A Tool for Filtering Large Conceptual Schemas

Antonio Villegas, Maria-Ribera Sancho, and Antoni Olivé

Department of Service and Information System Engineering
Polytechnic University of Catalonia
Barcelona, Spain
{avillegas,ribera,olive}@essi.upc.edu

Abstract. The wealth of knowledge the conceptual schemas of many real-world information systems contain makes them very useful to their potential target audience. However, the sheer size of those schemas makes it difficult to extract knowledge from them. There are many information system development activities in which people needs to get a piece of the knowledge contained in a large conceptual schema. We present an information filtering tool in which a user focuses on one or more entity types of interest for her task at hand, and the tool automatically filters the schema in order to obtain a reduced conceptual schema including a set of entity and relationship types (and other knowledge) relevant to that task.

Keywords: Large Schemas, Filtering, Entity Types, Importance.

1 Introduction

The conceptual schemas of many real-world information systems are too large to be easily managed or understood. There are many information system development activities in which people needs to get a piece of the knowledge contained in a conceptual schema. For example, a conceptual modeler needs to check with a domain expert that the knowledge is correct, a database designer needs to implement that knowledge into a relational database, a software tester needs to write tests checking that the knowledge has been correctly implemented in the system components, or a member of the maintenance team needs to change that knowledge. Currently, there is a lack of computer support to make conceptual schemas usable for the goal of knowledge extraction.

Information filtering [3] is a rapidly evolving field to handle large information flows. The aim of information filtering is to expose users only to information that is relevant to them. We present an interactive tool in which the user specifies one or more concepts of interest and the tool automatically provides a (smaller) subset of the knowledge contained in the conceptual schema that is likely to be relevant. The user may then start another interaction with different concepts, until she has obtained all knowledge of interest. We presented the theoretical background behind this tool in [4,5].

O. De Troyer et al. (Eds.): ER 2011 Workshops, LNCS 6999, pp. 353–356, 2011.

2 The Filtering Process

In this section we describe how a large conceptual schema can be filtered using the method implemented by our tool, which corresponds to the demonstration we intend to perform. The main idea is to extract a reduced and self-contained view from the large schema, that is, a filtered conceptual schema with the knowledge of interest to the user. Figure 1 presents the three steps of our filtering process.

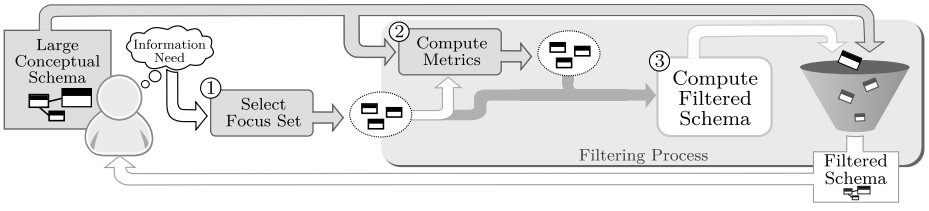

Fig. 1. Method overview

The first step consists in preparing the required information to filter the large schema according to the specific needs of the user. Basically, the user focus on a set of entity types she is interested in and our method surrounds them with additional related knowledge from the large schema. Therefore, it is mandatory for the user to select a non-empty initial focus set of entity types of interest.

During the second step our method computes the required metrics to automatically select the most interesting entity types to extend the knowledge selected in the focus set of the first step. The main goal of these metrics is to discover those entity types that are relevant in the schema but also that are close (in terms of structural distance over schema) to the entity types of the focus set. We presented a detailed definition of such metrics in [4].

Finally, the last step receives the set of most interesting entity types selected in the previous step and puts it together with the entity types of the focus set in order to create a filtered conceptual schema with the entity types of both sets. The main goal of this step consists in filtering information from the original schema involving entity types in the filtered schema. To achieve this goal, the method explores the relationships and generalizations/specializations in the original schema that are defined between those entity types and includes them in the filtered schema to obtain a connected schema.

3 The Filtering Tool

Our filtering tool is developed as a web client that interacts with a web service following the SOAP protocol. The filtering web service we have developed makes use of a modified version of the core of the USE tool [2] to load and maintain the knowledge of the large schema the user wants to explore. In our demonstration the large schema to filter consists of a subset of the HL7 V3 schemas [1] containing more than 2,500 entity types from healthcare domains [5].

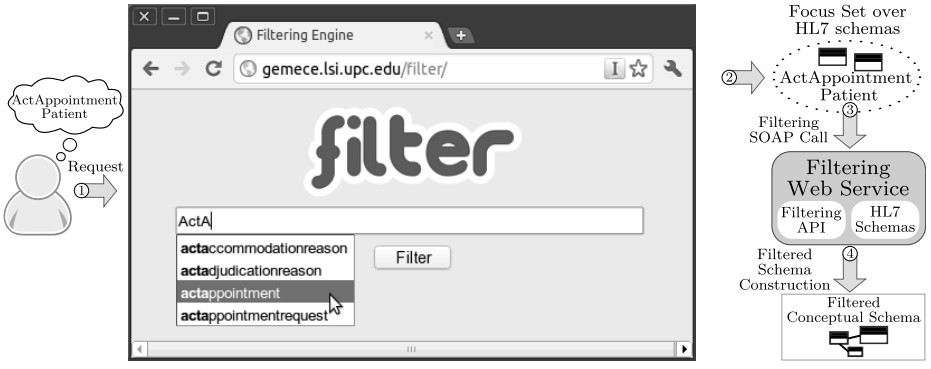

Fig. 2. Request of a user to our filtering tool

Figure 2 depicts the main components that participate in a user request to our filtering tool. The user writes in the search field the names of the entity types she is interested in. Our web client automatically suggests names while the user is writing to simplify that task and to help her discovering additional entity types. In the example of Fig. 2 the user focuses on the entity types *ActAppointment* and *Patient*. Once the request is complete our web client processes the focus set and uses the filtering API of the web service through a SOAP call. The web service analyses the request and constructs the related filtered conceptual schema following the filtering process described in the previous section.

Figure 3 shows the components of the response. The reduced schema produced by our web service is an XMI file containing 8 entity types. In order to increase the understandability of the schema we make use of an external service (`http://yuml.me`) to transform the filtered schema from a textual representation to a graphical one. As a result, the user can rapidly comprehend from the schema of Fig. 3 that *SubjectOfActAppointment* connects the entity types *ActAppointment* and *Patient*, which means that in the HL7 V3 schemas a patient is the subject of a medical appointment. Subsequently, the user can start again the cycle with a new request if required.

4 Summary

We have presented a tool to assist users to deal with large conceptual schemas that allows to focus on a set of entity types of interest and automatically obtains a reduced view of the schema in connection with that focus. Our implementation as a web service provides interoperability and simplifies the interaction with users. A preliminary version of the filtering tool can be found in `http://gemece.lsi.upc.edu/filter`.

Our immediate plans include the improvement of our tool by adding a more dynamic view of the filtered schema instead of the static image obtained by the present external service. As a result, we are introducing more interactive features such as selection of schema elements and new filtering interactions from that selection.

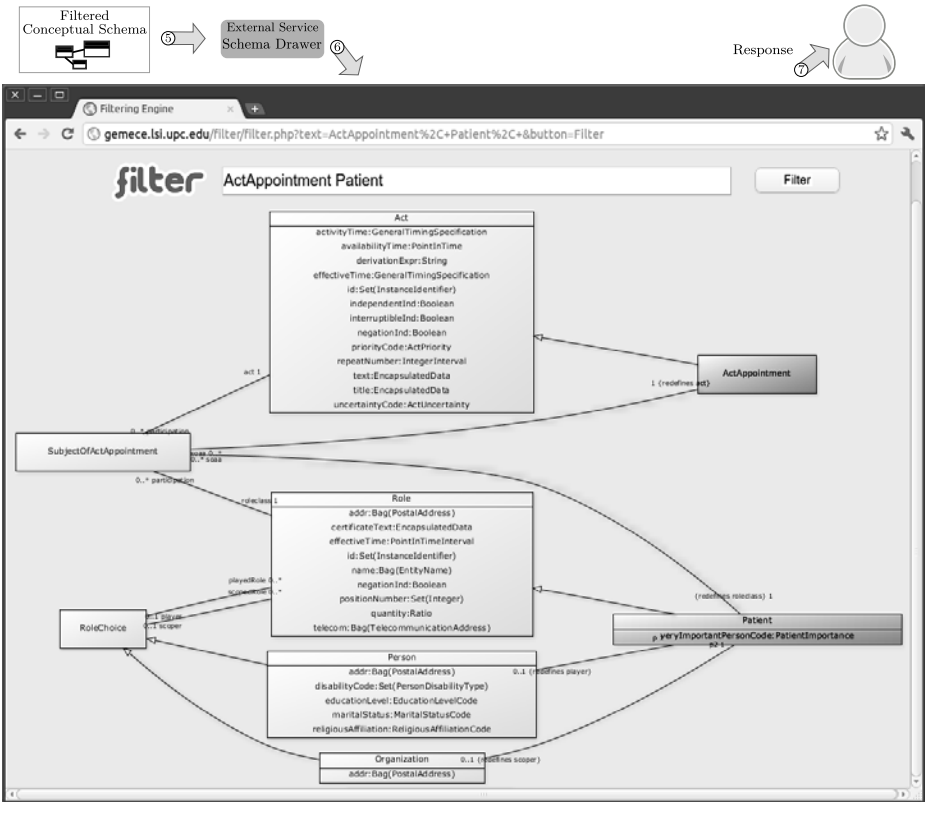

Fig. 3. Response of our filtering tool to the user

Acknowledgements. This work has been partly supported by the Ministerio de Ciencia y Tecnologia and FEDER under project TIN2008-00444/TIN, Grupo Consolidado, and by Polytechnic University of Catalonia under FPI-UPC program.

References

1. Beeler, G.W.: HL7 version 3 – an object-oriented methodology for collaborative standards development. International Journal of Medical Informatics 48(1-3), 151–161 (1998)
2. Gogolla, M., Büttner, F., Richters, M.: USE: A UML-based specification environment for validating UML and OCL. Science of Computer Programming (2007)
3. Hanani, U., Shapira, B., Shoval, P.: Information filtering: Overview of issues, research and systems. User Modeling and User-Adapted Interaction 11(3), 203–259 (2001)
4. Villegas, A., Olivé, A.: A method for filtering large conceptual schemas. In: Parsons, J., Saeki, M., Shoval, P., Woo, C., Wand, Y. (eds.) ER 2010. LNCS, vol. 6412, pp. 247–260. Springer, Heidelberg (2010)
5. Villegas, A., Olive, A., Vilalta, J.: Improving the usability of HL7 information models by automatic filtering. In: IEEE 6th World Congress on Services, pp. 16–23 (2010)