SimOLAP: A System for the semi-automatic implementation of Simulation Data Warehouses

Sandro Bimonte*, Nicolas Dumoulin*

*Irstea, 9 Avenue Blaise Pascal - CS 20085 - 63178 Aubière
sandro.bimonte@irstea.fr, nicolas.dumoulin@irstea.fr http://motive.cemagref.fr

Abstract. Data Warehouses and OLAP systems allow decision-makers exploring and analyzing huge volumes of data modeled according the multidimensional model, and extracted from heterogeneous data sources. Usually, DW design is a complex, and time and resources consuming task. Then, DW experts are necessary during design and implementation phases. In this paper, we present a new methodology and a tool allowing modelers (DW unskilled users) to design and implement DWs for analyzing simulation results data by themselves, without any intervention of DW experts.

1 Introduction and motivations

Nowadays in order to understand, explain and predict trends, dynamics modeling is extensively used to study complex phenomena and scenarios in different contexts: health, climate change, demography, etc. Nevertheless, to calibrate and validate these models, modelers need to make several replications of each simulation to get representative results, leading to huge volumes of results data sets. Although some efforts have been done to provide modelers with tools do design model experimentations (Reuillon et al., 2013), modelers cannot automatically store their simulation data results and explore them by means of interactive tools to validate their experimentations and discover unknown patterns. Then, modelers need to design and implement DWs by themselves in an incremental and iterative way allowing looking for the best way to analyze their data, without any intervention of DW experts. Thus, a hybrid DW design methodology where conceptual, logical and physical design phases are automatic is mandatory. Moreover, as stated in (Bimonte et al., 2013), DW unskilled users need real OLAP clients to validate generated prototypes according to an agile DW development methodology.

Several works propose to automatically generate multidimensional schema from data sources (Romero and Abelló, 2009) (i.e. data-driven approaches), but to best of our knowledge no work take into account data sources modeled as complex trees used to represent simulation data results. In requirement-driven approaches, the formalization of the requirements is usually expressed by DW experts using complex formalisms such as conceptual modelsor DW/DB languages such as SQL. They have a major drawback: they define a gap between users' intentions and their implementation, since DW experts have to translate them into a formalism that is not comprehensible by modelers. Thus, some works propose using the natural language to express needs and define DW schema, but too many ambiguities are issued for these methods to be useful in complex real projects.

2 Running example

To illustrate the motivations of this work, let us consider a simulation model designed in the European project PRIMA. In this project, the model takes as input an artificial population where individuals are described by their age, status and occupation, and are gathered in households respecting a variety of statistical constraints. Then, the dynamics of the model rules the evolution in time of the characteristics of individuals and households, with demographic and economic events. The simulation model results are made of such populations at different time steps (e.g. occurrence of activities for unemployed individuals). An example of simulation result structure is shown on figure 1. In this example, the simulation model outputs the age and status of individuals. These individuals belong to households, which are defined by a type. Individuals are located in municipalities. All these data are generated at different simulation time steps and generally for several replications (parameter rngSeedIndex). A simulation state is described by a set of parameters and the values of these variables for each time step of a simulation.

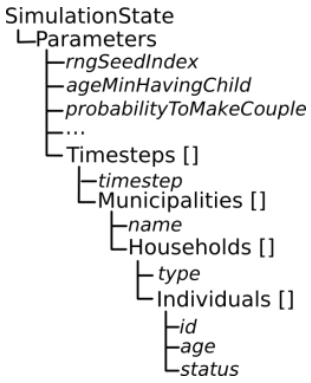

FIG. 1 – *Simulation model result structure*

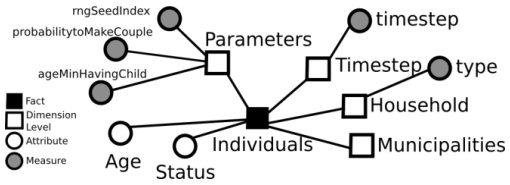

FIG. 2 – *Multidimensional schema*

The development of such models implies to perform a large number of simulations to test the influence of uncertain parameters or alternative hypotheses about the dynamics. This produces a huge amount of simulation results, which require specific tools to be analyzed properly. For example, let us consider a set of simulation results (figure 1), and a stakeholders group that wants to explore, compare and visualize the individual maximum age according to their household types, municipalities and simulation states, and for all replications. For this purpose, they can adopt the multidimensional schema represented in figure 2. This schema represents individual ages according to Household, Municipality and SimulationState values. In this schema Individual represents facts and Household, Municipality, Parameters and Timestep dimensions. The dimension Household contains the hierarchy Household $\longrightarrow$ Household Type, aggregating the measure age at different levels. After, we generate the corresponding spatial data cube that allows users analyze and explore simulation results by means of OLAP technologies. An example of OLAP query using this multidimensional schema is: “What is the average age of individuals per municipality and household type?”.

3 SimOLAP

The SimOLAP methodology (for more details (Mahboubi et al., 2013)) is based on the following steps: **1.** Modelers express the functional requirements by simply choosing an element of the simulation result data as fact (requirement-driven fact analysis). **2.** The multidimensional schema (i.e. measures, aggregations, dimensions and hierarchies) is automatically generated (conceptual design). **3.** The conceptual schema is automatically translated into a logical schema and prototyped (logical design and deployment). **4.** The ETL phase from results data is automatically implemented **5.** Modelers access and explore data using simple pivot tables, so as to validate the prototype (OLAP based validation). If the prototype is not validated, go back to step 1.

The SimOLAP methodology is fully automatic requiring users to only choose the data to be analyzed. Moreover, since ETL is automatically supported and users are provided with real OLAP report, our methodology can be considered as a hybrid automatic agile methodology.

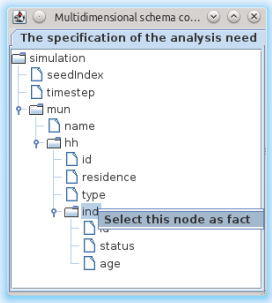

FIG. 3 – *SimOLAP DW design component*

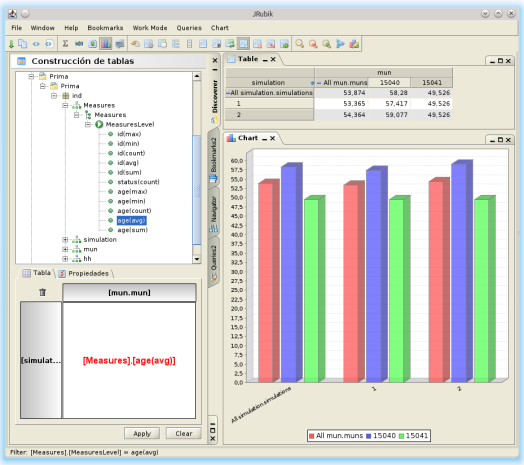

FIG. 4 – *SimOLAP DW Implentation component – OLAP Client*

The architecture of SimOLAP is composed of two main components: DW design and the DW implementation. The former allows modelers to choose their multidimensional analysis need (the fact), among all the elements of the simulation data structure, using a simple user-friendly visual interface (figure 3). In this way, decision-makers needs are mapped on data sources according to the hybrid driven methodology and the resulting multidimensional model is generated. The DW implementation component is based on typical Relational OLAP architecture where Postgres is used to store data (DW tier); the full-featured MDX-based OLAP Server Mondrian is used to implement the generated multidimensional model; finally the JRubik tool is used to let the modelers explore their warehoused data. Finally, according to the agile methodology, which minimizes DW experts' intervention, the ETL is implemented using some Java and PL/SQL procedures. SimOLAP thanks to the well-established mapping be-

tween simulation results and its multidimensional representation automatically generates ETL procedures, the SQL scripts for star-schema representation of the DW and its XML Mondrian representation. The results of the following OLAP query "What is the average age for per municipality and simulation run?" defined by the simple interaction with the graphical displays of the OLAP client JRubik is shown on figure 4.

References

Bimonte, S., E. Edoh-Alove, H. Nazih, M. Kang, and S. Rizzi (2013). Protolap: Rapid olap prototyping with on-demand data supply. In *International workshop on Data warehousing and OLAP, 28/10/2013-28/10/2013, San Francisco, USA*, pp. 61–66.

Mahboubi, H., S. Bimonte, G. Deffuant, J. Chanet, and F. Pinet (2013). Semi-automatic design of spatial data cubes from simulation model results. *International Journal of Data Warehousing and Mining 9*(1), 70–95.

Reuillon, R., M. Leclaire, and S. Rey-Coyrehourcq (2013). Openmole, a workflow engine specifically tailored for the distributed exploration of simulation models. *Future Gener. Comput. Syst. 29*(8), 1981–1990.

Romero, O. and A. Abelló (2009). A survey of multidimensional modeling methodologies. *International Journal of Data Warehousing and Mining (IJDWM) 5*(2), 1–23.

Résumé

Les entrepôts de données et les systèmes OLAP permettent aux décideurs d'explorer et d'analyser d'énormes volumes de données modélisées suivant un modèle multidimensionnel et extraites de sources de données hétérogènes. En général, la conception d'un ED est compliqué, et demande du temps et de nombreuses ressources. De plus des experts en ED sont nécessaires pendant les phases de conception et d'implémentation. Dans cet article, nous présentons une nouvelle méthodologie et un outil permettant aux modélisateurs (ne connaissant pas les ED) de concevoir et d'implémenter des ED pour analyser eux-mêmes des données résultantes de simulation, sans l'intervention d'expert en ED.