A Cascaded Syntactic Analyser for Basque

Itziar I. Aduriz, Maxux M. Aranzabe, Jose Maria J. Arriola, Arantza Díaz de Ilarraza, Koldo K. Gojenola, Maite M. Oronoz

A Cascaded Syntactic Analyser for Basque

I. Aduriz *, M. J. Aranzabe, J. M. Arriola, A. Díaz de Ilarraza,
K. Gojenola, M. Oronoz, L. Uria

IXA Group (http://ixa.si.ehu.es)
Department of Computer Languages and Systems
University of the Basque Country
P.O. box 649, E-20080 Donostia
mailto:jiporanm@si.ehu.es

* Department of General Linguistics
University of Barcelona
Gran Via de las Corts Catalans, 585, 08007 Barcelona
itziar@fil.ub.es

Abstract. This article presents a robust syntactic analyser for Basque and the different modules it contains. Each module is structured in different analysis layers for which each layer takes the information provided by the previous layer as its input; thus creating a gradually deeper syntactic analysis in cascade. This analysis is carried out using the Constraint Grammar (CG) formalism. Moreover, the article describes the standardisation process of the parsing formats using XML.

1 Introduction

This article describes the steps we have followed for the construction of a robust cascaded syntactic analyser for Basque. Robust parsing is understood as "*the ability of a language analyser to provide useful analyses for real-world input texts. By useful analyses, we mean analyses that are (at least partially) correct and usable in some automatic task or application*" (Ait-Mokhtar *et al.*, 2002). The creation of the robust analyser is performed based on a shallow parser. In this approach, incomplete syntactic structures are produced and thus the process goes beyond shallow parsing to a deeper language analysis in an incremental fashion. This allows us to tackle unrestricted text parsing through descriptions that are organized in ordered modules, depending on the depth level of the analysis (see Fig. 1).

In agglutinative languages like Basque, it is difficult to separate morphology from syntax. That is why we consider morphosyntactic parsing for the first phase of the shallow syntactic analyser, which, in turn, will provide the basis for a deeper syntactic analysis.

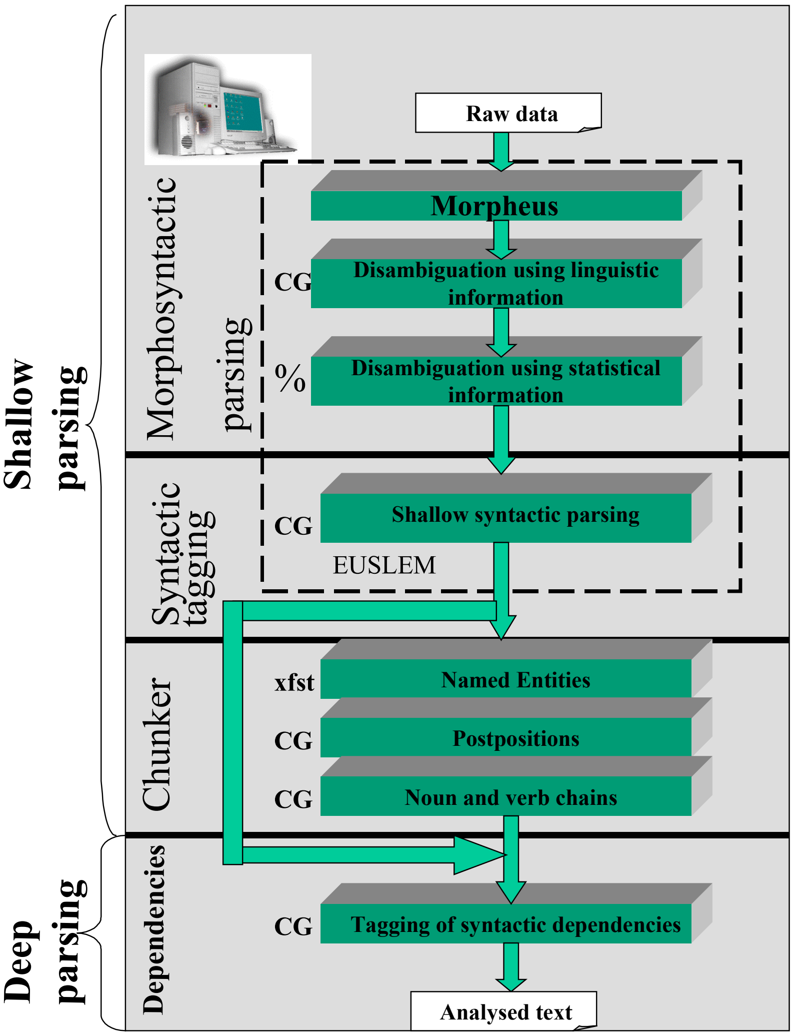


Fig. 1. Architecture of the system

In section 2 we briefly describe the main features of Basque. The steps followed in the process of creation of the cascaded parser are presented in section 3. Section 4 explains how the information is encoded in XML following the Text Encoding Initiative (TEI) guidelines. Finally, some conclusions and objectives for future work are presented.

2 Main Features of Basque

Basque is not an Indo-European language and differs considerably in grammar from the languages spoken in surrounding regions. It is an inflectional language in which grammatical relations between components within a clause are represented by suffixes. This is a distinguishing feature since the morphological information that words contain is richer than in surrounding languages. Given that Basque is a head final language at the syntactic level, the morphological information of the phrase (number, case, etc.), which is considered to be the head, is in the attached suffix. That is why morphosyntactic analysis is essential. In fact, Basque is known as a free-order language.

3 Syntactic Processing of Basque: The Steps Followed

We face the creation of a robust syntactic analyser by implementing it in sequential rule layers. In most of the cases, these layers are realized in grammars defined by the Constraint Grammar formalism (Karlsson *et al.*, 1995; Tapanainen & Voutilainen, 1994). Each analysis layer uses the output of the previous layer as its input and enriches it with further information. Rule layers are grouped into modules depending on the level of depth of their analysis. Modularity helps to maintain linguistic data and makes the system easily customisable or reusable.

Figure 1 shows the architecture of the system. The shallow parsing of the text begins with the morphosyntactic analysis. The information obtained is then separated into noun and verb chains. Finally, the deep analysis phase establishes the dependency-based grammatical relations between the components within the clause.

The results obtained in each parsing level of the sentence *Noizean behin itsaso aldetik Donostiako Ondarreta hondartzara enbata iristen da* ‘Once in a while, a storm arrives from high seas to the Ondarreta beach in Donostia’ will help in providing a better understanding of the mentioned parsing process.

3.1 Applied formalism

The parsing system is based on finite state grammars. The Constraint Grammar (CG) formalism has been chosen in most cases because, on the one hand, it is suitable for treating unrestricted texts and, on the other hand, it provides a useful methodology and the tools to tackle morphosyntax as well as free order phrase components in a direct way. The analyser used is CG-2 (www.conexor.com).

A series of grammars are implemented within the module of the shallow parsing which aim:

1. To be useful for the disambiguation of grammatical categories, removing incorrect tags based on the context;
2. To assign and disambiguate partial syntactic functions;
3. To assign the corresponding tags to delimit verb and noun chains.

Besides, dependency-based parsing is made explicit in the deep parsing module by means of grammars similar to those used in the shallow parsing module.

Even though CG originally uses mapping rules to assign the syntactic functions of grammatical categories defined by the context, in the above-mentioned modules these rules assign the corresponding syntactic tags to each analysis level. An example of a rule defined to detect the beginning of noun chains is shown below:

```
MAP (%INIT_NCH) TARGET (NOUN) IF (0 (GEN-GEL) + (@NC))
                                 (-1 PUNCT)
                                 (1 NOUN OR ADJ OR DET);
```

This rule assigns the noun-chain-initial tag (%INIT_NCH) to the noun if the following conditions are satisfied: a) the word is in any of both genitives (0 (GEN-GEL) + (@NC[1])); b) it has a punctuation mark on its left side (-1 PUNCT); c) there is a noun, an adjective or a determiner (1 N OR ADJ OR DET) on its right side.

3.2 Shallow Syntactic Analyser

The parsing process starts with the outcome of the morphosyntactic analyser MORFEUS (Aduriz *et al.*, 1998), which was created following a two-level morphology (Koskenniemi, 1983) and it deals with the parsing of all the lexical units of a text, both simple words and multiword units as a Complex Lexical Unit (CLU).

From the obtained results, grammatical categories and lemmas are disambiguated. The disambiguation process is carried out by means of linguistic rules (CG grammar) and stochastic rules based on markovian models (Ezeiza, 2003) with the aim of improving the parsing tags in which the linguistic information obtained is not accurate enough. Once morphosyntactic disambiguation has been performed, we should, ideally, be working on a morphosyntactically fully disambiguated text when assigning syntactic functions.

3.2.1 Disambiguation of Shallow Syntactic Functions

The aim of the syntactic disambiguation rules is to assign a single syntactic function to each word. This process is performed in two steps:

1. Assignment of syntactic functions. Words inherit their syntactic function from the EDBL database for Basque (Aldezabal *et al.*, 2001). Nevertheless, not all the syntactic functions derive from EDBL but some are inherited from CG syntactic and *mapping* rules due to the fact that they depend on the context.
2. Reduction of syntactic ambiguity by means of constrains.

The syntactic functions that are determined in the partial analysis are based on those given in Aduriz *et al*. (2000). The syntactic functions employed basically follow the same approach to syntactic tags found in ENGCG, although some decisions and a few changes were necessary. Basically, there are three types of syntactic functions:

[1] @NC> noun complement

1. Those that represent the dependencies within noun chains (@CM>[2], @NC> etc.).
2. Non-dependent or main syntactic functions (@SUBJ, @OBJ, etc.).
3. Syntactic functions of the components of verb chains (@-FMAINVERB[3], @+FMAINVERB, etc.).

The distinction of these three groups is essential when designing the rules, which assign the function tags for verb and noun chains detection.

Figure 2 shows the parsing of the sample sentence at this level.

```
/<Noizean_behin>/<CLU_EDBL>/
   ("noizean_behin" ADV ADVCOM @VC)
/<itsaso>/
   ("itsaso" NOUN COM @CM>)
/<aldetik>/
   ("alde" NOUN COM DEC NUMS DET DEC ABL @VC)
/<Donostiako>/<BEG_WC>/
   ("Donostia" NOUN LPN PLU- DEC NUMS DET DEC GEL @NC> @<NC)
/<Ondarreta>/<BEG_WC>/
   ("Ondarreta" NOUN LPN PLU- @CM>)
/<hondartzara>/
   ("hondartza" NOUN COM DEC NUMS DET DEC ALA @VC)
/<enbata>/
   ("enbat" NOUN COM DEC ABS NUMS DET  @OBJ @SUBJ @PRED)
   ("enbata" NOUN COM DEC ABS NDET  @OBJ @SUBJ @PRED)
   ("enbata" NOUN COM DEC ABS NUMS DET  @OBJ @SUBJ @PRED)
/<iristen>/
   ("iritsi" VERB SIM MVC INF ASP NF @-FMAINVERB)
/<da>/
   ("izan" AUXV A1 NR_HU  @+FAUXVERB)
/<.>/<PUNC_FS>/
```

Fig. 2. Morphosyntactic analysis

For instance, the syntactic function of the noun phrase *enbata* 'storm' (absolutive) is ambiguous because three syntactic analyses are possible (@SUBJ, @OBJ @PRED). Given that CG aims to assign a single function to each word, we need to choose whether to assign @SUBJ, @OBJ or @PRED to *enbata.* In this case, we choose the @SUBJ syntactic function by means of a CG rule that select it provided that there is agreement between *enbata* and the finite auxiliary verb *da.*

3.2.2 Delimiting Chains (*chunker*)

In the recognition process of entity names and postpositional phrases morphosyntactic information must be provided. Verb and noun chains make use of the syntactic functions provided by each word-form.

Entity names

For the recognition and categorization of entity names (person, organization and location) a combined system has been created. Firstly, the system applies a grammar

[2] @CM> modifier of the word carrying case in the noun chain

[3] @-FMAINVERB non finite main verb

that has been developed using an XFST tool (Xerox Finite State Transducer) (Karttunen *et al*. 1997) which detects the entity names using the morphological information. Then, entity names are classified through the application of a heuristic, which combines textual information and *gazetteers* (Alegria *et al.*, 2003).

The function tags defining the initial and final elements of entity names are: %INIT_ENTI_*, %FIN_ENTI_*, where "*" may be either LOC (location), PER (person) or ORG (organization).

Complex postpositions

Another characteristic feature of Basque is its postpositional system. The complex postpositions the system recognizes in this phase consist of both a case suffix followed by an independent word. For example: *gizon**aren*** ***aurrean*** 'in front of the man'. This type of complex postposition is taken into account in the recognition of noun chains (these noun chains also represent a postpositional system even though the postposition, in this case, consists of a single suffix). The function tags %INIT_POS and %FIN_POS define the beginning and the end of postpositional phrases.

Verb chains

The identification of verb chains is based on both the verb function tags (@+FAUXVERB, @-FAUXVERB, @-FMAINVERB, @+FMAINVERB, etc.) and some particles (the negative particle, modal particles, etc.).

There are two types of verb chains: continuous and dispersed verb chains (the latter consisting of three components at most). The following function tags have been defined:

- %VCH: this tag is attached to a verb chain consisting of a single element.
- %INIT_VCH: this tag is attached to the initial element of a complex verb chain.
- %FIN_VCH: this tag is attached to the final element of a complex verb chain.

The tags used to mark-up dispersed verb chains are:

- %INIT_NCVCH: this tag is attached to the initial element of a non-continuous verb chain.
- %SEC_NCVCH: this tag is attached to the second element of a non-continuous verb chain.
- %FIN_NCVCH: this tag is attached to the final element of a non-continuous verb chain.

Noun chains

This module is based on the following assumption: any word having a modifier function tag has to be linked to some word or words with a main syntactic function tag. Moreover, a word with a main syntactic function tag can, by itself, constitute a phrase unit. Taking into account this assumption, we recognise simple and coordinated noun chains, for which these three function tags have been established:

- %NCH: this tag is attached to words with main syntactic function tags that constitute a noun phrase unit by themselves.
- %INIT_NCH: this tag is attached to the initial element of a noun phrase unit.
- %FIN_NCH: this tag is attached to the final element of a noun phrase unit.

Figure 3 shows the parsing of the sample sentence with its corresponding chains. In it we can distinguish:

1. A complex lexical unit: *Noizean_behin* ‘once in a while’
2. A complex postposition: *itsaso aldetik* ‘from high seas’
3. An entity name: *Donostiako Ondarreta* ‘Ondarreta in Donostia’
4. Noun chains: *Donostiako Ondarreta hondartzara* ‘to the Ondarreta beach in Donostia’, and *enbata* ‘storm’

It is important to highlight that this process is parametrizable, allowing the user to choose to mark entity names but not postpositional phrases, etc.

```
/<Noizean_behin>/<CLU_EDBL>/
   ("noizean_behin"  ADV ADVCOM @VC %NCH)
/<itsaso>/
   ("itsaso"  NOUN COM @CM> %INIT_POS %INIT_NCH)
/<aldetik>/
   ("alde" NOUN COM DEC NUMS DET DEC ABL @VC
           %FIN_POS %FIN_NCH)
/<Donostiako>/<BEG_WC>/
   ("Donostia" NOUN LPN PLU- DEC NUMS DET DEC GEL
              %INIT_ENTI_LOC @NC> @<NC %INIT_NCH)
/<Ondarreta>/<BEG_WC>/
   ("Ondarreta"  NOUN LPN PLU- %FIN_ENTI_LOC @CM>)
/<hondartzara>/
   ("hondartza" NOUN COM DEC NUMS DET DEC ALA @VC
                %FIN_NCH)
/<enbata>/
   ("enbat" NOUN COM DEC ABS NUMS DET @OBJ @SUBJ @PRED %NCH)
   ("enbata" NOUN COM DEC ABS NDET @OBJ @SUBJ @PRED %NCH)
   ("enbata" NOUN COM DEC ABS NUMS DET @OBJ @SUBJ @PRED %NCH)
/<iristen>/
   ("iritsi"  VERB SIM MVC INF ASP NF @-NFMV %INIT_VCH)
/<da>/
   ("izan"  AUXV A1 NR_HU  @+NFAV %FIN_VCH)
/<.>/<PUNC_FS>/
```

Fig. 3. Analysis of chains

3.3 Deep Syntactic Analysis

The aim of the deep syntactic analysis is to establish the dependency relations among the components of the sentence. This process is performed by means of CG rules.

After considering several choices in the field of syntactic tagging, and taking into account the mentioned morphological and syntactic peculiarities of Basque, we decided to adopt the framework presented in Carroll *et al.* (1998, 1999). The dependencies we have defined, constitute a hierarchy that describes the most important grammatical structures such as relative clauses, causative sentences, coordination, discontinuous elements, elliptic elements and so on (Aduriz et al. 2002).

```
/<Noizean_behin>/<CLU_EDBL>/
   ("noizean_behin"  ADV ADVCOM @VC %NCH &NCMOD-CLU>)
/<itsaso>/
   ("itsaso"  NOUN COM @CM> %INIT_POS %INIT_NCH
              &NCMOD-POS12>)
/<aldetik>/
   ("alde"  NOUN COM DEC NUMS DET DEC ABL @VC
           %FIN_POS %FIN_NCH &NCMOD-POS22>)
/<Donostiako>/<BEG_WC>/
   ("Donostia"  NOUN LPN PLU- DEC NUMS DET DEC GEL
               %INIT_ENTI_LOC @NC> @<NC %INIT_NCH
               &NCMOD-GEL>)
/<Ondarreta>/<BEG_WC>/
   ("Ondarreta"  NOUN LPN PLU- %FIN_ENTI_LOC @CM> &NCMOD>)
/<hondartzara>/
   ("hondartza"  NOUN COM DEC NUMS DET DEC ALA @VC %FIN_NCH
                &NCMOD-ALA>)
/<enbata>/
   ("enbat"  NOUN COM DEC ABS NUMS DET @OBJ @SUBJ @PRED
            %NCH &NCSUBJ>)
   ("enbata"  NOUN COM DEC ABS NDET @OBJ @SUBJ @PRED
            %NCH &NCSUBJ>)
   ("enbata"  NOUN COM DEC ABS NUMS DET @OBJ @SUBJ @PRED
            %NCH &NCSUBJ>)
/<iristen>/
   ("iritsi"  VERB SIM MVC INF ASP NF @-NFMV %INIT_VCH)
/<da>/
   ("izan"  AUXV A1 NR_HU  @+NFAV %FIN_VCH &<AUXMOD)
/<.>/<PUNC_FS>/
```

Fig. 4. Dependency-based analysis

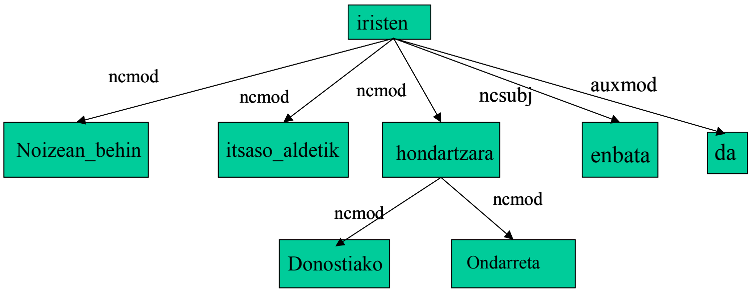


Fig. 5. Dependency tree

Figures 4 and 5 show an example of the adopted schema for the syntactic analysis as well as its corresponding syntactic tree. Notice that the nodes of the trees can be either single words or word-chains.

The syntactic dependencies between the components within the sentence are represented by tags starting with “&”. The symbols “>” and “<” attached to each dependency-tag represent the direction in which we find the sentence component whose dependant is the target word.

In the example we can see that the postpositional phrase *itsaso aldetik* ‘from high seas’ depends on the verb *iristen* ‘arrives’, which is on its right side. A post-process will make this link explicit.

4 Representation of all the phases of the analysis using XML

Figure 1 shows the global architecture of the robust syntactic analyser. The information to be exchanged among the different tools which constitute the parsing system is complex and diverse. Because of this complexity, we decided to use Feature Structures (FSs) to represent this information (Artola *et al*., 2002). Feature structures are coded following the TEI’s DTD for FSs (Sperberg-McQueen *et al.*, 1994), and Feature Structure Definition descriptions (FSD) have been thoroughly defined for each document created. The documents created as input and output of the different tools are coded in XML. The use of XML for encoding the information flowing between programs forces us to describe each document in a formal way, with the advantages it offers to keep coherence, reliability and maintenance.

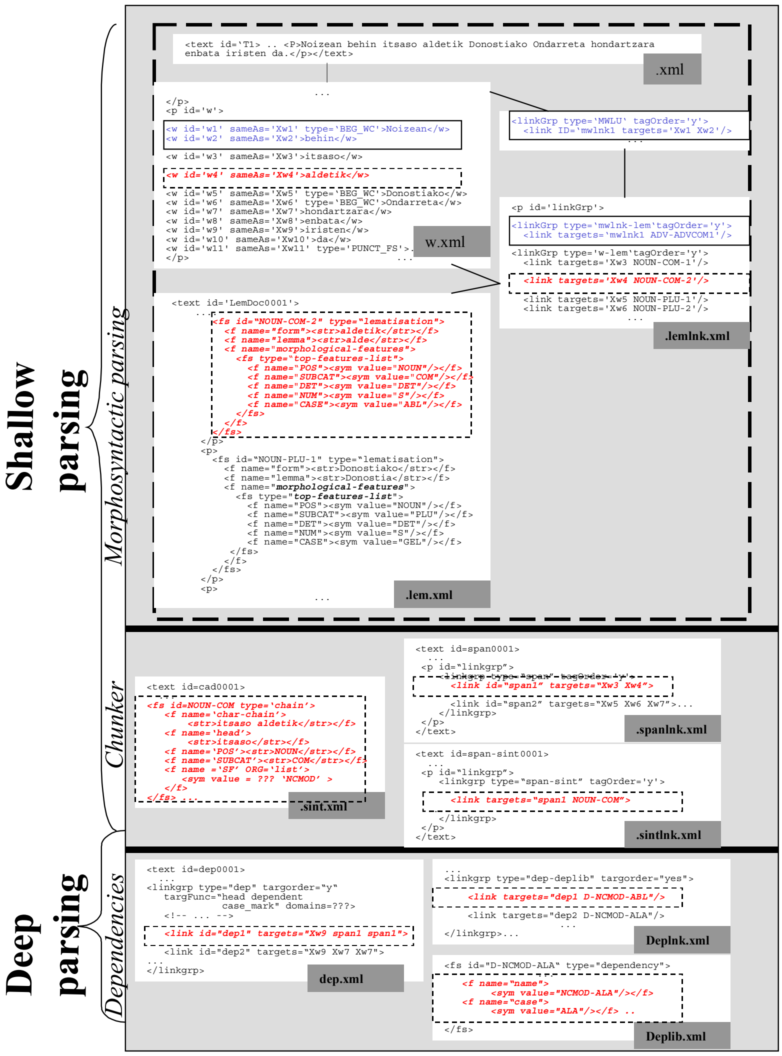
Figure 6 shows the representation of the sample sentence in XML format. The files described at the bottom constitute the tree of the above described analysis: *.dep.xml* (structure that establishes the syntactic relation between the head and its dependants), *.deplib.xml* (description of syntactic dependencies), *.deplnk.xml* (link between the two previous components).

5 Conclusions and Future Work

The present article outlines the process of the creation of a robust syntactic analyser for Basque. We want to remark that the morphosyntactic analyser has been widely used and tested in different projects (Verdejo *et al*., 2002). Regarding evaluation, we assessed the chain delimiter grammars according to Carroll (2003). We measured the correctness of the identified chunk boundaries, including ambiguous and unambiguous analyses. To achieve this, we based our analysis on a sample consisting of 260 sentences (totalling 4,873 words), where phrase boundaries have been marked. As a result we achieved an 83% precision rate (correctly selected chunks / number of chunks returned) and 81.4% recall rate (correctly selected chunks / actual chunks in the sentence).

As far as the deep syntactic analyser is concerned, we have defined a hierarchy of dependencies that describes the most important grammatical structures, such as relative clauses, causative sentences, coordination, discontinuous elements, elliptic elements and so on. To the present time, 150 CG rules have been defined and are currently being evaluated.

Moreover, we have already defined the structure of all the documents to be used in the syntactic analysis process in XML, but we are still working on the process for the automatic extraction of these documents. A library in C++ has been created in order



```
<text id='T1> .. <P>Noizean behin itsaso aldetik Donostiako Ondarreta hondartzara
enbata iristen da.</p></text>
```

```
                        ...
</p>
<p id='w'>
<w id='w1' sameAs='Xw1' type='BEG_WC'>Noizean</w>
<w id='w2' sameAs='Xw2'>behin</w>
<w id='w3' sameAs='Xw3'>itsaso</w>
<w id='w4' sameAs='Xw4'>aldetik</w>
<w id='w5' sameAs='Xw5' type='BEG_WC'>Donostiako</w>
<w id='w6' sameAs='Xw6' type='BEG_WC'>Ondarreta</w>
<w id='w7' sameAs='Xw7'>hondartzara</w>
<w id='w8' sameAs='Xw8'>enbata</w>
<w id='w9' sameAs='Xw9'>iristen</w>
<w id='w10' sameAs='Xw10'>da</w>
<w id='w11' sameAs='Xw11' type='PUNCT_FS'>.
</p>                                   ...
```

```
<linkGrp type='MWLU' tagOrder='y'>
  <link ID='mwlnk1 targets='Xw1 Xw2'/>
                    ...
```

```
<p id='linkGrp'>
<linkGrp type='mwlnk-lem'tagOrder='y'>
  <link targets='mwlnk1 ADV-ADVCOM1'/>
<linkGrp type='w-lem'tagOrder='y'>
  <link targets='Xw3 NOUN-COM-1'/>
  <link targets='Xw4 NOUN-COM-2'/>
  <link targets='Xw5 NOUN-PLU-1'/>
  <link targets='Xw6 NOUN-PLU-2'/>
                    ...
```

```
<text id='LemDoc0001'>
    ...
     <fs id="NOUN-COM-2" type="lematisation">
       <f name="form"><str>aldetik</str></f>
       <f name="lemma"><str>alde</str></f>
       <f name="morphological-features">
         <fs type="top-features-list">
           <f name="POS"><sym value="NOUN"/></f>
           <f name="SUBCAT"><sym value="COM"/></f>
           <f name="DET"><sym value="DET"/></f>
           <f name="NUM"><sym value="S"/></f>
           <f name="CASE"><sym value="ABL"/></f>
         </fs>
       </f>
     </fs>
   </p>
   <p>
     <fs id="NOUN-PLU-1" type="lematisation">
       <f name="form"><str>Donostiako</str></f>
       <f name="lemma"><str>Donostia</str></f>
       <f name="morphological-features">
         <fs type="top-features-list">
           <f name="POS"><sym value="NOUN"/></f>
           <f name="SUBCAT"><sym value="PLU"/></f>
           <f name="DET"><sym value="DET"/></f>
           <f name="NUM"><sym value="S"/></f>
           <f name="CASE"><sym value="GEL"/></f>
        </fs>
       </f>
     </fs>
   </p>
   <p>
                  ...
```

```
<text id=cad0001>
  ...
<fs id=NOUN-COM type='chain'>
   <f name='char-chain'>
       <str>itsaso aldetik</str></f>
   <f name='head'>
       <str>itsaso</str></f>
   <f name='POS'><str>NOUN</str></f>
   <f name='SUBCAT'><str>COM</str></f>
   <f name ='SF' ORG='list'>
      <sym value = ??? 'NCMOD' >
   </f>
</fs> ...
```

```
<text id=span0001>
  ...
 <p id="linkgrp">
    <linkgrp type="span" tagOrder='y'>
      <link id="span1" targets="Xw3 Xw4">
      <link id="span2" targets="Xw5 Xw6 Xw7">...
    </linkgrp>
 </p>
</text>
```

```
<text id=span-sint0001>
  ...
 <p id="linkgrp">
    <linkgrp type="span-sint" tagOrder='y'>
      <link targets="span1 NOUN-COM">
    </linkgrp>
 </p>
</text>
```

```
<text id=dep0001>
  ...
<linkgrp type="dep" targorder="y"
   targFunc="head dependent
             case_mark" domains=???>
   <!-- ... -->
   <link id="dep1" targets="Xw9 span1 span1">
   <link id="dep2" targets="Xw9 Xw7 Xw7">
...
</linkgrp>
```

```
...
<linkgrp type="dep-deplib" targorder="yes">
    <link targets="dep1 D-NCMOD-ABL"/>
    <link targets="dep2 D-NCMOD-ALA"/>
                  ...
</linkgrp>...
```

```
<fs id="D-NCMOD-ALA" type="dependency">
                 ...
   <f name="name">
        <sym value="NCMOD-ALA"/></f>
   <f name="case">
        <sym value="ALA"/></f> ..
</fs>
```

Fig. 6. Representation in XML

to implement the whole internal structure of XML documents. No specific knowledge about XML is required in order to define this kind of document.

Acknowledgements

This research is being supported by the University of the Basque Country (9/UPV00141.226-14601/2002), the Ministry of Industry of the Basque Government (XUXENG project, OD02UN52), the Interministerial Commission for Science and

Technology of the Spanish Government (FIT-150500-2002-244), and the European Community (MEANING project, IST-2001-34460).

We would especially like to thank Itsaso Esparza for helping us write the final version of the paper.

References

Aduriz I, Agirre E, Aldezabal I, Alegria I, Ansa O, Arregi X, Arriola J.M, Artola X, Díaz de Ilarraza A, Ezeiza N, Gojenola K, Maritxalar A, Maritxalar M, Oronoz M, Sarasola K, Soroa A, Urizar R, Urkia M 1998. A Framework for the Automatic Processing of Basque. In *Proceedings of the First International Conference on Language Resources and Evaluation*, Granada.

Aduriz I, Aldezabal I, Aranzabe M, Arrieta B, Arriola J, Atutxa A, Díaz de Ilarraza A, Gojenola K, Oronoz M, Sarasola K 2002. Construcción de un corpus etiquetado sintácticamente para el euskera. *Actas del XVIII Congreso de la SEPLN*, Valladolid, Spain.

Aduriz I, Díaz de Ilarraza A 2003. Morphosyntactic Disambiguation and Shallow Parsing in Computational Processing of Basque. Oyharçabal, B (Ed.) In *Inquiries into the lexicon-syntax relations in Basque (forthcoming)*.

Ait-Mokhtar S,.Chanod J.-P, Roux C. 2002. Robustness beyond shallowness: incremental deep parsing. *Natural Language Engineering, 8: 121-144.* Cambridge University Press.

Aldezabal I, Ansa O, Arrieta B, Artola X, Ezeiza A, Hernández G, Lersundi M 2001. EDBL: a General Lexical Basis for the Automatic Processing of Basque. *IRCS Workshop onLinguistic Databases*, Philadelphia (USA).

Alegria I.,Balza I.,Ezeiza N.,Fernandez I.,Urizar R. 2003. Named Entity Recognition and Classification for texts in Basque. *II Jornadas de Tratamiento y Recuperación de Información, JOTRI, Madrid. Spain.*

Artola X, Díaz de Ilarraza A, Ezeiza N, Gojenola K, Hernández G, Soroa A 2002. A Class Library for the Integration of NLP Tools: Definition and implementation of an Abstract Data Type Collection for the manipulation of SGML documents in a context of stand-off linguistic annotation. In *Proceedings of the Third International Conference on Language Resources and Evaluation,* Las Palmas de Gran Canaria, Spain.

Carroll, J. 2003. `Parsing'. In R. Mitkov (ed.), *The Oxford Handbook of Computational Linguistics*, Oxford, UK: OUP. 233-248.

Ezeiza, N 2003. Corpusak ustiatzeko tresna linguistikoak. Euskararen etiketatzaile sintaktiko sendo eta malgua. PhD thesis, University of the Basque Country.

Karlsson F, Voutilainen A, Heikkila J, Anttila A. 1995. Constraint Grammar: Language-independent System for Parsing Unrestricted Text. *Mouton de Gruyter*, Berlin.

Karttunen L., Chanod J-P., Grefenstette G., Schiller A. 1997. *Regular Expressions For Language Engineering*. Journal of Natural Language Engineering.

Koskenniemi K 1983. *Two-level Morphology: A general Computational Model for Word-Form Recognition and Production*. University of Helsinki, Department of General Linguistics. Publications 11.

Sperberg-McQueen C.M., Burnard L., 1994. *Guidelines for Electronic Text Encoding and Interchange*. TEI P3 Text Encoding Initiative.

Tapanainen P, Voutilainen A 1994. Tagging Accurately-Don´t guess if you know. In Proceedings of the 4th Conference on Applied Natural Language Processing, Washington.

Verdejo M.F., Gonzalo J., Màrquez LL., Padró LL., Rodríguez H., Agirre E. 2002. *HERMES, Hemerotecas electrónicas: Recuperación multilingüe y extracción semántica*, TIC2000-0335-C03. Jornada de Seguimiento de Proyectos en Tecnologías del Software. Programa Nacional de Tecnologías de la Información y las Comunicaciones.