

A Software System for Viewing and Querying Automatically Generated Topic Maps in the E-learning Domain

Liana Stanescu, Gabriel Mihai, Dumitru Burdescu, Marius Brezovan, and Cosmin Stoica Spahiu
University of Craiova, Faculty of Automation, Computers and Electronics, Romania
E-mail: {Stanescu_Liana, Burdescu_Dumitru, Brezovan.Marius, Stoica.Cosmin}@software.ucv.ro

Keywords: e-learning, topic maps, relational database, tolog, topic maps query

Received: February 8, 2010

Topic Maps represent a recent technology for structuring and retrieval of information, based on principles used in traditional indexes and thesauri, drawing inspiration from semantic networks. In the e-learning domain Topic Maps have a greater importance as a content management technology, but also because they fill in the gap between information and knowledge. Because many e-learning systems use a relational database for storing the learning content, this paper presents an original algorithm for automated building of a Topic Map starting from a relational database. The process is illustrated on a database used in TESYS e-learning system. The paper also presents a tool with two main functions: the Topic Map graphical view that allows learner navigation for studying the topics, and associations between them and Topic Map querying using tolog that facilitates the establishing of search criteria for learning resources filtering.

Povzetek: Predstavljena je platforma za poučevanje s prilagajanjem vsakemu učencu posebej.

1 Introduction

In the last few years, web-based e-learning has gained an increased popularity in many domains: technical, economic, medical, etc. It is well established that the e-learning platforms should offer learners interactive and flexible interfaces for access to learning resources and should also adapt easily to one's individual needs [1][2].

At the University of Craiova an e-learning system called TESYS has been created, which is used in distance learning for certain domains (economic), but also in the hybrid learning in medical and engineering domains to complete face-to-face lectures [8].

Over the few years of use there has been a tendency of passing from "course-centric" learning systems to "subject-centric" learning systems. "Course-centric" learning systems are the traditional ones which assume sequential run-over learning resources along with the lecture time schedules. In this way, learners acquire knowledge step by step in the order established by teacher. In this case, less motivated students often lose enthusiasm in the middle of the course, having difficulties in knowledge understanding [16].

This is why an e-learning system should offer students the possibility of designing their learning method in order to stay motivated. This goal can be achieved with the "subject-centric" learning systems based on Topic Maps (TM). Thus, the learners can choose their subjects and Topic Maps not only permit subjects to be visualized but also to relate with each other [16].

The paper presents an original algorithm for automated representation of a relational database with a

Topic Map. This aspect is useful in the e-learning domain because many e-learning systems are based on a relational database. For example, the Moodle database has around 200 tables. The information about courses and their organization into categories is stored in the following tables: course, course_categories, course_display, course_meta, course_request. The information about activities and their arrangement within courses is stored in the next tables: modules, course_allowed_modules, course_modules and course_sections. The database structure is defined, edited and upgraded using the XMLDB system [3]. Also, the Blackboard Learning System uses a relational database for storing necessary data [14].

The proposed algorithm will be illustrated on a database used in TESYS e-learning system [8].

Human intervention can be necessary in the e-learning domain during the TM generation process. As a result, the paper presents an improvement of the automated algorithm. It assumes a configuration file that allows user to specify the part of the database that will be reflected in the content of the Topic Map. This way the size of the Topic Map is substantially reduced because it will contain only the selected items. Also the semantic content of the generated TM is improved. The generation process becomes semi-automatic because the human intervention is needed for obtaining this file.

The paper also presents a Topic Map graphical view that allows learner navigation for studying the topics and associations that shape in fact the relationships between items in the database. Associations provide the context

information necessary to better understand a topic. Associations simulate the way humans think and hence are essential for knowledge modelling. This tool also allows the learning resources filtering by establishing the search criteria in Topic Map based on tolog. To achieve this goal, the paper proposes an original graphical interface that allows the user to build interactive queries on topic map even without knowing the tolog syntax. These queries are automatically generated and sent to the tolog engine for processing.

2 Related work

Topic Maps represent a recent technology for the structuring and retrieval of information, based on principles used in traditional indexes and thesauri, inspired from semantic networks. Topic Maps work with topics, the relationships between topics and links to resources about those topics. Topic Maps are independent of the resources they describe and are used in many different situations. As a result, the Topic Maps can be used in information access on the Web, in reference book publishing, or in the integration of corporate information repositories [7, 9, 10]. Most applications of Topic Maps fall into four broad categories: enterprise information integration, knowledge management, e-learning, and Web publishing. In the e-learning domain, Topic Maps play an important role as a content management technology, but also due to the fact that they fill in the gap between information and knowledge [19].

There are three possibilities for the creation of TM: manually, automatically or a combination of them. In Topic Map building there are two phases: the design phase and the authoring one. Classes are created in the design phase and instances in the authoring phase [9].

Manual Topic Map population may require lots of resources: time, money and human interventions. As a result, it is considered as a weak point in process of Topic Maps self-population. The available resources that can act as a source of input to auto-population are: ontology, relational or object-oriented database, metadata about resources, index glossary, thesaurus, data dictionary, document structures and link structures or unstructured documents [9]. In related literature the authors explain only the general principles of mapping these structures to the Topic Maps model [7, 9, 19], without any specific algorithm.

A transformation of ontology into a topic map is straightforward because ontologies use the same concepts. There are several proposals as regards how to map RDF and DAML+OIL ontologies to XTM topic maps on the model level [9]. For example in [21] the authors present a topic map-driven web portal of conference papers. The paper also discusses the tools for automatically creating topic maps, with particular emphasis on how the synergies between topic maps and RDF can be exploited in the process of auto-generating topic maps from structured and semi-structured source data.

In the case of a relational database consisting of tables, columns, rows, keys and foreign keys, the mapping can be done using the following principles [9]:

- Table -> topic class;
- Row -> topic instance of corresponding class;
- Column -> name or occurrence;
- Key -> topic id;
- Foreign key -> association

As we have already mentioned, one of the most important domain of successfully using the topic map concept is the e-learning. In this context we can mention papers that present interesting and modern modalities of using Topic Maps in e-learning. For example, TM4L is an e-learning environment providing editing and browsing support for developing and using Topic Maps-based digital course libraries. The TM4L functionality is enhanced by an interactive graphical user interface that combines a hierarchical layout with an animated view, coupled with context sensitive features [4, 5]. In Norway, school students are encouraged to create Topic Maps to record what they have learned, and the National School Curriculum itself now has its definitive expression in the form of a Topic Map [19].

Another author proposed Topic Map ontology focusing on both students and teachers as active producers of learning resources. Topic maps customize the interface, and the interface should also provide possibilities for online students to share learning resources like “on campus” students do [6].

In [15] we proposed original ways of using Topic Maps in medical e-learning. The TM is used for visualizing a thesaurus containing medical terms. The paper presents also the way in which the TM can be used for semantic querying a multimedia database with medical information and images.

There are some available TM authoring tools, but they are used by experts in knowledge representation, not by end-users (Ontopia Knowledge Suite [12], Mondeca Intelligent Topic Manager [13]). Also, there are few specialized education-oriented TM tools that can be used to facilitate the creation, maintenance, search, and visualization of Topic Maps-based learning resources.

Along with the Topic Map graphical visualization searching the information that satisfies a number of criteria is also important. A solution for querying a Topic Map is represented by the tolog language.

Tolog is a language for querying Topic Maps, inspired by Datalog (a subset of Prolog) and SQL. With tolog users can ask for all topics of a particular type, the names of all topics of a particular type, all topics used as association role types, all associations with more than two roles, and so on [17].

Tolog is a logic-based query language, which means that the basic operation consists of asking tolog in which cases a certain assertion holds true, and tolog will return all the sets of values that make the assertion true.

As a result, the paper presents a software system with the following contributions:

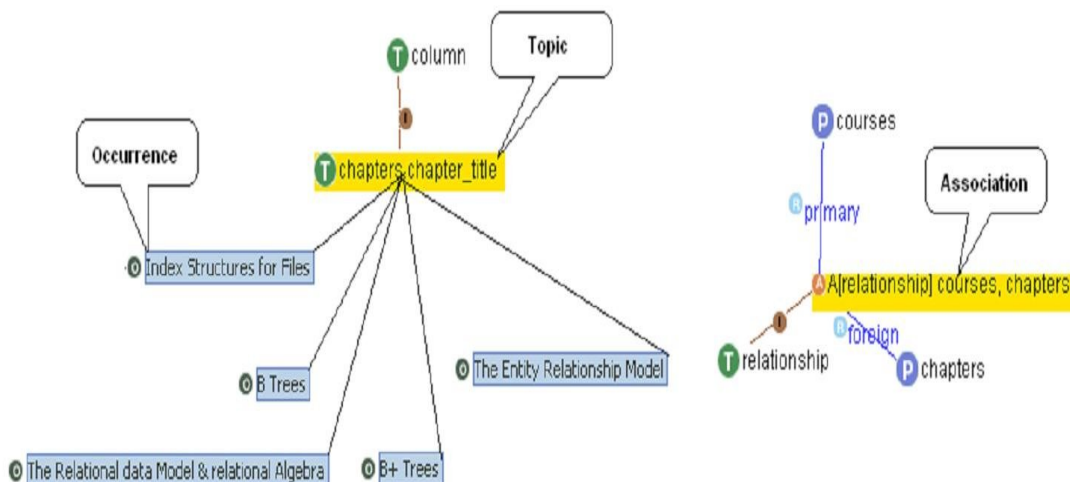


Figure 1: The basic concepts in a topic map.

1. An original algorithm of mapping any relational database to a topic map. The algorithm steps are exemplified on TESYS database used in the e-learning domain.
2. An improvement of this automated algorithm based on a configuration file that allows user to specify the database parts that will be reflected in the TM content.
3. A graphical interface for TM querying using tolog language. Users do not need to know tolog syntax and this aspect is important in the e-learning domain.

3 Topic Maps basic concepts

Topic Maps represent an ISO standard that provides concepts for describing knowledge and linking it to information resources. Topic Maps allow organizing and representing complex structures with the next basic concepts [7, 9, 10]: topics, occurrences of a topic, associations that connect related topics, topic classes, occurrence classes and association classes that help to distinguish different kind of topics, occurrences and associations, respectively (figure 1).

A topic can be defined as a syntactic construct that represents a subject inside a computer system. Through the agency of it, the real-world concept becomes a machine interpretable object. A topic can be an instance of zero, one or more classes and the classes itself are also topics. A topic has three characteristics:

- Names (base names)
- Occurrences
- Playing roles in associations

Occurrences link information resources to topics. It is important to notice that topics and information resources are placed on different layers and the users may navigate at the abstract layer that is the topic layer rather than within data. In fact, occurrences bind real resources to topics that are abstract concepts. An occurrence is either a resource reference or a resource data. A resource

reference links the relevant resource to the topic using XLink/XPointer URI. A resource data occurrence assigns a value to the topic [7, 9, 10].

The relationships between concepts are possible in Topic Maps through topic associations. The associations provide the context to better understand topics. They simulate the human thinking and so are important for knowledge modelling.

Topic Maps can improve navigation and information retrieval in large and complex information pools by adding semantics to these resources.

4 The system architecture

The system architecture is presented in figure 2. TMGenerator is the module that generates the content of the Topic Map using the information from the database

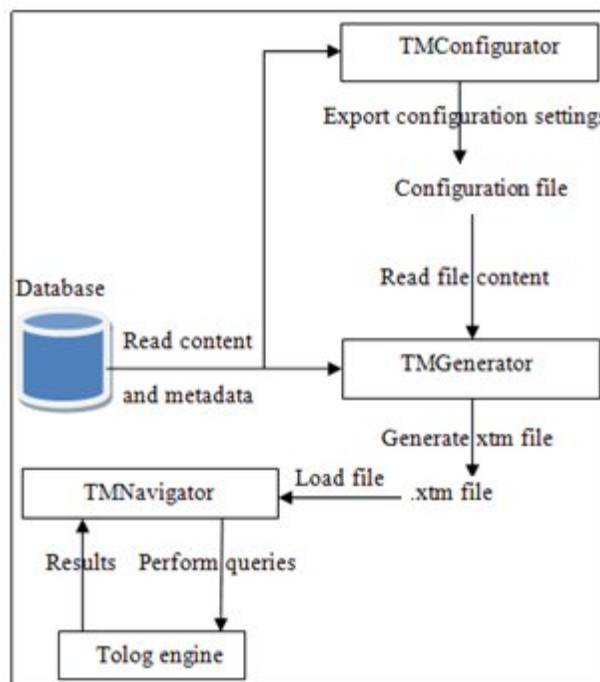


Figure 2: The system architecture.

and from a configuration file (when it is available). The content of a generated Topic Map is saved by the TMGenerator as an .xtn file.

TMConfigurator is the module used for making configurations. It uses the structure of the database and allows the user to select the tables and the columns, but also to specify the topics, the topics type, the associations and the association types that will be generated. The settings are exported in an .xml file that will be used by the TMGenerator when generating the Topic Map. The user has the possibility to represent in the TM only the necessary information from the database, thus reducing the size of the Topic Map.

TMNavigator is the name of a graphical application used for exploring the content of a Topic Map and for running tolog queries. These queries are analysed by the tolog engine and the results are sent back.

The functions of these modules will be explained in detail in the next sections.

5 TESYS database structure

Figure 3 illustrates a part of the relational database used by the e-learning system called TESYS [8]. This database will be used later to better explain the Topic Map automated building and also its graphical view.

The table named Courses stores data about electronic courses, each course being equivalent to a unit of curriculum or an academic subject in traditional learning. Usually, a course contains many chapters, and each chapter contains a number of topics that represent in fact learning objects. Each topic represents one unit of knowledge, being the smallest component. The learning object can be a piece of text, a video clip, a picture or a voiced text.

In this database structure the relationships between topics studied at the same course or different courses are important. If a topic uses knowledge presented in other topics, these topics must be linked. As a result, on Topics table a m:m recursive relationship is defined. This special relationship is implemented with Topic_connection table.

6 The algorithm for building the Topic Map starting from a relational database

The Topic Map generation process is a general one and it is realized by the module TMGenerator. This module uses the settings that were specified for accessing the database and for retrieving the content and the metadata (table's name, column's name, etc.). TMGenerator was designed to automatically generate the content of the Topic Map taking into account the xtn syntax [11]. When the generation process is completed the content is exported as an .xtn file that can be imported by the module TMNavigator for a graphical representation of the Topic Map. Initially, the TM generation process builds a topic type array: "database", "table", "row", "column", "relationship", "part", "whole", "primary", "foreign" that will be used as basic types for topics and associations.

From database meta-data the algorithm uses the following elements:

- The array with table names
- The relationships between tables

The general algorithm implemented in TMGenerator for generating a Topic Map stores internally (in memory) the content using the following data structures:

- *topicMap* – a structure that stores the content of a

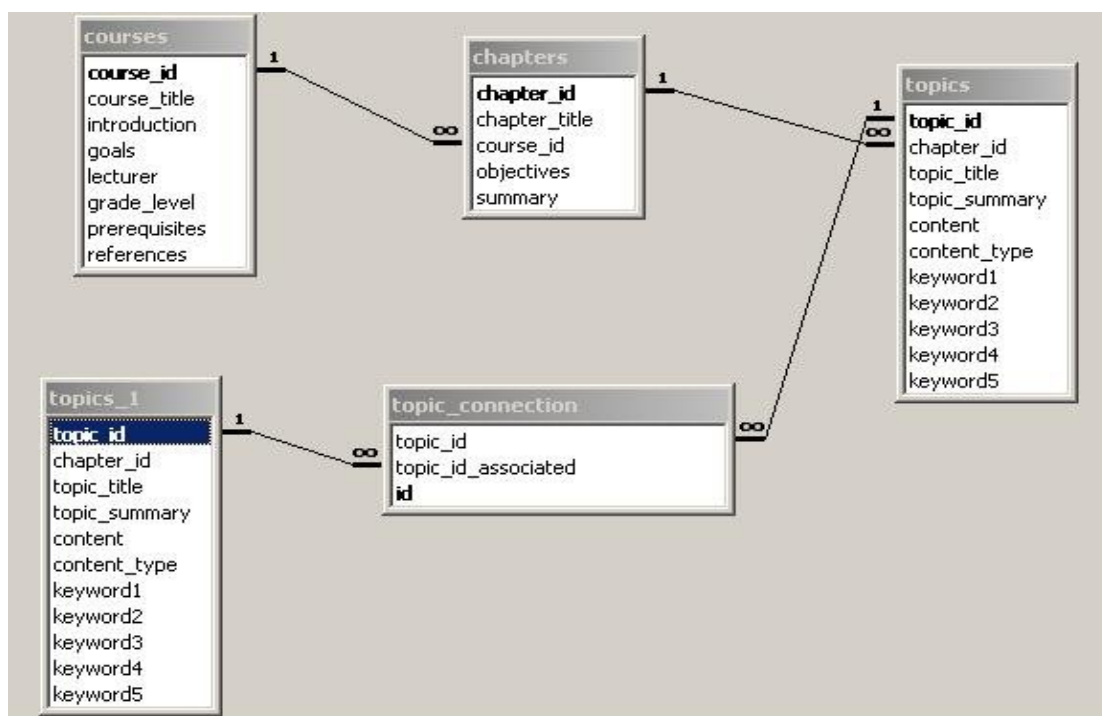


Figure 3: TESYS database structure.

Topic Map using 3 lists: a list of topics, a list of associations and a list of occurrences

- *topic* – a structure containing the information (id, instanceOf, baseName) about a topic item

- *association* – a structure containing the information (id, instanceOf, members) about an association item

- *occurrence* – a structure containing the information (instanceOf, resourceData) about an occurrence item

The algorithm uses the following procedures:

- GenerateTopic – used to generate a *topicMap* structure representing the content of a Topic Map

- GenerateTopic – used to generate a *topic* structure

- GenerateOccurrence – used to generate an *occurrence* structure

- GenerateAssociationForTables – used to generate an *association* structure representing the relationship between two tables

- GenerateAssociation – used to generate an *association* structure

The content of these procedures is presented bellow.

Procedure GenerateTopicMap()

- a) initialize topicMap;
 - b) for * each topic type in topicTypes
 - topic ← GenerateTopic(id,instanceOf, baseName);
 - topicMap.Topics.Add(topic);
 - c) databaseTopic ← GenerateTopic(dbName, "database", dbName);
 - d) topicMap.Topics.Add(databaseTopic);
 - e) for * each table in database
 - tableTopic ← GenerateTopic(tblName, "table", tblName);
 - topicMap.Topics.Add(tableTopic);
 - f) topics ← generate topics for table columns and rows
 - g) topicMap.Topics.Add(topics);
 - h) occurrences ← generate occurrences
 - i) topicMap.Occurrences.Add(occurrences);
 - j) for * relationship between tables
 - association ← GenerateAssociationForTables(params);
 - topicMap.Association.Add(association);
 - k) association ← GenerateAssociation(params);
 - l) topicMap.Association.Add(association);
 - m) for *each table and his rows
 - association ← GenerateAssociation(params);
 - topicMap.Association.Add(association);
 - n) for * each record involved in a 1:m relationship
 - association ← GenerateAssociation(params);
 - topicMap.Association.Add(association);
- return topicMap;

Procedure GenerateTopic(topicId,instanceOf,baseName)

- a) initialize topic;

b) topic.Id ← -topicId;

c) topic.InstanceOf ← -instanceOf;

d) topic.BaseName ← -baseName;

e) return topic;

Procedure GenerateOccurrence(instanceOf,resourceData)

a) initialize occurrence;

b) occurrence.InstanceOf ← -instanceOf;

c) occurrence.ResourceData ← -resourceData;

d) return occurrence;

Procedure GenerateAssociationForTables(associationId,instanceOf,primarySpecificRole,primaryTopicReference,foreignSpecificRole,foreignTopicReference)

a) initialize tblAssociation;

b) tblAssociation.Id ← -associationId;

c) tblAssociation.PrimaryMember.SpecificRole ← -primarySpecificRole;

d) tblAssociation.PrimaryMember.TopicReference ← -primaryTopicReference;

e) tblAssociation.ForeignMember.SpecificRole ← -foreignSpecificRole;

f) tblAssociation.ForeignMember.TopicReference ← -foreignTopicReference;

g) return tblAssociation;

Procedure GenerateAssociation(associationId,instanceOf, members)

1) initialize association;

2) association.Id ← -associationId;

3) association.InstanceOf ← -instanceOf;

4) for * each member in members

5) association.Members.Add(member);

6) end;

7) return association;

The Topic Map generation process starting from the ELearning database is detailed bellow and has the following steps:

1. Topic generation for database and tables

The database is represented by the algorithm as a topic that is an instance of the topic database. This topic has an id that contains the database name.

Example: For ELearning database is used the next procedure:

GenerateTopic("ELearning","database","ELearning")

In the same way, the algorithm creates a topic for each table in the database. The topic id is given by table's name, because this is unique in the database. For the database in figure 3 the topics for tables are generated: courses, chapters, topics, topics1, topic_connection.

Example: For table Courses is used the procedure: GenerateTopic("courses","table","courses")

2. Topic generation for table columns and records

The algorithm will generate a topic for each column in a table. The topic id is given by the next syntax: TableName.ColumnName. This is our choice because each topic must have a unique id.

Example: The procedure used for column `course_title` in table `Courses` is: `GenerateTopic("courses.course_title","column","course_title")`.

For each table record the algorithm generates a topic that has the following format for the id: `TableName.Row.PrimaryKeyValue`. The record content is considered as topic occurrence.

Example: For the record with primary key value 3 in table `Chapter` is used the procedure: `GenerateTopic("chapters.Row.3","row","chapters.Row.3")`

One of the columns in table `Chapters` is `chapter_title`. For this column content an occurrence is created using the procedure:

`GenerateOccurrence("chapters.chapter_title","B Trees")`.

3. Associations generation process

3.1 Associations corresponding to relationships between tables

For each relationship in the database it is generated an association of type "relationship". The id of each association is based on the tables' names, the primary key and the foreign key. This development mode takes into consideration to offer information about the database structure for facilitating the learning process.

Example: For the relationship 1: m between tables `Courses` and `Chapters` an association with the next id is generated: `"courses.course_id-chapters.course_id"`. This association is an instance of the topic relationship. In this association, the table `courses` contains the primary key and plays the role "primary" and the table `chapters` containing the foreign key plays the role "foreign". This association is generated by the procedure:

`GenerateAssociationForTables("courses.course_id-chapters.course_id","relationship","primary","courses","foreign","chapters")`.

3.2 Association between database and tables

The association between database and its tables is of the "part-whole" type. The topic representing the database plays the role "whole" and every topic representing a table plays the role "part".

Example: The association between the topic representing the database `ELearning` and topics representing the tables (`courses`, `chapters`, `topics`, `topics1`, `topic_connection`) is generated using the procedure:

`GenerateAssociation ("Database: ELearning.Tables", "part-whole", members)`

In this syntax, "members" is a list with member items, each member containing two fields: the role played in the association and the id of the corresponding topic.

3.3 Associations between table and records

The fact that a table contains records is represented by an association of "part-whole" type between table and its records.

Example: For table `courses` an association is generated using the next procedure:

`GenerateAssociation("Table:courses.Rows" ,"part-whole", members)`.

In this association the topic representing the table `Courses` plays the role "whole" and every topic representing a record plays the role "part".

3.4 Associations between records involved in a 1:m relationship

This association is of the "related-to" type. In order to be generated, for every value of the primary key, the records that contain the same value in the foreign key column must be founded. As a result, this association is established between the topics already generated for these records.

Example: Tables `Courses` and `Chapters` are defined by a 1:m relationship. The course entitled "Databases" contains 3 chapters stored in the table `Chapters`. This fact is represented by an association of the "related-to" type between the topic representing the corresponding record in the table `courses` and the topics representing connected records from the table `chapters`. Every topic plays the role "related".

The generation process is a generic one and it generates topics for each row, for each column value and a big number of associations. Other details of this process can be found in [20].

7 Improving the topic map building process

The process described in section 6 is a general one. It can be applied to mapping any relational database to a Topic Map. After a number of experiments we have decided to improve this process in order to have a better representation of the semantic content and to export only the necessary information from the database, thus reducing the size of the Topic Map. In the e-learning domain it is very important for the user to understand without a big effort the Topic Map content, otherwise the learner can be disoriented, especially when the Topic Map size is large.

By human intervention it is possible to generate a Topic Map that represents better the knowledge from the database. We have decided to complete the general algorithm by using configuration settings. For this purpose we have created an application that has a graphical interface called `TMConfigurator`. It retrieves the structure of a database and allows the user to select the tables and the columns, but also to specify the topics, the topics type, the associations and the association' types that will be generated. The settings are exported in an .xml file that will be used by the `TMGenerator` when generating the Topic Map. For the structure of the `TESYS` database used in this paper, an example of content from the configuration file is presented in figure 4.

Because the human intervention is needed for obtaining this file the generation process becomes semi-automatic.

8 Topic Map Graphical View

Topic map content is explored using the graphical interface with multiple views of the `TMNavigator`


```
<? xml version="1.0" encoding="UTF-8"?>
<configuration>
  <topics>
    <topic table_column="Courses.Course"
          topic_type="course"/>
    <topic table_column="subchapters.Subchapter"
          topic_type="subchapter">
      <occurrence table_column="LearningObjects.Name"
                  type="resource"/>
    </topic>
  </topics>
  <associations>
    <association type="has_chapters">
      <member primary_table_column="Courses.Course"
              role_type="whole"/>
      <member foreign_table_column="Chapters.Chapter"
              role_type="part"/>
    </association>
    <association type="has_subchapters">
      <member primary_table_column="Chapters.Chapter"
              role_type="whole"/>
      <member foreign_table_column="subchapters.Subchapter"
              role_type="part"/>
    </association>
  </associations>
</configuration>
```

Figure 4: The configuration file.

module. TMNavigator has a menu with two options (figure 5): File and Search. Using the File option the user can load a generated .xtm file (representing the content of a Topic Map) and then explore the content. The Search option can be used by the user to run tolog queries. This way the user can find the information he is looking for.

The viewing interface for Topic Maps is organized in two windows: the left window displays a list of all topics, topic types, associations, association types, occurrence types and member types. As a topic type we have: column, database, row, table, etc.

In Topic Map there are multiple association types: “part-whole” (defined between the topics that represent database and tables or between topics representing a table and its records), “relationship” (defined between

the topics representing tables implied in a 1:m relationship), “related-to” (defined between the topics representing the table records bound with a 1:m relationship).

The learner can select an item from the list displayed in the left window and he will see in the right window the item’s graphical representation. Topic map viewing tool intends to offer learner much information about the selected item. Unlike TM4L, our viewing tool displays for each topic involved in an association its occurrence content also. Example: for the topic that represents a record in Courses table, the learner will see information like: lecturer, grade_level, introduction, etc. Topic content can be visualized separately by selecting it in the left window.

Another original element in this graphical window is that the learner can see directly the record content involved in 1:m relationship implemented in Topic Map by “related-to” association. Beside these associations viewing that offer better understanding to the topic, the learner can go directly to study the associated topic.

Figure 5 presents details of a “related-to” type association. Figure 6 contains a “part-whole” type association.

The users can use the Topic Map as a navigation tool with which they can navigate through Topic Map depending on their subject of interest and this is a big advantage. They don’t have to be familiar with the logic of the database, they will learn about the semantic context, in which a collection and its single items are embedded and they may find useful items that they would not have expected to find.

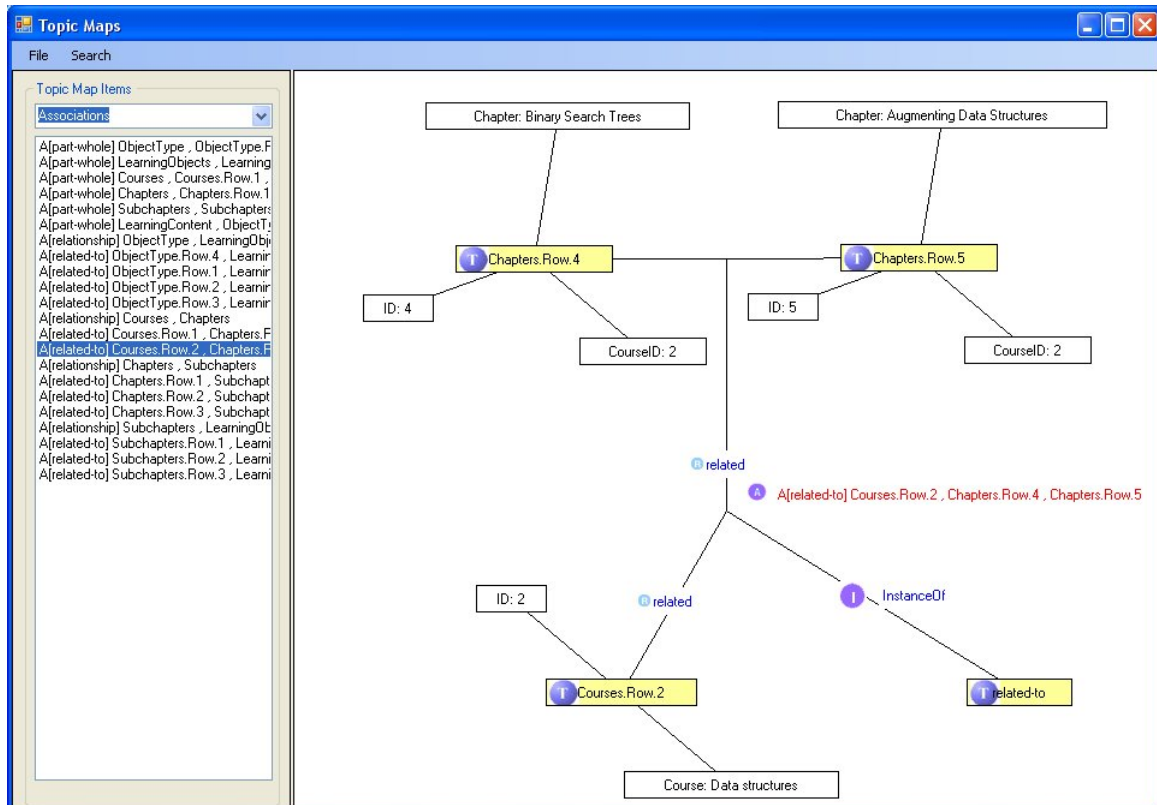


Figure 5: An association of type “related-to”.

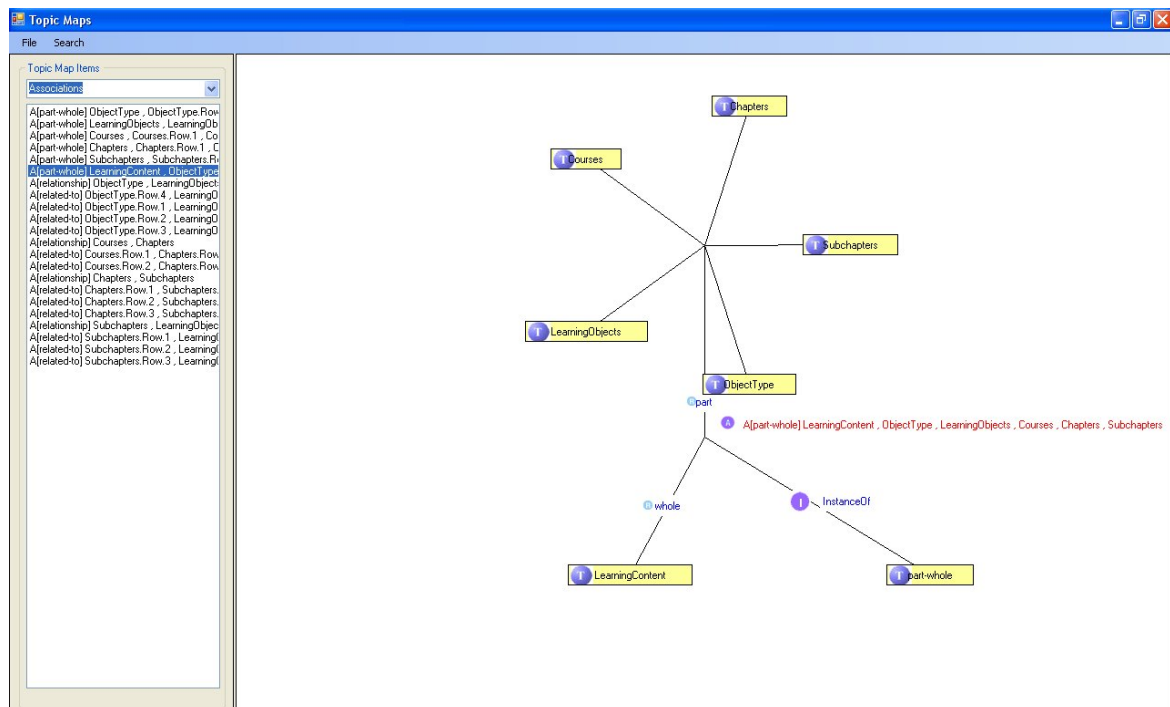


Figure 6: An association of type “part-whole”.

8 Topic Map querying with tolog query language

Along with the Topic Map graphical visualization searching the information that satisfies a number of criteria is also important. A solution for querying a Topic Map is represented by the tolog language. In our experiments we have used the tolog engine provided by Ontopia [17].

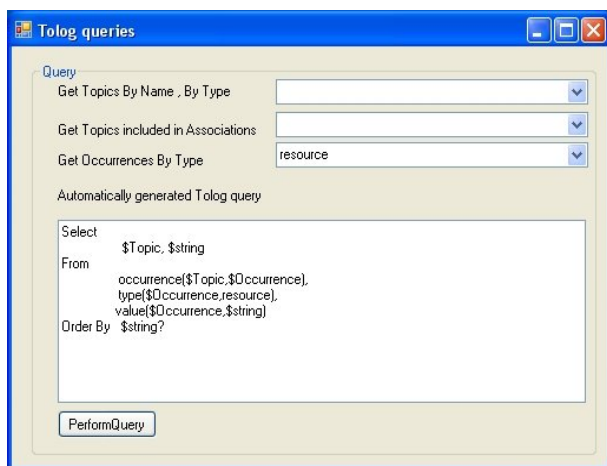


Figure 7: Graphical interface for tolog queries.

For running queries in this way, two operations are necessary:

- loading in engine the generated Topic Map
- building an interface for sending tolog queries to the engine

Tolog’s syntax is a complex one, normally used by experimented users and improper for the e-learning domain.

This is the reason why we propose an original graphical interface for running tolog queries. These queries are automatically generated as a result of user’s choice: he/she can select topics, topic types, associations or occurrence types (figure 7).

The user has the possibility to select a topic type from the available list of topics or the option “All topics” that will return a list with all topics defined in the Topic Map. For example, for the topic type “course”, the application generates the tolog query from the table 1 point a. After the execution, the result is a list with all topics that represent courses.

For example, for the association “has_chapters”, the application generates the tolog query from the table 1 point b. The result is the list of all topics involved in this type of association.

In some cases, a topic can have a number of occurrences. In the TESYS database, a subchapter has learning objects (image file, text file, etc.). All these are defined using the type “resource”. With the graphical interface, the learner can select an occurrence type, and the application generates a tolog query as in the table 1 point c.

Since the results from the tolog engine contain the topic’s name this information is used as a link. When a user clicks on a topic name the TMNavigator searches that name in the list of all topics (kept in memory after loading the xtm file). After the topic is found his details are automatically shown in the right window. If the results contain resource’s name (for example Projection.ppt) the TMNavigator generates a link that

can be used by the user to open/download the content of that file.

Table 1: Automatic generated queries and results.

	Query	Obtained results (examples)	
a)	instance-of (\$Topic, course)?	Course	
		Databases	
		Data structures	
b)	has_chapters (\$Course: whole, \$Chapter: part)?	Course	Chapter
		Data structures	Binary Search Trees
		Data structures	B-Trees
		Databases	The Entity Relationship Model
		Databases	Relational Algebra
c)	Select \$Topic, \$string From Occurrence Type (\$Occurrence, resource) Value (\$Occurrence, \$string) Order By \$string?	Topic	Title
		What is a binary search tree?	BinarySearchTree.ppt
		Searching a B-tree	SearchBTree.ppt
		Projection operator	Projection.ppt

9 Users’ feedback

A number of 60 students participated to the following experiment: they were asked to study the discipline Database using TESYS system, an on-line e-learning platform that uses a tree structure for displaying the learning content (the learner chooses the course, then a chapter, and finally a lesson). The existing relationships between learning objects are implemented as hyperlinks. The student can also use some search criteria. At the same time, they had to study the discipline Multimedia Applications Development using the Topic Map created with this software tool. The students’ opinion over these two learning modalities was recorded with a number of questions presented in table 2.

Table 2: Questionnaire.

Question	The number of “Yes answers
Is it easy to find the necessary subject in TM?	45
Is it easy to find associated subjects in TM?	51
Does TM enable the study in a free way?	35
Does TM keep you motivated?	48
Is TM easy to use?	41
Is the subjects order in the course difficult to understand using TM?	48
Is it easy to understand the whole structure of knowledge using TM?	55
Is the Topic Map querying process useful and fast?	53
Are there too many subjects displayed in TM?	34
Do you favour TM as a learning modality? Why?	45
Do you think that these 2 learning modalities complete one another?	58

The students’ answers emphasized the fact that the use of Topic Maps in the e-learning field presents positive aspects: they are easy to use and the student can easily pick a subject and see the relationships between subjects. Also, they found very useful the Topic Map querying process using tolog.

The results showed that many students prefer the new learning modality based on Topic Maps due to the following reasons:

- the graphical interfaces are more attractive and modern
- TM offers more dynamism in following the knowledge semantic relationships
- there are no limits in building queries (the user can specify many conditions and combine them in a free way) as in the TESYS system

It is very interesting to note that a large number of students (96%) consider that these two learning modalities can be used together, because they can complete one another.

10 Conclusions

The e-learning systems must contain powerful and intuitive tools for viewing the learning resources, for browsing the lessons or topics and relationships between them, and also for searching the relevant information. An important feature of an e-learning system is the presentation way of the semantic relationships between topics using an appropriate navigational structure.

This aim can be achieved using a modern concept - Topic Map. Topic Maps are an emerging Semantic Web technology that can be used as a means of organizing and retrieving information in e-learning repositories in a more efficient and meaningful way.

As a result, the paper presents the following important aspects:

1. The algorithm for Topic Map automated building starting from a relational database, which is not possible with existing Topic Maps software. This aspect is useful because there are many e-learning systems that store the educational content in a database.
2. An improvement of this algorithm with the help of a configuration file that is automatically generated as a result of user choices. This way, the Topic Map represents in a more appropriate manner the semantic content in the e-learning domain.
3. A graphical view that allows Topic Map navigation is useful in studying topics that represent in fact learning objects and associations between them.
4. A graphical window for building tolog queries on the Topic Map. This window allows learners to filter the information based on their interest in an interactive way.

The students have found this new modality of knowledge visualization and filtering useful, modern and interesting.

References

- [1] Rosenberg, M. (2001). E-Learning: Strategies for Delivering Knowledge in the Digital Age. In: New York, McGraw-Hill
- [2] Wentling, T., Waight, C., Gallaher, J., La Fleur, J., Wang, C., Kanfer, A. (2000). E-Learning: A Review of Literature, <http://learning.ncsa.uiuc.edu/papers/elearnlit.pdf>
- [3] Moodle (2009). http://docs.moodle.org/en/Development:Database_schema_introduction
- [4] Dicheva, D., Dichev, C., Dandan, W. (2005). Visualizing topic maps for e-learning. *Advanced Learning Technologies (ICALT 2005)*, IEEE Computer Society, 950 – 951
- [5] Dandan, W., Dicheva, D., Dichev, C., Akouala, J. (2007). Retrieving information in topic maps: the case of TM4L. *ACM Southeast Regional Conference*, pp.88-93
- [6] Kolås, L. (2006.) Topic Maps in E-learning: An Ontology Ensuring an Active Student Role as Producer. *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, Ed/ITLib Digital Library, Association for the Advancement of Computing in Education (AACE), pp. 2107-2113
- [7] Garshol, L.M. (2004). Metadata? Thesauri? Taxonomies? Topic Maps! *Journal of Information Science*, vol. 30, no. 4, pp.378-391, CILIP (Chartered Institute of Library and Information Professionals), ISSN 0165-5515
- [8] Stanescu, L., Mihaescu, M.C, Burdescu, D.D, Georgescu, E., Florea, L. (2008). An Improved Platform for Medical E-Learning. *Lecture Notes in Computer Science 4823*, Springer, pp.392-403
- [9] Rath, H. (2003) *The Topic Maps Handbook*. Empolis GmbH, Gutersloh, Germany
- [10] TopicMaps. Org, <http://www.topicmaps.org/>
- [11] XML Topic Maps (XTM) 1.0. <http://topicmaps.org/xtm/1.0/index.html>
- [12] Ontopia (2009). <http://www.ontopia.net/>
- [13] Mondeca,(2009). <http://www.mondeca.com/>
- [14] Blackboard (2009). <http://www.blackboard.com/-Teaching-Learning/Learn-Platform.aspx>
- [15] Stanescu, L., Burdescu, D., Mihai, G., Ion, A., Stoica,C. (2008). Topic Map for Medical E-Learning, *Studies in Computational Intelligence*, Springer-Verlag, Vol. 162/2008, pp.305-310
- [16] Matsuura, S., Naito, M. (2008). Creating a Topic Maps based e-Learning System on Introductory Physics, *Leipziger Beitrage zur Informatik: XII*, pp.247-260
- [17] Tolog language tutorial, Ontopia, (2007) <http://www.ontopia.net/omnigator/docs/query/tutorial.html>
- [18] Omnigator, (2010) <http://topicobserver.com:8080/-omnigator/models/index.jsp>.
- [19] Pepper S. (2009) Topic Maps. *Encyclopedia of Library and Information Sciences*, Third Edition DOI: 10.1081/E-ELIS3-120044331
- [20] Mihai, G., Stanescu, L., Burdescu, D, Brezovan, M., Stoica Spahiu, C. (2009) A Topic Map for “subject-centric” Learning. *Studies in Computational Intelligence* 237/2009, Springer, pp.141-150
- [21] Pepper, S., Garshol, L. M. Lessons on Applying Topic Maps. <http://www.ontopia.net/topicmaps/-materials/xmlconf.html#General-Principles>