Piotr Cybula
PhD Thesis
November 2010

Cached Queries as an Optimization Method in the Object-Oriented Query Language SBQL

Summary

The Ph.D. thesis focuses on the development of robust methods and data structures for query optimization for object-oriented databases. The thesis has been developed in the context of the Stack-Based Approach (SBA), a theoretical and methodological framework for developing object-oriented query and programming languages. The developed query optimization methods are based on the corresponding Stack-Based Query Language (SBQL). SBA is well documented, has strong theoretical fundamentals and practical solutions for developing modern object-oriented database management system (OODBMS). The orthogonality and very clear semantics of SBQL constructs enable large optimization opportunities.

Query optimization is a very wide issue. There are many different solutions for improving of query execution performance. The thesis focuses on one of them – an optimization by caching the results of previously executed queries (so-called cached, stored, materialized queries). Other main optimization method are many rewriting optimizations and data indexing. From the conceptual point of view transparency is the most essential property of a cached query. It implies that programmers need not to involve explicit operations on cached results into an application program. Caching of query results yields relatively most improvement in a query evaluation performance, i.e. significantly decreases the time of an anticipation for a response from database management system. The main reason is that receiving the results from a cache for previously performed query, instead of its consecutive reevaluation, is much quicker than time-consuming data exploration. Additional advantage of the query caching method is that reuse time of cached results is independent of query type, its complexity and current database state.

On the other hand there are some costs of the result materialization. Firstly, some memory resources are necessary for the cache storing the queries and their results. Secondly, the optimization method needs some time for: storing in the cache queries, their results together with proper structures for maintenance purposes, recognizing the usability of currently materialized results for new queries, removing some rarely used cached queries in order to optimal cache utilization and finally, updating of the cached results after database changes. Our work proofs that these costs are in many cases significantly less than cost of the traditional full query evaluation, so we proof thesis of cached queries being an optimization method.

A key aspect concerning the development of database query optimization methods is preservation of original query semantics. Consequently, for the designed optimization methods we have determined necessary rules and data structures in a context of the assumed object data model and the SBQL query language. We have developed:

- a data structures and algorithms for the query cache registry dealing with storing all necessary data of cached queries and smart query searching;
- a data structures and algorithms for cache maintenance and optimal utilization of assigned resources by removing rarely used results;
- query normalization and decomposition methods for reuse of cached results for semantically equivalent but syntactically different queries, raising the probability of cache utilization;
- algorithms for optimal queries selection and rewriting new queries with use of cached results;
- a data structures and algorithms for maximum reduction of cardinality of cached queries set which were influenced by a database update and should be corrected;
- algorithms for automatic cache synchronization with current state of a database.

A significant part of algorithms and solutions developed in the thesis have been verified and confirmed in the prototype module implementation of caching optimizer as a part of ODRA (*Object Database for Rapid Application development*) OODBMS project serving object-oriented query language SBQL. ODRA is a platform for modern object-oriented web and grid applications, and is developed by a group of scientists and IT students. The prototype implementation allowed for performing several experimental tests of the optimization results receiving significant improvements of query evaluation performance even by two orders of magnitude (less than 1% of the full query reevaluation time).