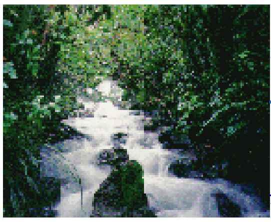

stanfordstreamdat
amanager

System Availability: Internet System and Source Code

The first version of our prototype Data Stream Management System is now available for public use. You may try the system over the internet, or you may download the code.

Internet System

You can try the STREAM prototype without downloading and installing the source -- we start up a server at Stanford and a client on your machine.**CLICK HERE**to give it a try, and*don't forget to read the Online Help so you know what you're doing*. If the server above is overloaded or unavailable, try**this one**instead.

Using the internet-accessible system you can try out much of the prototype's functionality: a few predefined streams and queries (including the Linear Road benchmark), registering new streams and queries, visualizing and monitoring query plans, and other features.

Source Code

If you're interested in installing the prototype system at your own site, thesource code is available for download. Currently (November '04) only the main server code is available -- it is accessed using a library or a thin command-line client. We plan to release the full graphical user-interface, including query plan visualization and monitoring, within a short time. The code release includes an extensiveuser manual.

Motivation

In applications such as network monitoring, telecommunications data management, web personalization, manufacturing, sensor networks, and others, data takes the form of continuous**data streams**rather than finite stored data sets, and clients require long-running**continuous queries**as opposed to one-time queries. Traditional database systems and data processing algorithms are ill-equipped to handle complex and numerous continuous queries over data streams, and many aspects of data management and processing need to be reconsidered in their presence. In the STREAM project, we are reinvestigating data management and query processing in the presence of**multiple, continuous, rapid, time-varying**data streams. We are attacking problems ranging from**basic theory results**to**algorithms**to implementing a comprehensive prototype**data stream management system**.

The STREAM project is supported in part by the National Science Foundation under grants IIS-0118173 and IIS-9817799.

News

(**ongoing**) We maintain aStream Query Repositoryas a resource for researchers in data streams.

(**March 2004**) A new overview paper on the STREAM project is available: STREAM: The Stanford Data Stream Management System. It will appear in a book on data stream management edited by Garofalakis, Gehrke, and Rastogi.

(**August 2003**) The latest informal get-together of data streams research groups was hosted by David Maier's group at OGI in Portland. Meeting notes are available from theStream TeamWeb page. The next Stream Team meeting is planned for spring '04, location TBD. [Summer '04: It was held in Berkeley, but the notes are still "being assembled"]

-
-
-
- () We instantiated the Linear Road Benchmark (see) with a.

People

Faculty

Rajeev Motwani

-
-

Students (grad and undergrad)

Arvind ArasuBrian BabcockShivnath BabuUtkarsh Srivastava

-
-
-
-
-

Alums

John CieslewiczMayur DatarKeith ItoJon McAlisterItaru Nishizawa (Hitachi, Ltd.)
Justin RosensteinVitthal Srinivasan

-
-

-
-
-
-
-
-

Talks

The Stanford Data Stream Management System.

Jennifer Widom's standard talk on the project, updated October 2004 Slides in PowerPoint

CQL: A Language for Continuous Queries over Streams and Relations.

Invited talk given by Jennifer Widom at the 2003 DBPL Workshop (Sept. 2003) Slides in PowerPoint

Randomization for Massive and Streaming Data Sets.

Talk given by Rajeev Motwani at the Stanford Computer Science Forum - Annual Affiliates Meeting (May 2003)

-
  -
  -
-
  -
  -
-
  -
- .Invited talk given by Rajeev Motwani at the(June 2002)

Demos

The STREAM prototype Data Stream Management System is available **here** to try over the Internet. You may also download the source code here.

We demo'd *StreaMon*, the adaptive query processing engine of the STREAM system, at the ACM SIGMOD 2004 Conference. Here was the demonstration proposal.

-
-
- We demo'd the STREAM system and its dynamic visualizer at the. Here was the.

Papers

Overviews and Surveys

The STREAM Group. Stanford Data Stream Management System (latest overview paper)

To appear in a book on data stream management edited by Garofalakis, Gehrke, and Rastogi.

The STREAM Group. STREAM: The Stanford Stream Data Manager (short overview paper)

IEEE Data Engineering Bulletin, Vol. 26 No. 1, March 2003

R. Motwani, J. Widom, A. Arasu, B. Babcock, S. Babu, M. Datar, G. Manku, C. Olston, J. Rosenstein, and R. Varma. Query Processing, Resource Management, and Approximation in a Data Stream Management System

In Proc. of the 2003 Conf. on Innovative Data Systems Research (CIDR), January 2003

This paper describes our ongoing work developing the Stanford Stream Data Manager (STREAM), a system for executing continuous queries over multiple continuous data streams. The STREAM system supports a declarative query language, and it copes with high data rates and query workloads by providing approximate answers when resources are limited. This paper describes specific contributions made so far and enumerates our next steps in developing a general-purpose Data Stream Management System.

B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and Issues in Data Stream Systems

Invited paper in Proc. of the 2002 ACM Symp. on Principles of Database Systems (PODS 2002), June 2002

In this overview paper we motivate the need for and research issues arising from a new model of data processing. In this model, data does not take the form of persistent relations, but rather arrives in multiple, continuous, rapid, time-varying data streams. In addition to reviewing past work relevant to data stream systems and current projects in the area, the paper explores topics in stream query languages, new requirements and challenges in query processing, and algorithmic issues.

-
-
-
-
- S. Babu and J. Widom.

System

U. Srivastava and J. Widom.Flexible Time Management in Data Stream Systems

In Proc. of the 2004 ACM Symp. on Principles of Database Systems (PODS 2004), June 2004

Flexible application-defined time poses challenges to a Data Stream Management System, since streams may be out of order and uncoordinated with each other, they may incur latency reaching the DSMS, and they may pause or stop. We formalize these challenges and specify how to generate heartbeats so that queries can be evaluated correctly and continuously in an application-defined time domain. Our heartbeat generation algorithm is based on parameters capturing skew between streams, unordering within streams, and latency in streams reaching the DSMS. We also describe how to estimate these parameters at run-time, and we discuss how heartbeats can be used for processing continuous queries.

B. Babcock, M. Datar, and R. Motwani.Load Shedding for Aggregation Queries over Data Streams

In Proc. of Intl. Conference on Data Engineering (ICDE 2004), March 2004

We present load shedding techniques for a restricted class of stream queries: Aggregation queries over sliding windows, possibly with selections, projections and foreign key joins with stored relations. We present optimal solutions for placing load shedders (operators which randomly drop tuples) in the query plan, which reduce the load on the system below the required threshold, while minimizing the inaccuracy introduced in the queries.

D. Thomas and R. Motwani.Caching Queues in Memory Buffers

In In Proc. of the Annual ACM-SIAM Symp. on Discrete Algorithms (SODA 2004), January 2004

We study the problem of maintaining queues in a cache, which occurs in a number of important settings like DataStream systems, Network Router design and Distributed Messaging services. We analyze why DataStream systems built on top of buffer managers that use traditional algorithms like LRU perform badly. We provide online competitive algorithms for this problem for different interesting cost models.

-
-
-

- B. Babcock, S. Babu, M. Datar, R. Motwani, and D. Thomas.

Query Language

A. Arasu and J. Widom. A Denotational Semantics for Continuous Queries over Streams and Relations

Technical Report, Mar. 2004

We present formal, denotational semantics for a generic continuous query language based on streams, time-varying relations, and three classes of operators over streams and relations.

-
- A. Arasu, S. Babu and J. Widom.

Query Processing

S. Babu, K. Munagala, J. Widom, and R. Motwani. Adaptive Caching for Continuous Queries

To appear in Proc. ICDE 2005, April 2005

We study the problem of using caches to improve performance and adaptivity in continuous multiway joins. We propose different cache types and algorithms for cache maintenance, monitoring cache cost and benefits, selecting caches to use, allocating memory to caches, and adapting over the entire spectrum between stateless

K. Munagala, S. Babu, R. Motwani, and J. Widom. The Pipelined Set Cover Problem

To appear in Proc. ICDT 2005, January 2005

A classical problem in query optimization is to find the optimal ordering of a set of possibly correlated selections or joins. We provide an abstraction of this problem as a generalization of set cover called

A. Arasu and J. Widom. Resource Sharing in Continuous Sliding-Window Aggregates

To appear in Proc. of VLDB 2004, Sep. 2004

We consider the problem of resource sharing when processing large numbers of continuous queries. We specifically address sliding-window aggregates over data streams, an important class of continuous operators for which sharing has not been addressed. We present a suite of sharing techniques that cover a wide range of possible scenarios: different classes of aggregation functions (algebraic, distributive, holistic), different window types (time-based, tuple-based, suffix, historical), and different input models (single stream, multiple substreams). We provide precise theoretical performance guarantees for our techniques, and show their practical effectiveness through a thorough experimental study.

U. Srivastava and J. Widom. Memory-Limited Execution of Windowed Stream Joins

To appear in Proc. of VLDB 2004, Sep. 2004

We address the problem of computing approximate answers to sliding-window joins over data streams when the available memory may be insufficient to keep the entire join state. The objective of the approximation may be either to return a maximum-size subset of the result or a random sample of the result. We introduce a new age-based model of stream arrival that is often more appropriate for addressing these problems than the traditional frequency-based model used in previous work. We also provide an algorithm for optimal memory allocation across multiple joins being executed in the system.

S. Babu, U. Srivastava, and J. Widom. Exploiting k-Constraints to Reduce Memory Overhead in Continuous Queries over Data Streams

Revised version to appear in ACM Transactions on Database Systems, Sept 2004

We introduce the important concept of

S. Babu, R. Motwani, K. Munagala, I. Nishizawa, and J. Widom. Adaptive Ordering of Pipelined Stream Filters

To appear in Proc. of the 2004 ACM Intl Conf. on Management of Data (SIGMOD 2004), June 2004

We consider the problem of pipelined filters, where a continuous stream of elements is processed by a set of commutative filters. We focus on the problem of ordering the filters adaptively to minimize processing cost in an environment where stream and filter characteristics vary unpredictably over time. Our core algorithm, A-Greedy (for Adaptive Greedy), has strong theoretical guarantees: If stream and filter characteristics were to stabilize, A-Greedy would converge to an ordering within a small constant factor of optimal. (In experiments A-Greedy usually converges to the optimal ordering.) We identify and study a three-way tradeoff among provable convergence to good orderings, run-time overhead, and speed of adaptivity.

A. Arasu, B. Babcock, S. Babu, J. McAlister, and J. Widom. Characterizing Memory Requirements for Queries over Continuous Data Streams

In ACM Transactions on Database Systems, March 2004.

We consider conjunctive queries with arithmetic comparisons over multiple continuous data streams. We specify an algorithm for determining whether or not a query can be evaluated using a bounded amount of memory for all possible instances of the data streams. When a query can be evaluated using bounded memory, our algorithm produces an evaluation plan based on constant-sized synopses of the data streams.

-
-
-
-
-
-
-

- U. Srivastava, S. Babu, and J. Widom.(short paper)

Distributed Streams

C. Olston, J. Jiang, and J. Widom.Adaptive Filters for Continuous Queries over Distributed Data Streams

In Proc. of the ACM Intl Conf. on Management of Data (SIGMOD 2003), June 2003

We consider an environment where distributed data sources continuously stream updates to a centralized processor that monitors continuous queries over the distributed data. Significant communication overhead is incurred in the presence of rapid update streams, and we propose a new technique for reducing the overhead. Users register continuous queries with precision requirements at the central stream processor, which installs filters at remote data sources. The filters adapt to changing conditions to minimize stream rates while guaranteeing that all continuous queries still receive the updates necessary to provide answers of adequate precision at all times.

-

- B. Babcock and C. Olston.

Statistics

A. Arasu and G. Manku.Approximate Counts and Quantiles over Sliding Windows

In Proc. of the 23rd ACM Symp. on Principles of Database Systems (PODS 2004), June 2004

We consider the problem of maintaining approximate counts and quantiles over fixed- and variable-size sliding windows in limited space. For quantiles, we present deterministic algorithms whose space requirements are O(1/e log(1/e)log N) and O(1/e log(1/e) log(eN) log N) in the worst-case for fixed- and variable-size windows, respectively, where N denotes the current number of elements in the window and e, the relative error. Our space bounds improve upon the previous best bounds of O(1/e^2 polylog (1/e,N)). For counts, we present both deterministic and randomized algorithms. The deterministic algorithms require O(1/e log^2 (1/e)) and O(1/e log^2 (1/e) log eN) for worst-case space for fixed- and variable-size windows, respectively, while the randomized ones require O(1/e log (1/(e d))) and O(1/e log(1/(ed)) log eN) worst-case space, where d denotes the probability of failure. We believe no previous work on space-efficient approximate counts for sliding windows exists.

B. Babcock, M. Datar, R. Motwani, and L. O'Callaghan.Maintaining Variance and k-Medians over Data Stream Windows

In Proc. of the 22nd ACM Symp. on Principles of Database Systems (PODS 2003), June 2003

We present a novel technique for solving two important and related problems in the sliding window model -- maintaining variance and maintaining k-medians clustering.

M. Datar and S. Muthukrishnan.Estimating Rarity and Similarity over Data Stream Windows

In Proceedings of European Symposium of Algorithms, Sept. 2002

We present solutions to two problems in the sliding window model: estimating

G. Cormode, M. Datar, P. Indyk, and S. Muthukrishnan. Comparing Data Streams Using Hamming Norm

In Proceedings of International Conference on Very Large Data Bases (VLDB 2002), August 2002

We present solution to the problem of computing Hamming norm over data streams. Hamming norm computation is more general than the well studied distinct value estimation problem. Our solution uses sketching techniques and works in the presence of inserts and deletes.

G. S. Manku and R. Motwani. Approximate Frequency Counts over Streaming Data

In Proc. of the 28th Intl. Conf. on Very Large Data Bases (VLDB 2002), August 2002

This paper present algorithms for computing frequency counts exceeding a user-specified threshold over data streams. Some applications are also presented.

M. Datar, A. Gionis, P. Indyk, and R. Motwani. Maintaining Stream Statistics Over Sliding Windows

SIAM Journal on Computing, Vol 31, No. 6, pp. 1794-1813

We consider the problem of maintaining statistics over sliding windows. We design data structures with small memory requirements and provide matching lower bounds.

-
-
-
-
-
-
- B. Babcock, M. Datar, and R. Motwani.

Clustering

S. Guha, A. Meyerson, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering Data Streams: Theory and Practice

IEEE Transactions on Knowledge and Data Engineering, vol. 15 (2003)

Under the data stream model, the data set to be processed is assumed to be too large to be processed together in RAM, and to be only accessible via linear scans, so that, for example, random access is unavailable. This model has recently attracted attention for its applicability to numerous types of data, including telephone records, web documents and clickstreams. For algorithms designed to analyze such data, the ability to process the data in a single pass, or a small number of passes, while using little memory, is crucial. We describe a one-pass, memory-efficient streaming algorithm that effectively clusters large data streams. We also provide empirical evidence of the algorithm's performance on synthetic and real data streams.

L. O'Callaghan, N. Mishra, A. Meyerson, S. Guha, and R. Motwani.
High-Performance Clustering of Streams and Large Data Sets

In Proc. of the 2002 Intl. Conf. on Data Engineering (ICDE 2002), February 2002

We give innovative techniques to transform theoretically well-founded algorithms for clustering into ones that perform well in practice. We further show that their performance is competitive with popular empirical approaches for clustering data streams.

-
-
- S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan.

Applications

- S. Babu, L. Subramanian, and J. Widom.

Last modified: November 7, 2004