



Figure 4: Evaluation of INSERT DATA and DELETE DATA queries with incremental and greedy reasoning, with and without named graphs.

SPARQL queries for insertion/deletion (update) and selection of instances, and compare processing times in different situations.

6. CONCLUSION

In this paper we present HyLAR+, a comprehensive reasoning framework that can perform reasoning tasks on both Web servers and users' clients. HyLAR+ is an improvement of a previous version and is designed to target Web applications in which some parts of data collection and processing are performed on the client side. Such applications usually handle small datasets and need to be reactive to data update and retrieval requests. To do so, HyLAR+ embeds a custom JavaScript rule-based reasoner that implements an incremental reasoning algorithm. HyLAR+ also supports named graphs and ships with a built-in set of OWL2-RL constructs. This rule set can be extended using first-order logic rules, either to improve the OWL2-RL specification coverage or to add domain-specific behaviors to an application. We present two evaluations that measure the reasoner efficiency regarding incremental reasoning and named graphs, for classification and update reasoning tasks.

During the demo, we will show the behavior of our reasoner with different settings and illustrate how to use HyLAR+ for importing ontologies from files and sending SPARQL INSERT DATA, DELETE DATA and SELECT queries. We will also show an example of adaptation decision heuristics to decide whether a query should be performed on the server or the client side.

The perspectives of our work are the following. We first want to make our framework compatible with the newly issued CHRjs⁹ rule-based constraint solver, in order to compare our reasoner to a CHR-based one. Regarding the rea-

⁹<http://chrjs.net/>

soner efficiency, we are working on still improving incremental reasoning performances by avoiding its over-deletion and rederivation steps. Regarding the decision of locating a reasoning task, we intend to provide a wider set of out-of-the-box heuristics, a simple API to access them, as well as a set of benchmarks to help making decisions w.r.t. ontology size and number of rules. Using HyLAR+ and such extensions, Web developers will be easily able to leverage the power of semantic reasoning in small client-side applications.

Acknowledgement

This work is supported by the French ANR (Agence Nationale de la Recherche) under the grant number <ANR-13-INFR-012>.

7. REFERENCES

- [1] M. Acosta, M.-E. Vidal, T. Lampo, J. Castillo, and E. Ruckhaus. Anapsid: An adaptive query processing engine for sparql endpoints. In *The Semantic Web-ISWC 2011*, pages 18–34. Springer, 2011.
- [2] D. Estrin. Small data, where n= me. *Communications of the ACM*, 57(4):32–34, 2014.
- [3] D. Guinard and V. Trifa. Towards the web of things: Web mashups for embedded devices. In *Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web (MEM 2009)*, in proceedings of WWW (International World Wide Web Conferences), Madrid, Spain, page 15, 2009.
- [4] A. Gupta, I. S. Mumick, and V. S. Subrahmanian. Maintaining views incrementally. *ACM SIGMOD Record*, 22(2):157–166, 1993.
- [5] I. Horrocks and P. Patel-Schneider. Knowledge representation and reasoning on the semantic web: OWL, 2010.
- [6] M. Krötzsch. *OWL 2 Profiles: An introduction to lightweight ontology languages*. Springer, 2012.
- [7] B. Motik, B. C. Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz. Owl 2 web ontology language: Profiles. *W3C recommendation*, 27:61, 2009.
- [8] M. Terdjimi, L. Médini, and M. Mrissa. Hylar: Hybrid location-agnostic reasoning. In *ESWC Developers Workshop 2015*, page 1, 2015.
- [9] R. Verborgh, O. Hartig, B. De Meester, G. Haesendonck, L. De Vocht, M. Vander Sande, R. Cyganiak, P. Colpaert, E. Mannens, and R. Van de Walle. Querying datasets on the web with high availability. In *The Semantic Web-ISWC 2014*, pages 180–196. Springer, 2014.