Semantic Web

Using Oracle Semantic Graph in a scientific knowledge portal for the pharmaceutical industry

Author : Marc Lieber

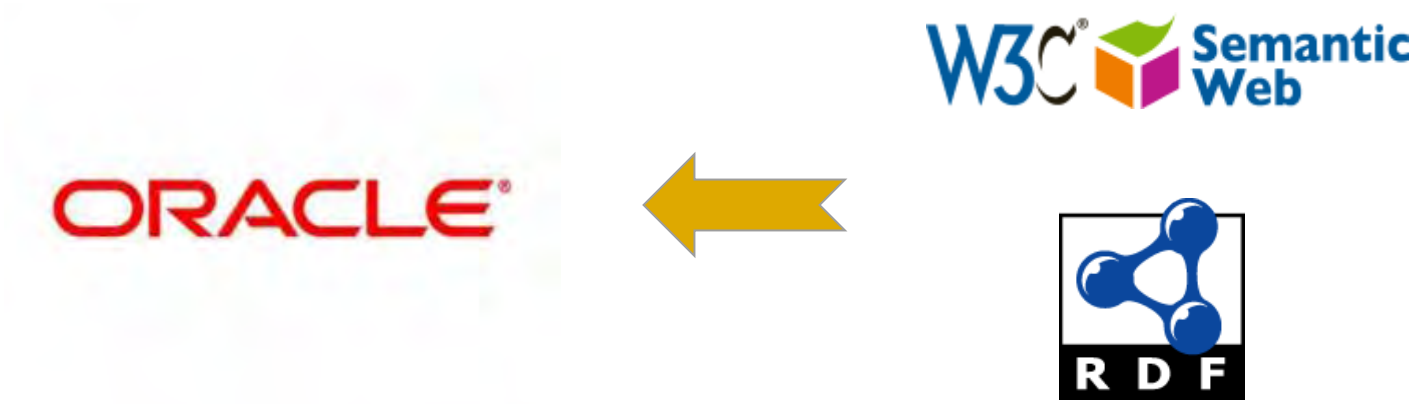
Date : 05/02/2013

AGENDA

1. Oracle RDF Triple Store
2. Pharma Ontology search tool at Novartis
3. Questions and answers

Introduction

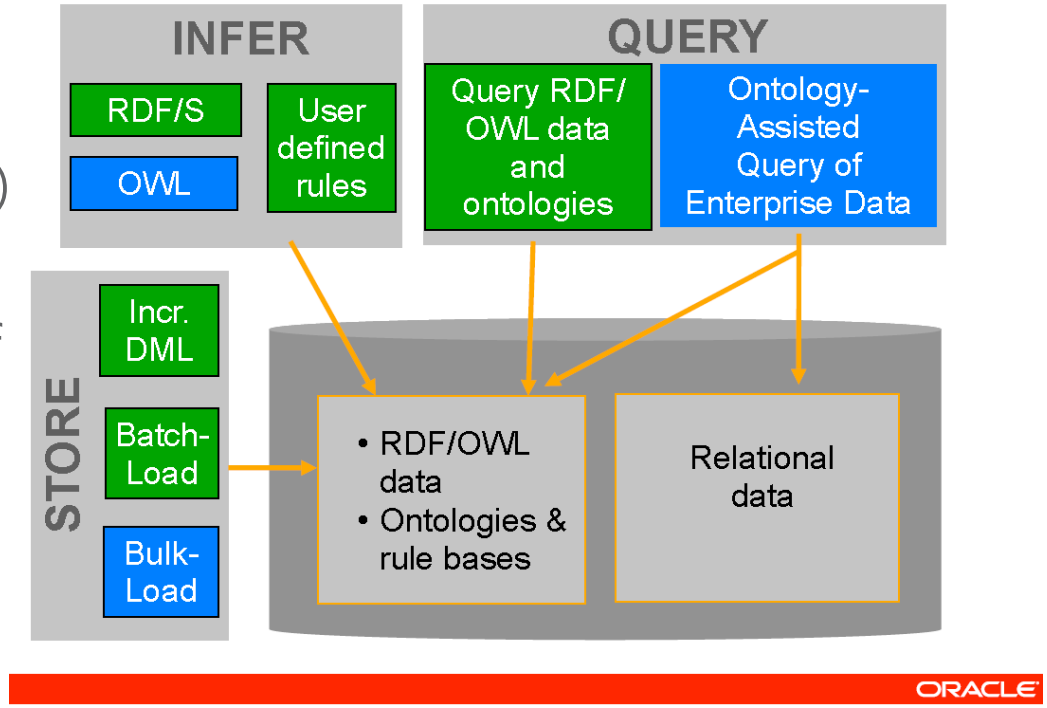
1. Oracle Semantic Graph is a way to store and maintain Ontology oriented data in the Oracle relational database



2. Our case study is a semantic data integration platform for the biomedical domain using Oracle Semantic Graph

Oracle RDF Triple Store

- Oracle Semantic Graph is an add-on to Oracle Spatial.
  - Spatial allows geoTemporal search and inferencing on semantic data
- Supports most of the W3C rules
- Use of named graphs (quad) since 11.2.0.3
- Scales up to 100's billions of triples
- Oracle specific adapters available for JENA, SESAME, TopBraid, Cytoscape and Protege



ORACLE Database RDF Query engine

SEM_MATCH: Adding SPARQL to SQL

SPARQL

```
PREFIX foaf: <http://...>
SELECT ?n1 ?n2
FROM <http://g1>
WHERE
 {?p foaf:name ?n1
  OPTIONAL {?p foaf:knows ?f .
            ?f foaf:name ?n2 }
  FILTER (REGEX(?n1, "^A"))   }
```

SQL

```
SELECT n1, n2
FROM TABLE(SEM_MATCH(
 '{?p foaf:name ?n1
   OPTIONAL {?p foaf:knows ?f .
             ?f foaf:name ?n2 }
   FILTER (REGEX(?n1, "^A"))  }',
SEM_MODELS('g1'),…,
SEM_ALIASES(
   SEM_ALIAS('foaf','http://…')), …))
```

projection

data selection

graph pattern

prefixes

SPARQL Query

- SQL query on a relational table

```
SQL> SELECT ename from EMP where JOB='CLERK';
```

- SPARQL in the *SEM_MATCH* function

```
{?s ?p ?o FILTER(sameTerm(?p,:ENAME)) .
 ?s :hasJob :Clerk}
```

```
SELECT s, p,s$_SUFFIX, p$_SUFFIX, o
  FROM TABLE(SEM_MATCH('{?s ?p ?o  FILTER(sameTerm(?p,:ENAME)) .
                        ?s :hasJOB :Clerk}',
    SEM_Models('emp1'),
    SEM_Rulebases('OWLPRIME'),
SEM_ALIASES(SEM_ALIAS('','http://www.example.org/emp#')),
    null,
    null,
    null));
```

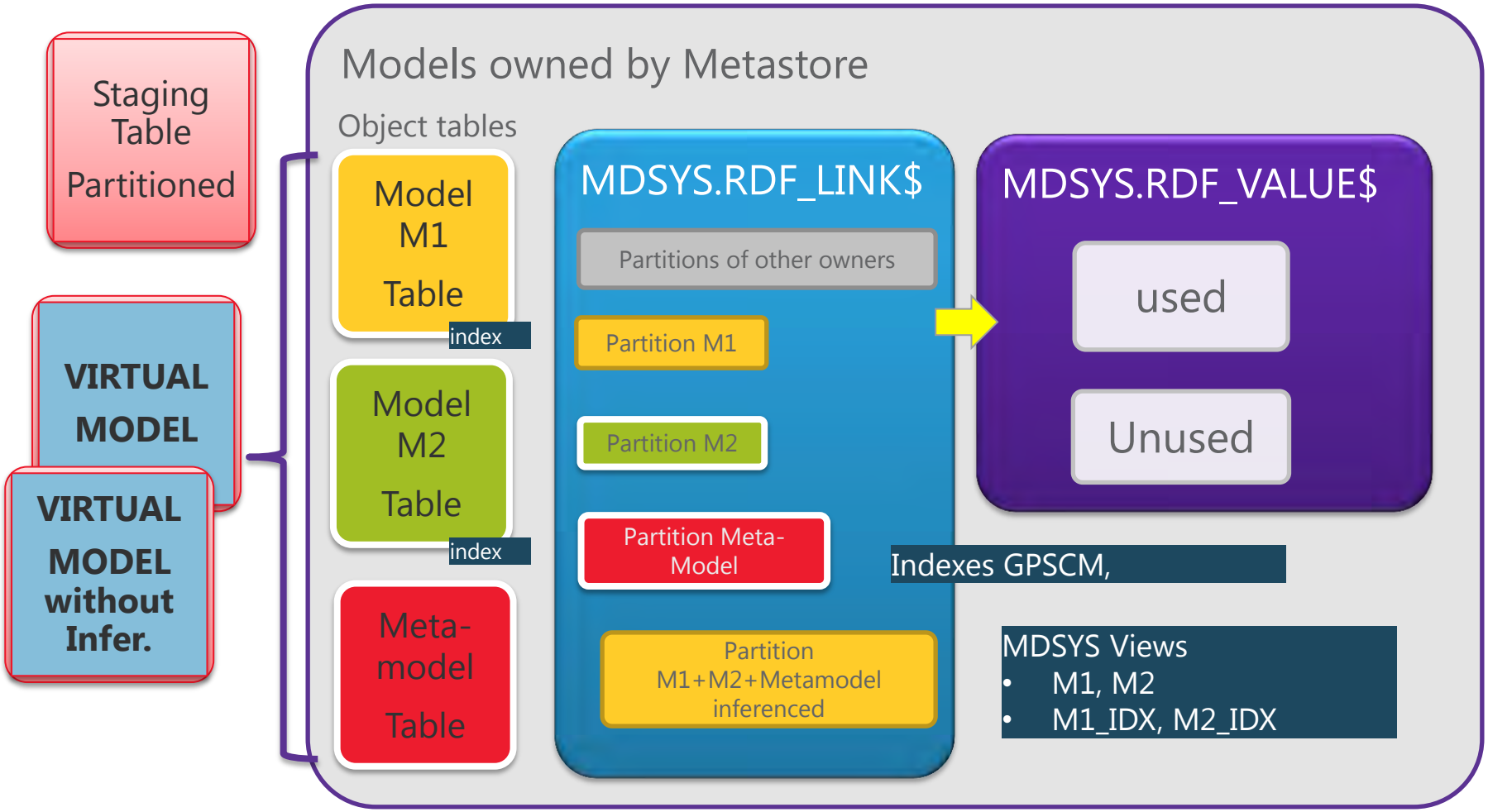
ery... ×

SQL All Rows Fetched: 4 in 0.35 seconds

S	P	S$_SUFFIX	P$_SUFFIX	O
http://www.example.org/emp#employee7369	http://www.example.org/emp#ENAME	employee7369	ENAME	SMITH
http://www.example.org/emp#employee7876	http://www.example.org/emp#ENAME	employee7876	ENAME	ADAMS
http://www.example.org/emp#employee7934	http://www.example.org/emp#ENAME	employee7934	ENAME	MILLER
http://www.example.org/emp#employee7900	http://www.example.org/emp#ENAME	employee7900	ENAME	JAMES

Physical implementation of Oracle Semantics

1. A good understanding of the physical implementation is necessary



Live demo Oracle Semantic Graph

1. SPARQL Queries using Joseki
2. SPARQL Queries using the SEM_MATCH function
3. Virtua Model implementation
4. SQL join Triple Store with RDF Tables
5. Inferencing

AGENDA

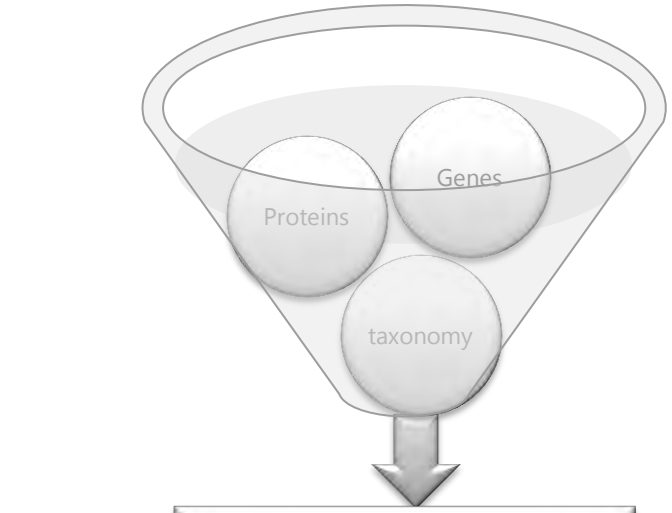
1. Oracle RDF Triple Store
2. **Pharma Ontology search tool at Novartis**
3. Questions and answers

Project Overview
Metastore Fundamentals

1. Consists of a semantic data federation layer based on controlled terminologies extracted from scientific data repositories
2. Organized around scientific concepts: Genes, Proteins, Indications, Anatomy, diseases, taxonomy etc…;
   - some hierarchically organized and classified
3. Complemented by referential knowledge (cross references to internal and external knowledge repositories)
4. Ontological relations between concepts materializing semantic network of scientific concepts
5. Content is monthly updated (concept type centric updates) during dedicated loading exercises

Project Overview

Workflow Loading Exercise (1)



Import with Jena into the staging Table

Pre-validate

Bulk Load

Post-Validate

Materialize in tables

Swap model Live and Work

Loading Exercise

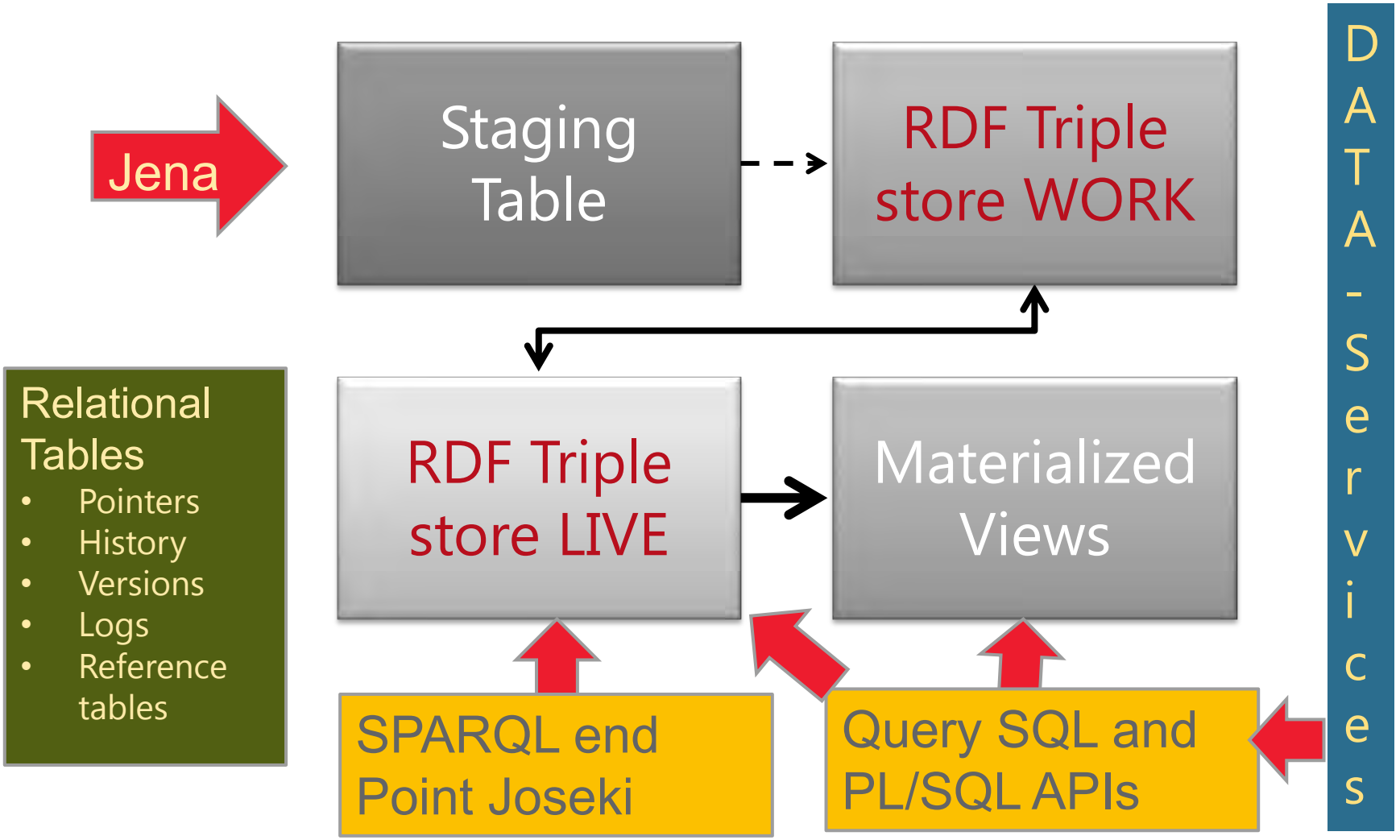
Contains one to many RDF/XML files to be uploaded

There is one file per concept type

- Each file is checked first against an XML Schema

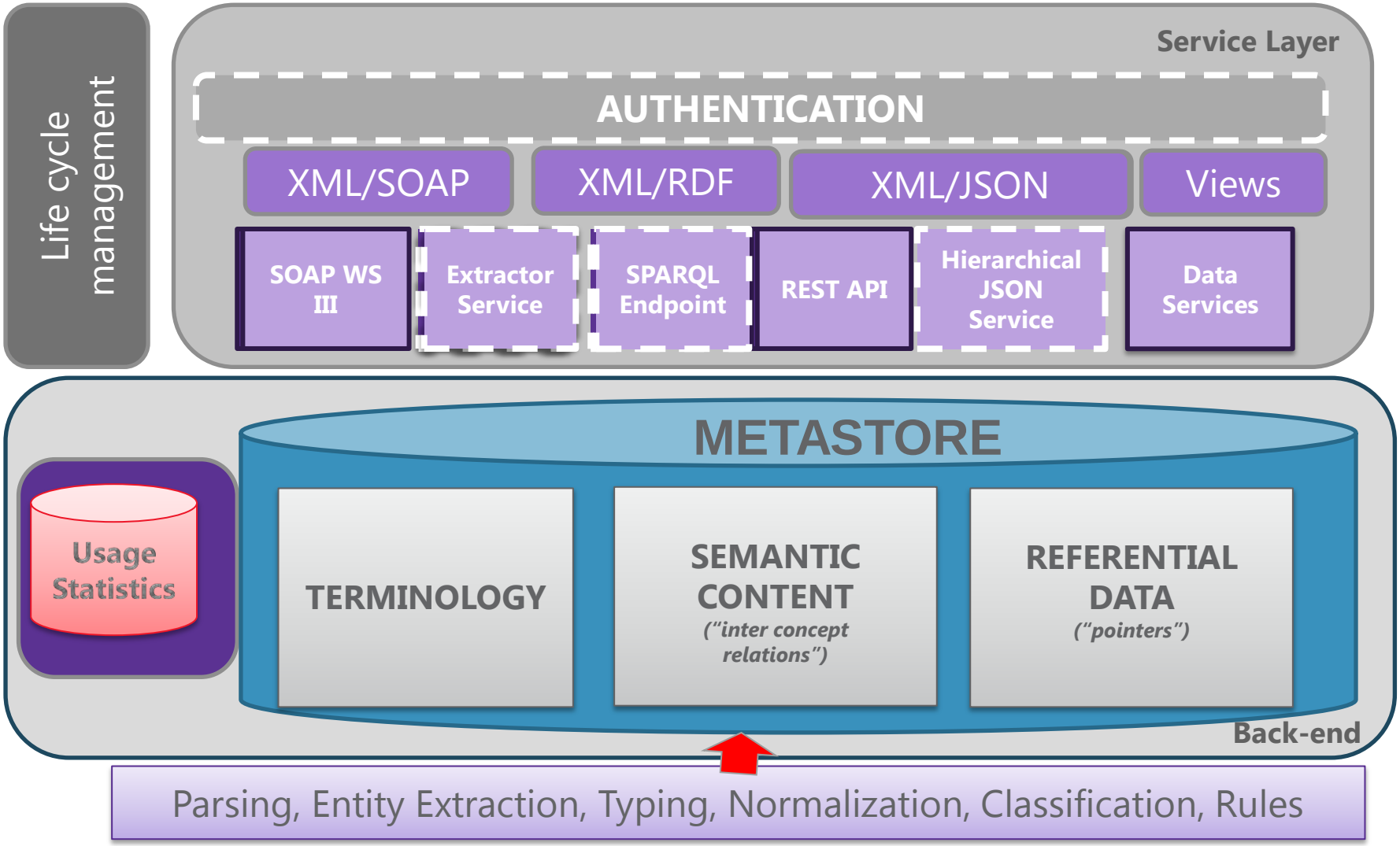
Project Overview

Workflow Loading Exercise (2)



Project Overview

Building blocks



Technical Implementation

Staging table and Model implementation

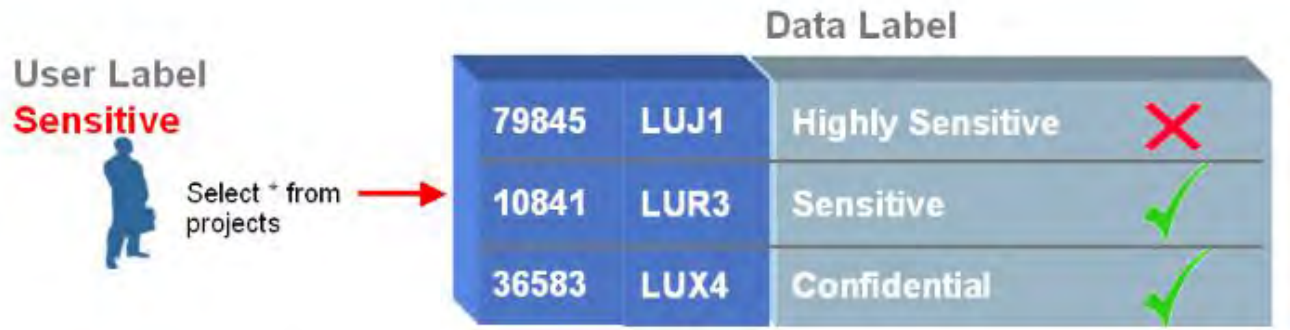
1. A Partitioned staging table stores the uploaded triples
   - The RDF/XML files read by Jena can be up to 1.2 Gigabytes in size
     - This is not a problem for the load into the staging table
   - The validation process checks for inconsistencies (dangling references, missing mandatory properties, …) before bulk loading into the semantic Model
2. The Metastore RDF Model has been duplicated into *MS3_LIVE and MS3_WORK*
   - Separates the productive Data from the work-in-progress Data
   - Note: Versioning using Oracle Workspace Management did not work

Special requirements : Versioning

1. Concepts are versioned
   - Each concept has an history and its content can be compared between versions
   - Only modified or new concepts should get a new version ID
     - Verified during the validation process on the staging table
     - the old triples related to this modifed concept are deleted and replaced by the latest version of the concept

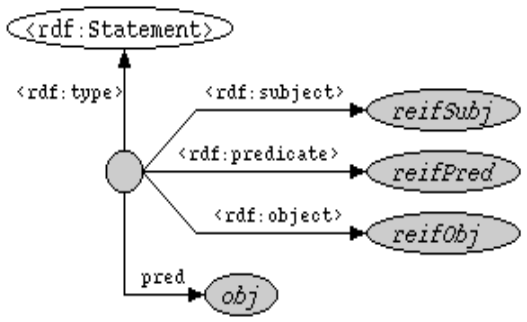
Special requirements : Security

1. We need a security concept on triple level
   - The default control of access to the Oracle Database semantic data store is at the model level.
   - Oracle does not recommend to use Virtual Private Database in the triple store
   - We use instead the new 11gR2 feature Oracle Label Security for Triples



Special requirements : Reification

1. Support for annotation on triple level : reification




```
<rdf:Statement rdf:ID="NVMTARSPHSP0687001-isMemberOf-NVMTGCAV38">
    <rdf:object rdf:resource="#NVMTGCAV38"/>
    <rdf:predicate rdf:resource="#isMemberOf"/>
    <rdf:subject rdf:resource="#NVMTARSPHSP0687001"/>
    <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">some text</rdfs:label>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >some text</rdfs:comment>
  </rdf:Statement>

<http://www.example.com#AAA> <rdf#type> <rdf#Statement> .
<http://www.example.com#AAA> <rdf#object> <http://www.example.com#NVMTGCAV38> .
<http://www.example.com#AAA> <rdf#predicate> <http://www.example.com#isMemberOf> .
<http://www.example.com#AAA> <rdf#subject> <http://www.example.com#NVMTARSPHSP0687001> .
<http://www.example.com#AAA> <rdfs#label> "some text"^^<xsd#string> .
<http://www.example.com#AAA> <rdfs#comment> "some text"^^<xsd#string> .
```

- Problem : complex, slows down query performances
- we decided to make it only visible in SQL
  - PL/SQL to transform reified triples into standart triples, the annotation are stored as column in the Semantic Object table

Technical Implementation

Data volume

1. Each model contains 1,001,544 concepts / 22 concept types
2. Stored 74,000,000 triples + 35,000,000 inferred triples for each Model
   - *RDF_LINK$* table size 183,000,000 rows; 19,237 Mb
   - *RDF_VALUE$* table size 137,000,000 rows; 23,314Mb
     - Only 34,000,000 rows are actually used in our Models → will be fixed
   - We use Keep Pool to cache the partition Model LIVE on *RDF_LINK$* + Keep Pool on *RDF_VALUE$*
     - Required an alter table storage (Buffer pool keep)
3. Expected growth : 100% more by the end of the year

Technical Implementation

Use of Named Graphs

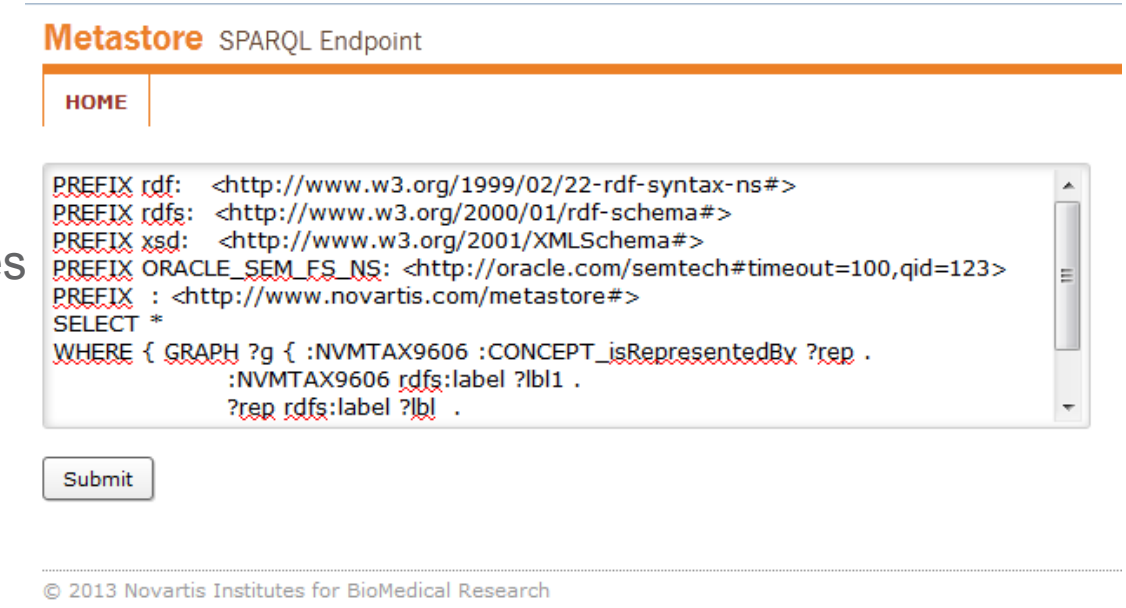
1. For better performance we switched to named graph
   - one for the semantic model, one for each concept
   - Every triple is now associated to a named graph
   - Inferred triples can also be associated with named graphs
2. Issues with blank nodes getting larger and larger …..
3. *SEM_MATCH* Query using name graphs :

```
SELECT * FROM TABLE(SEM_MATCH(
'select ?rep ?obj
  { GRAPH :gNVMTAX9606 {
 :NVMTAX9606 :CONCEPT_isRepresentedBy  ?rep .
 OPTIONAL{?rep :REPRESENTATION_hasSource ?obj . }}}',
 sem_models('MS3_LIVE'),SEM_RULEBASES('') , SEM_ALIASES(
SEM_ALIAS(null,'http://www.novartis.com/metastore#')),null,null,
' GRAPH_MATCH_UNNAMED=F PLUS_RDFT=T
',null,SEM_GRAPHS(':gNVMTAX9606')));
```

Client SPARQL End Point

Use of Named Graphs; Joseki SPARQL queries

1. We decided to rewrite the SPARQL end point to have a better control on what end users can do and to support
   - Session Kills
   - Timeouts
   - Oracle Hints
   - Named Graphs
   - List of predefined queries



Technical Implementation

PL/SQL implementation

1. The *REST* Webservice calls PL/SQL functions to retrieve the triples in a nt triple format
   - Example using our function *nt_describe*

```
SELECT ms3_util.nt_describe('NVMTAX9606',0,0,1)
FROM DUAL;
```

   - Returns triples in a CLOB

_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2acb <http://www.novartis.com/metastore#SOURCE_LogicalType> "CHAR_PREFIX" .
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2abc <http://www.novartis.com/metastore#SOURCE_LogicalType> "AUTHORITY" .
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2ac2 <http://www.novartis.com/metastore#SOURCE_LogicalType> "OTHER_SYMBOL" .
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2abe <http://www.novartis.com/metastore#SOURCE_LogicalType> "SCIENTIFIC_NAME" .
<http://www.novartis.com/metastore#NVMTAX9606> <http://www.w3.org/2000/01/rdf-schema#subClassOf> <http://www.novartis.com/metastore#NVMTAXN7> .
<http://www.novartis.com/metastore#NVMTAX9606> <http://www.w3.org/2000/01/rdf-schema#subClassOf> <http://www.novartis.com/metastore#NVMTAX314295> .
<http://www.novartis.com/metastore#NVMTAX9606> <http://www.w3.org/2000/01/rdf-schema#subClassOf> <http://www.novartis.com/metastore#TAXONOMY> .
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2abf <http://www.novartis.com/metastore#REPRESENTATION_hasSource> _:m59g3C6874747(
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2ab5 <http://www.novartis.com/metastore#REPRESENTATION_hasSource> _:m59g3C6874747
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2abd <http://www.novartis.com/metastore#REPRESENTATION_hasSource> _:m59g3C6874747
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2acc <http://www.novartis.com/metastore#REPRESENTATION_hasSource> _:m59g3C6874747(
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2ab3 <http://www.novartis.com/metastore#REPRESENTATION_hasSource> _:m59g3C6874747
_:m59g3C687474703A2F2F7777772E6E6F7661727469732E636F6D2F6D65746173746F726523674E564D544158393630363EgmORABNENC3P0H11H25HC45E65e36d7258E13b41d8d6c258E2ac5 <http://www.novartis.com/metastore#REPRESENTATION_hasSource> _:m59g3C6874747(

   - This way, we can optimize the *SEM_MATCH* queries but we still have sometimes performance issues (more than 5 sec. waiting time)

Next Step : Virtual Model

- Splitting the current triple store into multiple triple stores
  - One Model per concept type
  - A copy of each model for the working environment
    - If 22 concept types, then Metastore owns 44 models + 2 models for the metamodel
- Remove the “swap model” process and replace it by a drop and recreate Virtual Model
- Replace all blank nodes by a URI to solve the problem of Blank Nodes getting larger and larger
  - This will reduce also the size of *RDF_VALUE$*
- Performance should stay the same, maybe better because of the partitioning of *RDF_LINK$*

Core messages

- Oracle 11gR2 implementation of RDF Web semantics is a powerful new way of storing data in a database
- The advantage of using the Oracle Triple store are multiple
  - SPARQL and SQL interaction with relationally stored data
  - Use of SQL Hints, indexes and caching to increase performances
  - Standard DB Administration : Backup/recovery/replication, etc…
  - PL/SQL or Java programming
  - Supports large volumes of data
  - Good integration with standard RDF client tools such as Jena and Sesame
- Newcomers to RDF Web Semantics will need some time to get used to the various modeling concepts and to the SPARQL syntax
- Newcomers to Oracle Semantic Graph will need some time to fine tune the Oracle specific features but the effort is worthwhile!

AGENDA

1. Oracle RDF Triple Store
2. Pharma Ontology search tool at Novartis
3. **Questions and answers**

THANK YOU.

Marc Lieber

Marc.lieber@trivadis.com
www.trivadis.com

BASEL BERN LAUSANNE ZÜRICH DÜSSELDORF FRANKFURT A.M. FREIBURG I.BR. HAMBURG MÜNCHEN STUTTGART WIEN