ANNUAL REPORT FOR AWARD # 0238386

Jun Yang ; *Duke University*
CAREER: Techniques and Applications of Derived Data Maintenance

Participant Individuals:

Graduate student(s) : Hao He; Adam Silberstein; Junyi Xie; Badrish Chandramouli

Participants' Detail

Partner Organizations:

Other collaborators:

Pankaj K. Agarawal, Carla S. Ellis, and Kamesh Munagala. Faculty members in the Department of Computer Science, Duke University.

Rebecca L. Braynard, Ke Yi, and Hai Yu. Students in the Department of Computer Science, Duke University.

Yuguo Chen. Institute of Statistics and Decision Science, Duke University.

Amin Vahdat. Computer Science Department, University of California at San Diego.

Cliburn Chan, Lindsay G. Cowell, and Thomas B. Kelper. Center for Bioinformatics and Computational Biology, Duke University.

David F. Kong. Duke University Medical Center.

Ioana Stanoi, Haixun Wang, and Philip S. Yu. IBM T. J. Watson Research Center.

Activities and findings:

Research and Education Activities:

The focus of this CAREER project is on techniques and applications of derived data maintenance. Derived data is the result of applying some transformation, structural or computational, to base data. The use of derived data to facilitate access to base data is a recurring technique in many areas of computer science. Used in hardware and software caches, derived data speeds up access to base data. Used in replicated systems, it improves reliability and performance of applications in a wide-area network. Used as index structures, it provides fast alternative access paths to base data. Used as materialized views in databases or data warehouses, it improves the performance of complex queries over base data.

Used as synopses, it provides fast, approximate answers to queries or statistics needed for cost-based optimization. Derived data may vary in complexity: it can be a simple copy of base data, in the cases of caching and replication, or it can be the result of complex transformations, in the cases of indexes and materialized views. Derived data may also vary in accuracy: caches and materialized views are usually exact, while synopses are approximate. Regardless of the varying forms, purposes, complexity, and accuracy of derived data, it must be maintained when base data is updated. Thus, derived data maintenance is a fundamental problem in computer science. It is also an evolving problem: existing techniques are constantly challenged by the explosive growth in data volume and number of data producers and consumers, and by increasing diversity in data formats.

Traditionally, derived data maintenance has been tackled separately in different contexts, e.g., index updates and materialized view maintenance in databases, cache coherence and replication protocols in distributed systems. Although they share the same underlying theme, these techniques have been developed and applied largely disjointly. Newer and more complex data management tasks, however, call for creative combinations of the traditionally separate ideas. Semantic caching, which has received tremendous interests recently for its applications in caching dynamic Web contents, is a good example of incorporating the idea of materialized views into a cache. With ``outside-the-box'' thinking such as semantic caching, we can discover more techniques that combine multiple flavors of derived data to provide better solutions to problems.

In Year 2 of this project, we have investigated the following specific research problems:

1. Caching for stream data processing. We continued our investigation of operator state management for stream data processing from Year 1. Answering continuous queries over streams can be seen as incrementally maintaining views given update streams without accessing base data. With limited memory, choosing what to keep in the state to maximize result completeness is analogous to making replacement decisions in an auxiliary data cache. Our work on joining and caching stochastic streams elucidates the difference and relationship between join state management for streams and the classic caching problem. Their difference implies that optimal classic caching policies do not carry over directly to joining streams. Despite the difference, we show the two problems can be tackled under a unified framework, and we develop techniques for deriving provably optimal replacement policies in some cases and good heuristics in others. Recognizing the importance of update constraints and limitation of hard constraints, we exploit statistical properties of streams---soft, probabilistic constraints---to optimize replacement decisions. It is tempting to give heuristic solutions without understanding when and why they work (or fail). We strive to follow a principled approach to validating solutions both experimentally and analytically. While our work in Year 1 focused on laying out the theoretical foundations, in Year 2 we developed practical techniques for implementing replacement policies efficiently. The resulting approach does not need to know in advance the parameters values of the stochastic processes generating the input streams; instead, it relies on low-overhead

runtime monitoring of appropriate statistics. We worked on this problem in collaboration with Yuguo Chen, a colleague at Duke statistics, who has helped bring much rigor to the analysis of our solutions. This work is published in SIGMOD 2005.

2. Incremental maintenance of order-based XML labeling. Order-based element labeling (e.g., pre-/post-order or interval labels) for tree-structured XML data is an important technique in XML processing. It lies at the core of many fundamental XML operations such as containment join and twig matching. Element labels are a form of derived data. While labeling for static XML documents is well understood, less is known about how to maintain accurate labeling for dynamic XML documents, when elements and subtrees are inserted and deleted. Most existing approaches do not work well for arbitrary update patterns; they either produce unacceptably long labels or incur enormous relabeling costs. We have developed two novel I/O-efficient data structures, W-BOX and B-BOX, that efficiently maintain labeling for dynamic XML documents. We show analytically and experimentally that both, despite consuming minimal amounts of storage, gracefully handle arbitrary update patterns without sacrificing lookup efficiency. The two structures together provide a nice tradeoff between update and lookup costs: W-BOX has logarithmic amortized update cost and constant worst-case lookup cost, while B-BOX has constant amortized update cost and logarithmic worst-case lookup cost. We further developed techniques based on caching to eliminate the lookup cost for read-heavy workloads. This work is published in ICDE 2005.

3. Asymmetric batch incremental view maintenance. Incremental view maintenance is probably the most well-studied problem about derived data in the database community. It has a growing number of applications recently, including for example data warehousing and publish/subscribe systems. Batch processing of base table modifications, when applicable, can be much more efficient than processing individual modifications one at a time. We tackle the problem of finding the most efficient batch incremental maintenance strategy under a refresh response time constraint; that is, at any point in time, the system, upon request, must be able to bring the view up to date within a specified amount of time. The traditional approach is to process all batched modifications relevant to the view whenever the constraint is violated. However, we observe that there often exists natural asymmetry among different components of the maintenance cost; for example, modifications on one base table might be cheaper to process than those on another base table because of some index. We exploit such asymmetries using an unconventional strategy that selectively processes modifications on some base tables while keeping batching others. We develop a series of analytical results leading to practical algorithms that approximate an 'oracle algorithm' with perfect knowledge of the future. We demonstrate that our strategy offers substantial performance gains over traditional deferred view maintenance techniques. This work is published in ICDE 2005. So far, to obtain a provably good maintenance plan, we need to assume some knowledge of the arrival pattern of base table updates. Currently, we are developing more robust online algorithms that are competitive against any oblivious adversary, in collaboration with Kamesh Munagala, a colleague in Duke Computer Science.

4. Scalable continuous query processing. Continuous query processing has attracted much interest from the database community recently because of its wide range of traditional and emerging applications, e.g., trigger and production rule processing, data monitoring, stream processing, and publish/subscribe systems. In contrast to traditional query systems, where each query is run once against a snapshot of the database, continuous query systems support standing queries that continuously generate new results (or changes to results) as data updates continue to arrive in a stream. In this sense, continuous query processing has much in common with incremental view maintenance, and can be regarded also as a problem of derived data maintenance. One of the main challenges in continuous query processing is how to handle a large number of continuous queries in a scalable way. For each incoming data update, the system needs to identify the subset of continuous queries whose results are affected by the data update, and compute changes to these results. If there are many continuous queries, a brute-force approach that processes each of them in turn will be inefficient and unable to meet the response-time requirement of most target applications. One important insight gained by research on scalable continuous query processing is the interchangeable roles played by queries and data. In continuous query systems, continuous queries can be treated as data, while each data update can be treated as a query requesting the subset of continuous queries affected by the update. Thus, it is natural to apply indexing and query processing techniques traditionally intended for data to continuous queries. Most existing work on indexing continuous relational queries has focused on selections. As far as we know, there has been little work on how to process more complex continuous queries (e.g., joins) scalably. We have been developing efficient processing techniques in collaboration with Pankaj K. Agarwal, a colleague in Duke Computer Science. In particular, we have developed novel, 'input-sensitive' schemes for indexing continuous joins with range conditions. The performance of these schemes increases with the degree of clusteredness of the range conditions being indexed: More clustered queries lead to more efficient processing. We have also obtained other results including lower bounds on the inherent complexity of the problem, and data structures with space-time tradeoffs. The work is currently under submission.

5. Querying networked data. As networks continue to grow in size and complexity, distributed network monitoring and resource querying are becoming increasingly difficult. Our aim is to design, build, and evaluate a scalable infrastructure for answering queries over distributed measurements, using reduced costs (in terms of both network traffic and query latency) while maintaining required precision. To this end, we use the technique of bounded approximate caching. Each network node owns a set of numerical measurements and actively maintains bounds on these values cached at other nodes. We can answer queries approximately, using bounds from nearby caches to avoid contacting the owners directly. We focus on developing efficient and scalable techniques to place, locate, and manage bounded approximate caches across a large network. We have developed two approaches: One uses a recursive partitioning of the network space to place caches in a static, controlled manner, while the other uses a locality-aware distributed hash table to place caches in a dynamic and decentralized manner. Experiments using large-scale network emulation show that our techniques are very effective in reducing

query costs while generating an acceptable amount of background traffic; they are also able to exploit various forms of locality that are naturally present in queries, and adapt to volatility of measurements. We are building this system in collaboration with Amin Vahdat, a networking/distributed systems researcher at University of California at San Diego. The work is currently under submission.

The progress we have made in Year 2, as summarized above, are inline with the modified research plan outlined in the project report from Year 1. (1) We have broaden our study of derived data maintenance to continuous queries and subscriptions. (2) We have successfully tackled a number of derived data maintenance issues in the context of XML. (3) We have also begun investigating derived data in a network setting. In Year 3, we plan to continue to push batching and scalable processing techniques in derived data maintenance. Our target application is a wide-area publish/subscribe system with a rich subscription language that supports 'stateful' subscriptions beyond simple message filters. We are currently seeking additional funding to complement our work in this application area and to enable us to address scalable network dissemination problem in collaboration with networking researchers.

In terms of educational activities, I have continued to incorporate current research topics into both undergraduate and graduate database course at Duke University. The undergraduate and graduate database courses I offered in Fall 2004 and Spring 2005 covered a substantial amount of material drawn from the latest research. These courses helped catapult my group's entry into new fields. One student from my undergraduate class subsequently pursued his undergraduate honors thesis research with me based on his course project. The result is a BS degree with High Distinction, and a solid technical report which we hope to publish with some more work.

Findings:

We have made significant progress in studying the derived data maintenance problem in multiple application domains, including view maintenance, data warehousing, stream data processing, XML indexing, and scalable continuous query processing. Published results from this grant so far include: (1)~an efficient method for top-k view maintenance that incorporates the idea of caching (ICDE 2003); (2)~a novel XML structural index (ICDE 2004) and (3)~efficient incremental maintenance algorithms for XML structural indexes (SIGMOD 2004), which incorporate the use of auxiliary data; (4)~efficient maintenance of order-based labeling for dynamic XML documents, with different degrees of materialization to provide a tradeoff between query and update performance (ICDE 2005); (5)~a novel approach to batch incremental view maintenance that exploits asymmetry in maintenance cost components (ICDE 2005); (6)~a new framework and techniques for managing the state of a stream join to maximize result completeness, which is related to the classic caching problem (SIGMOD 2005); (7)~a new, input-sensitive approach to scalable processing of continuous join queries (technical report submitted for publication); (8)~a wide-area network querying system employing bounded approximate caching (technical report submitted for publication). Our research has also led to a Master's thesis by Zhihui Wang (MS, 2003) and an undergraduate honors thesis by Christopher N. Bond (BS with High Distinction, 2005). Based on these findings we believe that the direction we are currently pursuing is a promising one. For detailed descriptions of these findings please refer to the section of this report on research and education activities.

Training and Development:

The PI has advised the following students in the context of this project:

Ph.D. students: Hao He, Adam Silberstein, Junyi Xie, Badrish Chandramouli.

M.S. student: Zhihui Wang (thesis completed in 2003), Wenbin Pan (thesis completed in 2004).

Undergraduate student: Christopher N. Bond (BS with High Distinction, 2005).

Outreach Activities:

The PI has been active in running the Carolina Database Research Group (http://www.cs.duke.edu/cdb/) with a group of database researchers in North Carolina, including members from Duke, North Carolina State University, University of North Carolina at Chapel Hill, Charlotte, and Greensboro. We hold monthly meetings and are currently running a seminar series, which have been a great resource for facilitating student and faculty interaction across institutions and attracting student interests in database research.

Journal Publications:

Book(s) of other one-time publications(s):

Ke Yi, Hai Yu, Jun Yang, Gangqiang Xia, and Yuguo Chen, "Efficient Maintenance of Materialized Top-k Views" , bibl. Bangalore, India, (2003). *Proceedings* Published
of Collection: , "Proceedings of the 19th International Conference on Data Engineering (ICDE '03)"

Junyi Xie, Jun Yang, and Yuguo Chen, "On Joining and Caching Stochastic Streams" , bibl. Baltimore, Maryland, June 2005, (2005). *Proceedings* Published
of Collection: , "Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data"

Adam Silberstein and Jun Yang, "NeXSort: Sorting XML in External Memory" , bibl. Boston, Massachusetts, March 2004, (2004). *Proceedings* Published
of Collection: , "Proceedings of the 20th International Conference on Data Engineering (ICDE '04)"

Hao He and Jun Yang, "Multiresolution Indexing of XML for Frequent Queries" , bibl. Boston, Massachusetts, March 2004, (2004). *Proceedings* Published
of Collection: , "Proceedings of the 20th International Conference on Data Engineering (ICDE '04)"

Ke Yi, Hao He, Ioana Stanoi, and Jun Yang, "Incremental Maintenance of XML Structural Indexes" ,

bibl. Paris, France, June 2004, (2004). *Proceedings* Published
of Collection: , "Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data (SIGMOD '04)"

Zhihui Wang, "Multiple-View Maintenance with Semantic Caching" , bibl. Durham, North Carolina, August 2003, (2003). *Thesis* Published
of Collection: , "M.S. Thesis, Duke University"

Pankaj K. Agarwal, Junyi Xie, Jun Yang, and Hai Yu, "On Scalable Processing of Continuous Joins" , bibl. Durham, North Carolina, December 2004, (2004). *Technical Report* Submitted
of Collection: , "Technical Report, Department of Computer Science, Duke University"

Badrish Chandramouli, Jun Yang, and Amin Vahdat, "Distributed Network Querying with Bounded Approximate Caching" , bibl. Durham, North Carolina, June 2004, (2004). *Technical Report* Submitted
of Collection: , "Technical Report, Department of Computer Science, Duke University"

Adam Silberstein, Hao He, Ke Yi, and Jun Yang, "BOXes: Efficient Maintenance of Order-Based Labeling for Dynamic XML Data" , bibl. Tokyo, Japan, April 2005, (2005). *Proceedings* Published
of Collection: , "Proceedings of the 21st International Conference on Data Engineering"

Hao He, Junyi Xie, Jun Yang, and Hai Yu, "Asymmetric Batch Incremental View Maintenance" , bibl. Tokyo, Japan, April 2005, (2005). *Proceedings* Published
of Collection: , "Proceedings of the 21st International Conference on Data Engineering"

Other Specific Products:

Internet Dissemination:

http://www.cs.duke.edu/dbgroup/ddm/

Contributions:

Contributions within Discipline:

We have made contributions to multiple application domains of derived data maintenance, including including view maintenance, data warehousing, stream data processing, XML indexing, and scalable continuous query processing. A number of the contributions have been published in premier database conferences: ICDE 2003, ICDE 2004, SIGMOD 2004, ICDE 2005, and SIGMOD 2005. For detailed descriptions of these contributions please refer to the section of this report on research and education activities.

Contributions to Resources for Science and Technology:

The PI have been active in running the Carolina Database Research Group (http://www.cs.duke.edu/cdb/) with a group of database researchers in North Carolina, including members from Duke, North Carolina State University, University of North Carolina at Chapel Hill, Charlotte, and Greensboro. We hold monthly meetings and are currently running a seminar series, which have been a great resource for facilitating student and faculty interaction across institutions and attracting student interests in database research.

Special Requirements for Annual Project Report:

Unobligated funds: less than 20 percent of current funds

Categories for which nothing is reported:
Participants: Partner organizations
Products: Journal Publications
Products: Other Specific Product
Contributions to Other Disciplines
Contributions to Education and Human Resources
Contributions Beyond Science and Engineering
Special Reporting Requirements
Animal, Human Subjects, Biohazards

Submit Return

FASTLANE System Control We welcome comments on this system