Semantics and Complexity of RDF Stream Processing & Reasoning: Expression of Interest*

Harald Beck, Minh Dao-Tran, and Thomas Eiter

Institute of Information Systems, Vienna University of Technology
Favoritenstraße 9-11, A-1040 Vienna, Austria
{beck,dao,eiter}@kr.tuwien.ac.at

Abstract. Our expertise is in advanced model-based reasoning techniques to deal with incomplete information. We are interested in extending these techniques to handle streaming data. Our current project aims at establishing a strong theoretical foundation for stream reasoning. An important result of this project is LARS, a Logic-based framework for Analyzing Reasoning over Streams. We intend to utilize and extend LARS to clarify the operational semantics of off-the-shelf RSP engines in order to formally compare them and examine the computational cost of various RDF stream processing/reasoning features.

Authors' expertise. Our background is in the computational foundations of knowledge-based systems, in particular, reasoning systems and query languages. One of our focuses is the study of advanced model-based reasoning techniques that deal with incomplete information. We investigate the computational complexity of reasoning tasks and turn theoretical insights into efficient algorithms.

Research interests. We are interested in *extending advanced reasoning formalisms to handle streaming data*. We believe that insights obtained from data streams will get more interesting when logical *reasoning* can be applied. We think that stream processing should (a) be based on a fully declarative semantics, (b) allow for deriving new information (as in Complex Event Processing), beyond filtering and aggregation, and (c) potentially feature model generation and nonmonotonic negation (as in Answer Set Programming [5]).

We are currently working on a project called "Distributed Heterogeneous Stream Reasoning"[1] that aims at (i) establishing a strong theoretical foundation for stream reasoning, (ii) developing and optimizing algorithms for a practical realization of the theory, and (iii) building a prototype implementation as well as an evaluation toolbox for benchmarking stream processing/reasoning engines. Our ongoing efforts in these directions shall contribute to enable novel applications where complex reasoning on data streams is essential.

Relation to RSP community. Towards a theory for stream reasoning, we recently proposed LARS [4], a **L**ogic-based framework for **A**nalyzing **R**easoning

* Supported by the Austrian Science Fund (FWF) project P26471.

[1] kr.tuwien.ac.at/research/projects/dhsr/

over **S**treams, and showed how it captures CQL [2] (which underlies SPARQL extensions). We also studied the relation to the complex event processing system ETALIS [1] which is based on time intervals rather than time points.

We intend to utilize and extend LARS to clarify the operational semantics of different proposed RSP engines such as CQELS [9], C-SPARQL [3], and $\text{SPARQL}_{\text{Stream}}$ [6], based on which a formal comparison beyond [10, 11, 7] can be realized. That is to say, we aim for theoretical clarity where the *semantics* of these approaches agree, where they diverge, and in which way. In particular, this requires to *formally* understand the difference between pull-based vs. push-based querying. Moreover, we want to examine the computational cost of various RDF stream processing/reasoning *features*.

We hope to learn more on the demands of the RSP community in terms of use cases and required reasoning facilities. Based on this, we will tackle relevant theoretical questions towards recommendations for the standardization process.

Relevance to the broader ESWC audience. Work on its formal semantics [8] had a strong influence in the development and impact of SPARQL. We think that in the stream setting, a theoretical underpinning is no less important.

References

1. Darko Anicic, Sebastian Rudolph, Paul Fodor, and Nenad Stojanovic. Stream reasoning and complex event processing in ETALIS. *Semantic Web Journal*, 2012.
2. Arvind Arasu, Shivnath Babu, and Jennifer Widom. The CQL continuous query language: semantic foundations and query execution. *VLDB J.*, 15(2):121–142, 2006.
3. 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 Computing*, 4(1):3–25, 2010.
4. Harald Beck, Minh Dao-Tran, Thomas Eiter, and Michael Fink. LARS: A logic-based framework for analyzing reasoning over streams. In *AAAI*, 2015.
5. Gerd Brewka, Thomas Eiter, and Miroslaw Truszczyński. Answer set programming at a glance. *Communications of the ACM*, 54(12):92–103, 2011.
6. Jean-Paul Calbimonte, Óscar Corcho, and Alasdair J. G. Gray. Enabling ontology-based access to streaming data sources. In *ISWC (1)*, pages 96–111, 2010.
7. Daniele Dell'Aglio, Jean-Paul Calbimonte, Marco Balduini, Óscar Corcho, and Emanuele Della Valle. On Correctness in RDF Stream Processor Benchmarking. In *ISWC 2013*, pages 326–342, 2013.
8. Jorge Pérez, Marcelo Arenas, and Claudio Gutierrez. Semantics and complexity of SPARQL. *ACM Trans. Database Syst.*, 34(3), 2009.
9. 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 (1)*, pages 370–388, 2011.
10. Danh Le Phuoc, Minh Dao-Tran, Minh-Duc Pham, Peter Boncz, Thomas Eiter, and Michael Fink. Linked stream data processing engines: Facts and figures. In *ISWC - ET*, pages 300–312, 2012.
11. Ying Zhang, P. Minh Duc, O. Corcho, and J. P. Calbimonte. SRBench: A Streaming RDF/SPARQL Benchmark. In *ISWC*, 2012.