Update-driven Inference in Deductive Databases

Rainer Manthey
European Computer-Industry Research Centre (ECRC)
Arabellastr. 17, D-8000 München 81, Germany
(rainer@ecrc.de)

(Abstract of a one-hour tutorial)

1 Motivation

When thinking of deductive databases, most people will probably associate this notion rather closely with problems of query processing: deductive database research is essentially concerned with formalization and implementation of information retrieval in terms of logical inference. This reaction - which seems to come quasi automatically - is probably due to the fact that the majority of papers on deductive database aspects written in the past has been concerned with query processing. Moreover, this issue has been in the center of interest in database research anyway, not only in the deductive context. In a similarly straightforward manner do we associate rules and query answering. We all would probably agree that activating rules is essentially done in order to infer answers to queries: deduction in databases tends to be regarded mainly as query-driven inference.

We would like to question this often rather unconsciously accepted association. We claim that there is in fact a second main inference mode in deductive databases which we regard as equally relevant and equally important, but which up till now has not always been granted equal rights. Whenever one is determining the effects an update of a certain base relation has on the extension of a rule-defined relation depending on the respective base relation, we are in fact performing deduction as well. We would like to call this kind of deduction update-driven inference.

In connection with deductive databases, this “other” mode of inference has been investigated mainly for integrity checking purposes up till now [BMM91, Dec86, KSS87, Kue91, JJ91, LST87, Oli91, VBK91]. But there is a much wider class of potential applications of update-driven inference which have been mostly studied outside the mainstream of deductive database research. Maybe the most important of these applications is the maintenance of materialized views, i.e. of rule-defined relations the extension of which is
explicitly stored at all times in order to avoid costly derivations at query time [Han87].
Both integrity checking and refreshment of materialized views are special cases of mon-
toring conditions over changing databases. So-called monitors and alerters have been
suggested in the past [BuCle79]. General triggering facilities are re-gaining particular
interest at the moment in connection with production rule systems and active databases
[CW90, CW91, Day88]. All these features of databases rely on condition monitoring. If
combined with rule-based knowledge representation in the sense of deductive databases
they all require powerful facilities for update-driven inference as outlined above in order
to incrementally determine changes affecting the monitored conditions [QW91].

2 A Deductive Approach to Update-driven Inference

At the moment one can observe particular interest in problems related to condition
monitoring in the database research community. Respective studies are mostly conducted
in the context of active databases. The emphasis here is on methods for efficient and
controlled sequencing of database operations. In the context of condition monitoring,
such sequences are initially triggered by base relation updates and - through a chain of
intermediate updates on rule-defined relations - eventually lead to “terminal” actions like
rejection of the update, compensating updates or user interaction (in case of an integrity
violation), automatic refreshment of materialized views, report generation, alerting and
so on.

If regarded from a deductive point of view, it is not the actions directly or indirectly
Triggered by an update which are of primary interest. The deductive approach we are
presenting here is much more concerned with determining the effect of an update on
certain monitored relations by means of an inference process. Thus we are concerned with
reasoning about updates rather than actually performing chains of updates triggering
each other. Once this reasoning process has terminated, some meta-information reflecting
those effects of the respective update that have to be known due to the existence of certain
monitored conditions has been temporarily materialized. In a second phase these meta-
data are interpreted in a straightforward manner and give rise to the same terminal
actions as obtained by an active system.

The advantage of such an approach is that well-formalized logical means of specification
and deduction can be applied to the update propagation problem thus facilitating to
achieve sound and correct methods. Moreover, implemented and optimized inference
algorithms developed for other purposes can be applied for update-driven inference as
well. Of course there is a close correspondence between sequences of actions triggered by
an active system and sequences of inferences performed by a deductive system in reaction
to an update. Thus results obtained in the one area are relevant for the other, and vice
versa.
3 Principles of Update-driven Inference

Update-driven inference comprises two deductive processes. On the one hand update propagation has to be reflected by suitable inference steps. This process is by its very nature a sequence of forward reasoning steps, inferring induced updates bottom-up from base relation updates. In order to make sure that only those deduction sequences are performed that will finally reach a monitored condition, a second inference process is required which reflects a top-down propagation of monitored conditions by backward reasoning [GrLue90]. This monitor propagation phase is essential for controlling update propagation in view of avoiding serious inefficiencies (except in trivial cases). Monitor propagation is often neglected, but ought to be considered as an indispensable part of any approach to update-driven inference. Thus we are in fact dealing with two antagonistic processes - one top-down, one bottom-up - where the one process controls how the meta-data to be produced by the other are generated.

In our tutorial we will particularly elaborate on the main parameters to be considered while reasoning about update propagation. Though being a forward reasoning process in principle, update propagation exhibits a couple of important particularities which are essential for the correctness and soundness of any method. We discuss the main phenomena and the basic techniques required for controlling them. These techniques can be applied in various combinations and variations, thus constituting the basic repertoire of choices from which the many individual methods proposed in the literature are composed.

Finally, we will discuss the close relationships which can be observed between update- and query-driven inference. Deducing the answers to a particular query from base relations and rules can be viewed as an inference process consisting of two subprocesses as well. Whereas answers are generated bottom-up, queries have to be propagated top-down in order to avoid generation of derivable facts which do not contribute to answering the respective query [Bry90, Man90]. Query propagation controls answer generation in very much the same way as monitor propagation controls update propagation. Query-driven inference can also be regarded as a process of reasoning about a query. Meta-data are deduced (such as intermediate results and “magic” facts, e.g.) which finally give rise to some action (namely presentation of an answer to an interactive user, or feeding answers into a program consuming them in whatever manner). The research community just begins to understand and exploit these similarities. We ultimately expect a rather dramatic effect on the architecture of knowledge base management systems once the strong commonalities between update- and query-driven inference are properly reflected in terms of software organisation.

4 Conclusion

The main purpose of this contribution is to introduce a slightly unconventional view of deduction in rule-based databases. Contrary to the commonly accepted assumption that the main motivation for performing inference over a rule base is query processing, we claim that an activation of rules as a response to an update is equally important and motivates an alternative mode of inference. We focus on the basic principles of update-
driven inference the understanding of which is essential for motivating and developing more refined methods for particular purposes, such as integrity checking, for example. We outline the strong similarities between update- and query-driven inference and propose an exploitation of these similarities for design and implementation of KBMSs.

Acknowledgement: The work underlying this tutorial has been performed over the last five years in close collaboration with François Bry, who deserves explicit credit for his substantial contributions to the topic.

References


[JJ91] M. Jeusfeld and M. Jarke: "From Relational to Object-Oriented Integrity Simplification", Proc. 2nd Intern. Conf. on Deductive and Object-Oriented Databases (DOOD), München, Dec. 1991


[Oli91] A. Olivé: "Integrity Constraints Checking in Deductive Databases", Proc. 17th Intern. Conf. on Very Large Databases (VLDB), Barcelona, Sept. 1991
