Report on the first Workshop on Innovative Querying of Streams

Michael Benedikt
University of Oxford, UK
michael.benedikt@cs.ox.ac.uk

Dan Olteanu
University of Oxford, UK
dan.olteanu@cs.ox.ac.uk

1. INTRODUCTION

The first workshop on

INnovative QUErying of STreams (INQUEST)

was held on September 25-27, 2012 in the Department of Computer Science of the University of Oxford (UK). It was sponsored by the UK's Engineering and Physical Sciences Research Council (EPSRC), as part of the project "Enforcement of Constraints on XML Streams".

Stream processing represents a thriving area of research across the algorithms, databases, networking, programming languages, and systems research communities. Within the database community, a "classical" problem is query processing on streams of discrete tuple-oriented data. One goal of the workshop considers the way recent developments add complexity to this problem:

- how does the setting change when data to be considered by queries is not relational, but has nested structure, such as XML or JSON?
- conversely, how does the setting change when data to be considered consists of RDF triples?
- how does the presence of noise in the data impact query processing?
- how does stream processing change when querying requires not only access to the data, but reference to external knowledge, which can also be changing?
- how does processing change in a large-scale decentralized setting?
- what new demands on stream query processing arise from social media applications? Is it only the processing architecture that changes, or do the queries change as well?

In addition to looking at new developments in stream processing, the workshop aimed to bring together researchers with different perspectives on the topic. We solicited and received participation from researchers working primarily on stream architectures and systems as well as those working on stream algorithms; the participants included researchers working on the computation of particular aggregates in streaming fashion as well as those looking at high-level languages for describing queries.

The workshop was by invitation only. There were 52 registered participants, ranging over 20 institutions. The formal part of the workshop program consisted of 19 invited lectures, grouped by topic.

In what follows, we present the main ideas and issues proposed by the speakers. Finally, discussions arisen during the workshop and concluding remarks are presented. The slides of workshop talks can be found on the current workshop web page:
`http://www.cs.ox.ac.uk/dan.olteanu/inquest12/pmwiki.php`

2. STREAMING OF SOCIAL DATA

This session covered challenges in building scalable infrastructure for managing social media streams and in extracting valuable information from social media streams such as emergent topics.

Sebastian Michel considered the problem of emergent topics discovery by continuously monitoring correlations between pairs of tags (or social annotations) to identify major shifts in correlations of previously uncorrelated tags in Twitter streams [1, 2]. Such trends can be used as triggers for higher-level information retrieval tasks, expressed through queries across various information sources.

Mila Hardt gave two talks on aspects related to managing streams at Twitter, in particular on infrastructure to enable processing of 400 million tweets a day and real-time top queries. Mila explained how stream processing needs at Twitter eventually led to the development of the open-source projects Storm and Trident[1] for large-scale high-performance distributed stream processing. She also pointed out

[1]https://github.com/nathanmarz/storm

current challenges at Twitter in providing support for fault tolerance, online machine learning by trading off exploration and exploitation, and approximating aggregates (such as counts). An interesting exercise involving the audience was on thinking how topic ranking is done at Twitter.

Daniel Preotiuc-Pietro introduced the Trendminer[2] system for real time analysis of social media streams [19]. Trendminer's scalability relies on the MapReduce framework for distributed computing. Daniel also presented how to build regression models of trends in streaming data using TrendMiner [21].

3. STREAMING AND THE SEMANTIC WEB

Stream processing has emerged as an important challenge in the new field of managing linked and semantic data. The workshop featured three talks on efforts in managing streams of linked data: one by Emmanuel Della Valle, covering work done in Politecnico Milano, one by Manfred Hauswirth, covering work done at *DERI* on platforms for linked data stream, and by Darko Anicic, covering joint work with Sebastian Rudolph and others at Karlsruhe Institute of Technology.

The requirements of a stream processing system for semantic data include support for "continuous querying" – queries that remain in place, with answers evolving as new data arises – and support for reasoning with external knowledge. The approach presented in Della Valle's talk involves merging the approach used for relational continuous query language with SPARQL. The resulting language, *C-SPARQL* [4], allows one to filter from a stream, using continuous-query window commands to control the sampling method, but SPARQL graph patterns can now be used within the filters.

Anicic outlined a different language approach. The *ETALIS* system [3] supports stream reasoning by embedding both temporal relational rules within a logic programming formalism. To better support the standards suite of the semantic Web, *ETALIS* supports a proper extension of SPARQL for dealing with event-processing on streams, *EP-SPARQL*.

Of course, using stream processing on large-scale linked data involves more than just developing a language or even a query processing engine. Hauswirth's talk outlined the entire set of issues needed to build an application that integrates and processes sensor output using linked data. This includes a continuous query evaluation system specific to linked data, *CQELS* [18], but also addresses the modifications needed to storage, protocol, RESTFul services, date interchange formats, and data integration technology needed to exploit these query languages in real-world applications.

[2]https://github.com/sinjax/trendminer

4. STREAM MONITORING

Monitoring of streams is a good example of a subarea of streaming where different communities define the objectives in radically different ways, and attack the problem using very different techniques. For the verification community, monitoring appears in the form of run-time verification – for example, continuously monitoring reactive systems for violation. The focus is normally on temporal constraints. Issues of space consumption are critical, as in most stream-processing applications, but there is also a need to integrate the constraint language and the monitoring engine with data structures maintained in the code being monitored. In databases both the constraint languages and the monitoring model are normally quite different; constraints naturally focus on properties of data values (e.g. as in classical dependencies), while monitoring occurs both in batch mode and in response to discrete updates. Both of these communities have dealt with monitoring as a component with a very well-demarcated set of functionality within a larger system. In contrast, monitoring data has a broad meaning within data-oriented applications, with integrity-constraint validation being only one aspect of it.

Felix Klaedtke's talk came from the perspective of run-time verification. He focused on online monitoring of integrity constraints, where the constraints deal with the evolution of data over time, and are thus expressed in a variant of first-order temporal logic. He explained both the system and a set of algorithms for efficiently monitoring these constraints [5]. In this work, ideas from runtime verification and the database community (particularly, temporal databases) interact.

Lukasz Golab looked at properties of streams of relational data, focusing on two natural set of constraints that deal with both temporal and more traditional relational aspects. He defined sequential constraints, which generalize functional dependencies to account for order, and conservation laws [12] that are specific to the context of pairs of numeric streams corresponding to related quantities. He presented methods for checking these constraints in off-line fashion, as well as methods for seeing the extent to which they are violated.

Mariano Consens talked about monitoring in the broader sense – how can the quality and the accesses to data records be monitored off-line in the presence of large volumes of linked data. His work focuses on privacy issues in data, presenting an integrity language that allows one to formulate constraints expressing that a privacy violation has occurred. He also presented a system providing an end-to-end solution for auditing privacy constraints, including a means for integrating records from diverse datasources, for expressing privacy policies and constraints, and for detecting violations.

5. XML STREAMS

XML is notable for being a data model where very strong notions of streamability can be formalized for very expressive query and schema languages. Joachim Niehren looked at one natural formalization for node-selecting queries: the ability to determine at any point in an XML stream which nodes "must be" in the query result, where "must" means that they will be in the result in any possible extension. Niehren presented automata-theoretic methods of solving this "earliest answer problem", along with lower bounds.

While Niehren's talk focused on node-selecting languages such as XPath, Pavel LaBath looked at stream-processing of the World Wide Web consortium's XML transformation language, XSLT. He presented a subset of the language that can be effectively streamed [15]. A notable aspect of XSLT is that the W3C working group has looked to standardize a subset of the language that is appropriate for streaming applications.

6. UNCERTAIN STREAMS

Applications like location-based services (RFID) and text recognition (OCR) are driven by data that is low-level, imprecise, and sequential. To effectively exploit this low-level data, it must be transformed into higher-level data that is meaningful to a particular application. For example, in RFID applications, a sequence of raw sensor readings is transformed into a sequence of physical locations. In OCR, the low-level sequence of images on a page is transformed into a sequence of ASCII characters. Often, this transformation uses a probabilistic model like a Hidden Markov Model for RFID, Kalman Filter for tracking, Stochastic Transducer for Google's Ocropus tool for OCR, or approximates location data by uncertain ranges defined using continuous probability distributions over locations of moving objects. Besides the richness of data models, applications also need a variety of querying and monitoring facilities, such as continuous and probabilistic versions of spatial queries including nearest neighbour, range, and similarity queries, and queries specified by finite automata that can exploit the order of data items in the stream.

This workshop session featured three talks that covered complementary aspects of challenges in managing uncertain streams that are exemplary for most of the existing efforts in this research area.

Chris Ré overviewed work done in the Lahar [20] and Hazy research projects to effect transformations from low-level to high-level high quality uncertain streams modelled by Markov Sequences and subsequently to query such streams using transducers (i.e., automata with output) [13]. He presented several applications including a monitoring application based on uncertain RFID readings [20] and the GeoDeepDive application, which aims at unearthing data from the Geoscience literature by modelling OCR output using Stochastic Transducers and by integrating such models into relational database systems [14].

Reynold Cheng presented work on continuous nearest neighbour and range queries over imprecise location, where data is modelled by uncertain ranges defined by continuous probability distributions over locations of moving objects [7, 24]. In location-based services, saving communication bandwidth between servers and objects and mobile devices' battery is essential and Reynold showed how this can be effectively achieved by employing object filtering based on the probability that the object is close to a given query point.

Themis Palpanas surveyed techniques for modeling and processing data series with value uncertainty, an important model for temporal data, where each data point in the series is represented by an independent discrete or continuous random variable. He focused on the problem of answering similarity queries on uncertain data series, and described a novel technique for this problem [9]. In addition, he discussed the challenges of dealing with both value and existential uncertainty in processing streaming uncertain data.

7. STREAMING FRAMEWORKS AND SYSTEMS

A major goal of the workshop was to bring together, on the one hand, computer scientists working in particular stream-processing domains (XML, RDF, etc.) or particular streaming algorithms, with researchers studying broad stream-processing systems capable of expressing a wide range of applications. Nesime Tatbul's talk focused on relational stream processing engines. This included an overview of both language proposals, such as *STREAM CQL*, *StreamSQL*, and *MATCH-RECOGNIZE*, along continuous querying architectures, such as the DBMSbased architectures of systems like *Truviso* and native streaming systems *StreamBase*. The ultimate goal would be to have an architecture that could express the features of each of the differing approaches to relational stream-processing, along with a clear set of systems definitions and embeddings of each engine into the "universal architecture". Tatbul's talk gave one step towards this goal, a versatile framework, *SECRET* [6], for describing the semantics of such systems, along with example descriptions of how some of the leading systems fit into the framework.

Yanif Ahmad talked about a new architecture being developed at Johns Hopkins for building next-generation streaming applications. Instead of beginning with "merely" data management infrastructure, the approach described by Ahmad begins with K3 [22], an event-driven language for general-purpose programming, building into the language both support for declarative data manipulation languages (e.g. for view definitions) and control structures for parallel and distributed programming.

8. DISTRIBUTED STREAMS

Big data analytics requires partitioning of large data streams into thousands of partitions according to specific set of keys so that different machines can continuously process different data partitions in parallel. This workshop session focused on analyzing requirements of and on solutions for distributed stream processing systems in the face of machine failure, pay-as-you-go models of computation, high-quality data partitioning, and low-overhead communication.

Peter Pietzuch discussed an approach to elastic and fault-tolerant stateful stream processing in the cloud, which was tested using the Linear Road Benchmark on the Amazon EC2 cloud platform [10]. The key aspects of this approach are on-demand scaling by acquiring additional virtual machines and parallelizing operators at runtime when the processing load increases, and fault-tolerance with fast recovery times yet low per-machine overheads.

Milan Vojnovic discussed the problem of range partitioning for big data analytics, where the goal is to produce approximately equal-sized partitions since the job latency is determined by the most loaded node [23]. The key challenge is to determine cost-effectively and accurately the partition boundaries in the absence of prior statistics about the key distribution over machines for a given input dataset. Cosmos, the cloud infrastructure for big data analytics used by Microsoft Online Services Division, uses a solution to this problem based on weighted sampling. Milan further presented a solution to the problem of continuous distributed counting [16], which had been mentioned earlier by Mila Hardt in her talk about Twitter.

Minos Garofalakis overviewed his recent work on approximate query answering with error guarantees in a distributed data streaming setting, where the focus is on communication efficiency, in addition to the standard space and time-efficiency requirements. In particular, Minos talked about sketching for distributed sliding windows [17], tracking complex aggregate queries [8], sketches based on the Geometric method, and sketch prediction models [11].

9. ACKNOWLEDGMENTS

We would like to thank the Engineering and Physical Sciences Research Council of the UK, who have sponsored INQUEST as part of the project *Enforcement of Constraints on XML Streams*, EPSRC EP/G004021/1.

Many of the staff at University of Oxford's computer science department were instrumental in making the workshop happen. In particular, we are very grateful to Polly Dunlop and Elizabeth Walsh for managing all of the arrangements for the meeting. We also thank Christoph Haase for being the webmaster for INQUEST.

10. REFERENCES

[1] Foteini Alvanaki, Sebastian Michel, Krithi Ramamritham, and Gerhard Weikum. EnBlogue: emergent topic detection in web 2.0 streams. In *SIGMOD*, 2011.

[2] Foteini Alvanaki, Sebastian Michel, Krithi Ramamritham, and Gerhard Weikum. See what's enblogue: real-time emergent topic identification in social media. In *EDBT*, 2012.

[3] Darko Anicic, Sebastian Rudolph, Paul Fodor, and Nenad Stojanovic. Stream reasoning and complex event processing in ETALIS. *Semantic Web*, 3(4), 2012.

[4] Davide Francesco Barbieri, Daniele Braga, Stefano Ceri, Emanuele Della Valle, and Michael Grossniklaus. C-SPARQL: a continuous query language for RDF data streams. *Int. J. Semantic Comp.*, 4(1), 2010.

[5] David Basin, Felix Klaedtke, and Samuel Müller. Policy monitoring in first-order temporal logic. In *CAV*, 2010.

[6] Irina Botan, Roozbeh Derakhshan, Nihal Dindar, Laura M. Haas, Renée J. Miller, and Nesime Tatbul. SECRET: a model for analysis of the execution semantics of stream processing systems. *PVLDB*, 3(1), 2010.

[7] Jinchuan Chen, Reynold Cheng, Mohamed F. Mokbel, and Chi-Yin Chow. Scalable processing of snapshot and continuous nearest-neighbor queries over one-dimensional uncertain data. *VLDB J.*, 18(5), 2009.

[8] Graham Cormode and Minos N. Garofalakis. Streaming in a connected world: querying and tracking distributed data streams. In *EDBT*, 2008.

[9] Michele Dallachiesa, Besmira Nushi, Katsiaryna Mirylenka, and Themis Palpanas. Uncertain time-series similarity: Return to the basics. *PVLDB*, 5(11), 2012.

[10] Raul Castro Fernandez, Matteo Migliavacca, Evangelia Kalyvianaki, and Peter Pietzuch. Integrating scale out and fault tolerance in stream processing using operator state management. In *SIGMOD*, 2013.

[11] Nikos Giatrakos, Antonios Deligiannakis, Minos N. Garofalakis, Izchak Sharfman, and Assaf Schuster. Prediction-based geometric monitoring over distributed data streams. In *SIGMOD*, 2012.

[12] Lukasz Golab, Howard J. Karloff, Flip Korn, Barna Saha, and Divesh Srivastava. Discovering conservation rules. In *ICDE*, 2012.

[13] Benny Kimelfeld and Christopher Ré. Transducing markov sequences. In *PODS*, 2010.

[14] Arun Kumar and Christopher Ré. Probabilistic management of OCR data using an RDBMS. *PVLDB*, 5(4), 2011.

[15] Pavel Labath. Xslt streamability analysis with recursive schemas. In *RCIS*, 2012.

[16] Zhenming Liu, Bozidar Radunovic, and Milan Vojnovic. Continuous distributed counting for non-monotonic streams. In *PODS*, 2012.

[17] Odysseas Papapetrou, Minos N. Garofalakis, and Antonios Deligiannakis. Sketch-based querying of distributed sliding-window data streams. *PVLDB*, 5(10), 2012.

[18] Danh Le Phuoc, Minh Dao-Tran, Josiane Xavier Parreira, and Manfred Hauswirth. A native and adaptive approach for unified processing of linked streams and linked data. In *ISWC*, 2011.

[19] Daniel Preotiuc-Pietro, Sina Samangooei, Trevor Cohn, Nicholas Gibbins, and Mahesan Niranjan. Trendminer: An architecture for real time analysis of social media text. In *ICWSM*, 2012.

[20] Christopher Ré, Julie Letchner, Magdalena Balazinska, and Dan Suciu. Event queries on correlated probabilistic streams. In *SIGMOD*, 2008.

[21] Sina Samangooei, Daniel Preotiuc-Pietro, Jing Li, Mahesan Niranjan, Nicholas Gibbins, and Trevor Cohn. Regression models of trends in streaming data. Technical report, University of Sheffield, 2012.

[22] P. C. Shyamshankar, Zachary Palmer, and Yanif Ahmad. K3: Language design for building multi-platform, domain-specific runtimes. In *XLDI*, 2012.

[23] Milan Vojnovic, Fei Xu, and Jingren Zhou. Sampling based range partition methods for big data analytics. Technical Report MSR-TR-2012-18, Microsoft Research, 2012.

[24] Yinuo Zhang and Reynold Cheng. Probabilistic filters: A stream protocol for continuous probabilistic queries. *Inf. Syst.*, 38(1), 2013.