

Máster y Doctorado en
Sistemas Informáticos Avanzados

Informatika Fakultatea – Facultad de informática



Universidad del País Vasco Euskal Herriko Unibertsitatea

Máster S.I.A.

Master Thesis

Towards multilingual domain
module acquisition

Ángel Conde Manjón

Supervisors

Ana Arruarte Lasa

Department of Languages and Computer Systems
Computer Science Faculty, E.H.U./U.P.V.

Jon Ander Elorriaga

Department of Languages and Computer Systems
Computer Science Faculty, E.H.U./U.P.V.

informatika fakultatea facultad de informática



LSI

July 2012

ACKNOWLEDGMENTS

I would like to acknowledge Mikel Larrañaga, the author of the framework my thesis is based on, for his support and help. Furthermore, I would like to acknowledge my thesis directors Ana Arruarte and Jon Ander Elorriaga for the support they have provided and the assistance carried out to develop my work.

This project was possible thanks to the scholarship of the Department of Education of the Spanish Government that provided financial support.

ABSTRACT

DOM-Sortze is a framework for Semi-Automatic development of *Domain Modules*, i.e., the pedagogical representation of the domain to be learnt. *DOM-Sortze* generates *Domain Modules* for Technology Supported Learning Systems using Natural Language Processing Techniques, Ontologies and Heuristic Reasoning. The framework has been already used over textbooks in Basque language. This work presents the extension that adds English support to the framework, which is achieved with the modification of *ErauzOnt*. This is the tool that enables the acquisition of learning resources, definitions, examples, exercises, etc. used in the learning process. Moreover, some tests have been made to evaluate the performance of the tool with this new language. Principles of Object-Oriented Programming textbook for Object-Oriented Programming university subject is used for evaluation purposes. The results of this tests show that *DOM-Sortze* is not tight to a particular domain neither language.

Keywords: knowledge acquisition, content authoring, learning objects

CONTENTS

CONTENTS	I
LIST OF FIGURES	III
LIST OF LISTINGS	IV
LIST OF TABLES	V
LIST OF FORMULAS	VI
ACRONYMS	VII
CHAPTER 1 - INTRODUCTION	1
1.1. - MOTIVATIONS AND GOALS.....	1
1.2. - CONTEXT	2
1.3. - OUTLINE.....	4
CHAPTER 2 - STATE OF THE ART	5
2.1. - DOMAIN MODULE AUTHORIZING APPROACHES.....	5
2.1.1 - KONGZI	5
2.1.2 - Generation of ITSs from Spreadsheets.....	5
2.1.3 - IMAT.....	6
2.1.4 - ALOCOM: a Disaggregation Framework.....	6
2.1.5 - The Knowledge Puzzle Project - From Learning Objects to Learning Knowledge Objects.....	7
2.1.6 - Arikiturri - Automatic Generation of Exercises from Corpora.....	7
2.1.7 - MD2 project	7
2.2. - SUMMARY.....	7
CHAPTER 3 - DOM-SORTZE	9
3.1. - INTRODUCTION.....	9
3.2. - PREPROCESSOR	11
3.3. - LDO BUILDER.....	12
3.4. - ERAUZONT.....	14
3.5. - EL KAR-DOM.....	15
3.6. - SUMMARY.....	17
CHAPTER 4 - ERAUZONT	19
4.1. - OBTAINING LO FROM ELECTRONIC TEXTBOOKS USING ERAUZONT.....	19
4.1.1 - Generation of the DRs.....	21
4.1.2 - Enhancement of the DRs.....	25
4.1.3 - Assuring Cohesion in the DR Enhancement.....	28
4.1.4 - From DRs to LOs	28
4.2. - SUMMARY.....	30
CHAPTER 5 - EXTENDING ERAUZONT	31
5.1. - ADDING A NEW LANGUAGE TO ERAUZONT.....	31
5.2. - ADDING ENGLISH SUPPORT TO ERAUZONT.....	33
5.3. - EVALUATION OF ERAUZONT	36
5.3.1 - ErauzTest: a tool for the evaluation of the gathered LOs and the DR grammar.....	37
5.4. - EVALUATION OF ERAUZONT FOR ENGLISH.....	40
5.4.1 - Evaluation of the DR Grammar for English.....	40
5.4.2 - Evaluation of the LO Acquisition Process for English.....	41

5.5. - SUMMARY	42
CHAPTER 6 - CONCLUSIONS AND FUTURE WORK	43
BIBLIOGRAPHY	45
APPENDICES	57
APENDIX A - PATTERNS FOR DIDACTIC RESOURCES.....	57
A.1 - LIST OF DEFINITION PATTERNS.....	57
A.2 - <i>List of problems patterns</i>	58
A.3 - <i>List of example patterns</i>	58
A.4 - <i>List of Principle-Statements patterns</i>	59
A.5 - <i>List of Theorems patterns</i>	59

LIST OF FIGURES

Figure 1- General architecture of DOM-Sortze.....	9
Figure 2 - Class Hierarchy for the Tree-Like Document Representation.....	11
Figure 3 – ErauzOnt architecture.....	14
Figure 4 – Elkar-Dom architecture	15
Figure 5 – Snapshot of Elkar-DOM.....	16
Figure 6 – Search and selection of LOs using Elkar-DOM.....	17
Figure 7 – Generation of Learning Objects.....	20
Figure 8 – Generation of Didactic resources.....	21
Figure 9 – Classes for the Internal Representation of DRs.....	23
Figure 10 – Example of Gathered DRs	24
Figure 11 – Algorithm for the Composition of DRs.....	26
Figure 12 – Changes needed in DR generation process.....	32
Figure 13 – Changes needed from DOM-Sortze architecture perspective	33
Figure 14 – Diagram of the process of evaluation of an electronic textbook	37
Figure 15 – ErauzTest architecture.....	38

LIST OF LISTINGS

Listing 1- Fragment of the LDO.....	13
Listing 2 - Example of a LO.....	29
Listing 3 - Structure of the Output of the Linguistic Analysis.....	34
Listing 4 - Excerpt of the Part-of-Speech Information for a Sentence.....	35
Listing 5 - DocBook XML example.....	39
Listing 6 - Example of highlighted DocBook.....	39

LIST OF TABLES

Table 1 – Example of a pattern that allows identifying definitions	25
Table 2 – Example of a pattern that allows Identifying Principle Statements	25
Table 3 – Example of a pattern that allows Identifying Problem Statements	25
Table 4 – Example of two DRs that may be combined.....	25
Table 5 – Discourse Markers for Basque.....	28
Table 6 – Part of Penn Treebank Tag Set.....	34
Table 7 – Definition patterns.....	35
Table 8 – Example patterns.....	36
Table 9 – Problem patterns.....	36
Table 10 – Discourse Markers for English	36
Table 11 – LO highlight structure scheme	39
Table 12 – Accuracy of the DR Grammar	40
Table 13 – Recall of the LO acquisition process	41
Table 14 – Precision of the LO acquisition process	41
Table 15 – Table of definition patterns.....	58
Table 16 – Table of problems patterns	58
Table 17 – Table of example patterns.....	59
Table 18 – Table of principle-statements patterns.....	59
Table 19 – Table of theorems patterns	59

LIST OF FORMULAS

Formula 1 - Cosine equation.....27

ACRONYMS

ADL	Advance Distributed Learning - US Department of Defense
AI	Artificial Intelligence
AICC	Aviation Industry CBT Committee
API	Application Programming Interface
CBT	Computer-Based Training
CEN	European Committee for Standardization
CEN/ISSS/LT	European Committee for Standardization/Information Society Standardization System/Learning Technologies Work-
CM	Concept Map
CP	IMS Content Packaging
DC	Dublin Core
DR	Didactic Resource
ICT	Information and Communication Technologies
IMS	IMS Global Learning Consortium
ITS	Intelligent Tutoring System
LD	IMS Learning Design
LDO	Learning Domain Ontology
LMS	Learning Management System
LO	Learning Object
LOM	IEEE Learning Object Metadata
LOR	Learning Object Repository
LTSC	IEEE Learning Technology Standards Committee
NLP	Natural Language Processing
OAI	Open Archive Initiative
OAI-PMH	Open Archive Initiative Protocol for Metadata Harvesting

OL	Ontology Learning
PLQL	ProLearn Query Language
QEL	Query Exchange Language
QTI IMS	Question and Test Interoperability
RTF	Relative Term Frequency
SCO	Sharable Content Object
SCORM	Sharable Content Object Reference Model
SPI	Simple Publishing Interface
SQI	Simple Query Interface
SS	IMS Simple Sequencing
TSLs	Technology Supported Learning System

CHAPTER 1 - INTRODUCTION

In this chapter the motivation and goals of this work are presented. Then, the context of the work is shown and, finally, the outline of this thesis is described.

1.1. - MOTIVATIONS AND GOALS

Nowadays, Technology Supported Learning Systems (TSLs), such as Intelligent Tutoring Systems (ITSs), Adaptive Hypermedia systems (AHSs) and especially Learning Management Systems (LMSs), are being widely used at many academic institutions. TSLs require an appropriate *Domain Module*, i.e., the pedagogical representation of the domain to be learnt. Building the *Domain Module* is a hard task that entails not only selecting the domain topics, but also defining pedagogical relationships among the topics that determine how to plan the lessons, and providing the set of Didactic Resources (DRs) used during the learning process. The proliferation of Learning Objects Repositories (LORs) has brought the possibility of reusing existing Learning Objects (LOs) or DRs to build on-line courses on LMSs or other kinds of TSLs

Gathering the domain knowledge from already existing documents in a semi-automatic way may considerably reduce the development cost of the *Domain Modules*. Artificial Intelligence methods and techniques such as Natural Language Processing (NLP) and heuristic reasoning can be applied in order to achieve the semi-automatic generation of the *Domain Module*. In this way, teachers select the documents to be used as source data, and later supervise the results to complete or adapt the generated *Domain Module* to their requirements or teaching preferences. The acquisition of both the pedagogical relationships and the LOs relies on the identification of the most frequently used syntactic patterns.

Meeting these requirements a tool called *Dom-Sortze* has been developed. *Dom-Sortze* (Larrañaga, M., 2012) is a framework for the semi-automatic building of Domain Module from electronic textbooks using Ontologies, Natural Language Processing (NLP) techniques and heuristic reasoning. *Dom-Sortze* has been already tested over textbooks on the Basque Language, and it is intended to be enhanced so that it can support new languages such as English.

The main goal of this thesis is enhancing *Dom-Sortze* to support English and evaluate its performance over this language.

1.2. - CONTEXT

In the last few years the influence of new technologies in general, and Information and Communication Technologies (ICT) in particular, have highly increased.

The education has been affected by this revolution, providing means that enhance both teaching and learning. Years of research have facilitated the development of different kinds of TSLs such as LMSs, ITSs, Collaborative Learning Systems or Adaptive and Intelligent Web-based Educational Systems. LMSs such as *Moodle*¹ or *WebCT/Blackboard*² are currently being used at many academic institutions (Waits & Lewis, 2003; Parsad & Lewis, 2008). Furthermore, a positive relationship between the use of web-based learning technology and student engagement and desirable learning outcomes has been observed (P.-S. D. Chen et al., 2010). ITSs have also proved to improve the achievements of students (Anderson et al., 1995; Koedinger et al., 1997; Corbett et al., 1998; Mitrovic & Ohlsson, 1999; Arroyo et al., 2001; Mitrovic et al., 2004; Woolf et al., 2006).

In order to facilitate the construction of TSLs, an appropriate Domain Module (i.e. the pedagogical representation of the domain to be learnt) is required. The Domain Module is considered the core of any TSLs as it represents the knowledge about a subject matter to be communicated to the learner (Anderson, 1988; Wenger, 1987; Woolf, 2008; Nkambou, 2010). The Domain Module is used in ITSs to determine the content of the tutorial interaction, the selection of examples, questions and statements, and to assess the performance of the students (Stevens et al., 1982; Wenger, 1987).

Brusilovsky *et al.* (2003) claim that teachers should focus on Domain Module authoring while expert developers should carry out the development of the core of the TSLs. However, building the Domain Module is a hard task that might become easier by reusing existing materials (Casey & McAlpine, 2003). Main module authoring entails selecting the domain topics to be learnt, defining the pedagogical relationships among the learning topics, etc. Textbook authors deal with similar problems while writing their documents, which are structured in order to facilitate comprehension and learning. Electronic textbooks might be used as the source to build the Domain Module, reproducing how average teachers behave while preparing their subjects: they choose a set of reference books that provide the main Didactic Resources (DRs) - definitions, examples, exercises, etc., for the subject, and rely on them for scheduling their lectures.

First of all, a set of tools for sharing didactic resources is needed. In order to achieve this, some kind of standardization is needed. Therefore, Learning Objects (LOs) were designed to fulfill the task of reusing learning content. The IEEE Learning Technology

¹ - <http://moodle.org/>

² - <http://www.blackboard.com/>

Standards Committee (LTSC) defines a LO as “any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning” (LTSC, 2001). However, as (Wiley, 2000) states, this definition may be too vague as almost everything matches it, i.e., the notes teachers use for their classes may be considered LOs since they can be referenced during the learning process, even though its reusability in an application is quite limited. (Wiley, 2000) instead recommends considering LOs as any digital resource that can be reused to support learning.

LOs provide a means to facilitate knowledge reuse as they are “reusable pieces of educational material intended to be strung together to form larger educational units such as activities, lessons or whole courses” (Brooks et al., 2003).

Reusing a LO entails a way to describe it i.e. learning metadata and a way to store and manage LOs and their metadata. For this task Object Repositories (LORs) have been designed, which enable the possibility of finding and using the appropriate LOs. LORs that only manage metadata and do not store LOs, are also referred to as LO Referatories. Nowadays, there are many available LORs, such as ARIADNE (Duval et al., 2001; Ternier et al., 2009), Merlot (Cafolla, 2006), Edna (Adcock et al., 2000), or Edutella (Nejdl et al., 2002). LORs may contain either domain-specific content or general content.

ARIADNE (Duval et al., 2001; Ternier et al., 2009), which stands for Association of Remote Instructional Authoring and Distribution Networks for Europe, is a foundation that aims to promote the sharing and reusing of LOs. ARIADNE is in its core a distributed network of LOs, which uses standards for distributed digital resource management in order to enable interoperability (Ternier & Duval, 2006). The ARIADNE repository supports the storage of LOs and Learning Object Metadata (LOM) instances. LOs are described using the IEEE LOM standard (LTSC, 2001). The search interface of the repository is built on the Search Query Interface (SQI) specification (Simon et al., 2005). The publishing interface is based on the Simple Publishing Interface (SPI) specification (Ternier et al., 2008). The harvester collects metadata from external repositories in order to publish it in the ARIADNE repository and relies on the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) (Lagoze & de Sompel, 2001). ARIADNE provides services such as the metadata validation service, which validates metadata against an application profile or SAMgI (Meire et al., 2007), an automatic metadata generator. ARIADNE is part of the Global Learning Objects Brokering Exchange (GLOBE)¹ alliance of educational repositories together with Merlot (McMartin, 2004; Cafolla, 2006), Lornet², KERIS³, and LaClo⁴ among others.

¹ - <http://www.blackboard.com/>

² - <http://www.lornet.ca>

³ - <http://www.english.keris.or.kr>

⁴ - <http://www.laclo.org>

GLOBE provides a distributed network of LORs built on the IEEE LOM, SQI and OAI-PMH standards. The federated Search Engine allows query on the whole alliance.

1.3. - OUTLINE

This dissertation is divided in 6 chapters. Chapter 2 presents the state of the art related to this work.

Chapter 3 describes the general architecture of the *DOM-Sortze* framework.

Chapter 4 studies the tool of the framework called *ErauzOnt* and describes the process carried by the tool.

Chapter 5 presents extension that adds a new language to the framework and evaluates it.

Finally, the conclusions of the done work and future research lines are remarked in Chapter 6.

CHAPTER 2 - STATE OF THE ART

In this chapter current approaches of Domain Module authoring are presented.

2.1. - DOMAIN MODULE AUTHORING APPROACHES

Automatic or semi-automatic approaches for developing TSLs are required to lighten the development cost (Murray, 1999). This section presents some efforts aimed at covering different aspects of the TSLs development, from the construction of ITSs to the generation of reusable learning material.

2.1.1 - KONGZI

*KONGZI*¹ (Lu et al., 1995) is an authoring tool that was developed with the aim of automating the generation of ITSs. *KONGZI* was, probably, one of the first authoring tools that attempted to build ITSs automatically from documents. Later, *KONGZI* was enhanced to support the use of multimedia resources in the generated ITSs (W. Chen et al., 1997).

KONGZI can automatically produce exercises and tests, whose solutions can be automatically assessed, for learner evaluation. It uses some heuristics to automatically produce the exercises. One of these heuristics consists of lining two or more concepts with a non-existing relationship and asking the students to point out the mistake. The Student Model is updated according to his or her performance, and it is used to plan the learning sessions.

2.1.2 - Generation of ITSs from Spreadsheets

Lentini *et al.* (1995, 2000) developed a system for automatic knowledge acquisition and tutor generation for spreadsheet applications. The system processes existing spreadsheets to extract the knowledge and improve the spreadsheet application with tutoring facilities.

The generation of tutors consists of two stages: Acquisition of the Knowledge from the spreadsheet application, and the Generation of the Tutoring Facilities. Knowledge acquisition is performed in two steps. First, the application knowledge is gathered from the spreadsheet reconstructing the mathematical model coded into the spreadsheet scheme. The application knowledge is represented by a dependency graph, a directed acyclic graph. Next, the structure of the spreadsheet and the application knowledge are used to build the Meta-knowledge on Application Usage, a partition of the sheet into pieces that can be regarded as separate components of the overall scheme. This information is used by the Tutor Generator Module to enhance

¹- Kǒng Zǐ is the name of the Chinese philosopher known as Confucius

the processed spreadsheet with two kinds of tutoring support: a hypertext guide that describes the mathematical model coded in the spreadsheet, and an interactive tutor that supervises the end-user's activity.

2.1.3 - IMAT

IMAT (de Hoog et al., 1999) aimed at promoting the reuse of technical manuals, usually available in paper-based or electronic form, for Computer-Based Training (CBT). The training of maintenance is usually not part of the public curriculum, and therefore it is not an attractive market for educational publishers, which makes technical documents the only available source of information. However, technical documentation is designed for reference purposes but not for educational purposes, so it has to be revised to produce material for training purposes.

IMAT provides a set of tools to process the technical documents. The Document Analysis Tool breaks up the document, or its selected parts, into small parts or fragments, and indexes these fragments to facilitate their retrieval. The segmentation of the document relies on the original structure of the document (arrangement of chapters, sections, and paragraphs). To enable storage and retrieval, the document analysis is required to identify additional properties of the fragments such as the subject described in the fragment, the format of the fragment, and the way the information is represented (e.g., a list of parts or steps in a procedure).

The retrieved fragments can be copied&pasted into the authoring environment chosen by the author.

2.1.4 - ALOCOM: a Disaggregation Framework

The ALOCOM framework (Verbert, 2008) transforms documents (e.g., Powerpoint presentations, Wikipedia Pages and SCORM Content Packages) into a representation compliant to the Abstract Learning Object Content Model (ALOCOM) model (Verbert & Duval, 2004; Verbert et al., 2005). In this transformation process, the framework decomposes LOs into content components that can be accessed and, therefore, reused in new LOs. To facilitate content reuse, the metadata for the decomposed content components is automatically generated by *SAmgl* (Meire et al., 2007). Content inclusion is controlled to avoid duplicates.

2.1.5 - The Knowledge Puzzle Project - From Learning Objects to Learning Knowledge Objects

The Knowledge Puzzle Project is a framework which automatically composes instructional resources to fulfill a specific competence need just-in-time (Zouaq & Nkambou, 2009). (Wiley, 2000) claims that LOs “instructional design theory must be incorporated in any learning object implementation that aspires to facilitate learning”. To get such instructional theory-aware LOs, the knowledge representations used by ITs have been combined with the LOs to obtain the so-called Learning Knowledge Objects (LKO), i.e., active, independent and theory-aware LOs that can be considered tiny ITs. The core of the Knowledge Puzzle Project is the Organizational Memory (OM) a pool of knowledge in which LKO can be retrieved through dynamic aggregation.

2.1.6 - ArikIturri - Automatic Generation of Exercises from Corpora

ArikIturri (Aldabe, 2011), a system for the automatic generation of exercises, uses NLP techniques to build evaluation items from text corpora. ArikIturri is multi-lingual, it supports the generation of exercises in different languages, and has been tested in both Basque and English. ArikIturri supports the following kinds of exercises: fill-in-the-blank, word formation, multiple-choice questions, error correction questions and short answer questions.

2.1.7 - MD2 project

MD2 project (Padrón et al., 2005), a system for collaborative material development that aims for the reusability of didactic materials, is based on the fact that content creation and the learning design are conceived as different but convergent views of instructional design that require collaboration. The system stores all design rationales must be stored along with the products to be available for instructional designers in similar design situations, this is achieved using a control version system.

2.2. - SUMMARY

This chapter has presented some existing approaches for the Domain Module authoring.

CHAPTER 3 - DOM-SORTZE

In this chapter a framework for the semi-automatic building of Domain Module from electronic textbooks using Ontologies, Natural Language Processing (NLP) techniques and heuristic reasoning is presented.

3.1. - INTRODUCTION

DOM-Sortze entails a suite of applications and web-services which cope with the different tasks of building the *Domain Module*. Its architecture is presented in Figure 1 in which rounded boxes represent web services and rectangular boxes represent applications or modules. This web-service oriented architecture makes DOM-Sortze flexible and platform independent on the client side. However some platform-specific applications (mainly NLP tools) are used by the web-services.

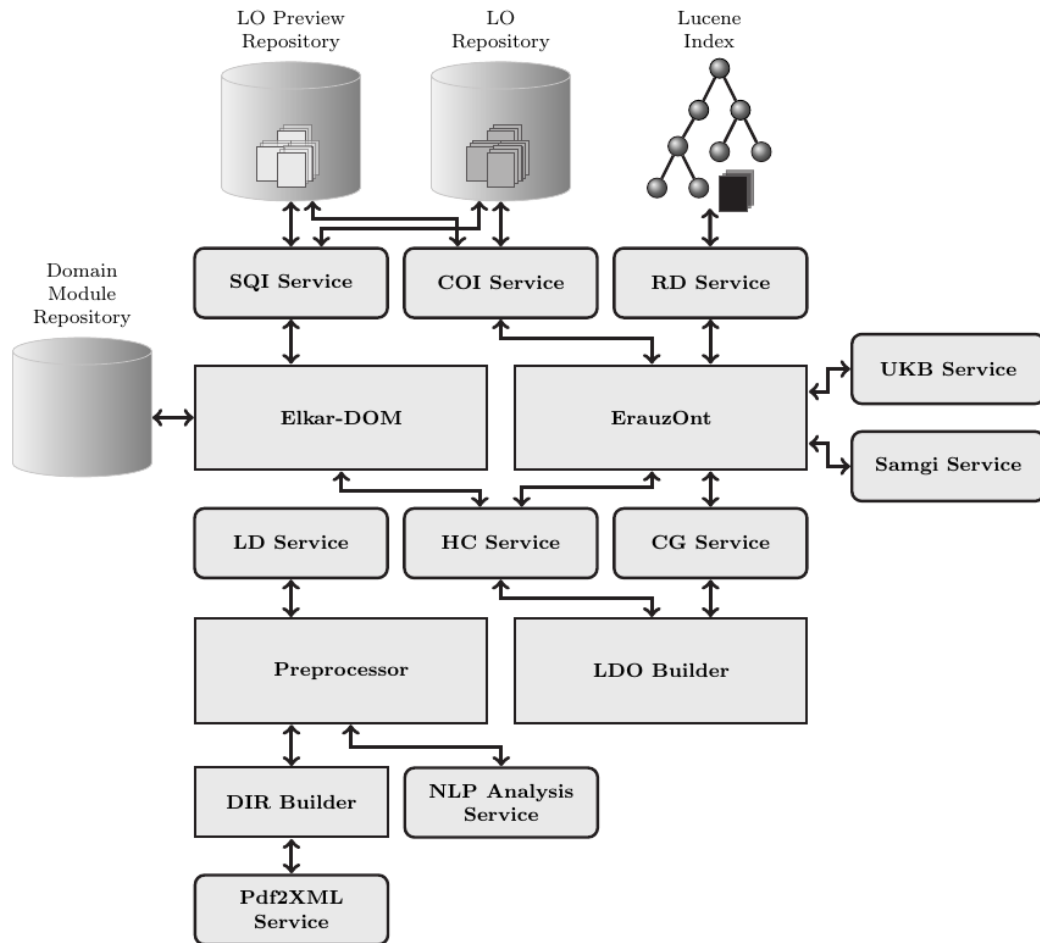


Figure 1- General architecture of DOM-Sortze

DOM-Sortze consists in four main applications – the *Preprocessor*, the *LDO Builder*, *ErauzOnt*, and *Elkar-DOM*. These carry out the tasks for building the *Domain Module*. The first three do the textbook processing tasks and the latter facilitates the intervention of authors, either instructional designers or teachers, in the *Domain Module* building process. These applications use some web-services as well to perform their job.

The storage of the LOs is provided by two repositories: the *LO Repository* stores the LOs (resources and metadata) and the *LO preview* repository keeps the preview files for the LOs. For making queries to the *Learning Objects Repositories* the Simple Query Interface (SQI) Service is used. The Content Object Inserting (COI) Service allows adding new LOs into the LOR.

The *Replicate Detection Service (RD service)* is used to determine whether a document or a fragment of a document has been processed before, preventing the processing of a document or fragment more than once. This service uses MD5 hash codes (Rivest, 1992) to achieve its goal. A *Lucene*¹ index is used to keep the information about processed resources, and a copy of the processed resources is tracked for safety, so that the Lucene index can be restored when a fails occurs.

The *Pdf2XML* service extracts the contents of the *pdf* files, providing a XML of the content of the document with its images. This service allows to the *Document Internal Representation Builder (DIR Builder)* to acquire the internal representation of the document and its outline. The *Natural Language Processing Analysis service (NLP Service)* returns the part-of-speech information for a text. The *Constraint Grammar Service (CG Service)* is used to carry out the grammar-based analysis and the *Heuristic Confidence Service (HC Service)* returns the confidence of the heuristics used during the analysis. The Graph Bases Word Sense Disambiguation and Similarity: UKB *Service* provides the similarity measures that we use to compound the LOs. The *SAmgl Service* facilitates the automatic annotation of the generated LOs.

In the following sections the main applications are described.

¹ - <http://lucene.apache.org/core/>

3.2. - PREPROCESSOR

Electronic textbooks are available in different electronic formats (e.g., *pdf*, *doc*, *html*, *etc.*), despite this, usually the documents are structured in a hierarchical structure (chapters, section...). However, authors or publishing companies used different numbering or structuring styles. Consequently, the textbooks have to be prepared before proceeding with the knowledge acquisition tasks.

The Preprocessor carries out the initial process of the document. It uses the *DIR Builder* module to obtain the internal representation of the document and its outline. The *DIR Builder* provides to the framework a way to process a document without worrying about its format. Nevertheless, currently only support *pdf* documents and, therefore, the *Pdf2XML* is used to build the internal representation of the document and its outline. The *Language Detection service (LD Service)* is utilized to identify the language in which the document is written. The *NLP Analysis Service* provides to the preprocessor part-of-speech information of the text fragments. Currently the Basque language is supported, and this service uses *EUSLEM* (Aduriz et al., 1996) to perform the linguistic analysis.

Textbooks are organized in a tree-like structure with chapters, sections, etc. Therefore, a Tree-Like class structure has been designed to represent the electronic textbooks. Figure 2 shows the class diagram of this structure.

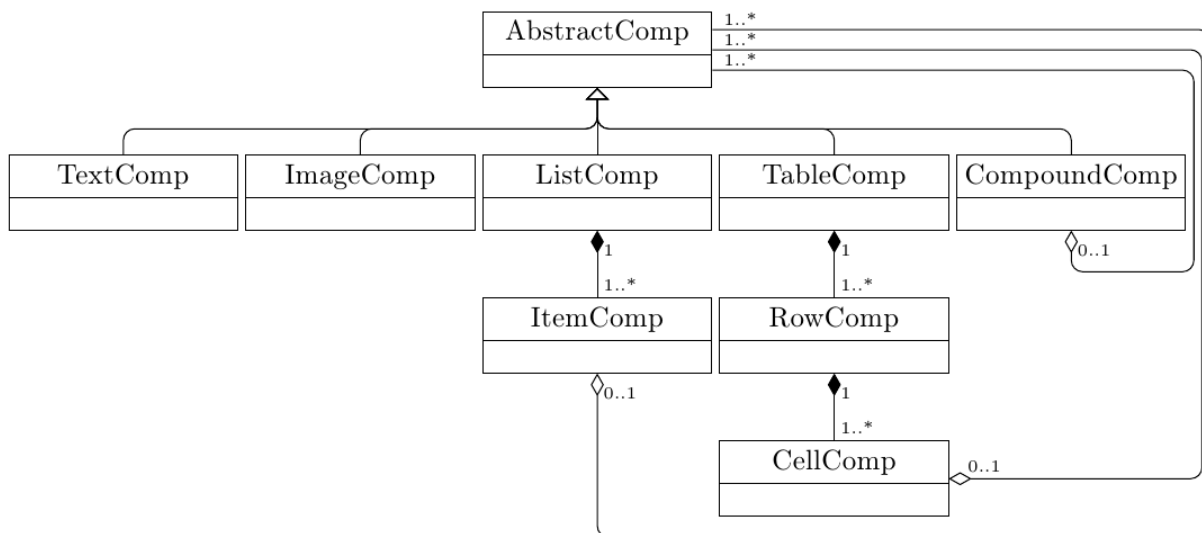


Figure 2 - Class Hierarchy for the Tree-Like Document Representation

3.3. - LDO BUILDER

The *LDO Builder* builds the Learning Domain Ontology (LDO), which contains the main domain topics and the pedagogical relations among them, from the internal representation of the document and its outline. The topics of the LDO are gathered using the whole document and its outline. For whole document topics identification *Erauzterm* (Alegria et al., 2004; Gurrutxaga et al., 2005) is used.

The pedagogical relationships are discovered among topics using a pattern-recognition approach. Some pedagogical relationships are defined from the outline by heuristics and a inference engine, while others, are recognized by the analysis of the whole textbook using the *Constraint Grammar* (CG) Service. This service uses the Constraint Grammar Formalism (Voutilainen & Tapanainen, 1993; Tapanainen, 1996) to recognize patterns. The reliability of the employed heuristics change from document to document. Thus, the HC Service is used to get the confidence of the patterns.

To describe the gathered LDO a XML-based formalism is utilized. The Listing 1 shows a fragment of a LDO described in this formalism. As we can see, the information about the heuristics and their confidence is also included to facilitate the supervision process depicted later.

The formalism for describing the LDO also supports the description of the kind of topic, its relevance and the difficulty level, although these features are not currently automatically elicited from the textbooks.

```
<?xmlversion=" 1 . 0 "encoding="UTF-8"?>
<TopicSet>
  ...
  <Topic>
    <ItemId>T2</ItemId>
    <ItemContent>Gailulogikoprogramagarriak (PLDak)</ItemContent>
    <DRS/>
  </Topic>
  <Topic>
    <ItemId>T21</ItemId>
    <ItemContent>PAL</ItemContent>
    <DRS/>
  </Topic>
  ...
</TopicSet>
<RelationSet>
  <Relation>
    <RelationID>IS-A36</RelationID>
    <Target>T2</Target>
    <Source>T21</Source>
    <Category>
      <InferredCategory>IsInferredCategory>
      <InferredBy>
      <UsedHeuristic>
      <HeuristicName>AH</HeuristicName>
      </UsedHeuristic>
      </InferredBy>
      <Confidence>0 . 9</Confidence>
    </Category>
  </Relation>
  ...
</RelationSet>
```

Listing 1- Fragment of the LDO

3.4. - ERAUNZONT

ErauzOnt (Larrañaga et al., 2011) is the application responsible of gathering the LOs from the electronic textbook. The architecture of *ErauzOnt* is presented in Figure 3. The Learning Object Extractor and Generator is the core of *ErauzOnt*, it is responsible of generating LOs from the internal representation of the electronic textbook. It uses the *CG Service* to identify the fragments of the text that may contain DRs and the *HC Service* to obtain the confidence of used heuristics. The *UKB Service* provides the resemblance for the ontology bases similarity measuring methods employed to see whether two DRs should be combined or not.

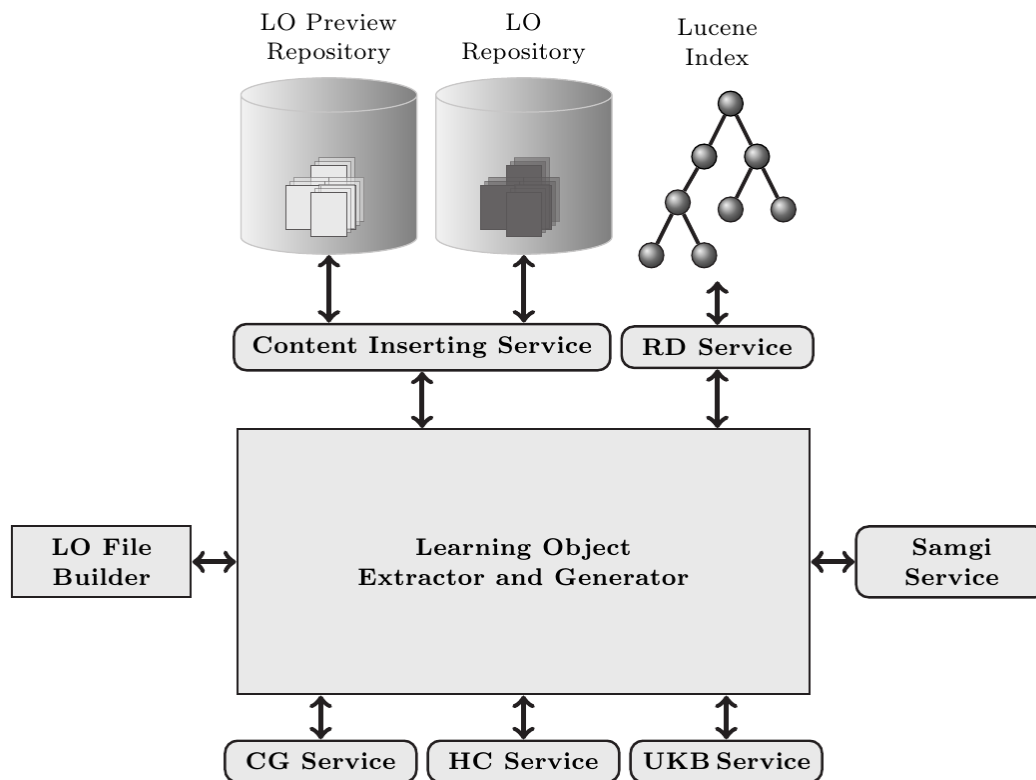


Figure 3 - ErauzOnt architecture

In the next chapter we will extend in describing ErauzOnt and its job, as this is the application which we focused our master thesis work.

3.5. - ELKAR-DOM

Elkar-DOM has two main goals. It allows to the user of the framework to supervise and modify the LDO and allows to the instructional designers or teachers to select the most appropriate LOs for each domain topic. *Elkar-DOM* is based in Elkar-CM (Arellano et al., 2006; Elorriaga et al., 2011) providing a graphical collaborative way using Concept Maps to fulfill its goals. The nodes of the concept map represent the topics and the links the relationships among them.

Elkar-DOM has been developed with the aim of enhancing collaboration in the domain knowledge building process. It allows synchronous collaboration based on token-passing. Several users could work at the same time seeing the current state of the domain ontology but only one of them can perform operations on it at a time. When a user wants to modify the Domain Module, he or she must request the token. Once obtained the token, the user could work on the ontology.

In the Figure 4 the architecture of *Elkar-Dom* is presented. The *SQI service* is used to search and retrieval of LOs from the LORs, and the *HC Service* is employed to modify the confidence of the heuristics as the acquisition of the LDO relies on them and their confidence. Besides, a client for interacting with the server, *Server Management Client*, and a client for authoring the Domain Module, *Domain Module Authoring tool*, has been developed.

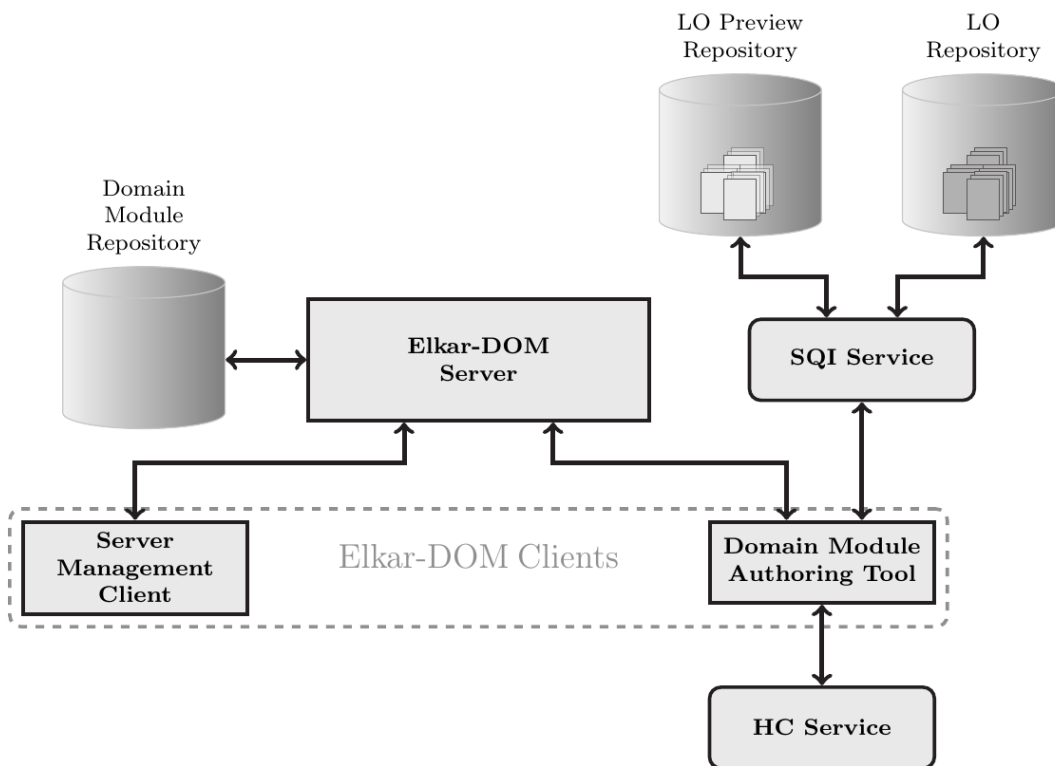


Figure 4 - Elkar-Dom architecture

The client side application that allows to the user to inspect and refine the LDO is shown in Figure 5. This tool is based on concepts maps, which have been used to allow knowledge elicitation and exchange (Coffey et al., 2004). Therefore, they also can be a powerful tool to facilitate users work in Domain Module authoring. Moreover, (Suthers, 2005) observed that concept maps also facilitate the interaction in collaborative tasks, such as collaborative learning. Therefore, concept maps might be an appropriate means for Domain Module authors to cooperate on the supervision of the Domain Module authoring in the same way they collaborate to prepare the material and the schedule for their courses.

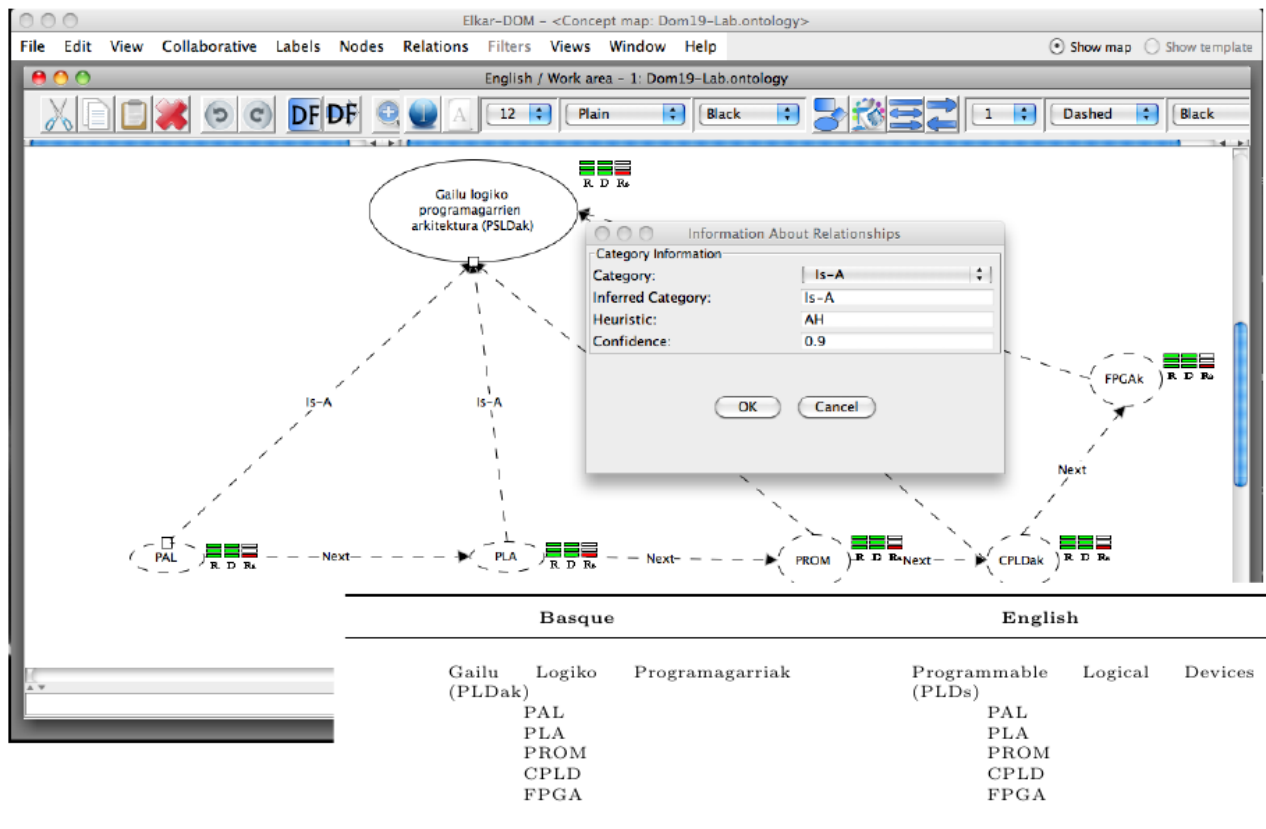


Figure 5 – Snapshot of Elkar-DOM

To get a complete Domain Module, the LOs to be used during the learning sessions must be provided for every domain topic. Elkar-DOM facilitates this task to the Domain Module authors, as it allows the search and retrieval of the LOs from the LOR through the *SQL Service*. The graphical interface is presented in Figure 6.

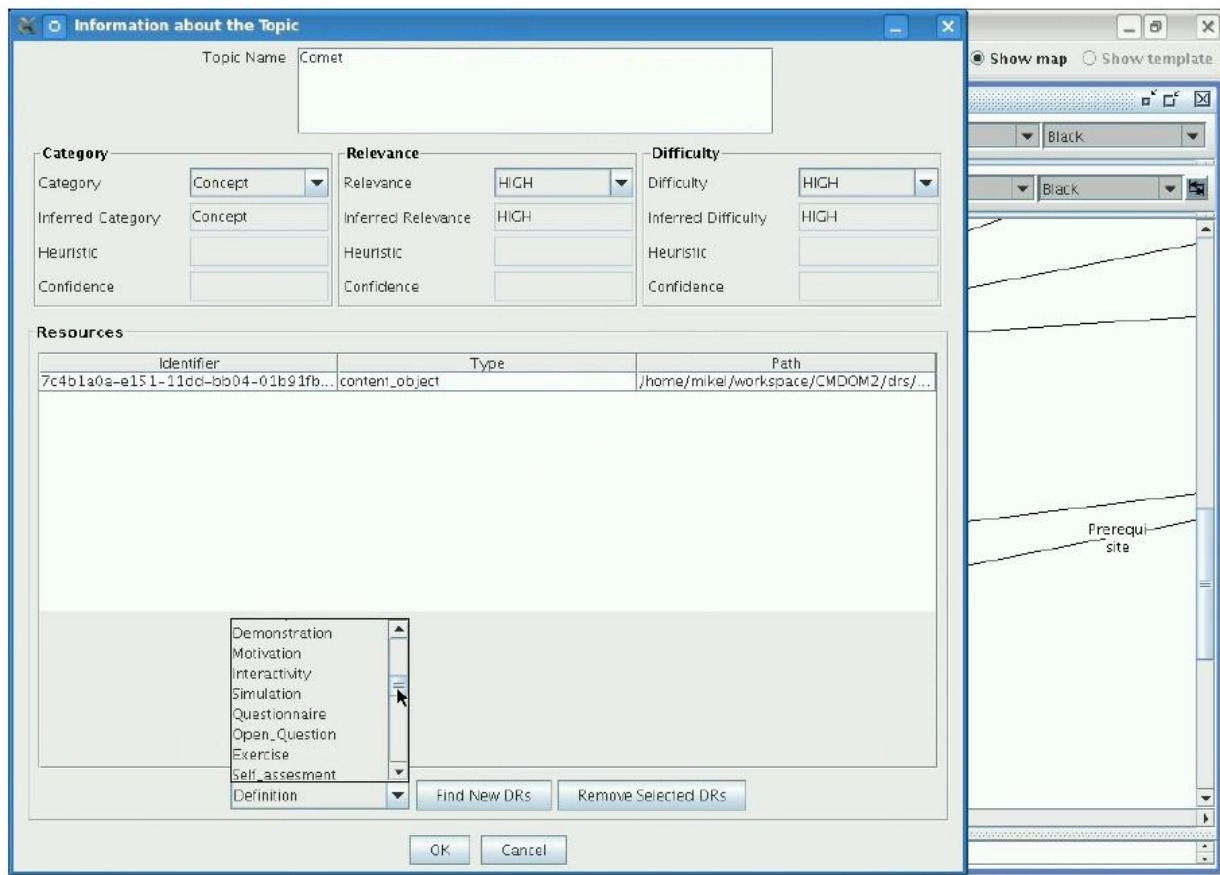


Figure 6 – Search and selection of LOs using Elkar-DOM

3.6. - SUMMARY

The architecture of *DOM-Sortze* has been presented in this chapter. The architecture is composed of several modules aiming to be scalable and modular. This makes easier the introduction of new features in the framework.

CHAPTER 4 - ERAUZONT

In this chapter the process that is carried by *ErauzOnt* and the components needed by the tool are described.

4.1. - OBTAINING LO FROM ELECTRONIC TEXTBOOKS USING ERAUZONT

The process that carries out *ErauzOnt* to acquire LOs requires an electronic textbook and the LDO which will guide the acquisition of LOs from the textbook. The LDO can be semi-automatically gathered from the electronic textbook using the LDO Builder of Dom-Sortze.

The generation of LOs from the electronic textbooks entails identifying and extracting the relevant DRs, their annotation with LOM and storage in the LOR. The DRs acquired are mainly text-based. However, they might also contain some images to illustrate the topics that are contained in the DRs.

LOs are gathered from the electronic textbook by carrying out the process described in Figure 7.

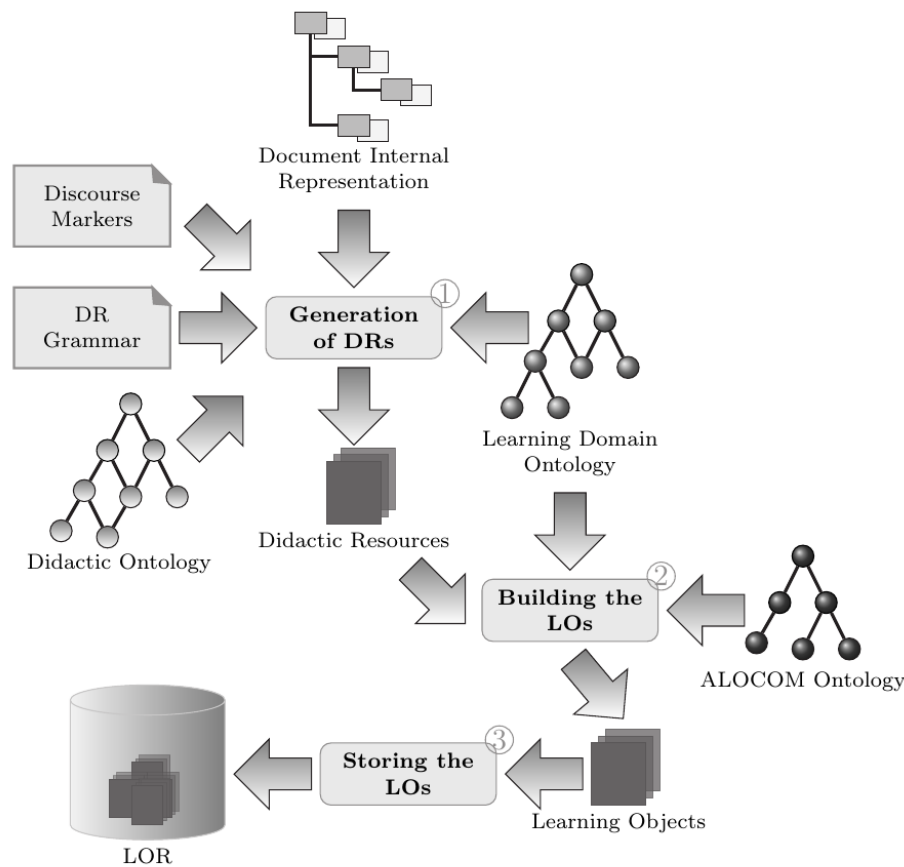


Figure 7 - Generation of Learning Objects

The LO generation aims to be domain-independent. Therefore, the only domain-specific knowledge used is the LDO, which has been gathered from the electronic textbook in the previous phase using the *LDO Builder*. The process that identifies and extracts the DRs is performed following a pattern-based approach. The searched text fragments are restricted to domain topics described in the LDO. The gathered DRs are aimed at being coherent and cohesioned. NLP techniques that combine a DR grammar and discourse markers are used, together with a didactic ontology (Meder, 2000, p. 200; Leidig, 2001), i.e. an ontology that describes the different kinds of DRs than can be used in learning sessions, to achieve this goal.

Once the DRs contained in the textbook have been identified and gathered, LOs are built from them. After this, the metadata for each LO is generated to assure that the LOs can be found and retrieved from the LOR they are stored in. This metadata can be manually built to each LO by teachers or can be automatic built trying to avoid differences and inconsistencies in the annotation process that a manually generated metadata may have. The LDO and the ALOCOM ontology (Verbert et al., 2005) are used to ensure LO reusability.

Finally, the LOs are stored in the LOR so that they can be reused either for the Domain Module being developed or any future TSLs. As in any semi-automatic approach, human intervention is desirable to assure the quality of the results. Thus, the supervision of the LO acquisition is also supported using Elkar-*DOM* tool.

In the next sections the generation process in detail, this process entails generating the DRs, enhancement of them, assuring the cohesion of the enhancement of the DRs and finally the process to build LOs from DRs.

4.1.1 - Generation of the DRs

This process is carried out by finding relevant text fragments for the LDO topics. Textbook authors usually use quite similar patterns (syntactic structures) for defining topics, describing theorems or proposing exercises. These patterns are used to gather some of the kinds of DRs described in the didactic ontology, namely, definitions, examples, facts, theories, principle statements, and problem statements, from the electronic textbooks.

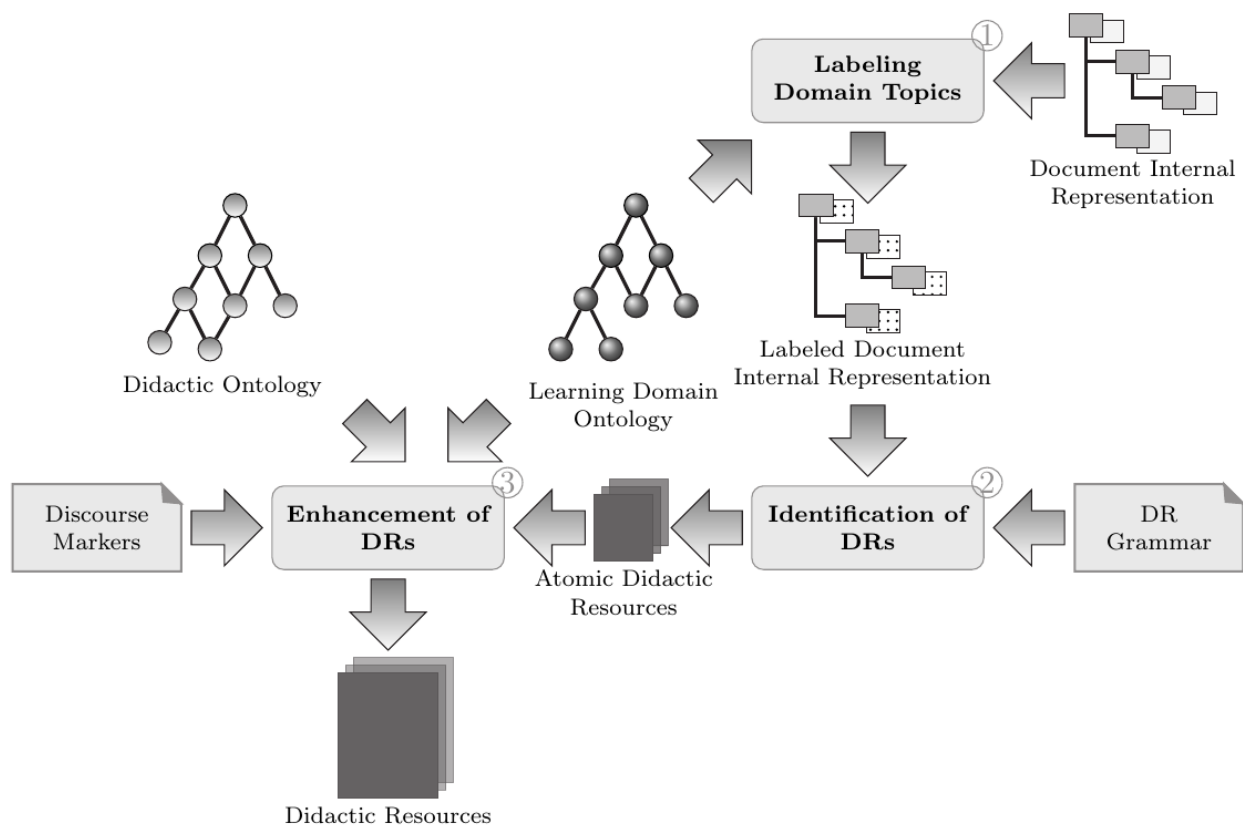


Figure 8 - Generation of Didactic resources

The process is described in Figure 8. The appearances of the LDO topics are labeled in the textbook internal representation built in the pre-process of the document. Next, the DR grammar is used to find text fragments that might contain appropriate resources. The DR grammar contains a set of rules that recognize the syntactic structures used to present the different kinds of DRs, e.g., topic definitions, examples, etc. Similar patterns are used for English in (Liu et al., 2003; Verbert, 2008) to look for definitions. The grammar for gathering the DRs from the electronic document has also been developed using the Constraint Grammar formalism.

The DR grammar was tested on electronic textbooks written in Basque language to observe its performance. Some of the initially defined rules were removed from the final version of the DR grammar, as they had low precision. The precision of the grammar rules is used to determine the confidence in these rules.

The identified atomic DRs contain the sentence that triggered the rule for the corresponding DR and all the sentences that follow it, as long as they refer to the same topic(s). Every DR is labeled with the domain topics and with the rules of the DR grammar that identified it. This information is used later in the LO annotation process.

The gathered DRs are then processed and enhanced in order to get more appropriate DRs and to assure the coherence and cohesion of their content. As a result of this process, some of the DRs might be combined with consecutive DRs or text fragments. The composite DRs are built as an aggregation of DRs of lower granularity and keep the information about why they were composed (cohesion maintenance or DR similarity) and the similarity rates. Besides, the referred topics and the DR grammar rules used to identify the DR are also kept in every DR (Figure 9).

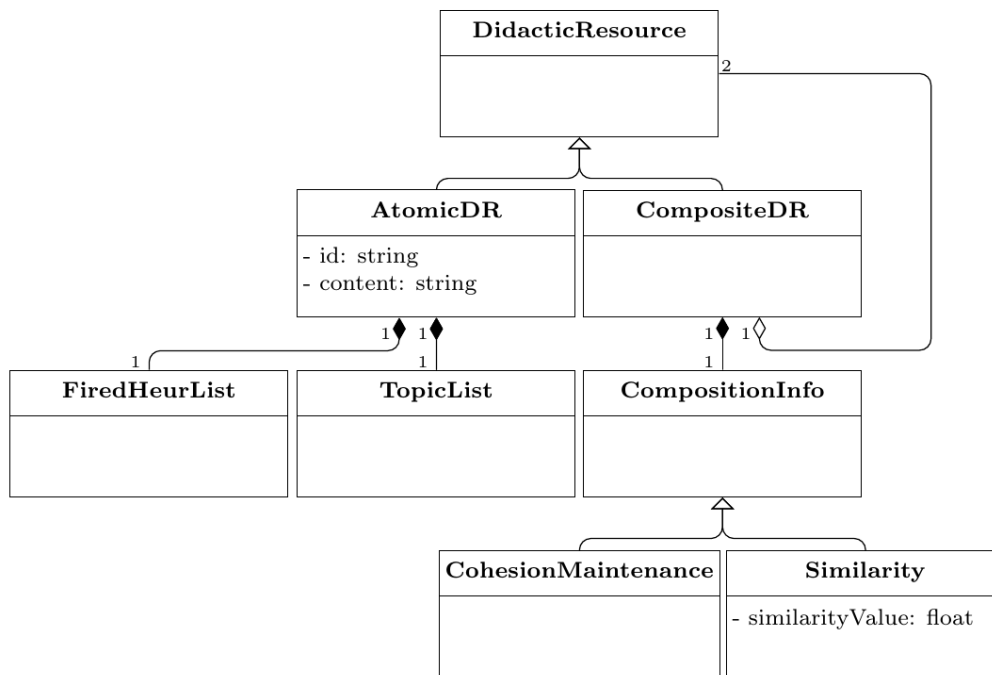
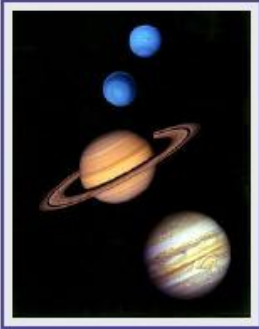


Figure 9 - Classes for the Internal Representation of DRs

4.1.1.1 - Identification of the DRs

Using DR grammar allows the system to locate sentences using any of the identified syntactic structures referring to LDO topics. For each selected sentence, an atomic DR is built. The atomic DRs also may contain the sentences that follow the selected one as long as they are not identified as other DR by the grammar, and they are content related. Content similarity is measured considering the domain topics referred in the text. Textbook authors may also include some sentences that do not necessarily include the domain topics but that connect different sentences that do refer to domain topics. An empirically established number of consecutive sentences of this kind are also allowed while building DRs, with the aim of being as complete and coherent as possible. Besides, every image found in the textbook is also considered a DR that requires no deeper processing.

...



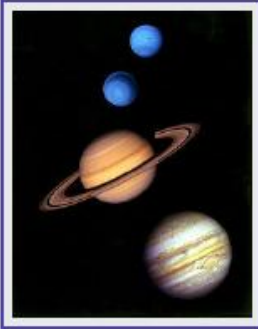
...

Planetak berezko argirik ez duten gorputzak dira, eta izar baten inguruan biraka mugitzen dira. Uste denez, Eguzki-Sistemako planetak Eguzkiarekin batera eratu ziren, eta pentsa daiteke antzeko planeta ugari izango direla beste izar batzuen inguruan.

Lurra planeta bat da.

...

...



...

Planets are non-self-luminous celestial bodies orbiting around a star. Solar system planets are thought to be formed together with the Sun, and it is also conceived that many similar planets might be found orbiting other stars.

The Earth is a planet.

...

Figure 10 - Example of Gathered DRs

Figure 10 shows a fragment of a document where some DRs can be detected. Three DRs are identified and constructed, the first is an image, and the last two are consecutive definitions. The pattern used to identify them is underlined. The definition of the “*planetak*” (“*planets*”) entails two sentences. The second one was added as it is related to similar domain topics, while the last sentence - “*Lurra planeta bat da.*” (“*The Earth is a planet.*”) - contains the definition of another topic, so a different DR has been built from it. In the next section some of the patterns used in DR grammar are presented.

4.1.1.2 - Example of patterns

The DR grammar includes a set of rules that recognize the different patterns or syntactic structures that were identified by manually analyzing a sample of documents. These patterns are the most common syntactic structures found in several topic *definitions, examples*, etc. The grammar for identifying DRs from electronic documents was developed using the Constraint Grammar formalism. In the next tables some of the used patterns are shown.

Pattern	@Topic definition (DAT^a) deitu
Example	<u>Unibertsoa</u> astro guztien multzoari eta betetzen duten espazioari deitzen zaio

Table 1 - Example of a pattern that allows identifying definitions

Pattern	@Topic,@Topic, oinarri izan
Example	Unibertsoko gainontzeko astroak bezala, Eguzkia, Lurra eta Ilargia mugitu egiten dira, eta era bat baino gehiagoko mugimenduak egiten dituzte, gainera. Lurreko fenomeno askok, esaterako <u>eguna</u> eta <u>gaua</u> , <u>eklipseak</u> , edo <u>itsasaldiak</u> , mugimendu horietan dute beren oinarria .

Table 2 - Example of a pattern that allows Identifying Principle Statements

Pattern	Erantzun galdera [det]
Example	Erantzun galdera ahu:

Table 3 - Example of a pattern that allows Identifying Problem Statements

4.1.2 - Enhancement of the DRs

The DRs identified by the grammar are usually quite simple; they entail a set of sentences about a particular domain topic or a group of domain topics. Those DRs can be enhanced in two ways in order to meet the principles for determining the granularity of the DRs stated by (Schoonenboom, 2006). In the one hand, combining two consecutive DRs, such as those shown in Table 4, may result in more useful DRs than the atomic ones. On the other hand, and to keep the cohesion of the DRs, previous fragments are added to a DR that contains references to those fragments.

Basque	
DR ₁	Planetak berezko argirik ez duten gorputzak dira, eta izar baten inguruan biraka mugitzen dira. Uste denez, Eguzki-Sistemako planetak Eguzkiarekin batera eratu ziren, eta pentsa daiteke antzeko planeta ugari izango direla beste izar batzuen inguruan.
DR ₂	Lurra Planeta bat da.

Table 4 - Example of two DRs that may be combined

Whether the referenced previous fragment is part of a DR, then both DRs are combined. Discourse markers, i.e., words or phrases that are used to link sentences, are employed to determine which DRs must be enhanced.

The enhancement of the DRs is crucial to obtain useful and reusable DRs, and is achieved following the algorithm presented in Figure 11 and based on similarity measuring methods. Every pair of consecutive DRs is tested to determine their resemblance. If they are considered similar, they are combined in a new DR that comprises them. Once the composition step has finished, the DRs undergo a cohesion assuring process (presented in the next section). This process is repeated as long as changes are made on the identified set of DRs.

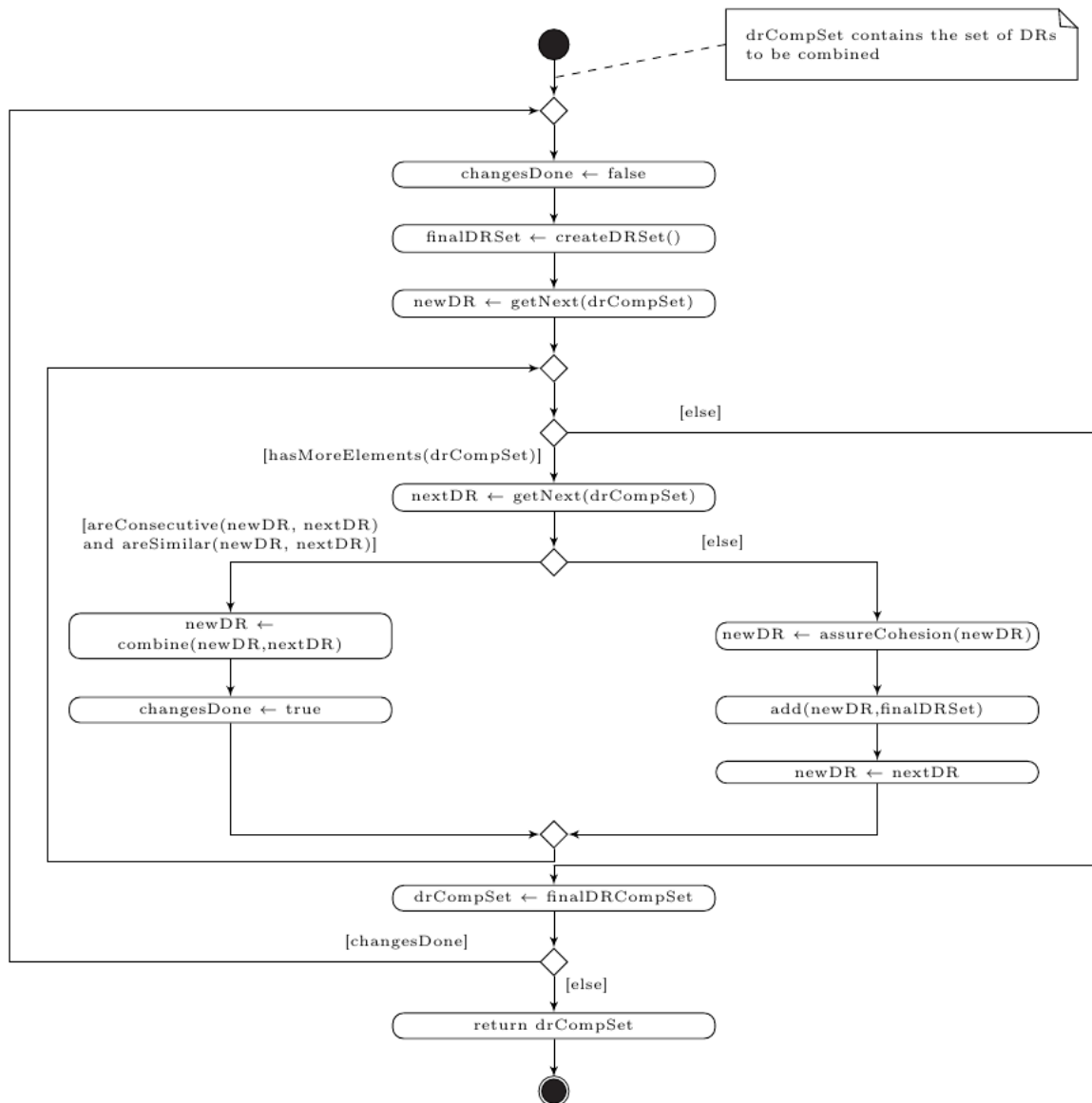


Figure 11 - Algorithm for the Composition of DRs

4.1.2.1 - Similarity measuring methods

Determining if two consecutive DRs are close enough is a key issue to obtain more accurate DRs. Two aspects are considered to determine if two DRs are suitable for combination. On the one hand, the content of the DRs is analyzed to measure their relatedness. On the other hand, the kind of DRs is considered. For instance, examples may enrich a topic definition, and thus, their combination may result in a better DR. However, problem statements are seldom combined with other DRs, unless a whole unit is expected to be built. Thus DR similarity or relatedness measuring methods for each of these aspects have been defined. These methods return a value in the [0, 1] range; the higher the value, the closer the DRs. Two DRs are considered similar if the obtained content similarity and the DR type similarity are beyond the corresponding threshold values or the combined similarity score is beyond the threshold, depending on the user's preferences.

4.1.2.1.1 - Content similarity measuring methods

Content similarity measuring methods determine the resemblance of two DRs according to their content, i.e., the topics of the domain they reference. ErauzOnt uses **Ontology Based Method**: this method uses the UKB tool (Agirre, Alfonseca, et al., 2009; Agirre & Soroa, 2009), an ontology based lexical similarity measuring application similar to Hughes and Ramage's Wordnet-based approach (Hughes & Ramage, 2007). For every analyzed fragment, UKB returns the stationary distribution of the LDO topics considering both the semantic relationships in the ontology and the topics referred in the analyzed fragment. The similarity is obtained using the cosine equation showed in Formula 1 on the stationary distributions of the compared fragments. This method proved to obtain the most accurate results compared to the instructional designers criteria.

$$\cos(\vec{d}, \vec{d}') = \frac{\vec{d} \times \vec{d}'}{|\vec{d}| |\vec{d}'|}$$

Formula 1 - Cosine equation

4.1.2.1.2 - DR type similarity measuring methods

For getting the similarity of the type of resource (example, definition, etc.) *ErauzOnt* uses **Didactic Ontology Method**: This method is similar to the Ontology-Based content similarity measure method but using the kinds of DRs instead of the domain topics. It uses a didactic ontology (Meder, 2000; Leidig, 2001), which represents the different kinds of DRs and relationships between those types, to compute the similarity between two DRs.

4.1.3 - Assuring Cohesion in the DR Enhancement

Discourse markers, i.e. words or expressions that connect part of a text with its context, are known to be related to the rhetorical relationships that govern the structure of the narratives (Knott & Dale, 1994; Taboada, 2006; Iruskieta et al., 2010). Therefore, they can be used as a means to assure (or at least try to assure) the cohesion in the gathered DRs. Sentences starting with particular discourse markers are likely to be related to the previous sentence or sentences. Therefore, DRs starting with a particular discourse marker will be enhanced by adding previous sentences or even the previous DR that the previous sentence or sentences are part of to assure the cohesion of the text. If the previous sentences are included in another DR both are combined in a new one.

Kind	Basque
References	Hau, hura, horiek, horri
Single	Gainera, horretarako, bestalde
Complex	Alde batetik → bestetik, hasteko → bukatzeko

Table 5 - Discourse Markers for Basque

Discourse markers are classified, independently of the related rhetorical relation, into three categories considering how the DRs that contain them have to be enhanced: single, complex and references. Single discourse markers - Gainera (Besides) or Horretarako (Therefore) - and references connect the sentences with the previous sentences. Complex discourse markers require two parts- for example, Hasteko . . . Bukatzeko . . . (First, . . . Finally, . . .). The system deals differently with each kind of discourse markers. If the DR starts with the second part of a complex discourse marker (e.g., Bukatzeko, . . .) it will add all the necessary sentences until the initial part (Hasteko, . . .) is included. References and single discourse markers usually regard up to an empirically gathered maximum number of sentences; thus, at most the maximum number¹ sentences are added in this case. Samples of the discourse markers for Basque are shown in Table 5.

4.1.4 - From DRs to LOs

The gathered DRs might be not only useful for the Domain Module being developed from the processed electronic textbook, but also for other Domain Modules. Thus, to facilitate their reuse, LOs, are built from the gathered DRs. Building reusable DRs entails two aspects: using an appropriate format to store and represent the content, and also describing it (annotating it) with LOM to allow searching in and retrieving those LOs from the LOR.

¹ The performed experiments showed that adding up to three previous sentences provided the best results. However, this value can be configured.

The generation of DRs might gather resources of different granularity ranging from atomic DRs to composite DRs that comprise finer grained DRs. Although the composite DRs might be more appropriate for a certain context, the entailed DRs might be also used in other contexts, so LOs are also built from the components of the composite DRs.

4.1.4.1 - LO File Format

Formats like *html*, *pdf*, *doc*, and *odf* are suitable for final presentation of a LO, but they are not appropriate for content reuse, as the components cannot be easily accessed. The ALOCOM framework (Verbert et al., 2008; Verbert, 2008) was developed to overcome this problem and facilitate the decomposition of composite LOs and make those components available for on-the-fly content reuse. This framework relies on the ALOCOM ontology (Verbert et al., 2005), which represents a content model for LOs and their components. The generated DRs are stored in a ZIP file that contains the XML file for the LO, based on the ALOCOM formalism, as well as the referenced images or other resources. Listing 2 shows an example of a LO using the ALOCOM format. ALOCOM ontology is used to categorize the LO too. Nevertheless, the ALOCOM ontology had to be enhanced to support theorems as they were not considered in the previous version.

Moreover, a preview file in rtf format is generated so that the user may have an approximate idea of the content of each LO while looking for resources about a certain topic.

```
<?xml version="1.0" encoding="UTF-8" ?>
<ALOCOMComponent id=" 467c3115-e0a6-11dd-aa6f-1b45350a80e7"
type="definition">
  <ALOCOMComponent type="definition">
    <ALOCOMComponent type="paragraph">
      <ALOCOMComponent type="text">Planets are space objects which do not
have their own light, and they move around a star.
      </ALOCOMComponent>
    </ALOCOMComponent>
  </ALOCOMComponent>
  <ALOCOMComponent type="example ">
    <ALOCOMComponent type="paragraph ">
      <ALOCOMComponent type="text">Earth is a planet.
    </ALOCOMComponent>
  </ALOCOMComponent>
</ALOCOMComponent>
</ALOCOMComponent>
</ALOCOMComponent>
```

Listing 2 - Example of a LO

4.1.4.2 - LO annotation

The likelihood to retrieve the desired LO from a large set or a LOR is a key issue to promote the use and reuse of LOs. For choosing a LO, the metadata that it contains is used. Because of this the metadata and its appropriateness is very important. While the manual creation of metadata can be considered for annotation of a single LO, it is not an option for larger LOs deployments (Duval & Hodgins, 2002; Cardinaels et al., 2005; Duval & Hodgins, 2004). Moreover, semi-automatic metadata generation can overcome metadata inconsistency problems by using ontologies (Kabel et al., 1999, 2004a, 2004b).

After an analysis of the LOM elements, and considering the kind of documents being processed, these elements were classified and it was concluded that most of them had similar values (Larrañaga et al., 2008a). Thus, the metadata generation is carried out in the following way. The initial metadata is automatically generated using SAMGI (Meire et al., 2007). Then, the metadata is enhanced with more information that has been extracted during the DR generation to improve some elements (keywords or Learning Resource Type). Most keyword annotation applications use statistical methods and rely on the frequency of the terms in the analyzed text, but do not consider semantic relationships among the topics. For example, a keyword extractor may identify Earth, Mars, Mercury, and Venus in a fragment of text if they appear in it, but it would not consider that all of them are planets, and therefore it would not infer planet as a keyword, as it is not aware of the semantic relationships among these topics. Thus, the LDO and the identified domain topics in the LO are used to get a more accurate keyword list, as the semantics relationships are taken into account. The Learning Resource Type is also specified in terms of the ALOCOM ontology (Verbert et al., 2005), which represents a content model for the LOs and its components.

For determining the Learning Resource Type, the rules of the DR grammar met by the content of the DR are used. As these rules may identify different kinds of DRs, the precision of the rules (% of times that the rule correctly identifies a DR) is used to determine which the most plausible kind is and which is therefore selected as the Learning Resource Type for the annotated LO.

4.2. - SUMMARY

In this chapter a summary of the *ErauzOnt* tool and its process has been presented. Also the components needed by this process have been described in order to know how the DRs are build.

CHAPTER 5 - EXTENDING ERAUZONT

In this chapter the process that has been followed to extend *ErauzOnt* to support a new language, English, is presented. An evaluation of this extension (Conde, A. et al., 2012), including the performance achieved by the DR grammar and LO acquisition process is also described.

5.1. - ADDING A NEW LANGUAGE TO ERAUZONT

The *ErauzOnt* framework has been developed to enable the automatic extraction of LOs from electronic textbooks. The framework aims to be applicable on any document no matter the domain it relates to. None of its components relies on implicit domain-specific knowledge. All the domain-specific knowledge are the domain topics and the relationships among those topics described on the LDO, which is the input for the LO extraction process together with the document to be analyzed.

ErauzOnt is designed to easily support new languages. Adding a new language entails building the LDO for the chosen language. The current specification for the LDO supports this feature and therefore, no further modifications are required.

Besides, for acquiring the relevant DRs from the textbooks *ErauzOnt* relies on NLP techniques, so an analyzer must be integrated for each supported language. The tool uses for Basque language the tool called *EUSLEM* (Aduriz et al., 1996). After providing a suitable analyzer for the desired language the output of it must be adapted to the format used by *ErauzOnt*. Also, it is necessary to configure to *ErauzOnt* use the new analyzer for that language. This is achieved establishing in *ErauzOnt* how the analyzer is called and which its configuration parameters are.

In addition, it is necessary to define the DR grammar that contains the syntactic patterns used for identifying the DRs, definitions, examples, principle statements, problems.... This process is described in section 4.1.1. Besides, the Discourse Markers for the new language need to be defined too. These markers are used to assure the cohesion of the generated DRs as shown in Section 4.1.3.

The changes that have to be done in order to support another language in *ErauzOnt* from the perspective of DR generation process can be seen in Figure 12. In Figure 13 these changes from the perspective of the architecture of DOM-Sortze can be observed

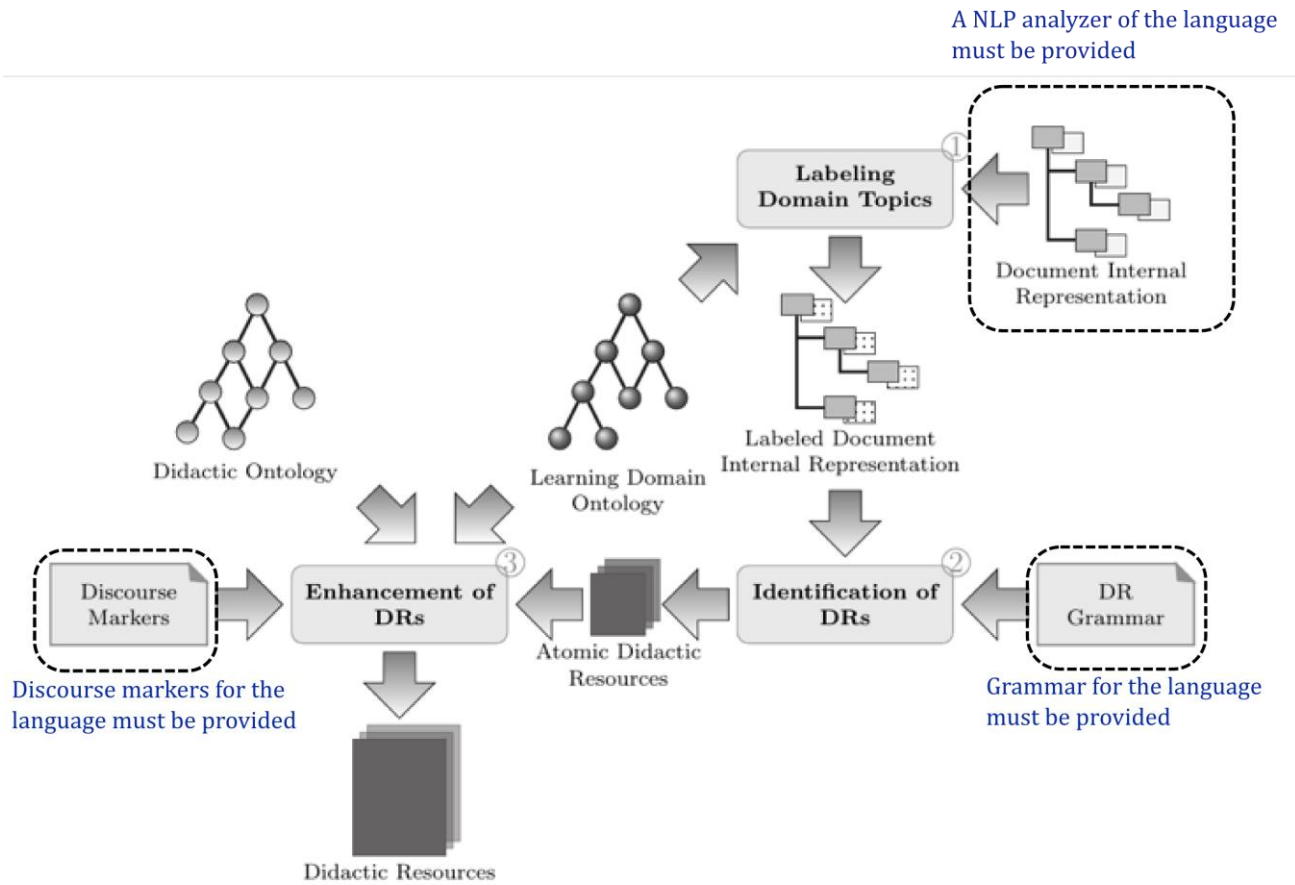


Figure 12 - Changes needed in DR generation process

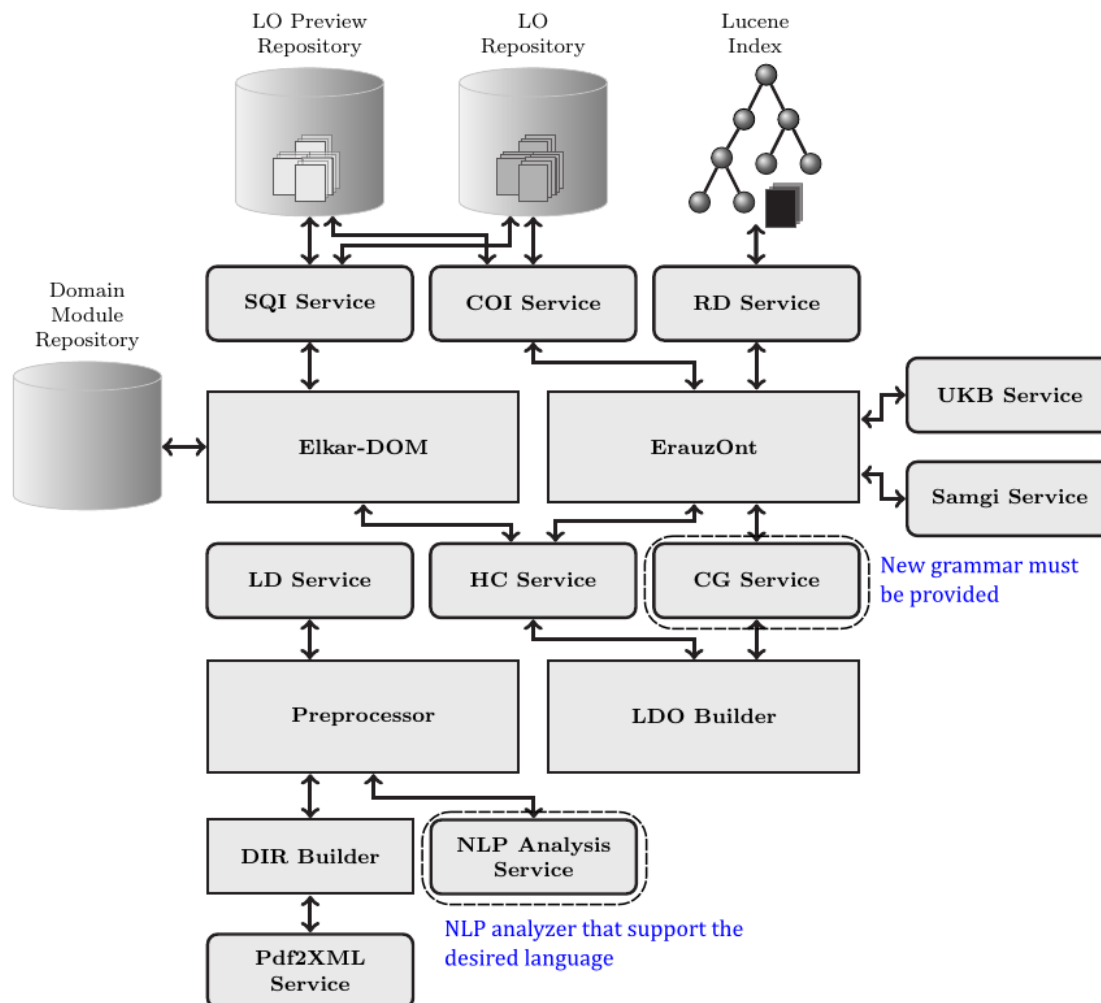


Figure 13 - Changes needed from DOM-Sortze architecture perspective

5.2. - ADDING ENGLISH SUPPORT TO ERAUZONT

In order to add English support to *ErauzOnt* it is necessary to use a tool that provides a part-of-speech analyzer for the language. For Basque language *ErauzOnt* use *EUSLEM* as analyzer, however *EUSLEM* only supports Basque language and therefore, for supporting English language another analyzer must be provided. For this work *FreeLing* (Atserias et al., 2002) has been chosen. *FreeLing* is a developer-oriented analyzer that supports several languages such as English or Spanish. The main advantage of *FreeLing* is that it is oriented to developers, which allows an easy integration with other systems and tools. It has few configuration files (only the directory for the language models, and the options for the analysis) which make the integration straightforward.

The next step entails the output of the analysis made by *FreeLing*. This output needs to be transformed to the format that *ErauzOnt* understands. *FreeLing* uses the PennTreeBank (Marcus et al., 1993) Tag Set for the *part-of-speech* analysis of English texts. Part of the Tag Set is shown in Table 6.

TAG	Description	Examples
NNPS	noun, proper, plural	Americans Americas Amharas Amityvilles
NNS	noun, common, plural	undergraduates scotches products bodyguards
PDT	pre-determiner	all both half many quite such sure this
PRP	pronoun, personal	hers herself him himself his self it itself me
PRP\$	pronoun, possessive	her his mine my our ours their thy your

Table 6 - Part of Penn Treebank Tag Set

Using this information, the *FreeLing* output is transformed to follow the structure described on Listing 3. An excerpt of part-of-speech information of a sentence acquired using *FreeLing* (after transforming it) is presented on Listing 4.

```
"<Word>"[ Extra Information ]
"Lemma" TAG ...
"<Word>"[ Extra Information ]
"Lemma" TAG ...
..
```

Listing 3 - Structure of the Output of the Linguistic Analysis

"<This>"	"this" DT Determiner "this" PRP Personal pronoun
"<computation>"	"computation" NN Noun, singular or mass
"<is>"	"be" VBZ Verb, 3rd person singular present
"<given>"	"give" VBN Verb, past participle "given" JJ Adjective "given" NN Noun, singular or mass
"<a>"	"1" Z null "a" DT Determiner "a" NN Noun, singular or mass "a" NNS Noun, plural
"<name>"	"name" NN Noun, singular or mass "name" VB Verb, base form "name" VBP Verb, non-3rd person singular present
"<:>"	":" Fd null
"<rectangle>"	"rectangle" NN Noun, singular or mass
"<.>"	"." Fp null

Listing 4 – Excerpt of the Part-of-Speech Information for a Sentence

In addition, it is necessary to define the DR grammar that contains the syntactic patterns used in English for the DRs. In the next Tables some of the patterns for DR identification are described.

Example	Pattern	Pattern (CG2)
A class <u>is an</u> abstract description of a set of objects.	{concept}+ {is are} + [determiner]	MAP:DEF (&DEF) (@ONT-TOPIC) IF (1 ("be" VBZ) LINK 1 (DT));
Java <u>refers to</u> a programming language.	{concept} + {refer} + [adverb]	MAP:DEF (&DEF) (@ONT-TOPIC) IF (1 ("refer" VBZ) LINK 1 ("<to>"));
Java <u>is defined as</u> a programming language.	{concept} + {is are} + {defined}	MAP:DEF (&DEF) ("be" VBP) IF (NEGATE -1 ("that")) (NEGATE -1 ("this")) (1 ("define" VBN) LINK 1 ("<as>") LINK *1 (@ONT-TOPIC));
<u>That is called</u> a method of a class.	{This That} +{is}+ {called} + {concept}	MAP:DEF (&DEF) ("be" VBP) IF (NEGATE -1 ("that")) (NEGATE -1 ("this")) (1 ("call" VBN) LINK *1 (@ONT-TOPIC));
Classes ; fundamental building blocks of Java programs.	{concept} + {:}	MAP:DEF (&DEF) TARGET (@ONT-TOPIC) IF (1 (":"));

Table 7 – Definition patterns

Example	Pattern	Pattern (CG2)
<u>For instance, the</u> Apple_class would extend the class_Fruit.	For instance e.g. for example as an example + [,] + [adverb] + {concept}	MAP:ADIB (&ADIB) TARGET ("for_instance" RB) IF (1 (DT) LINK 1 (@ONT-TOPIC));
String.toString() is an <u>example</u> of a method .	{example instance case illustration sample specimen} [of] {concept}	MAP:ADIB (&ADIB) TARGET ("example" NN) IF (1 (@ONT-TOPIC));

Table 8 – Example patterns

Example	Pattern	Pattern (CG2)
<u>Problem:</u> Given a rectangle compute its area.	{Problem} + {;}	MAP:ARIK (&ARIK) TARGET ("problem" NN) IF (1 (":" Fd));
<u>Answer the</u> following <u>question</u>	{Answer} + [determiner] + [next following] + {question}	MAP:ARIK (&ARIK) TARGET ("answer" VBP) IF (1 (DT) LINK 1 ("follow" VBG) LINK 1 ("question" NN));

Table 9 – Problem patterns

Besides, the Discourse Markers for English need to be defined too. These are described in Table 10.

Kind	Basque
References	This, that, these....
Single	Besides, therefore, however...
Complex	On the one hand → On the other hand, First → finally....

Table 10 – Discourse Markers for English

ErauzOnt can work with any language providing a NLP tool that works with the desired language, building the DR grammar for that language and defining the discourse markers. When a document written in a supported language is processed, *ErauzOnt* uses the appropriate resources, i.e., NLP analyzer, DR grammar and Discourse markers for the document according to the language it is written in.

5.3. - EVALUATION OF ERAUZONT

Evaluating *ErauzOnt* entails the following procedure: the teachers of the subject define the LDO that describes the topics to be learnt as well as the pedagogical relationships among the topics. The teachers manually analyze the textbook to identify and label the set of DRs (definitions, examples, etc.) that would like to use for mastering the main topics of the subject. Then, the LDO is used to process the textbook with *ErauzOnt*, and a set of LOs is obtained and stored in a learning object repository. The set of automatically elicited LOs is assessed by instructional designers

to determine their adequacy, to which end the set of LOs manually identified by the teachers is compared. The process is described in Figure 14.

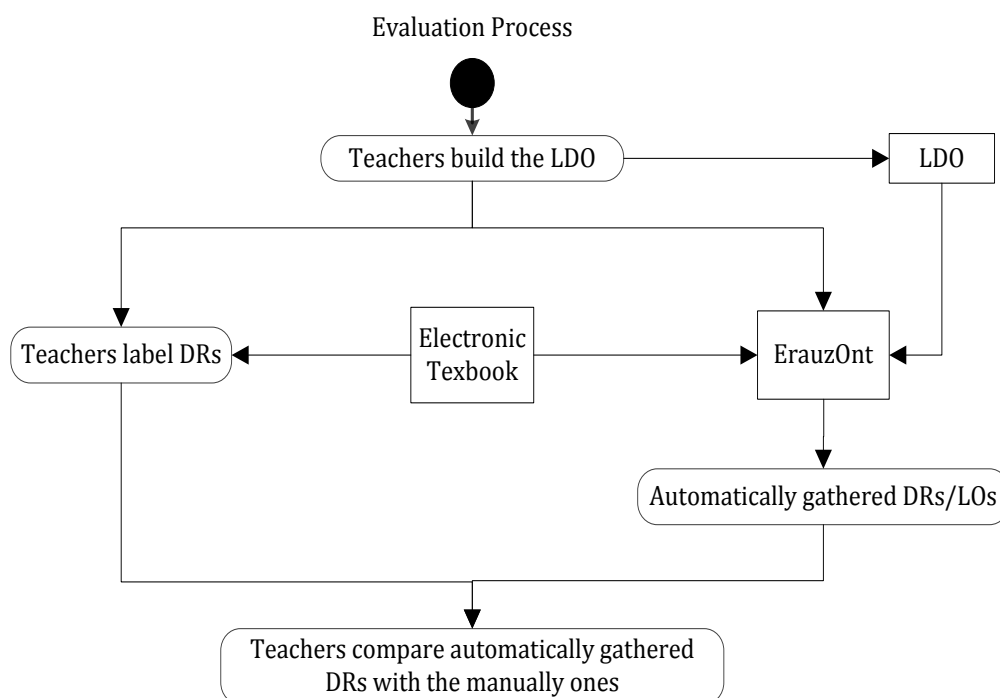


Figure 14 - Diagram of the process of evaluation of an electronic textbook

5.3.1 - ErauzTest: a tool for the evaluation of the gathered LOs and the DR grammar

The task of evaluating the performance of the framework requires a lot of manual effort. Therefore, a tool that aims in reducing this work has been developed.

The tool works as follows. It gets a list with all the simple LOs, and then it tests them to know which is the rule and topic that have been activated to build each one. After doing this, the tool tries to highlight in the electronic book all the LOs with their associated rules. If there are some LOs that can not be marked are stored in a Comma-Separated Values (*CSV*) like style document.

The architecture of the tool is presented in Figure 15. The tool depends on a file with all of simple LOs. This file is obtained getting the acquired LOs (an XML file) from the LOR database and filtering this file. It also uses the *NLP Analysis Service* and *CG Service* to get the information for each LO.

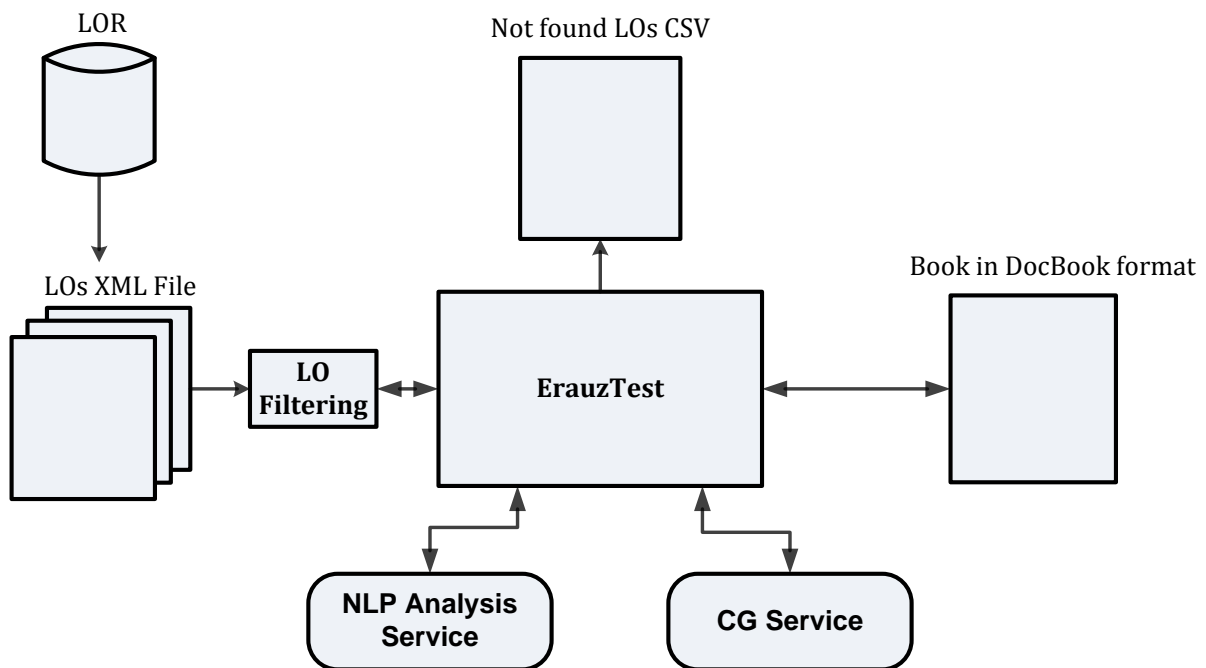


Figure 15 - ErauzTest architecture

The tool works with electronic textbooks using *DocBook*¹ format. *DocBook* is a semantic markup language for technical documentation; however, it can be used for any other sort of documentation. As a semantic language, *DocBook* enables users to create document content in a presentation-neutral form. *DocBook* is an XML language and its XML Schema is quite simple. An example of the format is shown in Listing 5. However, as the schema is quite simple, the system do not have a lot of options to highlight the LOs, and future upgrades of the tool may take advantage of more powerful formats like *Open Document*².

The conversion of the electronic textbook to *DocBook* format and backwards is achieved using *Open Office*³ suite.

1- <http://docbook.org/>

2 - https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=office

3 - <http://www.openoffice.org/>

```

<?xmlversion="1.0"encoding="UTF-8"?>
<bookxml:id="book"xmlns="http://docbook.org/ns/docbook"version="5.0">
  <title>Very simple book</title>
  <chapterxml:id="chapter_1">
    <title>Chapter 1</title>
    <para>Helloworld!</para>
    <para>I hope that your day is proceeding
    <emphasis>splendidly</emphasis>!</para>
  </chapter>
  <chapterxml:id="chapter_2">
    <title>Chapter 2</title>
    <para>Helloagain, world!</para>
  </chapter>
</book>

```

Listing 5 - DocBook XML example

The schema chosen to highlight each LO is presented in Table 11.

LO start	LO end	LO data
	\$\$\$\$\$	Id#rule trigger #rule

Table 11 - LO highlight structure scheme

In Listing 6 an example of highlighted *DocBook* opened in *OpenOffice* is shown.

The screenshot shows a document with three paragraphs of text. Each paragraph starts with a vertical bar icon (|||||||) and ends with a dollar sign icon (\$\$\$\$\$). The text in the paragraphs is highlighted. On the right side of the document, there are two red comment boxes. The first comment box is titled 'Comentario [19]' and contains the text '46504ac8-831e-11e1-9061-8b60af073d96##or##DEF-12'. The second comment box is titled 'Comentario [20]' and contains the text '452ba0e6-831e-11e1-9061-8b60af073d96##Objects##DEF-2'. Red lines connect the comment boxes to the corresponding paragraphs in the document.

Listing 6 - Example of highlighted DocBook

The process of evaluation of the performance gets easier with this tool as it is possible to compare the manual gathered DRs with the ones acquired by *ErauzOnt* using a visual tool like *OpenOffice*.

5.4. - EVALUATION OF ERAUZONT FOR ENGLISH

In this section an evaluation of the English extension of *ErauzOnt* is presented. The evaluation is made over a textbook oriented to Computer-Engineering students.

The evaluation has been carried out removing the images of the textbook to assess the performance of the acquisition of text-based LOs. The English analyzed textbook is *Principles of Object Oriented Programming*¹ (Wong, S & Nguyen, D., 2010), used in the Object-Oriented Programming subject of Rice University, Texas. The book consists of 67 pages.

For this evaluation both the performance of the DR grammar and the gathered LOs were tested, as this was the first experiment with *ErauzOnt* over documents written in English.

5.4.1 - Evaluation of the DR Grammar for English

The DR grammar has been evaluated by analyzing the atomic gathered LOs, i.e., the finest grained LOs. Each LO has been inspected to determine which rules were used to identify it and, therefore, to obtain the accuracy of the DR grammar.

The Table 12 shows the statistics about the evaluation of the DR grammar. The DR grammar is able to identify definitions, examples, problem statements, principle statements, facts and theories. However, not every kind of DR is always used. Neither facts nor theories were used in the analyzed textbook. The DR grammar built for identifying the syntactic patterns commonly used in DRs achieved 80.09% accuracy. The average of the rules acquisition ranges from 100.00% for the examples to 58.33% for the problem statements.

	Definitions	Examples	Problem Stat.	Principle Stat.	Total
Found	164	1	12	49	226
Correct	138	1	7	35	181
Accur. (%)	84.15	100.00	58.33	71.43	80.09

Table 12 - Accuracy of the DR Grammar

The DR grammar achieved similar results to previously conducted experiments over textbooks in the Basque language (Larrañaga et al., 2008b), except that the accuracy for problem statements was considerably lower, mainly because imperative cases, frequently used to state problem statements, are easier to identify in Basque, which uses an auxiliary verb for that purpose. The identification of the problem statements in English mainly relies on the appearance of keywords such as “exercise”.

¹ <http://cnx.org/content/col10213/latest>

5.4.2 - Evaluation of the LO Acquisition Process for English

The evaluation of the gathered LOs was carried out comparing the manually identified DRs with the automatically gathered ones. The evaluation of the gathered LOs considered both their appropriateness (precision) and the quantity of the manually defined DRs that were automatically identified (recall). An aspect to be considered to evaluate the gathered LOs is that while a LO might be the most accurate in a particular context, one of its components or a more complex LO (a composite LO that comprises it) might fit better in other situations.

In order to obtain the recall of the LO acquisition process, the automatically gathered LOs were compared to the manually identified ones. The teachers identified 54 DRs, 35 definitions, 2 problem statements and 17 combined DRs, i.e., DRs that entail two or more DRs of different kind. *ErauzOnt* achieved a 75.93% recall, i.e., 41 of 54 manually identified DRs were automatically gathered. 100% of the combined DRs, 62.86% of the definitions and 100.00% of the problem statements were automatically gathered. Problem statements are identified using verbs in imperative case or keywords such as “exercise” making its detection easy, whereas definitions usually appear in many different forms making them difficult to find. These results are presented in Table 13.

	Definitions	Problem Statements	Combined DRs	Total
Real	35	2	17	54
Found	22	2	17	41
Recall (%)	62.86	100.00	100.00	75.93

Table 13 - Recall of the LO acquisition process

These results are also similar to the ones found in (Larrañaga et al., 2012) for Basque language.

Determining the precision was not so straightforward because all the gathered LOs and their components had to be analyzed. Therefore, each generated LO was observed to determine whether it was valid, not only considering the subject for whom the textbook was analyzed but any other context.

	Definitions	Problem Statements	Combined DRs	Total
Found	140	2	229	371
Correct	121	2	199	322
Precision (%)	86.43	100.00	86.90	86.79

Table 14 - Precision of the LO acquisition process

Table 14 summarizes the information of the analysis of the automatically obtained LOs. *ErauzOnt* gathered 371 LOs, 140 definitions, 2 problem statements, and 229

combined LOs, i.e., LOs that comprise LOs of different kinds. Although the DR grammar also identified fragments that could be part of principle statements, these were elements of other kinds of LOs, either combined or not.

The overall achieved precision was 86.79%, i.e., 322 of the 371 LOs were considered usable for this course or any course that might be developed in the future. Problem statements obtained 100% precision, while definitions got 86.43% and combined LOs 86.90%. Considering these results, the pattern-based approach used by *ErauzOnt* to gather LOs from electronic textbooks prove to be accurate, useful and language independent.

5.5. - SUMMARY

In this chapter the process of adding a new language to *ErauzOnt* has been presented. Besides, English support extension has been described and evaluated showing similar results to experiments made for Basque language. As the evaluation process needs a lot of manual efforts, a tool for helping in the process has been described.

CHAPTER 6 - CONCLUSIONS AND FUTURE WORK

In this dissertation an introduction to a framework for semi-automatic building of the Domain Module from electronic textbooks using Ontologies, Natural Language Processing (NLP) techniques and heuristic reasoning called *Dom-Sortze* has been presented. Later on, a tool of the framework called *ErauzOnt* is described. This tool was firstly used for the extraction of LOs from textbooks in Basque. In this work *ErauzOnt* has been extended to support English, and it has been tested over the Principles of Object-Oriented Programming textbook, used in the Object-Oriented Programming subject, to evaluate its performance.

ErauzOnt was developed with the aim of being domain-independent and scalable, i.e., easy to enhance to support new languages. Improving *ErauzOnt* to enable the acquisition of LOs from textbooks in English was a task involving the search and adaptation of a new NLP tool (in this case *FreeLing*) that support English, and adapting some code from the framework itself.

In addition, the evaluation of this framework for English has been presented. In the evaluation, both the DR grammar that facilitated the identification of DR fragments and the generated LOs were evaluated. Furthermore as performing an evaluation of the framework needs a lot “manual” work, it has been developed and presented a tool for reducing the amount of this kind of work.

The analysis of the results proved that the DR grammar is an appropriate means to identify the fragments of the document that may compose an appropriate Learning Object.

The results show DR grammar achieving about 80% of accuracy, and LOs identification achieving more than 70% of accuracy.

The framework had already been tested over textbooks in the Basque language, covering different areas of the Nature Sciences, for secondary education students. The results of the experiment with a textbook written in English were quite similar to the previous experiments, so it might be deduced that *ErauzOnt* is neither tight to a particular language nor a concrete domain.

Further work on *ErauzOnt* comprises improving the treatment of images in the LO generation. Although *ErauzOnt* is currently able to process images in the electronic document, it only considers their position in the text, unaware of where the image is referenced and, therefore, useful. Hence, the treatment of the images must be improved so that they can be combined with the fragments of the document that reference them to get more accurate LOs.

Machine Learning methods will be used to infer new rules that might improve the identification of the LOs in the electronic textbooks, this will allow inferring new rules from previously analyzed textbooks and therefore, improving the results of *ErauzOnt*

The construction of multilingual Domain Modules is also being addressed. The Learning Domain Ontology supports the multilingual representation of the domain topics, and machine translation might be used to get approximate translations of the gathered LOs that would be looked for either on the Learning Object Repository or different resources.

BIBLIOGRAPHY

- Adcock, G., Ip, A., & Mason, J. (2000). Modelling Information to Support Value-Adding: EdNA Online. *WebNet Journal: Internet Technologies, Applications & Issues*, 2(3).
- Aduriz, I., Aldezabal, I., Alegria, I., Artola, X., Ezeiza, N., & Urizar, R. (1996). EUSLEM: A Lemmatiser / Tagger for Basque. *Proceedings of the 7th EURALEX International Congress on Lexicography, EURALEX 1996*, Vol. 1, pp. 17–26.
- Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Pasca, M., & Soroa, A. (2009). A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. *Proceedings of NAACL-HLT 09*.
- Agirre, E., & Soroa, A. (2009). Personalizing PageRank for Word Sense Disambiguation. *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2009*, pp. 33–41. The Association for Computer Linguistics.
- Aldabe, I. (2011). *Automatic Exercise Generation Based on Corpora and Natural Language Processing Techniques*. University of the Basque Country (UPV/EHU).
- Alegria, I., Gurrutxaga, A., Lizaso, P., Saralegi, X., Ugartetxea, S., & Urizar, R. (2004). An Xml-Based Term Extraction Tool for Basque. *Proceedings of the 4rd International Conference on Language Resources and Evaluation (LREC 2004)*.
- Anderson, J. R. (1988). The Expert Module. *Foundations of Intelligent Tutoring Systems*, pp. 21–54. Lawrence Erlbaum Associates, Inc.
- Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences*, 4(2).

- Arellano, C., Rueda, U., Niebla, I., Larrañaga, M., Lasa, A. A., & Elorriaga, J. A. (2006). Elkar-CM: a Multilingual Collaborative Concept Map Editor. *Proceedings of the 6th IEEE International Conference on Advanced Learning Technologies, ICALT 2006*, pp. 482–483. IEEE Computer Society.
- Arroyo, I., Schapira, A., & Woolf, B. P. (2001). Authoring and Sharing Word Problems with AWE. *Proceedings of the 10th International Conference on Artificial Intelligence in Education AIED-2001*, pp. 527–529. IOS Press.
- Atserias, J., Casas, B., Comelles, E., González, M., Padró, L., & Padró, M. (2002). FreeLing 1.3: Syntactic and semantic services in an open-source NLP library. *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*. Retrieved from <http://www.lsi.upc.edu/~nlp/freeling>
- Brooks, C., Cooke, J., & Vassileva, J. (2003). Versioning of Learning Objects. *Proceedings of the 3rd IEEE International Conference on Advanced Learning Technologies, ICALT 2003*, pp. 296–297. IEEE Computer Society.
- Brusilovsky, P., Sosnovsky, S. A., Yudelson, M., & Chavan, G. (2003). Interactive Authoring Support for Adaptive Educational Systems. *Shaping the Future of Learning through Intelligent Technologies, Proceedings of the 11th International Conference on Artificial Intelligence in Education, AIED 2003*, Vol. 97, pp. 97–103. IOS Press.
- Cafolla, R. (2006). Project Merlot: Bringing Peer Review to Web-based Educational Resources. *Journal of Technology and Teacher Education*, 14(2).

- Cardinaels, K., Meire, M., & Duval, E. (2005). Automating Metadata Generation: the Simple Indexing Interface. *Proceedings of the 14th International Conference on World Wide Web, WWW 2005*. ACM.
- Casey, J., & McAlpine, M. (2003). Writing and Using Reusable Educational Material - A Beginner's guide.
- Coffey, J. W., Eskridge, T. C., & Sanchez, D. P. (2004). A Case Study in Knowledge Elicitation for Institutional Memory Preservation Using Concept Maps. *Concept Maps: Theory, Methodology, Technology. Proceedings of the 1st International Conference on Concept Mapping, CMC 2004*, Vol. I, pp. 151–157.
- Conde, A., Larrañaga, M., Calvo, Iñaki, Arruarte, A., & Elorriaga, J. (2012). Automating the Authoring of Learning Material in Computer Engineering Education, Vol. IN-PRESS. Presented at the Frontiers In Education, FIE, Washington, Seattle.
- Corbett, A. T., Trask, H. J., Scarpinato, K. C., & Hadley, W. S. (1998). A Formative Evaluation of the PACT Algebra II Tutor: Support for Simple Hierarchical Reasoning. *Proceedings of the 4th International Conference on Intelligent Tutoring Systems, ITS 1998*, Vol. 1452, pp. 374–383. Springer.
- Chen, P.-S. D., Lambert, A. D., & Guidry, K. R. (2010). Engaging online learners: The impact of Web-based learning technology on college student engagement. *Computers & Education*, 54(4).
- Chen, W., Lu, R., Zhang, W., & Du, H. (1997). A Tool for Automatic Generation of Multimedia ICAI Systems. *Knowledge and Media in Learning Systems. Proceedings of the 8th International Conference on Artificial Intelligence in Education, AIED-1997*, Vol. 39, pp. 571–573. IOS Press.

- de Hoog, R., Barnard, Y., & Wielinga, B. J. (1999). IMAT: Re-using Multi-media Electronic Technical Documentation for Training. *Business and Work in the Information Society: New Technologies and Applications*, pp. 415–421. IOS Press.
- Duval, E., Forte, E., Cardinaels, K., Verhoeven, B., Durm, R. V., Hendrikx, K., Forte, M. W., et al. (2001). The ARIADNE Knowledge Pool System. *Communications of the ACM*, 44(5).
- Duval, E., & Hodgins, H. W. (2002). A LOM Research Agenda. *Proceedings of the 11th International Conference on World Wide Web, WWW 2002*, pp. 1–9. ACM Press.
- Duval, E., & Hodgins, H. W. (2004). Making Metadata Go Away: “Hiding Everything but the Benefits.” *Proceedings of the International Conference on Dublin Core and Metadata Applications*, pp. 29–35.
- Elorriaga, J. A., Arruarte, A., Calvo, I., Larrañaga, M., Rueda, U., & Herran, E. (2011). Collaborative Concept Mapping Activities in a Classroom Scenario. *Behaviour & Information Technology*.
doi:<http://dx.doi.org/10.1080/0144929X.2011.632649>
- Gurrutxaga, A., Saralegi, X., Ugartetxea, S., & Alegria, I. (2005). Erauzterm: euskarazko terminoak erauzteko tresna erdiautomatikoa. *IX. Jardunaldiak: Euskera zientifiko-teknikoa*. Mendebalde Kultur Alkartea.
- Hughes, T., & Ramage, D. (2007). Lexical Semantic Relatedness with Random Graph Walks. *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CONLL 2007*, pp. 581–589. Association for Computational Linguistics.

- Iruskieta, M., de Ilarraza, A. D., & Lersundi, M. (2010). Correlaciones en Euskera entre las Relaciones Retóricas y los Marcadores de Discurso. *Ways and Modes of Human Communication*. Universidas de Castilla-La Mancha.
- Kabel, S. C., de Hoog, R., Wielinga, B. J., & Anjewierden, A. (2004a). The Added Value of Task and Ontology Based Mark-Up for Information Retrieval. *Journal of the American Society for Information Science and Technology*, 55(4).
- Kabel, S. C., de Hoog, R., Wielinga, B. J., & Anjewierden, A. (2004b). Indexing Learning Objects: Vocabularies and Empirical Investigation of Consistency. *Journal of Educational Multimedia and Hypermedia*, 13(4).
- Kabel, S. C., Wielinga, B. J., & de Hoog, R. (1999). Ontologies for Indexing Technical Manuals for Instruction. *Workshop on Ontologies for Intelligent Educational Systems at 9th International Conference on Artificial Intelligence in Education, AI-ED' 99*, pp. 44–53.
- Knott, A., & Dale, R. (1994). Using Linguistic Phenomena to Motivate a Set of Coherence Relations. *Discourse Processes*, 18(1).
- Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent Tutoring Goes To School in the Big City. *International Journal on Artificial Intelligence in Education*, 8.
- Lagoze, C., & de Sompel, H. V. (2001). The open archives initiative: building a low-barrier interoperability framework. *Proceedings of the 1st ACM/IEEE Joint Conference on Digital Libraries, JCDL 2001*, pp. 54–62. ACM.
- Larrañaga, M., Calvo, I., Elorriaga, J. A., Arruarte, A., Verbert, K., & Duval, E. (2011). ErauzOnt: A Framework for Gathering Learning Objects from Electronic

- Documents. *Proceedings of the 11th IEEE International Conference on Advanced Learning Technologies, ICALT 2011*, pp. 656–658. IEEE Computer Society.
- Larrañaga, M., Conde, Á., Calvo, I., Arruarte, A., & Elorriaga, J. A. (2012). Evaluating the Automatic Extraction of Learning Objects from Electronic Textbooks Using ErauzOnt. In S. A. Cerri, W. J. Clancey, G. Papadourakis, & K. Panourgia (Eds.), *Intelligent Tutoring Systems*, Vol. 7315, pp. 655–656. Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://www.springerlink.com/index/10.1007/978-3-642-30950-2_107
- Larrañaga, M., Elorriaga, J. A., & Arruarte, A. (2008a). Building Learning Objects from Electronic Documents. *Proceedings of the 16th International Conference on Computers in Education, ICCE 2008*, pp. 141–146.
- Larrañaga, M., Elorriaga, J. A., & Arruarte, A. (2008b). A Heuristic NLP Based Approach for Getting Didactic Resources from Electronic Documents. *Times of Convergence. Technologies Across Learning Contexts, Proceedings of Third European Conference on Technology Enhanced Learning, EC-TEL 2008*, Vol. 5192, pp. 197–202. Springer.
- Larrañaga, M. (2012, October 10). *Semi-Automatic Generation of Learning Domain Modules for Technology Supported Learning Systems using Natural Language Processing Techniques and Ontologies*. E.H.U./U.P.V.
- Leidig, T. (2001). L3--Towards an Open Learning Environment. *ACM Journal of Educational Resources in Computing*, 1(1).

- Lentini, M., Nardi, D., & Simonetta, A. (1995). Automatic Generation of Tutors for Spreadsheet Applications. *Proceedings of the 7th International Conference on Artificial Intelligence in Education, AIED 1995*, pp. 59–66. AACE.
- Lentini, M., Nardi, D., & Simonetta, A. (2000). Self-instructive spreadsheets: an environment for automatic knowledge acquisition and tutor generation. *International Journal on Human-Computer Studies*, 52(5).
- Liu, B., Chin, C. W., & Ng, H. T. (2003). Mining Topic-specific Concepts and Definitions on the Web. *Proceedings of the 12th International Conference on World Wide Web, WWW 2003*, pp. 251–260. ACM Press.
doi:<http://dx.doi.org/http://dx.doi.org/10.1145/775152.775188>
- LTSC. (2001). 1484.12.1 IEEE LTSC Draft Standard for Learning Object Metadata. Retrieved from http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf
- Lu, R., Cao, C., Chen, Y., & Han, Z. (1995). On Automatic Generation of Intelligent Tutoring Systems. *Proceedings of the 7th International Conference on Artificial Intelligence in Education, AIED 1995*, pp. 67–74. AACE.
- Marcus, M. P., Marcinkiewicz, M. A., & Santorini, B. (1993). Building a large annotated corpus of English: the penn treebank. *Comput. Linguist.*, 19(2).
- McMartin, F. P. (2004). MERLOT: A Model for User Involvement in Digital Library Design and Implementation. *Journal of Digital Information*, 5(3).
- Meder, N. (2000). Didaktische ontologien. *Globalisierung und Wissensorganisation: Neue Aspekte für Wissen, Wissenschaft und Informationssysteme*, Vol. 6, pp. 401–416.

- Meire, M., Ochoa, X., & Duval, E. (2007). SAMgl: Automatic Metadata Generation v2.0. *Proceedings of the World Conference on Educational Multimedia, Hypermedia and Telecommunications 2007, ED-MEDIA 2007*, pp. 1195–1204. AACE. Retrieved from <http://www.editlib.org/p/25528>
- Mitrovic, A., & Ohlsson, S. (1999). Evaluation of a Constraint-Based Tutor for a Database Language. *International Journal on Artificial Intelligence in Education*, 10(1).
- Mitrovic, A., Suraweera, P., Martin, B., & Weerasinghe, A. (2004). DB-suite: Experiences with Three Intelligent, Web-based Database Tutors. *Journal of Interactive Learning Research*, 15(4).
- Murray, T. (1999). Authoring Intelligent Tutoring Systems: An analysis of the state of the art. *International Journal on Artificial Intelligence in Education*, 10.
- Nejdl, W., Wolf, B., Qu, C., Decker, S., Nilsson, M., Palmer, M., & Risch, T. (2002). Edutella: A P2P Networking Infrastructure Based on RDF. *Proceedings of the 11th International Conference on World Wide Web, WWW 2002*. ACM Press.
- Nkambou, R. (2010). Modeling the Domain: An Introduction to the Expert Module. *Advances in Intelligent Tutoring Systems*, Vol. 308, pp. 15–32. Springer.
- Padrón, C. L., Doderó, J., Díaz, P., & Aedo, I. (2005). The collaborative development of didactic materials. *Comput. Sci. Inf. Syst.*
- Parsad, B., & Lewis, L. (2008). *Distance Education at Degree-Granting Postsecondary Institutions: 2006--07*.
- Rivest, R. L. (1992, Abril). The MD5 Message-Digest Algorithm (RFC 1321).

- Schoonenboom, J. (2006). A Model for Determining the Size of Learning Objects. *Proceedings of the 6th IEEE International Conference on Advanced Learning Technologies, ICALT 2006*, pp. 46–50. IEEE Computer Society.
- Simon, B., Massart, D., Assche, F. V., Ternier, S., Duval, E., Brantner, S., Olmedilla, D., et al. (2005). A Simple Query Interface for Interoperable Learning Repositories. *Proceedings of the 1st Workshop on Interoperability of Web-based Educational Systems at the 14th International World Wide Web Conference, WWW 2005*, pp. 11–18. CEUR.
- Stevens, A., Collins, A., & Goldin, S. E. (1982). Misconceptions in Student's Understanding. *Intelligent Tutoring Systems*, pp. 13–24. Academic Press.
- Suthers, D. D. (2005). Collaborative Knowledge Construction through Shared Representations. *Proceedings of the 38th Hawaii International Conference on System Sciences, HICSS 2005*, Vol. 1, p. 5.1. IEEE Computer Society.
- Taboada, M. (2006). Discourse Markers as Signals (or Not) of Rethorical Relations. *Journal of Pragmatics*, 38(4).
doi:<http://dx.doi.org/10.1016/j.pragma.2005.09.010>
- Tapanainen, P. (1996). *The Constraint Grammar parser CG-2*. Publications of the University of Helsinki.
- Ternier, S., & Duval, E. (2006). Interoperability of Repositories: The Simple Query Interface in ARIADNE. *International Journal on E-Learning*, 5(1).
- Ternier, S., Massart, D., Assche, F. V., Smith, N., Simon, B., & Duval, E. (2008). A Simple Publishing Interface for Learning Object Repositories. *Proceedings of the World*

-
- Conference on Educational Multimedia, Hypermedia and Telecommunications 2008, ED-MEDIA 2008*, pp. 1840–1845.
- Ternier, S., Verbert, K., Parra, G., Vandeputte, B., Klerkx, J., Duval, E., Ordonez, V., et al. (2009). The Ariadne Infrastructure for Managing and Storing Metadata. *IEEE Internet Computing*, 13(4).
- Verbert, K. (2008, February). *An Architecture and Framework for Flexible Reuse of Learning Object Components*. Faculteit Ingenieurswetenschappen, Katholieke Universiteit Leuven.
- Verbert, K., & Duval, E. (2004). Towards a Global Component Architecture for Learning Objects: A Comparative Analysis of Learning Object Content Models. *Proceedings of the World Conference on Educational Multimedia, Hypermedia and Telecommunications 2004, ED-MEDIA 2004*, pp. 202–208. AACE.
- Verbert, K., Gašević, D., Jovanović, J., & Duval, E. (2005). Ontology-Based Learning Content Repurposing. *Proceedings of the 14th International Conference on World Wide Web, WWW 2005 - Special interest tracks and posters*, pp. 1140–1141. ACM Press.
- Verbert, K., Ochoa, X., & Duval, E. (2008). The ALOCOM Framework: Towards Scalable Content Reuse. *Journal of Digital Information*, 9(1).
- Voutilainen, A., & Tapanainen, P. (1993). Ambiguity Resolution in a Reductionistic Parser. *Proceedings of the 6th Conference on European Chapter of the Association for Computational Linguistics, EACL 1993*, pp. 394–403. Association for Computational Linguistics.

-
- Waits, T., & Lewis, L. (2003). *Distance Education at Degree-Granting Postsecondary Institutions: 2000--2001*.
- Wenger, E. (1987). Artificial Intelligence and Tutoring Systems. *Artificial Intelligence and Tutoring Systems*. Morgan Kauffmann Publishers, Inc.
- Wiley, D. A. (2000). *Learning Object Design and Sequencing Theory*. Brigham Young University.
- Wong, S, & Nguyen, D. (2010). *Principles of Object-Oriented Programming*.
- Woolf, B. P. (2008). *Building Intelligent Interactive Tutors: Student-centered Strategies for Revolutionizing E-learning*. Morgan Kauffmann Publishers, Inc.
- Woolf, B. P., Arroyo, I., Beal, C. R., & Murray, T. (2006). Gender and Cognitive Differences in Help Effectiveness During Problem Solving. *International Journal of Technology, Instruction, Cognition and Learning*, 3.
- Zouaq, A., & Nkambou, R. (2009). Evaluating the Generation of Domain Ontologies in the Knowledge Puzzle Project. *IEEE Transactions on Knowledge and Data Engineering*, 21(11).

APPENDICES

APENDIX A - PATTERNS FOR DIDACTIC RESOURCES

In this appendix the full list of patterns for Didactic Resources identification is shown.

A.1 - LIST OF DEFINITION PATTERNS

Pattern	Pattern (CG2)
{concept}+ {is are} + [determiner]	MAP:DEF (&DEF) (@ONT-TOPIC) IF (1 ("be" VBZ) LINK 1 (DT));
{concept} + {refer to satisfy} + [adverb]	MAP:DEF (&DEF) (@ONT-TOPIC) IF (1 ("refer" VBZ) LINK 1 ("<to>"));
{concept} + {is are} + {defined as being used to referred to employed to formalized as}	MAP:DEF (&DEF) ("be" VBP) IF (NEGATE -1 ("that")) (NEGATE -1 ("this")) (1 ("as" RB) LINK 1 (RB) LINK *1 (@ONT-TOPIC));
{concept} + {is are} + {called known as defined as}	MAP:DEF (&DEF) ("be" VBP) IF (NEGATE -1 ("that")) (NEGATE -1 ("this")) (1 ("call" VBN) LINK *1 (@ONT-TOPIC));
{concept} + {:}	MAP:DEF (&DEF) TARGET (@ONT-TOPIC) IF (1 (":"));
{concept}+ {,}+...{,}+...	MAP:DEF (&DEF) TARGET ("<,>") IF (NEGATE -1 (@ONT-TOPIC) LINK *-1 ("<,>")) (-1 (@ONT-TOPIC)) (*1 ("<,>") LINK *1 (VB));
{text} + {called} + {concept}	MAP:DEF (&DEF) TARGET ("call" VBN) IF (NEGATE -2 ("that")) (NEGATE -2 ("this"))(0 ("call" VBN) LINK 1 (@ONT-TOPIC));
{concept}+ {i.e.} + {defining text}	MAP:DEF (&DEF) (@ONT-TOPIC) IF (1 ("," Fc) LINK 1 ("i.e."));
{concept}+{=}{description}	MAP:DEF (&DEF) (@ONT-TOPIC) IF (1 ("=" Fz));
{text}+{,}{concept}+{,} +{text}	MAP:DEF-10(&DEF) TARGET ("<,>") IF (1 (@ONT-TOPIC) LINK 1 ("<,>"));
{what}+{is are}+ [determiner]+ {concept}	MAP:(&DEF) TARGET ("what" WP) IF (NEGATE -2 ("to")) (1 ("be" VBP) LINK 1 (@ONT-TOPIC));
{definition}+{or}+{concept}	MAP:(&DEF) TARGET ("or" CC) IF (1 (@ONT-TOPIC));
{concept}+ {{(definition)}	MAP:DEF (&DEF) TARGET (@ONT-TOPIC) IF (1 ("(" Fpa) LINK *1 (") Fpt));
{definition}+ {:}+ {concept}	MAP:DEF (&DEF) TARGET (": " Fd) IF (1 (@ONT-TOPIC));
{definition} + [of] + {concept}+ {- :}...	MAP:DEF (&DEF) TARGET ("definition" NN) IF (1 (@ONT-TOPIC) LINK 1 (":"));
{is} +{concept}+{'s{- :}... [determiner]	MAP:DEF (&DEF) TARGET ("be" VBZ) IF (1 (@ONT-TOPIC) LINK 1 (" 's" POS)

] definition}	LINK 1 (DT) LINK 1 ("definition" NN) LINK 1 ("-");
{is}+ {concept}+{s{-:}}+... [determiner] definition}	MAP:DEF (&DEF) TARGET ("be" VBZ) IF (1 (@ONT-TOPIC) LINK 1 ("s" POS) LINK 1 ("definition" NN) LINK 1 (":" Fd));
{To}+{Express Describe Define} {what}+{is are}+ [determiner]+ {concept}	MAP:DEF (&DEF) TARGET ("to" NNP) IF (1 ("express" VB) LINK 1 ("what" WP) LINK 1 ("be" VBZ) LINK 1 (DT) LINK 1 (@ONT-TOPIC));
{concept}+{s{-:}}+{definition}	MAP:DEF (&DEF) TARGET (@ONT-TOPIC) IF (1 (":"));

Table 15 - Table of definition patterns

A.2 - List of problems patterns

Pattern	Pattern (CG2)
{? . ;} {Answer} [determiner] [next following] {question}	MAP:ARIK (&ARIK) TARGET ("answer" VBP) IF (1 (DT) LINK 1 ("question" NN));
{Exercise}	MAP:ARIK-3 (&ARIK) TARGET ("exercise") IF (*1 (@ONT-TOPIC));
{Problem} {:}	MAP:ARIK-4 (&ARIK) TARGET ("problem" NN) IF (1 (":" Fd));

Table 16 - Table of problems patterns

A.3. - List of example patterns

Pattern	Pattern (CG2)
{example instance case specimen} +[of] +{concept}	MAP:ADIB (&ADIB) TARGET ("example" NN) IF (1 (@ONT-TOPIC));
{for instance e.g. for example as an example}+ [,] +{determiner} +{concept}	MAP:ADIB (&ADIB) TARGET ("for_instance" RB) IF (1 (@ONT-TOPIC));
{concept}+ {illustrates demonstrates shows exemplifies}...	MAP:ADIB (&ADIB) TARGET (@ONT-TOPIC) IF (1 ("illustrate" VBZ));
{concept}+ {is are}+ {called}	MAP:DEF (&DEF) ("be" VBP) IF (NEGATE -1 ("that")) (NEGATE -1 ("this")) (1 ("call" VBN) LINK *1 (@ONT-TOPIC));
{concept}+ {is are}+ [adverb] +{illustrated by demonstrated by shown by}	MAP:ADIB (&ADIB) TARGET (@ONT-TOPIC) IF (1 ("be" VBZ) LINK 1 (RB) LINK 1 ("illustrate" VBN) LINK 1 ("by" RB));
...{Example}+ {-:}+{example}	MAP:ADIB (&ADIB) TARGET (@ONT-TOPIC) IF (1 ("example" NN) LINK 1 (":" Fd));
{concept}+ {is/ some of}+ {concept} one of are +[determiner]	MAP:ADIB (&ADIB) TARGET (@ONT-TOPIC) IF (1 ("be" VBZ) LINK 1 ("one" DT) LINK 1 ("of" IN) LINK 1 (@ONT-TOPIC));
{Some}+ {concept} +{:}+ {list of topics}	MAP:ADIB (&ADIB) TARGET ("some" DT) IF (1 (@ONT-TOPIC) LINK 1 (":" Fd));
{concept}+ {is are has been have	MAP:ADIB (&ADIB) TARGET (@ONT-TOPIC) IF (1 ("be" VBP) LINK 1 ("mention"

been can be} +{mentioned among} [determiner] +{concept}	VBN) LINK 1 ("among" IN) LINK 1 (@ONT-TOPIC));
---	--

Table 17 - Table of example patterns

A.4. - List of Principle-Statements patterns

Pattern	Pattern (CG2)
{concept}+ {is are}+ {based on} {description}	MAP (&OD) TARGET (@ONT-TOPIC) IF (1* ("be" VBZ) LINK 1 ("base" VBN) LINK 1("on"));
{Description} +{:} +[determiner] {consequence of that is consequences of that are}	MAP (&OD) TARGET (":" Fd) IF (1* ("consequence" NNS) LINK 1 ("of") LINK 1("that") LINK 1 ("be" VBP));
{is are} +{due to caused by}	MAP (&OD) TARGET (@ONT-TOPIC) IF (1 ("be" VBP) LINK 1 ("cause" VBN) LINK 1 ("by") LINK 1* (@ONT- TOPIC));
{is are} +{initiated by}	MAP (&OD) TARGET (@ONT-TOPIC) IF (1 ("be" VBP) LINK 1 ("initiate") LINK 1("by"));
{concept} +{due to caused by because of because}	MAP (&OD) TARGET (@ONT-TOPIC) IF (1 ("because")) ;
{has have} +[determiner] +{consequence consequences}	MAP (&OD) TARGET (@ONT-TOPIC) IF (1* ("have" VBZ) LINK 1 (DT)) ;
{produce producing generate generating cause causing induce inducing}	MAP (&OD) TARGET ("produce" VBG) IF (*1 (@ONT-TOPIC));
{why how} ... {?}	MAP (&OD) TARGET ("why" WRB) IF (*1 (@ONT-TOPIC) LINK 1* ("?" Fit));
{happen happens can happen occur occurs can occur}...	MAP (&OD) TARGET (@ONT-TOPIC) IF (1 ("happen" VBP));
{Principle}	MAP (&OD) TARGET ("principle" NN) IF (*1 (@ONT-TOPIC));

Table 18 - Table of principle-statements patterns

A.5. - List of Theorems patterns

Pattern	Pattern (CG2)
{Theory Theories Theorem Theorems}	MAP:TEOR-1 (&TEOR) TARGET ("theory" NN) IF (*1 (@ONT-TOPIC));

Table 19 - Table of theorems patterns