COLDIC a generic tool for the creation, maintenance and management of Lexical Resources

Núria Bel, Sergio Espeja, Montserrat Marimon, Marta Villegas

Institut Universitari de Lingüística Aplicada, Universitat Pompeu Fabra, Barcelona, Spain
nuria.bel, sergio.espeja, montserrat.marimon, marta.villegas@upf.edu

Abstract

Although most of the Language Technologies applications need to develop and maintain large lexica, there has been a lack of generic tools for its creation, maintenance, and management which are independent of particular applications, and are well equipped for supporting lexicographic work. The most important obstacle to such generic tools was the proliferation of lexical models and formats: each application defined what information was required and how it should be declared.

The definition of standards for lexical encoding, as the one being developed in the Lexical Markup Framework (LMF, supported by the ISO and the e-content project LIRICS) will open the room for generic tools which are feasible and useful. Lexical management platforms can be tuned to the standard model and format, in order to create, merge or to maintain resources which can be used to feed different tools.

Besides, the existence of such standards can also enable the integration of high level supporting lexicographical tools, such as automatic acquisition, creation of analytical tools for corpus data assessment, etc.

We present in this paper a first approach for such a generic tool crucially based in the LMF model. COLDIC is a lexicographical management platform intended to be a generic tool independent of a particular technology and/or application.

1. Introduction

Computational lexica are repositories of information about words in particular languages that are traditionally developed for specific applications and tools. The quality of these resources is critical for the performance of the tool. For instance, Briscoe and Carroll (1993) observed that half of parse failures on unseen test data were caused by inaccurate lexical information, and Baldwin et al. (2004) identified that in parsing 20,000 strings from British National Corpus (BCN) a 40% of grammar failures were due to missing lexical entries, with a grammar dictionary of about 10,500 lexical entries.

Lexica are normally created and maintained by lexicographers that get little support. The information found in traditional dictionaries and terminological glossaries is not directly reusable in the encoding of word formal properties, like part of speech, gender, inflection paradigm, syntactic valency, semantic information etc. Besides, lexicon has to be tuned to specific domains, and the lexicographer must check whether a particular word is in the lexicon, and also that the encoding covers the use of the word in this new domain. The tuning of a lexicon to a new domain can involve the encoding of 4,000 to 20,000 entries.

Despite of the amount of work involved in manually crafting a lexicon, and the importance of the task, there has been a complete lack of generic tools that focus on supporting the lexicographer. For any particular project or application to develop a sophisticated lexicographical tool is usually out of its scope (and budget).

The main objective of COLDIC is to offer lexicographers working on lexica for Language Technology a tool that is particularly suited for lexical development tasks and that can be tuned and adapted to any application and model. The LMF model (Francopoulo et al. 2006) has been the basis for defining a database internal structure that guarantees the coverage of a large number of applications and languages. COLDIC also follows LMF to guarantee the interoperability for data exchange and access to other resources via web services[1].

Our tool has taken into account some basic factors to adequate the design and facilities to lexicographers, who can find an expressive enough framework to handle complex operations.

1) No technical background on databases should be required. The information to be encoded is declared in a DTD like file, which the system reads to build the database.
2) No previous knowledge on interfaces should be required. The system offers an LMF model based interface that guides the lexicographer to easily build queries and forms for introducing new data.
3) Information to be displayed in forms should be adjustable to different needs, and the forms can be fixed so to show or hidden particular features, as well as to use defaults for encoding of predefined groups.
4) Complexity of the model should be compensated by graphical views of the contents of the database.

[1] A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

2. Lexical Markup Framework: LMF

The Lexical Markup Framework was a joint ISO TC37/SC34 and LIRICS (EU e-content project) initiative to build a standardized abstract framework for the construction of computational lexica. The aim of LMF was to create a metamodel inclusive of the specificities of main lexicographical practices. This metamodel provides the user with a representation of lexical objects, the structure of the information underlying its description and its use. Used as a standard for the creation and use of electronic lexical resources, LMF is the basis for a real exchange of data between and among these resources, and for the use of these resources in remote, distributed applications based on web services. The ultimate goal of LMF is to facilitate true content interoperability across all aspects of electronic lexical resources.

LMF is defined in the LMF core package, together with two extensions: a Machine Readable Dictionary one and a NLP lexical resources one. While the core package describes the basic hierarchy of information of a lexical entry, the extensions (that use of LMF core components) address specific requirements for particular functionalities. LMF has been devised to handle only the structure of the lexical entry. Linguistic features that describe lexical items are defined in the ISO 12620 Data Category Registry[1], further guaranteeing standardization and interoperability.

3. COLDIC

COLDIC is a lexicographic platform for the creation and management of NLP lexica. One of its main features is that it provides an interface not only for human users, but also for consultation and information delivery asked by remote systems via web services. This interface is automatically generated at the same time than the database and needs no previous knowledge of web services by the user.

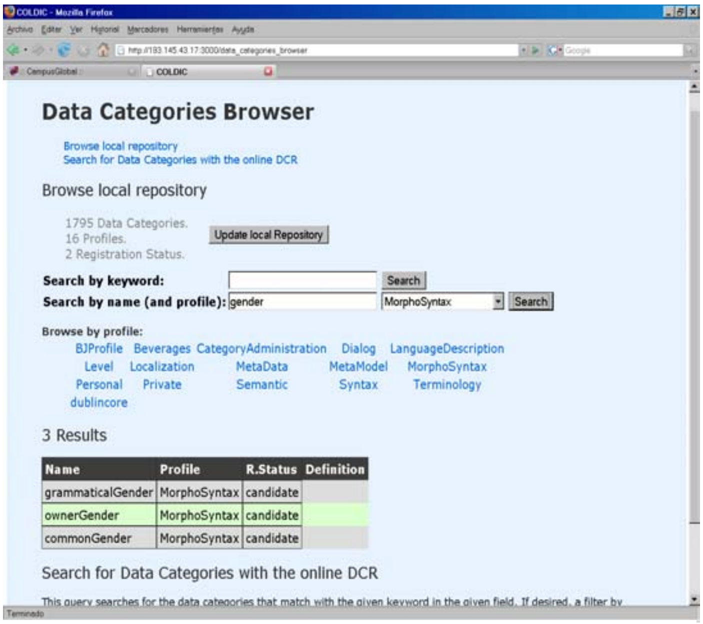
The other main features of the platform are:

- Reading and parsing of a LMF compliant lexical model DTD and generation of a relational database that can be managed with a core web based interface.
- Offering of a query builder tool that supports the user in the creation of content based queries, with advanced features as macro like queries with parametrized arguments.
- Automatic generation of a graphical view of the lexical model that is used as a support in the query and form builder tools.
- Support in the creation of encoding forms to assist lexicographers in the introduction of new data, search and validation of encoding features via Data Category Registration remote look up.
- Automatic creation of a number of standard and lexically oriented web services by exploiting the interoperability capabilities of the LMF standard and implementing the LMF for lexica API.

The creation of the database and its maintenance demand no specific training in databases as the LMF schema implemented in a XML Document Type Definition (DTD) is taken by the system to configure the platform. Special building functions support the user in the most common tasks of the lexicographic work, and the tool can be supported by analytical information about entries got from a pre-defined corpus.

Because user requirements might vary according to different needs, COLDIC comes equipped with a set of basic functionalities:

- Query and browse facilities by means of user build forms,
- import, export and migration of data,
- easy encoding of new data by means of user build forms, and direct access to different sources of information, such as the Data Categories repository,
- test and validation of both the data and the model, and
- lexicographic tools such as type definition, class extraction, evaluation, validation and statistical facilities.


COLDIC - Mozilla Firefox
Archivo Editar Ver Historial Marcadores Herramientas Ayuda
http://193.145.43.17:3000/data_categories_browser
Google
CampusGlobal
COLDIC
Terminado

Data Categories Browser

Browse local repository
Search for Data Categories with the online DCR

Browse local repository

1795 Data Categories.
16 Profiles.
2 Registration Status.

Update local Repository

Search by keyword: Search
Search by name (and profile): gender MorphoSyntax Search

Browse by profile:
BJProfile Beverages CategoryAdministration Dialog LanguageDescription Level Localization MetaData MetaModel MorphoSyntax Personal Private Semantic Syntax Terminology dublincore

3 Results

Name	Profile	R.Status	Definition
grammaticalGender	MorphoSyntax	candidate	
ownerGender	MorphoSyntax	candidate	
commonGender	MorphoSyntax	candidate	

Search for Data Categories with the online DCR

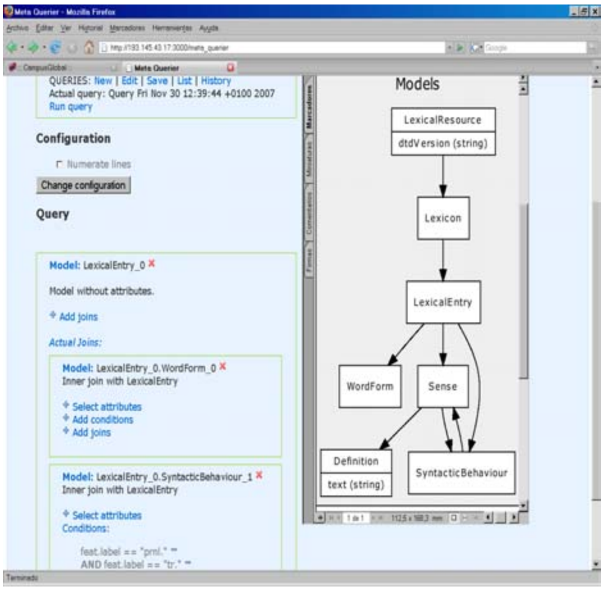
This query searches for the data categories that match with the given keyword in the given field. If desired, a filter by

Figure 1. *Window for searching at the DataCat repository. Lexicographer is supported with direct access and search/browse facilities for finding the standard DataCat for describing entries*

COLDIC consists of the following modules:

- The application generation module, which handles the creation of a relational database in terms of a LMF compliant DTD.
- The administration module, which handles profiles, imports and exports data.
- The core interface module, a graphical interface with facilities for building and executing queries and forms.
- The web services module, which offers a LMF compliant API.

[1] The Data Category Registry can be accessed at http://syntax.inist.fr/



Actual query: Query Fri Nov 30 12:39:44 +0100 2007

Configuration

□ Numerate lines

Query

Model: LexicalEntry_0

Model without attributes.

Actual Joins:

Model: LexicalEntry_0.WordForm_0
Inner join with LexicalEntry

Model: LexicalEntry_0.SyntacticBehaviour_1
Inner join with LexicalEntry

Conditions:

```
    feat.label == "prnl." "
    AND feat.label == "tr." "
```

Figure 2: *Query Builder, main window. The instantiation of the model according to user data is ploted to support the construction of the quey.*

4. Technical features and availability

COLDIC has been developed in Ruby (htpp://www.ruby-lang.org) and uses the Ruby on Rails framework (http://www.rubyonrails.org). Ruby is a dynamic, open source programming language with a focus on simplicity and productivity. Ruby on Rails is an open source web framework that favors convention over configuration in order to get less and more understandable code.

The core of COLDIC, the automatic building of the platform out of a DTD, is based in what we have called MetaRails plugins. MetaRails plugins are open source Ruby on Rails plugins that handles the following modules of the platform (Figure 1): Automatic database generation, web services generation, the *querier* and the forms editor. The open source project MetaRails, created for COLDIC development, can be found at http://meta-rails.rubyforge.org and is distributed under the GPL License. COLDIC is also released as a open source project that can be found at http://COLDIC.sourceforge.net distributed under the GPL license.

5. References

[1] Baldwin, T., E. M. Bender, D. Flickinger, et al. (2004). Road-testing the English Resource Grammar over the British National Corpus. In Proceedings of the FourthInternational Conference on Language Resources and Evaluation (LREC 2004), Lisbon.

[2] Briscoe, T. and J. Carroll. 1997. ‘Automatic extraction of subcategorization from corpora’. In Proceedings of the Fifth Conference on Applied Natural Processing, Washington.

[3] Francopoulo G., George M., Calzolari N., Monachini M., Bel N., Pet M., Soria C. 2006 Lexical Markup Framework (LMF), Proceedings of LREC Genoa.

[4] Kemps-Snijders et al. (2007). Data Category Registry API, Lirics Delivery 5.1. http://lirics.loria.fr.

[5] Kemps-Snijders and J. Nioche. (2007). API for Lexica, Lirics Delivery 5.1. http://lirics.loria.fr.

[6] Kemps-Snijders (2007). Data Category Usage Platform, Lirics Delivery 5.4. http://lirics.loria.fr.

[7] Lenci A., Bel N., Busa F., Calzolari N., Gola E., Monachini M., Ogonowski A., Peters I., Peters W., Ruimy N., Villegas M., Zampolli A. 2000. SIMPLE: A General Framework for the Development of Multilingual Lexicons, International Journal of Lexicography 13(4). Oxford University.

[8] Villegas, M.; Bel, N. (2002). "From DTDs to relational dBs. An automatic generation of a lexicographical station out off ISLE guidelines" dins LREC 2002 Third International Conference on Language Resources and Evaluation Proceedings. Las Palmas de Gran Canaria. Pp.. 694-700.