions under which the secondary predication could be nominalised by an infinitive clause seems to be very complicated while with the approach (B) the raised object is understood as a part of a cognitive operation, the result of which is manifested on the level of underlying structure.

### 4.3. Control (Equi) Verbs

As for the control verbs, the underlying structure proposed for Czech seems to be suitable for the PEDT scenario as well, see (29), (30) and Figure 7, 8. A special node with lemma *Cor* is used for the controllee and an arrow leads from this node to its controller. The list of the verbs sharing the attribute of control will be nearly identical for both languages.

(29) John refused to cooperate.
(30) The editor recommended the author to correct the errors immediately.

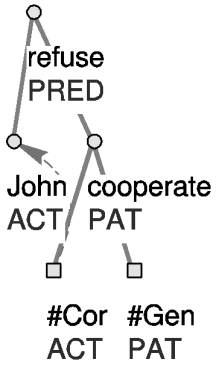



*Figure 7.*

We have concluded that though the notions of raising and control are assumed not to be theory dependent and therefore applicable in both scenarios (for PDT as well as for PEDT), the

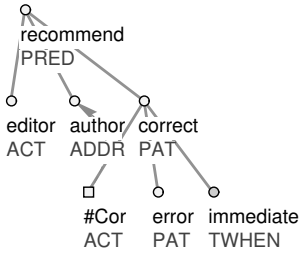



*Figure 8.*

differences between these two classes are not substantial (and they seem to be overestimated in the theoretical works).

### 4.4. Nominal Predicates

Analogical control constructions appear with some adjectives in the position of the nominal predicates in sentences with copula, see (31), (32) and Figure 9 for English:

(31) John is eager to please.
(32) John is eager to be pleased.

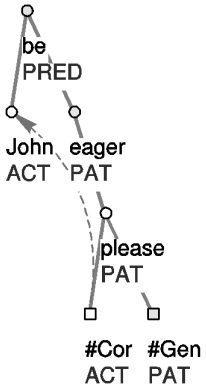



*Figure 9.*

The corresponding underlying structures for Czech sentences (33a), (34a) are similar to those for English (33b), (34b):

(33a) Jan je schopen to udělat.
(33b) John is able to do it.
(34a) Jan je ochoten být očkován.
(34b) John is willing to be vaccinated.

However, the list of English adjectives complemented by an infinitive clause is wider than in Czech. In (35), (36) and Figure 10 a control between ACT and the Sb of infinitive clause could be seen:

(35) She was quick to shut the door.
(36) Bob was reluctant to respond.

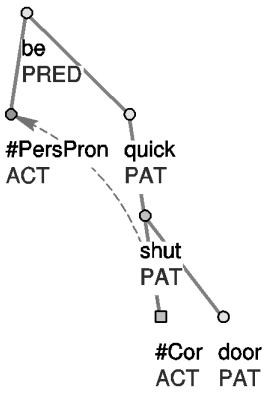



*Figure 10.*

### 4.5. Tough Movement

The object-to-subject raising (sometimes called tough movement) takes place with some evaluative adjectives in complex predicates, see (37a) and its transformed version after the raising operation (37b, Figure 11):

(37a) It is difficult to please John.
(37b) John is difficult to please.

This type of raising has no counterpart in Czech.

### 4.6. Causative Constructions

Causativity of constructions such as (38) (see Figure 12) and (39) is expressed by the lexical meanings of the "semiauxiliaries" *to make, to get, to have* and by the secondary predication

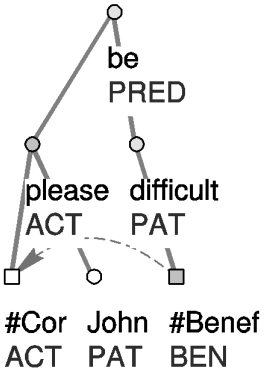



*Figure 11.*

denoting the caused event filling the position of the PAT(ient) of the semiauxiliary causative verb.

(38) John made Mary stay.

(39) John had Mary clean the window.

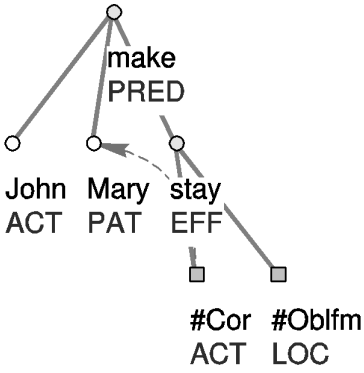



*Figure 12.*

The constructions with the Czech verb *nechat* [to let] and the analogical underlying structure (with raised subject-to-object position) correspond to this type of causativity.

## 5. Conclusions

In our contribution, we have briefly discussed certain issues of secondary predication in which English differs from Czech with the result that most of them probably can be handled without differences in underlying structures of the two languages.

There are, of course, other cases in which the TRs of the two languages certainly differ. We want only to note here that not all such differences concern syntactic relations (functors). Thus in the case of such grammatical categories as definiteness or as tense and verbal aspect the differences can be captured by distinctions in the repertoires and values of grammatemes (representing morphological values).

**Note** The present paper is an enlarged and modified version of the contribution by the same authors entitled *The Tectogrammatics of English: On Some Problematic Issues from the Viewpoint of the Prague Dependency Treebank* and submitted for publication in the Festschrift to honour Professor Anna Sågvall-Hein.

**Acknowledgement** This work was funded by GA-CR 405/06/0589, MSM 0021620838, and in part by the Companions project (`www.companions-project.org`) sponsored by the European Commission as part of the Information Society Technologies (IST) programme under EC grant number IST-FP6-034434.

## References

Bojar Ondřej, Cinková Silvie and Ptáček Jan (2007), *Towards English-to-Czech MT via Tectogrammatical Layer.* In NEALT Proceedings Series: Proceedings of the 6th International Workshop on Treebanks and Linguistic Theories (TLT 2007). 1. Bergen, Norway: North European Association for Language Technology, pp. 7–18.

Cinková Silvie, Hajič Jan, Mikulová Marie, Mladová Lucie, Nedolužko Anja, Pajas Petr, Semecký Jiří, Šindlerová Jana, Toman Josef, Urešová Zdeňka and Žabokrtský Zdeněk (2006), *Annotation of English on the Tectogrammatical Level.* Technical report UFAL TR 2006-35. Prague.

Čmejrek Martin, Cuřín Jan, Havelka Jiří, Hajič Jan and Kuboň Vladimír (2005), *Prague Czech-English Dependency Treebank Version 1.0.* In EAMT 2005 Conference Proceedings, p. 73–78.

Chomsky Noam (1981), *Lectures on Government and Binding.* Dordrecht: Foris.

Chomsky Noam (1965), *Aspects of the Theory of Syntax.* The MIT Press.

Davies W. D., Dubinsky S. (eds.) (2007): *New Horizons in the Analysis of Control and Raising.* Springer: Dordrecht, The Netherlands.

Dušková Libuše et al. (1994), *Mluvnice současné angličtiny na pozadí češtiny* [Grammar of Present-Day English on the Background of Czech], Academia, Prague.

Hajič Jan, Panevová Jarmila, Hajičová Eva, Sgall Petr, Pajas Petr, Štěpánek Jan,Havelka Jiří, Mikulová Marie, Žabokrtský Zdeněk and Ševčíková-Razímová Magda (2006). *Prague De-*

*pendency Treebank 2.0.* CD-ROM. Linguistic Data Consortium, Philadelphia, PA, USA. LDC Catalog No. LDC2006T01 URL<http://ufal.mff.cuni.cz/pdt2.0/>, quoted 2008-12-02.

Hajičová Eva, Partee Barbara H. and Sgall Petr (1998), *Topic-Focus Articulation, Tripartite Structures and Semantic Content.* Dordrecht: Kluwer.

Landau I. (2000), *Elements of Control.* Kluwer: Dordrecht, The Netherlands.

Mikulová Marie, Bémová Allevtina, Hajič Jan, Hajičová Eva, Havelka Jiří, Kolářová Veronika, Kučová Lucie, Lopatková Markéta, Pajas Petr, Panevová Jarmila, Razímová Magda, Sgall Petr, Štěpánek Jan, Urešová Zdeňka, Veselá Kateřina and Žabokrtský Zdeněk (2006), *Annotation on the tectogrammatical level in the Prague Dependency Treebank. Annotation manual.* Tech. Report 30 ÚFAL MFF UK. Prague.

Panevová Jarmila (1996). *More Remarks on Control.* In: Prague Linguistic Circle Papers, Vol.2, (ed. E.Hajičová, O.Leška, P.Sgall, Z.Skoumalová), J.Benjamins Publ. House, Amsterdam – Philadelphia, 1996, 101–120.

Panevová Jarmila (1986), *The Czech Infinitive in the Function of Objective and the Rules of Coreference.* In: J. L. Mey, ed. Language and Discourse: Test and Protest. Amsterdam: Benjamins, 123–142.

Quirk Randolph, Greenbaum Sydney, Leech Geoffrey and Svartvik Jan (2004), *A Comprehensive Grammar of the English Language.* Longman. First published 1985.

Rosenbaum Peter S.(1967), *The Grammar of English Predicate Complement Constructions.* The MIT Press, Cambridge, Mass.

Růžička Rudolf (1999), *Control in Grammar and Pragmatics.* Amsterdam/Philadeplhia: Benjamins.

Sgall Petr, Hajičová Eva and Panevová Jarmila (1986), *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects.* Dordrecht: Reidel Publishing Company and Prague: Academia.

Stockwell Robert P., Schachter Paul and Partee Barbara Hall (1973)., *The Major Syntactic Structures of English.* Holt, Winehart and Winston, New York.

Šindlerová Jana, Mladová Lucie, Toman Josef and Cinková Silvie (2007), *An application of the PDT scheme to a parallel treebank.* Proceedings of the conference Treebanks and Linguistic Theory 2007, Bergen, pp. 163–174.

Vauquois Bernard (1975), *Some problems of optimization in multilingual automatic translation.* In First National Conference on the Application of Mathematical Models and Computers in Linguistics. Varna, May 1975.

Zwicky Arnold, Sadock Jerrold M. (1975), *Ambiguity tests and how to fail them.* In: John P. Kimball, ed. Syntax and Semantics 4, Academic Press, New York, San Francisco, London, 1–36

PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 89 JUNE 2008 23–40

# Combining Statistical and Rule-Based Approaches to Morphological Tagging of Czech Texts

Drahomíra "johanka" Spoustová

**Abstract**

This article is an extract of the PhD thesis (Spoustová, 2007) and it extends the article (Spoustová et al., 2007). Several hybrid disambiguation methods are described which combine the strength of hand-written disambiguation rules and statistical taggers. Three different statistical taggers (HMM, Maximum-Entropy and Averaged Perceptron) and a large set of hand-written rules are used in a tagging experiment using Prague Dependency Treebank. The results of the hybrid system are better than any other method tried for Czech tagging so far.

## 1. Introduction

Inflective languages pose a specific problem for tagging due to two phenomena: highly inflective nature (causing sparse data problem in any statistically based system), and free word order (causing fixed-context systems, such as n-gram HMMs, to be even less adequate than for English).

The average tagset contains about 1,000–2,000 distinct tags; the size of the set of possible and plausible tags can reach several thousands. There have been attempts at solving this problem for some of the highly inflective European languages, such as (Daelemans, 1996), (Erjavec, 1999) for Slovenian and (Hajič, 2000) for five Central and Eastern European languages.

Several taggers already exist for Czech, e.g. (Hajič et al., 2001b), (Smith, 2005), (Hajič et al., 2006) and (Votrubec, 2006). The last one reaches the best accuracy for Czech so far (95.12%). Hence no system has reached – in the absolute terms – a performance comparable to English tagging (such as (Ratnaparkhi, 1996)), which stands above 97%.

We are using the Prague Dependency Treebank (Hajič et al., 2006) (PDT) with about 1.8 million hand annotated tokens of Czech for training and testing. The tagging experiments in this paper all use the Czech morphological (pre)processor, which includes a guesser for "unknown" tokens and which is available from the PDT website (PDT Guide, 2006), to disam-


Please cite this article as: Drahomíra "johanka" Spoustová, Combining Statistical and Rule-Based Approaches to Morphological Tagging of Czech Texts. The Prague Bulletin of Mathematical Linguistics No. 89, 2008, 23–40.

| | Name | Description |
|---|---|---|
| 1 | POS | Part of Speech |
| 2 | SUBPOS | Detailed POS |
| 3 | GENDER | Gender |
| 4 | NUMBER | Number |
| 5 | CASE | Case |
| 6 | POSSGENDER | Possessor's Gender |
| 7 | POSSNUMBER | Possessor's Number |
| 8 | PERSON | Person |
| 9 | TENSE | Tense |
| 10 | GRADE | Degree of comparison |
| 11 | NEGATION | Negation |
| 12 | VOICE | Voice |
| 13 | RESERVE1 | Unused |
| 14 | RESERVE2 | Unused |
| 15 | VAR | Variant |

*Table 1. Czech Morphology and the Positional Tags*

biguate only among those tags which are morphologically plausible.

The meaning of the Czech tags (each tag has 15 positions) we are using is explained in Table 1. A detailed linguistic description of the individual positions can be found in the documentation for the PDT (Hajič et al., 2006).

## 2. Components of the hybrid system

### 2.1. The HMM tagger

The HMM tagger is based on the well known formula of HMM tagging:

$$\hat{T} = \arg\max_{T} P(T)P(W \mid T) \tag{1}$$

where

$$\begin{aligned} P(W|T) &\approx \textstyle\prod_{i=1}^{n} P(w_i \mid t_i, t_{i-1}) \\ P(T) &\approx \textstyle\prod_{i=1}^{n} P(t_i \mid t_{i-1}, t_{i-2}). \end{aligned} \tag{2}$$

The trigram probability $P(W \mid T)$ in formula 2 replaces (Hajič et al., 2001b) the common (and less accurate) bigram approach. We will use this tagger as a baseline system for further improvements.

Initially, we change the formula 1 by introducing a scaling mechanism[1]: $\hat{T} = \arg\max_T(\lambda_T * logP(T) + logP(W \mid T))$.

[1] The optimum value of the scaling parameter $\lambda_T$ can be tuned using held-out data.

We tag the word sequence from right to left, i.e. we change the trigram probability $P(W \mid T)$ from formula 2 to $P(w_i \mid t_i, t_{i+1})$.

Both the output probability $P(w_i \mid t_i, t_{i+1})$ and the transition probability $P(T)$ suffer a lot due to the data sparseness problem. We introduce a component $P(ending_i \mid t_i, t_{i+1})$, where $ending$ consists of the last three characters of $w_i$. Also, we introduce another component $P(t_i^* \mid t_{i+1}^*, t_{i+2}^*)$ based on a reduced tagset $T^*$ that contains positions POS, GENDER, NUMBER and CASE only (chosen on linguistic grounds).

We upgrade all trigrams to fourgrams; the smoothing mechanism for fourgrams is history-based bucketing (Krbec, 2005).

The final fine-tuned HMM tagger thus uses all the enhancements and every component contains its scaling factor which has been computed using held-out data. The total error rate reduction is 13.98% relative on development data, measured against the baseline HMM tagger.

### 2.2. Morče

The Morče[2] tagger assumes some of the HMM properties at runtime, namely those that allow the Viterbi algorithm to be used to find the best tag sequence for a given text. However, the transition weights are not probabilities. They are estimated by an Averaged Perceptron described in (Collins, 2002). Averaged Perceptron works with features which describe the current tag and its context.

Features can be derived from any information we already have about the text. Every feature can be true or false in a given context, so we can regard current true features as a description of the current tag context.

For every feature, the Averaged Perceptron stores its weight coefficient, which is typically an integer number. The whole task of Averaged Perceptron is to sum all the coefficients of true features in a given context. The result is passed to the Viterbi algorithm as a transition weight for a given tag. Mathematically, we can rewrite it as:

$$w(C,T) = \sum_{i=1}^{n} \alpha_i.\phi_i(C,T) \tag{3}$$

where $w(C,T)$ is the transition weight for tag $T$ in context $C$, $n$ is number of features, $\alpha_i$ is the weight coefficient of $i^{th}$ feature and $\phi(C,T)_i$ is evaluation of $i^{th}$ feature for context $C$ and tag $T$.

Weight coefficients ($\alpha$) are estimated on training data, cf. (Votrubec, 2006). The training algorithm is very simple, therefore it can be quickly retrained and it gives a possibility to test many different sets of features (Votrubec, 2005). As a result, Morče gives the best accuracy from the standalone taggers.

[2] The name Morče stands for "MORfologie ČEštiny" ("morphology of Czech").

### 2.3. The Feature-Based Tagger

The Feature-based tagger, taken also from the PDT (Hajič et al., 2006) distribution used in our experiments uses a general log-linear model in its basic formulation:

$$p_{AC}(y \mid x) = \frac{\exp(\sum_{i=1}^{n} \lambda_i f_i(y, x))}{Z(x)} \tag{4}$$

where $f_i(y, x)$ is a binary-valued feature of the event value being predicted and its context, $\lambda_i$ is a weight of the feature $f_i$, and the $Z(x)$ is the natural normalization factor.

The weights $\lambda_i$ are approximated by Maximum Likelihood (using the feature counts relative to all feature contexts found), reducing the model essentially to Naive Bayes. The approximation is necessary due to the millions of the possible features which make the usual entropy maximization infeasible. The model makes heavy use of single-category Ambiguity Classes (AC)[3], which (being independent on the tagger's intermediate decisions) can be included in both left and right contexts of the features.

### 2.4. The rule-based component

The approach to tagging (understood as a stand-alone task) using hand-written disambiguation rules has been proposed and implemented for the first time in the form of Constraint-Based Grammars (Karlsson, 1995). On a larger scale, this aproach was applied to English (Karlsson, 1995) and (Samuelsson, 1997), and French (Chanod, 1995). Also (Bick, 2000) uses manually written disambiguation rules for tagging Brazilian Portuguese, (Karlsson, 1985) and (Koskenniemi, 1990) for Finish and (Oflazer, 1997) reports the same for Turkish.

#### 2.4.1. Overview

In the hybrid tagging system presented in this paper, the rule-based component is used to further reduce the ambiguity (the number of tags) of tokens in an input sentence, as output by the morphological processor (see Sect. 1). The core of the component is a hand-written *grammar* (set of rules).

Each rule represents a piece of knowledge of the language system (in particular, of Czech). The knowledge encoded in each rule is formally defined in two parts: a sequence of tokens that is searched for in the input sentence and the tags that can be deleted if the sequence of tokens is found.

The overall strategy of this "negative" grammar is to keep the highest recall possible (i.e. 100%) and to gradually improve precision. In other words, whenever a rule deletes a tag, it is (almost) 100% safe that the deleted tag is "incorrect" in the sentence, i.e. the tag cannot be present in any correct tagging of the sentence.

Such an (virtually) "error-free" grammar can partially disambiguate any input and prevent the subsequent taggers (stochastic, in our case) to assign tags that are "safely incorrect".

[3] If a token can be a `N`(oun), `V`(erb) or `A`(djective), its (major POS) Ambiguity Class is the value "`ANV`".

### 2.4.2. The rules

Formally, each rule consists of the description of the *context* (sequence of tokens with some special property), and the *action* to be performed given the context (which tags are to be discarded). The length of context is not limited by any constant; however, for practical purposes, the context cannot cross over sentence boundaries.

For example: in Czech, two finite verbs cannot appear within one clause. This fact can be used to define the following disambiguation rule:

- context: unambiguous finite verb, followed/preceded by a sequence of tokens containing neither a comma nor a coordinating conjunction, at either side of a word $x$ ambiguous between a finite verb and another reading;
- action: delete the finite verb reading(s) at the word $x$.

It is obvious that no rule can contain knowledge of the whole language system. In particular, each rule is focused on at most a few special phenomena of the language. But whenever a rule deletes a tag from a sentence, the information about the sentence structure “increases”. This can help other rules to be applied and to delete more and more tags.

For example, let's have an input sentence with two finite verbs within one clause, both of them ambiguous with some other (non-finite-verbal) tags. In this situation, the sample rule above cannot be applied. On the other hand, if some other rule exists in the grammar that can delete non-finite-verbal tags from one of the tokens, then the way for application of the sample rule is opened.

The rules operate in a loop in which (theoretically) all rules are applied again whenever a rule deletes a tag in the partially disambiguated sentence. Since deletion is a monotonic operation, the algorithm is guaranteed to terminate; effective implementation has also been found in (Květoň, 2006).

### 2.4.3. Grammar used in tests

The grammar is being developed since 2000 as a standalone module that performs Czech morphological disambiguation. There are two ways of rule development:

- the rules developed by syntactic introspection: such rules are subsequently verified on the corpus material, then implemented and the implemented rules are tested on a testing corpus;
- the rules are derived from the corpus by introspection and subsequently implemented.

In particular, the rules are not based on examination of errors of stochastic taggers.

The set of rules is (manually) divided into two (disjoint) reliability classes — *safe* rules (100% reliable rules) and *heuristics* (highly reliable rules, but obscure exceptions can be found). The safe rules reflect general syntactic regularities of Czech; for instance, no word form in the nominative case can follow an unambiguous preposition. The less reliable heuristic rules can be exemplified by those accounting for some special intricate relations of grammatical agreement in Czech.

The grammar consists of 1,727 safe rules and 504 heuristic rules. The system has been used in two ways:

- *safe rules only*: in this mode, safe rules are executed in the loop until some tags are being deleted. The system terminates as soon as no rule can delete any tag.
- *all rules*: safe rules are executed first (see *safe rules only* mode). Then heuristic rules start to operate in the loop (similarly to the safe rules). Any time a heuristic rule deletes a tag, the *safe rules only* mode is entered as a sub-procedure. When safe rules' execution terminates, the loop of heuristic rules continues. The disambiguation is finished when no heuristic rule can delete any tag.

The rules are written in the *fast LanGR* formalism (Květoň, 2006) which is a subset of a more general LanGR formalism (Květoň, 2005). The LanGR formalism has been developed specially for writing and implementing disambiguation rules.

## 3. Methods of combination

The motivation for the combination experiments is following: if we have several different methods solving the same problem with similar error rate, it is probable that they do not make exactly the same mistakes. If we identify the strong and weak aspects of each method and find the optimal way to combine them, the resulting method's performance should be better than the performance of all of its components.

In our experiments we use the components described above – three statistical taggers (Feature-based – „a", HMM – „b", Morče – „m") and two sets of hand-written rules („safe", safe + heuristics – „all"). Most of the ideas for the experiments were original, except the serial combination rules – tagger, which was already published in (Hajič et al., 2001b) and we only performed the same experiment with new versions of the components.

All the methods presented in this paper have been trained and tested on the PDT version 2.0[4]. Taggers were trained on PDT 2.0 training data set (1,539,241 tokens), the results were achieved on PDT 2.0 development-test data set (201,651 tokens), and for the best methods also the PDT 2.0 evaluation-test data set (219,765 tokens) was used. The morphological analysis processor and all the taggers were used in versions from April 2006 (Hajič et al., 2006), the rule-based component is from September 2006.

For evaluation, we use both precision and recall (and the corresponding F-measure) and accuracy, since we also want to evaluate the partial disambiguation achieved by the hand-written rules alone. Let $t$ denote the number of tokens in the test data, let $c$ denote the number of tags assigned to all tokens by a disambiguation process and let $h$ denote the number of tokens where the manually assigned tag is present in the output of the process.

- In case of the morphological analysis processor and the standalone rule-based component, the output can contain more than one tag for every token. Then *precision* ($p$), *recall*

[4]The results cannot be simply (number-to-number) compared to previous results on Czech tagging, because different training and testing data (PDT 2.0 instead of PDT 1.0) are used since 2006.

| Tagger | accuracy |
|---|---|
| Feature-based (a) | 94.27% |
| HMM (b) | 95.13% |
| Morče (m) | **95.43%** |

*Table 2. Evaluation of the taggers alone*

| | precision | recall | f-measure |
|---|---|---|---|
| morphology | 25.72% | 99.40% | 40.87% |
| safe rules | 58.76% | 98.90% | 73.72% |
| all rules | 67.36% | 98.24% | **79.92%** |

*Table 3. Evaluation of the rules alone*

($r$) and *F-measure* ($f$) characteristics are defined as follows:

$$p = h/c \qquad r = h/t \qquad f = 2pr/(p+r).$$

- The output of the stochastic taggers contains always exactly one tag for every token — then $p = r = f = h/t$ holds and this ratio is denoted as *accuracy*.

The initial performance of the components is presented in table Table 2 and Table 3

### 3.1. Serial combination rules – tagger

The simplest way of combining a hand-written disambiguation grammar with a stochastic tagger is to let the grammar reduce the ambiguity of the tagger's input. Formally, an input text is processed as follows:

1. morphological analysis (every input token gets all tags that are plausible without looking at context);
2. rule-based component (partially disambiguates the input, i.e. deletes some tags);
3. the stochastic tagger (gets partially disambiguated text on its input).

This algorithm was already used in (Hajič et al., 2001b), only components were changed — the ruled-based component was significantly improved and two different sets of rules were tried, as well as three different statistical taggers. The results (compared to the results of the standalone taggers) are presented in Table 4.

The best result was (not surprisingly) achieved with the set of safe rules followed by the Morče tagger.

An identical approach was used in (Tapanainen, 1994) for English.

| | – | safe rules | all rules |
|---|---|---|---|
| tagger a | 94.27% | 92.51% | 92.55% |
| tagger b | 95.13% | 95.48% | 95.30% |
| tagger m | 95.43% | **95.64%** | 95.44% |

*Table 4. Evaluation of the serial combination rules – tagger*

| Tagger | accuracy |
|---|---|
| tagger a | 99.31% |
| tagger b | 99.22% |
| tagger m | **99.25%** |

*Table 5. Accuracy of the taggers in SUBPOS disambiguation*

### 3.2. Serial combination with SUBPOS pre-processing

Manual inspection of the output of the application of the hand-written rules on the development data (as used in the serial combination described in the previous section) discovered that certain types of deadlocked ("cross-dependent") rules prevent successful disambiguation.

Cross-dependence means that a rule $A$ cannot apply because of some remaining ambiguity, which could be resolved by a rule $B$, but the operation of $B$ is still dependent on the application of $A$. In particular, ambiguity in the Part-of-Speech category is very problematic. For example, only a few safe rules can apply to a three-word sentence where all three words are ambiguous between finite verbs and something else.

If the Part-of-Speech ambiguity of the input is already resolved, precision of the rule-based component and also of the final result after applying any of the statistical taggers improves. Full Part-of-Speech information is represented by the first two categories of the Czech morphology tagset — POS and SUBPOS, which deals with different types of pronouns, adverbs etc. As POS is uniquely determined by SUBPOS (Hajič et al., 2006), it is sufficient to resolve the SUBPOS ambiguity only.

All three taggers achieve more than 99% accuracy in SUBPOS disambiguation (see Table 5). For SUBPOS disambiguation, we use the taggers in usual way (i.e. they determine the whole tag) and then we put back all tags having the same SUBPOS as the tag chosen by the tagger.

Thus, the method with SUBPOS pre-processing operates in four steps:

1. (morphological analysis)
2. SUBPOS disambiguation (any tagger)
3. rule-based component
4. final disambiguation (any tagger)

Results after performing the first, the second and the third step are presented in Tables 6, 7,

| | precision | recall | f-measure |
|---|---|---|---|
| tagger a | 30.05% | 98.92% | 46.10% |
| tagger b | 30.10% | 98.83% | 46.15% |
| tagger m | 30.10% | 98.87% | 46.15% |

*Table 6. Combination with SUBPOS pre-processing: results of the first step*

| | precision | recall | f-measure |
|---|---|---|---|
| tagger a + safe rules | 64.81% | 98.68% | 78.24% |
| tagger a + all rules | 70.53% | 98.36% | 82.15% |
| tagger b + safe rules | 65.07% | 98.59% | 78.40% |
| tagger b + all rules | 70.81% | 98.27% | 82.31% |
| tagger m + safe rules | 65.07% | 98.62% | 78.41% |
| tagger m + all rules | 70.81% | 98.30% | **82.32%** |

*Table 7. Combination with SUBPOS pre-processing: results of the second step*

| | tagger a | tagger b | tagger m |
|---|---|---|---|
| tagger a + safe rules | 92.81% | 95.68% | **95.78%** |
| tagger a + all rules | 93.08% | 95.69% | 95.77% |
| tagger b + safe rules | 92.76% | 95.63% | 95.72% |
| tagger b + all rules | 93.02% | 95.64% | 95.71% |
| tagger m + safe rules | 92.79% | 95.63% | 95.75% |
| tagger m + all rules | 93.05% | 95.64% | 95.73% |

*Table 8. Combination with SUBPOS pre-processing: final accuracy (lines – tagger and rules used in the first two steps, columns – tagger used in the third step)*

8, respectively.

The best result was achieved with tagger *a* in the first step, the set of safe rules in the second step and the tagger *m* in the third step. If we want to use only one tagger (i.e. the same in the first and the third step), the result with tagger *m* and the set of safe rules is nearly as good as the best result.

We performed also experiments with the second step (rules) omitted, because we wanted to check, whether the rules really have some significant impact on the final result, or if the only important step is the SUBPOS pre-processing.

The results in Table 9 show that rules are really important, because the method without

| | tagger a | tagger b | tagger m |
|---|---|---|---|
| tagger a | 92.96% | 95.18% | **95.42%** |
| tagger b | 92.90% | 95.13% | 95.37% |
| tagger m | 92.92% | 95.15% | 95.40% |

*Table 9. Combination with SUBPOS pre-processing: check of the rules efficiency (lines – tagger used in the first step, columns – tagger used in the last step)*

rules does not even reach the accuracy of the best of the standalone taggers.

### 3.3. Combining more taggers in parallel

This method is quite different from previous ones, because it essentially needs more than one tagger. It consists of the following steps:

1. (morphological analysis;)
2. running $N$ taggers independently;
3. merging the results from the previous step — each token ends up with between 1 and $N$ tags, a union of the taggers' outputs;
4. the rule-based component;
5. final disambiguation (single tagger).

This method is based on the assumption that different stochastic taggers make complementary mistakes, so that the recall of the "union" of taggers is almost 100%. Several existing language models are based on this assumption — (Brill, 1998) for tagging English, (Borin, 2000) for tagging German and (Vidová-Hladká, 2000) for tagging inflective languages. All these models perform some kind of "voting" — for every token, one tagger is selected as the most appropriate to supply the correct tag. The model presented in this paper, however, entrusts the selection of the correct tag to another tagger that already operates on the partially disambiguated input.

Results after performing the first two steps, the third and the final step are presented in Tables 10, 11, 12, respectively.

The best results were achieved with two taggers in Step 1 (*a* and *m*), the set of all rules in Step 3 and the tagger *b* in Step 4.

We also measured the accuracy of this method with the rules step omitted. The results of this experiment presented in Table 13 lead to two important conclusions: 1) the rules significantly improve the result (but) 2) the paralell combination without rules performs better than any other purely statistical method or combination.

## 4. Results

Table 14 shows overall results of the best methods in each category (depending on number of components) measured on the dev-test and on the eval-test data.

| | precision | recall | f-measure |
|---|---|---|---|
| $a \cup b$ | 92.18% | 96.90% | 94.48% |
| $a \cup m$ | 92.30% | 97.04% | 94.61% |
| $b \cup m$ | 93.19% | 97.05% | **95.08%** |
| $a \cup b \cup m$ | 90.81% | 97.66% | 94.11% |

*Table 10. Paralell combination: results of the first two steps (union of the tagger's outputs)*

| | precision | recall | f-measure |
|---|---|---|---|
| $(a \cup b)$ + safe rules | 93.56% | 96.74% | 95.12% |
| $(a \cup b)$ + all rules | 93.99% | 96.63% | 95.29% |
| $(a \cup m)$ + safe rules | 93.71% | 96.86% | 95.26% |
| $(a \cup m)$ + all rules | 94.15% | 96.77% | 95.44% |
| $(b \cup m)$ + safe rules | 94.11% | 96.90% | 95.48% |
| $(b \cup m)$ + all rules | 94.46% | 96.81% | **95.62%** |
| $(a \cup b \cup m)$ + safe rules | 92.67% | 97.46% | 95.00% |
| $(a \cup b \cup m)$ + all rules | 93.32% | 97.32% | 95.28% |

*Table 11. Paralell combination: results of the third step (union + rules)*

| | tagger a | tagger b | tagger m |
|---|---|---|---|
| $(a \cup b)$ + safe rules | 95.43% | 95.49% | 95.96% |
| $(a \cup b)$ + all rules | 95.54% | 95.58% | 95.96% |
| $(a \cup m)$ + safe rules | 95.56% | 96.03% | 95.73% |
| $(a \cup m)$ + all rules | 95.68% | **96.09%** | 95.82% |
| $(b \cup m)$ + safe rules | 95.81% | 95.58% | 95.77% |
| $(b \cup m)$ + all rules | 95.89% | 95.71% | 95.86% |
| $(a \cup b \cup m)$ + safe rules | 95.52% | 95.66% | 95.84% |
| $(a \cup b \cup m)$ + all rules | 95.69% | 95.80% | 95.95% |

*Table 12. Paralell combination: final accuracy (lines – taggers and rules used in the first three steps, columns – the tagger used in the last step)*

Table 15 shows the relative error rate reduction. The best method presented by this paper (parallel combination of taggers with all rules) reaches the relative error rate decrease of 11.48% in comparison with the tagger Morče (which achieves the best results for Czech so far).

| | tagger a | tagger b | tagger m |
|---|---|---|---|
| $a \cup b$ | 94.94% | 95.13% | **95.87%** |
| $a \cup m$ | 95.05% | **95.87%** | 95.46% |
| $b \cup m$ | 95.56% | 95.13% | 95.48% |
| $a \cup b \cup m$ | 94.85% | 95.14% | 95.47% |

*Table 13. Paralell combination: check of the rules efficiency (lines – taggers used in the first step, columns – the tagger used in the last step)*

| Components available | The best method | dev-test | eval-test |
|---|---|---|---|
| one tagger | m | 95.43% | 95.12% |
| two taggers | – | – | – |
| three taggers | $(a \cup m) + b$ or $(a \cup b) + m$ | 95.87% | 95.52% |
| one tagger + rules | SUBPOS m + safe rules + m | 95.75% | 95.44% |
| two taggers + rules | $(b \cup m)$ + disheu1 + m | 95.86% | 95.49% |
| thee taggers + rules | $(a \cup m)$ + disheu1 + b | **96.09%** | **95.68%** |

*Table 14. Overall results*

| Method | Morče | Parallel without rules |
|---|---|---|
| Parallel without rules | 8.20% | – |
| Parallel with all rules | 11.48% | 3.57% |

*Table 15. Relative error rate reduction*

### 4.1. Error analysis

Table 16 shows error rate (100% – *accuracy*) of various methods[5] on particular positions of the tags (13 and 14 are omitted). The most problematic position is CASE (5), whose error rate was significantly reduced.

The CASE confusion matrices 18 and 17 show the final situation in more detail. Ambiguity between nominative and accusative remains to be the most problematic even for the hybrid tagging methods.

[5] *Par* stands for parallel combination without rules, *Par+Rul* for parallel combination with rules.

| | a | b | m | Par | Par+Rul |
|---|---|---|---|---|---|
| 1 | 0.61 | 0.70 | 0.66 | 0.57 | 0.57 |
| 2 | 0.69 | 0.78 | 0.75 | 0.64 | 0.64 |
| 3 | 1.82 | 1.49 | 1.66 | 1.39 | 1.37 |
| 4 | 1.56 | 1.30 | 1.38 | 1.18 | 1.15 |
| 5 | 4.03 | 3.53 | 3.08 | 2.85 | 2.62 |
| 6 | 0.02 | 0.03 | 0.03 | 0.02 | 0.02 |
| 7 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 |
| 8 | 0.06 | 0.07 | 0.08 | 0.06 | 0.05 |
| 9 | 0.05 | 0.08 | 0.07 | 0.05 | 0.04 |
| 10 | 0.29 | 0.28 | 0.30 | 0.26 | 0.27 |
| 11 | 0.29 | 0.31 | 0.33 | 0.28 | 0.28 |
| 12 | 0.05 | 0.08 | 0.06 | 0.05 | 0.04 |
| 15 | 0.31 | 0.31 | 0.31 | 0.28 | 0.29 |

*Table 16. Error rate [%] on particular positions of tags*

| tg/an | - | 1 | 2 | 3 | 4 | 5 | 6 | 7 | X |
|---|---|---|---|---|---|---|---|---|---|
| - | 82753 | 37 | 41 | 0 | 18 | 3 | 4 | 7 | 21 |
| 1 | 53 | 26027 | 286 | 11 | 939 | 21 | 8 | 5 | 81 |
| 2 | 9 | 205 | 29363 | 21 | 146 | 0 | 25 | 14 | 24 |
| 3 | 1 | 41 | 70 | 5265 | 54 | 0 | 50 | 23 | 1 |
| 4 | 50 | 1835 | 404 | 12 | 21302 | 1 | 155 | 44 | 15 |
| 5 | 0 | 8 | 0 | 3 | 2 | 36 | 0 | 1 | 0 |
| 6 | 3 | 18 | 54 | 15 | 128 | 0 | 17914 | 3 | 3 |
| 7 | 29 | 26 | 19 | 8 | 73 | 0 | 0 | 9010 | 3 |
| X | 115 | 312 | 90 | 7 | 44 | 21 | 14 | 5 | 4242 |

*Table 17. CASE confusion matrix: paralell combination without rules (rows – output of the combination, columns – annotation)*

## 5. Conclusion

We have presented several variations of a novel method for combining statistical and hand-written rule-based tagging. The best variation improved the accuracy of the best-performing standalone statistical tagger by over 11% (in terms of relative error rate reduction), and the inclusion of the rule-component itself improved the best statistical-only combination by over 3.5% relative.

Our experiments produced a software suite which gives the all-time best results in Czech

| tg/an | - | 1 | 2 | 3 | 4 | 5 | 6 | 7 | X |
|---|---|---|---|---|---|---|---|---|---|
| - | 82747 | 39 | 43 | 2 | 18 | 3 | 2 | 7 | 23 |
| 1 | 50 | 26063 | 290 | 13 | 883 | 22 | 6 | 7 | 97 |
| 2 | 8 | 188 | 29397 | 23 | 128 | 0 | 18 | 16 | 29 |
| 3 | 0 | 37 | 71 | 5310 | 48 | 0 | 14 | 24 | 1 |
| 4 | 37 | 1561 | 406 | 13 | 21597 | 1 | 145 | 41 | 17 |
| 5 | 0 | 10 | 0 | 8 | 2 | 29 | 0 | 1 | 0 |
| 6 | 3 | 17 | 56 | 18 | 120 | 0 | 17917 | 3 | 4 |
| 7 | 31 | 22 | 20 | 8 | 62 | 0 | 0 | 9022 | 3 |
| X | 109 | 285 | 86 | 6 | 48 | 21 | 11 | 6 | 4278 |

*Table 18. CASE confusion matrix: paralell combination with rules*

tagging and which was used to re-tag the existing 200 mil. word Czech National Corpus. It should significantly improve the user experience (for searching the corpus) and allow for more precise experiments with parsing and other NLP applications that use that corpus.

Different variants of the method are available for different tasks – without the rule-bassed component, the accuracy is not much lower and the system runs ten times faster, which makes this variant suitable for large data processing.

## 6. Recent Advances and Outlook

The goal of this paper was to present the main results of the PhD thesis (Spoustová, 2007). There are also some new, unpublished results, which immediately follow the work described in the thesis and in this paper. We would like to present them here (very briefly) before they will be published in a definite form.

We have developed a method of a semi-supervised training of the Morče tagger. The main idea consists in the preparation of the training data: for every iteration, the training data set is unique. Each of the training sets begins with the PDT 2.0 train data set, which is followed by a (unique) part of the Czech National Corpus processed by the parallel combination with rules (the results of this combination are passed to the tagger instead of the human morphological annotation, which is not available for such a large corpus). Thus, every training set contains the same supervised part as the other sets and a unique unsupervised part.

We have experimented with various sizes of the unsupervised parts (from 500k tokens to 5M) and also with various numbers of iterations. During the last year also the supervised Morcče tagger, so we used the newest version ("gangrena").

The preliminary results (PDT 2.0 devel-test) are presented in Table 19. The table contains results of the standalone Morče tagger, results of the two versions of parallel combination, and finally, results of the semi-supervised taggers trained on the parallel combinations.

This preliminary results show that our method of semi-supervised training allows Morče

| Method | accuracy |
|---|---|
| Morče gangrena alone | 95.99% |
| Parallel combination without rules (P1) | 96.03% |
| Parallel combination with rules (P2) | 96.22% |
| Semi-supervised Morče trained on P1 | 96.22% |
| Semi-supervised Morče trained on P2 | 96.23% |

*Table 19. Accuracy of the semi-supervised Morče compared to other methods (devel-test)*

tagger to perform at least as good as the corresponding parallel combination. The output of the parallel combination is needed in the training stage of the tagger, but the tagging process is as fast and simple as when running the supervised tagger.

This method is in development for various languages (Czech, English, Slovak) and final results will be published soon in more detail.

## Acknowledgements

The research described here was supported by the projects *LC536* of *Ministry of Eduation, Youth and Sports* of the Czech Republic.

## Bibliography

Bick, Eckhard. 2000. The parsing system "Palavras" — automatic grammatical analysis of Portuguese in a constraint grammar framework. In: *Proceedings of the 2nd International Conference on Language Resources and Evaluation, TELRI*. Athens.

Borin, Lars. 2000. Something borrowed, something blue: Rule-based combination of POS taggers. In: *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, Vol. 1, pp. 21–26. Athens.

Brill, Eric and Jun Wu. 1998. Classifier combination for improved lexical disambiguation. In: *Proceedings of the 17th international conference on Computational linguistics*, Vol. 1, pp. 191–195. Montreal, Quebec.

Chanod, Jean-Pierre and Pasi Tapanainen. 1995. Tagging French — comparing a statistical and a constraint-based method. In: *Proceedings of EACL-95*, pp. 149–157. Dublin.

Collins, Michael. 2002. Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In: *Proceedings of EMNLP'02*, July 2002, pp. 1–8. Philadelphia.

Daelemans, W., Jakub Zavrel, Peter Berck, and Steven Gillis. 1996. MBT: A memory-based part of speech tagger-generator. In: *Proceedings of the 4th WVLC*, pp. 14–27. Copenhagen.

Erjavec, Tomaz, Saso Dzeroski, and Jakub Zavrel. 1999. Morphosyntactic Tagging of Slovene: Evaluating PoS Taggers and Tagsets. *Technical Report*, Dept. for Intelligent Systems, Jozef Stefan Institute. Ljubljana.

Hajič, Jan and Barbora Hladká. 1997. Tagging of inflective languages: a comparison. In: *Proceedings of ANLP '97*, pp. 136–143. Washington, DC.

Hajič, Jan. 2000. Morphological tagging: Data vs. dictionaries. In: *Proceedings of the 6th ANLP / 1st NAACL'00*, pp. 94–101. Seattle, WA.

Hajič, Jan, Pavel Krbec, Pavel Květoň, Karel Oliva, and Vladimír Petkevič. 2001. Serial Combination of Rules and Statistics: A Case Study in Czech Tagging. In: *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*. CNRS – Institut de Recherche en Informatique de Toulouse and Université des Sciences Sociales, pp. 260–267. Toulouse.

Hajič, Jan, Eva Hajičová, Jarmila Panevová, Petr Sgall, Petr Pajas, Jan Štěpánek, Jiří Havelka, and Marie Mikulová. 2006. Prague Dependency Treebank v2.0. CDROM. Linguistic Data Consortium, Cat. LDC2006T01. Philadelphia. ISBN 1-58563-370-4. Documentation also at `http://ufal.ms.mff.cuni.cz/pdt2.0`.

Karlsson, Fred. 1985. Parsing Finnish in terms of a process grammar. In: Fred Karlsson (ed.): *Computational Morphosyntax: Report on Research 1981-84*, University of Helsinki, Department of General Linguistics Publications No. 13, pp. 137–176.

Karlsson, Fred, Atro Voutilainen, Juha Heikkilä, and Arto Anttila (eds.). 1995. Constraint Grammar: a language-independent system for parsing unrestricted text. *Natural Language Processing*. Vol. 4, Mouton de Gruyter, Berlin and New York.

Koskenniemi, Kimmo. 1990. Finite-State Parsing and Disambiguation. In: *Proceedings of Coling-90*, University of Helsinki, 1990, pp. 229–232. Helsinki.

Krbec, Pavel. 2005. *Language Modelling for Speech Recognition of Czech.* PhD Thesis, MFF, Charles University Prague.

Květoň, Pavel. 2005. *Rule-based Morphological Disambiguation.* PhD Thesis, MFF, Charles University Prague.

Květoň, Pavel. 2006. Rule-based morphological disambiguation: On computational complexity of the LanGR formalism. In: *The Prague Bulletin of Mathematical Linguistics*, Vol. 85, pp. 57–72. Prague.

Oflazer, Kemal and Gökhan Tür. 1997. Morphological disambiguation by voting constraints. In: *Proceedings of the 8th conference on European chapter of the Association for Computational Linguistics*, pp. 222–229. Madrid.

Oliva, Karel, Milena Hnátková, Vladimír Petkevič, and Pavel Květoň. 2000. The Linguistic Basis of a Rule-Based Tagger of Czech. In: Sojka P., Kopeček I., Pala K. (eds.): *Proceedings of the Conference "Text, Speech and Dialogue 2000"*, *Lecture Notes in Artificial Intelligence*, Vol. 1902. Springer-Verlag, pp. 3–8. Berlin-Heidelberg.

*PDT Guide.* `http://ufal.ms.mff.cuni.cz/pdt2.0`

Ratnaparkhi, A.. 1996. A maximum entropy model for part-of-speech tagging. In: *Proceedings of the 1st EMNLP*, May 1996, pp. 133–142. Philadelphia.

Samuelsson, Christer and Atro Voluntainen. 1997. Comparing a linguistic and a stochastic tagger. In: *Proceedings of ACL/EACL Joint Converence*, pp. 246–252. Madrid.

Smith, Noah A., David A. Smith, and Roy W. Tromble. 2005. Context-Based Morphological Disambiguation with Random Fields. In: *Proceedings of HLT/EMNLP*, pp. 475–482. Vancouver.

Spoustová, Drahomíra “johanka”. 2007. *Kombinované statisticko-pravidlové metody značkování češtiny. (Combining Statistical and Rule-Based Approaches to Morphological Tagging of Czech Texts).* PhD Thesis, MFF UK.

Spoustová, Drahomíra “johanka”, Jan Hajič, Jan Votrubec, Pavel Krbec, and Pavel Květoň. 2007. Cooperation of Statistical and Rule-Based Taggers for Czech. In: Proceedings of Balto-Slavonic Natural Language Processing Workshop, ACL, Prague 2007. pp. 67–74. Prague.

Tapanainen, Pasi and Atro Voutilainen. 1994. Tagging accurately: don’t guess if you know. In: *Proceedings of the 4th conference on Applied Natural Language Processing*, pp. 47–52. Stuttgart.

Vidová-Hladká, Barbora. 2000. *Czech Language Tagging.* PhD thesis, MFF UK. Prague.

Votrubec, Jan. 2005. *Volba vhodných rysů pro morfologické značkování češtiny. (Feature Selection for Morphological Tagging of Czech.)* Master thesis, MFF, Charles University, Prague.

Votrubec, Jan. 2006. Morphological Tagging Based on Averaged Perceptron. In: *WDS’06 Proceedings of Contributed Papers*, MFF UK, pp. 191–195. Prague.

PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 89 JUNE 2008 41–96

# The Czech Academic Corpus 2.0 Guide

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

**Abstract**

The Czech Academic Corpus version 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 Academy of Sciences of the Czech Republic from 1971 to 1985. When the CAC project began there were only two computerized annotated corpora available since the 1960s - the Brown Corpus of American English and the LOB Corpus of British English. Both corpora became well known to corpus linguists, whereas the CAC remained hidden mainly because of the 1980s political regime in the Czech Republic.

The idea of transferring the internal format and annotation scheme of the CAC into the Prague Dependency Treebank (PDT) concept emerged during the work on the PDT's second version. The main goal was to make the CAC and the PDT fully compatible and thus enable the integration of the CAC into the PDT. The currently released second version of the CAC presents the complete conversion of the internal format and morphological and syntactical annotation schemes. The Czech Academic Corpus v. 2.0 is being published by the Linguistic Data Consortium.

## 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. A 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


Please cite this article as: Barbora Vidová Hladká, Jan Hajič, Jirka Hana,
Jaroslava Hlaváčová, Jiří Mírovský, Jan Raab, The Czech Academic Corpus 2.0 Guide. The Prague Bulletin of Mathematical Linguistics No. 89, 2008, 41–96.

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.

## 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

[1] This text contains both bibliographic references (e.g. Vidová Hladká et al., 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).

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 et al., 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 et al., 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 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).

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

*Table 1. Examples of lemmas and tags of particular word forms*

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 1 shows the syntactical annotation of the sentence *Obecná odpověď na tuto otázku je sotva možná.* (E.: *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`.

The conception of the main internal format of the CAC 2.0 (in PML format – see Chapter

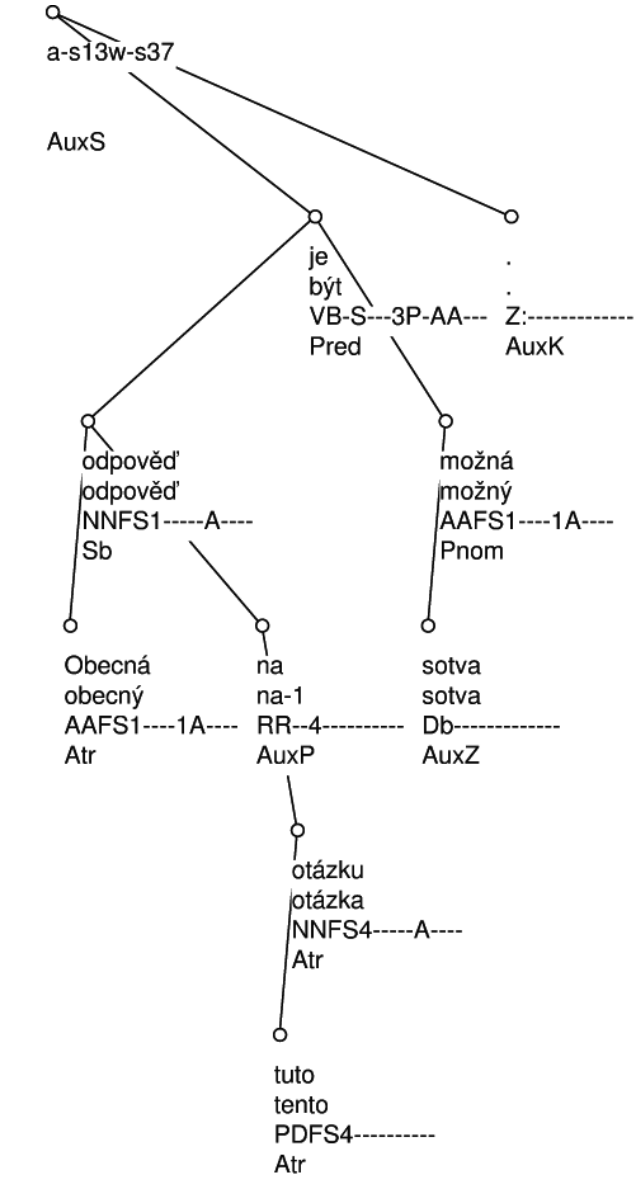



*Figure 1. Example of an a-layer annotation*

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 serves to illustrate the 1:1 ratio of the layers. The layers do not differ except for the final punctuation. Figure 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 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.

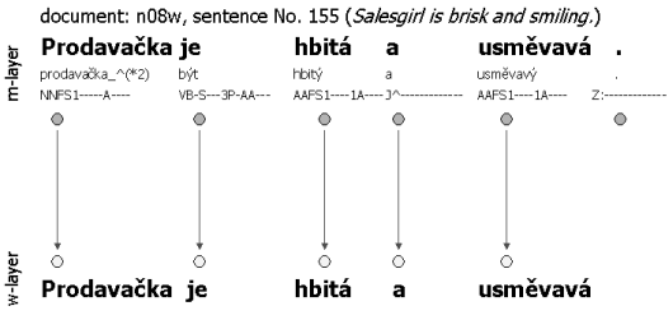



*Figure 2. Technical interconnection of the w-layer and m-layer: No changes other than the final-sentence punctuation*

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

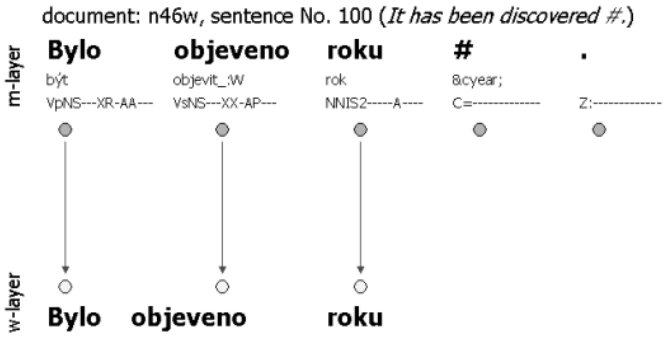



*Figure 3. Technical interconnection of the w-layer and m-layer: The insertion of a word token*

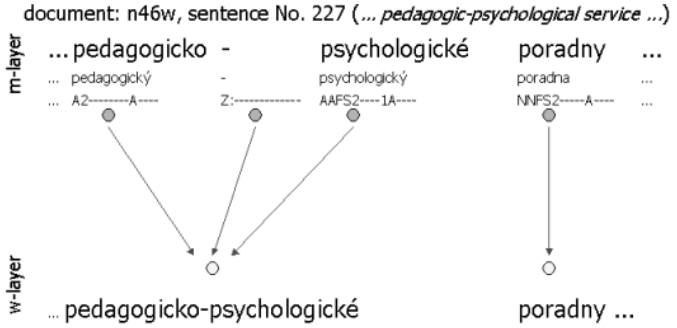



*Figure 4. Technical interconnection of the w-layer and m-layer: The division of a word token*

exception is the technical root, which has no counterpart on the m-layer. Figure 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 and 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 et al., 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. Let us note that in contrast to the PDT, no layer of underlying syntantic annotation is handled in the CAC. Based on the experiences from the morphological annotation, we have split this question into three subquestions: *Automatically?Semi-automatically?Manually?* The article by Ribarov, Bémová, and Vidová (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

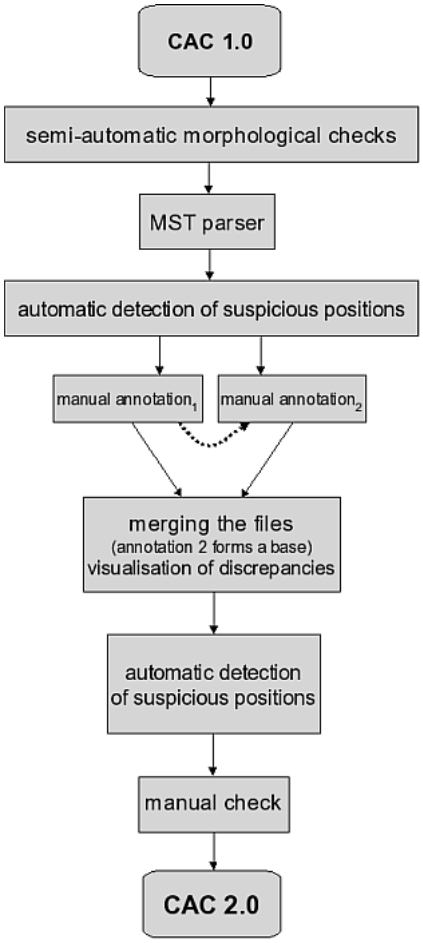



*Figure 5. CAC 2.0 preparation – data processing*

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 morphological 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 5 illustrates operations on the data since the CAC 1.0 release up until the CAC 2.0 release.

## 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 2 on page 51). Any references made regarding the content of the CD-ROM that resides deeper within the tree structure notes the full path to the file.

### 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.

| | |
|---|---|
| `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 |

*Table 2. CAC 2.0 CD-ROM – Directory structure*

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 and 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

```
<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>
...
```

*Table 3. The PML schema of the w-layer in the CAC 2.0*

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 on page 52 (`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).

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 4 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`).

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

*Table 4. Part of the header of the m-layer instance `n01w.m`*

```
<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>
...
```

*Table 5. Part of the header of the a-layer instance `n01w.a`*

Table 5 on page 53 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`).

```
<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 6. An example of sentence m-layer annotation in the PML format*

The annotation is expressed using XML elements and attributes named and used according to their corresponding PML schema. Table 6 illustrates an example of the morphological annotation of a part of the sentence *Váš boj je i našim bojem* (E.: *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 7 on page 56 shows an example of the analytic annotation of a sentence *Váš boj je i naším bojem.* (E.: *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.

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 8 on page 57 for an example of the complete annotation of the sentence *Váš boj je i naším bojem.* (E.: *Your fight is our fight too.*) in CSTS 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>
```



*Table 7. An example of sentence a-layer annotation in the PML 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
```

*Table 8. An example of sentence annotation in CSTS format*

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 6. 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.

| Style | Form | # docs | # sentences | # word tokens | # word tokens w/o punctuation | # 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 9. Size of the CAC 2.0 parts according to style and form*

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 9 on page 58 states the sizes of the individual parts of the data according to its style and form.

Table 10 contains separate quantitative data for the characters "#" and "?" that were manually inserted into the CAC to replace missing words and numbers written as digits.

| **Style** | **Form** | **# "#" characters (in a specified number of sentences)** | **# "?" (in a specified number of sentences)** | **# "#" or "?" (in a specified number of sentences)** | **# 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 |

*Table 10. Quantitative characteristics of the CAC 2.0 – replacement characters "#" and "?"*

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 11 on page 60 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.

| Characteristics | PDT 2.0 | | CAC 2.0 | |
|---|---|---|---|---|
| | # words (thousands) | # sentences (thousands) | # words (thousands) | # 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 |

*Table 11. A comparison of the PDT 2.0 and the CAC 2.0*

### 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 12 on page 61 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 describe the tools in more detail.

#### 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 6 on page 62 illustrates the Bonito main screen. The command of the tool is demonstrated in the following examples.

Figure 6 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

| Tool | Description | Purpose |
|---|---|---|
| Bonito | Corpus manager | • Searching within CAC 2.0 texts<br>• Searching within the morphological annotation of the CAC 2.0<br>• Searching within the analytical functions assigned to words in the CAC 2.0 as a part of the a-layer<br>• 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<br>• Morphological analysis<br>• Tagging (automatic disambiguation of morphological analysis results)<br>• Parsing (automatic syntactical analysis with analytical functions assignment) |

*Table 12. Tools – outline*

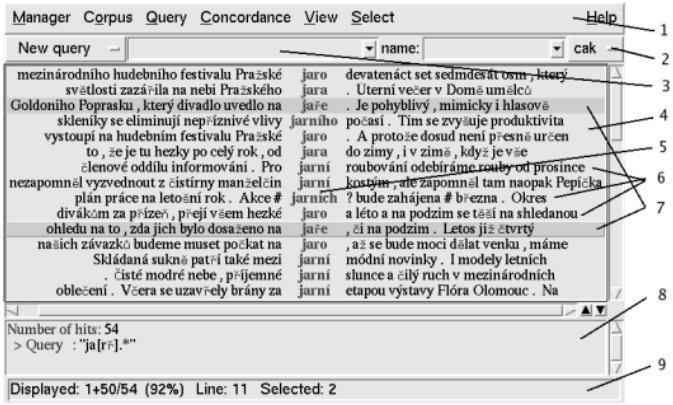



*Figure 6. Bonito: Main screen*

- 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 7.

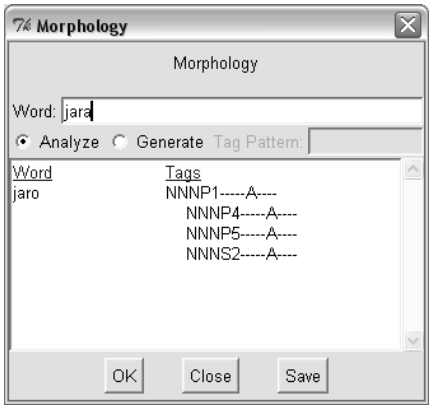



*Figure 7. Bonito: Running the morphological analyser*

The tutorial 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 8.

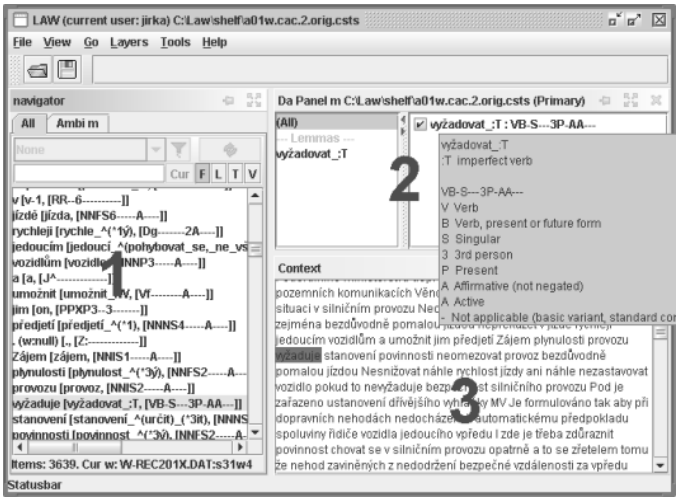



*Figure 8. LAW: Main screen*

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 9 on page 65 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.*)

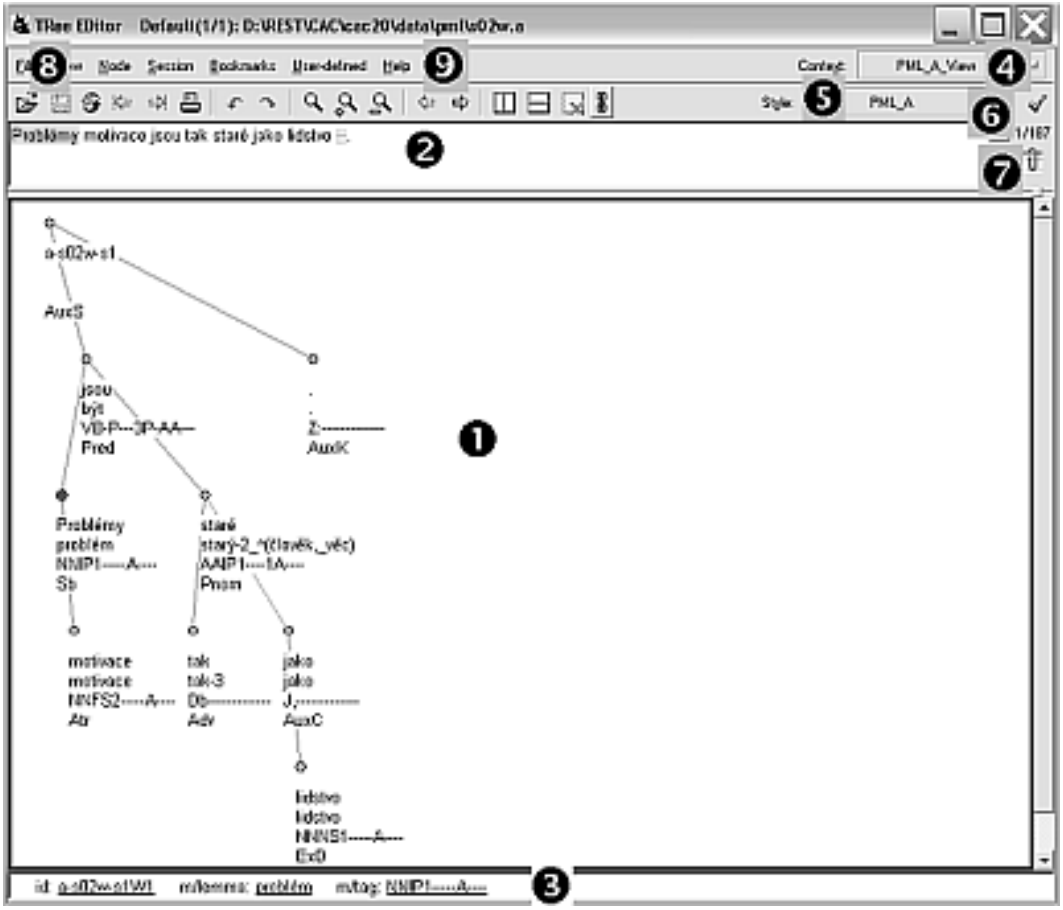



*Figure 9. TrEd: Main screen*

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.

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, see Mírovský (2008).

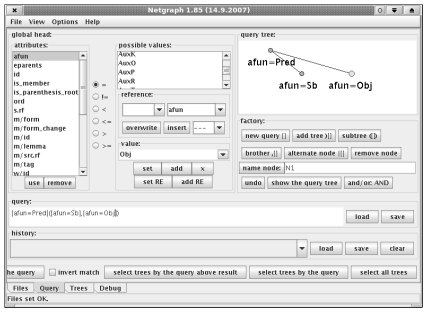



*Figure 10. Netgraph: Query formulation*

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 de-

pendent 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 10 on page 66 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.

The tree in Figure 11 could be one of the results the server returns.

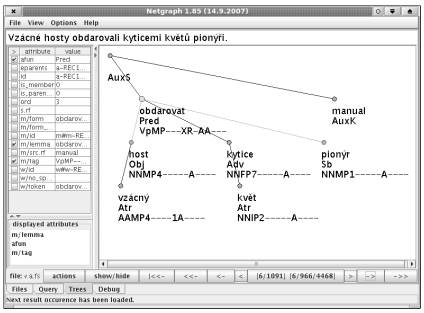



*Figure 11. Netgraph: Query result*

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 be-

ginning 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 (see 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 13 on page 69. 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.

| 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 |

*Table 13. Script tool_chain*

| 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_ˆ(zvr._zájmeno/částice)* P7-X3———- | *se_ˆ(zvr._zájmeno/částice)* P7-X3———- |
| *však* | *však* J^———- | *však* J^———- |
| *Neumannová* | *Neumannová_;S* NNFS1—–A—- NNFS5—–A—- | *Neumannová_;S* NNFS1—–A—- |
| *doběhla* | *doběhnout_:W* VpQW—XR-AA–1 | *doběhnout_:W* VpQW—XR-AA–1 |
| *pro* | *pro-1* RR–4———- | *pro-1* RR–4———- |
| *vytoužené* | *vytoužený_ˆ(*3it)* AAFP1—-1A—- AAFP4—-1A—- AAFS2—-1A—- AAFS3—-1A—- AAFS6—-1A—- AAIP1—-1A—- AAIP4—-1A—- AAIP5—-1A—- AAMP4—-1A—- AANS1—-1A—- AANS4—-1A—- AANS5—-1A—- | *vytoužený_ˆ(*3it)* 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:———- |

*Table 14. An example of text treated with morphological analysis and tagging*

**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 14 on page 69. 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 12 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.

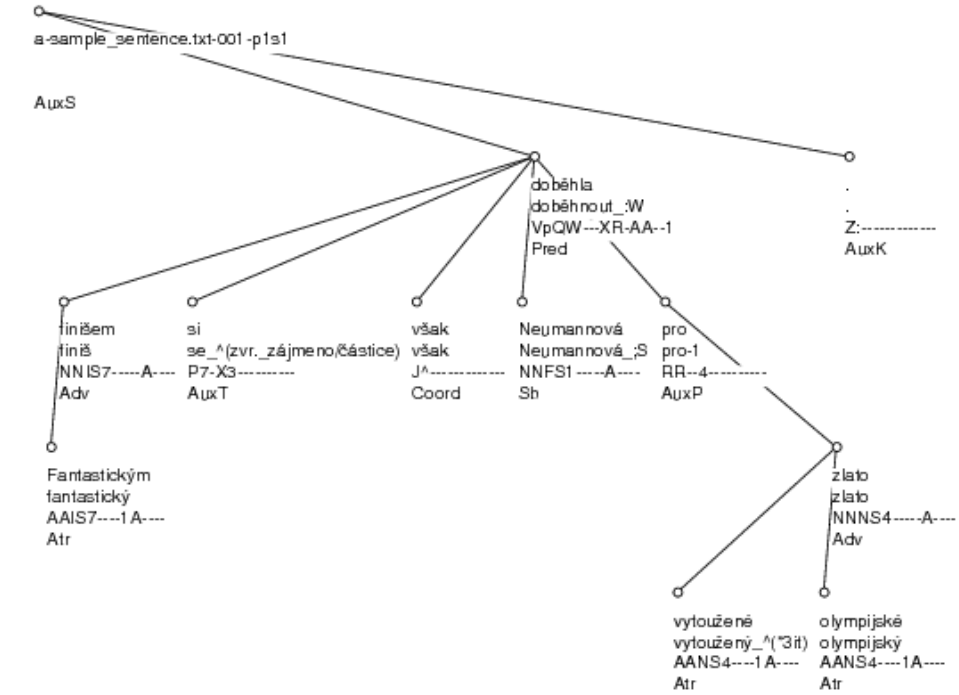



*Figure 12. An example of sentence parsing*

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.

## 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 by 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`).

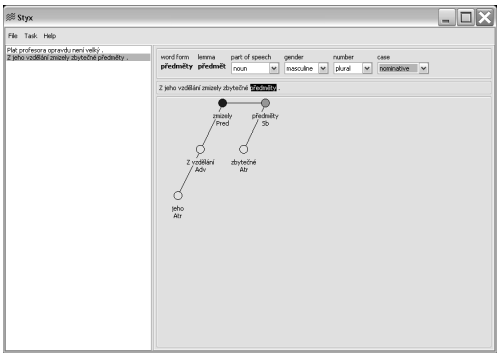



*Figure 13. STYX: Exercises*

The steps for using STYX are clearly illustrated in Figure 13. 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 14 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).

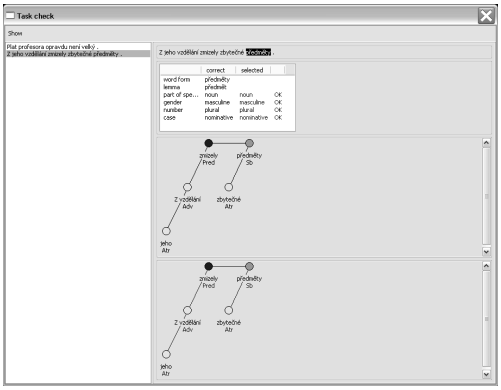



*Figure 14. STYX: Exercise evaluation*

### 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, see 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 15 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] (see Müller, Psutka, and 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 et al. (2006).


*Figure 15. The TrEd editor screen with the TrEdVoice module enabled*

## 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 15 on page 74 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 16 on page 74 lists all tutorials (videos, demos and texts) concerning the tools.

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

*Table 15. Data 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] | — |

*Table 16. Tool tutorials*

## 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 17 on page 75 summarises all tools contained on the CD-ROM and the possibility to run them in Linux and MS Windows operating systems.

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.

| 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 |

*Table 17. Tools compatibility with Linux and MS Windows operating systems*

**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.

## 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). 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.

## 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

## 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.

## Bibliography

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

Čermák, František and Renata Blatná. 2005. *Jak využívat Český národní korpus.* Nakladatelství Lidové noviny, Prague.

Hajič, Jan. 2004. *Disambiguation of Rich Inflection (Computational Morphology of Czech).* Karolinum, Prague.

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

Hana, Jiří, Daniel Zeman, Jan Hajič, Hana Hanová, Barbora Hladká, and Emil Jeřábek. 2005. Manual for Morphological Annotation. Technical Report TR-2005-27, Institute of Formal and Applied Linguistics ÚFAL/CKL MFF UK, Prague, Czech Republic.

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

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

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

Kučera, Ondřej. 2006. Pražský závislostní korpus jako cvičebnice jazyka českého (Prague Dependency Treebank as an Exercise Book of Czech). Master's thesis, MFF UK, Prague, Czech Republic.

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

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

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

Müller, Luděk, Josef Psutka, and Luboš Šmídl. 2000. Design of Speech Recognition Engine. *TSD 2000, Lecture Notes in Artificial Intelligence*, pages 259–264.

Pajas, Petr and Jan Štěpánek. 2005. A Generic XML-based Format for Structured Linguistic Annotation and its Application to the Prague Dependency Treebank 2.0. Technical Report TR-2005-29, ÚFAL/CKL MFF UK, Prague, Czech Republic.

Přikryl, Leoš. 2007. Rozhraní Rozhraní v mluveném jazyce pro korpusové anotační nástroje. Master's thesis, Charles University, Prague.

Psutka, Josef, Luděk Müller, Jindřich Matoušek, and Vlasta Radová. 2006. *Mluvíme s počítačem česky*. Academia, Prague.

Ribarov, Kiril. 2004. *Automatic Building of a Dependency Tree – The Rule-Based Approach and Beyond*. Ph.D. thesis, MFF UK, Prague, Czech Republic.

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

Savický, Petr and Jaroslava Hlaváčová. 2002. Measures of Word Commonness. *Journal of Quantitative Linguistics*, 9(3):215–231.

Šmilauer, Vladimír. 1972. *Nauka o českém jazyku*. Prague.

Vidová Hladká, Barbora, Jan Hajič, Jiří Hana, Jaroslava Hlaváčová, Jiří Mírovský, and Jan Votrubec. 2007. *Průvodce Českým akademickým korpusem 1.0*. Karolinum, Prague.

Votrubec, Jan. 2005. Volba vhodné sady rysů pro morfologické značkování češtiny (Selecting an Optimal Set of Features for the Morphological Tagging of Czech). Master's thesis, MFF UK, Prague, Czech Republic.

## Appendix A. Sources of the texts

| File | Written form | File | Transcription |
|---|---|---|---|
| a01w | Vyhláška č. 100 | a16s | Zelená vlna |
| a02w | Hospodaření s domovním bytovým majetkem | a17s | Zprávy o počasí |
| | | a18s | Přehled rozhlasových pořadů |
| a03w | Pracovní řád | a19s | Hlášení v metru |
| a04w | Národní pojištění 12/1977 | | |
| 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 18. Administrative documents*

| 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 | | |
| 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 19. Documents covering journalism*

| 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 |
| 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 | | |

*Table 20. Documents covering the scientific field*

## Appendix B. Description of lemmas

In the CAC 2.0, lemma has a form of string lemma_:P1_;P2_,P3_ˆ(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.

| 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 | ˆ | comment | explanatory note, derivational comments, other comments |

*Table 21. Additional information of the lemmas*

| Value | Description |
|---|---|
| B | abbreviation |
| T | imperfect verb |
| W | perfect verb |

*Table 22. Morpho-syntactic 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 23. Semantic 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 24. Style flags of the 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" |

*Table 25. Examples of lemmas*

## Appendix C. Description of tags

| 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 26. Part of speech*

| *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 Sub-part of speech = #) | 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 |

*continued on next page*

*Sub-part of speech*

*continued from previous page*

| Value | Description | POS |
|---|---|---|
| 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" (Case = 4, see Table 30), "si" (Case = 3, see Table 30), plus the same two forms with contracted -s: "ses", "sis" (distinguished by Person = 2, see Table 33; 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 |

*continued on next page*

*Sub-part of speech*

*continued from previous page*

| Value | Description | POS |
|---|---|---|
| K | Relative/interrogative pronoun "kdo" (lit. "who"), incl. forms with affixes -ž and -s (affixes are distinguished by the category Variant - see Table 40 - (for -ž) and Person- see Table 33 - (for -s)) | P – pronoun |
| 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", "čímkoli", "cosi", …, lit. "some", "some", "anybody's", "something") | P – pronoun |

*continued on next page*

*Sub-part of speech*

*continued from previous page*

| Value | Description | POS |
|---|---|---|
| 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 Negation (Table 36) as well as the Grade (Table 35) 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, see Table 36 (Negation set to A/N) and degrees of comparison (Table 35) Grade set to 1/2/3 (comparative/superlative), e.g. "velký", "zajímavý", …, 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 |
| 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" (Variant = 4, see Table 40), 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 |

*continued on next page*

| *Sub-part of speech* | | |
|---|---|---|
| *continued from previous page* | | |
| **Value** | **Description** | **POS** |
| 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 27: Sub-part of speech

| **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 feminine (i.e., Masculine animate/inanimate or Neuter); only for (some) pronoun forms and certain numerals |

*Table 28. Gender*

| 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 29. Number*

| 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 30. Case*

| 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 31. Possessive gender*

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

*Table 32. Possessive number*

| 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 33. Person*

| Value | Description |
|---|---|
| F | Future |
| H | R, P – Past or Present |
| P | Present |
| R | Past |
| X | Any |

*Table 34. Tense*

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

*Table 35. Grade*

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

*Table 36. Negation*

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

*Table 37. Voice*

| Value | Description |
|---|---|
| - | not applicable |

*Table 38. Reserve 1*

| Value | Description |
|---|---|
| - | not applicable |

*Table 39. Reserve 2*

| 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. |

*Table 40. Variant*

## Appendix D. Analytical function 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;<br>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)<br>dependency without a semantic difference |
| AdvAtr | dtto with reverse preference |
| Atv | complement (so-called determining) technically hung on a non-verbal element |
| AuxT | reflexive tantum |
| AuxG | other graphic symbols, not terminal |
| 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<br>and adnominal dependency without a semantic difference |
| ObjAtr | dtto with reverse preference |
| Atr | attribute |
| AuxP | primary preposition, parts of a secondary preposition |
| AuxS | root of the tree (#) |

*Table 41. Analytical functions (AF) in the CAC 2.0*

## Appendix E. World Wide Web links

| *World Wide Web links*<br>**Name (description)**<br>**Location** |
|---|
| **PROJECTS** |
| 1. Resources and tools for information systems<br>http://ufal.mff.cuni.cz/rest |
| 2. Morphological tagging of Czech (a complete guide)<br>http://ufal.mff.cuni.cz/czech-tagging |
| 3. Parsing of Czech (a complete guide)<br>http://ufal.mff.cuni.cz/czech-parsing |
| **INSTITUTIONS** |
| 4. Academy of Sciences of the Czech Republic<br>http://www.cas.cz |
| 5. Grant Agency of the Academy<br>of Sciences of the Czech Republic<br>http://www.gaav.cz |
| 6. Department of Cybernetics<br>of the University of West Bohemia in Plzen,<br>Czech Republic<br>http://www.kky.zcu.cz |
| 7. Ministry of Education, Youth and Sports<br>of the Czech Republic<br>http://www.msmt.cz |
| 8. Charles University in Prague, Czech Republic<br>http://www.cuni.cz |
| 9. Institute of Formal and Applied Linguistics,<br>Faculty of Mathematics and Physics,<br>Charles University in Prague, Czech Republic<br>http://ufal.mff.cuni.cz |
| 10. Linguistic Data Consortium, Philadelphia, PA, USA<br>http://www.ldc.upenn.edu |
| 11. Institute of Czech Language,<br>Academy of Sciences of the Czech Republic<br>http://www.ujc.cas.cz |
| *continued on next page* |

*World Wide Web links*

*continued from previous page*

**Name (description)**
**Location**

**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
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:

*continued on next page*

*World Wide Web links*

*continued from previous page*

| | **Name (description)**<br>**Location** |
|---|---|
| | Prague Treebanking for Everyone<br>http://lectures.ms.mff.cuni.cz/video/categoryshow/index/1 |
| 29. | Tutorial on STYX<br>http://lectures.ms.mff.cuni.cz/video/recordshow/index/2/27 |
| 30. | Tutorial on TrEd<br>http://lectures.ms.mff.cuni.cz/video/recordshow/index/2/23 |
| 31. | XML<br>http://www.w3.org/XML |
| | **TOOLS** |
| 32. | Bonito (graphical user interface of the Manatee corpus manager)<br>http://nlp.fi.muni.cz/projekty/bonito/ |
| 33. | LAW (morphological annotation editor)<br>http://www.ling.ohio-state.edu/ hana/law.html |
| 34. | Morče (morphological tagger of Czech)<br>http://ufal.mff.cuni.cz/morce |
| 35. | Netgraph (tool for searching dependency corpora)<br>http://quest.ms.mff.cuni.cz/netgraph |
| 36. | STYX (electronic exercise book of Czech based on PDT)<br>http://ufal.mff.cuni.cz/styx |
| 37. | TrEd (syntactical annotation editor)<br>http://ufal.mff.cuni.cz/ pajas/tred |
| 38. | TNT (Trigrams'n'Tags tagger)<br>http://www.coli.uni-saarland.de/ thorsten/tnt/ |

PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 89 JUNE 2008 97–106

# De la théorie à l'application : VALLEX, une démarche exemplaire

Patrice Pognan

**Abstract**

VALLEX est le fruit du temps : le temps de réfléchir, le temps de tester, le temps de faire, le temps d'utiliser. VALLEX est le contre-exemple prototypique de tout ce que souhaitent les politiques actuelles de la recherche : c'est pour les chercheurs sérieux le réconfort d'apprécier la richesse qu'apportent la pérennité d'une équipe et de ses thèmes de recherche, l'effet cumulatif des connaissances d'une génération de chercheurs à l'autre. L'histoire de VALLEX prend ses racines dans les années soixante et ne peut pas être dissociée de l'histoire de Petr Sgall et de ses disciples qui pour vivre l'aventure de la recherche ont dû d'abord lutter pour la survie de leur équipe, de ses idées, de ses programmes.

## 1. La théorie

Nous avons jugé inutile une énième présentation de la théorie, la Description Générative Fonctionnelle (DGF) et préféré en commenter les aspects qui nous semblent primordiaux. Le lecteur trouvera des descriptions précises en particulier dans [Lopatková, 2003, PBML 79-80] et dans [Žabokrtský, Lopatková, 2007, PBML 87].

Nous donnons en annexe une bibliographie conçue de manière particulière : nous avons ordonné dans le temps quelques publications qui nous semblent importantes de l'ensemble de l'équipe. Le résultat est frappant sous plusieurs aspects.

La première remarque est claire : les années soixante sont la période de genèse de la théorie, la DGF, réalisée par Petr Sgall. Les années 70, 80 et 90 (trente années de travail !) sont globalement les années de développement de la théorie avec l'élaboration constante d'outils de test de cette théorie par P. Sgall et ses disciples – collaborateurs Eva Hajičová et Jarmila Panevová. (Nous en donnerons plus bas une interprétation plus fine). Enfin, les années 2000 voient apparaître sur le devant de la scène tout un ensemble de jeunes chercheurs « seconde génération » de la DGF tournés vers les applications informatiques, en particulier dans le cadre des travaux autour du Prague Dependency Treebank (PDT) et vers la réalisation concrète d'un dictionnaire


Please cite this article as: Patrice Pognan, De la théorie à l'application : VALLEX, une démarche exemplaire. The Prague Bulletin of Mathematical Linguistics No. 89, 2008, 97–106.

de valences, Vallex, ce qui marque la matérialisation de la théorie en une suite d'applications.

Il convient de souligner que la richesse d'applications bien fondées scientifiquement n'apparaît de manière évidente que dans la cinquième décennie après le début des recherches, que c'est une nouvelle génération de chercheurs qui, avec l'appui constant des chercheurs de la première génération, crée et affine les produits dont la validité est issue de la théorie. Ceci devrait être un guide de réflexion pour les « décideurs » … Le fait d'arriver vers les numéros 90 d'une revue bi-annuelle (le PBML) entièrement créée, nourrie et gérée par une équipe laisse également rêveur …

Dans un deuxième temps, nous allons considérer les développements « forts » de la DGF des années 70, 80 et 90.

– Les années 70 sont celles du renforcement de la DGF. Elles sont marquées principalement par l'analyse de la partition thème / rhème [Sgall, Benešová, 1973], [Sgall, Hajičová, 1977, 1978], [Sgall, 1979] et les études sur le cadre verbal [Panevová, 1974, 1975, 1977], [Panevová, Sgall, 1976].

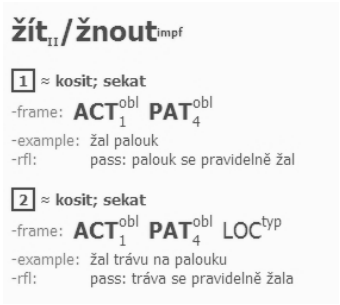
– Les années 80 sont celles de la maturité de la théorie et l'époque d'un faire-savoir important [Sgall, 1980, 1984], [Sgall, Hajičová, Panevová, 1986]. Ce sont également les années de recherche déterminante d'une part, vers la syntaxe profonde, le niveau tectogrammatical [Hajičová, Panevová, 1984] qui permet de formuler une interprétation sémantique de la phrase et du texte et d'autre part, pour la patiente mise en exergue de ce que nous considérons comme le maillon fondamental pour l'automatisation et les applications de type Vallex, l'ordre systémique sans lequel rien ne serait possible [Hajičová, Sgall, 1986].

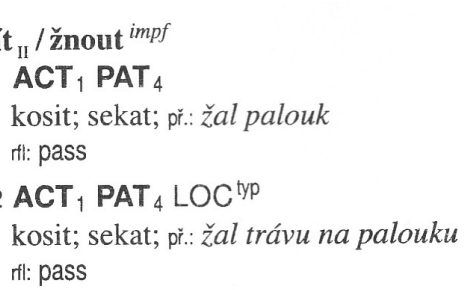
– Les années 90 voient l'apparition de concepts avancés tels que celui de contrôle [Panevová, 1996] et le renforcement des applications de grande envergure. Notons que l'équipe est connue sur toute son histoire pour ses applications dans les domaines de la traduction automatique, de l'indexation et de la recherche d'information.

Mais c'est certainement la prise en compte de l'ensemble de quarante ans de travaux (de 1960 à 2000) qui fait de la Description Générative Fonctionnelle la théorie (et la pratique !) capable de pleinement transformer les travaux de Tesnière en un système de calcul de la langue.

## 2. L'application Vallex

VALLEX existe en trois versions : une version HTML consultable en ligne, une version XML permettant l'utilisation du dictionnaire par programmation et une version papier qui vient d'être publiée [Lopatková, … 2008]. Cette version contient le dictionnaire de valences (environ 350 pages) dont l'organisation graphique s'inspire heureusement de l'interface HTML. Le dictionnaire est précédé d'une introduction détaillée de 20 pages et d'une bibliographie abondante.




Nous avons, à gauche, la forme issue de la consultation HTML et à droite, celle adoptée dans le dictionnaire. On y observe deux simplifications : les exemples ne sont pas donnés pour les formes réflexives et réciproques (un Tchèque reconstruit facilement ce type de construction, mais les exemples peuvent être utiles à un lecteur étranger) et les foncteurs représentant les participants internes sont considérés par défaut obligatoires. Ils ne portent un exposant que s'ils sont facultatifs (exposant « opt ») :

**splácet** *impf*, **splatit** *pf*
1 **ACT**$_1$ **ADDR**$_3^{opt}$ **PAT**$_4$ RCMP$_{za+4}^{typ}$ MANN$^{typ}$

Ce dictionnaire a été pensé comme outil pour le public tchèque. En témoigne l'introduction rédigée en tchèque. Il nous semble cependant regrettable que l'on n'ait pas pris en considération l'usage qu'un étranger, même sans connaissance du tchèque, peut en faire ne serait-ce qu'à titre d'exemple pour des travaux sur d'autres langues. Doubler l'introduction tchèque par une introduction dans une ou plusieurs langues internationales (au moins en anglais, mais aussi peut-être en français, espagnol, allemand) aurait été bienvenu. Cela paraît d'autant plus surprenant (et même contradictoire) que le rapport technique interne au laboratoire possède une très bonne introduction en anglais [Lopatková, … 2006] et qu'un article en anglais a été publié en juin 2007 dans le Prague Bulletin of Mathematical Linguistics [Žabokrtský, … 2007]. Le travail était quasiment fait ! A défaut, le lecteur étranger devra donc se munir du numéro 87 de cette revue.

Etant donnée l'existence de cet article, nous ne reprendrons pas, dans cette même revue, la description détaillée de Vallex. Nous nous contenterons d'insister sur quelques points qui nous semblent importants.

Il est intéressant que les auteurs aient suivi la Description Générative Fonctionnelle de P. Sgall dans la constitution d'entrées possédant simultanément tous les lemmes aspectuels, ce qui a pour mérite de montrer que la bipolarité aspectuelle tant prônée peut s'étendre de manière très fréquente à une triade due à l'itératif sachant que l'on peut être en présence d'un nombre de lemmes beaucoup plus élevé : jusqu'à 6 !

Dans une approche linguistique centrée sur le verbe, il convient, en premier lieu, d'insister sur la nécessité de posséder des informations précises sur le cadre verbal. L'information sur les groupes compléments du verbe (ce que les auteurs appellent en anglais à juste titre « comple-

mentation » étant donné qu'il peut s'agir de réalisations sous forme de syntagmes nominaux, de locutions adverbiales ou même de propositions subordonnées) représente une description syntaxico-sémantique précise signification par signification du verbe (ses différents sens). Nous pensons que le terme d'unité lexicale utilisé par les auteurs tant en anglais qu'en tchèque pour nommer le cadre verbal de chacune des significations du verbe n'est pas réellement approprié. C'est cette information qui fait la validité d'un tel dictionnaire pour la rédaction en tchèque et la traduction du ou vers le tchèque. Pour chacun des sens, ces cadres verbaux sont divisés en participants internes (actants), en « quasi-actants » (la différence, l'intention et l'obstacle) et en participants externes, libres (complémentation libre - circonstants). A l'intérieur du cadre verbal chaque foncteur peut être obligatoire ou facultatif et accompagné en indice de ses rections syntaxiques.

Les rections syntaxiques des foncteurs peuvent être gérées par classes d'équivalence notées par un symbole du type « AIM » (but) regroupant un ensemble de valeurs telles que « aby » (afin, pour), « ať » (que), « do+2 » (dans, à + génitif), …, « v zájmu+2 » (dans l'intérêt de + génitif), …c'est-à-dire des connecteurs syntaxiques, des prépositions simples ou dérivées, … Ce type de démarche reflète l'implémentation qui peut être faite pour une analyse automatique du tchèque.

Dans le même genre d'idée, certains foncteurs de temps ou de lieu pouvant alterner sont représentés par un foncteur prototypique (au nombre de 5), ce qui offre la souplesse nécessaire à une bonne analyse automatique.

Enfin, l'affectation à environ 45(cadre pour chacun des sens d'un verbe) d'une catégorie sémantique générale (il en existe pour le moment 22), par exemple « transport », « mouvement », « phase d'une action », … rapproche de travaux de nature sémantico-cognitive. Cette direction, pour le moment exploratoire, devrait être sérieusement étudiée et affinée.

Nous nous permettrons de souligner que l'usage du dictionnaire ne dispense pas de la consultation de Vallex sous sa forme HTML qui reste nécessaire grâce à la souplesse et la multiplicité des accès que donne l'informatique. Nous pouvons, en effet, y trouver un accès par ordre alphabétique des entrées verbales comme dans le dictionnaire, mais aussi en plus un accès par ensembles de configurations aspectuelles, par nombre de sens de chacun des verbes, par foncteur, par rection syntaxique, par classe sémantique, par type de contrôle, un accès pour les homographes, pour les formes réfléchies, pour les formes réciproques, …

## 3. Développements ultérieurs potentiels

### 3.1. Développements souhaitables dans le cadre de ÚFAL

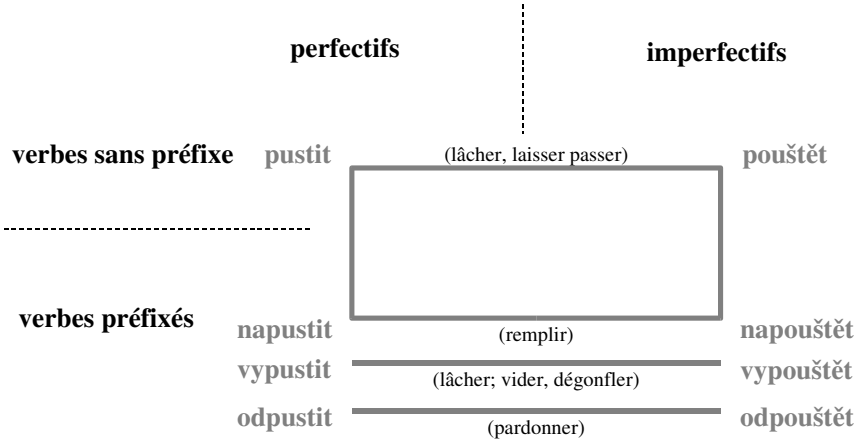
**3.1.1.** Outil tout à fait remarquable quelle que soit la forme considérée, Vallex requiert à notre avis encore au moins quelques développements.

Actuellement, les entrées verbales ne sont constituées que des ensembles aspectuels présentant le même radical, c'est-à-dire pour ce qui nous intéresse la même combinaison préfixe(s)-racine. Il convient de savoir qu'en terme de formation morphologique de l'aspect, on peut distinguer quatre groupes, de volumes très inégaux :

– les paires aspectuelles (2 ou 3) formées sur des verbes différents, par exemple brát, vzít (prendre) - traitées dans Vallex.

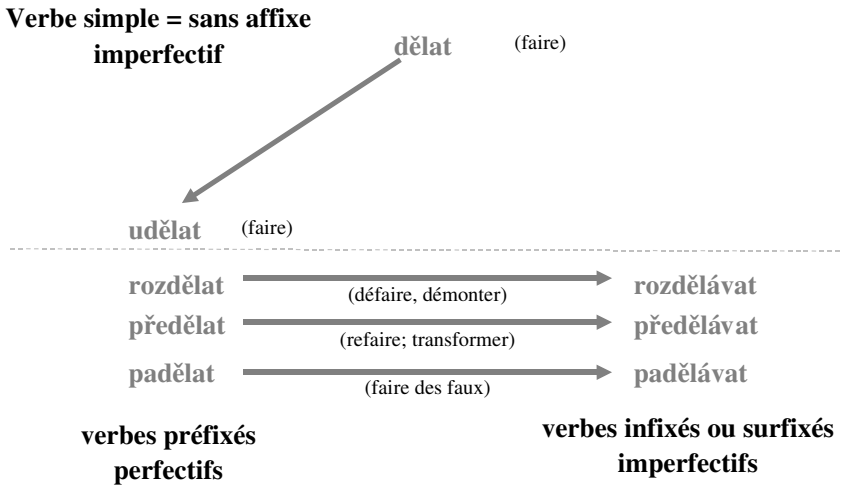
– les verbes bi-aspectuels, généralement des emprunts à des verbes étrangers qui sont regroupés dans une (sous-)classe en -ovat – catégorie traitée dans Vallex.

– la formation aspectuelle « en carré » pour les verbes perfectifs simples (c'est-à-dire non préfixés), par exemple :




Cette catégorie est traitée dans Vallex, au prix d'un renvoi d'une entrée perfective « koupit », « pustit » vers une entrée commune classée alphabétiquement suivant la forme imperfective : « kupovat, koupit » (acheter), « pouštět, pustit » (lâcher).

– la formation aspectuelle « en triangle », très majoritaire (vraisemblablement au moins 90 % des verbes sont concernés) :




Cette construction part de la forme simple imperfective, c'est-à-dire sans préfixe ni suffixe en dehors du morphème d'infinitif. Le verbe simple, imperfectif, forme son perfectif correspondant à l'aide d'un préfixe dit « zéro » parce que vide de sémantique. L'un des problèmes délicats (par exemple pour l'apprentissage de la langue) est que chaque verbe a un préfixe déterminé pour cet usage.

Les verbes préfixés sont perfectifs, c'est-à-dire que leur présent morphologique a une valeur de futur sémantique. Ils ont un sens différent de celui du verbe simple et ne peuvent donc pas lui correspondre en terme d'aspect. Ils ont besoin d'un imperfectif exprimant le même sens, mais dont le présent morphologique sera un présent sémantique. Cette valeur est obtenue ici par la présence d'un infixe d'itération qui sert aussi à la formation de l'imperfectif. Ces valeurs ayant même combinaison préfixe – racine, « horizontales », sont consignées dans Vallex.

Par contre, la paire « verticale » n'existe pas dans Vallex. Ainsi, à l'entrée « dělat » ne trouve-t-on que l'itératif « dělávat ». La forme « udělat » est isolée, ce qui n'est pas logique. De même, « děkovat » et « poděkovat » sont séparés et non reliés par un renvoi.

L'une des vertus de Vallex est son usage possible pour l'apprentissage du tchèque. A cette fin, le traitement de la paire « verticale » est absolument nécessaire pour savoir quels sont les verbes qui se correspondent. Quelle que soit la langue slave, cette correspondance n'est pas évidente, même pour les autochtones. C'est pourquoi, à plus forte raison, sa notation dans Vallex nous semble indispensable.

**3.1.2.** Une autre caractéristique de Vallex est liée au traitement automatique du tchèque. En effet, Vallex fournit des données nécessaires à l'analyse (ou la génération) automatique du cadre

verbal et donc à une analyse / génération syntaxique et sémantique de la proposition. Dans cette visée, il nous semble nécessaire de rechercher dès ce niveau de présentation des données le maximum d'automatismes. Il est vraisemblable qu'un certain nombre de transformations puissent être exprimées par des systèmes de règles là où il y a pour le moment duplication du cadre verbal pour des sens qui ne sont pas différents, mais liés l'un à l'autre par transformation de structure. Nous avons à l'esprit des exemples tels que celui de « žít 2 / žnout » (faucher) dont nous avons donné des extraits Vallex plus haut :

« žal trávu na palouku » : il a fauché l'herbe du pré (m.à.m sur le pré)

« žal palouk » : il a fauché le pré

Personnellement, nous donnerions les cadres verbaux dans cet ordre (2 – 1 de Vallex) car faucher le pré, c'est toujours faucher le « x » qui se trouve dessus même si ce « x » n'est pas exprimé (herbe, trèfle, luzerne, …). Lorsqu'il y a omission de « x » PAT, le LOC (ici le pré) subit une translation vers la valeur PAT. Cette transformation est-elle calculable ou plus exactement est-elle transposable dans le formalisme de Vallex ? La Description Générative Fonctionnelle a depuis longtemps adopté la translation des actants situés au-delà du Patient dans un point de vue mêlant les aspects syntaxiques et les aspects sémantiques.

**3.1.3.** Dans le même ordre d'idée et pour éviter de construire également les cadres verbaux de lexèmes dérivés de verbes, le calcul systématique d'une catégorie lexicale à une autre serait-il envisageable, possible ? Nous pensons particulièrement aux cadres verbaux des substantifs verbaux ou des adjectifs issus de participes verbaux. Seraient-ils déductibles des cadres (des « unités lexicales ») du verbe correspondant ?

### 3.2. Développements possibles à l'extérieur de ÚFAL

Deux situations nous semblent possibles : la réalisation d'autres Vallex pour des langues autres que le tchèque et l'intégration de Vallex (tchèque ou autre langue pour laquelle pourrait être réalisé un dictionnaire de valences) dans des projets de dictionnaires ou de didactique du tchèque.

**3.2.1.** Pour le premier point, notre équipe envisage des études sur le cadre verbal en slovaque (Diana Lemay et nous-même) et en albanais (Klara Lagji).

**3.2.2.** En relation avec ÚFAL, l'exploitation de Vallex comme composante syntaxico-sémantique de lexiques ou de dictionnaires tchèque - français nous semble nécessaire pour de tels projets. L'existence d'un dictionnaire français - tchèque ayant une composante Vallex pour les verbes nous semble encore plus nécessaire pour les besoins de Francophones souhaitant :

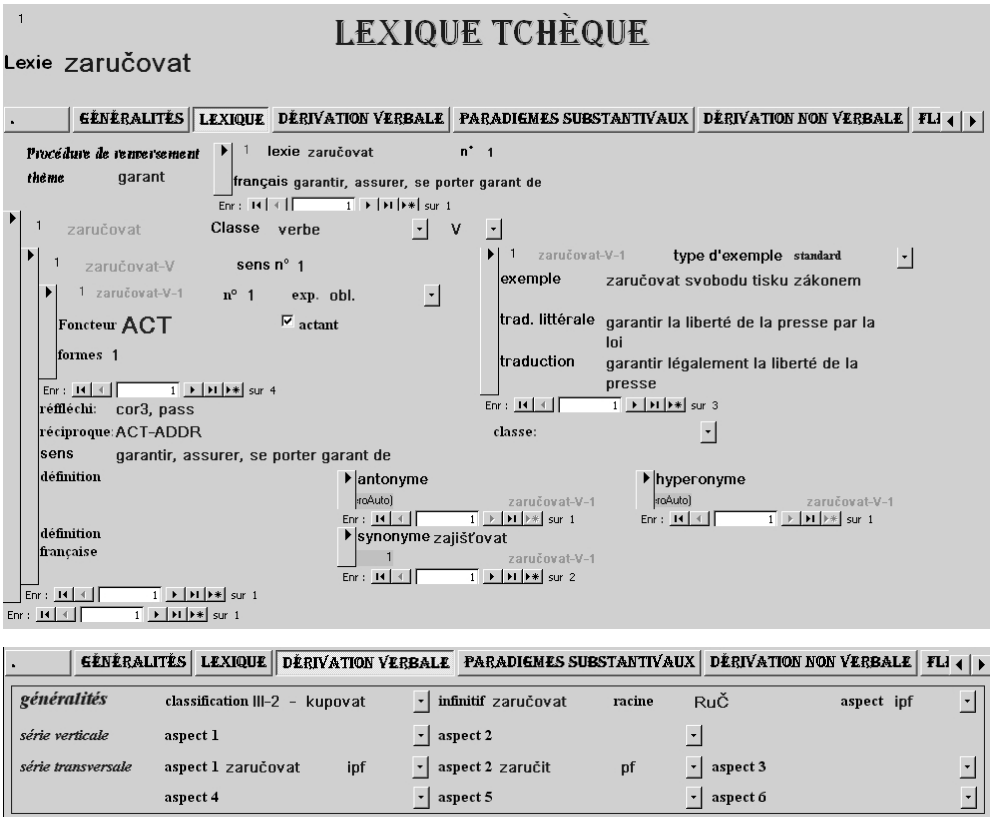
– apprendre le tchèque

– traduire en tchèque

– rédiger en tchèque.

C'est pourquoi nous définissons un projet de base de données tchèque – français englobant des informations Vallex qui sera par la suite renversée pour préparer un dictionnaire français

– tchèque avec la même masse lexicale.







Vallex peut également donner la matière à la constitution d'exercices sur serveur pour les apprenants du tchèque. Dans le cadre de la réalisation d'une méthode d'apprentissage du tchèque, nous ne négligerons pas cette possibilité. Cette méthode est envisagée à la suite de la méthode de slovaque réalisée dans le cadre du projet ALPCU (Lingua II) dont les auteurs sont Elena Baranová, Vlasta Křečková, Diana Lemay et nous-même.

En conclusion, nous soulignerons le fait que Vallex, heureux résultat d'une longue recherche, pourra à son tour donner lieu à d'autres développements en direction de la traduction, de la réalisation de lexiques et de dictionnaires et surtout de la didactique du tchèque.

## Bibliographie chronologique de l'équipe ÚFAL

1961 – Sgall, P. : "*Functional Sentence Perspective in a Generative Description*". Prague Studies in Mathematical Linguistics, no. 2.

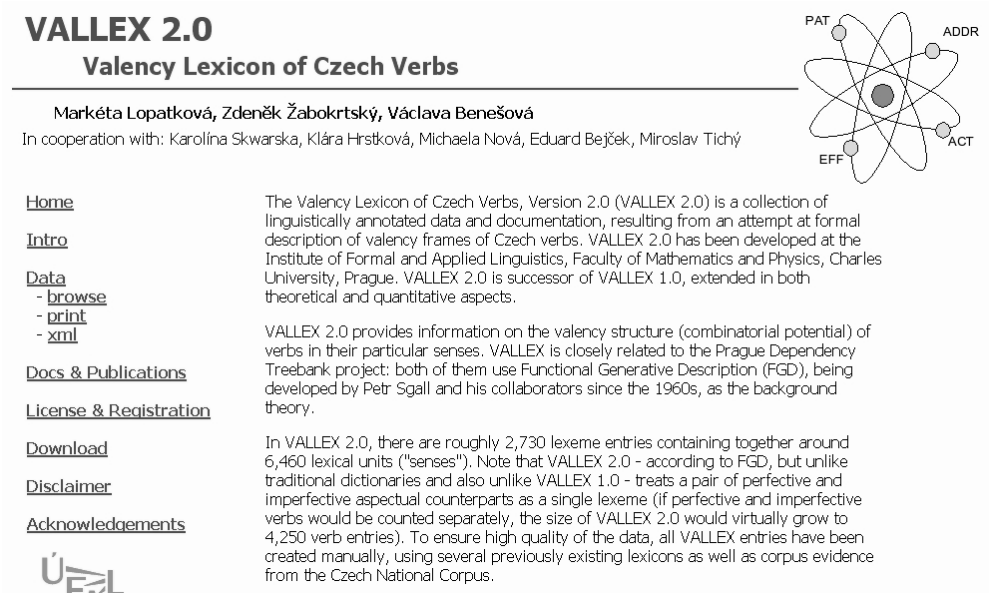
1967 – Sgall, P. : « *Generativní popis jazyka a česká deklinace* ». Academia, Prague.

1973 – Benešová, J., Sgall, P. : "*Remarks on the Topic/Comment Articulation*". Prague Bulletin of Mathematical Linguistics no. 19.
1974–1975 – Panevová, J. : "*On Verbal Frames in Functional Generative Description*". Prague Bulletin of Mathematical Linguistics no. 22 & 23.
1976 – Panevová, J., Sgall, P. : "*Verbal Frames and Free Adverbials*". International Revue of Slavic Linguistics no. 1.
1977 – Panevová, J. : "*Verbal Frames Revisited*". Prague Bulletin of Mathematical Linguistics no. 28.
1977–1978 – Sgall, P., Hajičová, E. "*Focus on Focus*". The Prague Bulletin of Mathematical Linguistics no. 28 & 29.
1979 – Sgall, P. : "*Towards a Definition of Focus and Topic*". The Prague Bulletin of Mathematical Linguistics no. 31 & 32.
1980 – Sgall, P. : "*Case and Meaning*". The Prague Bulletin of Mathematical Linguistics no. 33.
1980 – Sgall, P. : "*A Dependency-Based Specification of Topic and Focus. Formal Account*". SMIL 1-2. Prague.
1984 – Sgall, P. (ed.) : *Contributions to Functional Syntax, Semantics and Language Comprehension*. Academia, Prague.
1984 – Hajičová, E., Panevová, J. : "*Elementary and Complex Units of the Tectogrammatical Level*". Prague Bulletin of Mathematical Linguistics no. 42, Prague.
1986 – Hajičová, E., Sgall, P. : "*The Ordering Principle*". Prague Bulletin of Mathematical Linguistics no. 45, Prague.
1986 – Sgall, P., Hajičová, E., Panevová, J. : *The Meaning of the Sentence in its Semantic and Pragmatics Aspects*. Academia & Reidel.
1990 – Panevová, J., Sgall, P. : "*Dependency Syntax, its Problems and Advantages*". Prague Series of Mathematical Linguistics no. 10.
1996 – Panevová, J. : "*More Remarks on Control*". Prague Linguistic Circle Papers, John Benjamin.
2003 – Bojar, O. : "*Towards Automatic Extraction of Verb Frames*". Prague Bulletin of Mathematical Linguistics no. 79-80.
2003 – Lopatková, M. : "*Valency in the Prague Dependency Treebank : Building the Valency Lexicon*". Prague Bulletin of Mathematical Linguistics no. 79-80.
2006 – Kolářová, V. : "*Valency of deverbal nouns in Czech*". Prague Bulletin of Mathematical Linguistics no. 86.
2006 – Lopatková, M., Žabokrtský, Z., Benešová, V. : "*Valency lexicon of czech verbs VALLEX 2.0* ". Technical Report 34, UFAL MFF UK, Prague.
2007 – Žabokrtský, Z., Lopatková, M. : « *Valency Information in VALLEX 2.0. Logical Structure of the Lexicon* ». The Prague Bulletin of Mathematical Linguistics, N° 87.
2008 – Lopatková, M., Žabokrtský, Z., Kettnerová, V. : "*Valenční slovník českých sloves*". Karolinum, Prague.
2008 – Panevová, J. : "*VALLEX 2.0 – Valency Lexicon of Czech Verbes and Its Theoretical Background* ". Conférence donnée au LaLIC (Université de Paris-Sorbonne et INALCO), Paris.

**Bibliographie extérieure à l'équipe ÚFAL**

Daneš, F. (1971) : "*On Linguistic Strata*". Travaux de Linguistique de Prague no. 4.
Daneš, F. (ed.) (1974) : *Papers on Functional Sentence Perspective*. Academia, Prague.
Firbas, J. (1971) : "*On the Concept of Communicative Dynamism in the Theory of Functional Sentence Perspective*". Sborník prací filosofické fakulty brněnské university. Brno.
Mathesius, V. (1936) : "*On some Problems of the Systematic Analysis of Grammar*". Travaux du Cercle Linguistique de Prague, no. 6.
Tesnière, L. (1959) : "*Eléments de syntaxe structurale*", Paris.

**Annexe** : l'interface HTML de VALLEX



PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 89 JUNE 2008 107–108

# BOOK NOTICES

## Argument Realization

Beth Levin and Malka Rappaport Hovav

Cambridge University Press, New York, 2005, ISBN 978-0-521-66331-1, 286 pp.

Notice by Václava Kettnerová

This book provides an extensive survey of current theories of realization of verb arguments. Assuming a close relationship between meaning and syntactic behavior of verbs, the following issues concerning this linguistic phenomenon are identified as crucial: grammatically relevant facets of verb meaning, semantic classification of verbs, regular changes in argument structure, and the link between argument structure and its surface syntactic realization.

Firstly, basic notions connected with argument realization, as semantic roles, event conceptualizations, and thematic hierarchies among arguments are widely debated within the scope of individual theories, which are explored especially with respect to how efficiently they face the above mentioned challenges. Then algorithms of mapping from lexical semantic representation to syntax are discussed in great detail. The last chapter is devoted to a topical question of multiple argument realizations – regular variations in argument structure.

Explaining the main tenet and core terms of each theory in a comprehensive and detailed way and accompanied with abundant bibliographic references, the book may serve as a useful starting point for students and researchers in both syntax and semantics.

## Mathematical Linguistics

Andras Kornai

Springer-Verlag, London, 2008, ISBN 978-1-84628-985-9, 290 pp.

Notice by Pavel Schlesinger

This book introduces mathematical foundations of linguistics. The book mentions all common and important parts in the field of mathematical (computational) linguistics and it is organized into chapters called The elements, Phonology, Morphology, Syntax, Semantics, Complexity, Linguistic pattern recognition, Speech and handwriting. Within each chapter the reader can find a mathematical description of the topic of the chapter, based on eg. Automata and Language theory, Probability theory (esp. Hidden Markov Models), Machine learning concept or Information theory.

The author intended the book to be accessible to anyone with sufficient general mathematical maturity (graduate or advanced udndergraduate). He has tried to present the text in a way that there is no prior acuaitance with lingustics or languages assumed on the part of the reader. The author has designed his book to be suitable for an aggressively paced one-semester course or a more leisurely paced two-semester course., and for that purpose, there are many exercices throughout the whole book. In addition, each chapter ends with a section of futher reading.

The book ranks among the previous introductions to computational lingusitics such as *Chris Manning and Hinrich Schütze: Foundations of Statistical Natural Language Processing (MIT Press, 1999), Barbara H. Partee, Alice ter Meulen, Robert E. Wall: Mathematical Methods in Linguistics (Kluwer Academic Publishers, 1993)* or *Frederick Jelinek: Statistical Methods for Speech Recognition (MIT Press, 1999)* and one can only agree with Aravind Joshi's assessment in the official Springer notice that the book is well written and that it provides a rather non-standard but very attractive approach to mathematical linguistics.

PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 89 JUNE 2008

# INSTRUCTIONS FOR AUTHORS

Manuscripts are welcome provided that they have not yet been published elsewhere and that they bring some interesting and new insights contributing to the broad field of computational linguistics in any of its aspects, or of linguistic theory. The submitted articles may be:

- long articles with completed, wide-impact research results both theoretical and practical, and/or new formalisms for linguistic analysis and their implementation and application on linguistic data sets, or
- short or long articles that are abstracts or extracts of Master's and PhD thesis, with the most intersting and/or promising results described. Also
- short or long articles looking forward that base their views on proper and deep analysis of the current situation in various subjects within the field are invited, as well as
- short articles about current advanced research of both theoretical and applied nature, with very specific (and perhaps narrow, but well-defined) target goal in all areas of language and speech processing, to give the opportunity to junior researchers to publish as soon as possible;
- short articles that contain contraversing, polemic or otherwise unusual views, supported but some experimental evidence but not necessarily evaluated in the usual sense are also welcome.

The recommended length of long article is 12–30 pages and of short paper is 6-15 pages.

The copyright of papers accepted for publication remains with the author. The editors reserve the right to make editorial revisions but these revisions and changes have to be approved by the author(s). Book reviews and short book notices are also appreciated.

The manuscripts are reviewed by 2 independent reviewers, at least one of them being a member of the international Editorial Board.

Authors receive two copies of the relevant issue of the PBML together with 10 offprints of their article.

The guidelines for the technical shape of the contributions are found on the web site `http://ufal.mff.cuni.cz/pbml.html`. If there are any technical problems, please contact the editorial staff at `pbml@ufal.mff.cuni.cz`.

PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 89 JUNE 2008

# LIST OF AUTHORS

**Silvie Cinková**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
cinkova@ufal.mff.cuni.cz

**Jan Hajič**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
hajic@ufal.mff.cuni.cz

**Eva Hajičová**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
hajicova@ufal.mff.cuni.cz

**Jirka Hana**
Center for Human Resource Research
The Ohio State University
921 Chatham Lane, Suite 100
Columbus, OH 43221, USA
hanaZZ.1ZZ@osu.edu

**Barbora Vidová Hladká**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
hladka@ufal.mff.cuni.cz

**Jaroslava Hlaváčová**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
hlavacova@ufal.mff.cuni.cz

**Václava Kettnerová**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
kettnerova@ufal.mff.cuni.cz

**Jiří Mírovský**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
mirovsky@ufal.mff.cuni.cz

**Jarmila Panevová**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
panevova@ufal.mff.cuni.cz

**Patrice Pognan**
CERTAL
Institut National des Langues et
Civilisations Orientales
73, rue Broca
75013 Paris, France
mcertal@wanadoo.fr

**Jan Raab**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
raab@ufal.mff.cuni.cz

**Petr Sgall**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
sgall@ufal.mff.cuni.cz

**Pavel Schlesinger**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
schlesinger@ufal.mff.cuni.cz

**Drahomíra "johanka" Spoustová**
Institute of Formal and Applied Linguistics
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic
johanka@ucw.cz