XPath Rewriting Using Views: The More the Merrier

Bogdan Cautis

Telecom ParisTech, Paris, France
cautis@telecom-paristech.fr
http://www.telecom-paristech/~cautis

We report in this talk on recent results on richer classes of XPath rewritings using views.

The problem of equivalently rewriting queries using views is fundamental to several classical data management tasks. Examples include query optimization using a cache of materialized results of previous queries and database security, where a query is answered only if it has a rewriting using the pre-defined security views.

While the rewriting problem has been well studied for the relational data model, its XML counterpart is not yet equally well understood, even for basic XML query languages such as XPath, due to the novel challenges raised by the features of the XML data model.

We have recently witnessed an industrial trend towards enhancing XPath queries with the ability to expose node identifiers and exploit them using intersection of node sets (via identity-based equality). This development enables for the first time multiple-view rewritings obtained by intersecting several materialized view results. We present both theoretical and practical results on view-based rewriting using multiple views. First, we characterize the complexity of the intersection-aware rewriting problem. We then identify tight restrictions (which remain practically interesting) under which sound and complete rewriting can be performed efficiently, i.e. in polynomial time, and beyond which the problem becomes intractable. As an additional contribution, we analyze the complexity of the related problem of deciding if an XPath with intersection can be equivalently rewritten as one without intersection or union.

Then, going beyond the classic setting of answering queries using explicitly enumerated view definitions, we report on results on the problem of querying XML data sources that accept only a limited set of queries. This is motivated by Web data sources which, for reasons such as performance requirements, business model considerations and access restrictions, do not allow clients to ask arbitrary queries. They instead publish as Web Services a set of queries (views) they are willing to answer.

Querying such sources involves finding one or several legal views that can be used to answer the client query. Services can implement very large (potentially infinite) families of XPath queries, and in order to compactly specify such families of queries we adopt a formalism close to context-free grammars (in the same spirit in which a potentially infinite language is finitely specified by a grammar).

X. Du et al. (Eds.): APWeb 2011, LNCS 6612, pp. 438–439, 2011.

We say that query Q is *expressed* by a specification (program) $\mathcal{P}$ if it is equivalent to some expansion of it. Q is *supported* by $\mathcal{P}$ if it has an equivalent rewriting using some finite set of $\mathcal{P}$'s expansions. We present our results on the complexity of expressibility and support and identify large classes of XPath queries for which there are efficient (tractable) algorithms.

This survey mainly summarizes two recent papers, which are joint work Alin Deutsch, Nicola Onose and Vasilis Vassalos.