

The Synergy Distributed Stream Processing Middleware

Thomas Repantis Vana Kalogeraki
Department of Computer Science & Engineering
University of California, Riverside
{trep,vana}@cs.ucr.edu

Xiaohui Gu
IBM T.J. Watson Research Center
{xiaohui}@us.ibm.com

<http://synergy.cs.ucr.edu/>

A significant number of emerging on-line data analysis applications require the processing of data that get updated continuously, to generate outputs of interest or to identify meaningful events. Application domains include the monitoring of network traffic for intrusion detection, surveillance of financial trades for fraud detection, observation of customer clicks for e-commerce applications, customization of multimedia or news feeds, and analysis of sensor data in real-time. Distributed stream processing systems allow in-network stream processing to achieve better scalability and quality-of-service (QoS) provision. In this work we present *Synergy* [1], a distributed stream processing middleware. Synergy is a software running on every machine of a distributed system to offer distributed stream processing applications, under Quality of Service constraints, and while efficiently managing the system's resources. Unlike previously developed distributed stream processing systems, Synergy offers sharing-aware component composition. It enables efficient reuse of both data streams and processing components, while composing distributed stream processing applications with QoS demands. Synergy provides a set of fully distributed algorithms to discover and evaluate the reusability of available data streams and processing components when instantiating new stream applications. For QoS provision, Synergy performs QoS impact projection to examine whether the shared processing can cause QoS violations on currently running applications. We have implemented a prototype of the Synergy middleware and evaluated its performance on both PlanetLab and simulation testbeds. The experimental results show that Synergy can achieve much better resource utilization and QoS provision than previously proposed schemes, by judiciously sharing streams and processing components during application composition.

References

- [1] T. Repantis, X. Gu, and V. Kalogeraki. Synergy: Sharing-aware component composition for distributed stream processing systems. In *Proceedings of the 7th ACM/IFIP/USENIX International Middleware Conference, MIDDLEWARE, Melbourne, Australia*, November 2006.