Enabling Real Time Data Analysis

Divesh Srivastava, Lukasz Golab, Rick Greer, Theodore Johnson, Joseph Seidel,
Vladislav Shkapenyuk, Oliver Spatscheck, Jennifer Yates
AT&T Labs-Research
{divesh, lgolab, rxga, johnsont, spence, vshkap, spatsch, jyates}@research.att.com

ABSTRACT

Network-based services have become a ubiquitous part of our lives, to the point where individuals and businesses have often come to critically rely on them. Building and maintaining such reliable, high performance network and service infrastructures requires the ability to rapidly investigate and resolve complex service and performance impacting issues. To achieve this, it is important to collect, correlate and analyze massive amounts of data from a diverse collection of data sources in real time.

We have designed and implemented a variety of data systems at AT&T Labs-Research to build highly scalable databases that support real time data collection, correlation and analysis, including (a) the Daytona data management system, (b) the DataDepot data warehousing system, (c) the GS tool data stream management system, and (d) the Bistro data feed manager. Together, these data systems have enabled the creation and maintenance of a data warehouse and data analysis infrastructure for troubleshooting complex issues in the network. We describe these data systems and their key research contributions in this paper.

1. MOTIVATION

Network-based services such as VoIP, IPTV, video conferencing, online stock trading, etc., have become a ubiquitous part of our lives. Individuals and businesses have often come to critically rely on these services, and service disruptions can cause considerable impact to these customers. Consequently, network and service providers build and maintain reliable, high performance infrastructures to support such network-based services. For example, networks support diverse routing and restoration mechanisms to recover from failures. Similarly, service infrastructures are designed with redundant configurations to minimize disruptions.

Managing such infrastructures to deal with the stringent service demands requires the ability to rapidly detect, investigate and resolve complex service and performance impacting issues. These issues include not only hard failures, but also short duration, recurring events and performance degradations (e.g., packet loss) that can seriously impact customers. The key to effectively addressing these issues is to collect, correlate and analyze the massive amounts of data generated by network and service infrastructures in real time. These data include a large variety of information from a diverse collection of data sources, including network configuration and topology, network element fault logs, workflow logs, performance data (for individual elements and end-to-end measurements), traffic and service measurements, etc. In this paper, we describe our approach to address these data management challenges.

Proceedings of the VLDB Endowment, Vol. 3, No. 1

2. DATA SYSTEMS

The core of our solution to enable real time data analysis across the above mentioned diverse collection of data sources is the creation and maintenance of the highly scalable Darkstar [4] data warehouse and data analysis infrastructure. Darkstar collects and stores both real time and historical data from the sources. Many applications, both for offline analysis and real time alerting and diagnosis, have been built and are running on top of Darkstar.

Darkstar itself is made possible by a variety of data systems designed and built at AT&T Labs-Research. These include (a) the Daytona data management system, (b) the DataDepot data warehousing system, (c) the GS tool data stream management system, and (d) the Bistro data feed manager. We briefly describe these data systems and their key research contributions next.

2.1 Daytona

The Daytona data management system [3] is used by AT&T to solve a wide spectrum of massive data management problems. For example, as of August 2010, Daytona is managing several hundreds of terabytes of tabular data in a 7x24 production data warehouse, whose largest table contains over 1.25 trillion records stored in over 135K files. Daytona offers all the essentials of data management including a high-level query language including a sophisticated view mechanism, data dictionary, B-tree indexing, locking, transactions, logging, and recovery.

Daytona's query architecture is based on translating its high-level query language Cymbal (which includes SQL as a subset) completely into C and then compiling that C into object code. The system resulting from this architecture is fast, powerful, easy to use and administer, reliable, and open to UNIX tools. Two forms of data compression help reduce the amount of disk needed to store data by as much as an order of magnitude over conventional database technologies. Robust horizontal partitioning and effective high-level SPMD parallelization enable Daytona to handle hundreds of terabytes with ease. Fast, scalable in-memory operations are supported by in-memory tables with skip-list indices accompanied by scalar and tuple-valued (hashed) associative arrays.

2.2 DataDepot

DataDepot [2] is a streaming data warehousing system, which

automates the task of managing base tables and materialized views in Daytona databases. For example, DataDepot is used to manage the real time Darkstar warehouse, loading over 100 raw data feeds, maintaining over 300 tables and materialized views, and ingesting hundreds of millions of raw records per day. DataDepot combines aspects of a data stream management system with conventional data warehousing systems, by providing very long term data storage as well as continuous real time updates of materialized views.

DataDepot continuously identifies newly arrived flat file feeds (containing raw data) that are to be loaded, and loads the data into base tables (possibly performing pre-processing on the data using external scripts before loading). Like conventional data warehousing systems, DataDepot provides the ability to manage deeply nested levels of materialized views; unlike conventional warehousing systems, updates to the base tables propagate through all the dependent materialized views continuously in real time, instead of being performed in a large batch mode. DataDepot also enables old data in the database to be timed out or archived, as needed.

DataDepot is largely implemented using Daytona's Cymbal language for use on Daytona databases. DataDepot makes effective use of Daytona's support for horizontally partitioned tables, to represent base tables and materialized views spanning long time periods and containing many terabytes of data, using temporal partitioning criteria. DataDepot has a number of features to facilitate real time data loading, including (i) propagating updates through temporal partitions, (ii) supporting multi-granularity partitions, allowing for small (e.g., 5 minute) partitions for recent data and large (e.g., 1 day) partitions for historical data, (iii) a lightweight multiversion concurrency control to allow queries to execute while views are being updated, and (iv) a real time update scheduler that executes updates when new data arrive and is tuned to minimize the tardiness of updates to critical tables.

2.3 GS Tool

GS tool [1] is a high-performance data stream management system (DSMS) designed for monitoring of networks with high-speed data streams. For example, GS tool is operationally used within AT&T's IP backbone, and processes over 1.6 million packets per second monitored on an OC-768 backbone router link. GS tool is intended to be adaptable so it can be used as a high speed data analysis engine in many settings: traffic analysis, performance monitoring and debugging, protocol analysis and development, router configuration (e.g., BGP monitoring), network attack and intrusion detection, and various ad hoc analyses.

Data from an external source arrives in the form of a sequence of data packets at one or more interfaces that GS tool monitors. These data packets can be IP packets, Netflow packets, BGP updates, etc., and are interpreted by a protocol. The GS tool run-time system interprets each data packet as a record of fields using a library of interpretation functions. GS tool's query language GSQL is a pure stream query language with an SQL-like syntax, supporting filters, unions, joins, groupby/aggregation, user-defined functions and aggregations. All inputs to a GSQL query are data streams, and the output is a data stream; this choice enables the composition of GSQL queries for complex query processing, and simplifies the implementation. The output of GS tool can be aggregated in Daytona using Cymbal's parallelization and shared memory features, and ultimately stored in a DataDepot data warehouse.

GS tool uses a unique two-level query architecture, which is critical for its high performance, where low level subqueries are used for significant data reduction, and high level queries performs more complex processing over the output of the low level subqueries. GS tool uses a number of optimizations to lower the processing costs of the low level subqueries, including (i) subqueries are compiled into C code that are linked directly to the runtime library to avoid expensive runtime query interpretation, (ii) when possible, some of the subquery processing is pushed into the network interface card, and (iii) to spread out the processing load over time and thus improve schedulability, GS tool implements traffic-shaping policies in some of its operators.

2.4 Bistro

Data are typically collected from a wide variety of sources and organizations, using a range of mechanisms - some of it streamed in real time, while other data are obtained at regular intervals or in an ad hoc fashion, either pushed to servers by remote applications or pulled from remote locations.

The Bistro data feed manager simplifies and automates this complex task of data feed management, to efficiently handle incoming raw files, identify data feeds and distribute them to remote subscribers. Bistro supports a flexible specification language to define logical data feeds using the physical data files, and to identify feed subscribers. Based on the specification, Bistro performs matching of data files to feeds, file normalization and compression, efficient file delivery, and subscriber notification using a trigger mechanism. Since the sources of data feeds are typically not under our control, an important feature of Bistro is to perform extensive logging to track the status of all the feeds, monitor their progress (e.g., if the expected data are incomplete), detect and correct any errors, and alarm if it is unable to correct errors.

Bistro's trigger mechanism enables real time propagation of updates all the way from data sources to the DataDepot data warehousing system (or other subscribers of Bistro's data feeds). In particular, Bistro is responsible for ensuring that over 100 raw data feeds are available to DataDepot, for loading in the real time Darkstar warehouse.

3. CONCLUSIONS

Collecting, correlating and analyzing massive amounts of data from a diverse collection of data sources in real time is a necessity to support reliable, high performance network and service infrastructures. This paper presents a variety of data systems, including the Daytona data management system, the DataDepot data warehousing system, the GS tool data stream management system, and the Bistro data feed manager, which we have designed and implemented at AT&T Labs-Research to build an end-to-end solution for data management to enable real time data analysis.

As the needs of real time data analysis grow, we expect to enhance the functionality and scalability of our current systems, and build new systems (e.g., to address data quality concerns).

4. REFERENCES

[1] C. D. Cranor, T. Johnson, O. Spatscheck, and V. Shkapenyuk. A stream database for network applications. In *SIGMOD Conference*, pages 647–651, 2003.

[2] L. Golab, T. Johnson, J. S. Seidel, and V. Shkapenyuk. Stream warehousing with DataDepot. In *SIGMOD Conference*, pages 847–854, 2009.

[3] R. Greer. Daytona and the fourth-generation language Cymbal. In *SIGMOD Conference*, pages 525–526, 1999.

[4] C. R. Kalmanek, Z. Ge, S. Lee, C. Lund, D. Pei, J. Seidel, J. van der Merwe, and J. Yates. Darkstar: Using exploratory data mining to raise the bar on network reliability and performance. In *Workshop on Design of Reliable Communication Networks*, 2009.