Bridging RDF Graph and Property Graph Data Models
- LDBC 2016

Zhe Wu
Ph.D., Architect
Oracle Spatial and Graph

June, 2016
Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Overview of Graph

• What is a graph?
  – A set of vertices and edges (with optional properties)
  – A graph is simply **linked data**

• Why do we care?
  – Graphs are everywhere
    • Road networks, power grids, biological networks
    • Social networks/Social Web (Facebook, Linkedin, Twitter, Baidu, Google+,…)
    • Knowledge graphs (RDF, OWL)
  – Graphs are intuitive and flexible
    • Easy to navigate, easy to form a path, natural to visualize
    • Do not require a predefined schema
Enable Spatial and Graph use cases on every platform

Oracle’s Graph Strategy

Oracle Big Data Spatial and Graph

NoSQL

Big Data: Single Model Data Store

Database 12c: Polyglot (Multi-model) Data Store

Oracle Database Spatial and Graph

Spatial and Graph in Cloud Offerings
Direction of Development in Graph & Semantics Area

- RDF, OWL, SPARQL
- Property Graph
- Other Data Types
Direction of Development in Graph & Semantics Area
“Facets”

Security
User Interface
Programming interfaces
RDF, OWL, SPARQL
Property Graph
JSON, Spatial, Big Data, Relational
Scalability
Multiple Platforms
Tools
Fashion!
Solution
Searchability
RDF Graph Data Model

- Resource Description Framework
  - **URIs** are used to identify
    - Resources, entities, relationships, concepts
    - Data identification is a *must* for integration

- RDF Graph defines semantics

- **Standards** defined by W3C & OGC
  - RDF, RDFS, OWL, SKOS
  - SPARQL, RDFa, RDB2RDF, GeoSPARQL

- Implementations
  - Oracle, IBM, Cray, Systap
  - Franz, Ontotext, Openlink, Jena, Sesame, …
Property Graph Data Model

- **A set of vertices (or nodes)**
  - each vertex has a unique identifier.
  - each vertex has a set of in/out edges.
  - each vertex has a collection of key-value properties.

- **A set of edges**
  - each edge has a unique identifier.
  - each edge has a head/tail vertex.
  - each edge has a label denoting type of relationship between two vertices.
  - each edge has a collection of key-value properties.

- **Blueprints Java APIs**

- **Implementations**
  - Oracle, Neo4j, DataStax(Titan), InfiniteGraph, Dex, Sail, MongoDB ...

https://github.com/tinkerpop/blueprints/wiki/Property-Graph-Model
2 Graph Data Management & Analysis Products
Property Graph & RDF Graph

Property Graph Model
- Graph Search & Analysis
- Big Data analytics
- Entity analytics

RDF Data Model
- Data Integration
- Knowledge representation
- Inferencing

Link Analysis
- National Intelligence
- Public Safety
- Social Media search
- Marketing - Sentiment

Data Integration Semantic Web
- Life Sciences
- Health Care
- Publishing
- Finance

Application Area
Graph Model
Industry Domain
RDF Graph Support
Oracle Spatial & Graph 12c RDF Semantic Graph

- Oracle Exadata Database Machine ready
- Compression & partitioning
- Parallelism: load, inference, query
- High availability
- Manageability
- Performance
https://www.w3.org/wiki/LargeTripleStores
- Label security: triple-level
- Partners: ISVs, SIs, reasoners, ontologies
- W3C standards compliance
  - RDF, SPARQL, OWL, GeoSPARQL, RDB2RDF, SKOS

Load / Storage
- RDF graph triple/quad store
- Manages trillions of triples
- Optimized storage architecture
- B-tree indexing

Query
- SPARQL-Jena /Fuseki /Joseki
- SQL/graph query
- RDF Views on table data
- Semantic indexing framework
- Ontology assisted SQL query

Reasoning
- Forward-chaining, persistent, native
- Incremental & parallel reasoning
- RDFS, OWL2 RL, EL, SKOS
- User-defined rules & inferencing
- Secure (ladder-based) inferencing
- Plug-in architecture: TROWL, Pellet...

Tools & Analytics
- Visualization: Cytoscape & Commercial
- Ontology editing: Protégé & Commercial
- Reporting: OBIEE
- Analytics: Oracle Advanced Analytics
World’s Fastest Big Data Graph Benchmark
1 Trillion Triple RDF Benchmark with Oracle Spatial and Graph

- World’s fastest data loading performance
- World’s fastest query performance
- Worlds fastest inference performance
- Massive scalability: 1.08 trillion edges

- **Platform:** Oracle Exadata X4-2 Database Machine
- **Source:** w3.org/wiki/LargeTripleStores, 9/26/2014

Oracle Database 12c can load, query and inference millions of RDF graph edges per second

Millions of triples per second

<table>
<thead>
<tr>
<th></th>
<th>Query</th>
<th>Load</th>
<th>Inference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.13</td>
<td>1.42</td>
<td>1.52</td>
</tr>
</tbody>
</table>
Property Graph Support
Architecture of Property Graph Support

Graph Analytics
- Parallel In-Memory Graph Analytics (PGX)

Graph Data Access Layer (DAL)
- Blueprints & Lucene/SolrCloud

Scalable and Persistent Storage Management
- Oracle RDBMS
- Apache HBase
- Oracle NoSQL Database

Java APIs

Java APIs/JDBC/SQL/PLSQL

REST/Web Service

Java, Groovy, Python, ...

Property Graph formats
- GraphML
- GML
- Graph-SON
- Flat Files

RDF (RDF/XML, N-Triples, N-Quads, TriG, N3, JSON)
Oracle’s In-Memory Analyst vs Spark GraphX 1.1

In-Memory Analyst on 1 node is up to 2 orders of magnitude faster than Spark GraphX distributed execution on 2 to 16 nodes.
In-Memory Analyst on a single machine is 3x – 10x faster than a GraphLab 16-machine distributed execution
Linear Scalability Loading in NoSQL w/ Parallelism

Oracle Big Data Spatial and Graph: Property Graph – Data Access
Oracle NoSQL Database: Linear Scalability of Data Loading
(Degrees of Parallelism (DOP) = 36)

Data points for 3M, 70M, 1.5B, & 3B Edges
4-6 Seconds for Analytics on 4.8m Vertices w/ 68.9m Edges (2.9 GB) w/ Parallel In-Memory Analyst

Oracle Big Data Spatial and Graph: Property Graph - In-Memory Analyst
Apache HBase 1.0: Parallel Graph Analytics on LiveJ Data

Count triangles

Page Ranking

Copyright © 2016 Oracle and/or its affiliates. All rights reserved.
<table>
<thead>
<tr>
<th>Strengths and Weaknesses</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Semantic web/RDF graph</strong></td>
<td><strong>Property graph</strong></td>
</tr>
<tr>
<td>• Formal theoretical foundation, precise, lots of standards/curated terms/vocabularies, linked data</td>
<td>• Easy to learn (actually not much to learn)</td>
</tr>
<tr>
<td>• Steep learning curve</td>
<td>• Suitable for social network analysis</td>
</tr>
<tr>
<td>• Hidden complexity</td>
<td><strong>Property Graph</strong></td>
</tr>
<tr>
<td></td>
<td>• Lack of a standard query language</td>
</tr>
<tr>
<td></td>
<td>• Hard to deal with multiple property graphs</td>
</tr>
</tbody>
</table>
Query Languages for RDF Graph and Property Graph

RDF Graphs

• Standard query language:
  – W3C **SPARQL 1.1**

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?mbox
WHERE
  {?x foaf:name ?name .
   ?x foaf:mbox ?mbox }
```

Property Graphs

• No standard query language
• Multiple languages proposals:
  – PGQL (Oracle)
  – Cypher (Neo4j)
  – Gremlin (Tinkerpop)
  – GraphQL (LDBC)
PGQL: a Property Graph Query Language

- Closer to SQL (compared to other proposals: Cypher, Gremlin)
- Shipped with Oracle Big Data Spatial and Graph v1.2.0

```
SELECT y.name, e, p
FROM snGraph
WHERE (x WITH name = 'Paul')-[e:likes]-> (y),
     (z WITH name = 'Amber')-/p:likes*/-> (y),
     x.age > y.age
GROUP BY
ORDER BY
LIMIT
OFFSET
```

Return a “result set”

Match a graph pattern
Reference Implementation of PGQL Parser

• Open-sourced
  – https://github.com/oracle/pgql-lang
  – Apache 2.0 + Universal Permissive License (UPL) 1.0

• Example usage:

```java
public static void main(String[] args) throws PqglException {
    Pqgl pgql = new Pqgl();
    PqglResult result1 = pgql.parse("SELECT x FROM myGraph WHERE (n:Person)");
    System.out.println(result1.getErrorMessages());

    PqglResult result2 = pgql.parse("SELECT n FROM myGraph WHERE (n:Person)");
    GraphQuery query = result2.getGraphQuery();
}
```
Intermediate Representation (IR) for Graph Queries

- IR is independent of parser implementations
  - Parsers can be developed independently of query engines
  - Syntax changes (to PGQL) do not break existing query engines
- Can potentially be used in combination with other graph query languages
Can an application make use of both graph data models?

Bridging RDF Graph and Property Graph
Semantic Web/RDF Graph Coexists with Property Graph

- Step 1: Stick them into the same repository
Semantic Web/RDF Graph Coexists with Property Graph

- Step 2: Force them to speak the same language (Java, SQL, REST, ...)

[Diagram of two silhouettes talking to each other]
Semantic Web/RDF Graph Coexists with Property Graph

• Step 3: Disguise one as the other
  • Property graph view on RDF & RDF view on property Graph
Property Graph View on RDF Data

• Specify
  • Which set of assertions become “attributes”
  • Which set of assertions become edges

ns:vertex1 ns:name “marko” .
ns:vertex1 ns:age 29 .
ns:vertex1 ns:created ns:vertex3 .
Property Graph View on RDF Data

• Specify
  • Which set of assertions become “attributes”
  • Which set of assertions become edges

ns:vertex1 ns:name "marko" .
ns:vertex1 ns:age 29 .
ns:vertex1 ns:created ns:vertex3 .

• Challenge: dealing with multiple values
RDF View on Property Graph Data

• Use W3C RDB2RDF
  • Property graph modeled with relational table

<table>
<thead>
<tr>
<th>VID</th>
<th>K</th>
<th>T</th>
<th>V</th>
<th>VN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>name</td>
<td>1</td>
<td>BOB</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>name</td>
<td>1</td>
<td>The Mona Lisa</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• Define an R2RML mapping

• Open question: can we add a bit of RDF to a PG graph?

Copyright © 2016 Oracle and/or its affiliates. All rights reserved.
Summary

• Under active development
  • Semantic web/RDF/OWL improvement
  • Property graph in Oracle RDBMS

• Common challenges for graph users
  • Lack of a standard property graph query language
  • Steep learning curve for RDF/OWL users

• RDF Graph and Property graph data models can be used together