Streaming in a Connected World: Querying and Tracking Distributed Data Streams

Graham Cormode
AT&T Labs–Research
180 Park Avenue, Florham Park NJ
graham@research.att.com

Minos Garofalakis
Yahoo! Inc
2821 Mission College Blvd., Santa Clara CA
minos@acm.org

Introduction and Motivation

Today, a majority of data is fundamentally distributed in nature. Data for almost any task is collected over a broad area, and streams in at a much greater rate than ever before. In particular, advances in sensor technology and miniaturization have led to the concept of the *sensor network*: a (typically wireless) collection of sensing devices collecting detailed data about their surroundings. A fundamental question arises: how to query and monitor this rich new source of data? Similar scenarios emerge within the context of monitoring more traditional, wired networks, and in other emerging models such as P2P networks and grid-based computing.

The prevailing paradigm in database systems has been understanding management of *centralized* data: how to organize, index, access, and query data that is held centrally on a single machine or a small number of closely linked machines. In these distributed scenarios, the axiom is overturned: now, data typically streams into remote sites at high rates. Here, *it is not feasible* to collect the data in one place: the volume of data collection is too high, and the capacity for data communication relatively low. For example, in battery-powered wireless sensor networks, the main drain on battery life is communication, which is orders of magnitude more expensive than computation or sensing. This establishes a fundamental concept for distributed stream monitoring: if we can perform more computational work *within* the network to reduce the communication needed, then we can significantly improve the value of our network, by increasing its useful life and extending the range of computation possible over the network.

We consider two broad classes of approaches to such in-network query processing, by analogy to query types in traditional DBMSs. In the *one shot* model, a query is issued by a user at some site, and must be answered based on the current state of data in the network. We identify several possible approaches to this problem. For simple queries, partial computation of the result over a tree can reduce the data transferred significantly. For "holistic" queries, such as medians, count distinct and so on, clever composable summaries give a compact way to accurately approximate query answers. Lastly, careful modeling of correlations between measurements and other trends in the data can further reduce the number of sensors probed.

In the *continuous* model, a query is placed by a user which requires the answer to be available continuously. This yields yet further challenges, since even using tree computation, summarization and modeling, we cannot afford to communicate every time new data is received by one of the remote sites. Instead, the result of work on this problem has been a new tradeoff of reduced accuracy in the query answer for reduced communication cost. This has led to a variety of techniques for different query types to apportion the available "uncertainty" in the query answer between different sites, and to model the evolution of measured values to anticipate future values and so reduce communication further.

There are many problems relating to distributed data stream monitoring yet to be studied, and systems addressing these issues are just starting to appear. There is great potential for work in this area to have impact on the next generation of data management systems in a world that is inherently distributed and constantly changing. Our objective in this tutorial is to discuss the algorithmic foundations of this new world, illustrate some of the powerful techniques that have been developed to address these challenges, and outline interesting directions for future work in the area. This tutorial complements and builds on some of the first attempts at collecting work in this area [1] and tutorials which have emphasized the hardware and systems aspects [2]. The aim is to inspire attendees to contribute to this exciting, growing area of research.

Tutorial Outline

The tutorial comprises five main thematic units, roughly organized as follows. (1) *Introduction:* defining different querying and communication models. (2) *One-Shot Distributed-Stream Querying:* covering tree-based aggregation, robustness/loss issues, and probabilistic techniques. (3) *Continuous Distributed-Stream Tracking:* discussing adaptive slack-allocation schemes, predictive site-behavior models, and distributed triggers. (4) *Distributed Data-Stream Systems*, and (5) *Future Directions and Open Problems*.

About the Speakers

Graham Cormode is a Principal Member of Technical Staff in the Database Management Group at AT&T Shannon Laboratories in New Jersey. He obtained his PhD from the University of Warwick in 2002.

Minos Garofalakis is a Principal Research Scientist with the Community Systems group at Yahoo! Research in Santa Clara, California, and an Adjunct Associate Professor of Computer Science at the University of California, Berkeley. He obtained his PhD from the University of Wisconsin-Madison in 1998.

EDBT'08, March 25–30, 2008, Nantes, France.

[1] M. Garofalakis *(editor)*. Special Issue on In-Network Query Processing. *IEEE Data Engineering Bulletin*, 28(1), March 2005.

[2] S. Madden. Data Management in Sensor Networks *(Tutorial)*. In *Proc. of European Workshop on Sensor Networks*, 2006.