

From: Gregg Kellogg gregg@greggkellogg.net
Subject: Position Statement for the W3C Workshop on Web Standardization for Graph Data
Date: December 3, 2018 at 2:12 PM
To: group-data-ws-pc@w3.org



JSON-LD 1.1 is currently the only RDF syntax undergoing active development [1], and many notions of merging the RDF data model with those of other property graphs may be explored in that context. GraphQL [2] and property graphs, such as Apache TinkerPop [3], have also gathered a lot of support from the industry, but don't fit well with RDF. Blazegraph has addressed this through RDF extensions for considering triples as RDF resources, allowing a statement to appear as the subject or object in a triple with the RDF*/SPARQL* extensions [4]. However, RDF datasets address much the same problem.

Since 1.0, JSON-LD has been a dataset serialization language, with support for anonymous named graphs [5]. In the 1.1 syntax, these can be made even more transparent using `_graph containers_`.

For example, the Turtle* example:

```
@prefix : <http://bigdata.com> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix dct: <http://purl.org/dc/elements/1.1/> .

:bob foaf:name "Bob" .
<<:bob foaf:age 23>> dct:creator <http://example.com/crawlers#c1> ;
    dct:source <http://example.net/homepage-listing.html> .
```

Might be represented using named graphs in JSON-LD as:

```
{
  "@context": {
    "@base": "http://bigdata.com",
    "foaf": "http://xmlns.com/foaf/0.1/",
    "dct": "http://purl.org/dc/elements/1.1/"
  },
  "@graph": [
    {
      "@id": "bob",
      "foaf:name": "Bob"
    },
    {
      "dct:creator": { "@id": "http://example.com/crawlers#c1" },
      "dct:source": { "@id": "http://example.net/homepage-listing.html" },
      "@graph": {
        "@id": "bob",
        "foaf:age": 23
      }
    }
  ]
}
```

This results in the following quads:

```
<http://bigdata.com/bob> <http://xmlns.com/foaf/0.1/age> "23"^^<http://www.w3.org/2001/XMLSchema#integer> _:b0 .
<http://bigdata.com/bob> <http://xmlns.com/foaf/0.1/name> "Bob" .
_:b0 <http://purl.org/dc/elements/1.1/creator> <http://example.com/crawlers#c1> .
_:b0 <http://purl.org/dc/elements/1.1/source> <http://example.net/homepage-listing.html> .
```

By spreading the information across different graphs, we can create a BNode to represent the graph name of a graph containing the statement `:bob foaf:age 23`, and use that as the subject for the annotations on that graph for `dct:creator` and `dct:source`. This is not necessarily how many people use named graphs in RDF, but RDF 1.1 makes no statements on the semantics of named graphs, and this is a perfectly consistent way of using them.

Similarly, using the `schema:Role` pattern, you might consider the following JSON-LD:

```
{
  "@context": {
    "@base": "http://bigdata.com",
    "foaf": "http://xmlns.com/foaf/0.1/",
    "dct": "http://purl.org/dc/elements/1.1/",
    "schema": "http://schema.org"
  },
  "@id": "bob",
  "foaf:name": "Bob",
  "foaf:age": {
    "@type": "schema:Role",
    "dct:creator": { "@id": "http://example.com/crawlers#c1" },
    "dct:source": { "@id": "http://example.net/homepage-listing.html" }
  }
}
```

```
    "creator": { "@id": "http://example.com/creators/01",  
    "dct:source": { "@id": "http://example.net/homepage-listing.html"},  
    "foaf:age": 23  
  }  
}
```

This pattern uses an intermediate object to represent information about the foaf:age relationship. Something more normative could be included in a hypothetical RDF 2.0.

Lastly, a feature parallel and similar to named graphs could be considered (an alternate to the @graph keyword in JSON-LD) for statement reification, allowing statements to remain in the same graph (default or named).

Gregg Kellogg
gregg@greggkellogg.net

- [1] <https://www.w3.org/2018/json-ld-wg/>
- [2] <https://code.fb.com/core-data/graphql-a-data-query-language/>
- [3] <http://tinkerpop.apache.org>
- [4] https://wiki.blazegraph.com/wiki/index.php/Reification_Done_Right
- [5] <https://www.w3.org/TR/json-ld11/#graph-containers>