

# Triplesets: Tagging and Grouping in RDF Datasets

Atanas Kiryakov, Vassil Momtchev

Ontotext AD, 135 Tsarigradsko Chaussee, Sofia 1784, Bulgaria

{first.second}@ontotext.com

## Abstract

The need to augment the standard triple-based RDF data model towards quadruples is widely recognized. The most popular of such extensions is the notion of *named graphs* (NG). One prominent use of NG is related to management of *datasets*, integrating information from multiple sources. Technically, this requires storage, modification, and querying of modularized RDF graphs in semantic repositories. For NG to serve this purpose properly there needs to be a clear definition for the semantics of operations like addition and removal of quadruples from such datasets. While there is a consensus about the NG name, there is no single standard specification of the NG model, nor a specification of the abovementioned semantics.

Here we propose semantics for the operations addition and removal of  $\langle S, P, O, G \rangle$  quadruples from integrated datasets. We also motivate the need to further extend the RDF model and propose a specific mechanism called *triplesets*. The proposed mechanism is already supported in the OWLIM repository and it is used for large scale reasoning in LarKC and other projects. Finally, we outline the need of further standardization work related to this proposal with regard to serialization and querying.

## 1 Named Graphs – the Current State of Affairs

Resource Description Framework (RDF), [6], is a language for representing information about resources in the World Wide Web. Although it was designed to represent metadata about Web resources, RDF has much broader use as a generic data model for structured data management and reasoning.

*Named graph*, [3], is an RDF graph with a URI, [1], assigned as a name. In an extended RDF model one can deal simultaneously with multiple named graphs and make statements about them, by putting their URI name in subject position. A more concrete definition is provided in the specification of SPARQL, [8], where queries are evaluated against datasets, composed from multiple RDF graphs. In SPARQL, an *RDF Dataset* is defined as

$$\{ G, (\langle U1 \rangle, G1), (\langle U2 \rangle, G2), \dots (\langle Un \rangle, Gn) \}$$

where  $G$  and each  $G_i$  are RDF graphs, and each  $\langle U_i \rangle$  is a distinct IRI (an internationalized URI). The pairs  $(\langle U_i \rangle, G_i)$  are called *named graphs*, where  $\langle U_i \rangle$  is the name of the graph

gi.  $\mathbf{G}$  is called a *default graph* – it contains all triples, which belong to the dataset, but not to any specific named graph. The notion of a default graph is not present in [6].

## 2 Datasets

The SPARQL notion of the RDF dataset can be adopted and extended as a formal model for integrating information from multiple sources in RDF-based semantic repositories and other data management infrastructures. Intuitively, a *dataset* integrates several RDF graphs in such a way that each graph can be distinguished, manipulated, and addressed separately. In the RDF data model, extended with named graphs, the statements in a dataset are either parts of specific named graphs or belong to the default graph. In the case of data integration, NG can be used to model provenance and modularization: each named graph contains statements that represent a specific body of information, coming from a specific source. Such approach is taken in the Linked Data Semantic Repository, which integrates several of the central Linking Open Data, [10], datasets as described in [5].

Formally, a dataset can be represented as an RDF multi-graph, which, in its turn, can be represented as a set of quadruples of the following type:  $\langle \mathbf{s}, \mathbf{p}, \mathbf{o}, \mathbf{G} \rangle$ . Those can be seen as “contextualized statements”, where the first three elements of the quadruple,  $\langle \mathbf{s}, \mathbf{p}, \mathbf{o} \rangle$ , represent an RDF statement and the fourth element,  $\mathbf{G}$ , represents the name of the named graph the statement belongs to. Figure 1 presents such a dataset, where the statements are depicted in colours, specific for the NG they belong to.

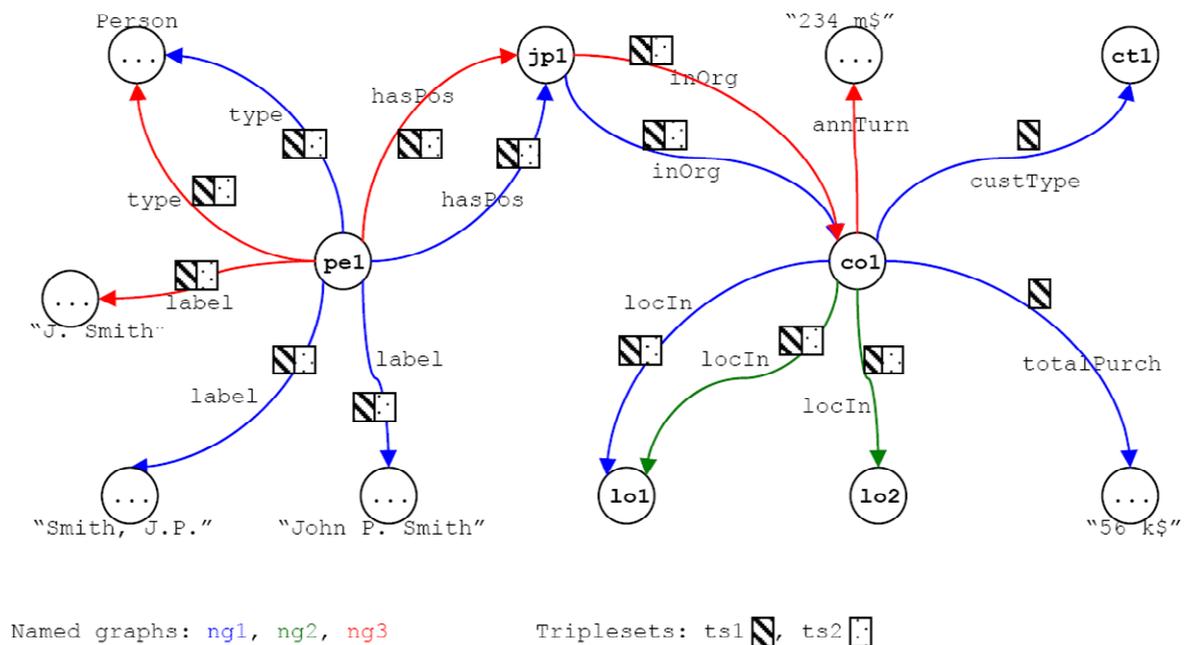


Figure 1. RDF Graph with Named Graphs and Triplesets

A dataset can be seen as an RDF multi-graph, because multiple arcs, labelled with one and the same predicate, can connect the same pair of nodes in the graph; such an example is the couple of arcs from  $pe1$  to  $Person$  in the upper-left hand side of the figure.

Alternatively, datasets can also be presented as sets of triples of  $\langle \mathbf{s}, \langle \mathbf{p}, \mathbf{NG} \rangle, \mathbf{o} \rangle$ , which would correspond to an RDF graph, where the arcs are labelled not simply with a predicate, but with

a pair of predicate and named graph. While such a representation would also be correct, we will stick with the former one, as we believe that it better corresponds to the scenario of managing integrated data and that it is more backward compatible with the original RDF graph notion.

### 3 The Semantics of Addition and Removal of Quadruples

Neither [6] nor [8] provide sufficient formal grounds for the semantics of NG in order to determine the behaviour of a semantic repository that supports such an extended RDF model. As SPARQL (ver. 1.0) supports no data modification, it is unclear what the formal consequences of adding or removing a statement from a named graph should be. Counting statements in a SPARQL dataset is also not specified. Aiming to fill this gap, the specification of the second generation of the ORDI framework, [7], defined these aspects of the named graphs semantics; and this is also the proposal that we elaborate on in this position paper.

Essentially, the semantics of these operations reflects the addition and removal of arcs in the RDF multi-graph depicted on Figure 1. Whenever an  $\langle s, p, o, g \rangle$  statement is added or removed from a dataset, the number of statements increases or decreases. The statements from the different named graphs count as independent facts. In such a model, updating one named graph (one module of information) in the dataset with a newer version of its contents, which contains, for instance, 5 more statements, will result in a dataset that is also 5 statements larger; disregarding whether some of the statements appear also in other NG. This semantics allows for fine-grained distinctions, supporting easy monitoring and manipulation of datasets integrated from multiple sources. Semantic repositories, based on such dataset model, can also easily simulate NG ignorant behaviour, e.g. count and manage statements disregarding their affiliation. Thus, the semantics proposed here allows the semantic repositories to support both provenance-aware and provenance-agnostic behaviour.

### 4 Triplesets

Comprehensive management of large integrated RDF datasets often requires a mechanism that allows one to designate, describe, and manage parts of such datasets. Here follow just a few such scenarios:

- Fine-grained access control and tracking of changes;
- Passing intermediate results between reasoning components in workflows for incomplete reasoning, as the ones in the architecture of LarKC, [4];
- RDF graph priming for context-aware reasoning and query evaluation, [9].

The dataset parts that one needs to deal with in such scenarios are independent from the NG used to handle provenance, in the manner described in section 2. Those are rather sub-graphs of the RDF multi-graph or subsets of the set of quadruples, representing the content of the dataset. The mechanism needed for such purposes should allow easy creation and re-organisation of such groups, without changing the content of the dataset. Once the semantics of NG is determined in the manner proposed in section 3, we need a different mechanism to fulfil such purposes. The main rationale is to implement a simple and query efficient extension to the triple and quadruple models that would enable grouping and tagging of the contents of the integrated RDF datasets, in a manner compatible with the existing RDF(S) and OWL semantics.

A *triple set*, as introduced in [7], is a mechanism to deal with parts of datasets or to group some of the statements in a dataset. An RDF dataset with named graphs and triple sets is depicted on Figure 1. The difference between named graphs and datasets can be explained as follows:

- Named graphs “own” the statements; e.g. each statement belongs to a specific named graph or to the default graph. When a statement is added or removed from a named graph, a particular arc appears or disappears from the multi-graph, which represents the datasets; respectively, the count of the arcs increases or decreases.
- Triple sets can be viewed as tags on the statements. When a statement is added to a triple set, this can be seen as an association operation, which does not add a new arc in the graph. When a statement is removed from a triple set, i.e. it is no longer a member of this group of statements, it does not disappear from the dataset, but is just un-tagged or disassociated.

Formally, the atomic element in the triple set model is a quintuple:

$$\langle S, P, O, G, \{TS_1, \dots, TS_n\} \rangle$$

where  $G$  is the name of a NG and  $\{TS_1, \dots, TS_n\}$  is a set of identifiers of the triple sets to which the contextualized statement  $\langle S, P, O, G \rangle$  is associated. Each statement (from each graph) can be a member of multiple triple sets. The content of a triple set is an RDF multi-graph, a subset of the set of all quadruples in the dataset. The names of the triple sets are URIs.

It is worth noting, that the above quintuple is provided for the sake of a formal specification of the semantics of the extended RDF data model. Semantic repositories can (and most of them do) implement alternative data representation and indexing structures, while supporting the same semantics.

## 5 Conclusion and Open Questions

The need for extending the RDF data model with triple sets is a result of the clear specification of the semantics reflecting addition and removal of statements from RDF datasets. Named graphs are used most often for tracking of provenance, for example, when multiple graphs from different sources are merged or referenced (e.g. when dealing with linked data). In such a scenario, strong “ownership” semantics should be enforced for the NG so that updating the contents of specific NG can have real impact on the contents of the dataset. Once named graphs are given such semantics, there is a need for a mechanism, which allows dealing with metadata about the contents of an integrated dataset. Triple sets are defined as a weaker mechanism for grouping quadruples (statements form a dataset) and assigning metadata to them. Moreover, since the triple sets allow for designation or tagging of dataset parts, they are especially useful when selecting parts of the dataset, e.g. in the course of multi-stage processing, where intermediate results should be passed from one component to another.

Triple sets are already supported by the BigOWLIM semantic repository, following the specification in [7]. They are also a standard feature of the LarKC data layer (see section 5.1 in [4]). Using extensions of the RDF model, different from the Named Graphs, have been recognized also by other semantic repository vendors; this is the case with the “models” used in the RDF support of ORACLE, [11].

To make triplesets first-order citizens of the RDF world, there should be a syntax allowing for human readable serialisation of datasets, which preserves the tripliset affiliation. An approach about extending the TRIG syntax, [2], to support triplesets is proposed in [7]; still, more work is required to provide a proper serialisation specification. There is also a need for extending SPARQL in a way that allows triplesets to be used for retrieval and filtering purposes through it.

## References

1. Berners-Lee, T., Fielding R., Masinter L.: *Uniform Resource Identifier (URI): Generic Syntax*. Network Working Group, Request for Comments: 3986, January 2005, <http://tools.ietf.org/html/rfc3986> (2005)
2. Bizer, Chris. 2005. *The TriG Syntax*. <http://sites.wiwiss.fu-berlin.de/suhl/bizer/TriG/>
3. Carroll, J. J; Bizer, B; Hayes, P; Stickler, P. 2005. *Named Graphs, Provenance and Trust*. WWW2005, <http://www2005.org/cdrom/docs/p613.pdf>
4. Kerrigan, M., Bradesko, L., Fortuna B.: *Rapid Prototype of the LarKC*. LarKC project deliverable D5.2.1, [http://www.larkc.eu/wp-content/uploads/2008/01/larkc\\_d521\\_rapid-prototype-of-the-larkc.pdf](http://www.larkc.eu/wp-content/uploads/2008/01/larkc_d521_rapid-prototype-of-the-larkc.pdf), (2009)
5. Kiryakov, A., Tashev, Z., Ognyanoff, D., Velkov, R., Momtchev, V., Balev, B., Peikov, I.: *Validation goals and metrics for the LarKC platform*. LarKC project deliverable D5.5.2. <http://www.larkc.eu/deliverables/> (2009)
6. Manola F., Miller, E. (eds.): *RDF Primer*. W3C Recommendation, 10 Feb 2004, <http://www.w3.org/TR/rdf-primer/>, (2004)
7. Momtchev, V., Kiryakov, A.: *Second Generation Ontology Representation and Data integration (ORDI) Framework, Specification*. Ontotext technical documentation, 13 Oct 2006. [http://www.ontotext.com/ordi/ORDI\\_SG/ORDI\\_SG\\_Specification.pdf](http://www.ontotext.com/ordi/ORDI_SG/ORDI_SG_Specification.pdf) (2006)
8. Prud'hommeaux, E., Seaborne, A: *SPARQL Query Language for RDF*, W3C Recommendation 15 January 2008, <http://www.w3.org/TR/rdf-sparql-query/>, (2008)
9. Todorova, P., Kiryakov, A., Ognyanoff, D., Peikov, I., Velkov, R., Tashev, Z.: *Spreading Activation Components*. LarKC project deliverable D2.4.1, (2009)
10. World Wide Web Consortium (W3C): *Linking Open Data*. W3C SWEO community project home page, as of January 2010. <http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData> (2010)
11. Wu, A.; Lopez, X.: *Building Enterprise Applications With Oracle Database 11g Semantic Technologies*. Presentation at Semantic Technologies Conference, San Jose (2009)