# A Computer Language for Linguistic Knowledge Description of Natural Languages

**Abstract.** The paper presents the main features of a new very general computer language GRAALAN (Grammar Abstract Language) created to express linguistic knowledge concerning individual natural languages or correspondences between two natural languages. The Linguistic chapters that can be described using GRAALAN for an individual natural language are: alphabet, morphologic structures, inflection rules, inflection forms, lexicon, syllabification and syntax. The correspondences between two natural languages that can be expressed using GRAALAN are: lexical correspondences (word or multiword expression correspondences), syntactic and morphologic correspondences. A GRAALAN to XML compiler implements already the features of the language. Some linguistic knowledge for Romanian language was already created using GRAALAN. GRAALAN is based on GDG - Generative Dependency Grammar, DT - Dependency Trees and AVT - Attribute Value Trees. GRAALAN generates by compilation XML linguistic knowledge bases that can be used to build different kinds of linguistic monolingual and bilingual applications.

## 1. Introduction

The use of computer in natural language processing allows at least two kinds of functions: the development of certain linguistic applications and the study of certain linguistic aspects. Both these groups of functionalities involve the formalization and the standardization of linguistic knowledge representation.

A huge effort was put in developing some formal models. As a consequence, a lot of languages models and language grammar types were proposed trying to solve the natural language representation. There are three of the linguistic models that seem to be more successful and used in some applications [1]: TAG (Tree Adjoining Grammar) [2], HPSG (Head-driven Phrase Structure Grammar) [3] and LFG (Lexical Functional Grammar [4]. During the last years another idea was more and more analyzed and studied: the dependency. Actually, it is a quite old idea [5] but new valences and strength became attractive. The present paper is based on a new concept that try to move farther these researches, integrating two directions that seems to be almost irreconcilable till now: the generative approach and the dependency approach [6].

The standardization of linguistic knowledge representation concern also a large set of directions: annotation [7], transliteration [8][9][10][11][12], character set

[13][14][15], language and country codes [16], [17], phonetic character set [18], subcategorization [19], lexicons [20][21].

The present paper propose a new very general computer language GRAALAN (Grammar Abstract Language) created to express in a unified system the linguistic knowledge concerning individual natural languages or correspondences between two natural languages. The Linguistic chapters that can be described using GRAALAN for an individual natural language are: alphabet (section 3), syllabification (section 4), morphologic structures (section 5), lexicon (section 6), inflection rules (section 7), inflection forms (section 8), and syntax (section 9). The correspondences between two natural languages that can be expressed using GRAALAN are: lexical correspondences (word or multiword expression correspondences), syntactic and morphologic correspondences (section 10). GRAALAN has also some specific features presented in section 11. A GRAALAN to XML compiler implements already the features of the language (section 12). In section 13 some conclusion will be presented concerning the implementation of GRAALAN for some linguistic knowledge of Romanian language.

## 2. GRAALAN features

We will consider ten of the most important levels of linguistic knowledge from the conventional grammars: sound, phonetic alphabet, normal alphabet, morpheme, synthetic inflection, analytic/synthetic inflection, sentence syntax, phrase syntax, text and bilingual levels. A set of fours levels groups in GRAALAN these conventional levels. For each GRAALAN level, different section types can be used in GRAALAN linguistic knowledge description. The correspondences between conventional grammars levels, GRAALAN levels and GRAALAN sections are represented in figure 1. We have the following structure of GRAALAN levels:

a) GRAALAN alphabet level corresponds to sound, phonetic alphabet and normal alphabet levels from conventional grammars and can be expressed using GRAALAN alphabet section.

b) GRAALAN morphology level corresponds to morpheme, synthetic inflection and analytic/synthetic levels from conventional grammars and can be expressed using GRAALAN lexicon, syllabification, morphological configurator, inflection rules and inflection forms section.

c) GRAALAN syntax level corresponds to sentence syntax, phrase syntax and text levels from conventional grammars and can be expressed using GRAALAN syntax section.

d) GRAALAN bilingual level corresponds to bilingual level from conventional grammars and can be expressed using GRAALAN bilingual correspondences section.

A full GRAALAN program can be described as follows:

1. {GRAALAN program} => {GRAALAN section} {GRAALAN program} | {GRAALAN section}
2. {GRAALAN section} => {alphabet} | {lexicon} | {syllabification} | {morphological configurator} | {inflection rules section} | {inflection forms} | {syntax} | {bilingual correspondences} | {messages}

GRAALAN, as a computer language, has also two specific features: messages section that allows the using of linguistic knowledge from different languages and macros feature that allows a good compression of GRAALAN text.

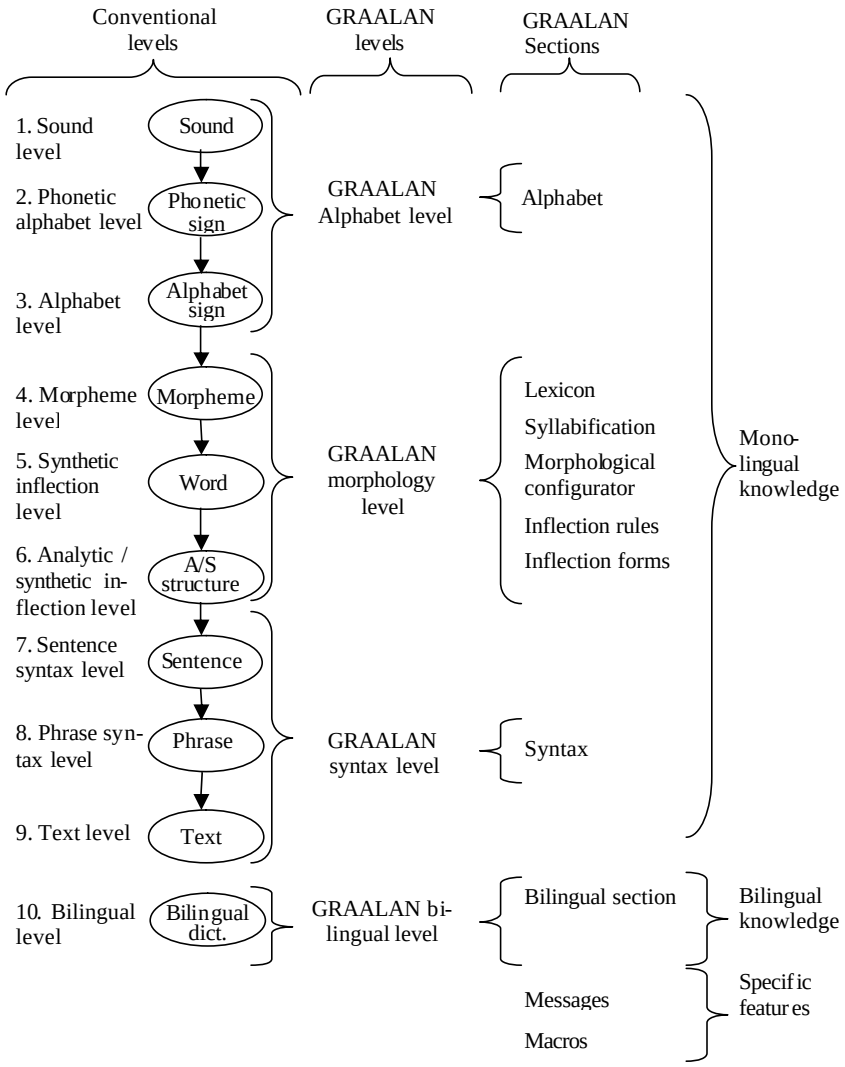


**Fig. 1.** GRAALAN sections

## 3. The alphabet

The alphabet used by a natural language is described in GRAALAN using the „Section Alphabet". The following information must be indicated in such a section:

*a) Phonetic alphabet* corresponding to a language is a subset of a more general alphabet (the phonetic alphabet used by GRAALAN is IPA – International Phonetic Alphabet [18], promoted by IPA – International Phonetic Association (http://www.arts.gla.ac.uk/IPA/index.html). There are two types of phonetic characters indicated in alphabet section: the specific to described language phonetic alphabet and a nonspecific to described language phonetic alphabet but useful in certain circumstances for this language.

*b) Normal alphabet*. The described language (the „source language") uses a certain alphabet. This alphabet is specific to the language. In certain situations it is useful to dispose also of some others characters that do not belong to specific alphabet (for example when we want to indicate the etymology of a word and the original word must be written with the characters of the original language). This is the non specific normal alphabet.

*c) The special character* used by the language. These are for example the characters used for the punctuation. There too specific and non specific characters of the described language.

*d) Groups* of characters with specific significance from the point of view of the pronunciation (for example diphthongs or triftongs). A group defines a sequence of the normal characters and the pronunciation of this sequence expressed with phonetic characters.

*e) (Alphabetic) Classes* are different sets of the above characters or groups. For example, we can create classes like: the class of vowels, the class of consonants, the class of capital letters, the class of small letters, etc.

All the characters used in GRAALAN description of a language are coded using UNICODE [14], [15].

GRAALAN Alphabet section can define also the alphabetic sorting method (that is used by the lexicon for example).

All the others GRAALAN sections that indicate some characters, morphemes, words must use only the character sets defined in GRAALAN alphabet section.

## 4. The syllabification

The rules to make the syllabification for a certain natural language are described in GRAALAN using a special section named Syllabification, using the information that was introduced in the section Alphabet. For each word, we will consider in GRAALAN three types of syllabifications:

*a) Euphonic syllabification* concerns the word written with the normal or special alphabet of the language and respects the pronunciation mode. A syllable is in this case something that is pronounced as single "time" (moment, tempo) during a discourse.

b) *Phonetic syllabification* concerns the word written with the phonetic alphabet and it is analogue to euphonic syllabification. Each ”phonetic syllable” corresponds to a ”euphonic syllable”. The phonetic syllabification is used for example in the versification with rhyme.

*c) Morphologic syllabification* concerns the word written with normal or special alphabet and it is a euphonic type syllabification but it respects also the morphological structure of the word: prefixes, suffixes, prefixoids, suffixoids, some other elements (morphemes) that compose the word. Generally, but with many exceptions, a „morphological syllable” can not contain characters that belongs to two different morphemes.

The elements of a word that are „separated” by syllabification (or not) are:

*Type (a)*: The normal alphabet characters (described in Alphabet section).

*Type (b)*: Groups (diphthongs, triftongs, etc) described in Alphabet section. We will name these groups *phonetic groups*.

*Type (c)*: Some special characters (like the characters "-" or space that appears in some compound words), described also in Alphabet section.

*Type (d)*: Other constitutive elements (morphemes) described in Lexicon section. We will name these groups *morphological groups*.

Syllabification section will contain a set of rules only for euphonic syllabification and phonetic syllabification. The rules will contain sequences of constitutive elements types with the indication of the place where the hyphen can be put. The morphological syllabification has not rules in Syllabification section because it is an euphonic syllabification with a restriction that was defined above.

The following rules are considered implicit:

a) The hyphen can appear between two constitutive elements but it can not appear at the beginning of a word, at the end of a word, into a phonetic or morphologic group.

b) A syllable can not contain characters belonging to two phonetic groups.

c) A syllable can not contain characters belonging to two morphological groups.

In order to can apply the syllabification rules, it is considered that the groups and the accent are marked in a certain mode.

In Syllabification can be indicated also the hyphenation mode, i.e. how to do the syllabification at the end of a line.

## 5. The morphological configurator

The Morphological Configurator section indicated in a formal mode the morphological structure of a language. The Morphological Configurator is in fact an AVT (Atribut Value Tree) [22] that has associated to each node not only a *name* but also some other information. An attribute type node contains a lexical category and a value type node contains a lexical category value.

The information associated to a category node is: the category name, the abbreviation of the category name, the indication if the category is inflected or not and eventually the name of a direct action (procedural program) if it is necessary. (Though

GRAALAN is a descriptive language, it can be completed, if necessary, with some procedural actions).

The information associated to a category value node is: the category value name, the abbreviation of the category value name, the indications if the corresponding category value belongs to a lemma (or not), to a lexicon entry (or not) and if it can be used as reference for other inflected forms. It can be also present the name of a direct action.

If the morphological tree contains some identical zones, GRAALAN offers the possibility to associate a label to the first apparition of such a zone and to use this label instead of the others apparitions (a sort of macro without parameters).

The morphological configurator contains also a subsection where some equivalences between different inflected forms can be epressed. An equivalence rule contains two lists: a list of definition labels and a list of reference labels. These labels must be present in the morphological tree. The significance of an equivalence rule is the following: the inflected form that have among its attributes all the attributes with values that are labeled with labels from the reference list is identical with the inflected form that have among its attributes all the attributes with values that are labeled with labels from the definition list.

## 6. The lexicon

The Lexicon section is a (very large) set of lexical elements. GRAALAN works with three types of lexical elements (entries, POS - Part of speech):

a) *Morphemes* are particles that participate to build words. GRAALAN accepts many types of morphemes: roots, prefixes, suffixes, prefixoids, suffixoids, etc.

b) *Words* are the basic elements of the syntactic buildings. There are the following entry types for words:

- *Lemma* is the basic form (canonical form) of a word. Usually, all the other forms of a word (inflected forms) are obtained starting from lemma.
- *Supplementary input*: is an in inflected form of a word that accompanies the lemmas in an ordinary dictionary (for example plural form of a noun).
- *Wordform* is an inflected form of another word (therefore not a lemma) but that usually appears in a dictionary (for example "you" that can be considered an inflected form of "I").

c) *Multi word expression* (MWE) that are groups of words that have usually a significance very different from the significance resulted from the words separately taken. A MWE is present in the lexicon not only as a simple sequence of words but it is present also as a DT (Dependency Tree) [6]. Using this representation, the MWE can be easily identified in a text and can be put in correspondence with a MWE in other languages (see section 10).

Each lexicon entry has associated many other information depending of POS type: semantic information (i.e.: gloss, synonyms, antonyms, paronyms, hipernyms, hyponyms, connotations, homonyms, meronyms), etymology (original language, original form, transliteration of the original form), syllabification (euphonic, phonetic and

morphologic), morphology (inflection situation, inflection rule identification, segmentation), sort (how the word must be put in a list of ordered words), etc.

Usually the GRAALAN text corresponding to a lexicon is not written by the linguist but it is generated using a lexicon tool that generates the GRAALAN text.

## 7. The inflection rules

The Inflection rules section contains the rules that must be used to generate all the inflected forms of an inflected word. These inflected forms can be expressed in normal alphabet of in phonetic alphabet.

There are two types of inflection rules: basic rules and compound rules. Each rule is identified by a label. The compound rule contains a label list of basic rules. Each lemma type lexicon entry refers one and only one compound inflection rule. A compound rule generates all the inflected forms of a word. A basic rule is in fact a tree (AVT) that indicates a set of inflection situations. An inflection situation is a word form and its characterization from the point of view of lexical categories and lexical category values associated to this word form. An inflected word will have usually many inflected situations. An uninflected word will have only one "inflected" situation. A word can have sometimes, for one inflection situation many *variants*.

The inflection (that represents a basic inflection rule) must be compatible with the morphological tree described in Morphology Configurator Section.

Each leaf of the inflection tree has associated one or many elementary inflection rules. Each elementary inflection rule contains two transformation sequences: one transformation that indicates how the corresponding inflection form (expressed in normal or special alphabet) is obtained making some transformations on the lemma or on other inflected form (expressed also in normal or special alphabet) and one transformation that indicates how the corresponding inflection form (expressed in phonetic alphabet) is obtained making some transformations on the lemma or on other inflected form (expressed also in phonetic alphabet). If the leaf has associated many elementary rules, each elementary rule is preceded by a condition that indicates when the transformation sequence must be used.

The condition that trigger the transformation sequence is a logical expression that uses some logical operators ("and", "or", "different from") and some operands. The operand is described indicating some of its characteristics like the type (a lemma, another inflected form, a character string, groups or classes defined in Alphabet section) or how the operand is scanned (from left to right or from right to left).

The transformation described in the rule can contain a combination of insert, delete, replace operations, indicating the operation start place, the direction (from left to right or from right to left) and the type of the element that is transformed (word or character).

Using these GRAALAN features a large set of inflection rules types can be described.

## 8. The inflection forms

The Inflection Forms section contains a set of entries, each entry corresponding to an inflected form. An inflected form can have many words. One of these words is considered as "head" and it will be used to do the sorting according to sorting method defined in the Alphabet section. The most important information for an entry in the Inflection Forms section is:

- The inflected form written using the normal alphabet and the special alphabet defined in Alphabet section.
- The inflected form written using the phonetic alphabet defined in Alphabet section.
- The reference of the word in the lexicon whose inflected form is the current entry.
- The characterizing of the inflection situation (i.e. a set of lexical categories and lexical categories values).
- How the inflected form is syllabified in different situations: euphonic, phonetic morphologic and at the end of the line (hyphenation).
- How the sorting is done (the word from the inflection form that is used, the sort from left to right, the sort from right to left).

Generally, the Inflection Form section is generated by a special tool based on lexicon, alphabet, inflection rules, and syllabification rules and corrected by the linguist.

Starting from an entry in the Inflection Forms section, the corresponding word (lemma) in the Lexicon section can by found. Starting from the lemma, the corresponding inflection rule (compound rule) can be found in Inflection Rules section. By applying the inflection situations from the Inflection Rules section, all the inflection forms can be found.

## 9. The syntax

A syntax description from the GRAALAN Syntax section is a sequence of labeled rules [6]. A rule has two parts: the left part and the right part. The left part of a rule contains a non terminal and an AVT. The AVT contains syntactic/lexical categories with their values. The right part of a rule contains one or many alternants. An alternant is formed by a set of subsections: the syntactic subsection, the dependency subsection and the agreement subsection.

*a) The syntactic subsection* is a sequence of (eventually labeled) NTPAs where the four letters N, T, P, A have the following meanings:

**N** means non-terminals i.e. the syntactic categories that can be described having a name and a structure. A non-terminal must be described in a syntactic rule (must appear at lest one time in the left part of a rule).

**T** means terminals i.e. a set of the words that can be found in the lexicon or can be obtained by applying some flexional rules on words from the lexicon.

**P** means pseudo-terminals i.e. non-terminals that contain only terminals. When we will describe a dependency tree or a grammar we will not cover all the words from the lexicon because in this case the number of rules from the grammar can be too big. So we can say that some non-terminals that we name pseudo-terminals (for example

some nouns or some verbs) will never be described in the grammar (we have not rule with these non terminals in the left part). These terminals will be takent from the lexicon.

**A** means procedural actions i.e. the set of the routines that can be used to represent a certain portion of the text that we analyze. For example a number represented like a sequence of digits or a mathematical formula or even an image with a certain significance that appear in a text can be "replaced" in grammars or dependency trees by a certain procedural action.

Each NTPA can have associated information about how this NTPA is linked with others NTPA by certain relations from the dependency subsection (these relations are indicated by their labels in the dependency section). Each NTPA can have associated information about how this NTPA is linked with others coordinate relations or governor/subordinate relations.

Each NTPA can have associated an AVT describing syntactic/lexical categories with their values.

*b) The dependency subsection* contains the description of the relations between the NTPA from the syntactic section (referred by their labels). There are two types of relations:

*The subordinate relation* SR is a relation between two N, T, P, A, or a coordinate relation CR, respecting some rules. One of the two elements is considered to be subordinated to the other element by the relation SR. Each SR can have associated information about how this SR is linked with others NTPAs or coordinated relations.

*The coordinate relation* CR is a relation between (usually) two N, T, P, A, or SR that are said to be coordinated by CR respecting some rules. These two elements are named "fixed entries". A CR can group also others "supplementary entries". Each CR can have associated information about how this CR is linked with others NTPAs, coordinate relations or governor/subordinate relations.

*c) The agreement section* contains a list of agreement rules [24]. An agreement rule is an expression of type "if(conditional expression) true (actions) false (actions) not applicabel (actions) undefined (actions)" working in a tetravalent logic (with the logical values: true, false, not applicable, and undefined). The conditional expression is a logical expression expressed between the categories values of the NTPAs from the syntactic section. The actions indicate some error messages or how the analyze will be continued after an agreement error is found.

The formalism used to describe the syntax has the important property to be reversible. This means that the same rules can be used to transform a surface structure in a deep structure (the dependency tree) and to transform the deep structure into a surface structure.

## 10. The bilingual correspondences

The Bilingual Correspondences section contains correspondences between the following elements types belonging to two languages: multiword expressions (MWE), words, syntactic structures and morphological structures.

*a) MWE correspondences.* MWEs [23] are represented in the lexicons of the two languages as structures (dependency trees - see section 6). These entries in lexicons are sorted (therefore searched and found) on the head of the dependency tree that represents the MWE. (In conventional MWE dictionaries, the sorting is done usually on the first or on the second word of the MWE). Some transformation relations must be defined between the elements of the two MWE (between the nodes of the dependency trees). Based on these rules, a transformation of the source expression into a target expression can be done. The elements (nodes) corresponding to terminals in the MWEs dependency tree are of three types:

- *Invariable elements* are elements must have always the same form in any instance of the MWE.
- *Partial variable elements* are elements that contain always the same word in any MWE instance but it can be inflected so it can have different forms.
- *Total variable elements* are elements that can be replaced in a particular instance by any word (that have eventually some restrictions) concerning the associated AVT (i.e. the combination of lexical categories and lexical categories values).

The dependency trees that correspond to MWEs will have only one "head". This head is a syntactic element (eventually a coordination relation) that in a certain MWE instance can be linked with the *context*. In a MWE instance, some of the MWE elements (named *MWE extensions*) that can be linked with other elements not belonging to the MWE.

The description of the correspondences of two MWEs belonging to two different languages is usually not reversible, therefore, in order have a complete bilingual correspondence between languages A and B we must define a Bilingual Correspondences section from A to B and a Bilingual Correspondences section from B to A.

*b) Word correspondences* are particular cases of the MWE correspondences. In fact, a word correspondence can be considered a MWE correspondence, when both MWEs have only one word.

*c) Syntactic structure correspondences* can be also be considered a particular case of MWE correspondences where all the nodes of the structure are of "total variable" type.

*d) Morphological structure correspondences* are also a particular case of MWE correspondence where there is only one node that have the "total variable" type, therefore only the associated AVTs have a certain significance (i.e. express the correspondences between a source morphological sub-tree and a target morphological sub-tree).

Generally, the Bilingual Correspondences section is generated by the linguist using a special tool based on lexicon, alphabet, inflection rules, and syllabification rules and corrected by the linguist. In this case, this tool generates the GRAALAN text.

## 11. GRAALAN special features

Among the special features of GRAALAN language, as a computer language, the most important are the *messages* and the *macros*.

*a) Messages.* In a GRAALAN based computer system many language and language correspondences can be present. This knowledge can be used *from* different

languages (the exploitation languages). If, in the linguistic description, some messages or some notions (like the names of lexical or syntactic categories or values) must be used, these elements must be present in many exploitation languages. So, the same message will have different variants, one for each exploitation language. The GRAALAN Message sections allow the representation of the messages in different exploitation languages.

*b) Macros.* GRAALAN implements a simple but strong macros mechanism. A macro can have some conventional parameters. In each place where the macro is needed, a macro call can be used with the effective parameters. The macros can be chained. The using of macros allows the compression of GRAALAN text written by the linguist (or generated by the special tools).

## 12. GRAALAN implementation

In order to use GRAALAN, a set of tools must be developed. Among these tools, the most important tools are the following (see figure 2) :

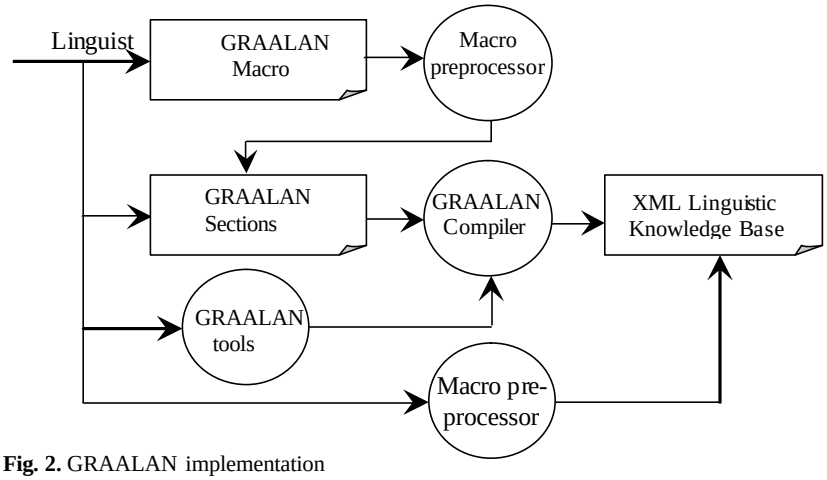


**Fig. 2.** GRAALAN implementation

- *GRAALAN Compiler* that converts GRAALAN text in XML (Extensible Markup Language) [25] building an LKB (Linguistic Knowledge Base). Each GRAALAN section is converted in an XML section. Each XML section is compatible with a specific XML DTD (Document Type Definition). An XML LKB has at least two major advantages: i) different applications can be developed starting from that format by using some standard tools to manage XML information; ii) XML is independent from any computer or operating system.

- *GRAALAN Macro processor* that analyses GRAALAN texts that uses macros and generates GRAALAN texts without macros that will be transformed in XML by GRAALAN Compiler.

- *GRAALAN LINK* that has the task to verify the links between different sections that was generated by the GRAALAN Compiler in XML (because not all the sections are compiled at the same time and even the same section can be compiled in many "chunks").

Some tools, especially for Lexicon and for Morphological Forms) can be used to facilitate the linguist work. These tools will generate automatically GRAALAN text (using information introduced by the linguist in a more user friendly interfaces) that will be, after that, compiled.

## 13. Conclusions

GRAALAN language is a strong instrument to build structured linguistic knowledge resources for individual natural languages or for correspondences between two natural languages belonging to a pair. The GRAALAN Compiler and the corresponding XML DTDs was developed. Some of the linguistic knowledge for Romanian language was written in GRAALAN, compiled and therefore generated in XML. Some linguistic knowledge between Romanian language and French language was conceived. All these demonstrated the viability and the force of GRAALAN formalism. The XML linguistic knowledge bases generated by GRAALAN compiler allows the possibility to build a lot of linguistic applications (machine translation included) and to make detailed linguistic analysis eventually using standard XML tools (for example different imlementations of DOM - Document Object Model [26]). Besides the GRAALAN Compiler, GRAALAN Macroprocessor and GRAALAN LINK many useful tools can be developed to complete a GRAALAN implementation, for example tools for lexicons, inflection form and bilingual correspondences creation.

## References


[1] S. Kahane: Grammaire d'Unification Sens-Texte. Vers un modèle mathématique articulé de la langue

[2] A. Joshi, L. Levi, L. Takabashi: Tree Adjunct Grammars, in Journal of the Computer and System Sciences, 1975

[3] C. Pollard, I. Sag: Head-Driven Phrase StructureGrammar, Stanford CSLI, 1994

[4] R. Kaplan, J. Bresnan: Lexical Functional Grammar. A Formal System for Grammatical Representation, in J. Bresnan (ed), The Mental Representation of Grammatical Relations, MIT Press, 1982

[5] L. Tesnière: Éléments de syntaxe structurelle, Paris, Klincksieck, 1959

[6] S. Diaconescu: Natural Language Understanding Using Generative Dependency Grammar, in Max Bramer, Alun Preece and Frans Coenen (Eds), ES 2002. Research and Development in Intelligent Systems XIX, Proceedings of ES2002, the Twenty second SGAI International Conference on Knowledge Based Systems and Applied Artificial Intelligence, Cambridge UK, December 2002, Springer, pp.439-452

[7] EAGLES Recommendations for the Morphosyntactic Annotation of Corpora, EAG–TCWG--MAC/R, Version of Mar, 1996

[8] ISO 9:1986 Transliteration of Slavic Cyrillic characters into Latin characters

[9] ISO 233:1984 Transliteration of Arabic characters into Latin characters

[10] ISO 259:1984 Transliteration of Hebrew characters into Latin characters
[11] ISO R 843:1968 Transliteration of Greek characters into Latin characters
[12] ISO 7098:1991 Romanization of Chinese
[13] ISO/IEC 6429:1992 Information processing -- Control functions for 7-bit and 8-bit coded character sets
[14] The Unicode Consortium: Addison-Wesley Professional; 5th edition (November 3, 2006,
[15] ISO/IEC 10646:1992 Information technology -- Universal Multiple-Octet Coded Character Set (UCS) ISO/IEC 10646-1:1993
[16] ISO 639:1988 Code for the representation of names of languages
[17] ISO 639-2:1995 Code for the representation of names of languages
[18] International Phonetic Association : Handbook of the International Phonetic Association, A Guide to the Use of the International Phonetic Alphabet
[19] EAGLES: Preliminary Recommendations on Subcategorisation, EAG---CLWG---SYNLEX/P Version of Aug, 1996
[20] EAGLES: Lexicon architecture Draft Report, EAG--LSG/IR--T1.1, Version of Oct, 1993
[21] Alessandro Lenci, Nuria Bel, Federica Busa, Nicoletta Calzolari, Elisabetta Gola, Monica Monachini, Antoine Ogonowski, Ivonne Peters, Wim Peters, Nilda Ruimy, Marta Villegas, Antonio Zampolli: SIMPLE: A General Framework for the Development of Multilingual Lexicons
[22] S. Diaconescu: Some Properties of the Attribute Value Trees Used for Linguistic Knowledge Representation, in 2nd Indian International Conference on Artificial Intelligence (IICAI-05) INDIA during December 20-22 2005
[23] S. Diaconescu: Multiword Expression Translation Using Generative Dependency Grammar, in Proceedings of ESTAL 2004 - ESPAÑA for NATURAL LANGUAGE PROCESSING October 20-22, 2004, Alicante, Spain
[24] S. Diaconescu: Natural Language Agreement Description for Reversible Grammars, in Tamás D. Gedeon, Lance Chun Che Fung (Eds.), AI 2003: Advances in Artificial Intelligence, 16th Australian Conference on AI, Perth, Australia, December 2003, Proceedings, pp. 161-172.
[25] W3C: Extensible Markup Language (XML) 1.0, Recommendation 10-Feb-98
[26] W3C: Document Object Model (DOM) Level 1 Specification (Second Edition), Version 1.0, September, 2000