Provenance Management for SPARQL Updates

Argyro Avgoustaki [1, 2], Giorgos Flouris [2], Irini Fundulaki [2]

{argiro, fgeo, fundul}@ics.forth.gr

[1] Computer Science Department, University of Crete

[2] Institute of Computer Science, FORTH

http://www.csd.uoc.gr

http://www.ics.forth.gr

Motivation

In the last few years we have witnessed an explosion in the publication of data in the form of Linked Data. Recording the provenance information of Linked Data is an essential task in order to effectively support trustworthiness, accountability and repeatability. In this context, our work:

- Introduces a new provenance model for SPARQL INSERT Updates
- Allows the **reconstructability** of SPARQL INSERT Updates from their provenance
- Provides algorithmic support via the **Provenance Construction** and the **Update Reconstruction** algorithms

Model Features

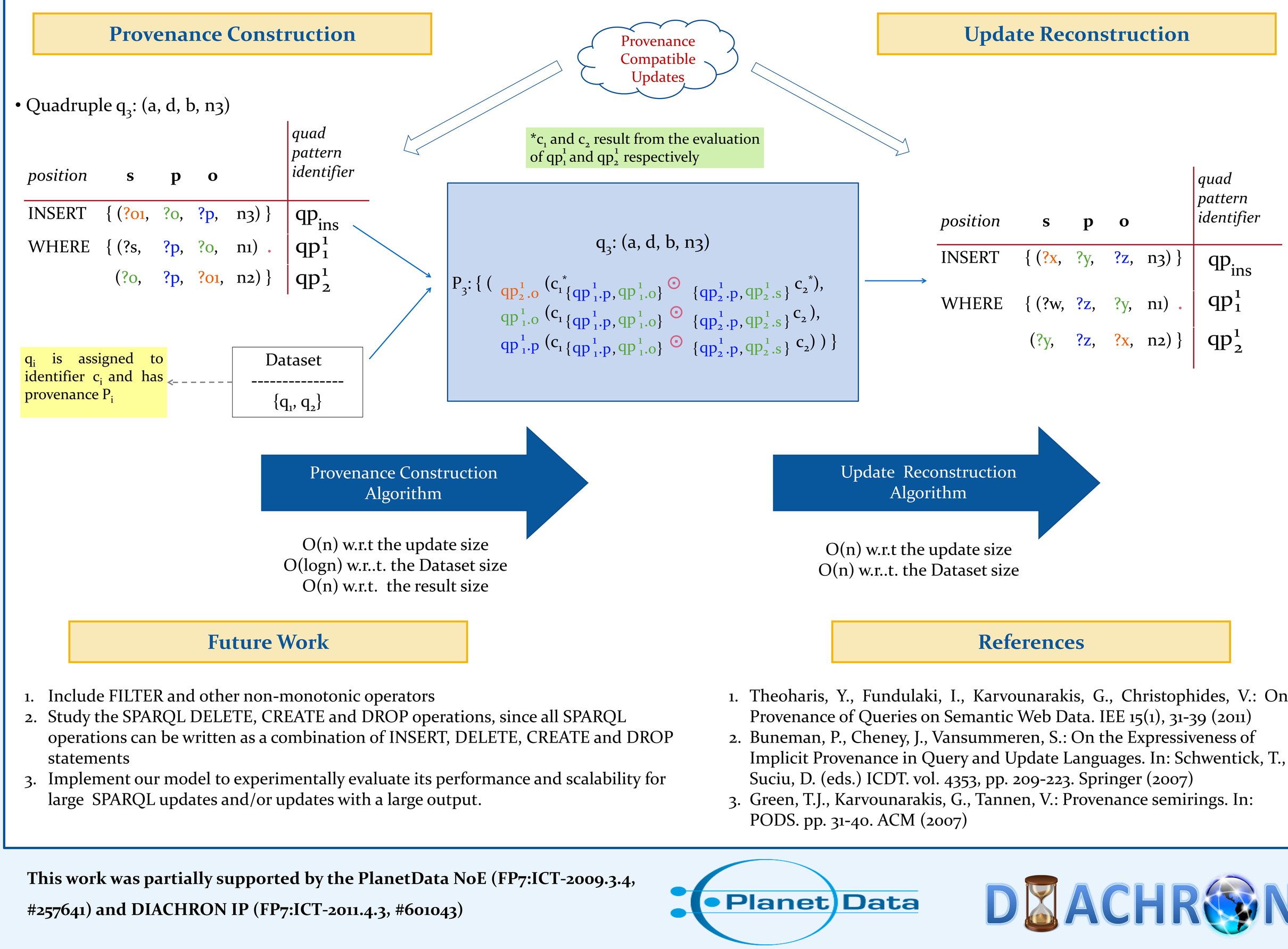
- ❑ suitable for encoding the *triple* and *attribute level provenance* of RDF quadruples
- ❑ uses complex algebraic expressions.
- ❑ represents the provenance of a quadruple as a set of *provenance expressions* indicating that the quadruple can be produced by different updates
- ❑ based on *how* and *where* provenance models
- ❑ supports unions of basic graph patterns

Provenance Model

	Quadruple (q) s	p	o	n	Provenance (P)	Case
c_1	a	b	d	n1	$P_1 = \{ (\bot, \bot, \bot) \}$	Attributes derived from const values
c_2	d	b	a	n2	$P_2 = \{ (qp^1_1.o(c_1), qp^1_1.p(c_1), qp^1_1.s(c_1)) \}$	Attributes derived by ``copying'' values
c_3	a	d	b	n3	$P_3 = \{ (qp^1_2.o(c_1 \{qp^1_1.p, qp^1_1.o\} \odot \{qp^1_2.p, qp^1_2.o\} c_2), qp^1_1.o(c_1 \{qp^1_1.p, qp^1_1.o\} \odot \{qp^1_2.p, qp^1_2.o\} c_2), qp^1_1.p(c_1 \{qp^1_1.p, qp^1_1.o\} \odot \{qp^1_2.p, qp^1_2.o\} c_2)) \}$	Attributes derived by ``copying'' values generated via joins
c_4	a	d	b	n4	$P_4 = \{ ((qp^1_1.s(c_3), qp^1_1.p(c_3), qp^1_1.o(c_3)) \oplus ((qp^2_1.o(c_2), qp^2_1.s(c_2), qp^2_1.p(c_2))) \}$	Attributes derived by ``copying'' values from two different sources

quadruple identifier

Provenance Construction



Update Reconstruction

Future Work

1. Include FILTER and other non-monotonic operators
2. Study the SPARQL DELETE, CREATE and DROP operations, since all SPARQL operations can be written as a combination of INSERT, DELETE, CREATE and DROP statements
3. Implement our model to experimentally evaluate its performance and scalability for large SPARQL updates and/or updates with a large output.

References

1. Theoharis, Y., Fundulaki, I., Karvounarakis, G., Christophides, V.: On Provenance of Queries on Semantic Web Data. IEE 15(1), 31-39 (2011)
2. Buneman, P., Cheney, J., Vansummeren, S.: On the Expressiveness of Implicit Provenance in Query and Update Languages. In: Schwentick, T., Suciu, D. (eds.) ICDT. vol. 4353, pp. 209-223. Springer (2007)
3. Green, T.J., Karvounarakis, G., Tannen, V.: Provenance semirings. In: PODS. pp. 31-40. ACM (2007)

This work was partially supported by the PlanetData NoE (FP7:ICT-2009.3.4, #257641) and DIACHRON IP (FP7:ICT-2011.4.3, #601043)