Integration of the Viewpoint Mechanism in Federated Databases

Fouzia Benchikha
LIRE Laboratory, Department of Computer Science, University Mentouri of Constantine, 25000 Constantine, Algeria
E-mail: fbenchikh@caramail.com

Mahmoud Boufaida
LIRE Laboratory, Department of Computer Science, University Mentouri of Constantine, 25000 Constantine, Algeria
E-mail: boufaida@hotmail.com

Lionel Seinturier
LIP6 Laboratory, Pierre et Marie Curie University, 75252 Paris, France
E-mail: lionel.seinturier@lip6.fr

ABSTRACT
The opening of the databases towards the new applications of increased complexity requires to re-examine the way of designing the database conceptual schema. Indeed, it would be desirable that the schema can be a specification that takes account into several points of view. Each point of view represents an aspect of the data description that is held by an independent database known as partial. This process confers a decentralised vision of the conceptual schema. Various models and mechanisms of views even less of viewpoints were developed, in particular within the framework of the object oriented databases. We focus on the basic concepts presented by the two approaches for the general architecture of MultiViewDataBase system, a system of federated databases which supports the concept of viewpoint. This last, on the one hand, allows to control the difficulty of the complex system development. On the other hand, it contributes to the solving of the problems involved in the schema integration in federated database systems.

Keywords
View, ViewPoint, Federation, Databases, Schema integration.

1. INTRODUCTION
The exponential proliferation of types of DataBases (DBs) and their opening towards the new applications requiring the collaboration of several experts deserve to re-examine the way of considering the schema of a database. Indeed, in these applications, it is difficult to work out a single abstraction which is appropriate to all the participants of a project. Each expert, according to his field of knowledge and his objectives, focuses on certain aspects of the universe of discourse, which are not necessarily the same ones for another expert. Thus it would be desirable that the schema of a database can be a specification, which takes account several *points of view*, while keeping to each one its specificity and allowing the sharing and the exchange of information.

SAC 2001, Las Vegas, NV

In this paper, we focus on the viewpoint aspect for which the multiple description of the entities is systematized since the design. Thus, any viewpoint represents an aspect of the description and is held by an independent database known as "partial". Therefore, we adopt an approach of development based on a decentralized elaboration (distributed) of databases. These latter share a basic schema called the "referential". We propose the integration of the partial databases in a federated system, named Multi-View-DataBases (MVDB). This last makes it possible to ensure a complete and coherent management of data.

However, in the distribution of the databases, the problem of resource sharing arises with much acuity. Various approaches were proposed including the global schema integration approach [1, 27] and the federation approach [3, 26, 2, 15]. In this last approach, the sharing is provided through a total or partial integration of multiple schemes. Nevertheless, the problem of the semantic conflicts during the integration of the schema does not find a solution. These conflicts are due essentially to the differences in names, values and significance of the same data present in different local bases. Their control proves to be automatically difficult and requires the manual intervention of the designer. In our work, we show the contribution of the viewpoint concept in the solving of conflicts during the schema integration. Then, we present the general architecture of MVDB that exploits this concept by supporting components conveying a semantic inherent to the resolution of the conflicts.

Our paper is organised as follows. In section 2, we present, briefly, the concept of views in databases and we argue the principal motivation and the introduction of the viewpoint concept in mainly object oriented databases. Section 3, deals with the management of distributed databases. We show that the federation scheme suits our objectives. An overview of the approach used in MVDB is presented in section 4.This approach is followed by its general architecture in section 5. Section 6 concludes our work.

2. THE VIEWPOINT MECHANISM IN DATABASES
In the computer science area, the concept of viewpoint, also called perspective, takes various significance according to its access in the different fields in particular in software engineering [9], in knowledge representation [12, 24], in databases [11, 20], in complex systems modelling [10], etc. Generally, we are

interesting in the viewpoint concept since one designs multi-users applications. These latter require either the co-operation of several tools as in software engineering, or the co-operation of several experts with each one its interests and its knowledge. For instance, the case of the advanced applications of CAD handled in the databases management systems or in knowledge presentation systems. We call this mode of perception and description of the real world's entities " multiple description". This one is defined as being the fact of conferring to a universe of discourse several partial descriptions as each one describes it in a given point of view. Various partial descriptions are complementary and together provide a complete description of the real world's entities.

In the database field, the concept of viewpoint was mainly studied within the framework of object oriented databases. The choice of the object oriented paradigm is primarily due to the basic concepts that it offers, in particular, a great power of expression, a better reutilisability and evolution of objects. These concepts correspond to the new aspirations quoted above. We distinguish two approaches for which the concept of viewpoint was considered. The view approach and the viewpoint approach.

2.1 View approach

Various techniques of views were developed in databases. We note the view model of Bertino [6], the view model of Abiteboul [14, 1] and the view model of Rundensteiner [25]. In these works, a view is considered, in general, like a request reification. Its result is a view-schema (or a view-class). This one provides to the user part of the global schema, a kind of *viewpoint* on the description of its entities (see figure 1). So a view is an external schema defined with the aim of adapting an existing structure to new needs [1]. This process does not substitute the viewpoint concept as described above. The view and viewpoint mechanisms concern respectively the exploitation step and the design one. A viewpoint must directly be related to the objects' description and confers new properties to them, not necessarily derived or calculated from others already existing.

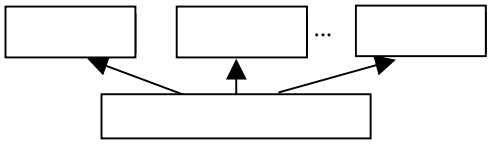

Figure 1. View Approach

2.2 Viewpoint approach

Let us notice few works on the integration of the viewpoint concept in the data models. CEDRE model [18] and CROME system [11] meet the need of this concept for multiple data description. They propose the elaboration of the conceptual schema according to various viewpoints that support different abstractions of the same universe of discourse. Among the essential objectives of such an approach**,** showed in figure 2 **,** there are :

- Conferring a decentralised vision of the conceptual schema. This facilitates the parallel work of several designers and allows a certain independence between various descriptions.
- Supporting an exchange of information between partial descriptions of the schema, which are not isolated.
- Endowing the universe of discourse with a first specification on which the partial representations base. ·
- Ensuring the coherence of the global schema described by several viewpoints.

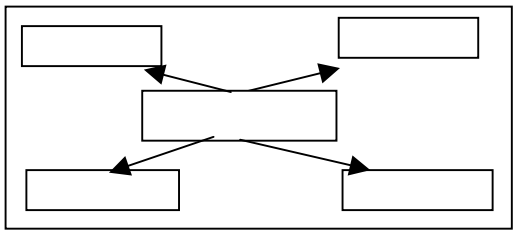

Figure 2. Viewpoint approach

In our work, we are particularly interested in this approach that offers a great power of structured data and leads to a possible distributed development of complex applications. However, if the viewpoint approach allows one to describe multiple data, how can it be wittingly managed and exploited?

3. DISTRIBUTED DATA MANAGEMENT

The distributed information systems become increasingly important because of the needs for organizations and the advent of the advanced techniques of the networks management. In the context of the databases, we present two approaches most commonly used for the achievement of distributed data : the schema integration and the federation scheme. We shortly present them and discuss each ones' opportunities in the context of our work.

3.1 Global Schema integration

Almost all the studies on the global schema integration were carried out in the Entity-Association model [4], in the relational model [7] or in the ERC model and its successor ERC+ closer to the object model [16, 23].

The schema integration concerns the database design step. All the developed methods are based *on various schemes of users* to elaborate a global schema of a database (see figure 3). This integration step should remove all the inconsistencies, the errors and the redundancies resulting from these schemes.

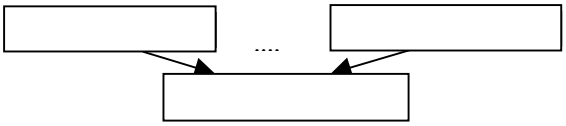

Figure 3. Global schema integration

The schema integration can entirely be led by hand. However, if the number of views schema increases, the task becomes almost impossible. The designer carrying out the integration deals with the resolution of the conflicts, which cannot be treated automatically. In this approach, the data sharing is obtained via the total integration of data. We notice that the objective is not to take into account the various descriptions to build a conceptual schema relatively structured with the various designers. The conceptual schema out of the integration is not different in its form from that resulting from a direct design. The specificity relating to the designers' views are lost at the conclusion of the design by integration. Thus, this approach does not meet the needs for structuring shown in section 2. We will see that the federation

approach, on the other hand, makes it possible to preserve the multiple descriptions of data.

3.2 The federation

On the contrary to approach, a federated database does not support a global schema. Its principal aim is to ensure the different databases autonomy and to preserve their independent management and handling and their administration. The architecture adopted by several works on federated databases is given in [26]. Five levels are to be distinguished (see figure 4).

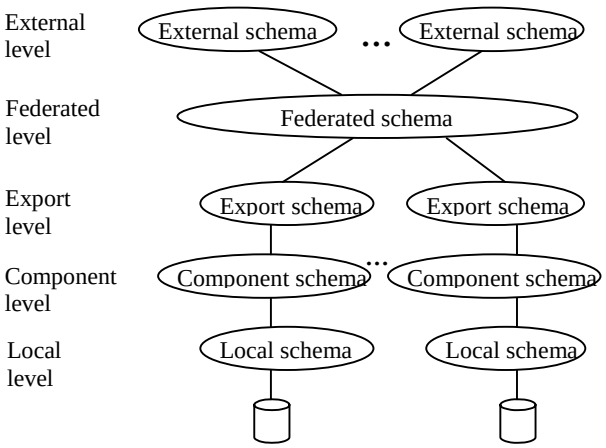

Figure 4. Federated database architecture

On the lower level, the local schemes of the existing databases appear in the federation. These schemes are translated, at the component level, in a common data model. The export level is composed of the exported schemes. These latter represent the databases' parts that are relevant for the federation. A complete integration of the exported schemes is carried out at the federated level. Finally, the external level provides relevant information for one of its users. However, all the levels are not necessarily present. For example, the component level is not essential if the local schema is already expressed in the common model.

In this approach, the information sharing occurs through import/export schemes. The participating databases have to show a part of their whole schema that they export to the federation by storing them in the federated dictionary. Any total integration is then previously done. This provides some flexibility in terms of autonomy of the participating databases. We adopt this method for the achievement of MVDB.

4. APPROACH OVERVIEW

In MVDB, the suggested design approach of the conceptual schema has multiple description based on that presented in 2.2 (see figure 5). We adopt the object model which fits well to this approach for its great power of data expression and structuring. Thus, the entities of the universe of discourse are described in a multiple and complementary way by several partial DBs schemes that share a basic description known as the referential. Each partial schema describes an aspect of the data or part of the data of the referential. Thus, an object can have a representation in one or more viewpoints. Then, it can be an instance in one or several databases.

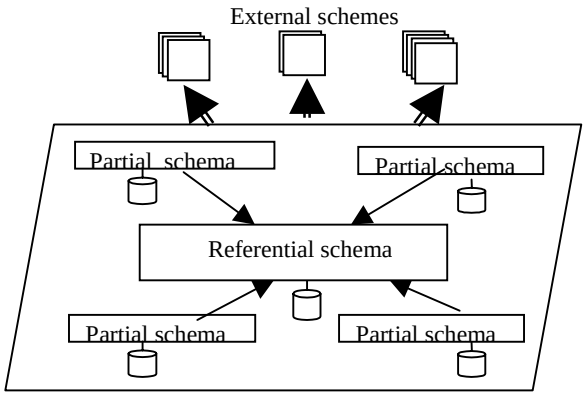

Figure 5. Multiview Approach

At the exploitation, an external schema keeps all the specificity of the multiple description of data as shown in figure 5. The user can express his needs in terms of viewpoint on data. The information sharing is done at the demand.

Our contribution consists in the integration of the viewpoint mechanism in a federated database. We take advantage of this mechanism to solve the conflicts usually met in the distributed data management. As a matter of fact, unlike the traditional approach (mono viewpoint) where the integrity constraints are defined on the global schema, we distinguish in the multiviewpoints approach two types of constraints :

- Local constraints : they contribute to ensure the local coherence of the entities in a database and independently of the other bases. There are many forms of integrity constraints that are inherent to an object database. Among the most frequently ones found there are, the referential integrity constraints, the cardinal constraints and the static integrity constraints on the entities' properties [8, 13, 5, 19]
- Global constraints : they contribute to ensure the global description coherence of the entities according to several viewpoints. Indeed, if the local coherence is ensured, what happens to the global schema, result of the integration of several partial schemes?.

If local constraints are apprehended, it is difficult to take into account the global constraints. Classically, the principal conflicts met during the schema integration are the names, the semantic and the structural conflicts. In our work, these latter can be solved by the viewpoint mechanism.

- Name conflicts : traditionally, the solving of this type of conflicts is done by assertions specifying the synonyms and the homonyms. In our context, the existence of the referential solves any conflict coming from a problem of synonymy. Thus, all the common properties are described by the referential schema. On the other hand, a conflict coming from homonyms is solved by the viewpoint mechanism itself. As a matter of fact, two distinct homonymous constructions can be differentiated by prefixing them, for example, by the name of the partial schema.
- Semantic and structural conflicts : they are of a weak or no presence, in a database schema designed according to various viewpoints. Nevertheless, each partial schema describes an aspect of the data semantically different from the other descriptions. In addition, the referential allows a representation, and by the same way a unified structure of the real world entities that will have different descriptions according to different viewpoints .

However, within the framework of a multiview database, other types of conflicts have to be distinguished. Let us consider, for example, the following cases :

- Mutual exclusion between partial DBs : when the entities' description by a partial schema compromises their description by another partial schema.
- Interdependency between partial DBs : when the partial schemes contain linked properties.
- Referential integrity between partial DBs : when the creation (possibly suppression) of a database entity requires a preliminary creation (possibly a suppression) of one (or many) entity (ies) of another databases.

We propose the storage of these constraints in a metadatabase that will be used for the solving of the schema integration conflicts during the federated database exploitation.

5. A GENERAL ARCHITECTURE OF MULTIVIEWDATABASE

The proposed Architecture for MVDB is based on the federation. The databases that take part in it show multiple descriptions of the same universe of discourse. Each description is held by a partial schema. In the following, we will show how the autonomy of each base is preserved. This architecture is made up of three levels : the local level, the federated level and the external level (see figure 6). The local level carries the partial DBs, the federated level allows to manage of the federated database whose services will be exploited by the external level.

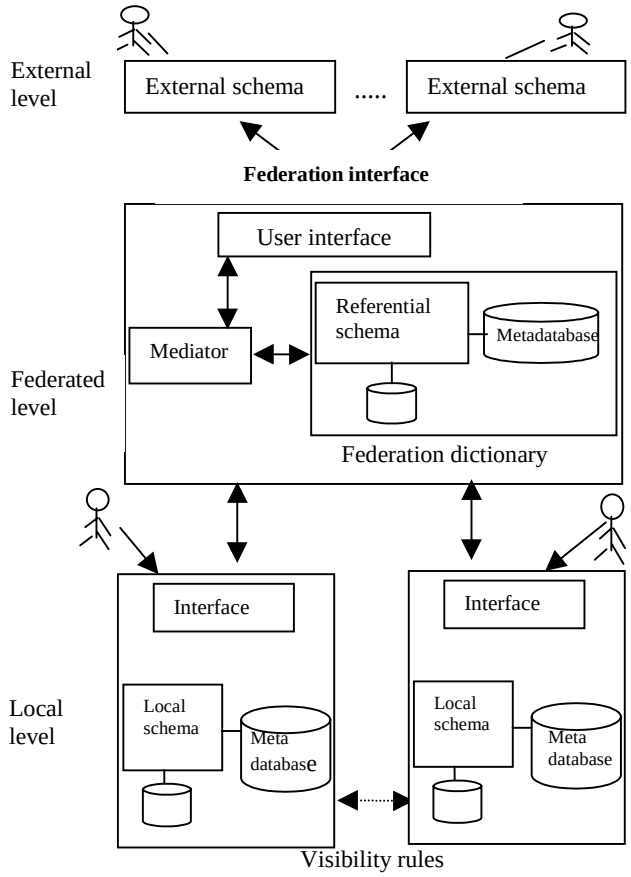

Figure 6. General architecture of MVDB

5.1 General description

The kernel of MVDB is the management module of the federated data. It is essentially made up of a user interface, a mediator, a metadabase and a referential. The common data model used is ODMG [21, 22]. All the present schemes in this architecture, ie, local schema, federated schema, basic schema (referential) and external schema, are based on ODMG. The heterogeneity problem is then not dealt with here. The uniformity of the used data model is important in particular for the management of both the persistence and the identity of the objects in the federated base.

At the local level, the partial databases are interdependent. This interdependence is expressed via visibility rules stored in metadatabases. These ones hold the communication between partial DBs. Note that each base can be individually handled. In the following, we will see that the conceptual schema of a local database presents a complete description of the entities (or the whole of the entities) of the referential according to a particular viewpoint. We will detail each component of our architecture.

5.2 Basic components description.

The federation interface and the viewpoint databases are the basic components of our architecture. We present them in more details.

- Federation interface : it is the kernel of MVDB which ensures the federated database management. It essentially contains :
  - A user interface that permits the communication with the users. Let us notice that these latter have a single system image regardless of the number of particular databases.
  - The federation dictionary contains the referential and the metadatabase. The referential is a basic schema that presents the export schemes. Thus, any database taking part in the federation imports a schema derived from the basic one and extend it with a particular description proposed according to a given viewpoint. The derived schema can concern all the basic schema if the partial description relates to all the entities of the universe of discourse. The metadatabase is a component that has an important role in the distributed data management. It stores two kinds of information : information relating to the types of data supported by the different viewpoint databases and information on the global constraints that are described in section 4 for the solving of the integration conflicts during the exploitation. The metadatabase is used by the mediator in dealing with the users requests.
  - The mediator is a processor dealing with the treatment of the users' requests. At a request reception, it checks its coherence basing on the information of the metadatabase. For example, any request that asks two exclusive databases will be rejected. In the same way, the mediator builds an execution plan of the request, if an order in querying DBs is necessary. Then, the mediator breaks up the user request into queries to individual sources that would return the data of interest. Finally, it merges data returned by the sources. Data integration, in our case, does not cause conflicts because the requests are primarily checked.

- Viewpoint databases : each database holding a particular entities' description of the universe of discourse is autonomous. However, its local schema presents a complete data description according to a viewpoint. It contains the imported part of the referential which comes to be added to the local description. Moreover, a metadatabase is associated to any database in order to ensure its autonomy of communication with the other bases. This

metadatabase is part of that related to the referential. Thus, two kinds of applications can coexist in our architecture. Global applications interested in the whole federated base and the local applications acting the particular DBs.

6. CONCLUSION

In this paper, we have presented a general architecture of a federated database that allows the management of multiple descriptions of the same universe of discourse. The viewpoint mechanism can be of an undeniable contribution for a distributed design of complex databases. On the other hand, it brings satisfactory solutions to the various schema integration conflicts.
Indeed, it could be interesting to generalize this approach, starting exclusively from an already existing set of DBs. Then the aim would be to find the common referential, the viewpoint databases and their interdependency. In addition, it would be interesting to develop or to extend a query language like OQL for dealing with the multi-viewpoint aspect of the objets. These last can then be query according to one or several viewpoints.

7. ACKNOWLEDGMENTS

This research was supported in part by the CMEP project number 99MDU454.

8. REFERENCES

[1] S. Abiteboul and A. Bonner, "Objects and views ", Proceedings of SIGMOD, pp 238-247, May 1991.

[2] R. Alonso and D. Barbara, "Negotiating Data Access in Federated Database Systems", IEEE Conference on Data Engineering, pp 56-65, 1989.

[3] M. Andersson, Y. Dupont, S. Spaccapietra, K. Yétongnon, M. Tresch and H. Ye, "FEMUS: A Federated Multilingual Database System", Advanced Database Systems, N.R. Adam and B.K. Bhargava (Rditors), Springer-verlag, New York, 1993.

[4] C. Batini, S. Ceri and S. Navathe, "Conceptual Database Design. An Entity-Relationship Approach", The Benjamin/Cummings Publishing Compagny, 1992.

[5] C. Bauzer Modeiros andP. Pfeffer, "Object Integrity Using Rules", Proceedings of ECOOP's91, P. America (Editors), Springer-Verlag, pp 219-230, Geneva, Switzerland, July 1991.

[6] E. Bertino, "A View Mechanism for Object-Oriented Databases", In 3rd International Conference on EDTB'92, pp 136-151, Australia, March 1992.

[7] J. Biskup and B. Convent, "A Formal View Integration Method", Proceeding of the ACM SIGMOD International Conference on Management of Data, pp 141-156, 1986.

[8] T. Bouaziz, "Classification des contraintes d'intégrité dans les bases de données orientées-objet", Ingénierie des systèmes d'information, vol 3 n°6, Hermès (Editeurs), pp 713-737, 1995.

[9] N. Carn, "représentation Orientée Objet de Système Opérationnel avec application au domaine spacial", Thèse de l'INP, Toulouse, October 1992.

[10] P. J. Charrel, D. Galaretta, C. Hanachi and B. Rothenburger, "Multiple Viewpoints for the Development of Complex Software", Actes d'IEEE International Conference on Systems, Man and Cybernetics, pp 556-561, October 1993.

[11] L. Debrauwer, "Des vues aux contextes pour la structuration fonctionnelle de bases de données à objets en CROME", Thèse de Doctorat, Université des sciences et technologies de Lile, France, 1998.

[12] L. Dekker, "FROME : Représentation Multiple et Classification d'Objets avec Points de Vues", Thèse de Doctorat, Université des sciences et technologies de Lile, France, 1994.

[13] X. Delannoy, "La cohérence dans les bases de données", Rapport de recherche RR 936 I, IMAG, Octobre 1994.

[14] C. S. Dos Santos. "Un mécanisme de Vues dans les systèmes de gestion de bases de données", Thèse de Doctorat d'Université, Univesité Paris-Sud, Orsay, Novembre 1995.

[15] D. Heimbigner and D. McLeod, "A Federated Architecture for Information Systems", ACM Transactions on office Information Systems,vol 3 n°3, pp 253-278, July 1985.

[16] J. Larson, S. Navache and R. Elmasri, "A Theory of Attribute Equivalence in Databases with application to Schema Integration", IEEE Transactions on Software Engineering, vol 15 n°4, April 1989.

[17] W. Litwin and A. Abdellatif, "An overview of the multi-database manipulation language MDSL", Proceedings of the IEEE, vol 75 n°5, pp 621-632, 1987.

[18] H. Naja, "CEDRE : un modèle pour une représentation multi-points de vue dans les bases d'objets", Thèse de Doctorat, Université Henri Poincaré, Nancy 1, July 1997.

[19] R. Nassif, Y. Qiu and Zhu, "Extending The Object-Oriented Databases: analysis , Design & Construction (DS-4), R.A. Meersman, W. Kent and S. Khosla (Editors), IFIP, Elsevier Science Publishers, 1991.

[20] G. T. Nguyen and D. Rieu, "Database Issues in Object-Oriented Design", Actes de la IVe International Conference TOOLS, pp 73-86, Paris, March 1991.

[21] R.G.G. Cattel (Editors), T. Atwood, J. Dubl, G. Ferran, M. Loomis and D. Wade, "The Object Database Standard: ODMG-93, Morgan Kaufmann, 1993.

[22] R.G.G. Cattel, D. Barry, D. Bartels, M. Berler, J. Eastman, S. Gamerman, D. Jordan, A. Springer, H. Strickland and D. Wade, "The Object Database Standard: ODMG-2.0, Morgan Kaufmann, 1997.

[23] C. Parent, H. Rolin, K. Yetongnon and S. Spaccapietra, "An ER Calculus for the Entity-Relationship Approach", Toronto, pp 361-384, 1989.

[24] D. Rieu, G. T. Nguyen, A. Culet, J. Escamilla and C. Djeraba, "Instanciation Multiple et Classification d'Objets", VIIèmes Journées Bases de Données Avancées, Lyon, 1991.

[25] E. Rundensteiner, "Multiview: a Methodology for supportind Multiple Views in Object-oriented Databases", Proceeding of the 18th VLDB Conference, pp 187-198, Canada, 1992.

[26] A. Sheth and J. Larson, "Federated Database Systems for Managing Distributed Heterogeneous, and Autonomous", ACM Computing Surveys, vol 22 n°3, pp 183-236, September 1990.

[27] J. M. Smith, P. A. Bernstein, U. Dayal, N. Goodman, T. Landers, K. W. T. Lin and E.Wong, "Multibase-Integrating Heterogeneous Distributed Database Systems", AFIP, National Computer Conference, pp 487-499, 1981.