DESIGN AND DEVELOPMENT OF TOOLS FOR THE CONSTRUCTION OF THE MONOLINGUAL WORDNETS FOR EACH OF THE PARTICIPATING BALKAN LANGUAGES

Deliverable D.3.1, WP3, BalkaNet, IST-2000-29388

BalkaNet

Identification Number	IST-2000-29388
Type	Report-Document
Title	Design and development of tools for the construction of the monolingual WordNets for each of the participating Balkan languages
Status	Final
Deliverable	D.3.1
WP contributing to the deliverable	WP3
Task	T.3.1
Period covered	January-June 2002
Date	June 2002
Version	1
Status	Confidential
Number of pages	71
WP/Task Responsible	UOA
Other Contributors	DBLAB, CTI, RACAI, DCMB, SABANCI, FIMU, PU

Authors	{Eleni Galiotou, Maria Grigoriadou, Anastasia Charcharidou, Evangelos Papakitsos, Stathis Selimis} **UOA** {Sofia Stamou} **DBLAB** {Cvetana Krstev, Gordana Pavlovic-Lazetic, Ivan Obradovic, Dusko Vitas} **MATF** {Ozlem Cetinoglu} **SABANCI** {Dan Tufis} **RACAI** {Karel Pala, Tomas Pavelek, Pavel Smrz} **FI MU** {Svetla Koeva} **DCMB** {George Totkov} **PU**
EC Project Officer	Erwin Valentini
Project Coordinator	Prof. Dimitris Christodoulakis Director, DBLAB Computer Engineering and Informatics Department Patras University GR-265 00 Greece Phone: +30 610 960385 Fax: +30 610 960438 e-mail: dxri@cti.gr
Keywords	Tools, Language Resources, Corpora, Electronic dictionaries
Actual Distribution	Project Consortium, Project Officer, EC

Abstract	This report describes the design and architecture of the tools that have been developed for the construction of each monolingual WordNet of the participating Balkan languages. During the task, members of the consortium have decided on the tools which were used in the implementation of the monolingual WordNets and they defined their design and architecture. In addition, a detailed recording of available tools that were applicable to the construction of each monolingual WordNet took place. Each participant has developed tools for his own language following the specifications and the methodology for the development of monolingual WordNets which were set in workpackage WP2 (T.2.1 and T.2.2) and the information contained in the available lexical resources of each language. Partners who already had a WordNet for their language have improved their existing tools and have built new ones where necessary. They also shared their knowledge and expertise with the other members of the consortium. In addition to this report, each participant delivers the tools developed for his own language along with the available lexical resources.
Status of abstract	Complete
Send on	June 2002

TABLE OF CONTENTS

TABLE OF CONTENTS	**5**
EXECUTIVE SUMMARY	**6**
INTRODUCTION	**9**
1. TOOLS AND RESOURCES FOR THE BULGARIAN WORDNET	**10**
1.1 Existing Language Resources	10
1.1.1 Electronic Dictionaries	10
1.1.2 Corpora	14
1.2 Development of Language Resources	15
1.3 The SysLiR Project Technical Documentation	15
1.4 Tools	18
2. TOOLS AND RESOURCES FOR THE CZECH WORDNET	**20**
2.1 Language Resources	20
2.2 Tools	21
2.3 VisDic	22
3. TOOLS AND RESOURCES FOR THE GREEK WORDNET	**25**
3.1 Language Resources for Greek	25
3.2 Tools	25
3.2.1 Existing Tools for the Extraction of Semantic Information	25
3.2.2 Morphological Processing Tools	28
3.2.3 Tools developed for the Extraction and Processing of Linguistic Information	29
3.3 Contribution of Tools towards the Base Concept Selection Process	35
References	37
4. TOOLS AND RESOURCES FOR THE ROMANIAN WORDNET	**39**
4.1 Language Resources for Romanian	39
4.1.1 Electronic Dictionaries	39
4.1.2 Corpora	43
4.2 Tools	43
4.2.1 Tools developed exclusively for the purpose of building the Romanian WordNet	44
References	47
5. TOOLS AND RESOURCES FOR THE SERBIAN WORDNET	**49**
5.1 Lexical Resources	49
5.2 Corpora	52
5.2.1 Corpus of contemporary Serbian	52
5.2.2 Parallel Corpora	53
5.3. Tools	53
References	55
6. TOOLS AND RESOURCES FOR THE TURKISH WORDNET	**57**
6.1 Existing Language Resources	57
6.2 Tools	60
6.3 Development of Language Resources	60
6.4 Problems encountered during the merging process of different resources	69
References	70
CONCLUSIONS AND FUTURE WORK	**71**

EXECUTIVE SUMMARY

WordNet (Fellbaum 1998, Miller et al. 1990), a lexical database with semantic relations between English words, was developed in the Cognitive Science Laboratory at the University of Princeton. Its success as a lexical resource in several computational linguistic tasks has led to the production of similar semantic lexical databases for many other languages. Following the initial design of WordNet, the EuroWordNet project (Vossen 1998) resulted in a multilingual lexical database with wordnets for eight European languages (Czech, Dutch, English, Estonian, French, German, Italian and Spanish) .

The goal of Balkanet is to develop a multilingual lexical database with semantic networks of the following languages: (Bulgarian, Czech, Greek, Romanian, Serbian and Turkish) along the general guidelines of EuroWordNet. For that purpose, each monolingual WordNet which is being developed independently will be incorporated in the BalkaNet database which in turn will be linked to EuroWordNet thus resulting in a global semantic database.

In this Balkanet framework, the deployment of computational tools for the monolingual as well as the multilingual databases has proved to be of major importance. In particular, the tools and resources used for the construction of individual WordNets for each of the participating languages had to take into account the particularities of these less-studied Balkan languages and gave significant insights as for their structure and the accessibility of data which were not widely promoted so far.
In this respect, each team has performed a recording of already available tools and resources that are useful to the development of the monolingual WordNets. Moreover, new tools were developed following the specifications and methodology set in workpackage WP2. Partners who had already a WordNet for their language have improved existing tools of have developed new ones according to the requirements of Balkanet. In addition, they shared their expertise and knowledge with other members of the consortium and tools such as VisDic developed by the Czech partner were used by other participants as well for the development of their own WordNet.

In this report, members of the Balkanet consortium describe the tools and resources that support the monolingual work at a local level. Therefore, they are tailored to the specific needs of each language and give a clear cut image not only of the work towards the construction of the semantic lexical database but of the infrastructure which is available for general Natural Language Processing tasks as well.

As for each language, the report on the monolingual work is based on the following resources and tools (either already available or developed from scratch):

- Bulgarian:

→ a. Monolingual and bilingual dictionaries such as : The Bulgarian grammatical dictionary, the Bulgarian frequency dictionary, the Bulgarian synonymy dictionary, the Bulgarian Explanatory dictionary, the semantic minimum dictionary, the English-Bulgarian and Bulgarian-English dictionary.

b. Corpora such as: A set of very large Bulgarian corpora of different genres and types of prose and poetry and a set of English-Bulgarian administrative documents.

→ A set of tools to exploit the above mentioned resources as well as the SySLiR indexing/retrieval system.

- Czech

→ a. Monolingual and bilingual dictionaries such as: The dictionary of written Czech, the dictionary of Literary Czech, the dictionary of Czech synonyms, the Czech synonymical dictionary and Thesaurus, the Valency dictionary of Czech.
b. Corpora such as: the text corpus ESO from which lists of collocations and other information were extracted and the Czech National Corpus.

→ A set of tools to exploit the above mentioned resources such as: The DIS shallow parser, a simple translating program to process a bilingual dictionary, a specialized program to create ILR and ILI, a program to compute mutual information scores for wordforms from Czech corpora, the Czech lemmatizer ajka, the I_PAR morphological database and the Polaris v1.5 tool which was later replaced by VisDic – a specialized browser and editor for WordNet-like databases implemented in XML format

- Greek

→ a. Monolingual dictionaries in electronic form such as the dictionary of Patakis Publications and the Triandafyllidis dictionary delivered by the Center of the Greek language.
b. The Greek part of the ECI corpus

→ A set of tools to exploit the above mentioned resources such as: The definitions extraction tool, the word frequency calculation tools, the synonyms and antonyms extraction tools, a tool for antonymic relations search in lemmata definitions, the "search for possible semantic relations tool", the "search for relations such as 'role-involved' tool", the extraction of POS-related information tool, the extraction of linked and compound lemmata information tool. Moreover, general purpose tools such as the M.A.S. (Morphological Analysis System) and the wordform generator were used.

- Romanian

→ a. Monolingual and bilingual dictionaries such as the Wordform Romanian dictionary, the explanatory dictionary of Romanian, the Romanian dictionary of synonyms, the Morphological Orthoepic and Orthographic dictionary, the Romanian frequency dictionary, the Romanian-English dictionary, the lexicon containing all POS defined in the MULTEXT-EAST specifications.
   b. Multilingual corpora developed within the MULTEXT-EAST and TELRI European projects and monolingual corpora such as a literary one and a journalistic one.

→ A set of tools to exploit the above mentioned resources such as: A tokenizer, a sentence aligner, a tagger, a translation equivalents extraction program, an editor for building synsets for the commonly agreed ILI concepts and an editor for gloss assignment.

- Serbian

→ a .Monolingual dictionaries in electronic form such as: the Serbian morphological electronic dictionary, the Serbian translation of the Oxford Dictionary of Computing, the Systematic dictionary of Serbo-Croatian.
b. Corpora such as: the Corpus of Contemporary Serbian and parallel corpora developed within the TELRI project.

→ A set of tools to exploit the above mentioned resources such as: A coding scheme conversion program, a program to convert the dictionary of contemporary Serbian into XML format, a fast text scanner and the INTX concordance package.

- Turkish

→ Monolingual and bilingual dictionaries in electronic form such as the TDK (Monolingual Dictionary of Turkish) and the Turkish-English bilingual dictionary as well as the synonyms database.

→ Tools to exploit the above mentioned resources such as Perl scripts in order to transfer these resources under the VisDic platform, a monolingual dictionary browser and a synset merge tool. In addition, the group has use the general tools such the WordNet browser, Periscope and VisDic for their monolingual work. Moreover, general purpose tools such as the Morphological Analyzer of Turkish and a Turkish Spelling Corrector were used.

This report is completed by the set of tools developed and used by each member of the consortium which are placed on the project's information server.

Future work on the design and development of tools will concentrate on tools for the Balkanet multilingual database which will be described in the deliverable D.3.2.

INTRODUCTION

The work reported in this deliverable is the output of the first task (T.3.1) of workpackage WP3 which deals with the design and development of tools for the construction of the monolingual WordNets for each of the languages participating in the Balkanet project (Bulgarian, Czech, Greek, Romanian, Serbian and Turkish).

In the course of the Balkanet project, we have adopted the **merge model** (Vossen 1998) so, the development of each WordNet is based on local resources with synsets and language-internal relations being developed separately and then linked to most equivalent concepts in the ILI record.
Consequently, each partner was led to develop his own tools and resources taking into account the particularities of each language.

During the first phase of T.3.1 every participant performed a detailed recording of already available language resources and tools at each site that could be useful to the construction of the respective monolingual WordNet. These tools and resources were evaluated and in certain cases adapted to the requirements of the task at hand.
The members of the consortium have also developed new tools following the specifications and the methodology set in workpackage WP2. Partners who had already a WordNet developed for their language shared their knowledge and expertise with the other members of the consortium. Moreover, they developed new tools where needed and improved the existing ones.

This report contains the catalog of tools that are being used in the development of the individual WordNets and of the available language resources for each Balkan language.
It is structured into 6 independent chapters – one for each participating language in alphabetical order (Bulgarian, Czech, Greek, Romanian, Serbian and Turkish).
Each chapter roughly contains the following information:

- A description of the available language resources (dictionaries and corpora) at each site and their contribution to the monolingual work of building the respective WordNets.
- A catalog of the tools which are developed for, or adapted to the extraction and processing of the necessary linguistic information.
- A description of the resources that have been built in the course of this task and new tools that were developed exclusively for use in the Balkanet project and follow the specifications set in previous workpackages.
- Examples of use of the particular tools and resources for the construction of the semantic network at a local level.

The deliverable D.3.1 is completed by the set of tools developed at each site and their instructions of use which are placed on the project's information server
http://dblab.upatras.gr

The tools which are developed by each partner for the purpose of building the respective local WordNet and are reported here, are the property of each participant. Some of them will be publicly available at the end of the project.

1. TOOLS AND RESOURCES FOR THE BULGARIAN WORDNET

1.1 Existing Language resources

1.1.1 Electronic dictionaries

A. Monolingual Dictionaries

- ***Bulgarian Grammatical Dictionary***

The linguistic work for the Bulgarian Grammatical Dictionary has been developed by S. Koeva (Koeva Svetla 1999, Bulgarian Grammatical Dictionary. Bulgarian language. Vol. 5. 47-52.). The electronic dictionary consists of about 80,000 lemmata and over 1,000,000 corresponding forms - basically all inflected simple words. The grammatical information included in the dictionary can be classified as follows:

- categorial - describing the basic word forms as parts of the grammatical classes linked to the parts of speech;
- paradigmatical - describing the word forms as parts of different grammatical subclasses with a common paradigm;
- grammatical - describing the word forms as parts of different grammatical types that alternate identically.

In the dictionary lexical entries are lemmas, each entry is associated with one FST (Finite State Transducer) that represents all corresponding inflected forms. For example, here are five entries:

бързо,ADV	*'fast'*
възстановителен,A5	*'recovering'*
година,N3	*'year'*
на,PREP	*'for'*
чета,V+I+T16	*'to read'*

ADV, A5, N3, PREP and V+I+T16 are names of inflectional FSTs. All words in the language that have the same set of suffixes are associated with the same inflectional FST. For instance all adjectives that alternate such as (*възстановителен*) – 'reconstructing' are associated with the FST A5 (*играл*ен,*A5* – 'playing', *волен,A5* – 'free', etc.).

The same five entries with all corresponding inflected forms are presented bellow:

бързо.ADV
възстановителен,възстановителен.A5:s
възстановителна,възстановителен.A5:sf
възстановителната,възстановителен.A5:sfd
възстановителни,възстановителен.A5:p
възстановителните,възстановителен.A5:pd
възстановителния,възстановителен.A5:smh
възстановителният,възстановителен.A5:sml
възстановително,възстановителен.A5:sn
възстановителното,възстановителен.A5:snd
година,година.N3+F:s
годината,година.N3+F:sd

години,година.N3+F:p
годините,година.N3+F:pd
годино,година.N3+F:v
на.PREP

For example the form *'възстановителната'* is associated with the lemma *'възстановителен'* which is an adjective (A) of the syntactic class 5; the form is singular (s), feminine (f), indefinite.

The dictionary can associate tokens with a lemma and linguistic information - part of speech (e.g. Noun), inflectional information (e.g. first person singular present), etc. The result of the application of the dictionary is lists of all recognized and unrecognized words.

- **Morphological dictionary of Bulgarian**

Present size - 69 236 entries (base forms) and about 1 500 000 word forms.

Part of speech	Count	Per cent
Noun	28222	40,76%
Verb	20335	29,37%
Adjective	12973	18,74%
Adverb	3035	4,38%
Preposition	66	0,10%
Conjunction	24	0,03%
Particle	57	0,08%
Pronoun	241	0,35%
Interjection	139	0,20%
Proper noun	3934	5,68%
Numeral	101	0,15%
Shorting	109	0,16%
Total	69236	100,00%

The words are divided into 231 inflectional types (proper nouns incl.), with base form recognition and automatic lemmatization (see 1.4. Tools - Bulgarian morphological processor for analysis, synthesis, and robust analysis of "unknown" words).

- ***Bulgarian Frequency Dictionaries***

Frequency Dictionaries (Koeva, S. "Automatic generation of lexeme frequency dictionaries" – For Words and Dictionaries, Lexicology and lexicography'98, Sofia, 2000, Academic Press, 109-117) are extracted automatically from corpora with a program that:

- Identifies tokens;
- Replaces the capital letters with small letters;
- Analyses statistically word forms.

The resulting dictionaries have the following format:

това,9.455529 *'this'*
с,8.797904 *'whit'*
ще,6.532752 *'will'*
има,5.210544 *'to have'*

The frequency dictionaries are different according to the corpora that they exemplify.

- ***Bulgarian Synonymy Dictionary***

The second edition of the Bulgarian synonymy dictionary was published recently – Nanov, L. and A. Nanova, Bulgarian Synonymy Dictionary, Sofia, 2000, Hejzal. For the purposes of the project we have it in electronic form. The dictionary consists of approximately 30,000 entries in alphabetical order. A great number of the headwords are polysemic, that is why each separate meaning generates a separate synonymy set. The synonymy sets approximate 40,000. The dictionary also includes over 4,000 phraseological units, which are listed at the end of the corresponding entries.

This synonymy dictionary comprises lexical synonyms. These are words belonging to one part of speech, identical or close in meaning and with different phonemic representation. Besides synonyms belonging to the modern literary Bulgarian language there are outdated, dialect, old and rare words belonging to the passive part of the language which can be found in the Bulgarian classical literature; as well as slang words common in everyday speech.

The relations of synonymy are established among words by way of their basic, secondary and metaphoric meaning. Some synonyms are semantically equivalent - others differ in terms of register, frequency of usage and slight semantic differences. A special attention is paid to the stylistic and expressive characteristics of the synonyms, which affects their interchangeability.

The relations of synonymy in the dictionary are presented in synonymy sets. A synonymy set consists of words united by a common meaning. In principle the first word in the set is the dominant synonym, which represents the common concept in the most precise way. It usually belongs to the neutral part of the vocabulary and participates in the set with its basic meaning. The rest of the synonyms are arranged according to the degree of semantic closeness and register. Generally, the neutral synonyms are placed first followed by the words belonging to the literary language and colloquial speech, poetic language as well as to the vulgar and dialect language and finally there are antiquated, old and slang words.

Bulgarian synonymy dictionary – 24 699 entries, available in electronic form.

Part of speech	Count	Percent
Noun	10124	40,99%
Verb	7931	32,11%
Adjective	5651	22,88%
Adverb	867	3,51%
Preposition	55	0,22%
Conjunction	17	0,07%
Particle	14	0,06%
Pronoun	12	0,05%
Interjection	11	0,04%
Proper noun	11	0,04%
Numeral	6	0,02%
Total	24699	99,99%

Every word belongs to 1.4 synsets (average). Every word has 3.4 word synonyms and 0.2 phrase synonyms (average).An automated approach for improving of conventional (non-computer based) synonym dictionaries is developed. The algorithm for forming of appropriate synsets (inc. the discovering new synsets) is realised and experimented.

The improving procedures and semi-automated extraction on synsets are applied on the Bulgarian Synonymy Dictionary.

- ***Semantic Minimum Dictionary***

The Semantic Minimum Dictionary is written by I. Kasabov and published in 1992 (Kasabov, I. Semantic Minimum Dictionary, Sofia, Sofia University Press). It is provided in electronic form by the author. The dictionary consists of about 850 words and their meanings that represent the necessary and sufficient units, which form the core of the whole Bulgarian lexical system. The words are organised on the basis of two complementary principles - by way of alphabetical-explanatory and lexico-semantic fields. The explanatory definitions are implemented with the help only of the resources of the dictionary - that is using the same 850 words. The two parts of the dictionary are related by referring indexes. This is a semantic dictionary not only because it gives definitions of the words it includes, but also because it is constructed completely according to semantic principles.

The dictionary represents the meanings and the words that belong to the central part of the Bulgarian vocabulary in a system and indicates all existing semantic relations among them. The most important characteristic of the dictionary is that the principles of compilation are founded on a specific theory of the semantics of the word and the relations of the words in the dictionary with other units and with their corresponding semantic fields.

B. Bilingual dictionaries

- ***English-Bulgarian Dictionary***

The English-Bulgarian Dictionary (in electronic form) consists of 52,434 English headwords and corresponding Bulgarian (one or more) translation equivalents. The dictionary provides also information for part of speech and transcription. The Bulgarian contributors of LINUX spread the English-Bulgarian Dictionary.

- ***Bulgarian-English Dictionary***

The Bulgarian-English Dictionary (in electronic form) consists of 23,949 Bulgarian headwords and corresponding English (one or more) translation equivalents. The Bulgarian contributors of LINUX spread the English-Bulgarian Dictionary.

The dictionaries are corrected and edited (using the morphological processor). At this time the electronic English-Bulgarian Dictionary consists of 58 000 English headwords and corresponding Bulgarian (one or more) translation equivalents and 42 500 hesdwords for the Bulgarian-English Dictionary. The semi-automated extraction of Bulgarian synsets will be applied (using WordNet synsets, bilingual and Bulgarian synonymy dictionaries).

C. Explanatory Dictionary

The Bulgarian explanatory dictionary (in electronic form) consists of 52 434 Bulgarian headwords and corresponding Bulgarian (one or more) explanatory notes. The dictionary provides also information for part of speech, grammatical features,

meaning of the word (Bulgarian glosses), stylistic and expressive characteristics, etc. The dictionary is corrected and edited (using the morphological processor).

1.1.2 Corpora

A. Monolingual

- ***Very large Bulgarian corpora***

The corpora consist of approximately 50,000 words extracted from texts published mainly in electronic form (some texts are scanned). Bulgarian authors have written predominantly the texts included in the corpora.

Bulgarian texts have been selected from different genres and types of prose and poetry. The range of texts in prose covers periodicals, fiction, science fiction, administrative documentation, and scientific texts. The approximate proportion of texts in corpora is 10% literary texts, 60% journalistic texts, and 30% administrative texts.

B. Bilingual

- ***English-Bulgarian translated administrative documents***

We have at our disposal thanks to Bulgarian Ministry of Finances approximately 22 MB Bulgarian texts that are translation equivalents of the administrative documents of European Community. The corresponding texts in English are accessible by the Internet.

1.2 Development of Language resources

- ***Structured Bulgarian corpora***

The development of Bulgarian structured linguistic corpora has been one of the stages of the project BalkaNet. The corpora have been created in the framework of the existing similar corpora in Brown University.

The corpora consist of 1,000,805 words extracted from texts published mainly in electronic form. An important requirement, which has been strictly observed in compiling the corpora is that, the texts have been written by Bulgarian authors. Some exceptions, however, have been made: the extracts from the genres of the love story and the western are taken from foreign language sources translated into Bulgarian because of the lack of original Bulgarian texts in these genres.

It has been decided that the corpora should be divided into 500 text units - approximately 2000 words each, in this way sentence boundaries have been preserved. The majority of texts consist of more than 2000 words and only a small number of less than 2000.

All samples have been selected from different genres and types of prose. Poetic texts have not been included as the characteristics of poetic language are of different type and introduce a different range of linguistic problems. Each text starts at the beginning of a sentence, but not necessary at the beginning of a paragraph or a bigger textual unit. Extracts from initial, middle and final parts of texts of a given genre have been included in the corpus. The range of texts covers periodicals, fiction, science fiction and administrative documentation.

After the texts have been compiled they have been processed in the needed format and have been incorporated in the corpus. There have been a lot of spelling and punctuation mistakes. Some of them are obviously due to technical reasons: in most cases, however, there has been an evident lack of knowledge of the grammatical rules of Bulgarian. This holds in the greatest extent for the periodicals. These mistakes could not have been ignored as the corpus is processed by computer programs, which cannot register misspelled words or grammatical mistakes. For these reasons it has been necessary for us to carry out one more stage in our work - that is the correction and editing of the whole corpus.

The texts were sampled from 15 different text categories according the model of Brown corpora. The number of texts in each category varies:

A. PRESS: REPORTAGE (44 texts)
B. PRESS: EDITORIAL (27 texts)
C. PRESS: REVIEWS (16 texts)
D. RELIGION (17 texts)
E. SKILL AND HOBBIES (36 texts)
F. POPULAR LORE (47 texts)
G. BELLES-LETTRES (76 texts)
H. MISCELLANEOUS: GOVERNMENT & HOUSE ORGANS (31 texts)
J. LEARNED (80 texts)
K: FICTION: GENERAL (29 texts)
L: FICTION: MYSTERY (24 texts)
M: FICTION: SCIENCE (6 texts)
N: FICTION: ADVENTURE (29 texts)
P.FICTION: ROMANCE (29 texts)
R. HUMOR (9 texts)

1.3 The SysLiR Project : Technical description

The SysLiR project is an indexing/query-based searching system constructed mainly for linguistic researches. The need for very fast response on queries has led us to create structures optimized for searching using various parameters.

1.3.1. Indexing Tool

- Structures definitions

- "warray" and "darray"

```
char *warray;
char *darray;
```

These arrays are keeping strings of words and corresponding file names of documents. All strings are separated by '\0' character. Indexes in these arrays are used as words and document identifiers.

- "occurrence_index" structure.

```
typedef struct occurrence_index{
        int doc_ID;
        int num_para;
```

```
        int num_sent;
        int position;
    } occurrence_index_t;
```

This is the base structure holding information about any given word that has appeared in the whole documents corpus. The "doc_ID" is index in warray, pointing to the start of corresponding document file name string. "num_para" is an integer which shows the number of paragraph in which the word occurs. Analogically, "num_sent" and "position" represent the number of sentence and position from the beginning of the document. Words are sorted in blocks. Each block contains all the appearances in corpus of given word.

- "inv_index" structure.

```
    typedef struct inv_index{
        int word_index;
        int count;
        occurrence_index_t *occurrence;
    } inv_index_t;
```

The inverted index ("inv_index") holds whole unique appearance of words. "word_index" is an index in warray for a given word. "count" is a total count of that word in the whole corpus. "occurrence" points to the beginning of a block in "occurrence_index" where all appearances for a given word are listed. The size of that block is equal to "count".

- "lexicon_struct" structure.

```
    typedef struct lexicon_struct{
        int word_ID;
        occurrence_index_t *word_occurrence;
    } lexicon_t;
```

The lexicon is a linking structure for the document and its words. "word_ID" is an index in "inv_array" for a given word. "word_occurrence" lists a number of times a word appears in "occurrence_index". In fact, "lexicon" is words listed in same order of occurrence as they are processed.

- "ord_index" structure.

```
typedef struct ord_index{
    int doc_ID;
    int count;
    lexicon_t *word;
} ord_index_t;
```

Ordinal index ("ord_index") keeps all documents of corpus. "doc_ID" is an index in darray, pointing to the beginning of document file name string. "count" is a total count of words in the document. "word" points to a block in lexicon where all the words of that document are listed. The block is as long as "count" points.

- "hash_table" structure.

```
typedef struct hash_table{
        int array_index;
        int ID;
} hash_t;
```

Lastly, "hash_table" structure keeps a record of all words or documents in their hash tables. "array_index" is an index in warray or darray for a given word or document. "ID" is an index in "inv_array" or "ord_array" pointing the exact location of a word or corresponding document.

- Functions description.

- "main" function.

The "main" function briefly follows the algorithm of indexing. First, open the configuration file and for all files listed there process every word. Each word is added to "inv_array" (add_word_to_inv function) and each document is added to "ord_array" (add_doc_to_ord function). After that configuration file is rewinded. The processing of the files starts from the very beginning. Second, for the "lexicon_array" and "occur_index_array" memory is allocated in amounts corresponding to the count of words encountered in the first pass. The values for each record are set (add_and_refresh function). All tables are stored in files (store_tables function).

- "hash" function.

```
int hash(char, unsigned int);
```

Every character from input string and C1 constant (defined in defines.h) are added, as integers, to a sum. This sum is truncated up to size of hash table and returned as result.

- "search_ht" function.

```
int search_ht( hash_t **, char *, char *, unsigned int);
```

This function returns a hash key of incoming string or '-1' if that string is not in hash table. String is translated to an integer by the "hash" function. Collisions are resolved by adding a special constant C3 (defined in defines.h).

- "rehash" function.

```
unsigned int rehash(hash_t ***, char*, unsigned int);
```

This function makes rehash when hash table is filled up to a FILL_COEF (defined in defines.h). New array is allocated and all pointers to hash_struct are copied in new array using new hash values.

- "ins_arrayw" function.

    int ins_arrayw(char *, char **);

This function inserts a string to "warray". Returned value is an index of just inserted string.

- "ins_arrayd" function.

    int ins_arrayd(char *, char **);

Analogically to ins_arrayw, but inserts string to "darray".

- "add_doc_to_ord" function.

    void add_doc_to_ord(char *, int *);

This function is used for inserting a document into "ord_array". "ord_array" is resized if needed.

- "add_word_to_inv" function.

    void add_word_to_inv(char *, int *);

This function is used for inserting a document into "inv_array". "inv_array" is resized if needed.

- "add_and_refresh" function.

    void add_and_refresh(int, int, int, int, int);

This function is used for updating records in "ord_array" and "inv_array" and adding values in "lexicon_array" and "occurrence_array".

- "store_tables" function.

    void store_tables(unsigned int, unsigned int);

This function is used for outputting all tables to external files.
Each file contains on the first line the size of the table and the tab separated values of tables, one cell per line. Exception is files for "darray" and "warray", where strings are dumped separated by '\0' character and no '\n'.

1.4 Tools

The following tools have been developed and will be used in the course of building Bulgarian WordNet:

- **pre-processing tools** for Bulgarian texts (part of speech tagger, tokeniser, sentence splitter and paragraph splitter, procedure for clause extraction, noun phrase grammar, procedures for anaphora resolution, procedure for heading identification, etc.)
- **Bulgarian morphological processor** (Bulmorph) (morphological analysis and synthesis, robust analysis of unknown words). The Bulmorph is being distributed by ELDA (European Language resources Distribution Agency)
- **Bulgarian text editor** (spell-checking, detecting some wrong syntactic constructions, etc.)
- **system for extraction and improving synsets** (using WordNet 1.6, bilingual and synonymy dictionaries)
- **system for extraction of metaconcepts** (in experimental stage). This system uses as input connected text and a model of government of one or more verbs and extracts the semantics of some words in the texts.

2. TOOLS AND RESOURCES FOR THE CZECH WORDNET

2.1 Language Resources

The following resources have been used in the course of building Czech WordNet:

- ***Dictionary of Written Czech* (*SSJC*)**, by B. Havranek et al, Academia, Praha 1960-61,
  present size: approx. 192 000 entries, in electronic form, XML format, with automatic
  lemmatization, word form recognition and basic word derivation. *SSJC* is being used for
  building Czech WordNet, though in many cases it lacks reliable and consistent information
  about the word senses and collocations.

- ***Dictionary of Literary Czech (SSC)*,** by J. Filipec et al, Academia, Praha 1994,
  present size: approx. 50 000 entries, in electronic form, XML format, with automatic
  lemmatization, word form recognition and basic word derivation. It is derived from *SSJC*
  as it smaller and newer version.

- **Version 2.0 of bilingual *Lingea Lexicon***: Czech-English-Czech, by Lingea Ltd., Brno 1997-98, present size: approx. 125 000 entries, in electronic form (on CD ROM) with automatic lemmatization, word form recognition and basic word derivation. It was mainly used as a source for finding English equivalents matching with WordNet 1.5 data.

- ***Dictionary of Czech Synonyms*,** by K.Pala and J.Vsiansky, Lidove Noviny Publishers, Prague, 1994-5, size: approx. 22 000 synsets (entries), its electronic version is with automatic lemmatization, word form recognition and basic word derivation is integrated into Czech localization of MS Word.

- ***Czech Synonymical Dictionary and Thesaurus I, II, III (Český slovník věcný a synonymický I, II, III),*** ed. by J.Haller, SPN Prague 1969-1977 – this is the only Czech thesaurus-like dictionary, however, it has not been finished because almost all its authors have died one after another. It does not exist in an electronic form. The dictionary contains useful collection of data that had been consulted when the hyperonym trees for Czech were created.

- A list of Czech collocations containing approximately 100 000 lines extracted from ***text corpus ESO*** (about 160 mil. word forms, built and maintained at the Faculty of Informatics), sorted according several criteria (MI score, frequencies, syntactic criteria) and prepared for the processing of the present-day Czech collocations. About 2000 collocations obtained in this way have already been integrated into Czech WordNet. The same procedure has been recently applied also to ***Czech National Corpus*** (presently having size about 101 mil. Czech word forms but the results from CNC have not been fully processed yet.

- Fully tagged and disambiguated ***corpus DESAM*** (both structurally and grammatically) that has been compiled at the Faculty of Informatics. It contains mostly newspaper and magazine texts from 1992-96, with size about 1 mil. Czech word forms. It has also been used for finding collocations.

- ***Corpus ESO*** built at the Faculty of Informatics in the course 1998 from newspaper and magazine texts (1996-98), presently having size about 650 mil. Czech word forms, partially tagged (lemmatized), in its earlier version it has been used as the main source of collocations.

- ***Valency dictionary of Czech*** containing approx. 15 000 items with all respective surface valency frames – it is used for inserting syntactic information into Czech WordNet.

2. 2 Tools

The following tools have been used in the course of building Czech Wordnet:

- ***VisDic*** – specialized browser and editor for wordnet like databases implemented in XML format (developed at FI MU NLP Laboratory mainly by T. Pavelek and implemented in C++ under MS Windows and Linux),

- ***Shallow parser DIS*** *(and its improved version VADIS*) able to analyze dictionary entries in *Dictionary of Written Czech* and also in *SSC (Dictionary of Literary)Czech* and select entries containing genus proximum definitions (with hypo/hyperonymy relation), implemented in Prolog under Unix,

- ***A simple translating program able to process a bilingual dictionary*** (Czech-English-Czech): it used a simple pattern matching and associated Czech entries with their English equivalents and then tried to link them with the WordNet 1.5 synsets. This program helped us to enlarge the number of the synsets considerably (its error rate was about 20%). The program was written in C and ran under Unix,

- A ***specialized program for creating ILR, i.e. hyperonym trees, ILI chains*** etc.. It worked in 3 cycles, employed ILI and WN 1.5 and Czech data as resources. It was implemented in C and runs under Unix.

- ***A program able to compute MI (mutual information) score for word forms from Czech text corpora*** (particularly from the corpus ESO built and maintained at Faculty of Informatics and also from Czech National Corpus) was written by P.Rychly (in Python programming language, under Linux) and used to obtain the list of present-day Czech collocations – the resulting file contained about 100,000 lines, we have selected about 2,000 items),

- ***Czech lemmatizer ajka*** by R.Sedlacek and K.Osolsobe, containing approx. 165,000 Czech stems and able to perform full morphological analysis of an arbitrary Czech text (also used for tagging corpora texts in corpus ESO and others, e.g. Czech National Corpus). The program is written in C and can be run under

Unix (Linux).

- ***Morphological database I_PAR*** (developed and implemented by M. Veber in C under Linux) – it will be used for linking morphologically related stems and associating the synsets with selected ILR. It can be adapted for other languages as well, especially Slavonic ones.

- ***Polaris 1.5 tool*** obtained from M.Louw and P.Vossen of Amsterdam has been explored and acquired for building Czech WordNet at the beginning of EuroWordNet Project 2, particularly for the processing of the sense links and building equivalent relations as well as ILR. After developing VisDic it is not used any more.

2. 3. VisDic

VisDic is a graphical application for browsing and editing dictionaries stored in XML format. It was developed primarily for editing wordnet databases, but with respect to universality. VisDic can be configured for any type of dictionary - monolingual, bi-lingual, thesaurus or just a plain corpora. A graphical design of the application comes from a fact, that more dictionaries can be edited at the same time. Each dictionary has its own sub-window consisting of three components (see Fig. 2.1):

1. **Text entry** at the top of window is used for looking-up data the user wish to display. Usually there is a need to type just one word. However more complicated queries can be typed as well.
2. **List box** containing found entries is filled after the query from text entry is entered. Each found entry will be displayed as one line in the box. After user clicks on it, the entry becomes active.
3. **Notebook** is the most important window placed on the bottom of dictionary sub-window. In this component a user can see more information about the active entry and all of its relations with other entries.

The last part - notebook contains more user-defined bookmarks. Each specifies a different view of the active entry. VisDic allows 6 types of view. All of them can be further configured:

- **Text view** is user defined. It displays information in more readable format which can be specified in the configuration file by an user. Typically it is a complete text overview of the entry, e.g. the full entry in a dictionary
- **Tree view** displays entries arranged to a tree according to user defined parent and children relations, e.g. the most useful hyperonymical-hyponymical relation in the WordNet can be displayed by this view. Similarly the holonymical-meronymical relation can be shown as well just by specifying different parent and children relations.
- **XML view** contains entry in the raw XML format. Here you can see, how the information are stored in the dictionary. It can be useful for some purposes.

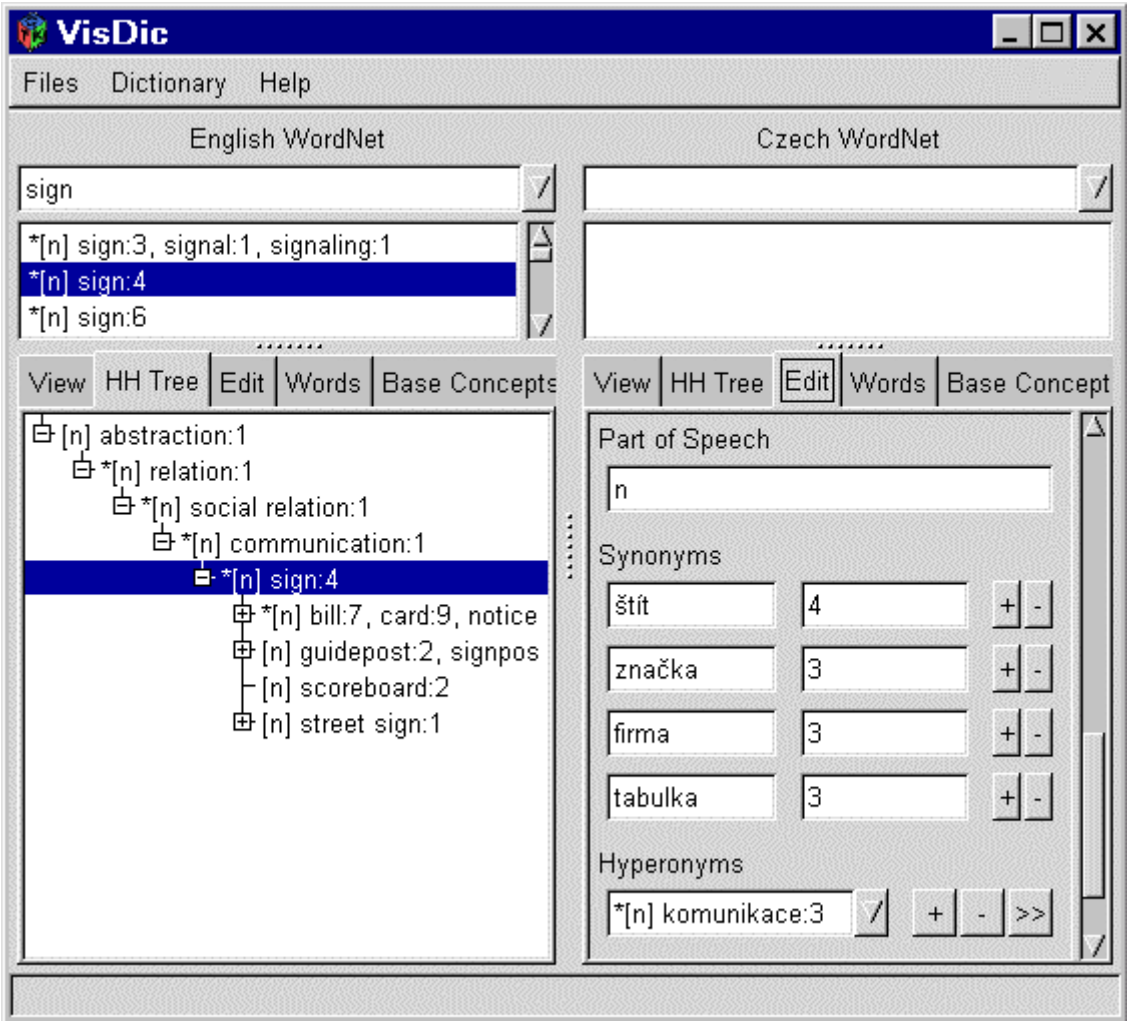


Figure 2.1: VisDic with two dictionaries – English WordNet and Czech WordNet

- .**Edit view** allows to edit the actual entry. A simple XML tag can be edited in a single line text box or in the multi-line text box. An XML tag containing a link to other entry, like a hyperonym relation must contain a valid entry link, so the value must be chosen from a list of them. A combobox serves for these purposes. Just like with searching entries, a user need to type a query. The corresponding entries will be shown in the combobox. After selecting a correct entry, the value will be given. On the right side of these edit boxes, there are three buttons signed +, - and >>. The first one inserts an entry after the actual, the second one removes the actual entry. The third button is shown only if the edited tag is a link to another entry. This link can be followed by clicking on the button. At the end, there are some other buttons. Clicking on *New*, a user can create a new entry with unique key. Clicking on *Add* will add actual entry to the dictionary, clicking on *Delete* the actual entry will be updated.
- **Word view** shows a list of words. They are all words stored in specify entry of a dictionary, e.g. all WordNet literals. You can select a word and transfer it to upper combo box by clicking on it with the right mouse button. It searches the corresponding entries automatically.
- **Entry view** shows a list of entries restricted by some query, e.g. all Base Concepts of some WordNet can be displayed in such a view. When you click on one of these entries, it will become active.

- Windows version of VisDic

VisDic was primarily developed on Linux platforms in C++ language using GIMP Toolkit 1.2. It was tested mainly on RedHat 7.1 and Debian 2.2. However, it should be compatible with other Linux distribution (SuSE, Mandrake) supporting X-Windows and GTK libraries.
At the end of April 2002, VisDic was recompiled on Windows platforms as well by means of project Cygwin and GTK 2.0 dynamically linked libraries (DLL) for Windows. The behavior should be exactly the same as on Linux systems. However, it was necessary to rewrite some parts of source code. The main reason was that Windows DLL files of GTK 2.0 in Windows needs all texts to be encoded in unicode UTF-8 character set. But each dictionary in VisDic can be stored in different encoding. Therefore, the following functions were implemented:

- Transfer of texts stored in inner dictionary representation to UTF-8 in case, when VisDic displays these information on the screen
- Transfer of texts typed by user from UTF-8 encoding back to inner representation of a dictionary.

A conversion of texts between encodings was provided by `iconv` library.
An installation is more easier than on Linux systems. The installation package contains VisDic binary files, English WordNet and ILI dictionary.

Further information and VisDic installation packages can be found on http://nlp.fi.muni.cz/projekty/mt/visdic.

3. TOOLS ANS RESOURCES FOR THE GREEK WORDNET

3.1 Language resources for Greek

The language resources used for the development of the Greek WordNet are the following:

- ***The electronic dictionary of the Patakis Publishing Co.***
  It consists of 82,021 lemmata with 67,944 definitions in a Microsoft Access 2000 format

- ***The "Triantafyllidis" electronic lexicon of the Center of the Greek language***

It consists of 50,506 lemmata with 98,103 definitions in a Microsoft Access 2000 format

- ***The Greek part of the E.C.I. Corpus.***

The E.C.I. (European Corpus Initiative) is a large scale corpus, which is a joint project of the Universities of Edinburgh and Geneva on behalf of the A.C.L. The Greek part of the E.C.I., having more than 2 million word-tokens, actually contains approximately 94,000 words. These words are produced by 33,000 different lexemes, through the morphological process of inflection.
The Greek part of the E.C.I. is composed of 48 files, arranged according to their subject in 11 classes. These subject classes are sports, economics, education, medicine, philosophy, astrology, law, literature, politics & sociology, science and technology.
The Greek part of the E.C.I. is also organized in lists that contain the words and their individual frequencies of appearance in the text.

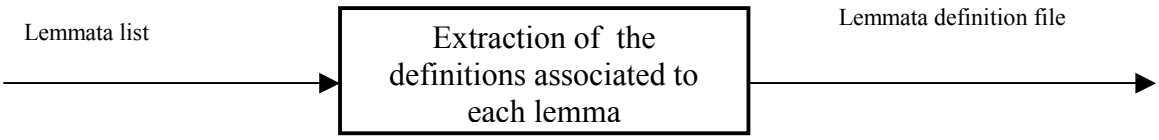
3.2 Tools

3.2.1 Existing tools for the extraction of semantic information

The following computational tools were developed during the "Dialexiko" project that resulted to the creation of an initial fragment of the Greek WordNet and they are applicable to our work in the Balkanet. The tools were developed using Visual basic v 6.0 in order to facilitate the extraction of semantic information from the Patakis lexicon which was in Word format.

- ***The 'definitions extraction' tool***

It extracts the definitions associated to each lemma and produces an output that is considered as a corpus and can be used as input to other tools.

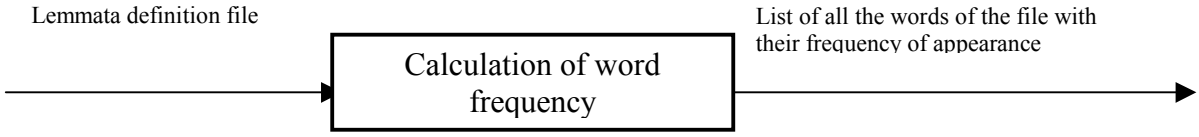


Example:

Άλφα , το πρώτο γράμμα του ελληνικού αλφαβήτου .
Χρησιμοποιείται ως α ΄ συνθετικό και σημαίνει έλλειψη , στέρηση ή αντίθεση ( )
Μπαίνει στην αρχή της λέξης χωρίς να επηρεάζει τη σημασία της

- ***The 'word frequency calculation' tool***

It calculates the word frequency in the lemmata definition file. At output, it produces a list of all the words of the file with their frequency of appearance in that file.
The definitions extracted from the Patakis dictionary take the place of the corpus against which the semantic correlations were tested. Therefore, word frequency calculation was useful in making assumptions on some of the base concepts.

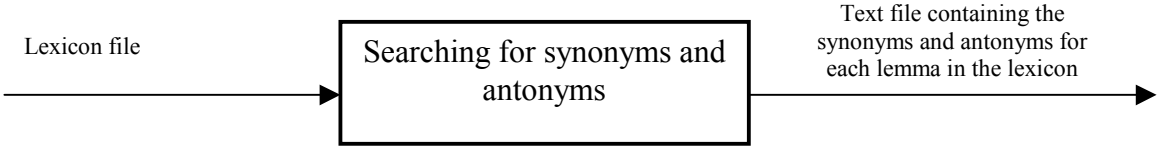


Example from a search in a short text file:

1 αβά
1 αββαείον
1 Άλφα
1 αλφαβήτου
1 ανήκει
1 αντίθεση
2 από

- ***The 'synonyms and antonyms extraction' tool***

It extracts all the synonyms and antonyms associated to each lemma in the lexicon which is accompanied by the indexes syn and ant respectively used as a helpful device.



Example:

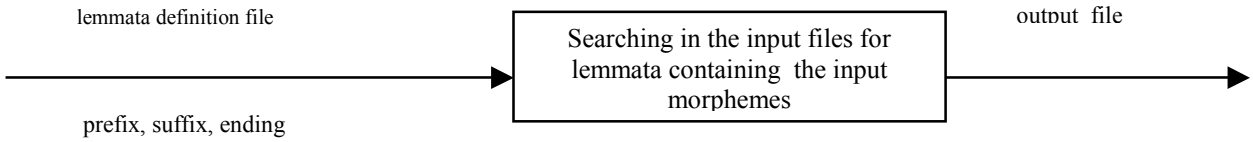
αβαθής syn : άβαθος-,ρηχός- ant: βαθύς-
αβαθμολόγητος syn : ant: βαθμολογημένος-
άβαθος syn : αβαθής-,ρηχός-,ξέβαθος- ant: βαθύς-
αβαλσάμωτος syn : αταρίχευτος- ant: βαλσαμωμένος-,ταριχευμένος-

- ***The 'search for antonymic relations in the definitions of lemmata' tool***

In the lemmata definition file the tool looks for lemmata file which are formed on the basis of a given prefix, suffix, or ending.

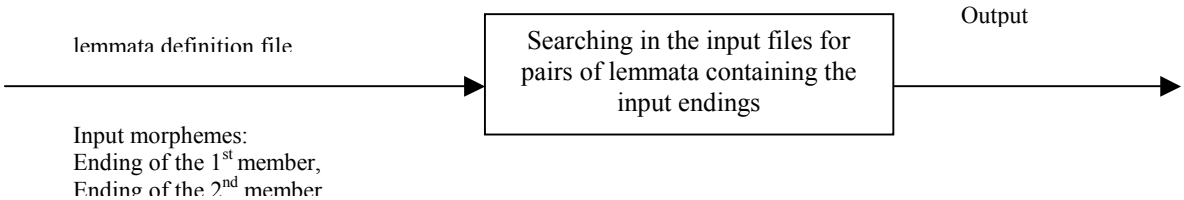
The output of this tool is a text file containing the lemmata which correspond to the user's input choices. Finally, it is used in order to give, among others, an insight

as to whether certain negative prefixes such as *α-* 'un-' express the relation of antonymy.



➢ ***Tool searching relations such as "role-involved" in the lemmata definition file***

It receives two different endings and looks for paired lemmata, which have the same stem –by convention, the first three characters of a lemma are considered as stem- and different endings. The aim of the tool in question is to look for possible relations such as 'role-involved', etc. in the lemmata included in the lexicon. The input to this tool is the file containing all the lemmata as well as some additional files produced on the basis of the initial pre-processed file. The output of the tool is a file containing pairs of words that may be semantically linked

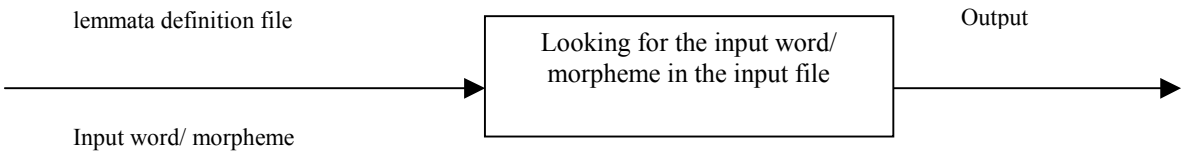


Example:

ακροβάτ - ης (acrobat)
ακροβατ - ώ (verb denoting the acrobat's action)

➢ ***The 'search for possible semantic relations' tool***

It receives as input a word or a morpheme and then it looks for that word/ morpheme (in any possible form) in the definitions of lemmata. The output contains lemmata the definition of which definitions include the input word/ morpheme. The underlying assumption is that if a lemma is identified in the definition field of another lemma, the two lemmata might be in a semantic relation (e.g., hyponymy).



3.2.2 Morphological Processing Tools

- ***The morphological analysis system.***

The morphological analysis system (M.A.S.) is a computational tool for morphological analysis tasks of Greek (Papakitsos 2000, Papakitsos et al. 1998, 2000). The M.A.S. can analyze the words into their individual morphemes of any kind (namely endings, suffixes, roots and prefixes), provided that these morphemes are listed in separately structured catalogues (database). The M.A.S. consists of 3 software files, 2 input-output files, 5 database files and a user-instruction file (Readme.doc). Especially:

- The software files contain the source-code of the morphological analyzer.
- The database files contain lists of morphemes, where there is one file for each class of morphemes (namely endings, suffixes, prefixes, roots and free-morphemes). Each file has an appropriate internal structure suitable for facilitating the morphological processing.
- The input-output files contain the input data for the morphological analyzer and the produced results of the morphological processing.

The M.A.S. works as follows: the morphological analyzer reads each word to be analyzed from the input file, one by one. Then, with the help of the database files, it analyzes the word into its constituent morphemes. Finally the results of the analysis are automatically written down into the output file, from where they can be accessed.

Technical description.

The M.A.S. is internally structured as it is described below:

The 3 software files are named parse.exe, parser.tpu and initial.tpu. The first one is the executable file of the morphological analyzer. It is called (activated) by typing “parse” (and <enter>) at the prompt-line. The first action to be taken by the analyzer is to read the database files and to create a RAM-version of them, along with key-numbers for every morpheme.

The 5 database files are type-text files. Each one contains one class of morphemes, namely endings (Endings.src), suffixes (Suffix.src), roots (Roots.src), prefixes (Prefix.src) and free-morphemes (Words.src). In any of the database files, every morpheme preceded by its accompanying properties (inflectional class and morphosyntactic attributes) is listed in a separate line:

```
1 N--V ικ
1 A--N ιζ
1 MSN3 ος
1 MSG2 ης
```

The words to be analyzed every time, must be listed each in a separate line of the type-text input file, named “input.txt”, without stress (any stress mark must have been removed firstly). The morphological analyzer reads every line of the input file, proceeds to the morphological analysis of every word in there and finally, it writes down the results of the analysis into the type-text output file, named “output.txt”. The results for every analyzed word in the output file, are listed in two lines as in the following example:

αντιδιαμετρικος:

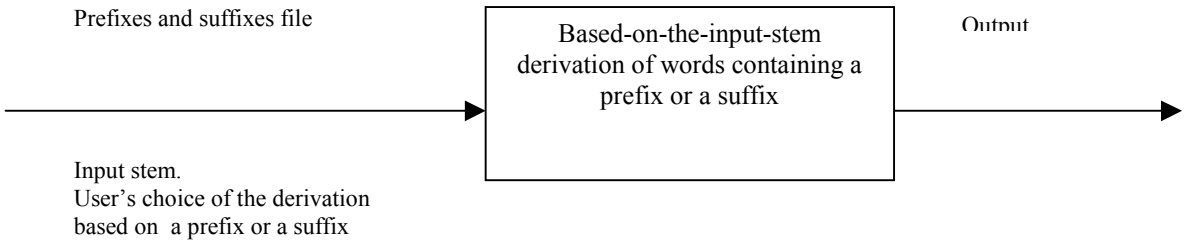
αντι-δια-μετρ-ικ-ος P.549-P.281-R.639-S.744-E.23 N--AMNS1

On the upper line, there is the word as it appears in the input file, followed by a colon

character. On the bottom line, the above word is written down having dash characters to separate its morphemes. Then an identification of each morpheme follows, having the class of each morpheme (a capital character), a dot and the key-number of it (P.549 … E.23), in the same order that they appear in the word (e.g. for the first morpheme "αντι", which is a prefix, its identification is P.549). Finally, there is a string of 8 characters, denoting the other attributes of the word, according to the M.A.S. encoding manner.
The M.A.S. has a simple internal process for validating the results of the analysis. For best results, an enriched database is required.

- ***The wordform generator***

The tool takes as input a word and produces all the declined forms, the derivatives and certain compounds in a semi-automatic way. In particular, the stem in question is combined with prefixes and/or suffixes producing actual words



3.2.3 Tools developed for the extraction and processing of the necessary linguistic information

The already existing tools were redesigned and restructured in order to support the extraction of linguistic information from lexicon files in Microsoft Access 2000 format. A new set of tools was also added since all our lexical resources (Patakis and Triantafyllidis electronic lexica) are in MS Access 2000 format.

First of all, a method for the part of speech (POS) characterization regarding the lexicon's lemmata has been developed. The method was based on grammatical information concerning POS, which was incorporated in the lexicon.
By the application of the above mentioned method the lemmata of the lexicon are grouped in separate tables of nouns, verbs, adjectives etc. These tables have an auxiliary role in the extraction of semantic information.

- ***The 'extraction of POS- related information' tool***

It extracts the definitions associated to each lemma - along with the lemma - for a given POS or all POSs existing in the lexicon. In addition, it extracts the lemmata from the lexicon either grouped by their POS or altogether. The user can inspect the resulting output or save it (in .xls format).

An excerpt of the noun table (along with additional information concerning etymology and grammar) follows:

LEMMA_ID	LEMMA	ETYMOLOGY	GRAMMAR	LINK_ TEXT	LINK_ID	Part_of_Speech
108	αβιογένεση	[λόγ. <νλατ. ABIOGENESIS <A- = α-1 + BIO- <λατ. BIO- <αρχ. βίο(ς) + αρχ. γένε(σις) -ση]	η , [AVIOJıNESI] O33		0	ΟΥΣΙΑΣΤΙΚΟ
112	αβιταμίνωση	[λόγ. <γαλλ. AVITAMINOSE <A- = α-1 + VITAMIN(E) = βιταμίν(η) -OSE = -ωσις -ωση]	η , [AVITAMvNOSI] O33		0	ΟΥΣΙΑΣΤΙΚΟ
119	αβλέμονας	[ίσως <αρχ. *ϋβλέμμων, αιτ. -ονα ‘όπου δε φτάνει το βλέμμα’]	ο , [AVLıMONAS] O5		0	ΟΥΣΙΑΣΤΙΚΟ
120	αβλεψία	[λόγ. <ελνστ. ϋβλεψία ‘ανικανότητα να δει κανείς’]	η , [AVLEPSvA] O25		0	ΟΥΣΙΑΣΤΙΚΟ
130	αβουλησία	[λόγ. <ελνστ. ϋβουλησία]	η , [AVULISvA] O25		0	ΟΥΣΙΑΣΤΙΚΟ
131	αβουλία	[λόγ. <αρχ. ϋβουλία]	η , [AVULvA] O25		0	ΟΥΣΙΑΣΤΙΚΟ
153	αβρότητα	[λόγ. <αρχ. κβρότης, αιτ. -ητα]	η , [AVRσTITA] O28		0	ΟΥΣΙΑΣΤΙΚΟ
154	αβροφροσύνη	[λόγ. αβρόφρ(ων <αβρ(ός) -ο- + -φρων) -οσύνη]	η , [AVROFROSvNI] O30α		0	ΟΥΣΙΑΣΤΙΚΟ
156	αβτζής	[τουρκ. AVC† -ς]	ο , [AVDZvS] O8		0	ΟΥΣΙΑΣΤΙΚΟ
162	άβυσσος	[λόγ.: 1: αρχ. τβυσσος μ (λαϊκό: ο άβυσσος, η άβυσσο)· 2: σημδ. γαλλ. ABYSSE ή αγγλ. ABYSS (<αρχ. τβυσσος)· 3: ελνστ. σημ.]	η , [αVISOS] O36		0	ΟΥΣΙΑΣΤΙΚΟ
167	αγαθό	[αρχ. ϋγαθόν, πληθ. (στη σημ. 1α) τά ϋγαθά (1β: λόγ. σημδ. γαλλ. BIENS· 2α: λόγ. <αρχ. ϋγαθόν· 2β: ελνστ. σημ.)]	το , [ΑγΑθσ] O38		0	ΟΥΣΙΑΣΤΙΚΟ
170	αγαθοεργία	[λόγ. <αρχ. ϋγαθοεργία]	η , [ΑγΑθΟERJvA] O25		0	ΟΥΣΙΑΣΤΙΚΟ
172	αγαθοπιστία	[λόγ. αγαθόπιστ(ος) -ία μτφρδ. γαλλ. BONNE FOI]	η , [ΑγΑθΟPISTvA] O25		0	ΟΥΣΙΑΣΤΙΚΟ
174	αγαθοποιία	[λόγ. <ελνστ. ϋγαθοποιύα]	η , [ΑγΑθΟPIvA] O25		0	ΟΥΣΙΑΣΤΙΚΟ
177	αγαθοσύνη	[ελνστ. ϋγαθωσύνη (ορθογρ. κατά το επίθημα -οσύνη)]	η , [ΑγΑθΟSvNI] O30		0	ΟΥΣΙΑΣΤΙΚΟ
178	αγαθότητα	[λόγ. <ελνστ. ϋγαθότης, αιτ. -ητα]	η , [ΑγΑθσTITA] O28		0	ΟΥΣΙΑΣΤΙΚΟ
180	αγαλακτία	[λόγ. <νλατ. AGALACTIA (στη νέα σημ.) <αρχ. ϋγαλακτία ‘έλλειψη γάλακτος’· λόγ. <νλατ. AGALAXIA <ελνστ. ϋγάλαξ ‘που δεν έχει γάλα’ -IA = -ία (σφαλερά αντί AGALACTIA)]	η , [ΑγALAKTvA] αγαλαξία η [ΑγALAKSvA] O25		0	ΟΥΣΙΑΣΤΙΚΟ
181	αγαλαξία			αγαλακτία	180	ΟΥΣΙΑΣΤΙΚΟ

- ***The ‘extraction of linked lemmata and lemmata acting as compounds ’ tool***

The electronic lexicon follows the convention that lemmata that are the feminine form of a masculine noun (e.g. αδελφή[sister] - αδελφός [brother]) or constitute the alternative form of another lemma (e.g. αϊτόπουλο versus αετόπουλο[eaglet]) are "linked" together. This means that the corresponding senses of a lemma that is linked to another, correspond to the linked lemma, as well.
This tool extracts all the linked lemmata from the lexicon and there is also the capability of extracting only the linked lemmata of feminine-masculine type. An excerpt of the data of this type follows below:

LEMMA_ID	LEMMA	LINK_TEXT	LINK_ID	TAG	SENSE	Part_of_Speech
327	αγελαδάρισσα	αγελαδάρης	326	0	βοσκός αγελάδων.	ΟΥΣΙΑΣΤΙΚΟ
329	γελαδάρισσα	αγελαδάρης	326	0	βοσκός αγελάδων.	ΟΥΣΙΑΣΤΙΚΟ
337	αγελαδοτρόφος	αγελαδοτρόφος	336	0	αυτός που ασχολείται με την αγελαδοτροφία.	ΟΥΣΙΑΣΤΙΚΟ
374	αγιογδύτισσα	αγιογδύτης	373	1	άρπαγας, κλέφτης, που μπορεί να κλέψει και ιερά αντικείμενα από εκκλησίες· ιερόσυλος.	ΟΥΣΙΑΣΤΙΚΟ
374	αγιογδύτισσα	αγιογδύτης	373	2	ως υβριστικός χαρακτηρισμός προσώπου που χρησιμοποιεί αδίστακτα κάθε μέσο για να αποσπάσει χρήματα, αισχροκερδής, στυγνός εκμεταλλευτής	ΟΥΣΙΑΣΤΙΚΟ
375	αγιογδύτισσα	αγιογδύτης	373	1	άρπαγας, κλέφτης, που μπορεί να κλέψει και ιερά αντικείμενα από εκκλησίες· ιερόσυλος.	ΟΥΣΙΑΣΤΙΚΟ
375	αγιογδύτισσα	αγιογδύτης	373	2	ως υβριστικός χαρακτηρισμός προσώπου που χρησιμοποιεί αδίστακτα κάθε μέσο για να αποσπάσει χρήματα, αισχροκερδής, στυγνός εκμεταλλευτής	ΟΥΣΙΑΣΤΙΚΟ
379	αγιογράφος	αγιογράφος	378	0	ζωγράφος ιερών εικόνων και θρησκευτικών παραστάσεων.	ΟΥΣΙΑΣΤΙΚΟ

Similarly, this tool extracts the lemmata that can act as compounds, with or without their corresponding senses, as it is illustrated below:

LEMMA_ID	LEMMA	ETYMOLOGY	GRAMMAR	LINK_TEXT	LINK_ID
6	α- 1	[αρχ. στερ. πρόθημα ϋν- συνήθ. πριν από φων.: αρχ. ϋν-άξιος ϋ- πριν από σύμφ.: αρχ. ϋ-δάκρυτος, σπάν. το αντ.: αρχ. ϋ-όρατος (αρχικά παρήγε μεταρ. επίθ.: αρχ. ϋ-δάκρυτος, αλλά επεκτάθηκε στην παραγωγή και άλλων επιθ.: αρχ. τ-μοιρος, ϋ-σεβής και τελικά στην παραγωγή ουσ.: νεοελλ. α-λυγισ-ιά) λόγ. <διεθ. A-, AN- <λατ. A-, AN- <αρχ. ϋ-, ϋν-: α-λογικός, αν-αερόβιος <γαλλ. ALOGIQUE, ANAιROBIE· ανα-: μσν. ανα-: μσν. ανά-λουστος 'άλουστος', επέκτ. από επίθ. που άρχιζαν με α-: αρχ. ϋν-αμάρτητος ή με επανάληψη του αρνητικού: αν-ά-λουστος· ανε-: μσν. ανε-: μσν. ανέ-γνοιαστος, επέκτ. από επίθ. που άρχιζαν με ε-: ελνστ. ϋν-έξοδος· ανη-: επέκτ. από επίθ. που άρχιζαν με η-: αρχ. ϋν-ήκουστος, μσν. αν-ήμπορος (<μσν. ημπορώ) με νέα ανάλ. ανη-, με βάση τον τ. μπορώ, νεοελλ. ανη-πρόκοπος 'ανεπρόκοπος']	[A] αν-1 [AN], συνήθ. πριν από φωνήεν ά- [α] ή άν- [αΝ], όταν ο τόνος ανεβαίνει στο πρόθημα ανα-1 [ΑΝΑ] ή ανά- [ΑΝα], μερικές φορές πριν από σύμφωνο (σπάν., λαϊκότρ.) ανε- [ΑΝΕ] ή ανέ- [ΑΝι] ανη- [ΑΝΙ] ή ανή- [ΑΝν], αναλογικά προς λέξεις που άρχιζαν από α, ε, η		0
7	αν-1			α- 1	6

LEMMA_ID	LEMMA	ETYMOLOGY	GRAMMAR	LINK_TEXT	LINK_ID
8	ά-			α- 1	6
9	άν-			α- 1	6
10	ανα-1			α- 1	6
11	ανά-			α- 1	6
12	ανε-			α- 1	6
13	ανέ-			α- 1	6
14	ανη-			α- 1	6
15	ανή-			α- 1	6
16	α- 2	[αρχικό α- σε ρηματ. επίθ. που ερμηνεύτηκε ως στερ. α- 1 (με υποχωρ. κίνηση του τόνου για ένδειξη στερητικής σημ., αναλ. προς επίθ. με α-1): μσν. ά-γγιχτος]			0
17	α- 3	[μσν. προτακτ. α- από συμπροφ. με το αόρ. άρθρο και ανασυλλ.: μσν. μασχάλη >αμασχάλη [MIA-MA >MIAMA >MI-AMA], ράθυμος >αράθυμος [ENA-RA >ENARA >EN-ARA]· στα ρ. από συμπροφ. με τα ρηματ. μόρια να, θα: δράχνω >αδράχνω [NA-δRA >NAδRA >N-AδRA]]			0
18	α- 4	[όπως στο α-3 με συνίζ. για αποφυγή της χασμ.: ελαφρός >αλαφρός [ENA-ELA >ENALA >EN-ALA], εγγίζω >αγγίζω [NA-ENGI >NANGI >N-ANGI]]			0
168	αγαθο-	[□: θ. του επιθ. αγαθ(ός) (<αρχ. ϋγαθός) -ο- ως αΔ συνθ.· □□: λόγ. <αρχ. ϋγαθο- θ. του επιθ. ϋγαθό(ς) ως αΔ συνθ.: αρχ. ϋγαθο-εργία, ελνστ. ϋγαθο-ποιός]	[ΑγΑθΟ] αγαθό- [ΑγΑθσ], όταν ο τόνος κατά τη σύνθεση ανεβαίνει στο αΔ συνθετικό		0
169	αγαθό-			αγαθο-	168

➢ ***The 'synonyms and antonyms extraction' tool***

It extracts the synonyms and antonyms associated to the lexicon's lemmata. An excerpt of the antonyms list follows below:

LEMMA_ID	LEMMA	TAG	SENSE	ANTONYM	LINK_TEXT	LINK_ID
33	αβαθής -ής -ές	0	που δεν έχει βάθος, άβαθος, ρηχός.	βαθύς		0
34	αβαθμολόγητος -η -ο	0	που δε βαθμολογήθηκε ακόμα.	βαθμολογημένος		0
36	αβαθούλωτος -η -ο	0	που δεν είναι ή που δεν έγινε βαθουλός.	βαθουλωμένος.		0
39	αβαλσάμωτος -η -ο	0	που δε βαλσαμώθηκε· αταρίχευτος.	βαλσαμωμένος.		0
40	άβαλτος -η -ο	0	που δεν τον έχουν βάλει, δεν τον έχουν ακόμα τοποθετήσει στο μέρος για το οποίο προορίζεται· ατοποθέτητος.	βαλμένος		0
92	αβέβαιος -η -ο	II2	(για ενέργεια) που γίνεται με τρόπο που δείχνει αμφιβολία ή δισταγμό.	σταθερός, σίγουρος		0
92	αβέβαιος -η -ο	II1β	που αμφιβάλλουμε για την αλήθεια ή την ορθότητά του.	σίγουρος		0
93	αβεβαιότητα	0	η κατάσταση ή η ιδιότητα του αβέβαιου.	βεβαιότητα, σιγουριά		0

➢ ***The 'search for semantic relations' tool***

This tool can be used for the tracing of semantic relations such as 'role-involved', antonymic relations, 'part-of' etc. The user can search the lexicon's lemmata for certain prefixes, suffixes or morphemes in order to investigate semantic relations, for instance, acquiring an insight as to whether certain negative prefixes such as *α-* 'un-'

express the relation of antonymy. There is also the choice for the user to search in the definitions of lemmata for specific expressions such as `"x is a part of y" or "x is a kind of y".` The tool can be used also for seeking a word or a morpheme (given by the user), in any possible form, in the definitions of lemmata. The output contains those lemmata the definitions of which contain the input word/morpheme. The underlying assumption is that if a lemma is identified in the definition field of another lemma, there is a possibility that a semantic relation holds between the two lemmata (e.g., hyponymy). An example of the latter is the search for the string "δέντρ-"[tree] in the definitions of the lexicon for the tracing of hyponymy-hyperonymy relation, concerning the lemma "δέντρο"[tree] follows below:

LEMMA_ID	LEMMA	SENSE	LINK_TEXT
595	**αγριομηλιά**	ονομασία άγριων δέντρων που συνήθ. συγγενεύουν με τη μηλιά.	
599	αγριόξυλο	ξύλο συνήθ. από άγριο δέντρο, πολύ σκληρό και ακατάλληλο για ξυλουργικές εργασίες.	
608	**αγριοσυκιά**	ονομασία άγριων δέντρων που συνήθ. συγγενεύουν με τη συκιά.	
1590	**ακαγιού**		ακαζού
1589	**ακαζού**	το κοκκινωπό ξύλο του ομώνυμου τροπικού δέντρου· μαόνι	
1610	**ακακία**	είδος δέντρων ή θάμνων που έχουν αγκαθωτά κλαδιά, πλούσιο φύλλωμα και μικρά, σφαιρικά άνθη, κίτρινα ή λευκά, εύοσμα ή άοσμα, ανάλογα με την ποικιλία του φυτού.	
3099	**αμυγδαλιά**	φυλλοβόλο, καρποφόρο δέντρο με λευκορόδινα άνθη που καρπός του είναι το αμύγδαλο	
3305	αναδάσωση	δεντροφύτευση απογυμνωμένων δασικών εκτάσεων	
7142	**αρκουδοπούρναρο**	(σπάν.) αειθαλές μικρό δέντρο με οδοντωτά και αγκαθωτά φύλλα και μικρούς στρόγγυλους καρπούς κόκκινου χρώματος· ου.	
7337	**αρτόδεντρο**	δέντρο που καλλιεργείται σε τροπικές χώρες για τους καρπούς του, που χρησιμοποιούνται ως ψωμί.	
8885	**αφροξυλιά**	ονομασία θάμνου ή μικρού φυλλοβόλου δέντρου, που φύεται ή καλλιεργείται στην Ελλάδα και σε όλη την Ανατολή και που σε νεαρή ηλικία έχει κλαδιά με πολύ μαλακό και ελαφρό ξύλο.	
8970	**αχλαδιά**	οπωροφόρο δέντρο του οποίου καρπός είναι το αχλάδι· απιδιά	

In addition, the user can look for semantic relations, based on the morphology of lemmata. The tool receives two different endings and looks for paired lemmata, which have the same or approximately the same stem and different endings. The paired lemmata can be of type noun-noun, noun-verb or adjective-noun. The aim of the tool in question is to look for possible relations such as 'role-involved' (type noun-verb), etc. in the lemmata included in the lexicon. An excerpt of the results for searching of

endings "-της –ω" (type noun-verb, indication of existence of 'role-involved' relation) follows (the lemmata for which the relation was detected are marked in bold):

NOUN_ID	NOUN	VERB	VERB_ID
514	**αγορητής**	**αγορεύω**	513
642	αγρότης	αγρικώ	20274
642	αγρότης	αγρεύω	548
703	**αγωνοθέτης**	**αγωνοθετώ**	705
1057	**αεροβάτης**	**αεροβατώ**	1058
1250	**αθλητής**	**αθλώ**	1271
1265	**αθλοθέτης**	**αθλοθετώ**	1268
1379	αιματίτης	αιματώνω	1403
1883	**ακονιστής**	**ακονίζω**	1880
1892	**ακοντιστής**	**ακοντίζω**	1887
2030	**ακροβάτης**	**ακροβατώ**	2034
2199	αλαλίτης	αλαλάζω	2192

- ***Contrastive word frequency calculation***

Word frequency calculation was considered useful in making assumptions on some base concepts, so we have implemented a method to calculate the word frequency for each one of the available lexical resources. As concerning the two dictionaries, the word frequency calculation was made against the definitions they include. The calculation was made separately for each one of the lexical resources. In addition, these results were contrasted to each other, in order to facilitate various observations as the comparison between the frequency of appearance of the same word in the different lexical resources. For the purposes of the word frequency calculation, the 'word frequency calculation tool' (existing tool, developed for the 'Dialexiko' project) was used. This tool calculates the word frequency in a text file and produces a list of all the words of the file with their frequency of appearance in that file.

Brief description of the method:
At first, the extraction of the dictionaries' definitions took place. As the dictionaries are in Microsoft ACCESS 2000 format, the suitable queries were used so that the definitions have been extracted in a Word format.

The second step involved the use of the 'word frequency calculation tool' in order to calculate the word frequency for the dictionaries' definitions and the text files comprised the ECI Corpus. In particular in the case of the ECI Corpus, the tool was applied in each file separately and, in the end, all the resulted files containing the list of all the words of each file with their frequency of appearance in that file, were merged.

In the last step, the previous results (text files) were imported into Microsoft ACCESS, forming a resource for further processing and concluding such as comparison between the various lexical resources concerning the word frequency,

detection of the most/less frequent used words, tracing the differences in word using in varied resources as dictionaries and corpuses, etc.

In the table below lies an excerpt of the final results which include a total of 130,844 different words and their number of appearance in each lexical resource. The first column of the table contains words and the rest three columns illustrate the number of their appearance in each of the resources.

Λέξη	Πλήθος_στο_Λεξικό_ Τριανταφυλλίδη	Πλήθος_στο_Λεξικό_ Πατάκη	Πλήθος_στο_ ECI_CORPUS
αβά	2	1	0
αβαγιαννούς	0	0	1
αβαείο	1	0	0
αβαείου	0	1	0
αβαθείς	1	0	0
αβαθές	3	0	0
αβαντάζ	0	0	2
αβάντες	1	0	0
αβάντζα	1	0	0
αβάπτιστα	0	2	1
αβάπτιστο	0	2	0
αβάπτιστος	0	1	0
αβαράρω	1	0	0
αβαρίες	0	0	1
Αβαρίζ	0	0	7
Άβαρος	1	0	0
Αβασάνιστα	1	2	4
Αβασικός	0	0	1
Αβάσιμα	1	0	0
Αβάσιμες	1	0	1
Αβάσιμη	1	3	4
Αβάσιμο	2	1	1
Αβάσιμοι	0	0	2
Αβάσιμος	2	2	3
Αβάσιμους	0	0	2

3.3 Contribution of Tools towards the Base Concept Selection Process

The tools described above were used for the processing of the lexical resources available for Greek so as to extract from the latter lexicographic information on which the development of the core Greek WordNet would be based.

More specifically, following the decisions made during the second project progress meeting in Istanbul[1], each partner would have to propose 500 new terms as candidate Base Concepts. The need for additional Base Concepts besides the ones already

[1] For further details regarding the discussions that took place during the meeting please refer to the minutes of the meeting, released on the project's information server (http://dblab.upatras.gr)

developed by the consortium emerged from the decision made by all contractors that lexicalized patterns of Balkan languages should be reflected in the final multilingual database. In this respect partners would have to process separately the monolingual lexical resources having at their disposal in order to conclude on each language's most representative terms and suggest them as new candidate Base Concepts. Afterwards, a common list of all proposed Base Concepts was compiled and checked by the FI MU contractor in order to eliminate duplicates. Once the above process was completed all Base Concepts proposed by all or by more than one language were considered as the new Base Concepts that would be incorporated in the BalkaNet multilingual lexical database. Base Concepts proposed by one language were included in the respective monolingual WordNet and formed local Base Concepts. The selection process took place by each contractor separately based on the tools and lexical resources available. In particular, during the selection of the new 500 Greek Base Concepts the following procedure was adopted.

Members of the UOA contractor developed tools for the processing of the Greek lexical resources. The tools developed as well and their functionality are explicitly described above. Following on from this, linguists of the DBLAB team processed the frequency lists of terms extracted by the explanatory dictionaries as the latter were obtained by the UOA team. More specifically, two distinct lists comprising the most frequent terms of both explanatory Greek dictionaries, namely Patakis and Triantafillidhs dictionaries, were used as the basis of the Greek Base Concepts selection. Linguists of DBLAB isolated from the abovementioned lists the most frequent terms and considered them as candidate Base Concepts. Following, they checked these candidates against the already existing Base Concepts in order to trace which of them might already be present in the BalkaNet database. Terms already incorporated in VisDic editor, as Base Concepts were eliminated form the final list suggested by the Greek team.

The methodology followed for tracing Base Concepts that were already present in VisDic is described hereinafter. First of all candidate terms were translated in English using bilingual (Greek-English) machine-readable dictionaries. Once all candidate terms were translated they were checked against the ILI records in order to trace which of them were already present. Missing terms from the ILI were considered as the new candidates and they were linked with the respective Greek terms. Once the process was completed, Greek partners had come up with a list of 500 new Greek Base Concepts linked to the ILI records, which were proposed to the rest contractors. All partners following similar methodologies come up with respective lists and common terms proposed as Base Concepts were considered as the final set of Base Concepts to be included in the BalkaNet multilingual lexical database. The abovementioned procedure is currently being followed for the determination of the new set of terms that will form the basis on which new synsets will be developed for all monolingual WordNets. The approach followed ensured maximal overlap among the monolingual WordNets and guarantees a degree of compatibility among languages, in terms of vocabulary coverage and completeness.

References:

Bloksma L., Díez-Orzas P. and P. Vossen (1996) *User Requirements and Functional Specification of the EuroWordNet project.* EuroWordNet Project LE2-4003, Deliverable D001, 66 p.

Díez-Orzas P., Ph. Forrest and M. Louw (1996) *High Level Architecture of the WordNet Database.* EuroWordNet Project LE2-4003, Deliverable D007, 74 p.

Fellbaum C. (ed.) (1998) *WordNet: An Electronic Lexical Database.* MIT Press, 405 p.

Galiotou E., G. Giannoulopoulou, M. Grigoriadou, A. Ralli, C. Brewster, A. Arhakis, E. Papakitsos, A. Pantelidou (2001) Semantic Tests and Supporting Tools for the Greek WordNet, *NAACL Workshop on WordNet and Other Applications (Poster Session*), Carnegie Mellon, Pittsburgh, PA, pp. 183-185.

Grigoriadou M., Galiotou E., Papakitsos E., Pantelidou A. (2001) Computational Tools to Support the Greek WordNet (in Greek), *Proceedings of the Workshop on Lexical Databases- Electronic Language Resources (held during the 22nd Meeting of Linguistics)*, Aristotle University, Thessaloniki, pp. 33-40.

Mackridge P. (1985) *The Modern Greek Language.* Oxford University Press.

Miller G., R. Beckwith, C. Fellbaum, D. Gross, K.J. Miller (1990) Introduction to WordNet: An On-line Lexical Database, International Journal of Lexicography, 3/4,pp. 235-244.

Papakitsos E. (2000) *A Contribution to the Morphological Processing of Modern Greek: Functional Decomposition - Cartesian Electronic Lexicon* (in Greek), PhD Thesis, Department of Informatics, University of Athens

Papakitsos E., M. Grigoriadou and A. Ralli. (1998) Lazy tagging with functional decomposition and matrix lexica: an implementation in Modern Greek*, Literary and Linguistic Computing*, 13/4.

Papakitsos E., Grigoriadou M., Ralli A. (2000) "Functional Decomposition and Lazy Word-Parsing in Modern Greek", *Proceedings of NLP2000*, Lecture Notes in Artificial Intelligence 1835, Springer, University of Patras, pp. 27-37

Polaris user's guide (1999) The EuroWordNet Database Editor. Lernout & Hauspie, Belgium, 86 p.

Sproat R.W. (1992) *Morphology and Computation.* MIT Press.

Vendler Z. (1967) *Linguistics in Philosophy.* Ithaca: Cornell University Press

Verkuyl H. (1972) *On the compositional nature of the aspects.* Dordrecht: Reidel.

Verkuyl H. (1989) *Aspectual classes and aspectual distinctions.* "Linguistics and Philosophy", 12, pp. 39-94

Vossen P. (ed.) (1998) *EuroWordNet: A Multilingual Database with lexical Semantic Networks.* Kluwer Academic Publishers, 179 p.

Vossen P., L. Bloksma, H. Rodriguez, Climent S., N. Calzolari, A. Roventini, F. Bertagna, A. Alonge and W. Peters (1998a) *The EuroWordNet Base Concepts and Top Ontology.* EuroWordNet Project LE2-4003, Deliverable D017 D034 D036, 50 p.

Vossen P., C. Kunze, A./ Wagner, D. Dutoit, K. Pala, P. Sevecek, K. Vider, L. Paldre, H. Orav and H. Oim. (1998b) *Set of Common Base Concepts in EurowordNet–2,* EuroWordnet LE-48328, Deliverable 2D001, 41 p.

Vossen P., L. Bloksma, P. Boersma, F. Vardejo, J. Gonzalo, H. Rodriguez. G. Rigau, N. Calzolari, C. Peters E. Picchi, S. Montemagni and W. Peters (1998b) *EuroWordNet Tools and Resources Report.* EuroWordNet Project LE2-4003, Deliverable D021D025, 9 p.

Vossen P. (ed.) (1999) *EuroWordNet General Document,* EuroWordNet (LE2-4003,LE4-8328),Final Document, 108p.

4. TOOLS AND RESOURCES FOR THE ROMANIAN WORDNET

4. 1 Language resources for Romanian

4.1.1 Electronic dictionaries

A. Monolingual dictionaries

➢ ***Wordform Romanian dictionary***

One delivery of the Multext-East project was a large wordform lexicon more than 450,000 entries (which are contracted forms standing for 1,895,837 proper inflected forms). A dictionary entry has the following structure:
word-form <TAB> lemma <TAB> MSD <TAB> comments
where word-form represents an inflected form of the lemma, characterised by a combination of feature values encoded by MSD code (Morpho Syntactic Description); the forth column, comments, which is optional, is currently ignored and may contain either comments or information processable by other tools. The morpho-syntactic descriptions are provided as strings, using a linear encoding. In this notation, the position in a string of characters corresponds to an attribute, and specific characters in each position indicate the value for the corresponding attribute. That is, the positions in a string of characters are numbered 0, 1, 2, etc., and are used in the following way:

- the character at position 0 encodes part-of-speech;
- each character at position 1, 2,..., n, encodes the value of one attribute (person, gender, number, etc.), using the one-character code;
- if an attribute does not apply, the corresponding position in the string contains the special marker '-' (hyphen).

By convention, trailing hyphens are not included in the lexical MSDs. Such specifications provide a simple and relatively compact encoding, and are in intention similar to feature-structure encoding used in unification-based grammar formalisms. When the word-form is the very lemma, then the equal sign is written in the lemma-field of the entry ('=').
The attributes and mainly the values of the attributes were chosen considering only word-level encoding. This is why, some values described in grammar text-books, but assuming compounding (such as compound tenses), were not considered in the MULTEXT-EAST encoding.
Some examples of wordform Romanian dictionary entries are given below.

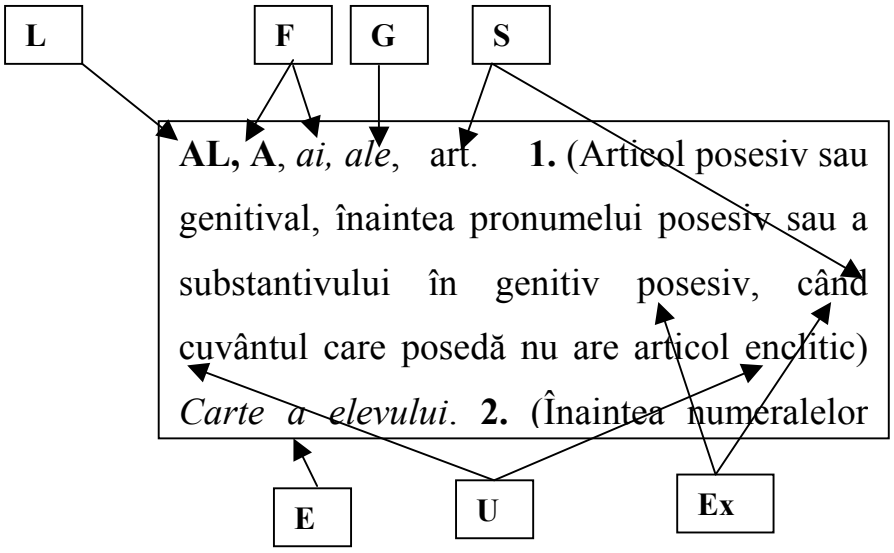
femei femeie Ncfp-n	(women)	-noun
omenesc = Afpms-n	(human)	-adjective
suporta suporta Vmii3s	(support)	-verb
repede = Rgp	(quickly)	-adverb
vouă voi Pp2-pd--------s	(you)	-pronoun
împotriva = Spsg	(against)	-adposition
aceea acela Dd3fsr---o	(that)	-determinaer

For full meaning of the MULTEXT-EAST encoding we refer to Dan Tufis et al. (1997), “Corpora and Corpus-Based Morpho-Lexical Processing” in *Recent Advances in Romanian Language Technology* at http://www.racai.ro/books. This dictionary was intensively used for lemmatisation, tagging and selection of the candidate literals to be

included into the Romanian wordnet of the BALKANET project.

- ***Explanatory Dictionary of Romanian***

The reference dictionary we use is The Explanatory Dictionary of Romanian (DEX,1996), work of the Romanian Academy Institute of Linguistics. This most authoritative lexicographic source for contemporary Romanian was partially digitized and converted into a lexical database (XML encoded) by RACAI under the European Project CONCEDE (Tufiş et al.1999). This core XML-dictionary has been extended to the full content of the printed dictionary by a follow-up project funded by Romanian Academy and finalized within the scope of BALKANET.
Information associated to a head-word in DEX observes the structuring and typographical conventions as shown in Figure 4.1 and explained below.



L - lemma (head-word); F - inflected forms; G - grammatical information; H - homographs; S - sense; Ss - secundary sense; D - definition; V - variants; E - etymology; Ex - examples; U - phrasal unit I - usage information

Figure 4.1: The layout of an entry in DEX

The order of different classes of information is (more often than not) as follows:

1. The lemma form of the head-word
2. Inflected forms, where the case; when the inflected forms have specific grammatical information this is specified together with the corresponding inflected form. If the inflected form is associated with a specific sense of the head-word, then the inflected form is accompanied by a reference to the specific sense;
3. Global grammatical information
4. The explanation(s) of the head-word; they are either direct definitions (grouped on various senses) or indirect definitions (specified as references to other head-words or in case of functional words by explaining their usage)
5. Information on the pronunciation, variants, irregular inflectional paradigms
6. Etymological information

- ***Romanian Dictionary of Synonyms***

Another extremely useful lexical resource we relied on was the Romanian Dictionary

of Synonyms-RDS (Seche, Seche 1997), which was transposed into electronic form by the NLP group at the University A.I. Cuza din Iaşi. The electronic form of RDS has been converted into an XML format so that the same query interface we developed for DEX works also with RDS. This dictionary has been constructed entirely for the sake of BALKANET project.
There are 16 601 entries for verbs and nouns in RDS. Three examples are given below.

6, abandon, abandonare, părăsire
7, a abandona, a lăsa, a părăsi
8, a abandona, a renunţa

The number at the beginning of the line represents the number of the synset in the dictionary.

- ***Morphological, Orthoepic and Orthographic Romanian Dictionary***

Another valuable resource we use is the electronic form of the Morphological, Orthoepic and Orthographic Romanian Dictionary (DOOM, 1982). This dictionary has 67243 entries and offers information about spelling, pronounciation and morphological aspects of the Romanian words.
This dictionary was turned into an electronic dictionary (XML encoded) during the initial phase of the BALKANET project.
Be the example below. It has the following structure:
– liniúță **s. f.** (**sil.** -ni-u-), **g.-d. art.** liniúței; **pl.** liniúțe (“dash”)
Linguistic information associated to a noun lemma, for example, refers to the following aspects:

- accent;
- part of speech (ex. **s. f.**) ;
- hyphenation (**sil.**);
- genitive-dative articled form (**g.-d. art.**);
- plural form (**pl.**).

- ***Romanian Frequency Dictionary***

According to the aims of the project regarding the interlingual coverage, language representativity, maximum usage of the core wordnet and scalability we started a series of quantitative analysis on a very large corpus made of several novels and a collection of journalistic texts, collected from the web. The corpus (containing more than 100 million words) was automatically tagged, lemmatized and the content words of interest (common nouns, verbs, adjectives and adverbs) were counted and sorted according to their frequency. We extracted this way, a list of more than 30,000 Romanian lemmas. Based on the frequency in the running texts, this list was divided into three parts, corresponding to the first 10,000 most frequent lemmas (I), the next most frequent 10,000 lemmas (II) and rest of the lemmas (III).
In deciding which is the most important subset of a lexical stock for a language, the frequency in running texts is considered by many lexicographers to be a very subjective criterion. Among the strongest arguments they would come with is the volume and representativity of the texts included into the corpus subject to the quantitative analysis. With more and more texts available on the net, the size of the data is not anymore a significant issue, but the representativity remains a systematic complain. The exact definition of what representative texts should be included into a corpus for quantitative data analysis is a long-standing debate and we won’t get into this. Considering that our data consisted, almost entirely, of journalistic texts, the

representativity issue could certainly be raised. The Frequency Dictionary of Romanian Words–FDRW (Julliand et all., 1965) published long time ago, based on a balanced corpus of 500,000 words of Romanian literature, legal texts, poetry and journalism contains a list of most frequent 5,000 lemmas. In spite of being quite contested, it is still used by many Romanian linguists as a reference. The comparison we made revealed that most of the 5000 words in FDRW were also in our list, although not with the same frequency ranges.
As frequency in running texts is a disputable criterion for deciding what words should be encoded into a core dictionary/thesaurus/ontology we considered that this criterion should be complemented with others, less controversial in the world of traditional lexicography.
Among the criteria one could find pleas for, we opted for two that we could easily turn into operational selectors. The one is the number of senses a headword would have in a reference dictionary. The second one is the number of word definitions that use the headword in case. A third criterion, not considered yet, might be the number of derivatives of a given headword (this last criterion is preferred by most Romanian etymologists).
In this phase of the BALKANET project we concentrated our attention to the Romanian nouns and the experimental data reported below refers to nouns. Since the technical procedures do not depend on the specific part of speech, the same would apply for verbs, adjectives and adverbs.
Considering only the first two frequency ranges described above (the first most 20,000 words in the journalistic corpus) we extracted from our Explanatory dictionary more than 8000 entries for nouns and nominal compounds (accounting for almost 35,000 senses) so that the definitional productiveness DP (the number of sense definitions a noun participates in) was at least 3. The list was sorted according to the definitional productivity. The frequency dictionary of nouns has been constructed for the purpose of the BALKANET project and for later up-scaling of the final deliverable of the project .

Noun	**Definitional productivity**	**Number of definitions**	**$FRECV_{range}$**
acțiune	2279	13	I
persoană	1979	9	I
parte	1882	94	I
formă	1286	21	I
obiect	1204	16	I
fapt	1044	11	I
apă	743	29	I
. . .	. . .	. . .	. . .
rasism	3	1	II

Table 4.1: scoring the headword candidates

B. Bilingual and multilingual dictionaries

From multilingual parallel corpora and using our translation equivalents extraction program (Tufiş, Barbu 2001a, 2001b) we constructed a bilingual Romanian English dictionary (also XML-encoded) of about 8000 entries. This bilingual lexicon has been hand validated and extended with new entries from several public domain sources.
We also conducted experiments of translation equivalents extraction on the "1984" multilingual corpus containing six translations of the English original: Estonian,

Hungarian, Slovene, Czech, Bulgarian and Romanian. This corpus was developed within the Multext-East project, published on a CD-ROM (Erjavec *et al*. 1998) and recently improved within the CONCEDE project. TRACTOR-*TELRI* (www.tractor.de) distributes this newer version. The lexicons contained all parts of speech defined in the MULTEXT-EAST lexicon specifications except for interjections, particles and residuals. Table 4.2 shows the evaluation results for four languages, where we found voluntary native speaker evaluators.

Bitext	ET-EN	HU-EN	RO-EN	SI-EN
Entries	1 911	1 935	2 227	1 646

Table 4.2: Partial evaluation of the BASE algorithm after 4 iteration steps

Although these dictionaries were built before the BALKANET project, we plan to use them for the validation of ILI projection of the monolingual wordnets of the BALKANET project.

4.1.2 Corpora

Within the Multext-East and TELRI European projects (Erjavec et al. 1997), (Tufiş, Bruda, 1997), (Tufiş et al. 1997, 1998, 1999) there were created one 7-language heavily annotated parallel corpus based on Orwell's famous novel "1984" and one 25-language heavily annotated parallel corpus based on Plato's "The Republic". The annotation initially used was TEI compliant, but it was later on converted into CES (Ide, 1998). These are two relatively small corpora (about 110,000 tokens in each language) but given the accuracy of tagging and interlingual sentence alignment (hand validated) they were extremely useful for various applications ranging from building language models for morpho-syntactic tagging (Tufiş, 1999) and document classification (Tufiş et al., 2000) to automatic sense discrimination (Erjavec et al., 2001).
Besides the multilingual corpora we constructed two other much larger monolingual corpora: a literary corpus based on various novels (containing about 1,500,000 tokens) and a journalistic corpus (containing more than 100,000,000 tokens). Both corpora were automatically tokenized, tagged and lemmatized.
These corpora were developed within other projects but they will be extremely useful for the validation phase of the BALKANET project.

4.2 Tools

- ***Tokenizer***

The recognition of multiword expressions as single lexical tokens, and the splitting of single words into multiple lexical tokens (when it is the case) is generically called text segmentation and the program that performs this task is called segmenter or tokenizer.

We use Philippe di Cristo's multilingual segmenter MtSeg (http://www.lpl.univ-aix.fr/projects/multext/MtSeg/) developed for the MULTEXT project. The segmenter comes with tokenization resources for many Western European languages, further enhanced in the MULTEXT-EAST project (Erjavec and Ide, 1998; Dimitrova *et al.*,

1998; Tufiş *et al.*, 1998) with corresponding resources for Bulgarian, Czech, Estonian, Hungarian, Romanian and Slovene. The segmenter is able to recognize dates, numbers, various fix phrases, to split clitics or contractions (where the case) etc.

- ***Sentence aligner***

We use a slightly modified version of the Gale and Church's CharAlign sentence aligner (Gale and Church, 1993).

- ***Tagger***

We use a tiered-tagging approach with combined language models (Tufiş, 1999) based on TnT (Brants, 2000), a trigram HMM tagger. This approach has been shown to provide for Romanian an average accuracy of more than 98.5%. The tiered-tagging model is based on using two different tagsets. The first one, which is best suited for the statistical processing, is used internally while the other one (used in a morpho-syntactic lexicon and in most cases more linguistically motivated) is used in the tagger's output. The mapping between the two tagsets is in most cases deterministic (via a dictionary lookup) and in the rare cases where it is not, a few regular expressions may solve the non-determinism. The idea of tiered tagging is working not only for very fine-grained tagsets, but also for very low-information tagsets, such as those containing only part of speech. In such cases the mapping from the hidden tagset to the coarse-grained tagset is strictly deterministic. In (Tufiş, 2000) we showed that using the coarse grained tagset directly (14 non-punctuation tags) the best accuracy was 93%, while using a tiered tagging and combined language model approach (92 non-punctuation tags in the hidden tagset) the accuracy was never below 99.5%.

- ***Translation Equivalents Extraction Program***

From the multilingual parallel corpora mentioned before and using our translation equivalents extraction program (Tufiş, Barbu 2000, 2001a, 2001b) we constructed a bilingual Romanian English dictionary (also XML-encoded). This bilingual lexicon has been hand validated and extended for the purpose of BALKANET project with new entries from several public domain sources.

These 4 tools were developed for the purpose of other projects but we used and will use them also in BALKANET.

4.2.1 Tools developed exclusively for the purpose of building the Romanian Wordnet.

All the above-mentioned resources have been integrated by means of a series of tools developed for the purpose of the BALKANET project. They are user-friendly and allow for editing and mapping the Romanian synonymy series in RDS to the sense definitions in DEX and ILI records from EuroWordNet. The output of these tools is further subject to primary local consistency checks (such as detecting word sense appearing in more than one synset) and generated as an XML-encoded file appropriate for import in VisDic.

- ***The editor for building synsets for the commonly agreed ILI concepts***

We have thus assembled the basic linguistic material that the lexicographer should use in making the decisions (linking) necessary for building the core Romanian wordnet. All this information is currently available in a java-based editor, showing in different frames, the following information (see figure 4.2):

- the list of the base concepts (upper-left frame), identified by the ILI record and an English word in the synset mapped on this concept (ex. *life_3_03941565-n*)
- the synset (life_3 living_1), its gloss and top-ontology description, possible translations and association boxes (right-upper frame)
- the numbered sense definitions from the Explanatory Dictionary of Romanian for the selected translation (left-lower frame);
- synonyms of the selected Romanian translation word (right-lower frame)
- pop-up menus for selecting the relevant sense numbers and the equivalence relation to the ILI concept.

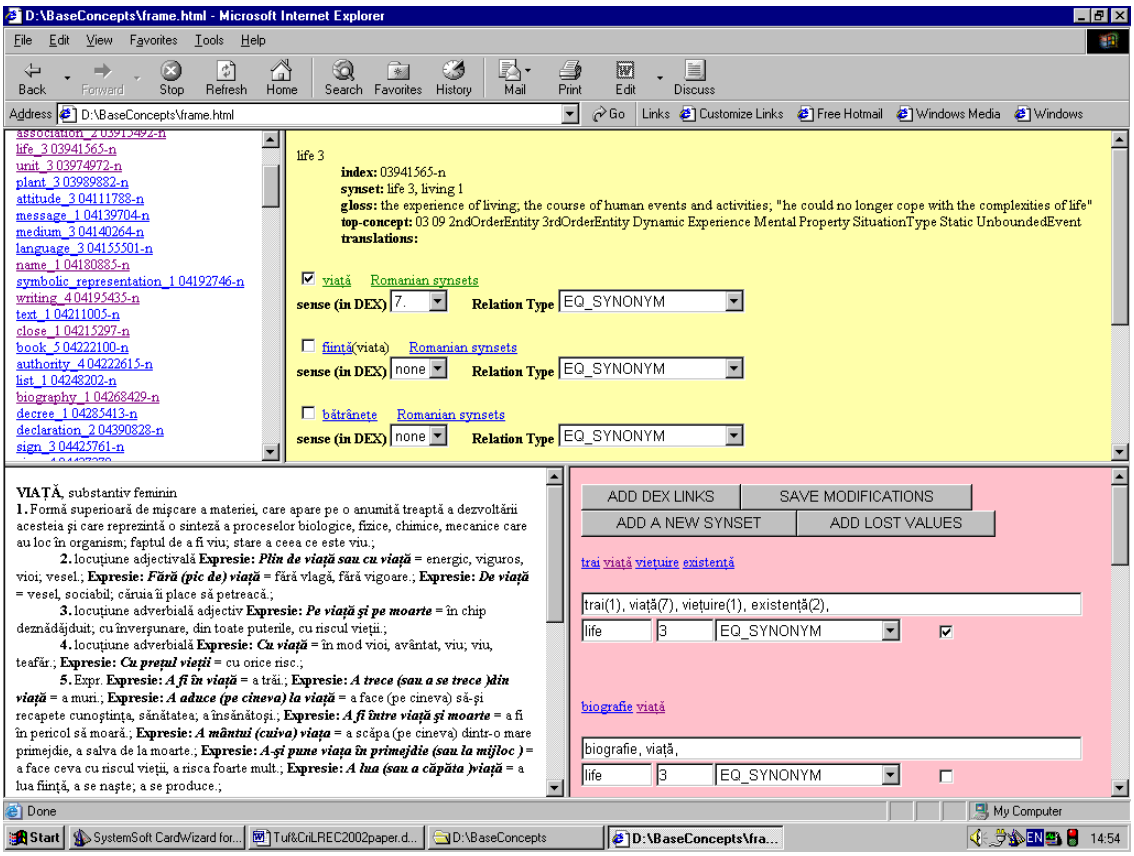


Figure 4.2: The editor for building synsets for the base meanings

➢ ***The editor for gloss assignment***

Once the synsets were constructed and mapped onto base concepts, the second phase was to add a Romanian gloss to each Romanian synset. In the vast majority of cases, the definitions extracted from DEX corresponding to the senses in a synset were different in wording so, the lexicographers had to chose the best definition, closest to the definition of the corresponding base concept. Figure 4.3 shows that the base concept 08232464-n corresponding to the 5th sense of the English word *register* (a book in which names and transactions are listed) corresponds in Romanian to the synset (catastif_1 condică_1 registru_1). The selected senses for the three Romanian words have in DEX different definitions. By checking the box to the right of the third definition (lower frame in Figure 4.3) the lexicographer decided that the definition given to *registru_1* is the one to be attached to the synset.

It is worth mentioning that during the gloss assignment phase it became apparent that several synsets were not correct, requiring modifications. In some cases, the Romanian Explanatory Dictionary includes under the same definition two senses that

are differentiated in ILI as two distinct concepts. In such cases, the general strategy was to split the Romanian definition and attach the relevant part as a gloss.

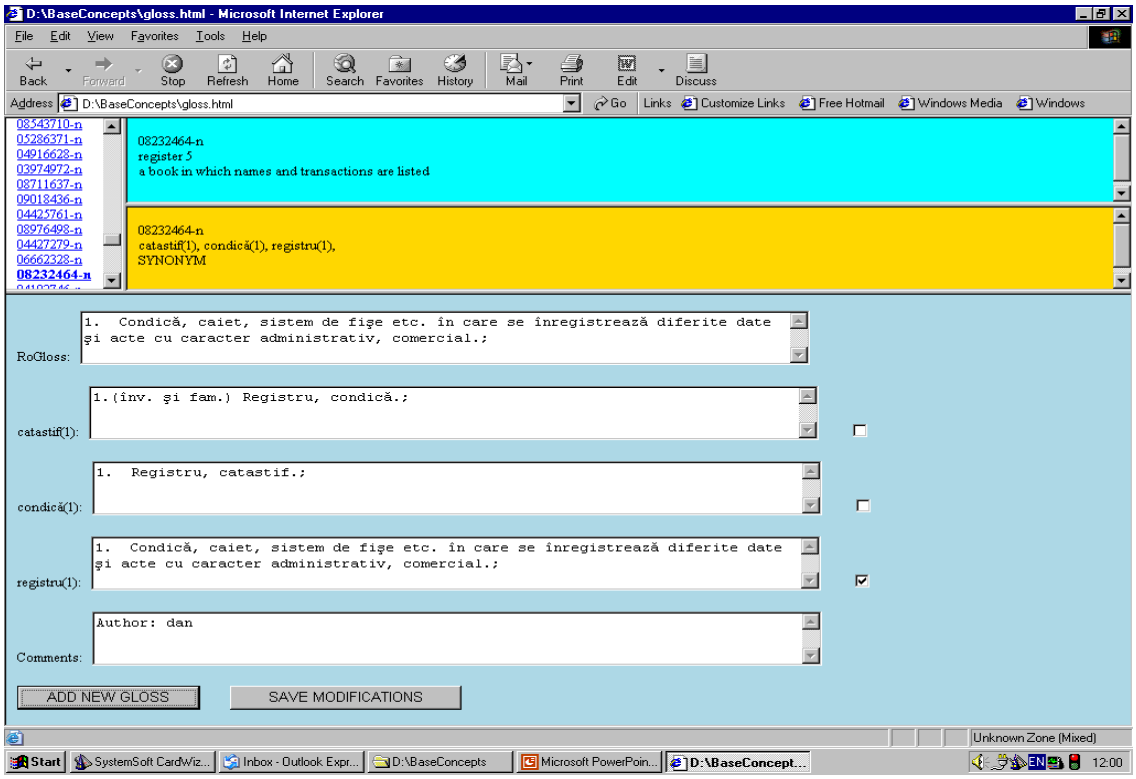


Figure 4.3: The editor for gloss assignment

➢ ***The conflict finder***

Another tool is the one for consistency checking, which we applied for all the monolingual wordnets in the BALKANET project. The errors are displayed in an explicit way so that the lexicographers may revise the wordnets for appropriate actions. The types of errors checked are the following:

- all literals appearing in a synset should have attached a sense number
- no sense (literal and sense number) should appear in two or more synsets
- each synset should have an equivalence relation to a unique base concept.

This function, being of general interest has been imported into the newest version of VisDic.

Figures 4.4 shows the first pages for the language specific error-reports.

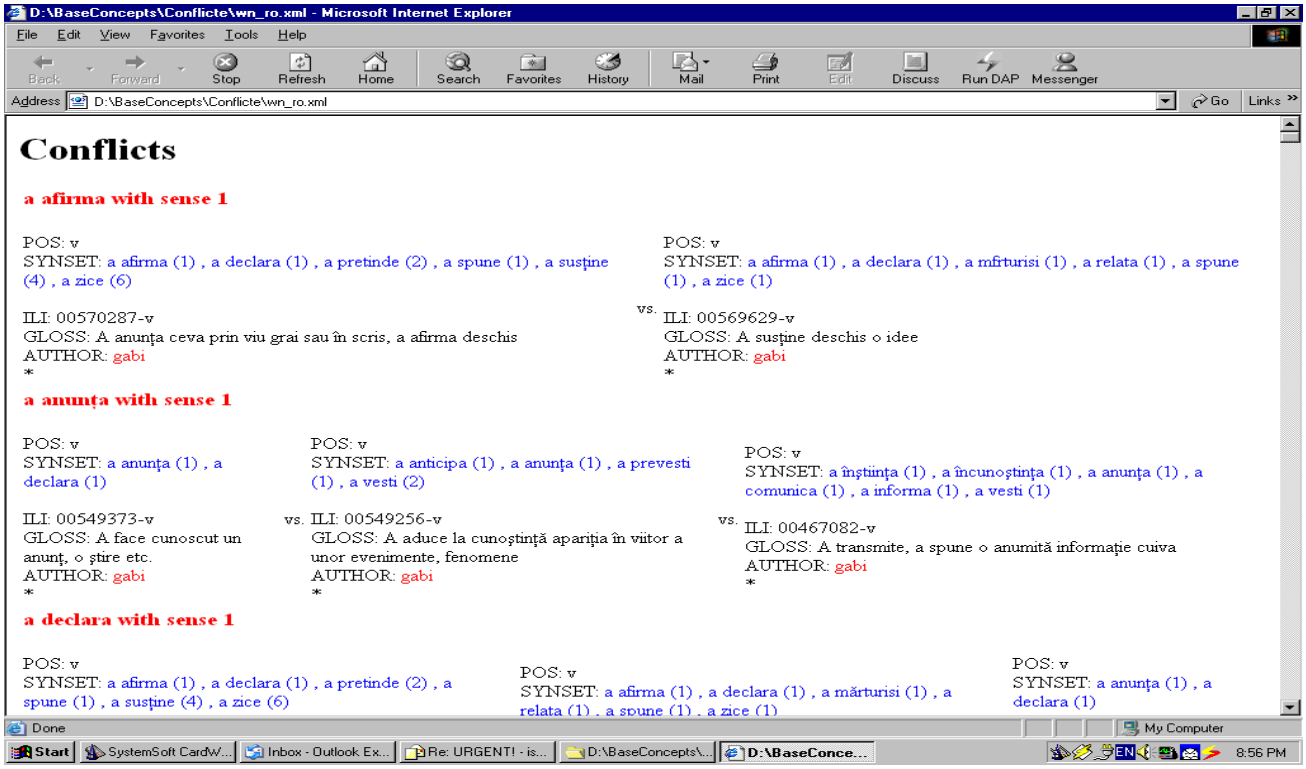


Figure 4.4: Romanian conflict file

References:

Brants, T. (2000). "TnT – A Statistical Part-of-Speech Tagger", in *Proceedings ANLP-2000,* 2000, Seattle, WA.

Dimitrova, L., Erjavec, T., Ide, N., Kaalep, H., Petkevic, V. and Tufiş, D. (1998). "Multext-East: Parallel and Comparable Corpora and Lexicons for Six Central and East European Languages", in *Proceedings ACL-COLING'1998*, Montreal, Canada, pp. 315-319.

Erjavec T., Ide, N. (1998). "The Multext-East corpus", in *Proceedings LREC'1998*, Granada, Spain 1998, pp. 971-974.

Erjavec T., Ide N., Tufiş D.(1997). Encoding and Parallel Alignment of Linguistic Corpora in Six Central and Eastern European Languages" in Michael Levison (ed) *Proceedings of the Joint ACH/ALL Conference* Queen's University, Kingston, Ontario, June 1997 (also on http://www.qucis.queensu.ca/achallc97)

Erjavec, Tomaz, Ann Lawson & Laurent Romary. 1998. *East Meet West: A Compendium of Multilingual Resources.* TELRI-MULTEXT EAST CD-ROM, 1998, ISBN: 3-922641-46-6.

Erjavec T., Ide N., Tufiş, D.(2001) *Automatic Sense Tagging Using Parallel Corpora*, in Proceedings of the 6th Natural Language Processing Pacific Rim Symposium, Tokyo, Japan, 27-29 November, pp. 212-219, 2001

Gale, W.A. and Church, K.W. (1993). "A Program for Aligning Sentences in Bilingual Corpora", in *Computational Linguistics*, 19/1, pp. 75-102.

Ide, N. (1998) *Corpus Encoding Standard: SGML Guidelines for Encoding Linguistic Corpora* First International Language Resources and Evaluation Conference, Granada, Spain. See also http://www.cs.vassar.edu/CES/.

Julliand, A., Edwards P.M.H, Julliand I. (1965). The Frequency Dictionary of Rumanian Words. *Mouton & CO.,* London-The Hague-Paris, 1965

Seche L., Seche M.(1997). *Dicţionarul de sinonime al limbii române.* Univers Enciclopedic, Bucureşti, 1997

Tufiş D., Şt. Bruda (1997). Structure Markup in CES and Preliminary Statistics on Romanian Translation of Plato's "Republica", *Proceedings of International Seminar on Encoding,* Ljubliana, February, 1997, also in *TELRI News*, nr. 5, May, 1997.

Tufiş, D. (1999). Tiered Tagging and Combined Classifiers In F. Jelinek, E. Nöth (eds) *Text, Speech and Dialogue*, Lecture Notes in Artificial Intelligence 1692, Springer.

Tufiş D., Barbu A.M., Pătraşcu V., Rotariu G., Popescu C. (1997). Corpora and Corpus-Based Morpho-Lexical Processing, in Tufiş D., P. Andersen (eds.) *Recent Advances in Romanian Language Technology*, Editura Academiei, 1997.

Tufiş, D., Ide, N., Erjavec, T. (1998). "Standardized Specifications, Development and Assessment of Large Morpho-Lexical Resources for Six Central and Eastern European Languages", in *Proceedings LREC'1998*, Granada, Spain, 1998, pp. 233-240.

Tufiş, D., Rotariu, G., Barbu, A.M. (1999) TEI-Encoding of a Core Explanatory Dictionary of Romanian. In Kiefer, F. and Pajzs J. (eds.) *Papers in Computational Lexicography*, Hungarian Academy of Sciences, 1999, pp. 219-228

Tufiş D., Popescu C., Roşu R.(2000 a): Automatic classification of documents by random sampling in *Proceeding of the Romanian Academy*, Series A, vol 1, no. 2, p. 18-28, 2000

Tufiş, D. (2000 b). Blurring the distinction between machine readable dictionaries and lexical databases. R*esearch Report, RACAI-RR56*, 1999

Tufiş D., Barbu A.M.(2001a) *Computational Bilingual Lexicography: Automatic Extraction of Translation Dictionaries*, in International Journal on Science and Technology of Information, Romanian Academy, ISSN 1453-8245, Vol.4, No.3-4, 2001, pp.325-352

Tufiş D., Barbu A.M.(2001b) *Extracting multilingual lexicons from parallel corpora*, in Proceedings of the ACH-ALLC conference, New York, 12-17 June, 2001.

5. TOOLS AND RESOURCES FOR THE SERBIAN WORDNET

5.1 Lexical resources

- ***Serbian morphological electronic dictionary***

The construction of Serbian morphological dictionary using the methodology developed by M. Gross at LADL (Gross, 1989) has started several years ago. A lot of time was spent in defining precisely for this purpose the notion of Serbian lemma, as well as in morphological classification of Serbian inflective words (Vitas, 1993; Krstev, 1997; Vitas, 2000; Vitas, 2001). This process produced for inflective words the following number of different classes:

category	number of classes
pronouns	22
numbers	13
nouns	180
adjectives	40
verbs	330

The actual production of Serbian morphological dictionary in format suitable for use with Intex, the integrated environment for dictionary production and corpus exploitation (Silberztein, 1993), has started at the end of 2000 by D. Vitas and C. Krstev. The amount of lexical coverage in Serbian morphological dictionary at the beginning of BalkaNet project and now is represented in the following table:

category	number of lemmas before BalkaNet	now	
		N. of lemmas	N. of inflective forms
verbs	3000	14000	410000
adjectives	5500	10000	120000
nouns	0	7000	35000
pronouns	120	120	120
numbers	190	190	190
adverbs	800	800	800
prepositions, conjunctions, etc.	325	325	325

The number of lemmas corresponds actually to the number of entries in the dictionary of type DELAS, while the number of inflective forms corresponds to the number of entries in the dictionary of type DELAF. A few entries from the Serbian dictionary DELAS are:

bombardiranxe,N300+VN+DerIratiOvati
bombardovanxe,N300+VN+DerOvatiIrati
izranxavlxen,A1+PP
nad,PREP+p4
nad,PREP+p6
nigde,ADV+Ek

nigdje,ADV+Ijk
docyepati,V601+Perf+It+Ref

One entry in DELAS dictionary consists of lemma followed by codes describing its features. For instance, codes in upper case following the comma correspond to part of speech (N for nouns, V for verbs, etc.) while number following this code denotes the inflectional class. Codes following the plus sign describe lemma's various features. For instance the code +Ek can be applied to lemmas belonging to different part of speech and it designates that the lemma it is applied to belongs to the ekavian pronunciation. The other codes are specific to certain part of speech: codes +Perf+It+Ref applied to the verb docyepati mean that this verb is perfectiv, intransitive and reflexive, the code +p6 applied to the preposition nad mean that it is used with instrumental, and code +VN applied to the noun bombardovanxe means that it is gerund or verbal noun.

A few entries from the Serbian DELAF dictionary are:

crtanxa,crtanxe.N300+VN:ns2q:np1q:np1q:np4q:np5q
crtanxe,crtanxe.N300+VN:ns1q:ns4q:ns5q
crtanxem,crtanxe.N300+VN:ns6q
crtanxima,crtanxe.N300+VN:np3q:np6q
crtanxu,crtanxe.N300+VN:ns3q
pevasmo,pevati.V1:Axp:Ixp
pevaste,pevati.V1:Ayp:Iyp
pevala,pevati.V1:Gsf:Gpn
pevacxe,pevati.V1:Fzs:Fzp
pevana,pevati.V1:Tfs:Tnp

The entry in DELAF dictionary is more complex. For every inflective form it contains its corresponding lemma, its part-of-speech code and inflective code. The codes after the colon contain the inflectional information. For instance, the code :ns6q associated to the inflective form crtanxem of the lemma crtanxe means that this noun is of neuter gender (n), it is in singular form (s), in instrumental case (6) and it is inanimate (q).

The separate dictionary of toponyms has also been produced with various semantic features added to each entry. This dictionary has not yet been incorporated in DELAS/DELAF system. The excerpt from this dictionary for Greece and Athene, that includes the nouns for a man and a woman from Greece, and two possesive adjectives is:

Grk,.Nms+Hum+Isogr+LngEL
Grkov,.AP+Hum+IsoGr+LngELG+DER(Grk)
Grkinya,.Nfs+Hum+IsoGR+LngEL+DER(Grk)
Grkinyin,.AP+Hum+IsoGR+LngGR+DER(Grkinya)
grcxki,.NA+LngEL
Atina,.Nfs+PR+Top+PGgr+PVgr+PGr4+IsoGR
atinski,.AR+Top+PGgr+PVgr+PGr4+IsoGR+DER(Atinski)
Atinyanin,.Nms+Hum+IsoGR+LngGR+DER(Atina)
Atinyaninov,.AP+Hum+IsoGr+LngELG+DER(Grk)
Atinyanka,.Nfs+Hum+IsoGR+LngEL+DER(Atina)
Atinyankin,.AP+Hum+IsoGr+LngELG+DER(Grk)

The early version of this dictionary has been deposited in Tractor, the TELRI Research Archive of Computational Tools and Resources.

➢ ***Serbian machine readable dictionaries***

- Serbian translation of *Dictionary of Computing*, Oxford University Press, 1986 (deposited at Oxford Text Archive)
- Dictionary of verbs with valencies that contains the most frequent 700 hundred Serbian verbs. One excerpt from this dictionary for verb vezivati (*to tie/to fasten)* is

VEZIVATI,
vezujem, nesvrsx. obav. (kompl.) prel.
1 a) A(zx, n)
b) A(zx, n)+I(n)
v) A(zx, n)+A-za(zx, n)
g) A(n)+D(zx)
2 (zaduzxiti koga cyme)
A(zx)+I(n)
3 (pridobiti koga cyme)
A(zx)+A-za(zx, n)
3' (vremenski ogranicyiti sxto kome)
A(zx, n)+A-za(zx, n)

- An electronic version of the index of "The Systematic Dictionary of Serbo-Croatian" by R. Jovanovic and L. Atanackovic was produced by scanning and the use of OCR. The dictionary consists of more than 80.000 terms and phrases distributed within 3906 nests.

pabircyiti 2535
pavilxon 3115, 2685, 2693, 2769, 2997
pavit 3547
pavlaka 430, 2591
Pag 3228
paganac 1510, 1518
paganizam 1510, 1518
paganin 1510, 1518
paganka 1510, 1518
paganska bozxanstva 1510
paganska religija 1510

The production of different tools is now in progress in order to:

⇒ Reconstruct the nests from the index;
⇒ Separate simple terms from compounds; and
⇒ Separate verbs from other word categories (categories are marked neither in the dictionary itself nor in its index). This task, already performed, gave as a result:

pabircyiti 2535
padati 108, 257, 287, 912, 1012, 1458, 1596, 2127, 3218, 3222, 3237, 3248,
pazariti 2116

paziti 757, 776, 1276, 3117
pajati 1114, 2755

5.2 Textual resources

NLP team at the Faculty of Mathematics has collected textual resources for more than 20 years. Some of these resources have been deposited at Oxford Text Archive and Tractor (www.tractor.de). On the bases of these resources the NLP team of the Faculty of Mathematics has initiated the work on:

5.2 Corpora

5.2.1 Corpus of contemporary Serbian

By now, lot of material has been collected for the corpus of contemporary Serbian, covering different areas of use, such as:

- Newspapers, from daily newspapers to weeklies and monthlies covering various topics (politics, health, fashion, religion, etc.);
- Agency news;
- Elementary school handbooks, university books and various manuals (as software manuals, etc);
- Literature, comprising of more than 50 novels and 10 stories collections from Serbian contemporary writers, mainly 20th century (14 of them added after the start of BalkNet project);
- Essays on various subjects (philosophy, sociology, history, law, etc.)
- Translated texts, including novels, short stories, newspapers ("Le Mond Diplomatique"), essays, etc.

Texts were collected in different ways: downloaded from the web, received directly from publishers or writers, they were scanned or retyped. Therefore, and due to their different primary purpose, they are in different encoding and format. Now the process is underway to put all these texts in the same format. The first goal is to unify the various used encoding schemes: from old and obsolete YU-ASCII, to ISO 8859 and Windows code pages, to Unicode (UCS-16 and UTF-8) and SGML entity encoding. All these encoding schemes are used for Latin and Cyrillic alphabet, as both alphabets are in regular use. Our opinion is that for corpus exploitation the alphabet used for printing is not relevant, so this has to be unified also. The program has been produced during the duration of BalkaNet project that will be described in C.1

Besides the alphabet, the texts differ in other aspects of its encoding. Some texts are in rough form, that means without any structural encoding, the others are in HTML format, that does not provide much of the structural possibilities, while the smallest part, but not quite insignificant, is in SGML/XML format in accordance with TEI/CES guidelines. The level of tagging in this last part of texts varies also, from the paragraph level and sentence level to the level in which various phrase components are tagged as well (names, dates, etc.). Our aim is to produce corpus texts in uniform structural encoding at the level of divisions and paragraphs. At the same time the lexical resources and programming tools will be developed that would enable the automatic preparation of texts (see C.4), as well as tagging of some phrase elements (see A.1).

After the beginning of BalkaNet project, two software tools for corpus management and exploitation were acquired and tested: **Bonito** developed by the NLP group at the Faculty of Informatics in Brno, and **IMS** developed at the University of Stuttgart. Although testing proved Bonito to be a good corpus management program that satisfies our needs, due to some technical restrictions of Yugoslav academic network we cannot use it while the server rests in Brno. Therefore, for the time being we plan to use IMS on our own machine. As the IMS is Linux-based command-oriented software we plan to develop for it in some later phase the appropriate interface.

We plan to continue to use Intex which has proven to be a very useful software for various tasks, such as preparation of texts for corpus (automatics sentence and phrase tagging, indexing, disambiguation, etc.) and linguistic investigations on chosen corpus extracts.

5.2.2 Parallel corpora

Compilation of Serbian parallel corpora began with the participation in the TELRI project, and production of CD "East meets West - A Compendium of Multilingual Resources", where can be found, among other resources Plato's *Republic* aligned in 26 languages and Orwell's *1984* aligned in 9 languages, including Serbian. We have continued to collect texts, mainly for French-Serbian parallel corpus where French is source and Serbian target language. Some of the collected texts are tagged to the sentence level and aligned using the Vanilla alignment program (Danielsson, 1997; Krstev, 2000). The result of the alignment is:

*** Link: 1 - 2 ***
<seg id='VernFr.1.1.7.4'>En tout cas, il n'était prodigue de rien, mais non avare, car partout où il manquait un appoint pour une chose noble, utile ou généreuse, il l'apportait silencieusement et même anonymement.</seg> .EOS
<seg id='VernSr.1.1.7.4'>U svakom sluèaju nije bio rasipnik, ali ni tvrdica.</seg>
<seg id='VernSr.1.1.7.5'>Gde god je ne¹to trebalo za neku plemenitu, korisnu ili velikodu¹nu stvar, on je davao æuteæi i neopa¾eno.</seg> .EOS

Since the beginning of the BalkaNet project several texts have been added to the corpus and aligned: Gustav Flober's *Bouvard et Pécuchet* aligned with its Serbian translation, and Jules Vern's *Le tour du monde en quatre-vingt jours* aligned with its Serbian and Croatian translations.

The acquisition of texts from French monthly "Le Mond Diplomatique" has started in March 2001, when the publication of its Serbian translation began. The collecting of both source text (downloading from the "Le Mond Diplomatique" site) and target text (directly from the publisher) continued when the project BalkNet started. The texts are in the phase of preprocessing in order to be aligned.

5.3 Tools

One of the stages of the BalkaNet project was a development of system of programs for preprocessing of lexical and textual tools in order to obtain them in standardized form and to exploit them. Some of the tools are completed and some are stll under development.

- ***Coding scheme conversion program***

One of developed programs is a program for conversion from different encoding schemas (UTF-8, ISO 8859-2 and ISO 8859-5, etc.) into another encoding scheme. The name of the program is konvert and it uses to auxiliary files. In each line of the first file kod-spc.txt the conversion pairs are given - character in input file and character in output file. For instance, for conversion from ISO 8859-5 to internal encoding of corpus and dictionaries:
Ø Sx
ø sx
Ò T
ò t
Ó U
ó u
The other auxiliary file preskoci.txt contains strings of text that should not be converted, which is sometimes useful for copyright notes and alike. The program is invoked with one argument, input directory. All the files from that directory and its subdirectories are scanned and converted into one output file which is given the name of directory. This approach proved useful for file downloaded from Internet.

- ***Conversion of the Dictionary Recnik Matice Srpske into XML format***

A program has been developed for the conversion of a one-volume dictionary of contemporary Serbian, not yet published, into XML format. The program tags all the structural elements of one dictionary entry. The program has been developed for Linux platform, but makefiles are included for Microsoft Visual C++ and CygWin for the use with Windows.

The text of dictionary is prepared in Word. First it is converted in Word 6.0/95 format, and then in HTML format using package wvWare, version 0.6.5. After that, program parse is invoked with two arguments, input file in HTML format and output file in XML format. The output from program parse for the entry propatiti (*to suffer a great deal*) is:

```
<odrednica>
  <glava>pròpatiti</glava>
  <vrsta>Glagol</vrsta>
  <trajnost>svrsxen</trajnost>
  <znacenja>
    <z><i>prezxiveti, izdrzxati veliku patnxu.</i></z>
    <z><i>namucyiti se, postradati. -</i></z>
  </znacenja>
  <izrazi>
    ~ zbog lxubavi.
    Vojska je mnogo propatila od pegavca.
  </izrazi>
</odrednica>
```

Program query has been developed that enables simple search of thus obtained dictionary. It is invoked by: query arguments string. The arguments are:

-u	--ulaz	input file, obligatory
-i	--izlaz	output file, optional, standard output if not present
-p	--prefiksi	list of prefixes that modify the query string, optional
-s	--sufiksi	list of suffixes that modify the query string, optional
-o	--opsirno	verbose output

The option --opsirno specifies the amount of date to be written in the output file after successful search. If prefix and suffix files are not given, program treats string as a dictionary entry. If found, query is enhanced by all of its inflected forms given in the entry text. The search is than repeated and query is performed on the complete dictionary text.

Prefix and suffix files modify the input string. These files contain the lists of prefixes/suffixes that are concatenated with query string to obtain the list of new query strings. The program than continues the search as described before.

Unfortunately, the dictionary itself has not yet been obtained from the publisher so the program was developed on test data.

- ***Fast text scanner***

A fast text scanner was constructed written in JAVA that produces text index without any preprocessing and relying only on available internal memory. The program is very flexible: it works with different coding schemes (including multi-byte codes), allows the definition of words (or word boundaries) and stop-lists in form of regular expressions, definition of sorting order and sorting style (normal and reverse). The complete description of this program is given in the appendix of this document, in a file index.html (directory doc).

- ***Vertical text production***

A program is under development for automatic conversion of the daily newspaper "Politika" downloaded from its web-site into vertical text format as required by corpus management software. This program has to perform the following tasks: first it has to check whether all texts were correctly downloaded as stated in index that accompanies every issue. Then, every paragraph in text is tagged with attributes for issue date, text type (what can automatically be extracted from file name) and text URL. In the last phase, the vertical text is produced in which to every token the attributes are added as inherited from the appropriate paragraph.

- ***Using the Intex concordances independently***

A program is under development for production of INTEX concordances outside the INTEX environment, so that the concordances can be used independently from the system itself.

References

Danielsson, P./Ridings, D. (1997): *Practical Presentation of a "Vanilla" Aligner*, Presentation held at the TELRI Workshop in Alignement and Exploitation of Texts, Ljubljana, February, 1 –2

Gross, M. (1989): La construction de dictionnaires électronique. In: *Annales des télécomunication* 44(1-2), 4-19

Krstev, C. (1997): *One approch to the Text Modelling and Algorithms of its Transformation*. Ph.D. thesis, Faculty of Mathematics, University of Belgrade

Krstev, C./Vitas, D. (2000): *Local Grammars and Structural Derivation in the Meaning Extraction*. In: Proceedings of the 5th TELRI Seminar "Corpora and Meaning Extraction", Ljubljana, September 2000

Vitas, D. /Krstev, C./Pavlovic-Lazetic, G. (2001): Flexible Dictionary Entry. In: *Current Issues in Formal Slavic Linguistics*, Linguistic International, 461-468

Silberztein, M (1993): *Dictionnaires élelctroniques et analyse automatique de text (le system INTEX)*, Paris: MASON

Vitas, D. (1993): *Mathematical Model of Serbo-Croatian Morphology (Nominal Inflection)*. Ph.D. thesis, Faculty of Mathematics, University of Belgrade

Vitas, D./ Krstev, C./ Pavlovic-Lazetic, G./ Nenadic, G. (2000): *Recent Results in Serbian Computational Lexicograaphy*. In: Proceedings of the Symposium Contemporary Mathematics, Faculty of Mathematics, Belgrade

6. TOOLS AND RESOURCES FOR THE TURKISH WORDNET

6.1 Existing Language Resources

A. Monolingual Dictionaries

- **The TDK dictionary**

The edition used for electronic form of the dictionary is published in 1988, by TDK-Türk Dil Kurumu (Turkish Language Society) [1]. The raw data for the electronic form was captured via OCR at Bilkent University during 1994.

The data was then intensively cleaned up and modified by Halici Software, the industrial partner in the TU-LANGUAGE Project supported by NATO Science for Stability Program, in collaboration with Turkish Language Society (TDK). This version was provided back to Bilkent University for use in the TU-LANGUAGE and related research activities [2].

The data are stored in 4 files:

Senses.txt: It contains 55K headwords. When all the entries with different sense numbers are counted, the total is 75K. The file format is

HEADWORD;ENTRY_NO;MULT_NO;SENSE_NO;NUM_EXAMPLES;WORD_TYPE;
MEANING;USAGE_TYPE;CONTEXT

"açık";1;0;1;2;"s.";"Açılmış, kapalı olmayan, kapalı karşıtı";;
"açık";1;0;2;2;;"(Renk için) Koyu olmayan";;
"açık";1;0;3;2;"is.";"Kapalı olmayan yer";;

Multiples.txt: It contains 15K multiword constructs. The file format is

HEADWORD;ENTRY_NO;MULT_NO;MULTIPLE;NUM_SENSES;USAGE_TYPE;
CONTEXT

"açık";1;1;"açık açık";1;;
"açık";1;2;"açık alınla";1;;
"açık";1;3;"(Birine) açık bono vermek";1;"mec.";
"açık";1;4;"açık kapamak";1;;
"açık";1;5;"açık kapı bırakmak";1;;
"açık";1;6;"açık konuşmak";1;;

Examples.txt: It contains 30K examples for the headwords. The file format is

HEADWORD;ENTRY_NO;MULT_NO;SENSE_NO;EXAMPLE_NO;EXAMPLE;AUTHOR

"açık";1;0;1;1;"Açık pencere.";
"açık";1;0;2;2;"Açık sarı saçlı zayıf bir kadın keman çalıyordu.";"Ö. Seyfettin"
"açık";1;0;3;1;"Açık hava sineması.";

Abbrevia.txt: It contains 280 abbreviations used in the dictionary. For example, "Fransızca";"Fr." (abbreviation for French) is a sample entry from this file.

We, as the Turkish WordNet team, updated and restructured the TDK Dictionary files, to make it more useful for the Balkanet Project. There were 3 basic steps of this process:

- Restructuring TDK Format

aslan: 1. Kedigillerden, erkekleri yeleli, yırtıcı, Afrika'da yaşayan, uzunluğu 160 cm, kuyruğu 70 cm ve ucu püsküllü, çok koyu sarı renkli güçlü bir memeli türü, arslan
2. Gürbüz ve yiğit adam
Aslan: 1. Zodyak üzerinde, Yengeç ile Başak burçları arasında yer alan burcun adı
In the original TDK format, it is possible to have two or more entries for a headword. Sense numbers of each entry begin at 1. But there is no such distinction in the WordNet format. So, for compability we restructered the SENSE_NO column in the Senses.txt file. Here are the examples of the old format and the new format:

Old Format
"aslan";1;0;1;0;"is.";"Kedigillerden... "
"aslan";1;0;2;0;;"Gürbüz ve yiğit adam";"mec.";
"Aslan";2;0;1;0;"is.";"Zodyak üzerinde... "

New Format
"aslan";1;0;1;0;"is.";"Kedigillerden... "
"aslan";1;0;2;0;;"Gürbüz ve yiğit adam";"mec.";
"Aslan";2;0;3;0;"is.";"Zodyak üzerinde... "Spelling Checking
We manually checked the spelling of 90 K entries of senses.txt and 15 K entries of multiple.txt

- Normalization

There were vowels with a diacritical mark (â, î, û) in the original files of TDK. The use of these marks is disappearing rapidly, therefore discrepancy occurs between the various electronic resource. In the case of "â" and "î" we did normalization by completely removing the diacritical mark. In the case of "û" a decision is necessary to replace by "u" or "ü". Finally, "û"s in 43 words have been changed to "u" and "û"s in 4 words changed to "ü".

➢ **Synonyms Database**

This electronically available synonyms database was a Bilkent University project deliverable in 1996 [3]. The original format of the database is

400000001,'acar',r
400000001,'güçlü',r
400000001,'becerikli',r

200000003,'acele_etmek',v
200000003,'ivmek',v
200000003,'ivedilenmek',v
200000003,'çabuk_davranmak',v

where the first column is the ID number, the second contains the synset members and the last is the POS of the member. Words having the same ID number are members of the same synset. The script convert.pl is written to convert this format into our standard synset format acar, güçlü, becerikli and acele etmek, ivmek, ivedilenmek, çabuk davranmak.

B. Turkish English Bilingual Dictionary

This resource was built in Computer Research Laboratory at New Mexico State University, in 1999. The original files are separated according to the first letter of the words. There are also files for conjunctions, interjections, lexicalized collocations, compound verbs, and proper nouns. The file format is

akıcı
pos1;a
s1.1;fluid
s1.2;fluent

akıl
pos1;a
s1.1;wise
s1.2;intelligent
pos2;n
s2.1;mind
s2.2;opinion

The total number of single common words in the dictionary is 14436. This number increases to 19392 if all the senses of these words are counted. The file was converted into XML format, to make it VisDic compatible. The converted format of the file is

```
<ENTRY>
        <TR>akıl</TR>
        <TPOS>a
                <SENSE>1<ENG>wise</ENG></SENSE>
                <SENSE>2<ENG>intelligent</ENG></SENSE>
        </TPOS>
        <TPOS>n
                <SENSE>1<ENG>mind</ENG></SENSE>
                <SENSE>2<ENG>opinion</ENG></SENSE>
        </TPOS>
</ENTRY>
```

6.2 Tools

- **Turkish Morphological Analyzer**

The morphological analyzer was developed by using the two-level description of Turkish morphology in 1994 [4]. Recently, the updated version of the analyzer is available .

When the user types a word, e.g., “koyun” (sheep) as the input, the analyzer outputs the possible results with explanations of inflectional and derivational suffixes. In the example below, possible roots are “koyu”(dark), “koy”(bay), “koy”(to put), and “koyun”(sheep).

analyze>
1.[[CAT=ADJ][ROOT=koyu][CONV=NOUN=NONE][AGR=3SG][POSS=2SG][CASE=NOM]]
2. [[CAT=NOUN][ROOT=koy][AGR=3SG][POSS=NONE][CASE=GEN]]
3. [[CAT=NOUN][ROOT=koy][AGR=3SG][POSS=2SG][CASE=NOM]]
4. [[CAT=NOUN][ROOT=koyun][AGR=3SG][POSS=NONE][CASE=NOM]]
5. [[CAT=VERB][ROOT=koy][SENSE=POS][TAM1=IMP][AGR=2PL]]

analyze> bye!

- **Turkish Spelling Corrector**

This corrector has been developed using the two-level transducer for Turkish developed with Xerox software and the correction algorithm described in [5] in 1996. The corrector generates all possible Turkish words that are within a small distance of the given incorrect word, where distance is measured by the number of character insertions, deletions, changes and transpositions. No attempts are done for ranking.

When the user types an input word, e.g., kaleö, which is meaningless in Turkish, the candidate outputs are

1. kale (castle)
2. kalem (pencil)
3. kalen (your castle)

6.3 Development of Language Resources

Our team began working on the Turkish WordNet by translating the English Base Concepts into Turkish and by constructing synsets for these concepts, using electronic resources. We then extracted and refined more than 20K Turkish synsets using several semi-automatic methods. Another step was to specify the next set of concepts to be used by all partners after having completed the BCs. Here are the tools and applications we used:

- **VisDic**

- **TDK in VisDic**

TDK is the acronym of our main monolingual dictionary of Turkish, which is electronically available. The database is in TXT format. Since we constructed the

Turkish BCs in VisDic, we preferred to transform the TDK dictionary into XML format, so that we could use it in VisDic. We used a Perl script for converting the dictionary from TXT format to XML format. .def and .cfg files for the VisDic version of the dictionary were also built.

The XML format of a typical entry is

```
<ENTRY>
        <HEADWORD>agah</HEADWORD>
        <SENSE>1
                <POS>s</POS>
                <SGLOSS>Bilir, bilgili, haberli, uyanık</SGLOSS>
        </SENSE>
        <MULTIPLE>agah olmak
                <MGLOSS>bilgi edinmiş olmak</MGLOSS>
        </MULTIPLE>
</ENTRY>
```

- **Synsets in VisDic**

Since we used VisDic for constructing the Turkish BC WordNet, we transferred all of our resources to the VisDic platform. One of these resources was our synset candidates database. These synsets were derived from our monolingual dictionary and from the online synonyms database we already had. When the modification of the synsets is completed, the XML file in
VisDic will also be updated. An example for the XML format is

```
<SYNSET>
        <POS></POS>
        <SYNONYM>
                <LITERAL>ab<SENSE></SENSE></LITERAL>
                <LITERAL>su<SENSE></SENSE></LITERAL>
        </SYNONYM>
        <ILI></ILI>
        <GLOSS></GLOSS>
</SYNSET>
```

- **Perl Scripts**

As a starting point, our basic electronic lexical resources were our monolingual dictionary database and the synonyms database. We extracted a large number of synsets from these data. We would like to give the exact numbers of synsets we constructed. But it should be noticed that they are drafts yet and need revising. Therefore, the elimination process will decrease the total number of synsets at least by 30%.

Our monolingual dictionary contains about 90.000 headwords. This number also covers different senses of the words. Our idea is to find the words which have only one word as a gloss, so that we can form synsets having two members. For example,

in our dictionary the word *abartı* (exaggeration) has the gloss *abartma*. So the pair (*abartı, abartma)* is a synset. We call them 1+1 synsets, one from the headword and one from the gloss. The same idea is applicable to words that have multi-word glosses where all the words are separated with commas. For example the entry: '*bade: şarap, içki*' (wine) leads to the synset (*bade, şarap, içki*), which is a 1+2 synset.

We have defined our criteria in the form of regular expressions and extracted the matching patterns from the files by coding them in Perl. As a result, we obtained:

Synset type	Quantity
1+1	6642
1+2	3278
1+3	974
1+4	199
1+5	30
1+6	3
Total	11126

Our synset database is pretty clean and well-formatted. We only rearranged it by using Perl, in order to make it more readable. It can be divided into four groups based on the POS of the synset.

POS	# of synsets
Noun	1417
Verb	1223
Adjective	989
Adverb	271
Total	3900

The Perl scripts used in the process and brief explanations are as follows:

candidates.pl: This script has two phases. The first phase takes the candidate wordlist for the subset2 of wordnets and finds all senses of the concepts. Each item in the wordlist is checked for its POS. Matching entries are written to the output file.

The second phase finds the candidates which are immediate hyponyms of BCs. The output file of the first phase is the input file of the second. An ILI number is taken from the file containing the ILI numbers of the BCs, and the line containing this ILI number in the hyperonymy relation is written to the final output file.

countsenses.pl: This program counts the occurrence of the members within the main synset file. The members can be either single words or multi-word constructs. That is, "adam", "adam gibi", and "adamca" are all treated as different concepts. The output is sorted both alphabetically and by count number in descending order in two separate output files.

counttdksenses.pl: This script counts the number of senses of each entry in the TDK dictionary. A column in the main database file of the TDK dictionary, stores the sense number of the entry. The last line of a specific entry gives the number of senses of

that entry. The second database file of the TDK Dictionary contains multi-word constructs. The number of senses of each entry of that database is given as a separate value. The output is stored in a file in alphabetical order.

convert.pl: We had an electronically available synonyms database, which is in the format

400000001;'acar';r
400000001;'güçlü';r
400000001;'becerikli';r

where the first column is the ID number, the second contains the synset members and the last is the POS of the member. Words having the same ID number are members of the same synset. The script convert.pl is written to convert this format into our standard synset format acar, güçlü, becerikli.

convert_tr.pl: The approach we adopted in constructing Turkish BCs was to directly translate the BCs from English. The translation was done manually. Then we transformed our synsets into the standard XML format of VisDic. Since this XML format is common to all wordnets, we took the English BCs XML file as a template and inserted our synsets instead of their English counterparts into the appropriate tags, by using a Perl script.

In the XML format, the middle of an entry contains the synset which is language dependent. The parts before and after the synset part are common to all languages. The “convert_tr.pl” script reads an entry from the English BC file, writes the former part of the entry to the output file, appends the Turkish BC synset to the former part and finally writes the latter part of the read entry. At the end, the line in the output file is an entry of Turkish BCs.

findBC.pl: This short script finds the Base Concepts in the English WordNet file and stores them in a new file. It performs pattern matching based on the fact that BCs have an asterisk mark within the <BASE> tag.

gapBC.pl: In EWN, there are some non-BC concepts whose hyponyms are BCs, which we called “gap BCs”.

The script takes an entry from the BC file and finds its hyperonym in the EWN file. If this new entry is not a BC and not a root, it is written to the output file. The script then proceeds with its hyperonym. If the entry has no hyperonym, then it is a root.

The program uses two subroutines. One of them determines if the input item is a member of the input array. It returns 1 if it succeeds and 0 if not. The other subroutine finds the entry of the given ILI number from the EWN file and returns the entire record as an output.

leaves.pl: This program finds the immediate hyponyms of the BCs. Input files are EWN and a file containing the ILI numbers of the BCs. An ILI number is taken from the file, the line containing this ILI number in the hyperonymy relation is found. If the found entry is not a BC, it is written to the output file.

restructure.pl: Before transforming the dictionary into XML format, the sense numbers of the monolingual dictionary database were rearranged. In the original resource we have, one headword may have more than one entry in the dictionary, and each entry may have more than one sense. So, the sense number of a word is represented by two numbers, e.g. 1.1, 1.2, 2.1, 2.2. But in wordnet format, there is no such distinction. Therefore, we omitted the entry numbers and assigned new sense numbers in ascending order, i.e. 1, 2, 3, 4 respectively in the previous example.

The script reads the headword of a line; if this is not the same as the previous one, the script assigns sense numbers starting from 1. Otherwise, it checks the "multiple number" stored in the file. If "multiple number" is zero, this shows that the entry in question is a headword and has a sense number which follows that of the previous entry having the same headword. The script also assigns a POS to each entry, although this information is not explicitly provided for each entry in the original file.

set_op.pl: This script takes two files as inputs and compares the files. The entry format of the inputs files should be identical. The output files are the intersection of the first and second input file, the difference of the first input file from the second, and the difference of the second one from the first.

synset.pl: This is the basic script we used to derive our synsets from the TDK Dictionary. At the beginning of this section, the motivation behind the method we used is explained with examples.

A line is read from the main database of the TDK Dictionary. If the gloss meets the search criterion, the headword is checked. If it is not a multiword construct, the word and the gloss are written to the output file. If it is a multiword construct, that construct is found from the second database file of the TDK Dictionary and written to the output file together with the gloss.

synset2xml.pl: This PERL script transforms the format of the file which contains more than 20K synset candidates. These synset candidates are stored in a TXT file in the form of w_1, w_2,..., w_n. In order to display our candidate synsets in VisDic, the script reads each line in the TXT file and rewrites the contents in a special XML format we developed for VisDic.

tdk2xml.pl: This script is used to convert our monolingual dictionary to XML format. We have two main database files in TXT format. One of the files contains headwords and their senses and the other file contains multi-word constructs. Each sense of a headword is a separate line, but in XML format the senses of an entry are brought together.

The code reads a line from the input file, writes the appropriate columns of it as a new line to the output file if it is the first sense of a headword. If the line is one of the other senses of the headword, the data is also sent to the output file following the same line. If the entry in the input file is a multi-word construct, the script finds the related construct from the second input file and takes all other data related to that entry from the first input file.

- **Monolingual Dictionary Browser**

This is an application developed for navigating the TDK Monolingual Dictionary of Turkish. The user can search the headword and retrieve the gloss and other features of that word, and also search for a specific word contained in the glosses and retrieve the headwords which meet the search criteria.

By clicking on "Meaning-Word Search" the user can see all glosses which contain the searched word. This feature is extremely useful for automatically extracting relations from the dictionary. Figure 6.1 shows a sample for "Meaning-Word" search.

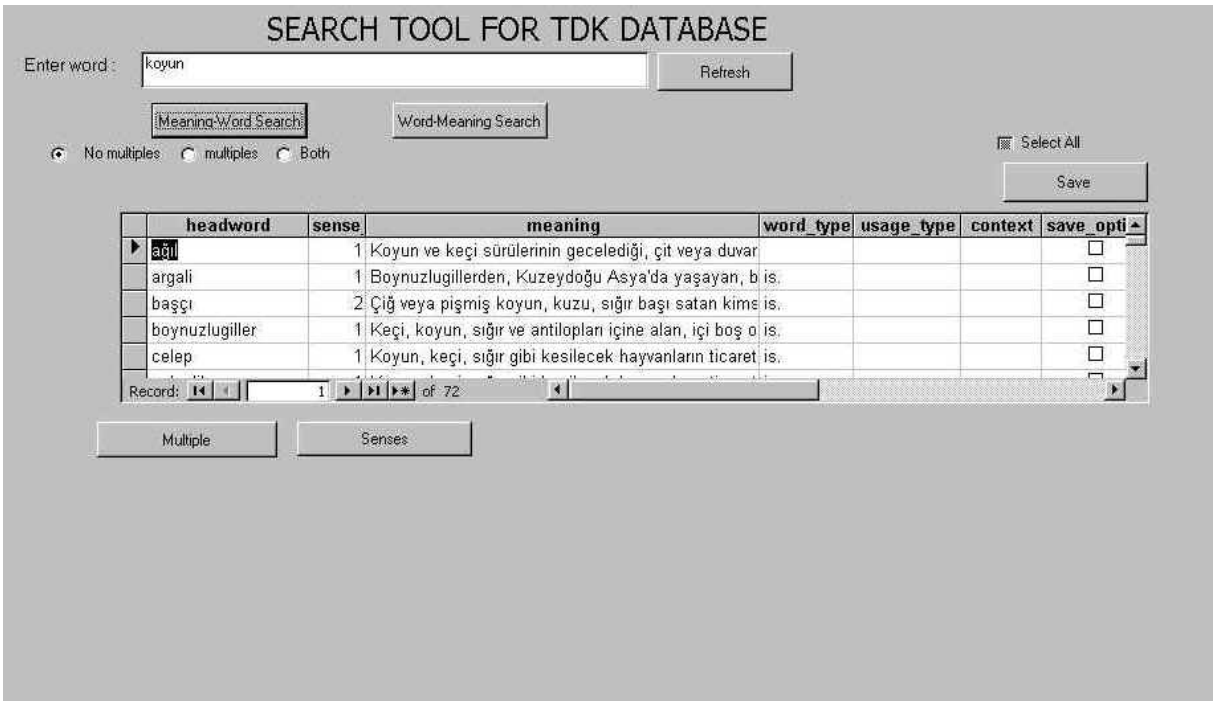


Figure 6.1: A "Meaning-Word" search

The user can select a headword from the search results and view that word's multiword constructs or definitions of their senses. In Figure 6.2., the word "kol" (arm) is chosen from the table and all the multiword constructs and sense definitions are displayed on the tables below the search results table.

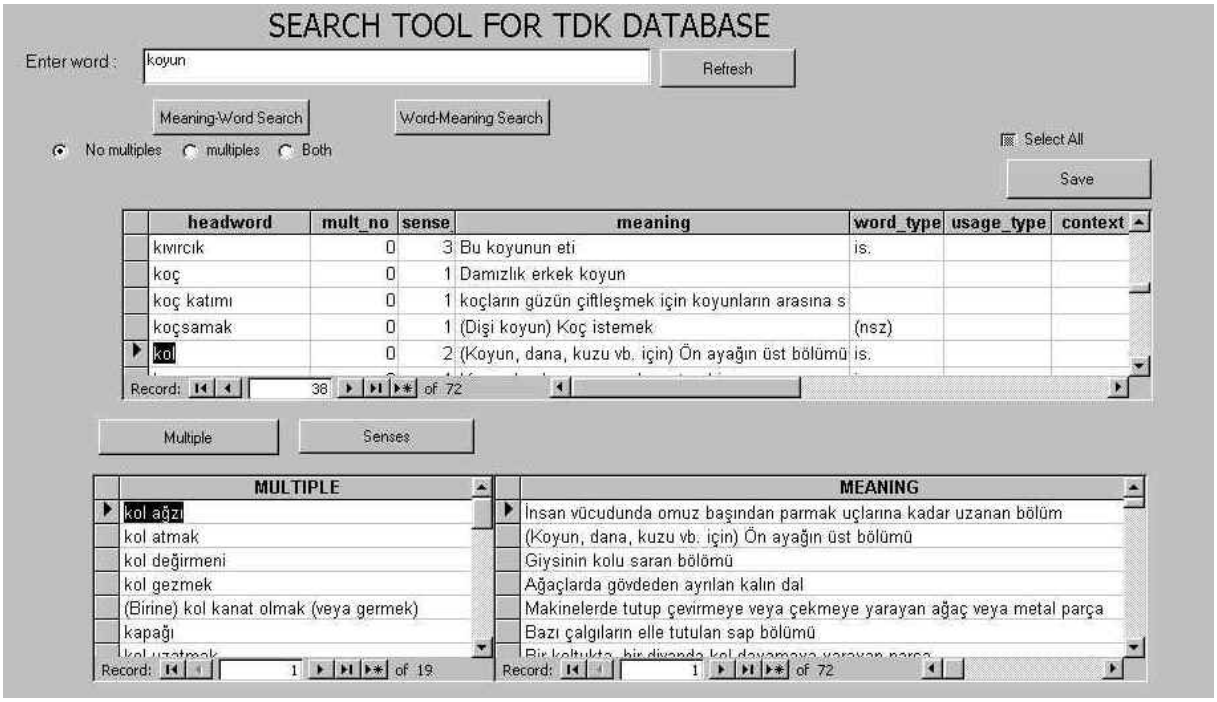


Figure 6.2: Multiword constructs and sense definitions

By typing a word in the "Enter word" window and clicking on "Word-Meaning Search", the tool can be used as a regular dictionary which displays the gloss of the searched word. Figure 6.3. shows a sample "Word-Meaning Search".

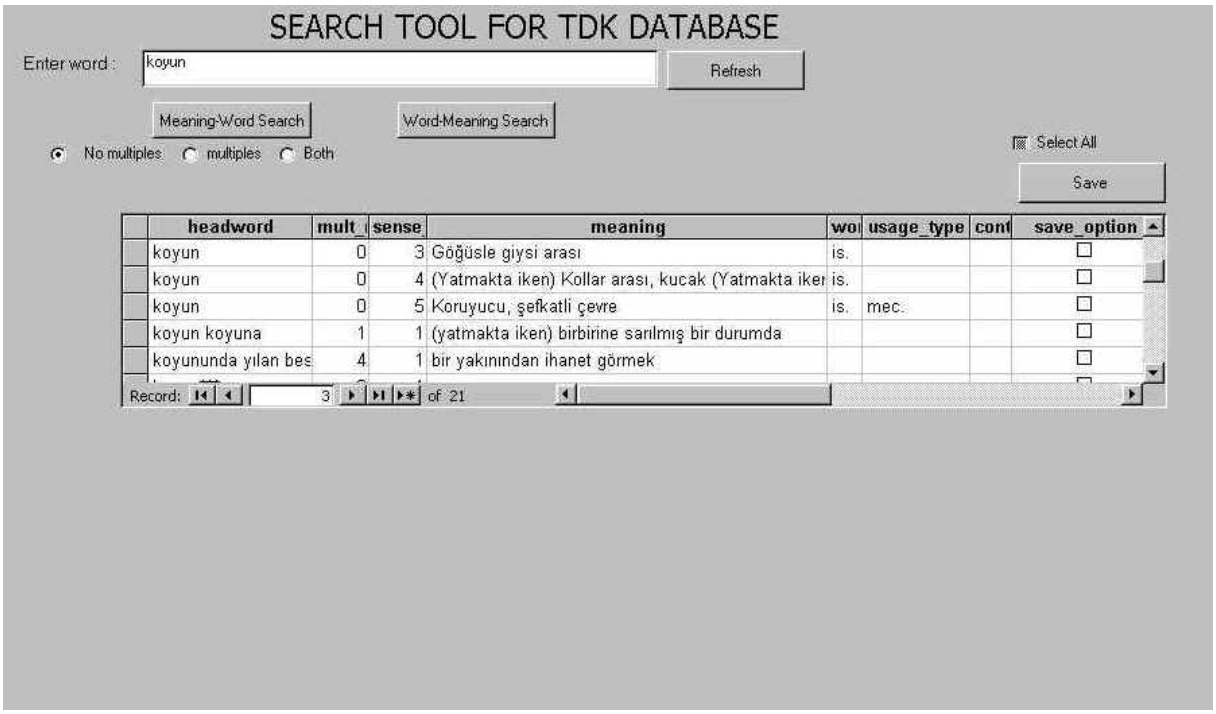


Figure 6.3: A "Word-Meaning Search"

The user can limit a headword search by excluding idioms, or alternatively, view all constructs that contain the searched word. The choice is done by clicking on “No multiples”, “Multiples” or “Both”.

The results of both the “Meaning-Word Search” and the “Word-Meaning” search can be saved as an HTML file. The name of the file is automatically assigned in the form of <word>_meaningword.htm or <word>_wordmeaning.htm where <word> denotes the searched word. Figure 6.4. shows a “Meaning-Word Search” for the word “koyun” (sheep) in HTML format.

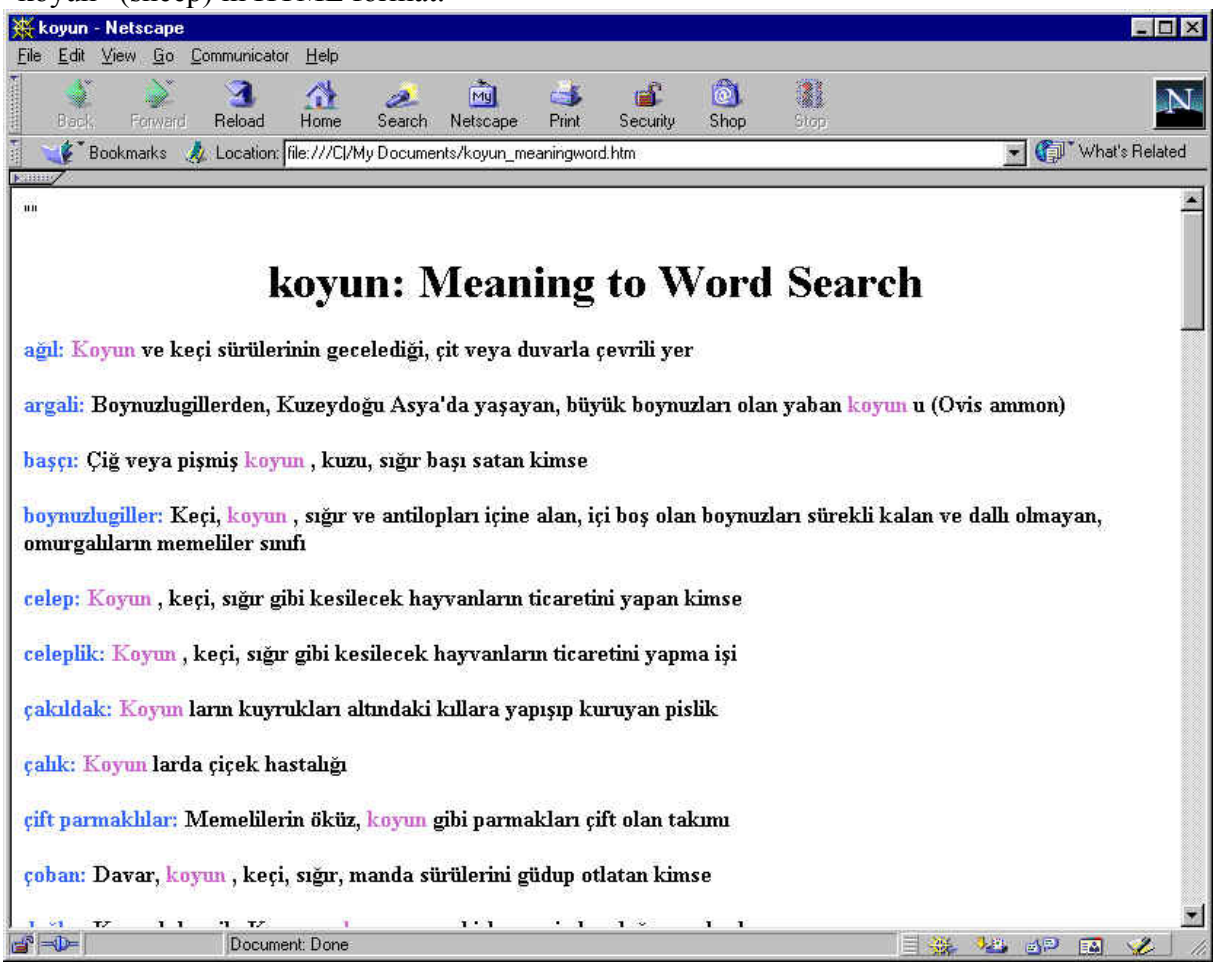


Figure 6.4: Search results in HTLM format

- **Synset Merge Tool**

We have derived approximately 20K synsets from our resources, but they require manual revision. Some of the synsets are subsets of some others, or some synsets have common members. In order to deal with synset modification operations, we developed an MS Access application which we called the “Synset Merge Tool”.

The user types in a word in the combo-box or s/he chooses it from the list and presses the “Synsets” button. All synsets containing the word are listed in a table. We also included the synsets of the 1310 BCs so that they can be enriched with new members,

if necessary. To distinguish BCs from others, we used the ILI numbers as an indicator. Figure 6.5 shows the synsets of the word “abartı” (exaggeration)

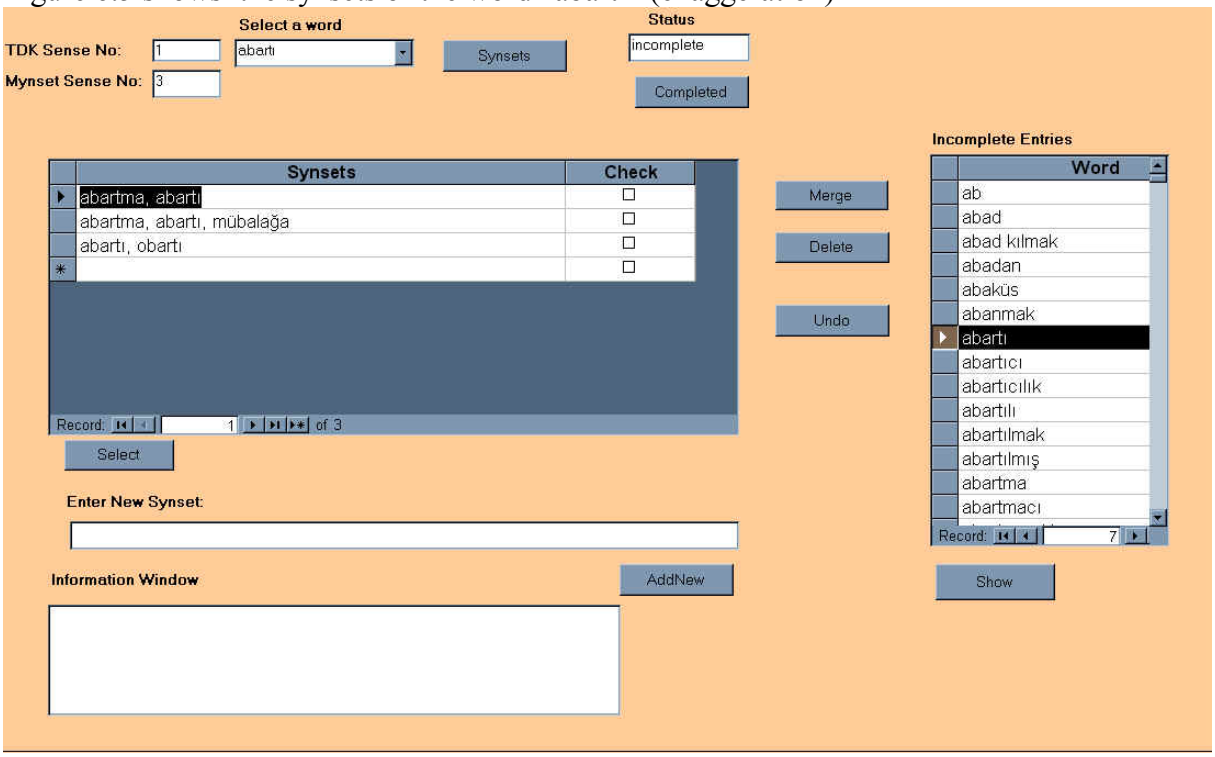


Figure 6.5: Synsets of the word “abartı” (exaggeration)

There is one checkbox for each synset on the table. The user checks the synsets s/he wants to process and presses one of the buttons at the bottom of the window. The “Merge” button merges two or more synsets into a new one. The checked synsets are deleted from the database and the new, merged synset is added. When the user wants to delete the checked synsets, s/he presses the “Delete” button. Information regarding modified synsets is given in the “Information Window”. Figure 6.6. shows a sample operation.

There are boxes displaying the number of the senses of the word and the number of occurrences of the word in the synset file. In the ideal case, the number of occurrences of a word should be less than or equal to the number of senses of that word. If not, some of the synsets containing the word have to be merged or deleted. Therefore, the user can decide to merge, delete or modify the synsets with the help of these criteria.

The “Status” box shows whether the modifications on an entry are completed or not. By default, all entries have the value “incomplete”. When the user finishes dealing with an entry s/he presses the “Completed” button which helps keep track of completed entries.

The “Incomplete Entries” window allows the user to see the remaining work to be done. All synset members who are yet to be processed are listed in this window. If the status of an entry is turned to “complete”, this entry is automatically removed from the “Incomplete Entries” window.

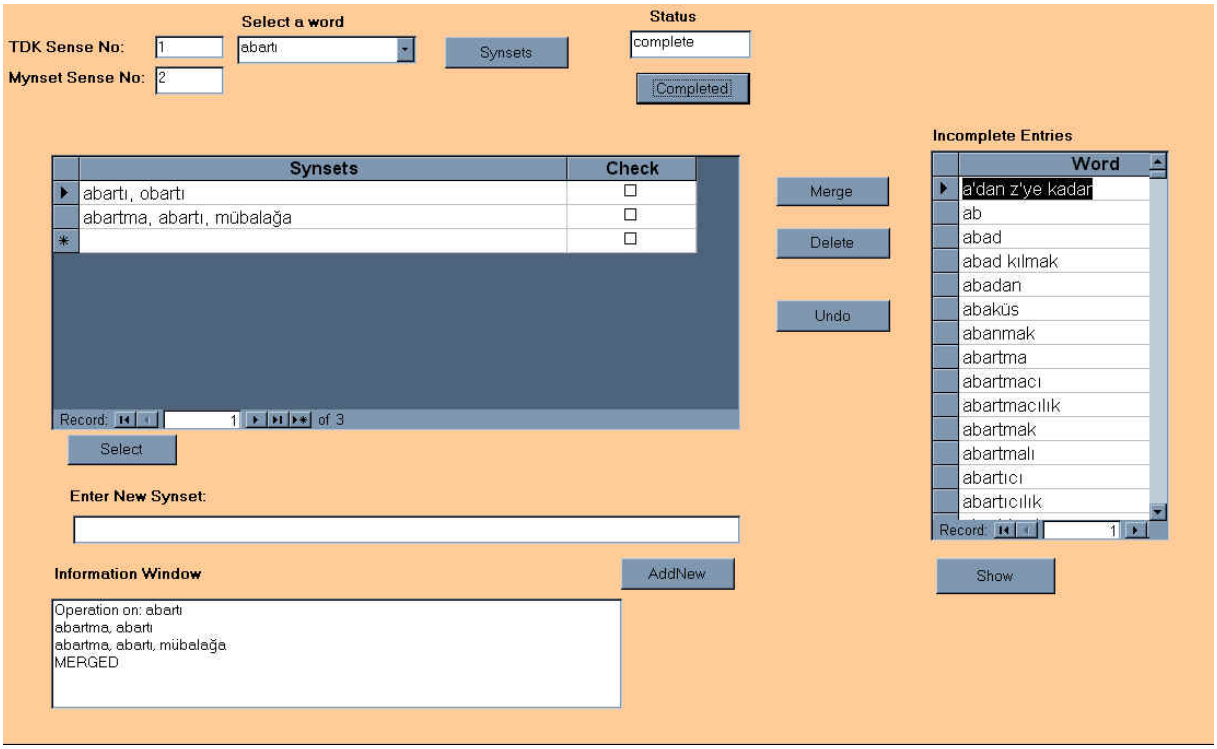


Figure 6.6: Merge operation on the word “abartı” (exaggeration)

When the user highlights a line in the “Incomplete Entries” window and clicks on the “Show” button under this window, the highlighted entry is automatically displayed in the main “Synsets” window.

When the user wants to modify an existing synset (add, delete or modify some synset members), in the main Synsets window, s/he highlights the synset which is to be modified and clicks on “Select”. This generates a copy of the highlighted synset in the “Enter New Synset” window. Clicking on the “Add New” button adds the new synset in the “Enter New Synset” window to the synset database.

All actions are displayed on the “Information Window”. This information is also stored in a log file called “synset.log”.

6.4 Problems encountered during the merging process of different resources

The basic problem we encountered was the difference in the format of the resources. We used the TDK Monolingual Dictionary and our existing synonym database to extract synset candidates. The synsets extracted from TDK were in the format

?headword/sense no: synonym-1, synonym-2, ?synonym-k?:"acemi"/1, "Toy, beceriksiz" (novice)

and the format of our existing synonym database was:

400000080,1,'beceriksiz',r
400000080,2,'acemi',r
400000080,3,'bilgisiz',r
400000080,4,'vukufsuz',r
400000080,5,'malumatsúz',r

In order to integrate our candidate synsets into the existing database, we transformed both formats into a single one. The resulting merged database had the following format:

acemi/1, beceriksiz, toy (from the TDK Dictionary)

acemi, beceriksiz, bilgisiz, malumatsúz, vukufsuz (from the existing synonym database)

We only had sense number information for the headword of synset candidates extracted from the TDK dictionary. The remaining synset members did not have any sense numbers and we had to assign them manually.

References

Kemal Oflazer (1996) Error-tolerant Finite State Recognition with Applications to Morphological Analysis and Spelling Correction. *Computational Linguistics,* Vol: 22, No:1.

Kemal Oflazer (1994), Two-level Description of Turkish Morphology, *Literary and Linguistic Computing*, Vol. 9, No:2.

CONCLUSIONS AND FUTURE WORK

In this deliverable we have presented the computational tools and the language resources that were used for the construction of monolingual WordNets for each of the participating languages (Bulgarian, Czech, Greek, Romanian, Serbian, Turkish). Note that the report represents the first part of the deliverable D.3.1.
In addition, each partner has delivered the software and excerpts of the lexical resources which were developed for his language separately and they are currently available on the information server.

In the content of this workpackage, each participant first performed a detailed recording of already available tools that were applicable to the construction of the respective monolingual WordNet. Moreover, each partner developed new tools following the specifications and methodology set in workpackage WP2.
Participants who already had a WordNet developed for their own language redesigned and restructured their tools in order to meet the new requirements of the project.
In addition, they shared their expertise with the other members of the consortium.

The fact that a merge model approach is followed in the Balkanet project, has led each partner to develop independently his own tools and resources taking into account the particularities of each language.
Therefore, the individual contributions to this report vary to a significant extent according to the specific needs and requirements for each individual language and the availability of resources.
Nevertheless, the tools and resources described in the report can be grouped under the following categories:

- Language resources
  - Dictionaries
  - Corpora
- Tools used for the extraction and processing of linguistic information from dictionaries or corpora
  - Tools already available at each site
  - Tools developed exclusively to support the monolingual work for the construction of the individual WordNets

The tools reported here, are of a significant help the linguists of each team in their work on the extraction of the set of defining terms that form the basis for the development of the core monolingual WordNets.

Following the work reported here, and during the second part of workpackage WP3, the teams will work on the design and development of the multilingual database along with the necessary tools. The monolingual work and infrastructure created during the first part of WP3 will serve as the basis in order to capture the differences in lexical expressiveness of the different languages.