NG4J – Named Graphs API for Jena

### Named Graphs

Named Graphs are a simple extension to the RDF abstract syntax and enable statements to be made about RDF graphs. This is beneficial for applications like data syndication, provenance tracking, ontology versioning, signing RDF, and access control to parts of RDF repositories. Named Graphs form the basis for SPARQL, W3C’s upcoming RDF query language.

### NG4J

The Named Graphs API for Jena is an implementation of Named Graphs on top of the Jena Semantic Web Framework. NG4J extends Jena with:

- graph- and quad-centric methods for manipulating sets of Named Graphs
- Jena Model view on a graphset’s merge
- provenance-enabled Jena Statements
- TriQL query language for queries across graphs
- parsers and serializers for the TriG and TriX syntaxes, including support for TriX syntactic extensions with XSLT
- database-persisted graphset storage
- API support for the Semantic Web Publishing (SWP) vocabulary
- creation and verification of graph signatures using X.509 certificates

### TriG

A serialization format for sets of Named Graphs, based on Turtle

```turtle
@prefix swp: <http://www.w3.org/.../swp-2/>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
@prefix : <http://example.org/>.

:G1 { 
  :Bob foaf:mbox <mailto:bob@example.org>. 
} 

:G2 { 
  :Bob foaf:mbox <mailto:bob2@example.com>. 
} 

:warrant { 
  :warrant a swp:Warrant. 
  :warrant swp:authority :Alice. 
  :warrant swp:assertedBy :warrant. 
  :G1 swp:assertedBy :warrant. 
  :G1 swp:digestMethod swp:JjcRdfC14N-sha1. 
  :G1 swp:digest "YjRhNz..."^^xsd:base64Binary. 
  :G2 swp:assertedBy :warrant. 
  :G2 swp:digest "NmM2NW..."^^xsd:base64Binary. 
} 
```

### TriX

A serialization format for RDF and sets of Named Graphs that plays well with XML tools. An XSLT-based extension mechanism can be used to define more concise variations of the syntax.

```xml
<TriX xmlns="http://www.w3.org/2004/03/trix/trix-1/">
  <graph>
    <uri>http://example.org/graph1</uri>
    <graph>
      <plainLiteral>"A Good Book"</plainLiteral>
    </graph>
  </graph>
  <graph>
    <uri>http://example.org/graph2</uri>
    <graph>
      <uri>http://example.org/books.rdf</uri>
      <triple>
        <uri>http://example.org/source</uri>
        <uri>http://example.org/aBook</uri>
        <uri>http://example.org/title</uri>
        <graph>
          <uri>http://example.org/books.rdf</uri>
          <triple>
            <uri>http://example.org/graph1</uri>
            <graph>
              "http://www.w3.org/2004/03/trix/trix-1/">
              <graph>
                <plainLiteral>"A Good Book"</plainLiteral>
              </graph>
            </graph>
          </graph>
        </triple>
      </graph>
    </graph>
  </graph>
</TriX>
```

### TriQL

A language for queries across Named Graphs, based on RDQL. The example fetches all email addresses of Bob that have been asserted by Alice in 2005, with the respective graph names. In NG4J, TriQL queries are executed against instances of the NamedGraphSet class.

```sql
SELECT ?email, ?source WHERE { 
  ?source ex:hasMailbox ?email . 
  ?source ex:assertedBy :warrant. 
  :warrant swp:authority :Alice. 
  :warrant swp:assertedBy ?warrant. 
  AND ?date >= "2005-01-01"^^xsd:date 
  AND ?date < "2006-01-01"^^xsd:date 
} USING swp FOR <http://example.org/> 
foaf FOR <http://xmlns.com/foaf/0.1/> 
swp FOR <http://www.w3.org/.../swp-2/> 
do FOR <http://purl.org/dc/elements/1.1/> 
```

Further information about NG4J and download: http://www.wiwiss.fu-berlin.de/suhl/bizer/ng4j/