SUMMARY:

Charles Garrod, Amit Manjhi, Anastasia Ailamaki, Bruce Maggs, Todd Mowry, Christopher Olston, and Anthony Tomasic.
Scalable query result caching for web applications.
In *Proc. of the VLDB Endowment*, pages 550-561, 2008.

DATE: 18 January 2010

The objective of the work presented in this paper is to improve the scalability of web applications by reducing the load of backend database systems. The proposed approach, named as Ferdinand, is query result caching among several proxy servers, so that users connect directly to proxy servers rather than the centralized server. This approach faces two limitations. The first one is low cache hit rates which results in limited scalability, and the other one is inefficiency in consistency maintenance of query result caches. For the first limitation, Ferdinand takes advantage of cooperation among proxy servers for finding the query result, and for the second one, it uses a highly scalable publish/subscribe system.

Each proxy server contains a cache consistency module, and is a member of a Distributed Hash Table (DHT) overlay as well. Each time a proxy receives a query, it tries to respond it using its own cache. If it is not possible, the DHT module hashes and forwards the query to the appropriate proxy, called as “master” proxy server for that query. It decides whether to handle the request itself (if possible) or to forward it to the the central database server. Updates to the database are always forwarded to the central database server. For each update issued to a proxy, its consistency module publishes an update notification to the multicast groups related to that update. Proxy servers that receive the update notification invalidate the possibly affected query results in their caches. Ferdinand maps database queries and updates to multicast groups using offline analysis which is based on query-update independence analysis.

The paper reports the results of some experiments based on three benchmark applications. The aim of experiments are to determine the relative contribution of each idea in Ferdinand to performance of web applications separately. The evaluation of both ideas, cooperative caching and publish/subscribe consistency management, shows outperformance of Ferdinand in comparison to the related works. However, Ferdinand does not show this outperformance in high-latency environments because of the cost of cache misses.

SUMMARIZED BY: Fathiyeh Faghihekhorasani