Preference-Based Datacube Analysis with MYOLAP

Paolo Biondi, Matteo Golfarelli, Stefano Rizzi

DEIS - University of Bologna
Viale Risorgimento 2, Bologna, Italy
paolo.biondi5@unibo.it
matteo.golfarelli@unibo.it
stefano.rizzi@unibo.it

***Abstract*— In this demonstration we present MYOLAP, a Java-based tool that allows OLAP analyses to be personalized and enhanced by expressing "soft" query constraints in the form of user preferences. MYOLAP is based on a novel preference algebra and a preference evaluation algorithm specifically devised for the OLAP domain. Preferences are formulated either visually or through an extension of the MDX language, and user interaction with the results is mediated by a visual graph-like structure that shows better-than relationships between different sets of data. The demonstration will show how analysis sessions can benefit from coupling ad-hoc preference constructors with the classical OLAP operators, and in particular how MYOLAP supports users in expressing preference queries, analyzing their results, and navigating datacubes.**

I. INTRODUCTION

Expressing preferences when querying databases is a natural way to avoid empty results on the one hand, information flooding on the other; preferences also allow for ranking query results so that the user may first see the data that better match her tastes. This features are even more important when querying datacubes through OLAP tools, mainly because: (1) multidimensional databases store terabytes of facts, so OLAP queries may easily flood users with too much information; (2) OLAP users may not know exactly what they are looking for during a session, and preferences enable them to specify patterns to describe the type of information they are interested in, knowing that the most similar data will be returned when no data exactly match those patterns.

For instance, a decision maker may want to analyze high average incomes for 2009 on a datacube storing census data. Since she does not know for sure which are the key factors of this phenomenon, she has to formulate a wide set of OLAP queries characterized by different group-by sets, thus obtaining a huge set of results. Alternatively, since she suspects that high incomes are a state-scale phenomenon, she can formulate a single query including a "hard" constraint in the form of a selection predicate on year 2009 and a group-by expressing the minimum granularity for the required data, and annotate this query with a "soft" constraint in the form of a preference on high incomes grouped by state. Unfortunately, though most commercial OLAP tools provide sophisticated and customizable ways to view and navigate multidimensional data, they give no support at all to express such a type of user preference.

In [1] we argued that preferences in the OLAP domain are characterized by three peculiarities, and we presented an approach for taking them into account: (1) Preferences can be expressed not only on attribute values, that have categorical domains, but also on measure values, that have numerical domains; (2) Preferences can also be formulated on the datacube schema, in particular on the aggregation level of facts (*group-by set*); (3) The space on which preferences are declared is dramatically larger than that of typical transactional databases due to the presence, besides elemental facts, also of aggregated facts.

In this demonstration we present MYOLAP, a Java-based tool based on our approach, and we show how it enhances analysis sessions by supporting users in expressing preference queries and efficiently exploring their results. The main functionalities of MYOLAP are:

1) *Formulation*. Users can express OLAP queries and annotate them with preferences, formulated either visually or through an extension of the MDX language [2]. Preference formulation relies on a preference algebra including a set of base constructors on attributes, measures, and hierarchies.
2) *Analysis*. To effectively explore query results, users visually interact with a graph-like structure (*domination graph*) that emphasizes better-than relationships between different sets of facts. Preferred facts are then displayed in a multidimensional table.
3) *Navigation*. Following the OLAP session paradigm, users interact with multidimensional tables to iteratively formulate new queries in two ways: by applying standard OLAP operators such as roll-up and drill-down, which is done preserving the annotating preference, and by applying a new *next rank* operator that descends the domination graph by exploring the next best-matching facts.

II. APPROACH OVERVIEW

Our approach builds on *qualitative* preferences. A preference on a datacube is a *strict partial order* (i.e., an irreflexive and transitive binary relation) on the space of all facts at all group-by sets of the datacube. Remarkably, the wide expressiveness required by the OLAP domain makes classic

approaches, such as those based on the skyline operator, unsuitable.

In MYOLAP, preferences on the space of facts are inductively engineered by writing a *preference expression* that can be either a *base constructor* or a *composition operator* applied to two preference expressions. To meet the peculiarities of OLAP preferences as mentioned in Section I, we provide a set of ad-hoc base constructors operating either on attributes, measures, or hierarchies; some examples are:

- POS(a, c), that operates on attributes. Facts for which attribute a takes value c are preferred to the others.
- BETWEEN(m, v_{low}, v_{high}), that operates on measures. Facts whose value of m is between v_{low} and v_{high} are preferred; the other facts are ranked according to their distance from the $[v_{low}, v_{high}]$ interval.
- CONTAIN(h, a), that operates on hierarchies. Facts whose group-by set includes attribute a belonging to hierarchy h are preferred to the others.

Preference composition relies on two operators: Pareto composition ($\otimes$), where the composed preferences are considered equally important, and Cascade composition ($\triangleright$), where the composed preferences are considered of progressively decreasing importance. For instance, the preference expression on the CENSUS datacube for the example reported in Section I is BETWEEN(AvgIncome,500K,1000K) $\otimes$ CONTAIN(RESIDENCE, State), that states that facts yielding high average incomes and aggregated by State are preferred to the others. Remarkably, adopting the substitutability semantics [3] allows for closing the set of composition operators on the set of preferences, thus obtaining an algebra.

Preference evaluation in MYOLAP relies on a novel graph-theoretical representation, called *weak better-than graph* (wBTG), for domination relationships between facts [1]. In the wBTG of a given preference expression, each node is associated to a predicate that selects a set of facts from a datacube, and each arc represents a domination relationship between the facts in two nodes. Two types of nodes are distinguished: a *full* node selects one class of equivalent (i.e., substitutable) facts, while a *dotted* node selects two or more equivalence classes of facts induced by a numerical preference such as BETWEEN. Besides, two types of dominations (*strict* and *weak*) are captured by wBTG's, which allows the efficiency in preference evaluation to be considerably improved. While full nodes and strict domination are already used in the literature to prune the search space when evaluating preferences on categorical domains only [3], introducing dotted nodes coupled with weak domination enables an effective evaluation of preferences on both categorical and numerical domains. Given an OLAP query annotated with a preference expression, the *weak-and-strict limiting algorithm* (WEST) introduced in [1] uses the wBTG to efficiently answer this query according to the *best match only* (BMO) model, where all and only the facts not worse than other facts are returned. As discussed in detail in [1], WEST outperforms the main preference evaluation approaches in the literature. More specifically, processing a medium-complexity preference query on a datacube including 20 millions of facts (with no materialized views) takes about 150 seconds; the most expensive queries are those based on the Pareto composition of several constructors operating on (numerical) measures. Noticeably, in the architecture used for this demonstration there is a clear performance overhead due to the management of main-memory data structures by Mondrian.

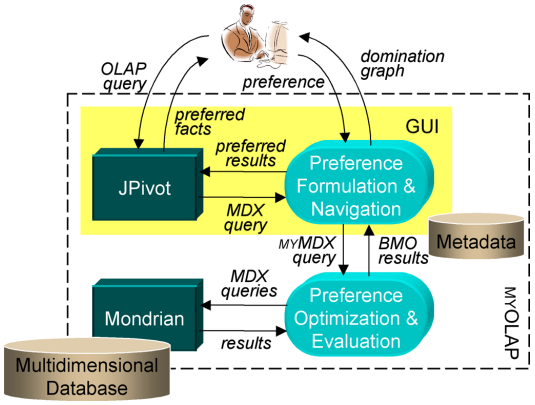

Fig. 1. The MYOLAP architecture

In our approach, preference queries are expressed in an extension of the MDX language that we call MYMDX. MDX (*MultiDimensional eXpressions*) is a de-facto standard for querying multidimensional databases [2]. Some of its distinguishing features are the possibility of returning query results that contain tuples with different aggregation levels and the possibility of specifying how the results should be visually arranged into a multidimensional representation. MYMDX allows an MDX query q to be annotated with a preference expression p through a PREFERRING clause. For instance, the MYMDX query for the example shown in Section I is:

SELECT {AvgIncome} **ON COLUMNS**, $<$ . . .*crossjoins*. . . $>$ **ON ROWS**
FROM [CENSUS] **WHERE** [TIME].[Year].[2009]
PREFERRING AvgIncome **BETWEEN** 500000 **AND** 1000000
AND RESIDENCE **CONTAIN** State

The WEST algorithm uses the wBTG of p for answering q on a datacube according to the BMO model. Basically, the algorithm carries out a topological ordering traversal of the wBTG; for each node b being traversed, an MDX query is generated and executed to select from the datacube the subset of facts satisfying q and the predicate associated to b [1].

III. ARCHITECTURE

The MYOLAP architecture is sketched in Figure 1. A brief explanation of the main components and how they are involved in the data flow is given below:

1) Open source tool JPivot is used for formulating multidimensional queries, for displaying query results in tabular form, and for OLAP interaction with results. Queries can be formulated both visually and in MDX.
2) The *preference formulation & navigation* (PFN) component takes the MDX query returned by JPivot and

annotates it with a preference formulated by the user via a simple visual interface. The annotated query, expressed in MYMDX, is then handed on the *preference optimization & evaluation* (POE) component.

3) The POE component takes the MYMDX query and feeds it into the WEST algorithm, that builds the associated wBTG and progressively generates one different MDX query for each node in the wBTG.
4) Mondrian executes each MDX query and puts the results into its data space. The POE component accesses data and post-processes them by dropping the dominated facts to determine the BMO result.
5) The PFN component visualizes the domination graph for the query, i.e., a graphical representation of the wBTG that guides the user in exploring the BMO result. Rendering a multidimensional view of the facts selected by the user is in charge of JPivot.
6) At this point, the user can interact with the results through OLAP operators such as roll-up and drill-down, as well as modify the annotating preference, thus generating a new query.

IV. INTERACTING WITH MYOLAP

MYOLAP assists users in formulating queries, analyzing their results, and navigating the underlying datacube. The interface is organized into three main areas, namely an *OLAP panel*, where OLAP queries are formulated and multidimensional results are shown, a *formulation panel*, where preferences are composed and edited, and a *preference panel*, where users interact with domination graphs.

A. Formulation

Queries can be formulated using the MYMDX language in the textual tab of the formulation panel. More interestingly, they can also be visually formulated by iteratively composing base preference constructors. Remarkably, users do not need to know the underlying multidimensional schema in detail because MYOLAP guides them in selecting names and parameters. Figure 2 shows an example of visual preference formulation (the preference panel is top right in the figure); in particular, a BETWEEN constructor operating on measure INCTOT is being selected.

B. Analysis

To provide a richer perspective on data analysis and help users to find their way through data that satisfy different parts of their preference expressions, the analysis of BMO results takes place at two distinct levels, as shown in Figure 3 and described below.

In the domination panel (bottom right in the figure), user interaction with data is mediated by the domination graph, with an emphasis on sets of equivalent facts (represented by nodes) and on better-than relationships between them (represented by arcs); this gives users an overall view of query results, providing a higher abstraction related to the preference structure. A more detailed information is conveyed by colors: a white node in the domination graph has been evaluated and turned out to be empty; a node colored from green (better) to red (worse) contains a set of preferred facts; a grey node is dominated. At this level, the user can analyze the BMO result by selecting one of the nodes, which leads to displaying in the OLAP panel the corresponding facts; the preference predicates that are fulfilled by the selected node are displayed in bold within the formulation panel.

The OLAP panel (left in the figure) operates according to the multidimensional paradigm, and data are visualized in tabular form. All facts displayed are equally preferred by the user, and they can even be characterized by different aggregation levels.

C. Navigation

Following the OLAP session paradigm, a user can change her view on the underlying datacube (i.e., formulate a new query) first of all by applying a standard OLAP operator to the tabular representation in the OLAP panel, which is done preserving the annotating preference.

To introduce the second navigation option we recall that, while the BMO result of a preference query only includes data that are either equivalent or incomparable according to the user preference, its overall result includes data that are differently ranked according to the user preference [4]. Navigation of the overall query results can be achieved by applying a new OLAP operator, called *next rank*, that descends the domination graph by moving to its next level, so that the next best-matching facts can be retrieved and analyzed.

V. DEMONSTRATION SCENARIOS

The demonstration will run on IPUMS, a public database storing census microdata for social and economic research, that includes a CENSUS datacube with five hierarchies, namely RACE, TIME, SEX, OCCUPATION, and RESIDENCE [5]. The CENSUS datacube includes about 10 millions facts.

The demonstration has three main goals: (1) show how effective preferences are in avoiding empty results and information flooding, and in exactly returning the facts of interest; (2) show how MYOLAP reduces the formulation effort to retrieve the preferred facts; (3) show how the double interaction level provided by MYOLAP makes visualization of complex results more intuitive and simplifies their navigation. In particular, some analysis sessions will be run to show that traditional OLAP queries without preferences can easily return too many or no data, which entails further effort for users to progressively refine their queries. Preference queries move such effort to the system: all the user has to do is express her preference criteria. Preference effectiveness is closely related to the expressiveness of the preference algebra. The demonstration will stress the expressiveness of base constructors —with particular emphasis on those related to hierarchies, that are specific to the multidimensional domain— as well as the distinguishing capability of WEST to handle preferences on both categorical and numerical domains. Finally, we will verify how the formulation and navigation efficiency is greatly

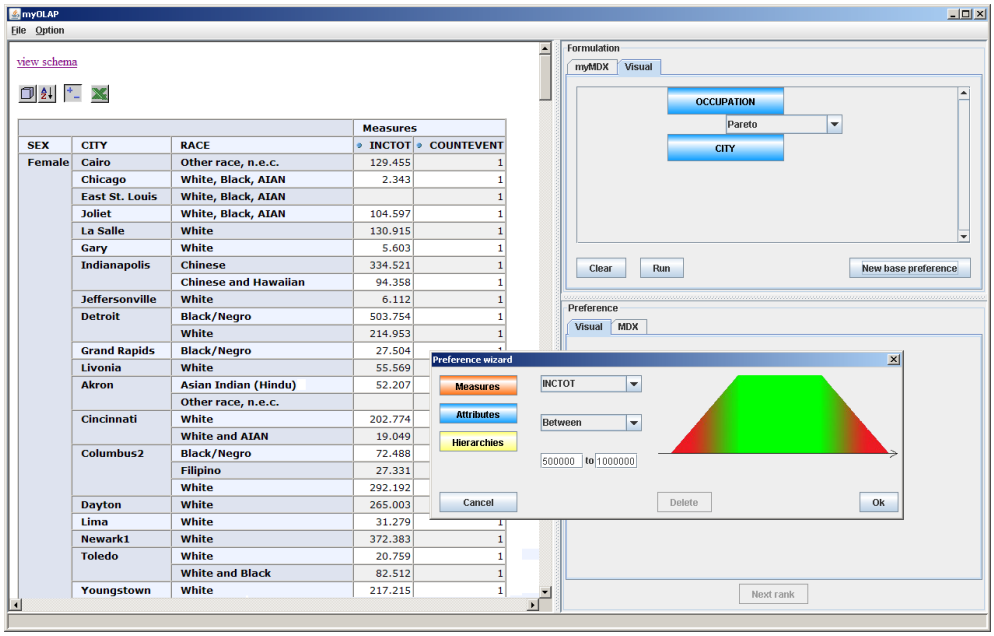

Fig. 2. Preference formulation

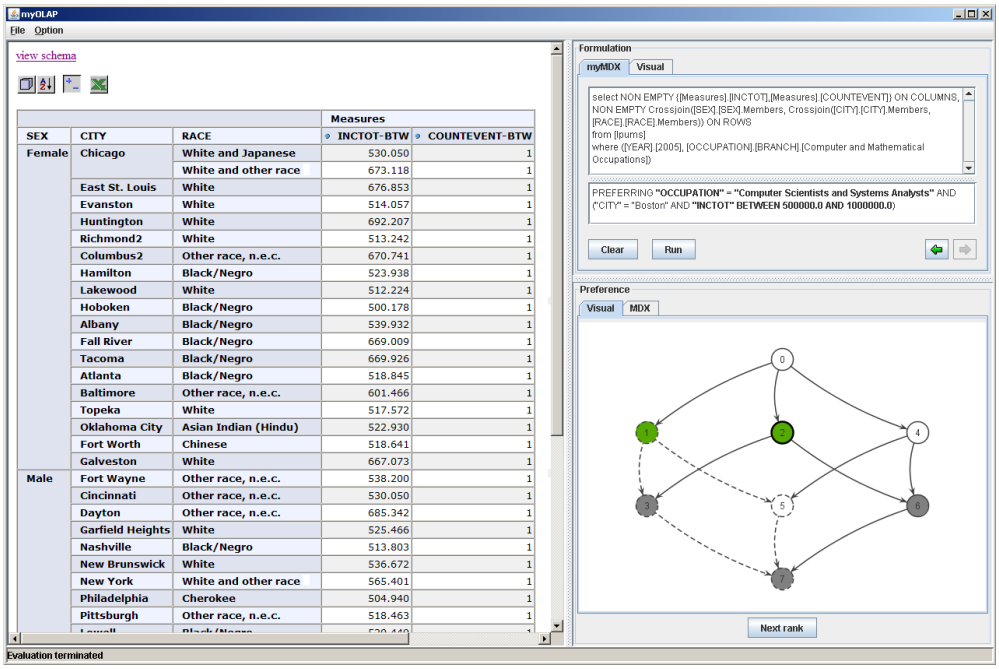

Fig. 3. Preference navigation

improved by the visual interface that, in line with the OLAP paradigm, allows users to directly interact with datacubes.

REFERENCES

[1] M. Golfarelli, S. Rizzi, and P. Biondi, "MYOLAP: An approach to express and evaluate OLAP preferences," *IEEE Trans. on Knowledge and Data Engineering, to appear*, 2010.

[2] Microsoft, "MDX reference," http://msdn.microsoft.com/ en-us/library/ms145506.aspx, 2009.

[3] T. Preisinger, W. Kießling, and M. Endres, "The BNL++ algorithm for evaluating Pareto preference queries," in *Proc. PREFERENCE*, Riva del Garda, Italy, 2006.

[4] J. Chomicki, "Preference formulas in relational queries," *ACM TODS*, vol. 28, no. 4, pp. 427–466, 2003.

[5] Minnesota Population Center, "Integrated public use microdata series," http://www.ipums.org, 2008.