Schema Repository for Database Schema Evolution[1]

Hassina Bounif
EPFL, Swiss Federal Institute of Technology (Switzerland)
Hassina.bounif@epfl.ch

Rachel Pottinger
University of British Columbia (Canada)
rap@cs.ubc.ca

Abstract

The paper presents a schema repository, an original repository containing different kinds of database schemas. The repository is part of a multidisciplinary approach for schema evolution called the predictive approach for database evolution. The schema repository has a dual role in the approach: (1) during the data-mining process, the repository identifies and analyzes trends on collected schemas belonging to the same domain. (2) the repository is used in the building of the requirements ontology — a domain ontology that contributes in the database design and its evolution. This paper presents both the design and a heuristic-based method to populate such a repository.

1. Introduction and Motivation

With the growth of Internet use and the progress made in the technology resources, especially with the emergence of Enterprise Resource Planning systems (ERPs), databases, the core of any information system, need to adhere to these changing environments by being flexible to the changes on their schemas and respective data. Existing techniques that allow changes on database schemas (e.g., versioning and modification approaches) are posterior solutions — they react to changes rather than plan ahead for them. Adopting such solutions to implement this process of change is not efficient when facing complex changes and generates shortcomings such as high costs in human resources and in financial support.

Our research concentrates on a new approach for database schema evolution; by focusing on the conceptual level, we find a new perspective on the problem of schema evolution.

The objectives of the approach consist in preparing for evolution by anticipating changes before they occur: potential changes are inspected and integrated into the schema for future use. An overview of our structured approach called the Predictive Approach for Database Schema Evolution is illustrated in figure 1.

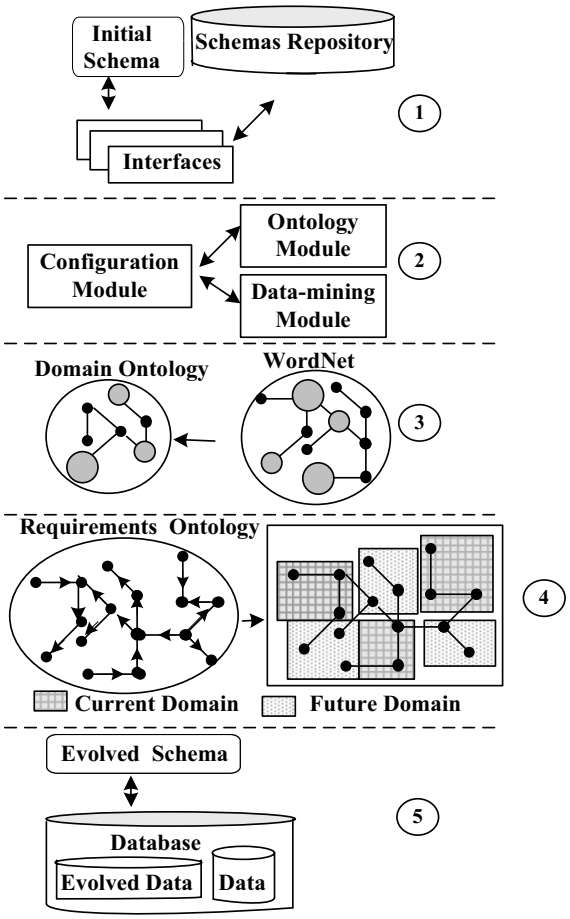

Figure 1. Overview of the Predictive Approach for Database Schema Evolution

The paper presents a schema repository, an original repository containing database schemas that has a double function in the predictive approach: (1) during the data-mining process, the repository identifies and analyzes trends on different kinds of schemas belonging to the same domain. For this purpose, two types of search by contents are applied on the corpuses obtained from the repository: descriptive analysis search and changes

[1] This work is carried out as part of IM2 (Interactive Multimodal Information Management) (http://www.im2.ch), Swiss National Competence Centre in Research (NCCR), supported by the Swiss National Research Fund

analysis search. (2) the repository is used to build a requirements ontology, a domain ontology that contributes in the database design and its evolution. The repository is populated using an evolutionary heuristic algorithm.
The contributions of this paper are as follows:

1- Presentation of the schema repository structure and population.
2- Presentation of the heuristic-based method to construct this repository.
3- Presentation of the two functions of the schema repository in the predictive approach: 1) in the data mining process 2) in the construction of the requirements ontology.
4- Presentation of the related work concerning the use of repositories in databases integration and evolution.

The paper is organized as follows. Section 2 provides an overview of the architecture of the schema repository. Section 3 describes the heuristic used in populating the repository. Section 4 describes the role of the schema repository in the predictive approach for database schema evolution. Section 5 describes related work, and Section 6 concludes.

2. Schema Repository Architectural View

Each schema repository is designed to contain many different schemas and versions of schemas that model a specific domain. Currently it is designed to contain schemas in any current and recent schema model, such as ER, relational, object and object-relational, XML, and ontological schemas expressed with OWL technology. We focus on two schemas domain: meetings and Geographical Information Systems (GIS). These schemas originated from of three main sources:

1) A collection of free schemas available on the web
2) A collection of schemas from research institutions. E.g., a collection of schemas for the meeting domain can be downloaded from *http://mmm.idiap.ch/*
3) A collection of synthetic schemas. A synthetic schema is a schema created based on applications and associated interrelated domains. The creation process is described in Section 2.1.

2.1. Constructing Synthetic Domain-Related Schemas

To create the synthetic schemas, we developed a new strategy called Key Vocabulary Strategy (**KVS**). We developed our own strategy for the following two reasons:
1- This guarantees that our repository is applicable beyond domains where real schemas were available [1].
2- This allowed us to focus our efforts on the repository rather than schemas construction, which has been well studied.

The KVS strategy is based on three hypotheses:
H1: Each application is recognizable with a set of parameters called key vocabulary parameters. For example the number of tables in the suggested schema for these applications.
H2: Applications related to a specific domain give concepts and relationships that could describe that domain.
H3: Applications related to a specific domain can be used to identify other potential applications for that domain in order to create schemas from them.

2.2. Repository Structure Description

The repository structure is simple to set up and is flexible enough to support different domains. There are three repository topologies: 1) One schema topology, 2) Multiple schemas topology, 3) Hybrid topology. We define each topology as we define the components of the repository. The different topologies are illustrated in figures 2 and 3.

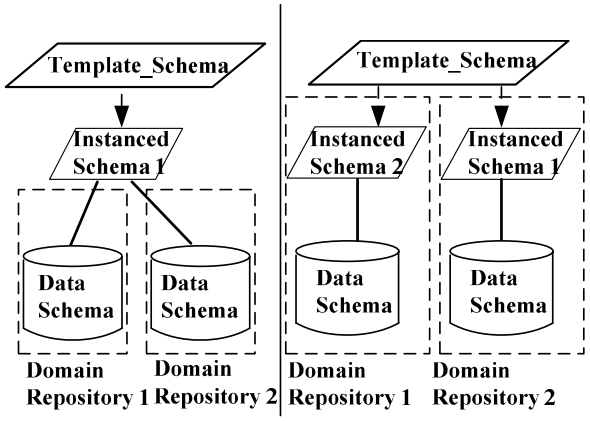

Figure 2. Presentation of the One Schema and Multiple Schemas Topologies

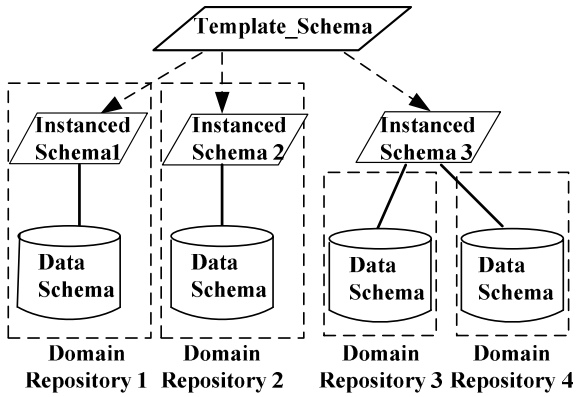

Figure 3. Presentation of the One Schema and Multiple Schemas Topologies

The repository is composed of three main components: the *template_schema*, the *instanced schemas* and finally the *data schemas* for each domain to be modeled. Each is described below in more detail.

1- Template Schema: the schema of the schema repository. According to One Schema Topology (figure 2), the template schema consists of five entities implemented using the relational model as follows:

Versioned Schema	Represents a version of the schema to be stored
Schema	Represents the initial schema
Concept	It is an entity in ER schema, a class in Object schema, a class in an Ontology schema or a node in xml schema
Concept Attribute	Represents a property of a concept
Relationship	Represents a set of relations between two concepts or a concept with its own

2 - Instanced Schemas: are the schemas created for specific domains. The instantiation of a schema for meeting domain is presented below in the figure 4 using UML modeling:

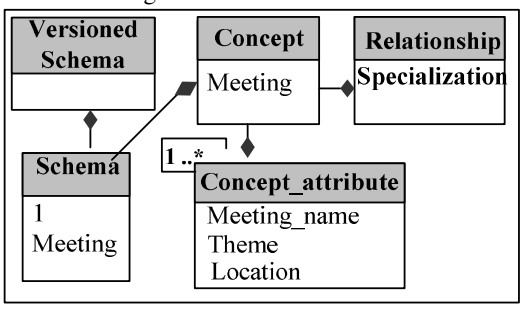

Figure 4. Overview of the Schema using UML Modeling

3- Data Schemas: represents the meta-data extracted from the stored schemas.

3. Repository Population Heuristic

We have adopted a heuristic approach to populate the repository. We have chosen such approach because the heuristics facilitate the schemas selection process and allow to maximize the predictive accuracy of the repository and to make it interesting, relevant and useful for its double function.

Generally, heuristics are classified into three categories that are respectively: *constructive heuristic, heuristic for local search and evolutionary heuristic*. The evolutionary, heuristic, the category we are interested in, acts on a population of individuals (of the solutions or pieces of solutions) who cooperate and adapt themselves individually. We have chosen to conceive an algorithm to fill the schemas repository with appropriate schemas based on such category because we need:

1- The cooperation among selected schemas in order to detect new schemas that fulfill the specified requirements.

2- The adaptability of the selected schemas: the repository contains new versions of the schemas that have been modified.

The algorithm operates in four main steps which are:

1- At first, the processing starts generating the schemas that represent the initial population of the repository called also parents population.

2- A random representative sample is chosen from the parent population

3- A children population is generated based on parent sample selected in the previous step

4- Finally, the selection of the appropriate individuals from this children population.

The complete Skelton of the algorithm is presented below:

Heuristic Method Algorithm
Initialization (parameters name_domain, similar_name, similar_similar_domain)

1- Generate an initial population P1 of schemas according to the following criterion:

40% of do the schemas belong to the same domain of the data base to be modeled

40% of schemas belong to a similar domain

20% of schemas belong to a similar domain of the similar domain

Stop Criteria :

- *Number of schemas in the repository that respects the stated percentages of each category as presented on the top*

Do while no the criterion of stop is satisfied,

1. Choose from the table concepts a representative sample of the present concepts. The representative sample should also contain the concepts present in schemas versions

2. Apply a search operation to look for new concepts according to the criterion presented in top while keeping the same percentages.

3. Select the concepts that are not present in the table concepts.

4. Store the concepts selected in the table concepts

4. Schema Repository Role in the Predictive Approach for Database Schema Evolution

In the predictive approach, the repository is used in two main phases: the Mining process phase and in the Ontology construction phase.

4.1. In the Data Mining Process

In the data-mining process, the corpuses extracted for the repository are exploited into two types of searches by content:

4.1.1. Descriptive Analysis Search

This analysis allows making a set of descriptive statistical operations on the concepts of the repository such as the calculation of their frequencies and their classification. This is illustrated in figure 5.

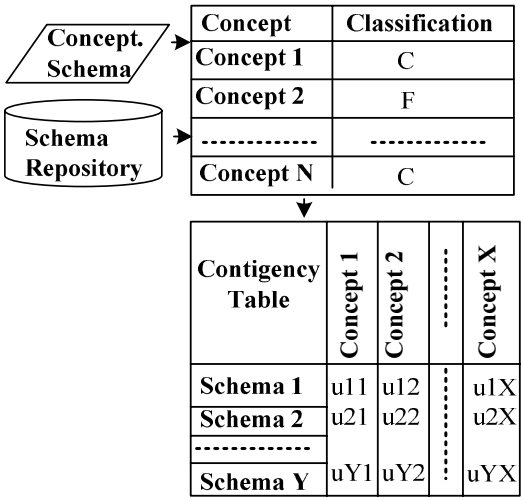

Figure 5. Descriptive Analysis Search

4.1.2. Changes Analysis Search

The changes that occur on the entities during the evolution of their corresponding schemas are visible on their different schema versions. Therefore, the different versions of the schemas are also stored in the repository if they exist. To identify these entities that undergo changes, we have defined both:

1- A list of changes that could be located on the schema during its evolution. It is illustrated in the following table.

List of Changes
Create a component
Create a concept
Create an attribute
Delete a component
Update a component
Change its name
Change its component kind
Change its component type
Change its parent
Change its child

2) A multidimensional matrix called changes history matrix. This matrix allows tracing the history of the changes of schemas for the entire repository. This is illustrated in figure 6.

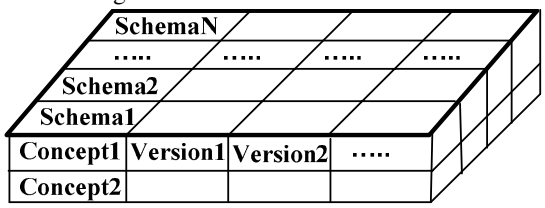

Figure 6. A Changes History Matrix

4.2. In the requirements Ontology Construction

The schema repository has an important role in the requirements ontology construction. It is the main source from which data is taken out. However, before to expose the schema repository role, it is crucial for the readers to understand what requirements ontology is and what its objectives are in the predictive approach for evolution.

4.2.1. Requirements Ontology Definition and Objectives

A Requirement Ontology (RO) is a domain ontology that represents a new way to model user's requirement for a database schema to be modelled and evolved over time.

Intuitively, it consists of two kinds of partitions, the ones representing current requirements and the ones corresponding to potential future requirements, called respectively Current Domain and Future Domain.

In [2] and [3], ontology offers better ways for database design such as:

1- Suggesting missing entities and relationships

2- Generating a conceptual model from scratch

3- Identifying the relevant entities to be included by the database designer in the schema

4- Helping in the process of schemas integration by checking on the ontology for synonyms to identify the concepts that are common to the schemas to be integrated.

In the predictive approach for database schema evolution, the requirements ontology has the advantage to be used for the database design and its evolution in the following ways:

1- Suggesting entities and relationships which could represent potential future requirements. These selected entities are included afterwards in the database schema.

2- Generating a whole conceptual schema that includes all potential future requirements from scratch.
3- Identifying the entities that undergo changes and including them in the initial schema, taking into consideration their evolved states.

4.2.2. Requirements Ontology Construction

The ontology construction process is divided into three main steps. This is illustrated in figure 7.

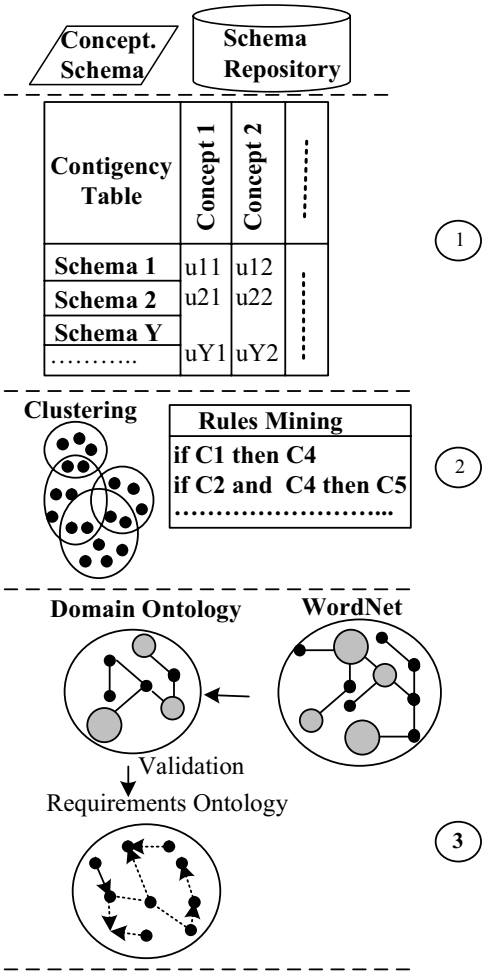

Figure 7. Requirements Ontology Process Construction

The different steps from the data schemas to the domain ontology are as follows:

1-Working data that correspond to concepts extracted from both the schemas repository and the conceptual schema of the database to be modeled are put into data matrix.

2-Exploratory data analysis using data mining algorithms (clustering and rules mining) to choose the concepts necessary for building the domain ontology as well as the semantic relations that link these concepts in an unsupervised way.

3-Domain ontology is built then formally validated using description logic in order to have as final result a requirements ontology.

5. Related Work

Statisticians have employed repositories in data mining and data warehouse tasks in order to analyze their corresponding data and make decisions from that. However, recently, the database community has joined them as well. Different research works use such repositories to study the schemas integration, matching and evolution. For example, in the work presented in [4] corpuses that model similar concepts are analyzed to benefit from their similarity and difference in modeling in order to generate modeling statistics. This tendency is increasing especially because 1) the perception of the real world is more and more complex and most of the time database designer cannot achieve the ideal representation that fits it. 2) the huge amount of available data and the technology development.

6. Conclusion

We have presented a schemas repository, an original repository that could enclose different kinds of schemas and which is a part of new multidisciplinary approach for database schema evolution. The repository could be used for other purposes such as schemas integration in databases or in ontologies. The next step of this work is the completion of the acquisition of schemas and the integration of the schema repository in the prototype for the database schema evolution that is under development.

7. References

[1] Liu, H. and H. Motoda (1998). *Feature Selection: for knowledge discovery and data mining*, Kluwer Academic Publishers. .

[2] Sugumaran, V. and V. C. Storey (2002). "Ontologies for conceptual modeling: their creation, use, and management." Data Knowledge. Engineering 42(3): 251-271.

[3] Sugumaran, V. and V. C. Storey (2002). "An Ontology-Based Framework for Generating and Improving Database Design". Natural Language Processing and Information Systems, 6th International Conference on Applications of Natural Language to Information Systems, NLDB, Stockholm, Sweden, Springer.

[4] Madhavan, J., P. A. Bernstein, et al. (2005). "Corpus-based Schema Matching". Twenty First International Conference on Data Engineering (ICDE'2005), Tokyo, Japan., IEEE Computer Society.