

# Self maintenance of materialized XML views with non-cooperative data sources

Virginie Sans, PhD Student, *ETIS/CNRS Laboratory, France*

**Abstract**— Integrating miscellaneous data belonging from web sources has always been a topic of interests. Mediation architecture is mainly devoted to this task, because it enables to query those sources and to federate the results in order to give an answer to another application like a web-service. However the conditions on the network is not always at its best, therefore the results are sometimes stored as materialized views. XML is used in these architectures to present an uniform answer to the final user. The issue in this context is to maintain consistency of the views when underlying data sources has been updated. Our work is related to the self maintenance of materialized XML views when the data sources are non-cooperative. The first step of our approach is to detect and to identify the sources updates. Our views are computed using the XAlgebra, an algebra based on XQuery requests which handle XML results as tabular datas. We propose an approach for maintaining XML views in this context, mainly based on an extension of this algebra. To that purpose, we had new operators and identifiers. Our process is based upon a partial recovery of underlying sources which is computed over the view itself..

Using XML views is a standard way to integrate data from heterogeneous sources. When these data are distributed, these views can be computed in the context of a mediation architecture [Widerhold, 1992]. The two main components of a mediation architecture are the mediator and wrappers. A wrapper is dedicated to the translation of the query given by the mediator into the natural language of the data source. The mediator collects the wrapper's answer and give the answer into a common language which is in our case, XML. In order to handle with network's delay, and for the sake of efficiency, views are materialized at the mediator level. When a data source is being update, the view should be maintain.

Previous works on relational data have already established that an incremental maintenance is better than a recomputation process [Gupta, 1995]. Incremental maintenance consists in the recomputation or the update of a smaller subset of data that are not consistent. Previous publications have been the topic of interests in the context of relational and object views. However, semi-structured data have their own specificities, thus these results can not be applied in this context.

We assume in our work, that data sources are non cooperative, thus they didn't give us any information about their related update operations. We must, first detect when data sources updates have been proceed and what has been proceed : insertion, deletion, modification. Our view computation process involves the use of an algebra dedicated to XML queries called XAlgebra [Dang, 2003]. We introduce new concepts on this algebra. On one hand, we introduce identifiers called XTid to annotate data in order to track data sources among the view handling process. On the second hand, we propose a new operator that enables to extract from a view data belonging from a particular source. This operator called XRecover is strongly related to our identifiers.

The first step of our work is to detect that an underlying source has been updated. To that purpose, we use a comparison between a timestamp stored in the wrapper's logger and the actual time, and also a comparison on checksums computed on the data. When we find that  $i$  source has been updated, we applied the XRecover operator to the view to partially restore the old version of the  $i$ . In order to identify the update operation, a Diff algorithm is performed on these two versions of  $i$ .

Given the Delta instructions of the Diff algorithm, the XQuery request used to compute the view, we give for each operation (union, join, cartesian product...) the incremental rules for the maintenance of materialized views.

Our approach has already given promising results. Our perspectives is to applied these approach to hypermedia services to enhance their availability or in the context of mobile sources.

## REFERENCES

- [Widerhold, 1992] Widerhold G., Mediators in the Architecture of Future Information Systems, IEEE Computer, vol.25, N.3, pp.38-49, 1992
- [Gupta, 1995] A. Gupta and I. S. Mumick. Maintenance of materialized views: problems, techniques, and applications. In Bulletin of the Tech. Com. on Data Eng., 18(2), pages 3-18, 1995.
- [Sayed, 2002] M.El-Sayed, L.Wang, L.Ding, E.A Rudensteiner. «An Algebraic Approach for Incremental Maintenance of materialized XQuery Views», Proceedings of the 4th Int. Workshop on WIDM02, 2002
- [Dang, 2003] T.T.Dang-Ngoc, Fédération de données semi-structurées avec XML, Thèse de doctorat, Université de Versailles Saint Quentin, 2003