Building Database Infrastructure for Managing Semantic Data

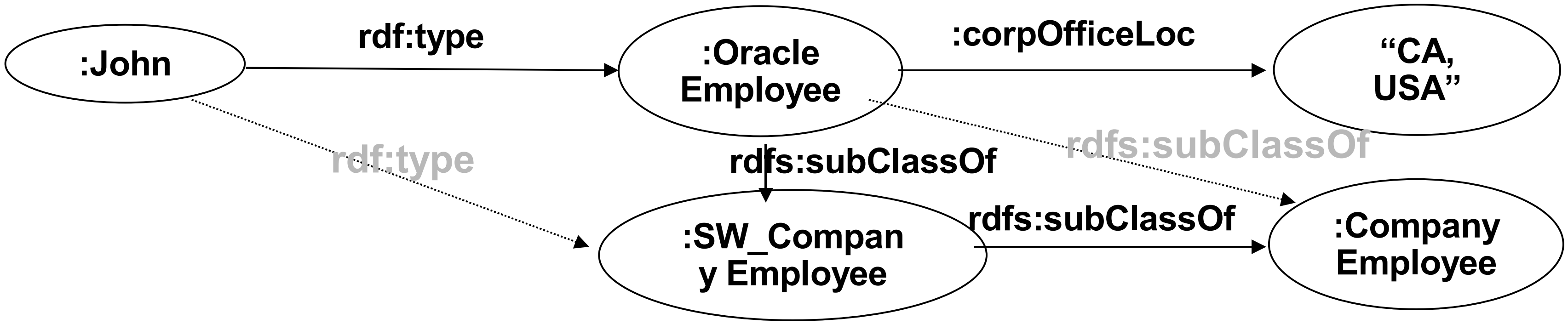
Agenda

- Semantics support in the database
- Our model
  - Storage
  - Query
  - Inference
- Use cases: Enhancing database queries with semantics

ORACLE

Semantic Technology

- Facts are represented as <u>triples</u>
- Triple is the basic building block in the semantic representation of data
- Triples together form a graph, connecting pieces of data
- New triples can be inferred from existing triples
- RDF and OWL are W3C standards for representing such data



Using a Database for Semantic Applications

- Database queries can be enhanced using semantics
  - Syntactic comparisons can be enhanced with semantic comparisons
- All database characteristics become available for semantic applications
  - Scalability: Database type scale backed by decades of work difficult to match by specialized stores
  - Security, transaction control, availability, backup and recovery, lifecycle management, etc.

ORACLE

Using a Database for Semantic Applications (contd.)

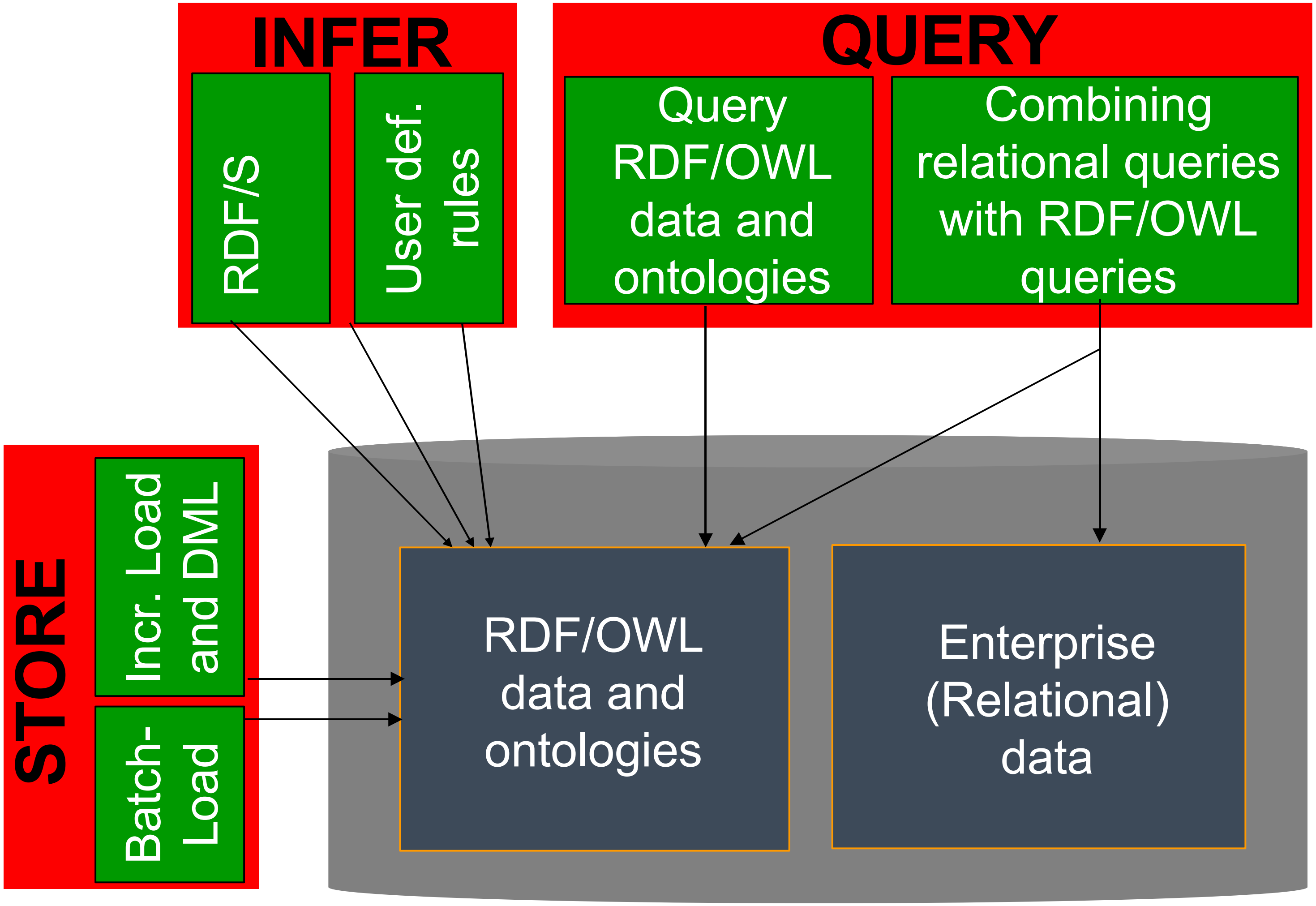
- SQL (an open standard) interface is familiar to a large community of developers
  - Also SQL constructs can be used for operating on semantic data
  - Existing database users interested in exploring semantics to enhance their applications
- Databases are part of infrastructure in several categories of applications that use semantics
  - Biosurveillance, Social Networks, Telcos, Utilities, Text, Life Sciences, GeoSpatial

Our Approach

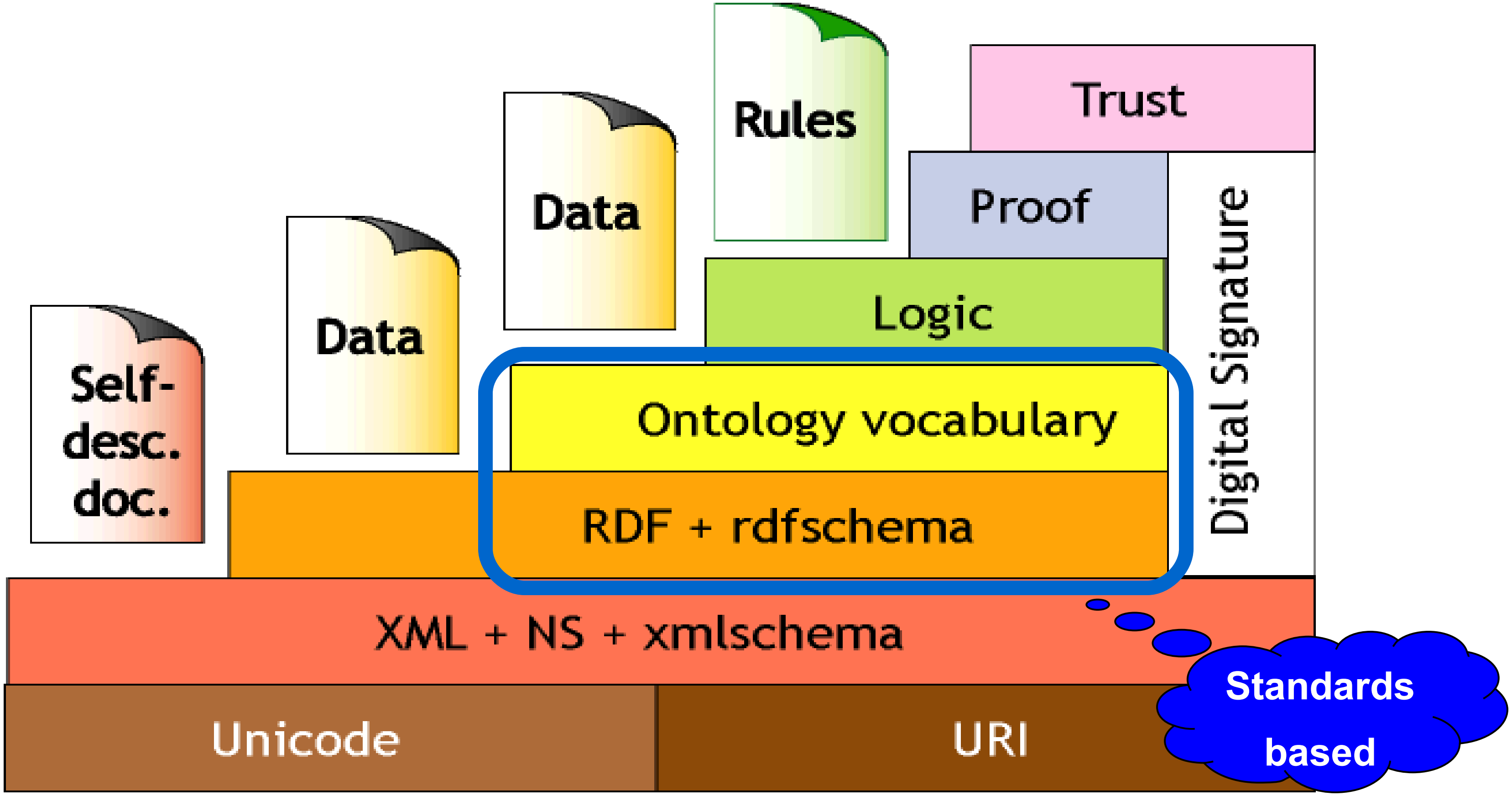
- Provide support for managing RDF data in the database for semantic applications
  - Storage Model
  - SQL-based RDF query interface
  - Query interface that enables combining with SQL queries on relational data
  - Inferencing in the database (based on RDFS and user-defined rules)
  - Support for large graphs (billion+ triples)

ORACLE

Technical Overview

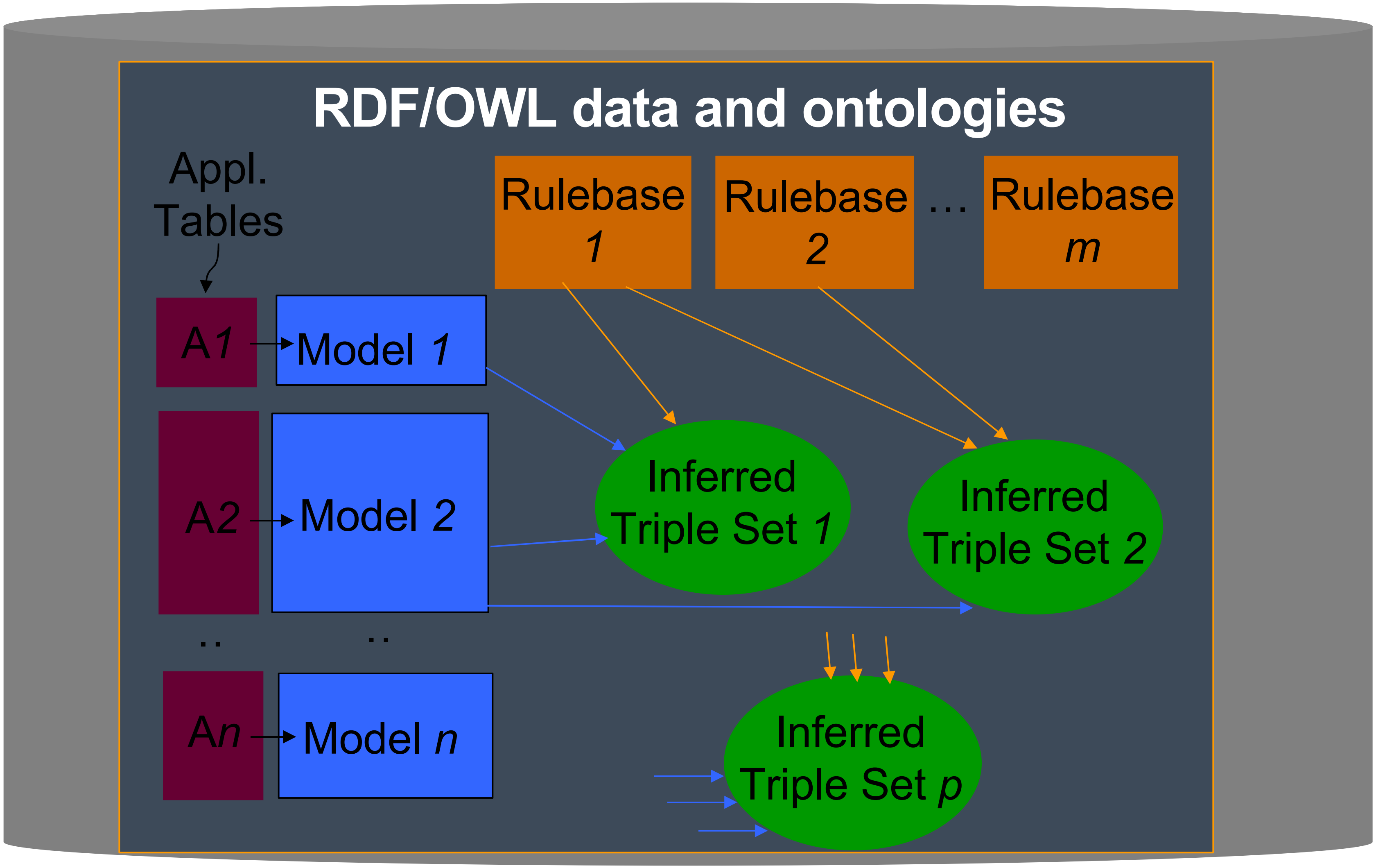


Semantic Technology Stack

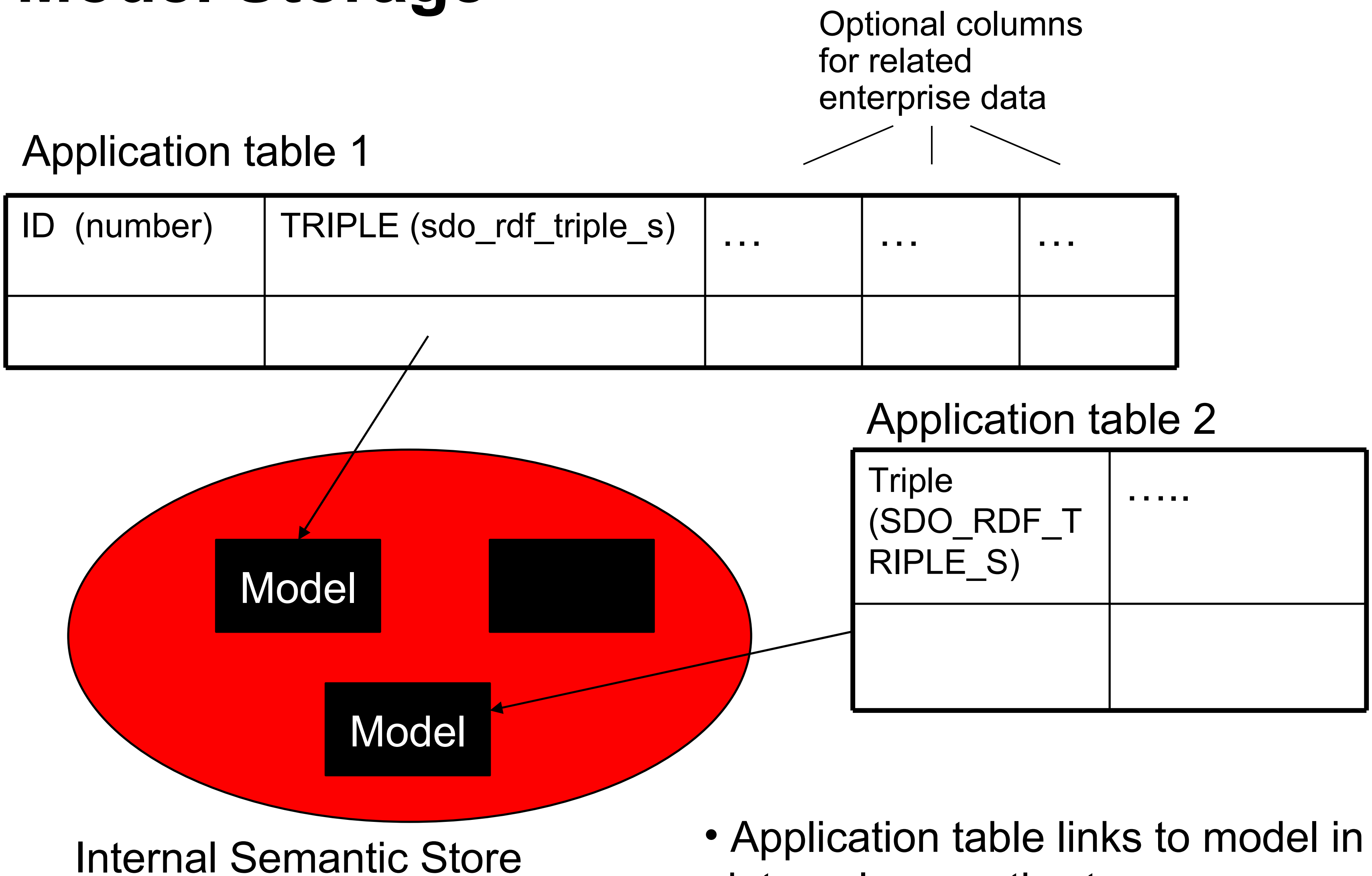


Semantic Technology
Storage

Storage: Schema Objects



Model Storage



- Application table links to model in internal semantic store

Internal Semantic Store

IdTriples

Model	S_*id*	P_*id*	O_*id*	…
Partition containing Data for Model *1*				
…				
Partition containing Data for Model *n*				
Partition containing Data for Inferred Triple Set *1*				
…				
Partition containing Data for Inferred Triple Set *p*				

UriMap

Value	*Id*	Type	…
Mapping: Value ←1:1→ Id			

Rulebase

Rb	rule	ante	filter	cons
Ante. [+ Filter] => Cons.				

Storage: Highlights

- Generates hash-based IDs for values (handles collisions)
- Does canonicalization to handle multiple lexical forms of same value point
  - Ex: “0010”^^xsd:decimal and “010”^^xsd:decimal
- Maintains fidelity (user-specified lexical form)
- Allows long literal values (using CLOBs)
- Handles duplicate triples
- No limits on amount of data that can be stored

Semantic Technology

Query

RDF Querying Problem

- Given
  - RDF graphs: the data set to be searched
  - Graph Pattern: containing a set of variables
- Find
  - Matching Subgraphs
- Return
  - Sets of variable bindings: where each set corresponds to a Matching Subgraph

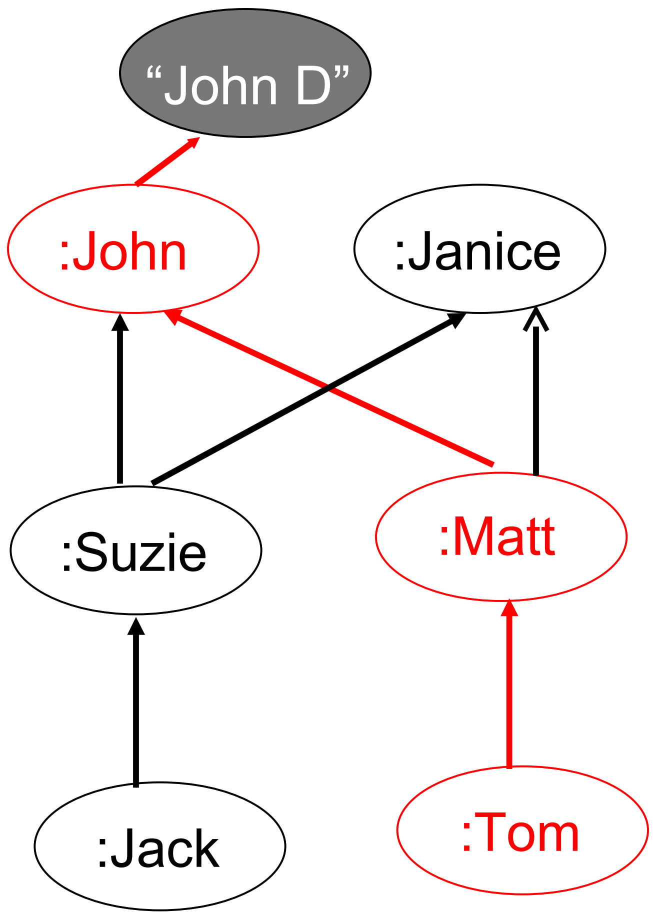
Query Example: Family Data

Data: :Tom :hasParent :Matt
:Matt :hasFather :John
:Matt :hasMother :Janice
:Jack :hasParent :Suzie
:Suzie :hasFather :John
:Suzie :hasMother :Janice
:John :hasName “JohnD”

Graph pattern ‘(:Tom :hasParent ?x)
(?x :hasFather ?y)
(?y :name ?name)',

Variable bindings: x = :Matt
y = :John
name = “John D”

Matching subgraph:
‘(:Tom :hasParent :Matt)
(:Matt :hasFather :John)
(:John :name “John D”)',



RDF Query Approaches

- General Approach
  - Create a new (declarative, SQL-like) query language
  - e.g.: RQL, SeRQL, TRIPLE, N3, Versa, SPARQL, RDQL, RDFQL, SquishQL, RSQL, etc.
- Our SQL-based Approach
  - **Embedding** a graph query in a SQL query
  - SPARQL-like graph pattern embedded in SQL query
- Benefits of SQL-based Approach
  - Leverages all the powerful constructs in SQL (e.g., SELECT / FROM / WHERE, ORDER BY, GROUP BY, aggregates, Join) to process graph query results
  - RDF queries can easily be combined with conventional queries on database tables thereby avoiding staging

SDO_RDF_MATCH Table Function

- Input Parameters

  SDO_RDF_MATCH (

Query,	← SPARQL-like graph-pattern (with vars)
Models,	← set of RDF/OWL models
Rulebases,	← set of rulebases (e.g., RDFS)
Aliases,	← aliases for namespaces
Filter	← additional selection criteria

)
- Return type in definition is AnyDataSet
- Actual return type is determined at compile time based on the graph-pattern argument

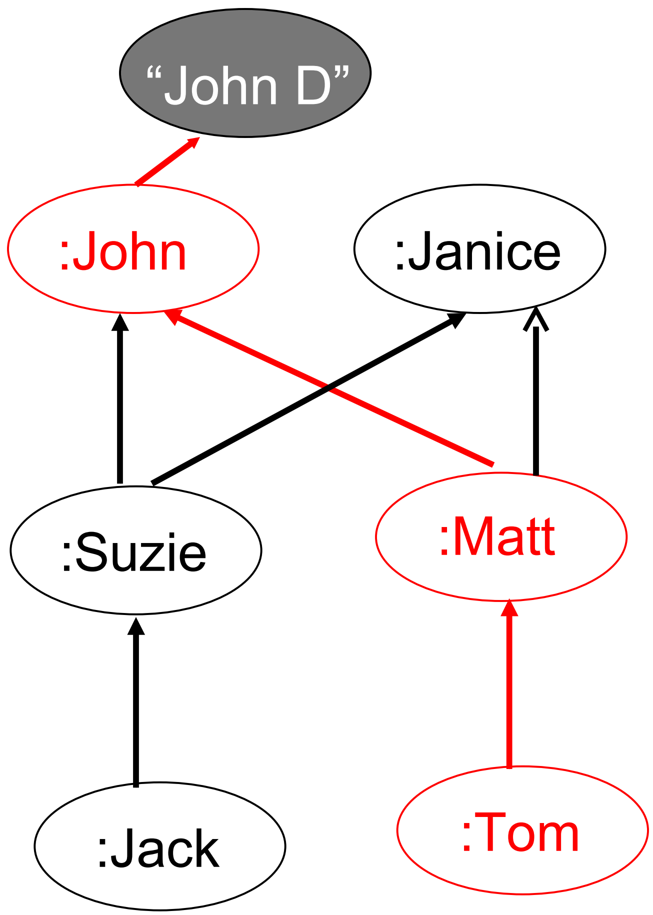
ORACLE

Query Example: SQL-based interface

```
select x, y, name from
    TABLE(SDO_RDF_MATCH(
    '(:Tom :hasParent ?x)
     (?x     :hasFather ?y)
     (?y     :name          ?name)',
      SDO_RDF_Models('family'),
      .., .., ..));
```

Returns the name of Tom's grandfather

X	Y	NAME
Matt	John	"John D"



ORACLE

Combining RDF Queries with Relational Queries

- Find salary and hiredate of Tom’s grandfather(s)
- SELECT emp.name, emp.salary, emp.hiredate
  FROM emp, TABLE(SDO_RDF_MATCH(
  ‘(:Tom :hasParent ?y)
  (?y :hasFather ?x)
  (?x :name ?name)’,
  SDO_RDF_Models(‘family'),
  …)) t
  WHERE emp.name=t.name;

ORACLE

RDF_MATCH Query Processing

- Subsititute aliases with namespaces in search pattern
- Convert URIs and literals to internal IDs
- Generate Query
  - Generate self-join query based on matching variables
  - Generate SQL subqueries for rulebases component (if any)
  - Generate the join result by joining internal IDs with UriMap table
  - Use model IDs to restrict IdTriples table
- Compile and Execute the generated query

Table Columns returned by SDO_RDF_MATCH

Each returned row contains one (or more) of the following cols (of type VARCHAR2) for each variable ?x in graph-pattern:

Column Name	Description
x	Value matched with ?x
x$rdfVTYP	Value TYPe: URI, Literal, or Blank Node
x$rdfLTYP	Literal TYPe: e.g., xsd:integer
x$rdfCLOB	CLOB value matched with ?x
x$rdfLANG	LANGuage tag: e.g., “en-us”

Projection Optimization: Only the columns referred to by the containing query are returned.

Optimization: Table Function Rewrite

- TableRewriteSQL( )
  - Takes RDF Query (specified via arguments) as input
  - generates a SQL string
- Substitute the table function call with the generated SQL string
- Reparse and execute the resulting query
- Advantages
  - Avoid execution-time overhead (linear in number of result rows) associated with table function infrastructure
  - Leverage SQL optimizer capabilities to optimize the resulting query (including filter condition pushdown)

Semantic Technology
Inference

Inference: Overview

- Native inferencing in the database for
  - RDF, RDFS
  - User-defined rules
- Rules are stored in rulebases in the database
- RDF graph is entailed (new triples are inferred) by applying rules in rulebase/s to model/s
- Inferencing is based on forward chaining: new triples are inferred and stored ahead of query time
  - Minimizes on-the-fly computation and results in fast query times

Inferencing

- RDFS Example:

  A rdf:type B, B rdfs:subClassOf C

  => A rdf:type C

  Ex: Matt rdf:type Father, Father rdfs:subClassOf Parent

  => Matt rdf:type Parent

- User-defined Rules Example:

  A :hasParent B, B :hasParent C

  => A :hasGrandParent C

  Ex: Tom :hasParent Matt, Matt :hasParent John

  => Tom :hasGrandParent John

Creating a rulebase and rules index (SQL based)

- Creating a rule base
  - create_rulebase(‘family_rb’);
  - insert into mdsys.RDFR_family_rb values(
    ‘grandParent_rule',
    ‘(?x :hasParent ?y) (?y :hasParent ?z)’,
    NULL,
    '(?x :hasGrandParent ?z)',
    …..);
- Creating a rules index
  - create_rules_index(‘family_idx’,sdo_rdf_models(‘family’),sdo_rdf_rulebases(‘rdfs’,’family_rb)

Query Example: Family Data

select y, name from TABLE(SDO_RDF_MATCH(

‘(:Tom :hasGrandParent ?y)

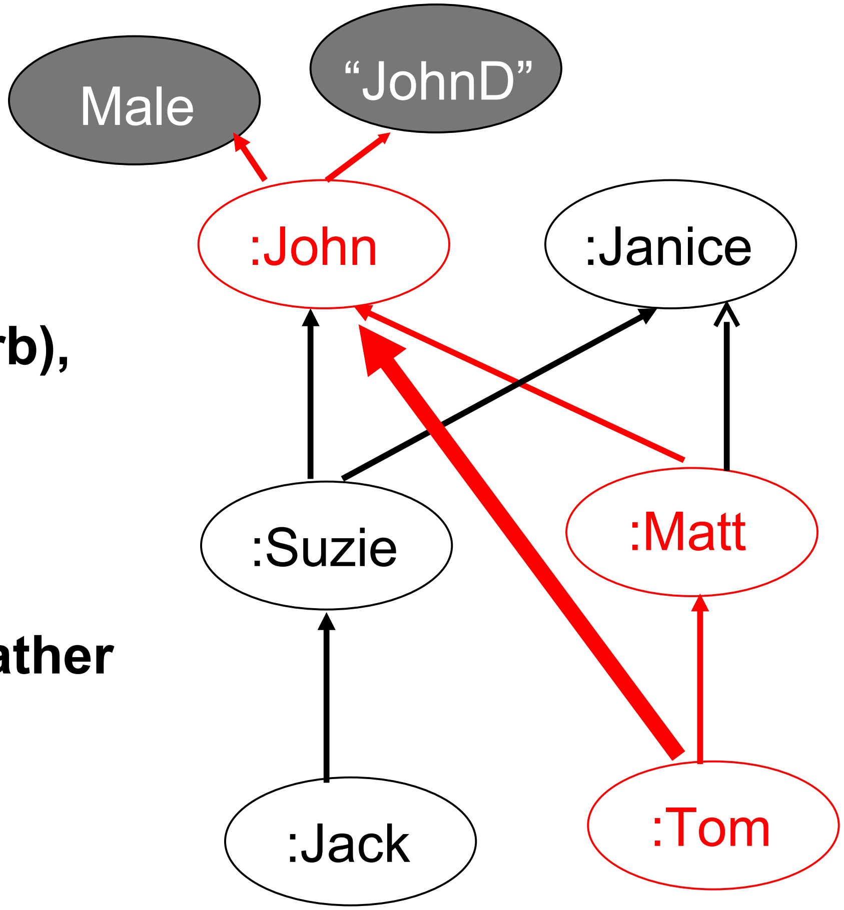
(?y :name ?name)’

(?y rdf:type :Male),

SDO_RDF_Models('family'),

SDO_RDF_Rulebases(‘family_rb),

.., ..));



Returns the name of Tom’s grandfather

Y	NAME
John	‘John D’

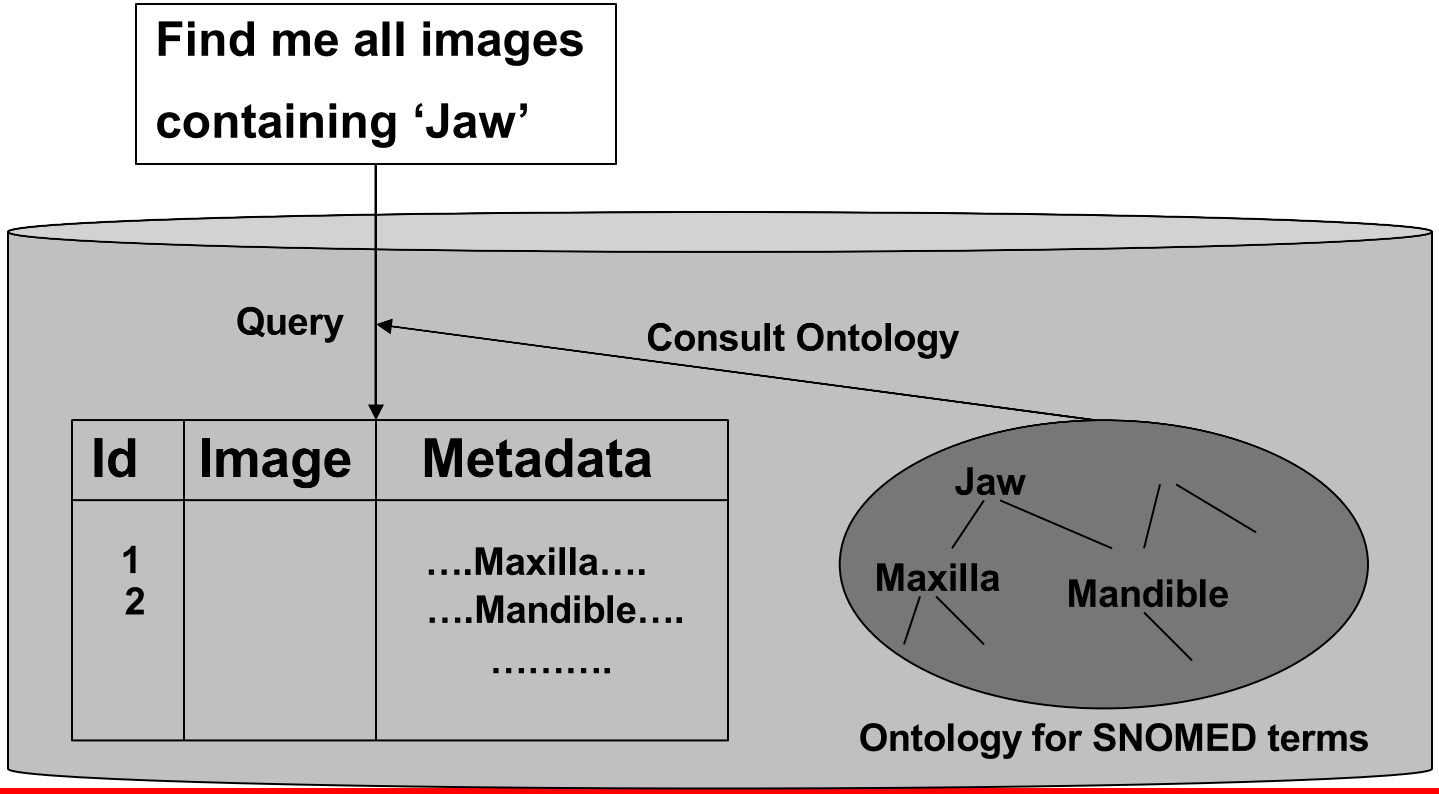
ORACLE

Semantic Technology
Enhancing Database Queries with Semantics

Semantics Enhanced Search

Medical Information Repositories

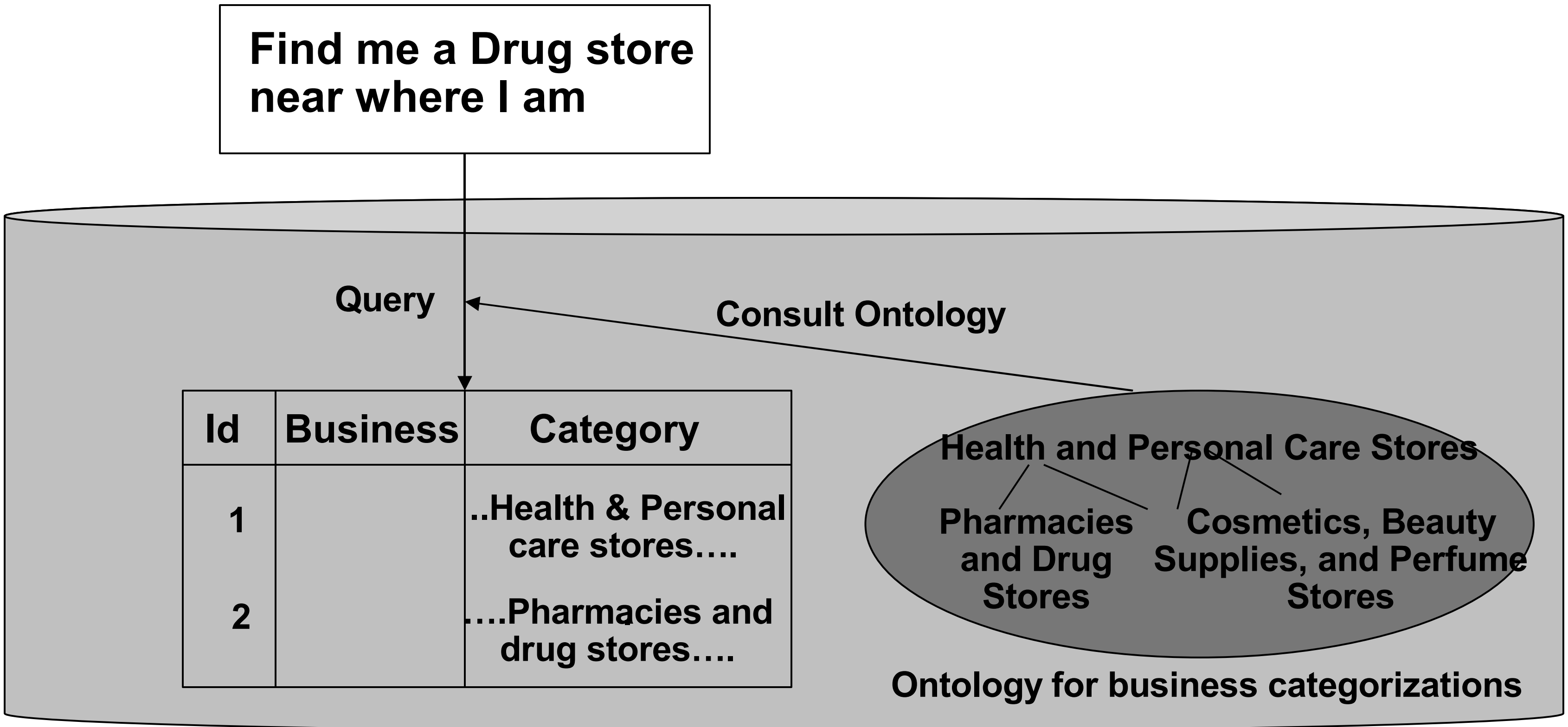
- Multiple users might use multiple sets of terms to annotate medical images
  - Difficult to search across multiple medical image repositories

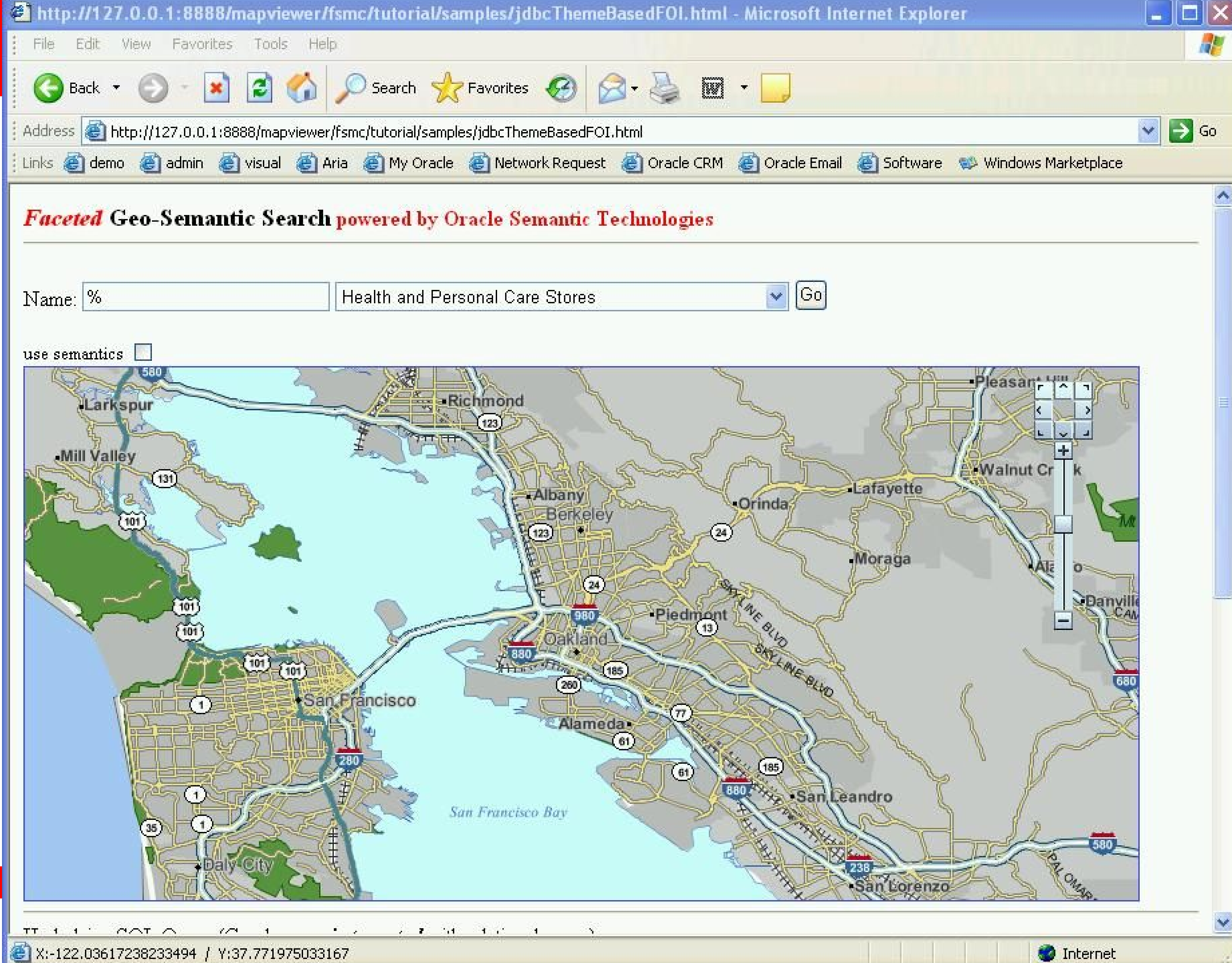


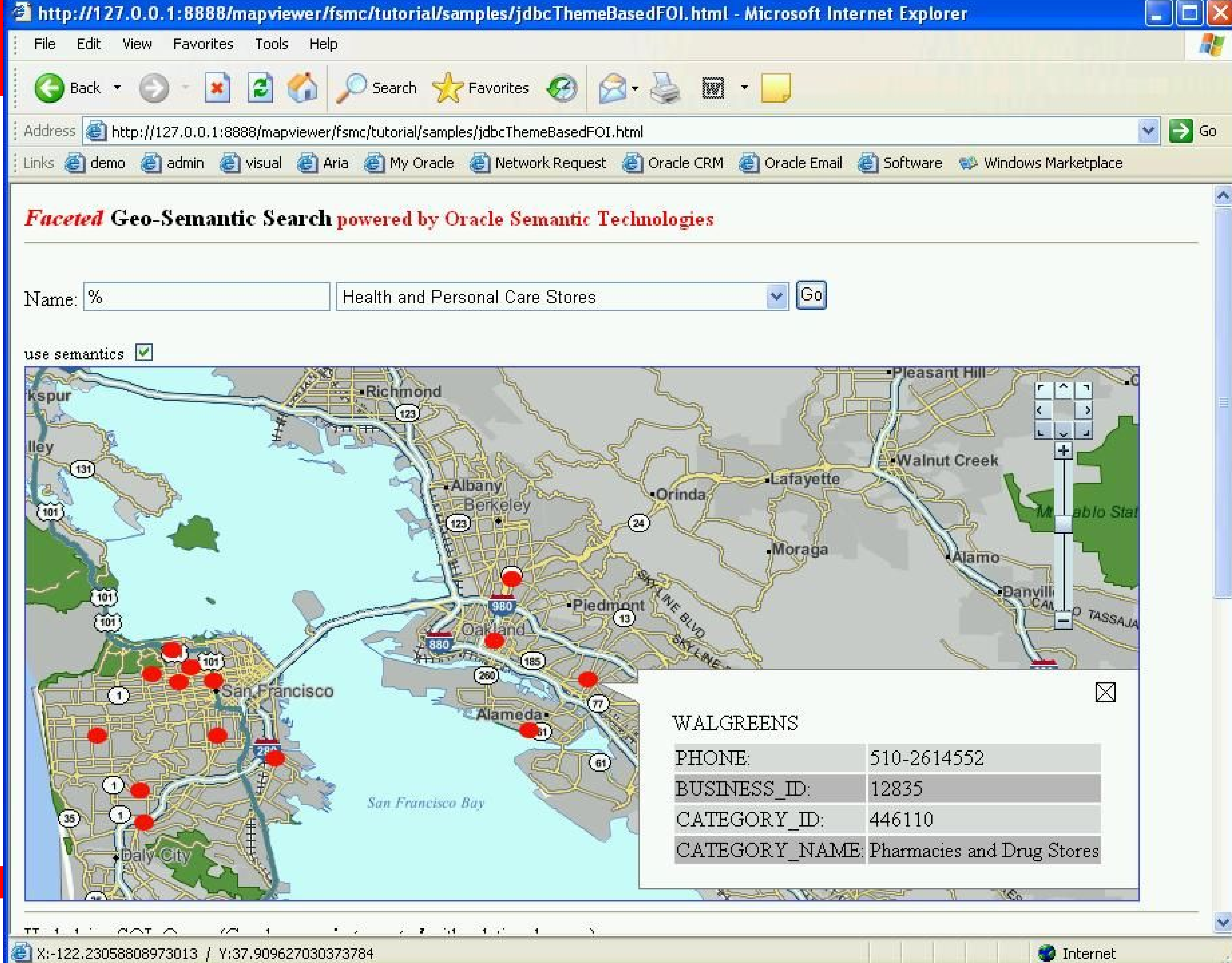
Semantics Enhanced Search

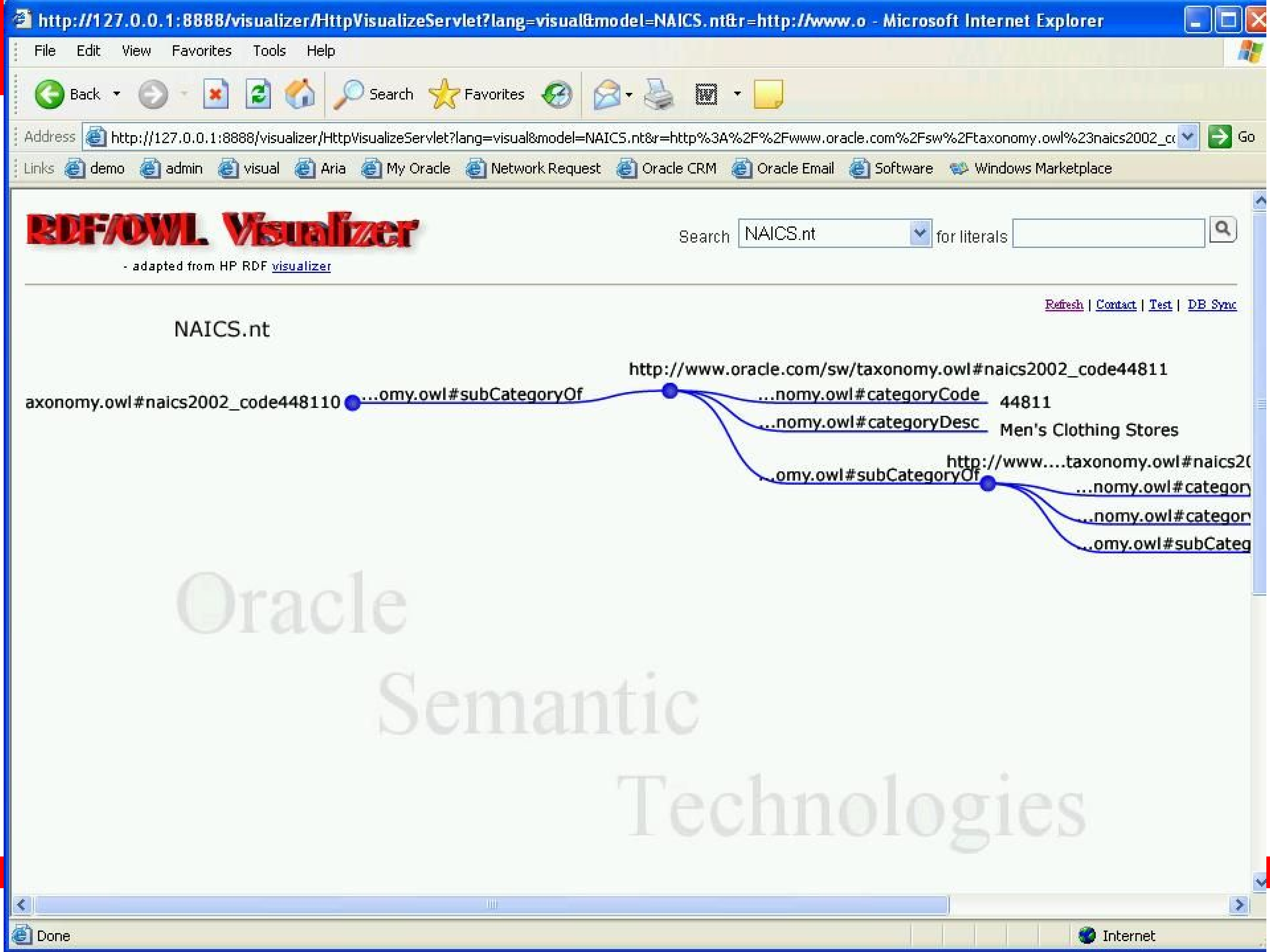
Geo-Semantics

- Enhance geo-spatial search with semantics
  - Create an ontology using business categorizations (from the NAICS taxonomy) and use that to enhance yellow pages type search




http://127.0.0.1:8888/mapviewer/fsmc/tutorial/samples/jdbcThemeBasedFOI.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://127.0.0.1:8888/mapviewer/fsmc/tutorial/samples/jdbcThemeBasedFOI.html Go
Links demo admin visual Aria My Oracle Network Request Oracle CRM Oracle Email Software Windows Marketplace
Faceted Geo-Semantic Search powered by Oracle Semantic Technologies
Name: %
Health and Personal Care Stores
Go
use semantics
Larkspur
Mill Valley
Richmond
Albany
Berkeley
Orinda
Lafayette
Moraga
Piedmont
Oakland
San Francisco
Alameda
San Leandro
San Lorenzo
Daly City
San Francisco Bay
SKY LINE BLVD
X:-122.03617238233494 / Y:37.771975033167
Internet


http://127.0.0.1:8888/mapviewer/fsmc/tutorial/samples/jdbcThemeBasedFOI.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
http://127.0.0.1:8888/mapviewer/fsmc/tutorial/samples/jdbcThemeBasedFOI.html
Go
Links demo admin visual Aria My Oracle Network Request Oracle CRM Oracle Email Software Windows Marketplace
Faceted Geo-Semantic Search powered by Oracle Semantic Technologies
Name: %
Health and Personal Care Stores
Go
use semantics
Richmond
Pleasant Hill
Albany
Berkeley
Orinda
Lafayette
Walnut Creek
Moraga
Alamo
Danville
Piedmont
Oakland
San Francisco
Alameda
Daly City
San Francisco Bay
WALGREENS
PHONE: 510-2614552
BUSINESS_ID: 12835
CATEGORY_ID: 446110
CATEGORY_NAME: Pharmacies and Drug Stores
X:-122.23058808973013 / Y:37.9096270030373784
Internet


http://127.0.0.1:8888/visualizer/HttpVisualizeServlet?lang=visual&model=NAICS.nt&r=http://www.o - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
http://127.0.0.1:8888/visualizer/HttpVisualizeServlet?lang=visual&model=NAICS.nt&r=http%3A%2F%2Fwww.oracle.com%2Fsw%2Ftaxonomy.owl%23naics2002_c
Go
Links demo admin visual Aria My Oracle Network Request Oracle CRM Oracle Email Software Windows Marketplace
RDF/OWL Visualizer
- adapted from HP RDF visualizer
Search NAICS.nt for literals
Refresh | Contact | Test | DB Sync
NAICS.nt
http://www.oracle.com/sw/taxonomy.owl#naics2002_code44811
axonomy.owl#naics2002_code448110
...omy.owl#subCategoryOf
...nomy.owl#categoryCode
44811
...nomy.owl#categoryDesc
Men's Clothing Stores
...omy.owl#subCategoryOf
http://www....taxonomy.owl#naics2(
...nomy.owl#categor
...nomy.owl#categor
...omy.owl#subCateg
Oracle
Semantic
Technologies
Done
Internet

Biosurveillance

- Biosurveillance application: Track patterns in health data
- Data from 8 emergency rooms in Houston at 10 minute intervals
- Data converted into RDF/OWL and loaded into the database
- 8 months data is 600M+ triples
- Automated analysis of data to track patterns:
  - Spike in flu-like symptoms (RDF/OWL inferencing to identify a flu-like symptom)
  - Spike in children under age 5 coming in

Data Integration in the Life Sciences

"Find all pieces of information associated with a specific target"

- Data integration of multiple datasets
  - Across multiple representation formats, granularity of representation, and access mechanisms
  - Across In-house and public sets (Gene Ontology, UniProt, NCI thesaurus, etc.).
- Standardized and machine-understandable data format with an open data access model is necessary to enable integration
  - Data-warehousing approach represents all data to be integrated in RDF/OWL
  - Semantic metadata layer approach links metadata from various sources and maps data access tool to relevant source
- Ability to combine RDF/OWL queries with relational queries is a big benefit
- Lilly and Pfizer are using semantic technology to solve data integration problems

Use Case: SenseLab Overview

SenseLab

Login

The SenseLab Project is a long term effort to build integrated, multidisciplinary models of neurons and neural systems, using the olfactory pathway as a model. This is one of a number of projects funded as part of the Human Brain Project whose aim is to develop neuroinformatics tools in support of neuroscience research. The project involves novel informatics approaches to constructing databases and database tools for collecting and analyzing neuroscience information, and providing for efficient interoperability with other neuroscience databases.

- Overview
- Membrane Properties Resource

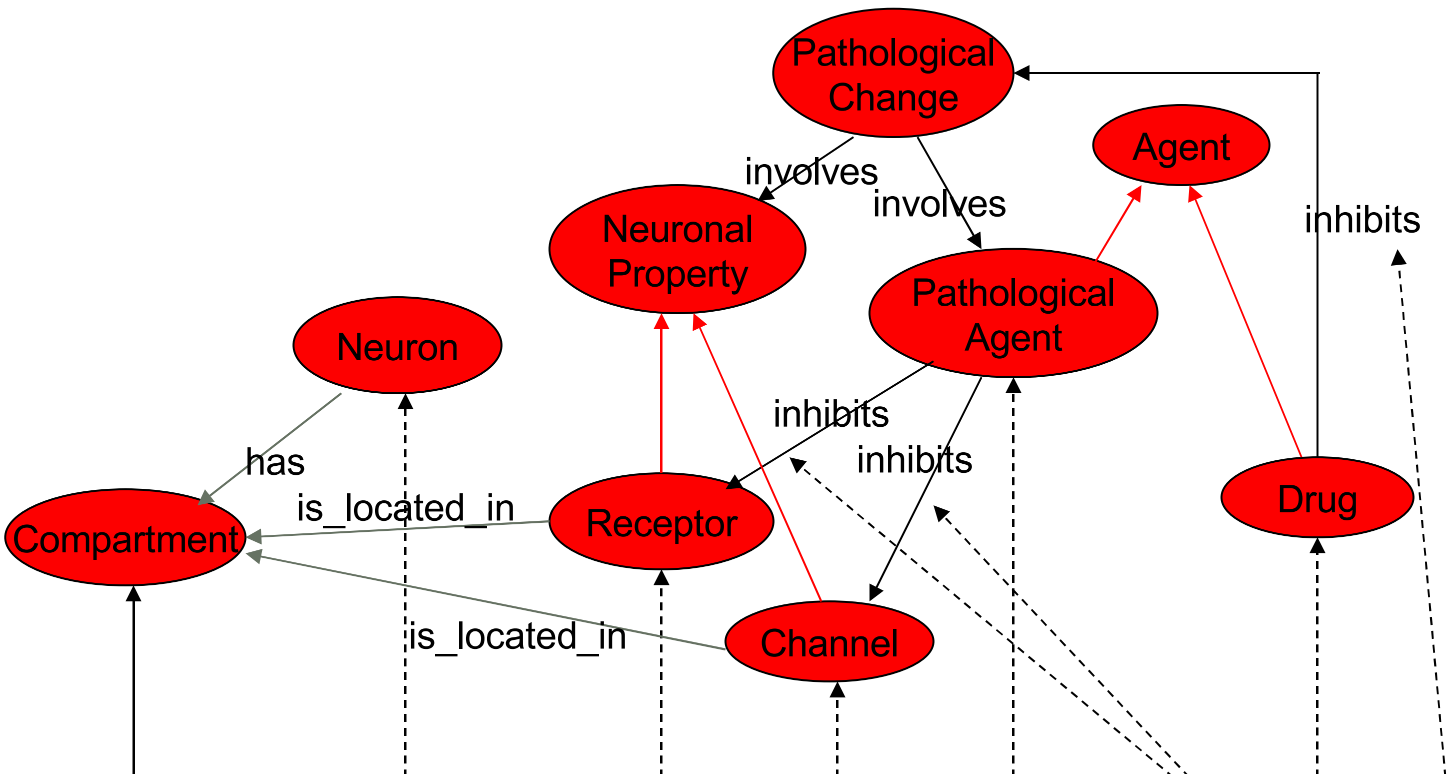
Brain Database Research

Neuronal Databases	CellPropDB	NeuronDB	ModelDB
Olfactory Databases	ORDB	OdorDB	OdorMapDB
Disease Databases	BrainPharm		

Part of this work published in the Workshop on Semantic e-Science

Courtesy: SenseLab, Yale University

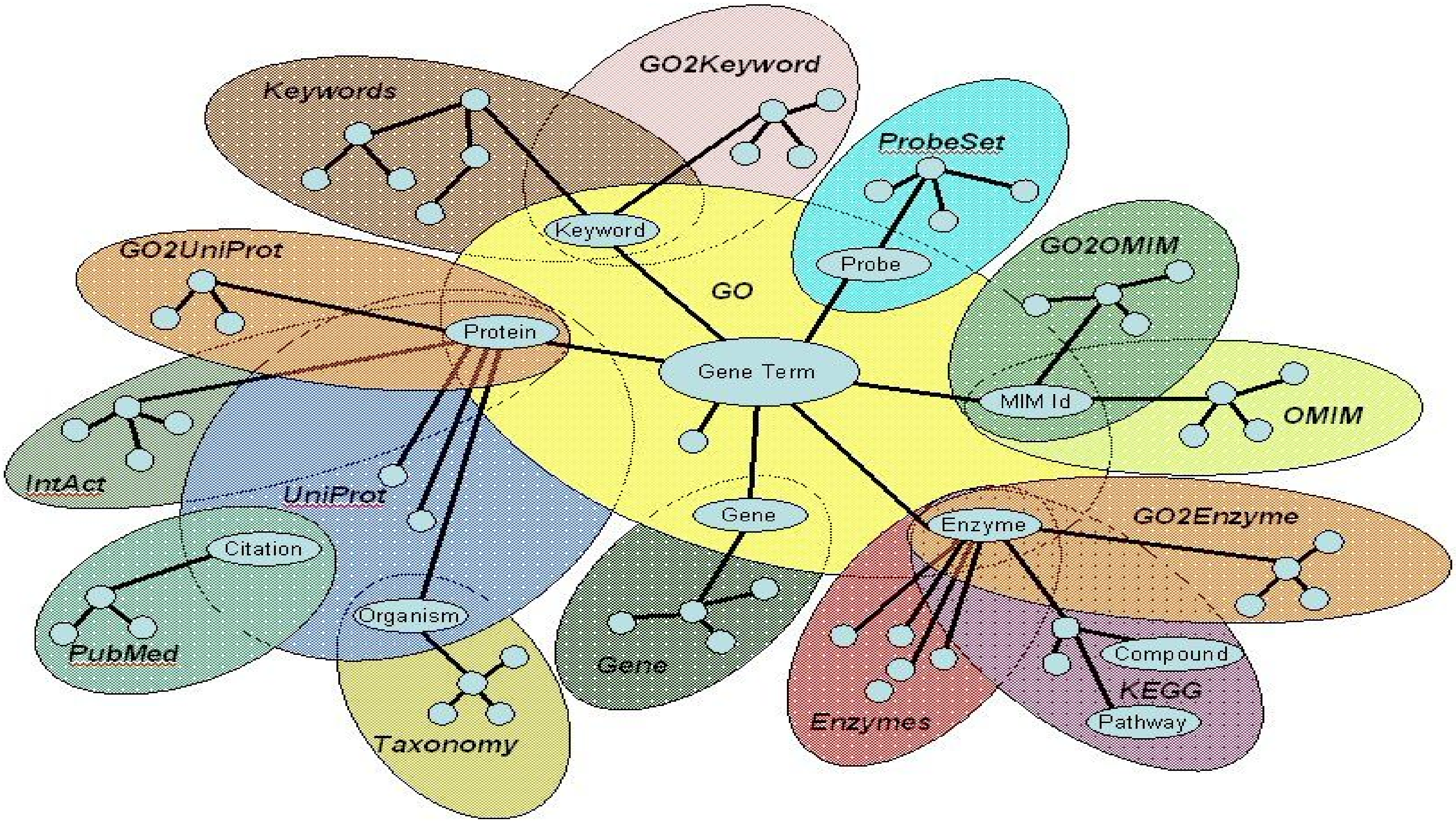
Relational to Ontological Mapping



Compartment	Cell: NeuronDB	Receptor	Channel	Pathological Agent (PA)	PA Action	Drug	Drug Action	Stage	Note	Detail
Soma	CA1 pyramidal neuron		I A	beta Amyloid	Inhibits			Early	View	66240
	Olfactory bulb mitral cell	GabaA						Early	View	66750
Dendrite	CA1 pyramidal neuron		I A	beta Amyloid	Inhibits			Early	View	66240
	Olfactory bulb mitral cell	GabaA						Early	View	66750
Unspecified	Oocyte		I L high threshold	beta Amyloid	Inhibits			Early	View	66252
								Early	View	66753
	CA1 pyramidal neuron			beta Amyloid	Inhibits			Early	View	66758
	CA1 pyramidal neuron	NMDA	I Calcium	beta Amyloid	Inhibits		Inhibits		View	66250

Courtesy: SenseLab, Yale University

Use Case: Integrated Bioinformatics Data

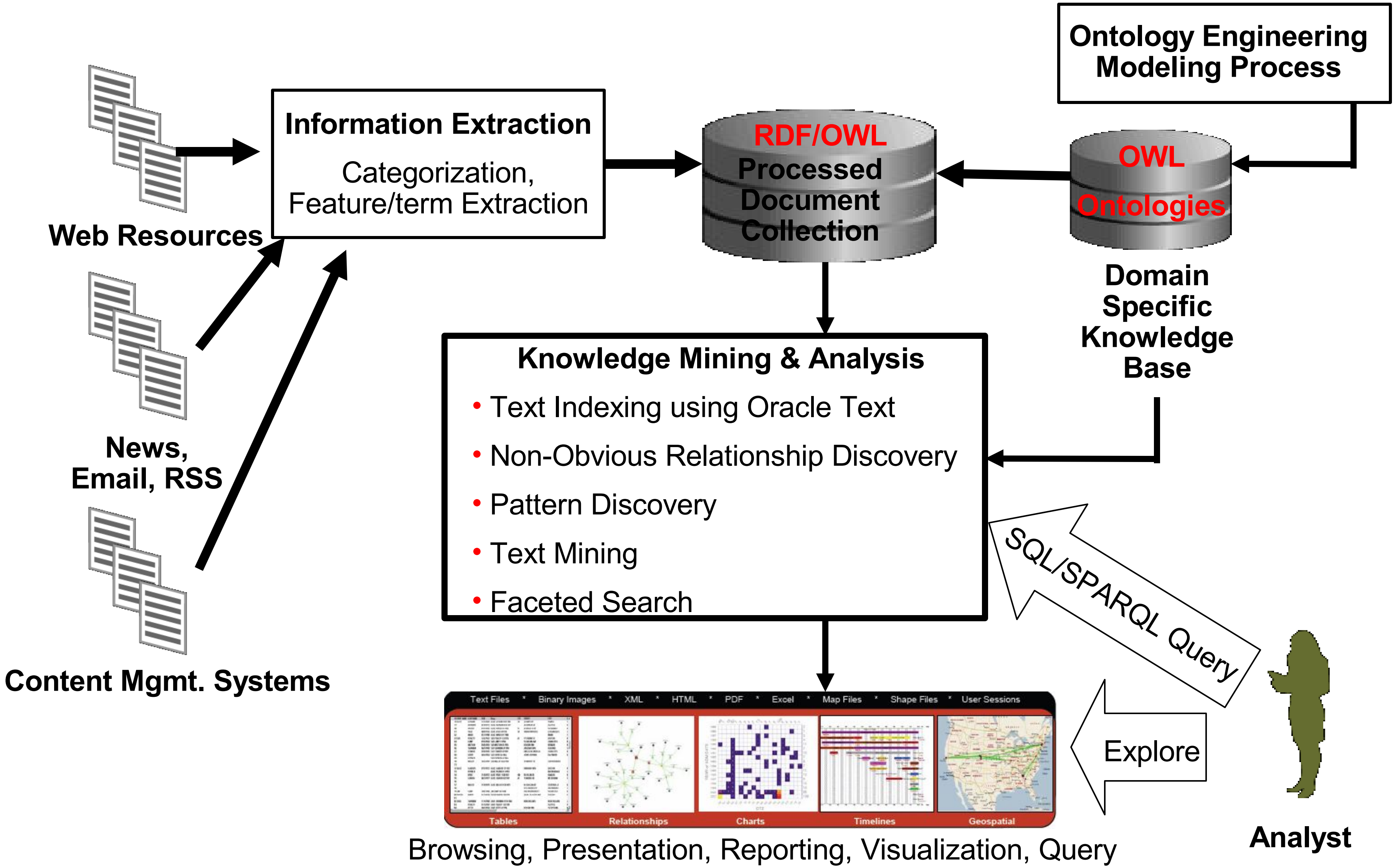


ORACLE

Part of this work published in Journal of Web Semantics

Source: Siderean Software

Use Case: Knowledge Mining Solutions



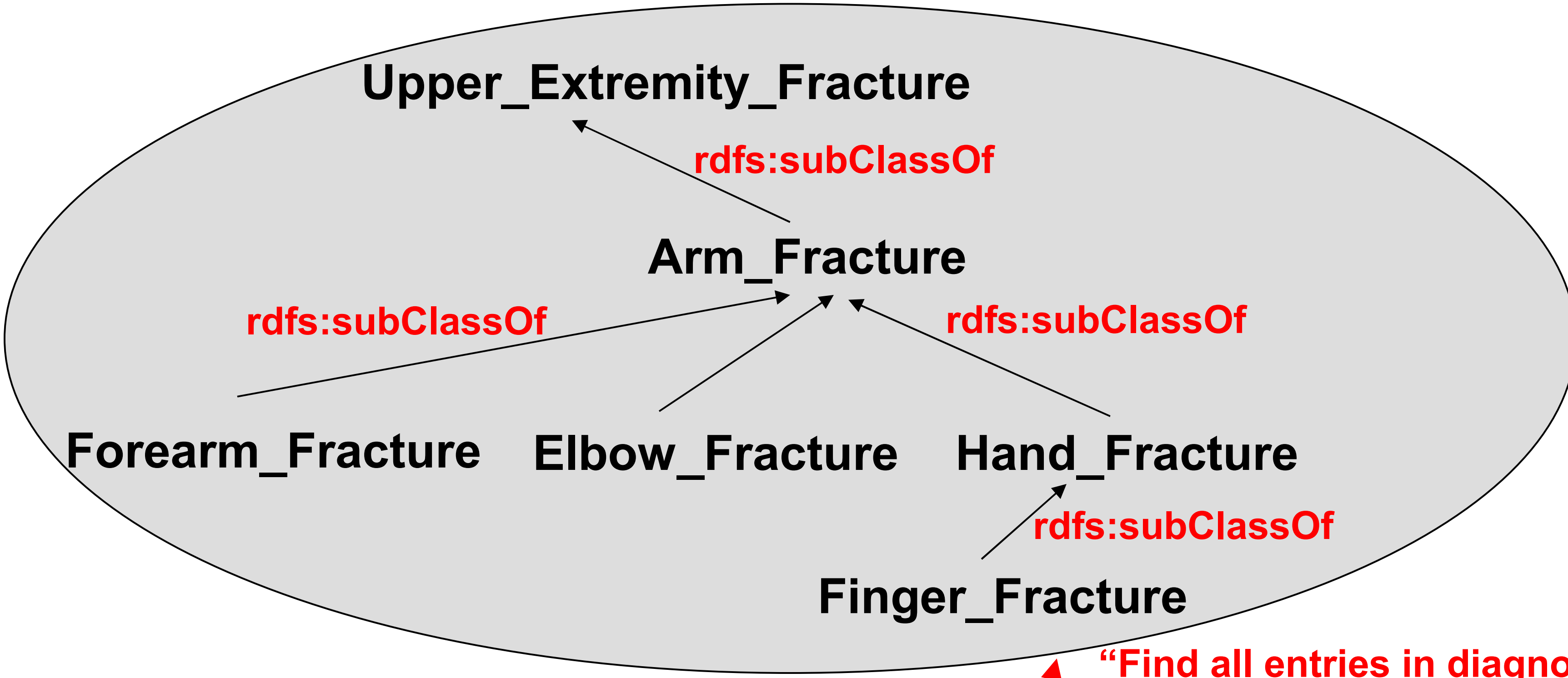
Safe Harbor Statement & Confidentiality

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.

Semantic Operators in SQL

- Two new first class SQL operators to semantically query relational data by consulting an ontology
  - SEM_RELATED (<col>,<pred>, <ontologyTerm>, <ontologyName> [,<invoc_id>])
  - SEM_DISTANCE (<invoc_id>) ← Ancillary Oper.
  - Can be used in any SQL construct (ORDER BY, GROUP BY, SUM, etc.)
- Semantic indextype
  - An index of type semantic indextype introduced for efficient execution of queries using the semantic operators

Ontology-assisted Query



“Find all entries in diagnosis column that are related to ‘Upper_Extremity_Fracture’”

Patients

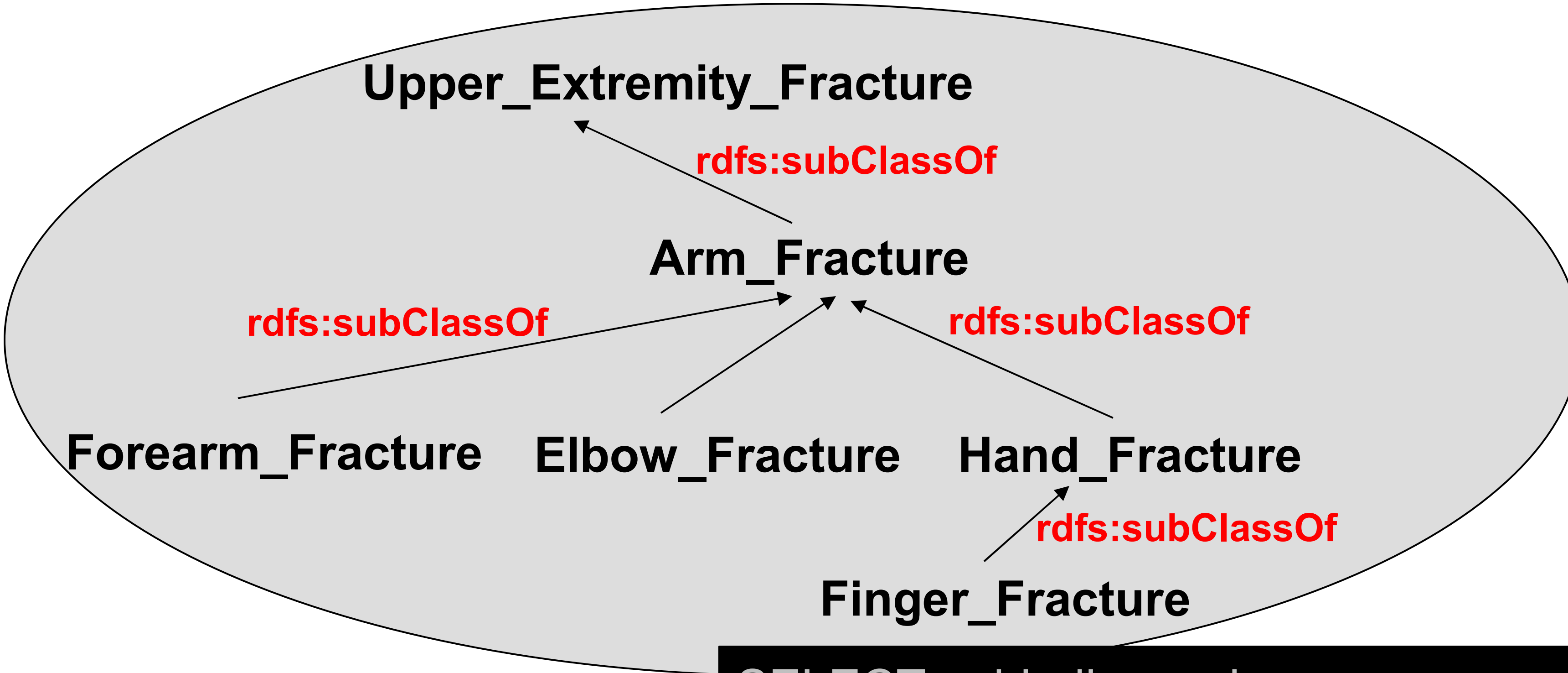
ID	DIAGNOSIS
1	Hand_Fracture
2	Rheumatoid_Arthritis

Syntactic query will not work:

SELECT p_id, diagnosis FROM Patients WHERE diagnosis = ‘Upper_Extremity_Disorder’;

ORACLE

Ontology-assisted Query

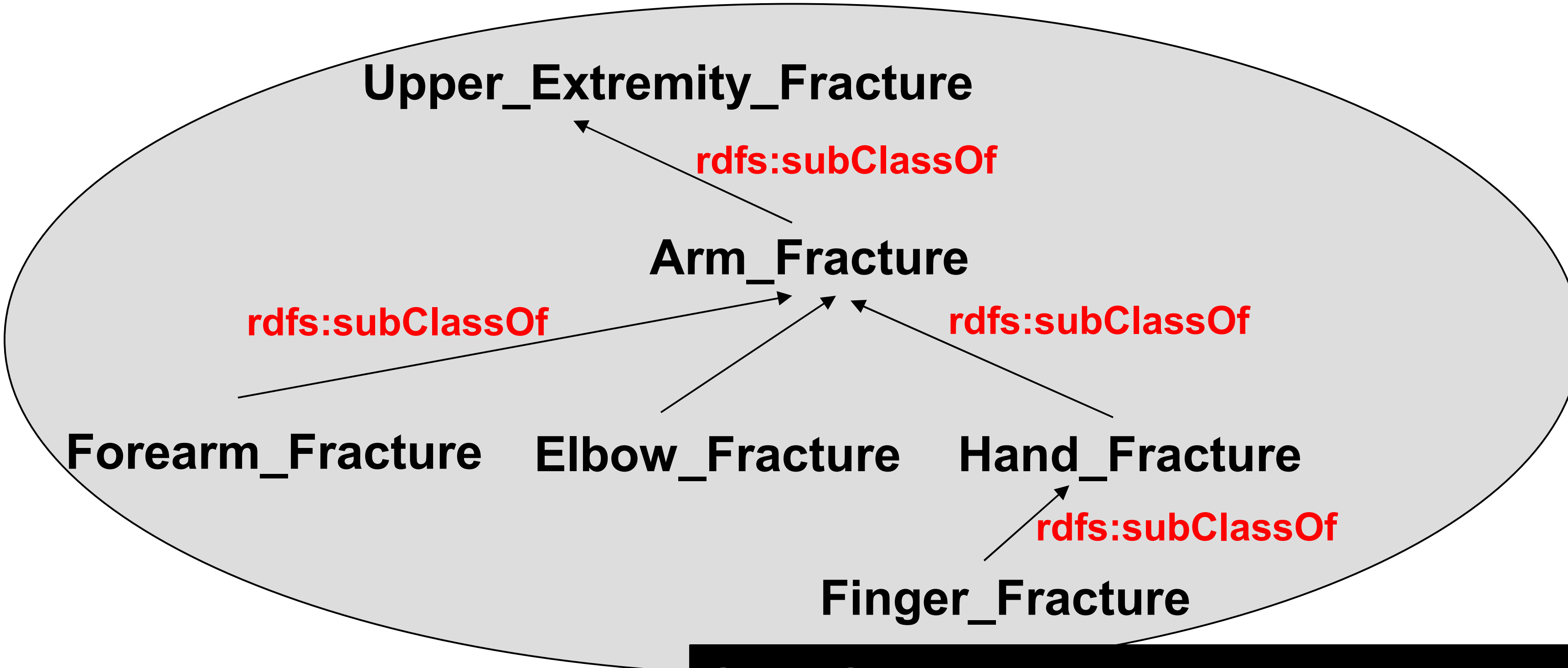


Patients

ID	DIAGNOSIS
1	Hand_Fracture
2	Rheumatoid_Ar

```
SELECT p_id, diagnosis
FROM Patients
WHERE SEM_RELATED (
        diagnosis,
        'rdfs:subClassOf',
        'Upper_Extremity_Fracture',
        'Medical_ontology') = 1;
```

Ontology-assisted Query



Patients

ID	DIAGNOSIS
1	Hand_Fractur
2	Rheumatoid_

```
SELECT p_id, diagnosis
FROM Patients
WHERE SEM_RELATED (
        diagnosis,
        'rdfs:subClassOf',
        'Upper_Extremity_Fracture',
        'Medical_ontology' = 1
AND SEM_DISTANCE() <= 2;
```

Summary

- Semantic Technology support in the database
  - Store RDF/OWL data and ontologies
  - Infer new RDF/OWL triples via native inferencing
  - Query RDF/OWL data and ontologies
  - Ontology-Assisted Query of relational data

  - More information at:

    http://www.oracle.com/technology/tech/semantic_technologies/index.html

ORACLE