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.
Demos

• Core Database Semantic Technologies Features
  – Load, Inference, Query

• Semantic Indexing of Documents
  – Lymba integration (Video available)

• Semantic Data Visualization/Editing
  – Tom Sawyer integration (Video available)
  – Cytoscape integration
  – TopBraid Composer

• OBIEE Integration with RDF Data

• Federated Query
Demo:
Core Database Semantic Technologies Features
Capabilities Overview of Release 11.2

NLP engines, Tools, Editors, Complete DL reasoners, ...

SQL/PLSQL APIs & JAVA APIs (Jena, Sesame)

**INFER**
- RDF/S
- OWL/SKOS
- User defined rules

**QUERY**
- Query RDF/OWL data and ontologies
- Ontology-Assisted Query of Enterprise Data

**STORE**
- Incr. DML
- Batch-Load
- Bulk-Load

Built-in Security and Versioning for semantic data

- RDF/OWL data
- Ontologies & rule bases

Relational data
Semantic Indexing- Key Components

- **Extensible Information Extractor**
  - Programmable API to plug-in 3rd party extractors into the database.

- **SemContext Indextype**
  - A custom indexing scheme that interacts with the extractor to manage the metadata extracted from the documents efficiently and facilitates semantic search via SQL queries.

- **SEM_CONTAINS Operator**
  - To identify documents of interest based on their extracted metadata, using standard SQL queries.

- **SEM_CONTAINS_SELECT Ancillary Operator**
  - To return additional information (SPARQL Query Results XML) about the documents identified using SEM_CONTAINS operator.
Rowid | Article | Source
---|---|---
1 | Indiana authorities filed felony charges and a court issued an arrest warrant for a financial manager who apparently tried to fake his death by crashing his airplane in a Florida swamp. Marcus Schrenker, 38 | CNN
2 | Major dealers and investors | NW
... | ... | ...

CREATE INDEX ArticleIndex
ON Newsfeed (Article)
INDEXTYPE IS SemContext
PARAMETERS ('my_policy')

SELECT Sem_Contains_Select(1)
FROM Newsfeed
WHERE Sem_Contains (Article, '{?x rdf:type rc:Person .
?x :hasAge ?age .
FILTER(?age >= 35)}',1)=1
AND Source = 'CNN'

SELECT Sem_Contains_Select(1)
FROM Newsfeed
WHERE Sem_Contains (Article, '{?x rdf:type rc:Person .
?x :hasAge ?age .
FILTER(?age >= 35)}',1)=1
AND Source = 'CNN'

Analytical Queries
On Graph Data

Semantic Indexing - Overview

Triples table with rowid references

<table>
<thead>
<tr>
<th>Subject</th>
<th>Property</th>
<th>Object</th>
<th>graph</th>
</tr>
</thead>
<tbody>
<tr>
<td>p:Marcus</td>
<td>rdf:type</td>
<td>rc::Person</td>
<td>&lt;.../r1&gt;</td>
</tr>
<tr>
<td>p:Marcus</td>
<td>:hasName</td>
<td>‘Marcus”^^…</td>
<td>&lt;.../r1&gt;</td>
</tr>
<tr>
<td>p:Marcus</td>
<td>:hasAge</td>
<td>“38”^^xsd::</td>
<td>&lt;.../r1&gt;</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Performance of Oracle’s Semantic Technologies
## Performance

Oracle Performs Best on a Balanced Hardware Setup!

<table>
<thead>
<tr>
<th>Degrees of Parallelism</th>
<th>Data set</th>
<th>Load (B triples/min.)</th>
<th>OWL Inference (M triples/min.)</th>
<th>Query (M answers/min.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>LUBM 8K</td>
<td>1.1B / 28’ 11”</td>
<td>869M / 62’</td>
<td>149+ m in 4.3’</td>
</tr>
<tr>
<td>64</td>
<td>LUBM 8K</td>
<td>1.1B / 53’ 49”</td>
<td>869M / 114’</td>
<td>149+ m in 4.3’</td>
</tr>
</tbody>
</table>

**Setup:**

**Balanced Hardware:** Sun M8000 -- 16 SPARC64 VII+ 3.0 GHz CPUs [64 total Cores 128 Parallel Threads] -- 512 GB Ram -- Dual F5100 Flash Arrays (160 total drives)

**Storage required:** 330GB + 110GB of temporary table space

**Software:** Oracle Database 11.2.0.2.0 + Patch 9825019: SEMANTIC TECHNOLOGIES 11G R2 FIX BUNDLE 3

SGA\_TARGET=256G and PGA\_AGGREGATE\_TARGET=206G

* 1.1B / 28’ 11” means 1.1 billion triples in 28 minutes and 11 seconds
Performance

Oracle Performs Best on a Balanced Hardware Setup!

<table>
<thead>
<tr>
<th>Degrees of Parallelism</th>
<th>Data set</th>
<th>Load (B triples/min.)</th>
<th>OWL Inference (B triples/min.)</th>
<th>Query (B answers/min.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>LUBM 25K</td>
<td>3.4B/105’</td>
<td>2.7B/160’</td>
<td>0.47B/8.7’</td>
</tr>
<tr>
<td>64</td>
<td>LUBM 25K</td>
<td>3.4B/186’</td>
<td>2.7B/210’</td>
<td>0.47B/12.3’</td>
</tr>
</tbody>
</table>

Setup:

**Balanced Hardware:** Sun M8000 -- 16 SPARC64 VII+ 3.0 GHz CPUs [64 total Cores 128 Parallel Threads] -- 512 GB Ram -- Dual F5100 Flash Arrays (160 total drives)

**Storage required:** 900GB + 300GB of temporary table space

**Software:** Oracle Database 11.2.0.2.0 + Patch 9825019: SEMANTIC TECHNOLOGIES 11G R2 FIX BUNDLE 3

SGA_TARGET=256G and PGA_AGGREGATE_TARGET=206G
Semantics in Oracle Database: Summary
Summary: Oracle Database Semantic Technologies

- Scalable to billions of triples
- RAC & Exadata scalability
- Compression & partitioning
- SQL*Loader direct path load
- Parallel load, inference, query
- Oracle DataGuard availability
- Triple-level DoD-strength security
- Choice of SPARQL or SQL
- Native inference engine
- W3C standards compliance
- Semantic Indexing of text & docs
- Growing ecosystem of 3rd party tools partners

Key Capabilities:

Load / Storage
- Native RDF graph data store
- Manages billions of triples
- Optimized storage architecture

Query
- SPARQL-Jena/Joseki, Sesame
- SQL/graph query, b-tree indexing
- Ontology assisted SQL query

Reasoning
- RDFS, OWL2 RL, EL+, SKOS
- User-defined SWRL-like rules
- Incremental, parallel reasoning
- Plug-in architecture
Demo:

OBIEE Integration with RDF Data
(Screenshots)
Investment RDF Data from dava.gov
Transformed SPARQL Query Result

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```xml
<root>
  <item>
    <as>mary21631</as>
    <class_ticker>WPSSX</class_ticker>
    <series_name>STEWART W P & CO GROWTH FUND INC</series_name>
    <class_name>STEWART W P & CO GROWTH FUND INC</class_name>
    <zip_code>10022</zip_code>
    <state>NY</state>
    <city>NEW YORK</city>
    <entity_arg_type>10</entity_arg_type>
    <address_1>577 MADISON AVENUE</address_1>
    <address_2>/
  </item>
  <item>
    <as>mary476</as>
    <class_ticker>FDITX</class_ticker>
    <series_name>Fidelity Contrafund</series_name>
    <class_name>Fidelity Contrafund</class_name>
    <zip_code>2109</zip_code>
    <state>MA</state>
    <city>BOSTON</city>
    <entity_arg_type>10</entity_arg_type>
    <address_1>FIDELITY INVESTMENTS COMPANY</address_1>
    <address_2>22 DEVONSHIRE STREET</address_2>/
  </item>
  <item>
    <as>mary475</as>
    <class_ticker>FNEX</class_ticker>
    <series_name>Fidelity Advisor New Insights Fund</series_name>
    <class_name>Institutional Class</class_name>
    <zip_code>2109</zip_code>
    <state>MA</state>
    <city>BOSTON</city>
    <entity_arg_type>10</entity_arg_type>
    <address_1>FIDELITY INVESTMENTS COMPANY</address_1>
    <address_2>22 DEVONSHIRE STREET</address_2>/
  </item>
</root>
```
Business Model

Relational, XML, … or RDF data sources
Finance RDF Data from DoD

A simple navigation interface. Welcome!

**SPARQL Endpoint**
http://127.0.0.1:8080/oseki/oracle

**SPARQL SELECT Query Body**
```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX fn: <http://www.w3.org/2007/python-function#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT * WHERE {
  ?agency_name <http://localhost/ontologies/budget.owl#hasQuarter> ?quarter .
  ?agency_name <http://localhost/ontologies/budget.owl#hasBudgetAmount> ?budget_amount .
}
```

**Results**

<table>
<thead>
<tr>
<th>Row</th>
<th>AGENCY_NAME</th>
<th>ADDR</th>
<th>BUDGET_AMOUNT</th>
<th>QUARTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>te_Advanced_Research_Projects_Agency</td>
<td>DEF. ADV. RESEARCH PROJ.</td>
<td>46303183</td>
<td>FY201Q4</td>
</tr>
<tr>
<td>2</td>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>UNITED STATES NAVY</td>
<td>1883024</td>
<td>FY201Q4</td>
</tr>
<tr>
<td>3</td>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>UNITED STATES NAVY</td>
<td>132679</td>
<td>FY201Q4</td>
</tr>
<tr>
<td>4</td>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>UNITED STATES NAVY</td>
<td>135180</td>
<td>FY201Q4</td>
</tr>
</tbody>
</table>
Transformed SPARQL Query Results

This XML file does not appear to have any style information associated with it. The document tree is shown below:

```xml
<root>
  <agency name="Defensive_Advanced_Research_Projects_Agency">
    <budget_amount>4601033</budget_amount>
    <quarter>FY2010Q4</quarter>
  </agency>
  <agency name="United_States_Navy">
    <budget_amount>1033024</budget_amount>
    <quarter>FY2010Q4</quarter>
  </agency>
  <agency name="United_States_Navy">
    <budget_amount>32079</budget_amount>
    <quarter>FY2010Q4</quarter>
  </agency>
  <agency name="United_States_Navy">
    <budget_amount>255080</budget_amount>
    <quarter>FY2010Q4</quarter>
  </agency>
  <agency name="United_States_Navy">
    <budget_amount>345.06</budget_amount>
    <quarter>FY2010Q4</quarter>
  </agency>
  <agency name="United_States_Navy">
    <budget_amount>0</budget_amount>
    <quarter>FY2010Q4</quarter>
  </agency>
</root>
```
Demo:

Federated Queries across Oracle Database and Remote SPARQL Web Service endpoint
NYTimes RDF Data about CA

Data stored In Oracle Database
DBPedia RDF Data about CA

- Data stored On the Web
Federated Query

A simple navigation interface. Welcome!

SPARQL Endpoint: [http://127.0.0.1:8080/joseki/nytimes_]
load example query: Fetch subclass & superclass

PREFIX oracle_sem_pr: <http://oracle.com/saneom#>
select ?subject ?subjectInDbpedia ?relationInDbpedia ?resource where {
  ?subject ?subjectInDbpedia ?subjectInDbpedia
  service <http://dbpedia.org/sparql> {
    ?subject <http://www.w3.org/2000/01/rdf-schema#Label> "California"@en
  } filter (!sameTerm(?subjectInDbpedia, <http://dbpedia.org/resource/Cuba>))
}

Submit Query
Show Advanced Options

<table>
<thead>
<tr>
<th>Row</th>
<th>NYTIMES_SUBJECT</th>
<th>P</th>
<th>SUBJECT</th>
<th>SUBJECT_IN DbPEDIA</th>
<th>RELATION_IN DbPEDIA</th>
<th>RESOURCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>56975600069322233571rdf</td>
<td>attributen</td>
<td>56975600069322233571</td>
<td>dbpedia_res_California</td>
<td>dType</td>
<td>owl:Thing</td>
</tr>
<tr>
<td>2</td>
<td>56975600069322233571rdf</td>
<td>attributen</td>
<td>56975600069322233571</td>
<td><a href="http://dbpedia.org/resource/California">http://dbpedia.org/resource/California</a></td>
<td>dbpedia_ont/AdministrativeRegion</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>56975600069322233571rdf</td>
<td>attributen</td>
<td>56975600069322233571</td>
<td>StatesOfTheUnitedStates</td>
<td>dbpedia_ont/Place</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>56975600069322233571rdf</td>
<td>attributen</td>
<td>56975600069322233571</td>
<td>dbpedia_res_California</td>
<td>dType</td>
<td>dbpedia_ont/PopulationPlace</td>
</tr>
<tr>
<td>5</td>
<td>56975600069322233571rdf</td>
<td>attributen</td>
<td>56975600069322233571</td>
<td>dbpedia_res_California</td>
<td>dType</td>
<td>dbpedia_ont/Place</td>
</tr>
</tbody>
</table>

Data Merged together!