



Bringing the Semantic Web Closer to its Tipping Point

*by Lowering the Cost of Data and Content Integration and enabling Searching
and Querying over Billions of Facts on the Web*

Also, showing
How linked data established Nietzsche as the most popular German entertainer

July 2010

Presentation Outline

- **Introducing Ontotext**
 - Why RDF is Good for Data Integration
 - The benefits of light-weight inference
 - Interlinking Text and Data; Hybrid Search
- **What makes the OWLIM semantic repository special**
 - Best Scalability and Query Efficiency
 - Resilient Cluster Setup for Critical Query Loads
 - Optimized for Data Integration
- **Unique Linked Data Management Expertise**
 - Linked Data: Introduction and Challenges
 - FactForge: Fast Track to the Center of the Web of Data

Ontotext

- **Semantic technology developer** est. in year 2000
- **Global leader** in semantic databases and semantic annotation
- **Staff: 50** employees and multiple contractors
- Investment acquired in July 2008
 - A financial investor obtained minority share in a deal for 2.5M Euro
- Involved in several joint ventures:
 - **Innovantage**: online recruitment intelligence provider in UK
 - **Namerimi**: national search engine in Bulgaria

Ontotext Positioning

- **Leading semantic technology provider**
 - Top-5 core semantic technology developer
 - Supplying engines and components to vendors and solution developers
- **Unique technology portfolio:**
 - **Semantic Databases:** high-performance RDF DBMS, scalable reasoning
 - **Semantic Search:** text-mining (IE), Information Retrieval (IR)
 - **Web Mining:** focused crawling, screen scraping, data fusion
 - **Web Services and BPM:** WS annotation, discovery, etc.
- **Good recognition in the SemTech community**
 - Ontotext pages are ranked #1 for “semantic annotation” and “semantic repository” at GYM

Customer Base (selected)

- **The British Broadcasting Corporation (BBC)**
 - Runs its World Cup 2010 site on top of BigOWLIM
 - Learn more at <http://www.ontotext.com/owlim/in-use.html#bbc>
- **The National Archives**
 - The UK Government's official archive contracted Ontotext to implement semantic search for the Government Web Archive
- **AstraZeneca**
 - Analysis and retrieval of clinical trial reports
 - Integration of biomedical databases for drug target identification

All of the above need to integrate massive amounts of heterogeneous data and provide efficient search and

Ontotext – Partners and Research Funding

- Network of **technology partners**
 - GATE team (**UK**: Sheffield University)
 - TopQuadrant (**USA**: Mountain View, CA; Alexandria, VA)
 - Profium (**Finland**)
 - System Simulations, Talis (**UK**)
 - BPEng (**Italy**: Trento)
 - Saltlux (**Korea**)
- Part of EC **research projects** with total budget above 100 MEuro
 - 3M Euro research grants secured for Ontotext for 2010-2014
 - Partnering with SAP, IBM, Wikimedia, Google Labs, BT, Telefonica, KT, and tens of the leading European universities

Presentation Outline

- Introducing Ontotext
 - **Why RDF is Good for Data Integration**
 - The benefits of light-weight inference
 - Interlinking Text and Data: Hybrid Search
- What makes the OWLIM semantic repository special
 - Best Scalability and Query Efficiency
 - Resilient Cluster Setup for Critical Query Loads
 - Optimized for Data Integration
- Unique Linked Data Management Expertise
 - Linked Data: Introduction and Challenges
 - FactForge: Fast Track to the Center of the Web of Data

Why RDF is Good for Data Integration

- RDF data does not require ‘hard’ schemas
 - As in the “column stores” (e.g. BigTable, CStore, etc.)
 - The physical representation is independent from the logical scheme
- Designed as a data representation for the Web
 - Datasets can be combined, even when they have ‘conflicting’ schemas or vocabularies
 - The sources of data can be explicitly exposed and tracked
- Linking different identifiers for the same concepts across datasets is easy (owl:sameAs)
 - At the same time data can be merged without need for identifier re-writing – everything is based on globally unique identifiers

Physical data representation: RDBMS vs. RDF

| Person | | |
|--------|----------|--------|
| ID | Name | Gender |
| 1 | Maria P. | F |
| 2 | Ivan Jr. | M |
| 3 | ... | |

| Parent | |
|--------|-------|
| ParID | ChiID |
| 1 | 2 |
| ... | |

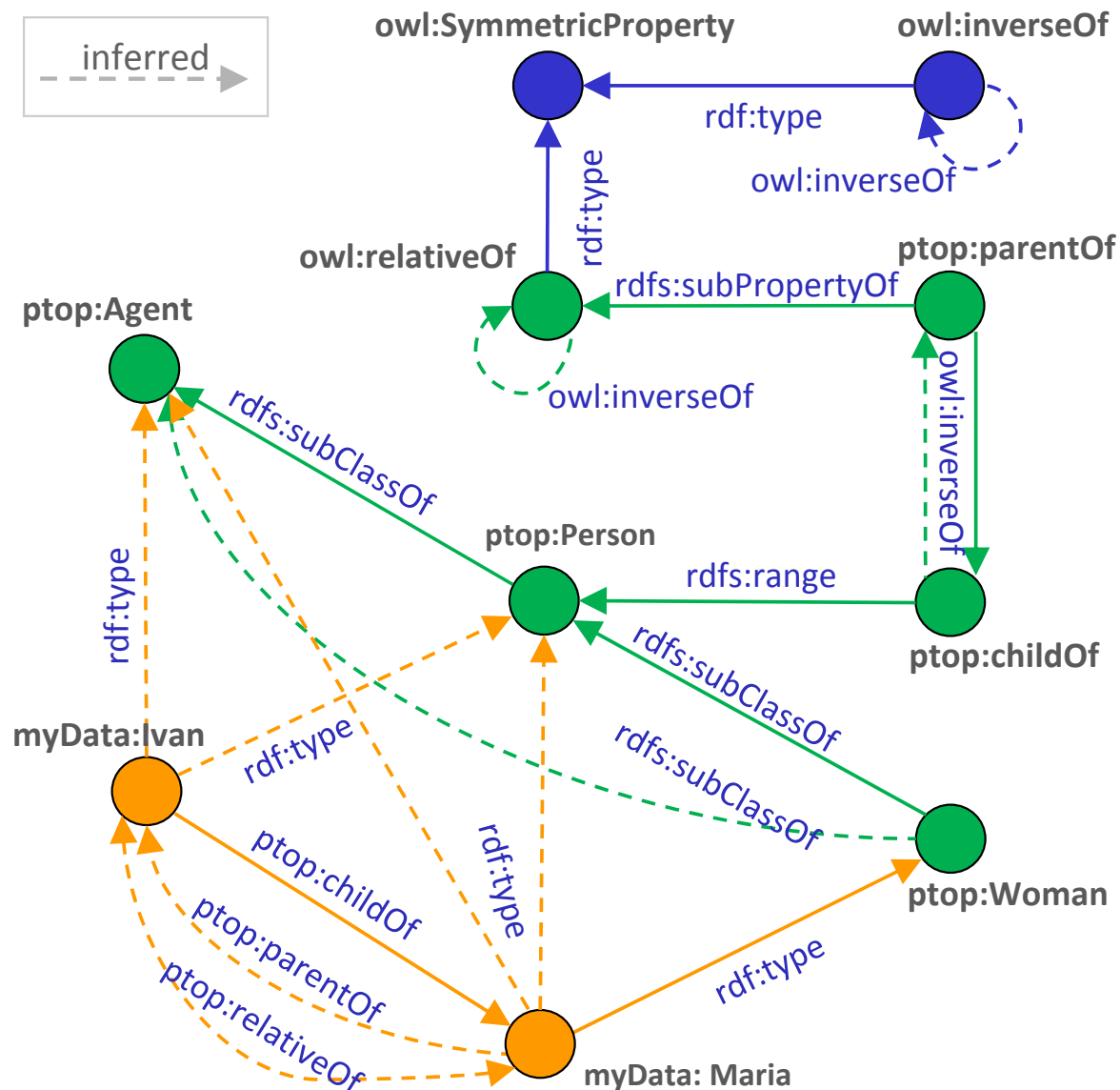
| Spouse | | | |
|--------|------|------|----|
| S1ID | S2ID | From | To |
| 1 | 3 | | |
| ... | | | |

Relational Tables

| Statement | | |
|------------|------------|---------------|
| Subject | Predicate | Object |
| myo:Person | rdf:type | rdfs:Class |
| myo:gender | rdfs:type | rdfs:Property |
| myo:parent | rdfs:range | myo:Person |
| myo:spouse | rdfs:range | myo:Person |
| myd:Maria | rdf:type | myo:Person |
| myd:Maria | rdf:label | "Maria P." |
| myd:Maria | myo:gender | "F" |
| myd:Maria | rdf:label | "Ivan Jr." |
| myd:Ivan | myo:gender | "M" |
| myd:Maria | myo:parent | myd:Ivan |
| myd:Maria | myo:spouse | myd:John |
| ... | | |

RDF Representation

RDF Features a Graph Data Model



The Benefits of Lightweight Semantics

We build upon **lightweight semantics** that are easy to understand, deploy, and manage

For instance, think of ontologies as database schemata with simple interpretation rules. Plenty of obvious (but useful) implicit facts can be inferred and match queries right away

Lightweight Inference - Simple Rules

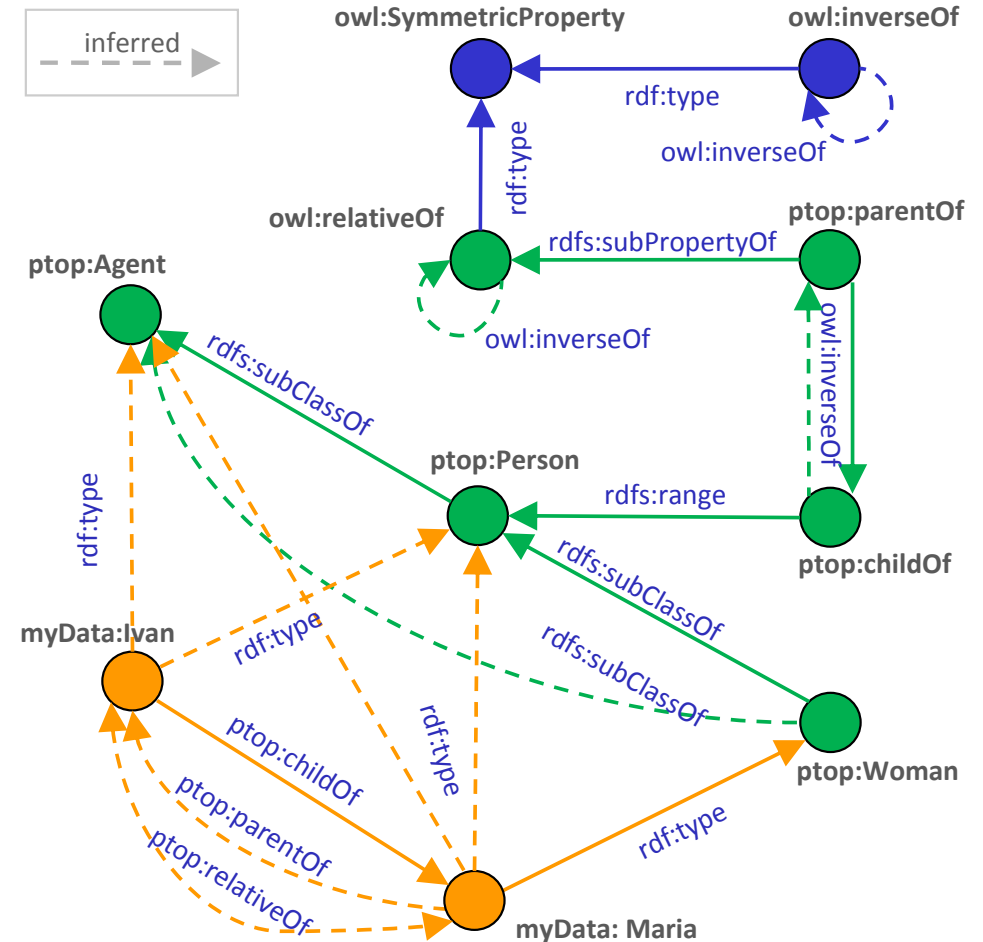
```
<C1,rdfs:subClassOf,C2>  
<C2,rdfs:subClassOf,C3>  
⇒ <C1,rdfs:subClassOf,C3>
```

```
<I,rdf:type,C1>  
<C1,rdfs:subClassOf,C2>  
⇒ <I,rdf:type,C2>
```

```
<P1,owl:inverseOf,P2>  
<I1,P1,I2>  
⇒ <I2,P2,I1>
```

```
<P1,rdf:type,owl:SymmetricProperty>  
⇒ <P1,owl:inverseOf,P1>
```

The rule entailment language used by OWLIM is a simplification of Datalog, used in DBMS since the 1980's



Lightweight Inference - Simple Rules

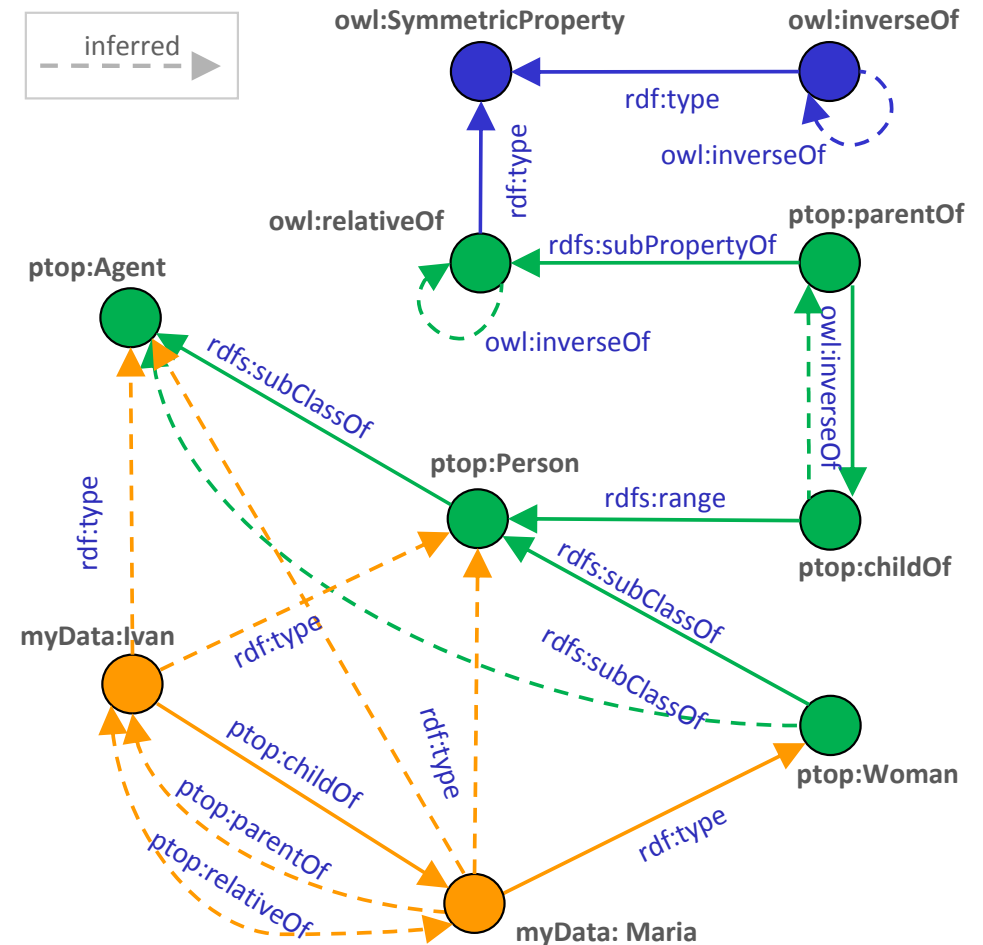
The database will return 'Ivan' as a result of a query for

Maria relativeOf ?x

when the fact asserted was

Ivan childOf Maria

This type of “intelligence” can be achieved in many ways, but semantic repositories offer the cleanest approach, delivering best efficiency and lowest cost through the entire data lifecycle

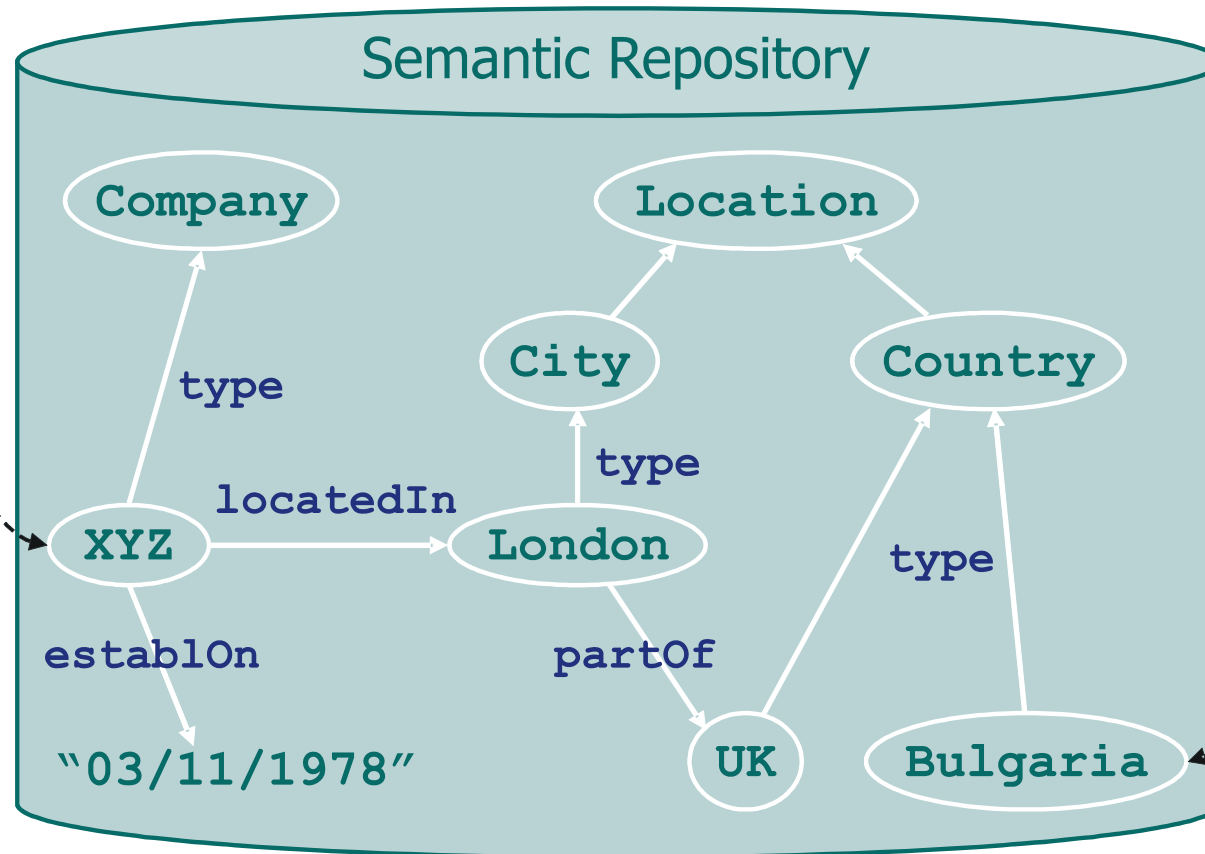


Presentation Outline

- Introducing Ontotext
 - Why RDF is Good for Data Integration
 - The benefits of light-weight inference
 - **Interlinking Text and Data: Hybrid Search**
- What makes the OWLIM semantic repository special
 - Best Scalability and Query Efficiency
 - Resilient Cluster Setup for Critical Query Loads
 - Optimized for Data Integration
- Unique Linked Data Management Expertise
 - Linked Data: Introduction and Challenges
 - FactForge: Fast Track to the Center of the Web of Data

Interlinking Text and Data

XYZ announced profits in Q3, planning to build a \$120M plant in Bulgaria, and more and more and more and more text ..



Highlight, Hyperlink, Explore and Navigate

The image shows two overlapping windows from a Microsoft Internet Explorer browser. The left window, titled 'KIM Explorer - Microsoft Internet Explorer', displays a table of properties for 'The Associated Press, a NewsAgency, Trustedtip!'. The table has two columns: 'Property' and 'Value'. Below this is a 'Related Entities' section with a table of 'Resource' and 'Link to The Associated Press'. The right window shows a news article titled 'Moving on New Front to Cut'. A large orange arrow points from the 'May 11, 2004 7:46 AM' timestamp in the KIM Explorer window to the same timestamp in the news article. Another orange arrow points from the 'Environmental Protection Agency' link in the news article to the 'correspondent' link in the KIM Explorer window. The KIM Explorer window also shows a sidebar with a list of classes (BusinessObject, InformationResource, Location, Statement, Vehicle) and a 'Place Links' checkbox.

| Property | Value |
|-----------|----------------------|
| hasAlias | The Associated Press |
| hasAlias | AP |
| hasAlias | Associated Press |
| locatedIn | New York |
| locatedIn | New York |

| Resource | Link to The Associated Press |
|---------------|------------------------------|
| correspondent | withinOrganization |
| correspondent | withinOrganization |
| correspondent | withinOrganization |
| correspondent | withinOrganization |

Copyright © 2004 Ontotext Lab, Sirma AI, Bulgaria

Classes: ☒ BusinessObject, ☒ InformationResource, ☒ Location, ☒ Statement, ☒ Vehicle

Place Links: ☒

News Article: Moving on New Front to Cut
May 11, 2004 7:46 AM
SEF HEBERT
ed Press W
GTON (AP) - The government is moving on a new front to cut air pollution. This time ferry boats and harbor tugs, farm tractors and train locomotives, and dirt movers at construction sites are the targets.
The Environmental Protection Agency is issuing new regulations aimed at cutting the amount of smog-causing chemicals and fine soot that comes from these off-road diesel-powered vehicles

The KIM Platform

- A platform offering services and infrastructure for:
 - automatic semantic annotation of text
 - text mining
 - semantic indexing and retrieval of content
 - query and navigation across heterogeneous text and data

- KIM can match a query like:

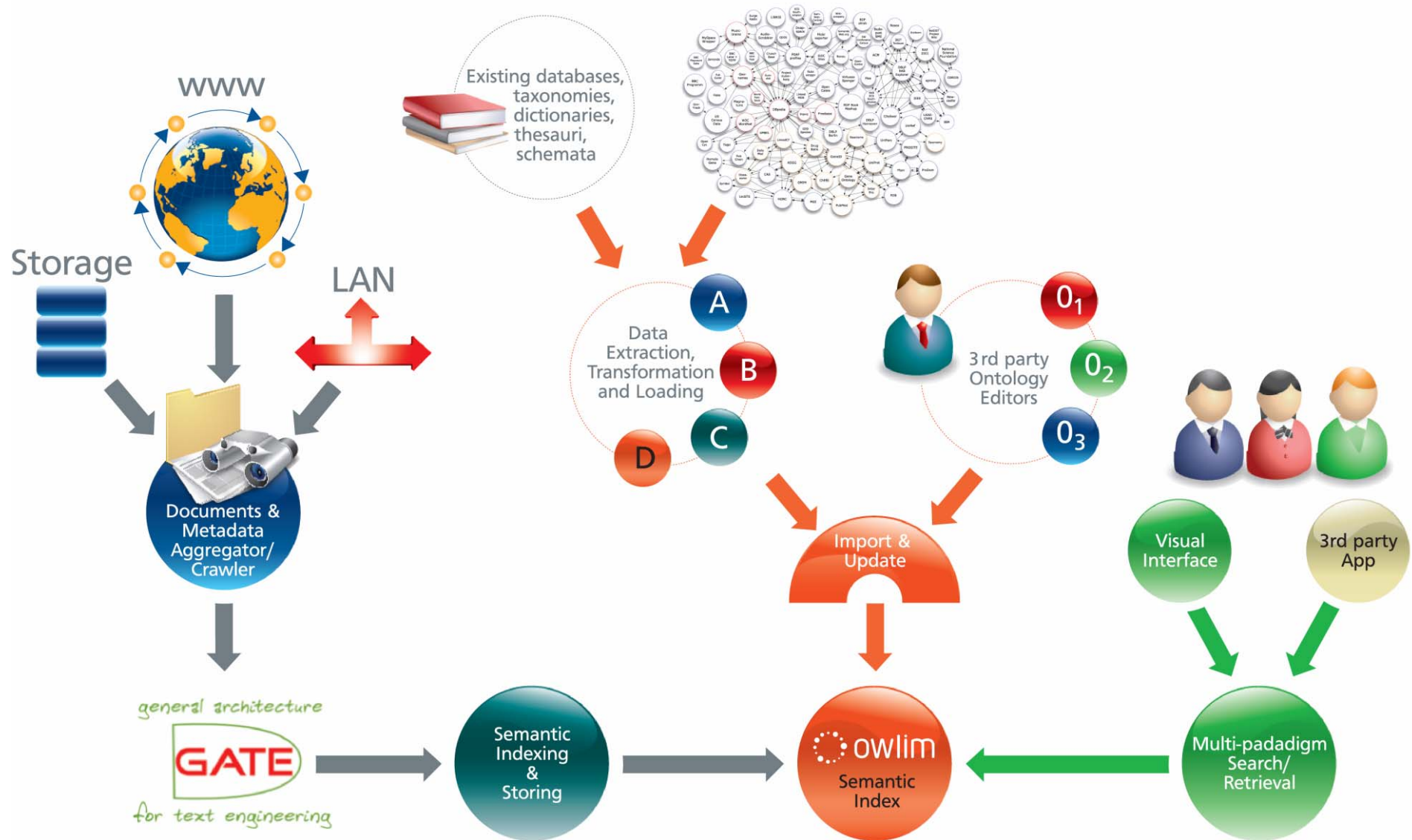
Documents about a telecom company in Europe, John Smith, and a date in the first half of 2002.

- With a document containing:

At its meeting on the 10th of May, the board of Vodafone appointed John G. Smith as CTO

- “vanilla” Information Retrieval fails to deliver

Semantic Annotation and Search Ecosystem



Elevator Pitch

We link your data, your content, and the web!

In **10 weeks** we can build a solution that:

- integrates **10 databases** with the linked data cloud
- mines **10 million documents** and web pages

and lets you search and navigate all this information

- in **10 different ways**
- from a **\$10,000 server**

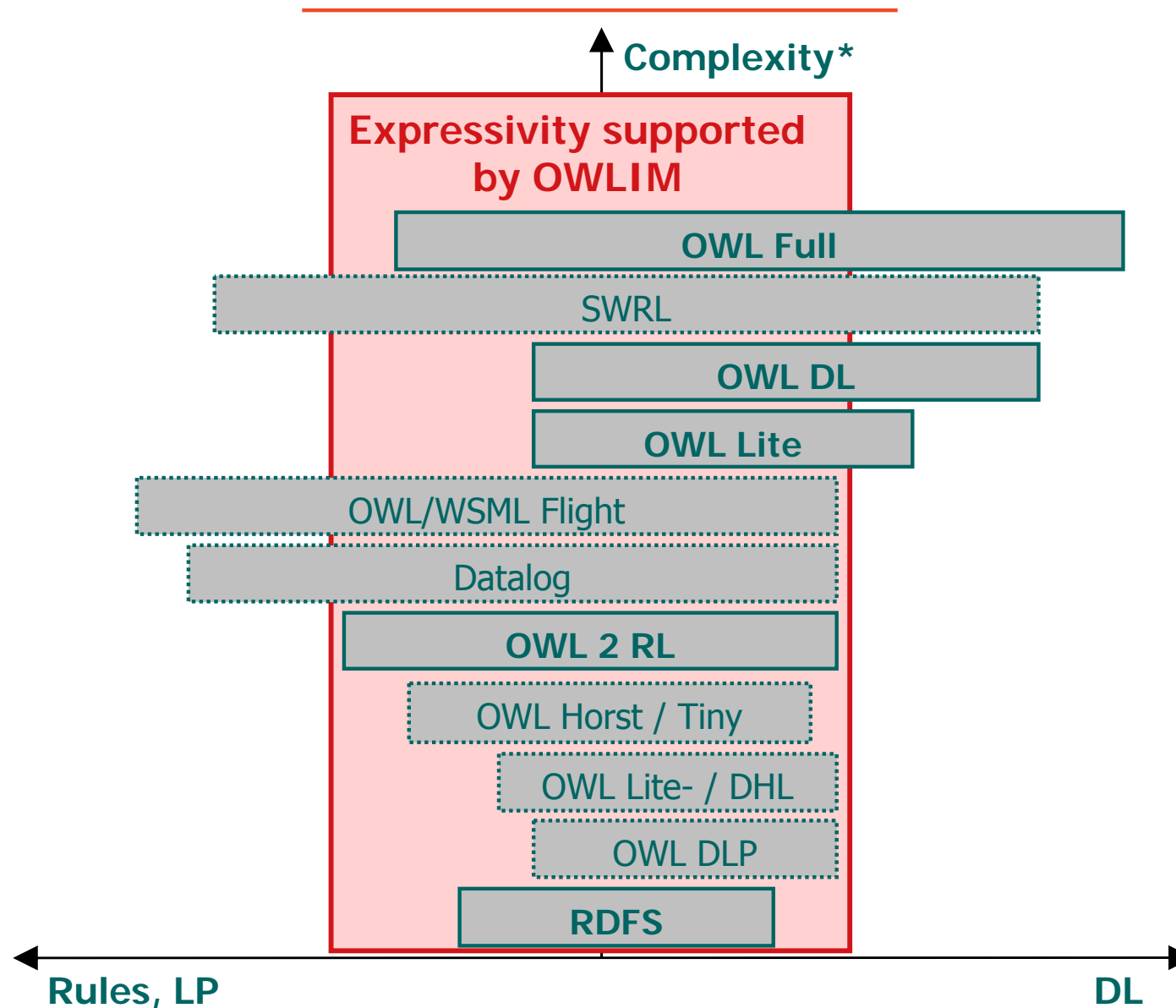
Presentation Outline

- Introducing Ontotext
 - Why RDF is Good for Data Integration
 - The benefits of light-weight inference
 - Interlinking Text and Data: Hybrid Search
- **What makes the OWLIM semantic repository special**
 - Best Scalability and Query Efficiency
 - Resilient Cluster Setup for Critical Query Loads
 - Optimized for Data Integration
- Unique Linked Data Management Expertise
 - Linked Data: Introduction and Challenges
 - FactForge: Fast Track to the Center of the Web of Data

Semantic Repository for RDFS and OWL

- OWLIM is a family of **scalable semantic repositories**
 - **SwiftOWLIM**: fast in-memory operations, scales to ~100M statements
 - **BigOWLIM**: the most efficient enterprise-grade engine, optimized for data integration, massive query loads and critical applications
- OWLIM is designed and tuned to provide
 - Efficient management, integration and analysis of heterogeneous data
 - Light-weight, high-performance reasoning
 - The inference is based on logical **rule-entailment**
 - **RDFS**, **OWL Horst** and **OWL2 RL** are supported
 - **Custom semantics** can be defined via rules and axiomatic triples

Naïve OWL Fragments Map



OWLIM in Use (selected)

- BigOWLIM is integrated into the **Semantic Web Publishing stack** powering the **BBC's 2010 World Cup Website**
 - http://www.bbc.co.uk/blogs/bbcinternet/2010/07/bbc_world_cup_2010_dynamic_sem.html
- BigOWLIM is used for data integration in:
 - The life sciences: LinkedLifeData.com platform is a public service consolidating 25 of the most popular biomedical databases. It provides search and SPARQL query facilities over some 4 billion statements
 - Linked Data: FactForge.net is one of the most advanced portals (slide 53)
- SwiftOWLIM is bundled in:
 - **GATE** – the most popular text-mining platform
 - **TopBraid Composer** – the most robust RDFS/OWL editor

BigOWLIM Excellence

- BigOWLIM is the only engine that can **reason with more than 10B** statements, on a \$10,000 server
- BigOWLIM offers the most **efficient query evaluation**
 - It is also the only engine for which full-cycle benchmarking results are published for the LUBM(8000) benchmark or higher
- BigOWLIM is the **most scalable RDF database engine**
 - It passes LUBM(90000), indexing over 20B explicit and implicit statements while still being able to answer queries efficiently
- **Multiple independent opinions** justify these claims
 - Please, refer to <http://www.ontotext.com/owlim/references.html>
 - In essence, all recent independent evaluations rank BigOWLIM as #1

BigOWLIM's Key Features

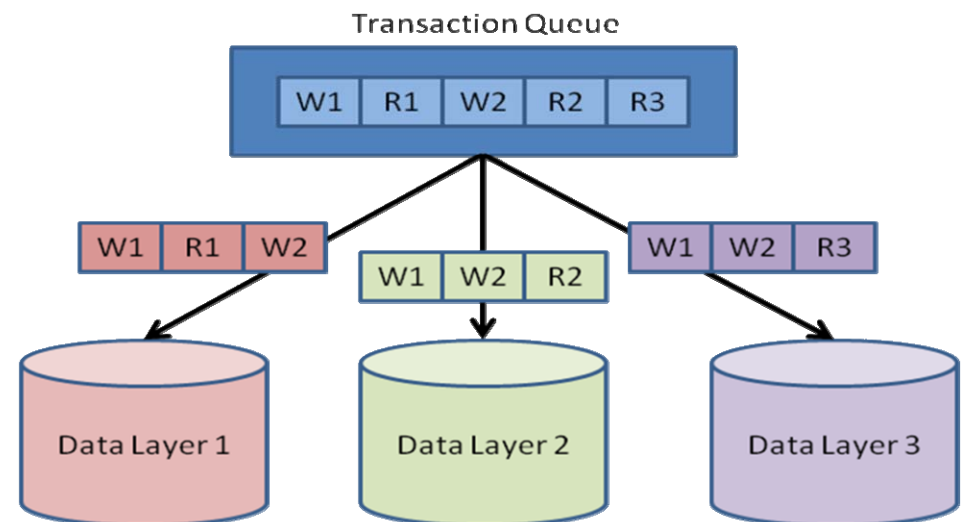
- **Pure Java implementation** compliant with Sesame
 - The latter brings interoperability benefits and support for all popular RDF syntaxes and query languages, including **SPARQL**
- **Clustering support** brings resilience, failover and horizontally scalable parallel query processing
- **Optimized owl:sameAs** handling
 - delivers dramatic improvements in performance and usability when huge volumes of data from multiple sources are integrated
- **Full-text search**, based on either Lucene or proprietary techniques

BigOWLIM's Key Features (2)

- **High performance retraction** of statements and their inferences
 - While forward-chaining and materialisation speed up query answering, this technique removes the performance degradation that materialisation-based systems face when retracting statements
- Powerful **consistency checking** mechanisms
- **RDF rank**, similar to Google's PageRank, can be calculated for the nodes in an RDF graph and used for ordering **query results by relevance**
- **Notification mechanism**, to allow clients to react to updates in the data stream

BigOWLIM Replication Cluster

- Distribution through data replication improves:
 - Scalability with respect to **concurrent user requests**
 - Resilience – failover, online re-configuration
- How does it work?
 - Each data write request is multiplexed to all repository instances
 - Each read request is dispatched to one instance only
 - To ensure **load-balancing**, read requests are sent to the instance with the smallest execution queue at this point in time

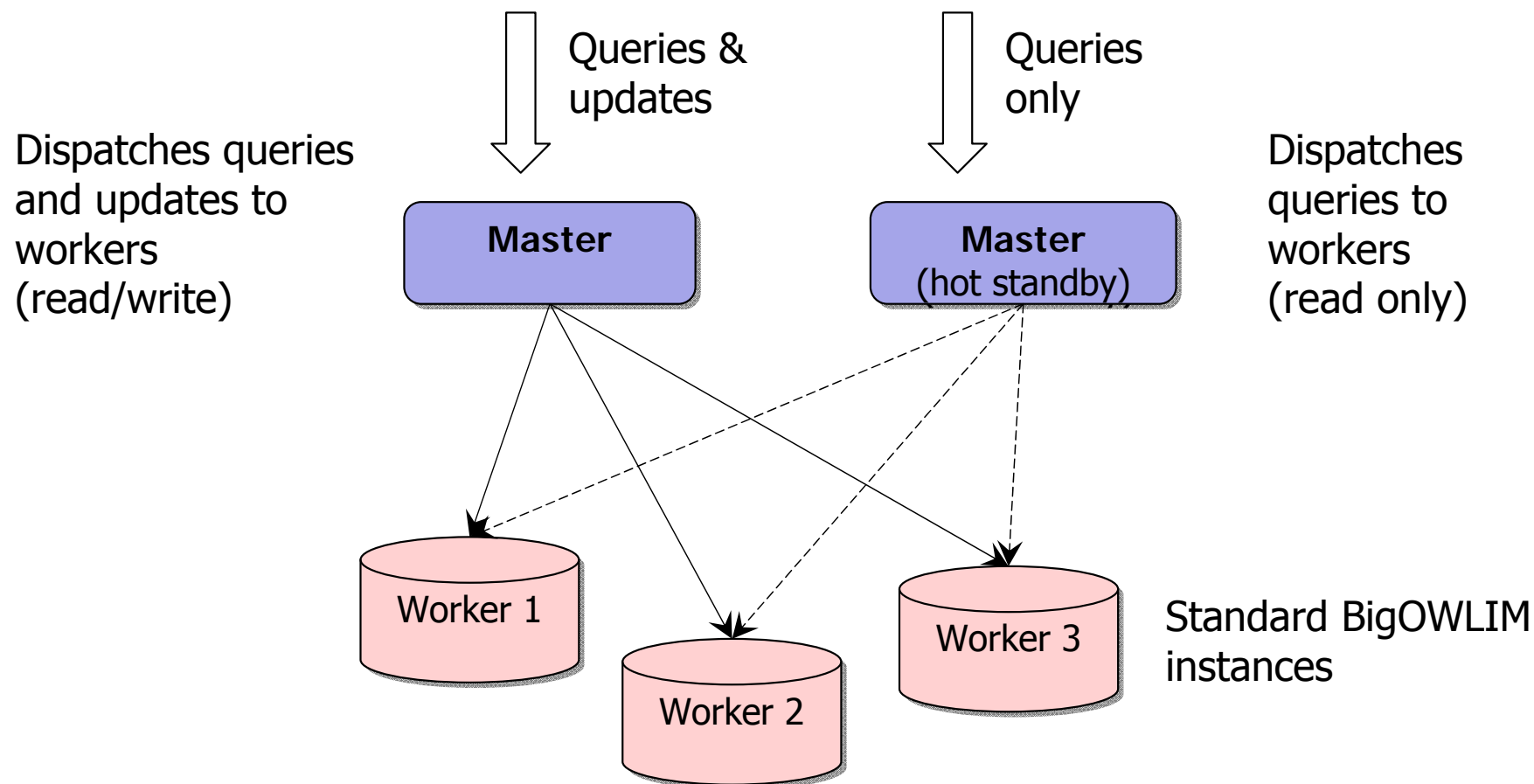


Replication Cluster - Behaviour

- The **query performance** of the cluster represents the **sum of the throughputs** that can be handled by each of the instances
 - **Millions** of read requests per day
 - **Thousands** of updates per hour (with inference!)
- Failover:
 - Failure of a node leads to **graceful performance degradation**
 - **Fully operational** even when there is only one instance working
 - Comprehensive logic guarantees that a problematic update transaction will not affect all nodes of the cluster
- Cluster can be **reconfigured when running**

Replication Cluster - Types of Nodes

- Two types of nodes
- Flexible topologies possible
- Resilience to failure of workers and masters



RDF Rank

- BigOWLIM uses a modification of PageRank over RDF graphs
- The computation of the RDFRank-s for FactForge (several billion statements) takes just a few minutes
- Results are available through a system predicate
- **Example:** get the 100 most “important” nodes in the RDF graph

```
SELECT ?n {?n onto:hasRDFRank ?r}  
ORDER BY DESC(?r) LIMIT 100
```

Full-Text Search

- Full-text search is different from SQL-type queries
 - Queries are formulated and evaluated in a different way
 - Different indices are required for efficient handling
- **Node Search** is one variety of FTS in BigOWLIM
 - URI and literals are retrieved using a set of tokens that should appear in them
 - The matching criteria are determined via system predicates (exact, ignore case, prefix,...)

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX onto: <http://www.ontotext.com/>
SELECT ?x ?label WHERE {
    ?x rdfs:label ?label .
    <3d:> onto:prefixMatchIgnoreCase ?label.
}
```

RDF Search - Advanced FTS in RDF Graphs

- Objective:
 - Be able to search in an RDF graph by keywords
 - Get usable results (standalone literals are not useful in many cases)
- What and how to index:
 - Index URIs
 - Acquire text representation for each URI, by collecting the text from its RDF molecule
 - RDF Molecule: the description of the node, including all outgoing statements
 - Index the text representations with standard FTS methods
- What to return as result:
 - List of URIs, ranked by FTS + RDFRank metric
 - Present them with human-readable labels and text snippets

RDF Search – Advanced FTS in RDF Graphs (2)

- The ranking is based on the standard vector-space-model relevance, boosted by RDFRank
- Sample Query

```
PREFIX gossip: <http://www.....gossipdb.owl#>
PREFIX onto: <http://www.ontotext.com/>
SELECT * WHERE {
    ?person gossip:name ?name .
    ?name onto:luceneQuery "American AND life~" .
}
```

owl:sameAs Optimisation

- **owl:sameAs** declares that two different URIs denote one and the same resource or object in the world
 - Most often, it is used to align different identifiers of the same real-world entity used in different data sources
- Example, encoding that there are three different URIs for Bulgaria and two for Sofia (that is part of Bulgaria)

dbpedia:Sofia owl:sameAs geonames:727011

geonames:727011 geo-ont:parentFeature geonames:732800

dbpedia:Bulgaria owl:sameAs geonames:732800

dbpedia:Bulgaria owl:sameAs [opencyc-en:Bulgaria](#),

owl:sameAs Optimisation (2)

- According to the standard semantics of owl:sameAs
 - It is a transitive and symmetric relationship
 - Statements asserted using one of the equivalent URIs should be inferred to appear with all equivalent URIs placed in the same position
 - Thus the 4 statements in the example lead to 10 inferred statements :

```
geonames:727011 owl:sameAs dbpedia:Sofia
geonames:732800 owl:sameAs dbpedia:Bulgaria
geonames:732800 owl:sameAs opencyc-en:Bulgaria
opencyc-en:Bulgaria owl:sameAs dbpedia:Bulgaria
opencyc-en:Bulgaria owl:sameAs geonames:732800
dbpedia:Sofia geo-ont:parentFeature geonames:732800
dbpedia:Sofia geo-ont:parentFeature opencyc-en:Bulgaria
dbpedia:Sofia geo-ont:parentFeature dbpedia:Bulgaria
geonames:727011 geo-ont:parentFeature opencyc-en:Bulgaria
geonames:727011 geo-ont:parentFeature dbpedia:Bulgaria
```

owl:sameAs Optimisation (3)

- BigOWLIM features an optimisation that allows it to use a single master-node in its indices to represent a class of sameAs-equivalent URIs
- This optimisation:
 - Avoids inflating the indices with multiple equivalent statements
 - Imagine a statement, which has 5 sameAs-equivalents of its object, 2 of its predicate, and 3 of its object. Such statement would have 30 replicas in the indices after forward-chaining if such an optimisation is not used
 - Optionally expands query results
 - The sameAs equivalence can result in multiplication of the bindings of the variables in the process of query evaluation with both forward- and backward-chaining. This leads to expansion of the result-set with rows which differ only by referring to different URIs of one and the same class

owl:sameAs Optimisation (4)

- The owl:sameAs optimisation is carefully designed and implemented to make sure that:
 - All the inferences that follow from the application of the standard owl:sameAs semantics are inferable with the optimisation also
 - One can correctly determine the “original” version of the statement, i.e. which URIs were used when the statement was asserted
 - One can still get all the variations of all statements, if desired
 - the standard semantics can be simulated upon retrieval in a manner which makes the owl:sameAs an implementation detail which is transparent to end users who are not worried about expanded result sets
- Without this optimisation reasoning with linked data becomes inefficient and the query results become overly inflated

OWLIM

<http://www.ontotext.com/owlim>

Based on published results¹ and independent evaluations²:

**OWLIM is the most scalable and the most efficient
semantic repository in the world!**

**It also offers the most comprehensive reasoning support
and the most advanced data management features**

¹ http://www.ontotext.com/owlim/benchmarking_index.html

² <http://www.ontotext.com/owlim/references.html>

Presentation Outline

- **Introducing Ontotext**
 - Why RDF is Good for Data Integration
 - The benefits of light-weight inference
 - Interlinking Text and Data: Hybrid Search
- **What makes the OWLIM semantic repository special**
 - Best Scalability and Query Efficiency
 - Resilient Cluster Setup for Critical Query Loads
 - Optimized for Data Integration
- **Unique Linked Data Management Expertise**
 - Linked Data: Introduction and Challenges
 - FactForge: Fast Track to the Center of the Web of Data

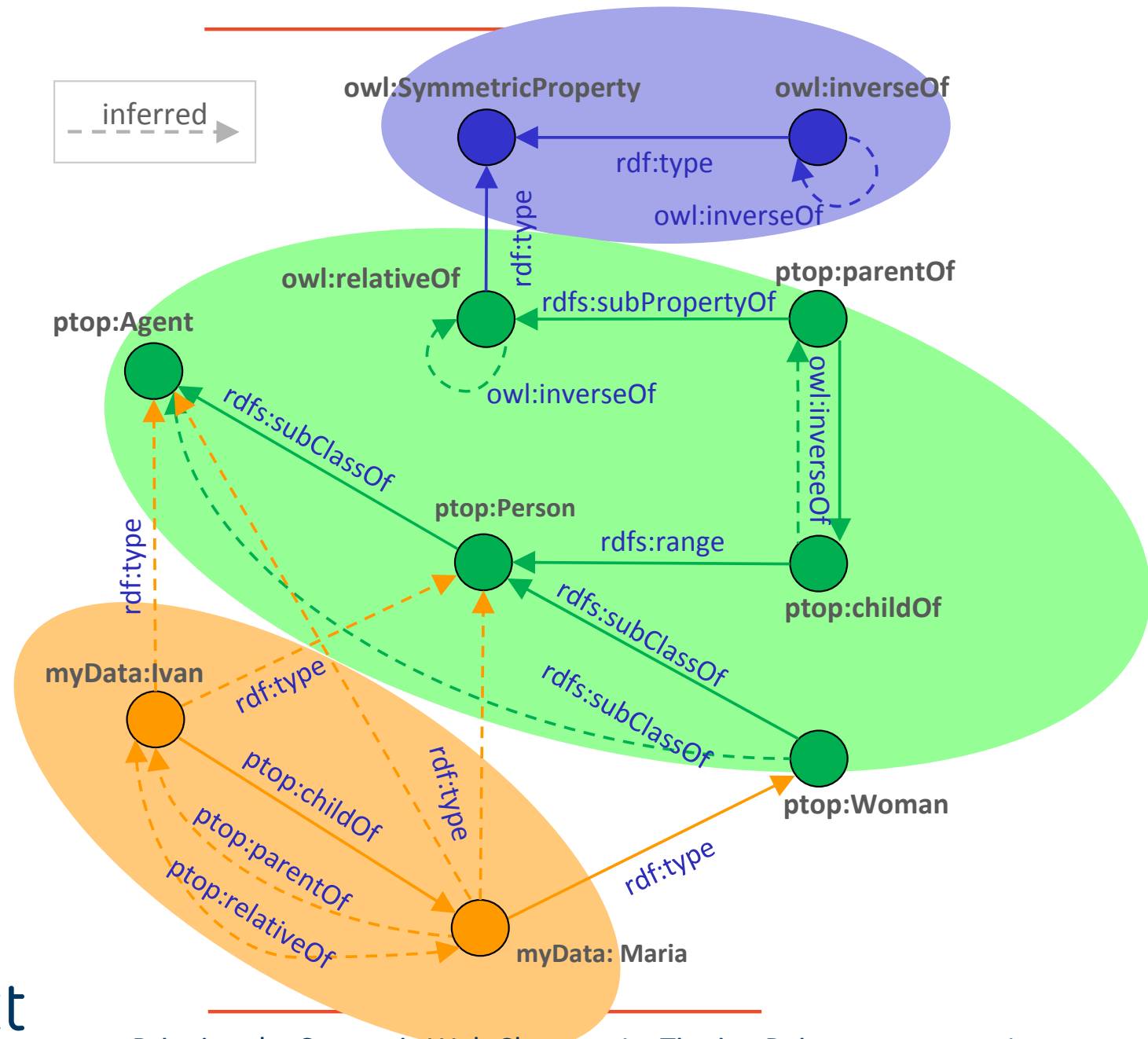
Linked Data

- The notion of “linked data” is defined by Tim Berners-Lee in <http://www.w3.org/DesignIssues/LinkedData.html>
- It outlines an approach for bootstrapping a web of data, a prerequisite for the Semantic Web
- It prescribes that
 - Data should be published on the WWW as RDF graphs
 - the basic Semantic Web representation format
 - In such a way that one can explore them across servers by following the links in the graph
 - in a manner similar to the way the HTML Web is navigated
 - It is viewed as a method for sharing and connecting pieces of data, information, and knowledge on the Semantic Web using URIs and RDF

Linked Data (2)

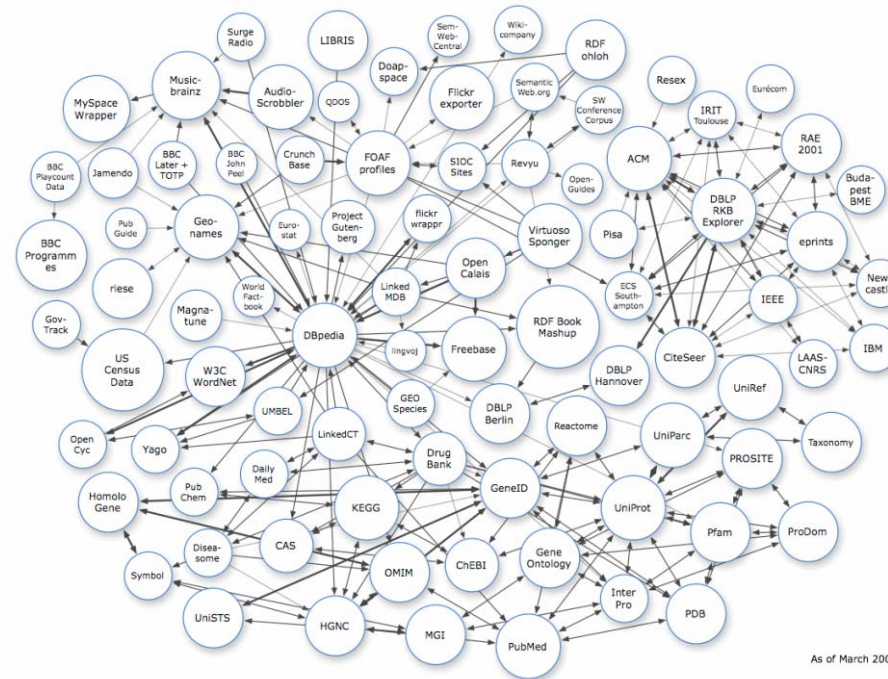
- Technically, “linked data” are constituted by publishing and interlinking open data sources, following 4 principles. These are:
 - Using URIs (globally unique identifiers) as names for things
 - Using HTTP URIs, so that people can look up those names
 - Providing useful information when someone looks up a URI
 - To be concrete, linked data publishers should make sure that HTTP GET with a URI from the RDF graph returns the description of the resource, i.e. the set of statements where it appears as a subject (also known as an RDF molecule)
 - Including links to other URIs from other datasets, so people can discover more things

Linking Data Across Different Servers



Linking Open Data (LOD)

- **Linking Open Data W3C SWEO Community project**
<http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>



- Initiative for publishing “linked data” which already includes 50+ interlinked datasets and about 15B facts

Why Do People *Not* Use Linked Data?

- Plenty of people in the IT world have heard about linked data and like the idea
- However, the impact of linked data on enterprises is still very limited
- Because:
 - There are no well established opinions about what linked data can “buy” for the enterprise or best practices for using it
 - What are the concrete benefits?
 - It is not clear what it would cost
 - What are the problems?
 - What are the associated risks?

Linked Data in the Enterprise: Why?

- **To facilitate data integration**
 - One can use LOD as an “interlingua” for enterprise data integration
 - Additional public information can help alignment and linking
- **To add value to proprietary data**
 - Public data can allow more analytics on top of proprietary data
 - For instance, by linking to spatial data from Geonames
 - **Better description and access to content, e.g. search for images**
- **Make enterprise data more open**
 - To make enterprise data easier to use outside the enterprise
 - Public identifiers and vocabularies can be used to access them

Linked Data in the Enterprise: Challenges

- **LOD is hard to comprehend**
 - Diversity comes at a price
 - One needs to make a query against 200 different schemata and hundreds of thousands of classes and properties
- **Data quality is poor**
 - Many of the datasets are well positioned to be used as “master data” but their quality is very far from enterprise standards
 - No kind of consistency is guaranteed
 - Low commitment to the formal semantics and intended usage of the ontologies and schemata
 - These problems are addressed at <http://pedantic-web.org/>

Linked Data in the Enterprise: Challenges (2)

- **LOD is unreliable**
 - High down times even of the most central “data sites”
 - There is no one to guarantee any service levels
- **Querying linked data is slow**
 - Most of the servers behind LOD today represent experimental, proof-of-concept environments
 - Evaluation of SPARQL queries against them is slow
 - Dealing with data distributed on the web is slow
 - A federated SPARQL query that uses 2-3 servers within several joins can be **very** slow

Reason-able Views to the Web of Data

- *Reason-able views* represent an approach for reasoning and management of linked data
- Key ideas:
 - Group **selected datasets and ontologies** in a compound dataset
 - Clean up, post-process and enrich the datasets if necessary
 - Do this conservatively, in a clearly documented and automated manner, so that:
 - the operation can easily be performed each time a new version of one of the datasets is published
 - Users can easily understand the intervention made to the original dataset
 - Load the compound dataset in a **single semantic repository**
 - Perform inference with respect to **tractable OWL** dialects
 - Define a **set of sample queries** against the compound dataset
 - These determine the “level of service” or the “scope of consistency” contract offered by the reason-able view

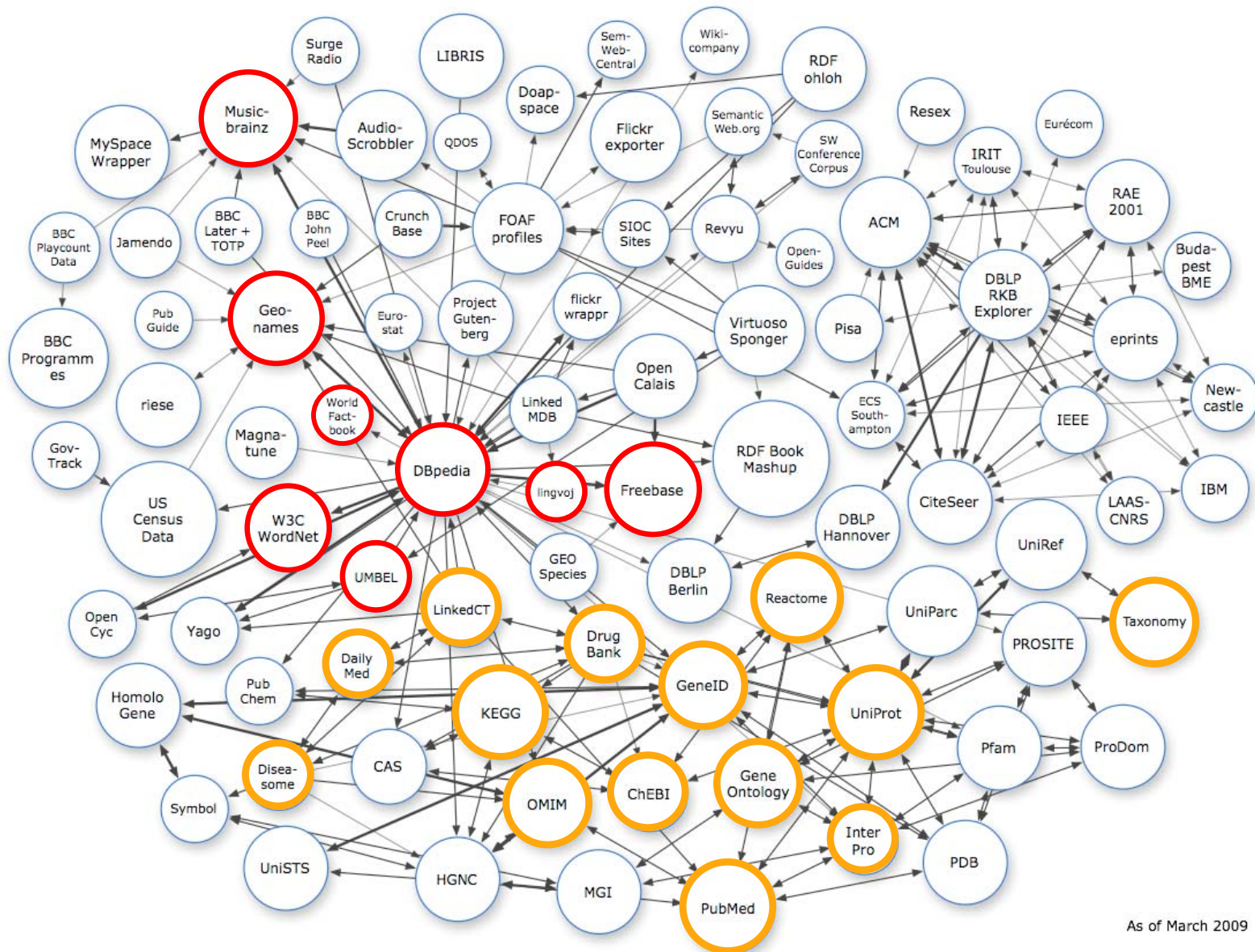
Reason-able Views: Objectives

- **Make reasoning and query evaluation feasible**
- **Guarantee a basic level of consistency**
 - The sample queries guarantee the consistency of the data in the same way in which regression tests do for the quality of software
- **Guarantee availability**
 - In the same way in which web search engines are usually more reliable than most of the web sites; they also do caching
- **Easier exploration and querying of unseen data**
 - Lower the cost of entry through URI auto-completion and *RDF search*
 - Sample queries provide re-usable extraction patterns, which reduce the time for acquaintance with the datasets and their interconnections

Two Reason-able Views to the Web of Linked Data

- **FactForge** (indicated in red on the next slide)
 - Some of the central LOD datasets
 - General-purpose information (not specific to a domain)
 - 1.2B explicit plus 0.9B inferred indexed statements, **10B retrievable**
 - The largest upper-level knowledge base
 - <http://www.factforge.net/>
- **Linked Life Data** (indicated in yellow)
 - 25 of the most popular life-science datasets
 - Complemented by gluing ontologies
 - **2.7B explicit** and 1.4B inferred, total of **4.1B indexed statements**
 - The largest non-synthetic dataset that was used for reasoning
 - <http://www.linkedlifedata.com>

Linking Open Data Datasets and Views (red and yellow)



Presentation Outline

- Introducing Ontotext
 - Why RDF is Good for Data Integration
 - The benefits of light-weight inference
 - Interlinking Text and Data: Hybrid Search
- What makes the OWLIM semantic repository special
 - Best Scalability and Query Efficiency
 - Resilient Cluster Setup for Critical Query Loads
 - Optimized for Data Integration
- Unique Linked Data Management Expertise
 - Linked Data: Introduction and Challenges
 - **FactForge: Fast Track to the Center of the Web of Data**

FactForge: Fast Track to the Center of the Web of Data

- **Datasets:** DBPedia, Freebase, Geonames, UMBEL, MusicBrainz, Wordnet, CIA World Factbook, Lingvoj
- **Ontologies:** Dublin Core, SKOS, RSS, FOAF
- **Inference:** materialization with respect to OWL 2 RL
 - Seems to completely cover the semantics of the data
 - **owl:sameAs optimization** in BigOWLIM allows smaller indices without loss of semantics, but big gains in performance
- Free public service at <http://www.factforge.net>,
 - Incremental **URI auto-suggest**
 - **Query** and **explore** through Forest and Tabulator
 - **RDF Search:** retrieve ranked list of URIs by keywords (see slide 32)
 - **SPARQL end-point**

FactForge Loading and Inference Statistics

| Dataset | Explicit Indexed Triples ('000) | Inferred Indexed Triples ('000) | Total # of Stored Triples ('000) | Entities ('000 of nodes in the graph) | Inferred closure ratio |
|-------------------------|---------------------------------|---------------------------------|----------------------------------|---------------------------------------|------------------------|
| Sechmata and ontologies | 11 | 7 | 18 | 6 | 0.6 |
| DBpedia (categories) | 2,877 | 42,587 | 45,464 | 1,144 | 14.8 |
| DBpedia (sameAs) | 5,544 | 566 | 6,110 | 8,464 | 0.1 |
| UMBEL | 5,162 | 42,212 | 47,374 | 500 | 8.2 |
| Lingvoj | 20 | 863 | 883 | 18 | 43.8 |
| CIA Factbook | 76 | 4 | 80 | 25 | 0.1 |
| Wordnet | 2,281 | 9,296 | 11,577 | 830 | 4.1 |
| Geonames | 91,908 | 125,025 | 216,933 | 33,382 | 1.4 |
| DBpedia core | 560,096 | 198,043 | 758,139 | 127,931 | 0.4 |
| Freebase | 463,689 | 40,840 | 504,529 | 94,810 | 0.1 |
| MusicBrainz | 45,536 | 421,093 | 466,630 | 15,595 | 9.2 |
| Total | 1,177,961 | 881,224 | 2,058,185 | 283,253 | 0.7 |

Post-processing

- Several kinds of post-processing were performed
 - *Goal*: to allow easier navigation and browsing
 - *Mechanisms*: the results are available through system predicates
 - For instance: preferred labels, text snippets and RDF Ranks for all nodes
- Final Statistics
 - Number of entities (RDF graph nodes): 405M
 - Number of inserted statements (NIS): **1.2B**
 - Number of stored statements (NSS): **2.2B**
 - Number of retrievable statements (NRS): **9.8B**
 - 7.6B statements “compressed” through BigOWLIM’s **owl:sameAs** optimisation

FactForge Provides Unique Query Capabilities

- Unmatched **factual knowledge querying capabilities**
 - **Scope**: the largest and most diverse integrated dataset, including 8 of the central LOD datasets
 - **Analytical power**: the semantics of the data and links between them is fully accounted for during query evaluation
 - **RDFRank indicates importance** in the integrated dataset
- No other service supports even one of the following:
 - Query evaluation against DBPedia, Freebase or Geonames considering their semantics
 - Allows simultaneous querying of multiple datasets, considering the **semantics of the links** between them

Guess who is the most popular German entertainer?

```
PREFIX rdf: ...           (run the query at http://factforge.net/sparql)

SELECT * WHERE {
    ?Person dbp-ont:birthPlace ?BirthPlace ;
            rdf:type opencyc:Entertainer ;
            ff:hasRDFRank ?RR .
    ?BirthPlace geo-ont:parentFeature dbpedia:Germany .
} ORDER BY DESC(?RR) LIMIT 100
```

- Without FF, answering such queries in real time is impossible:
- Uses data from: DBPedia, Geonames, UMBEL and MusicBrainz
- Inference over types, sub-classes, and transitive relationships
- The most popular entertainer born in Germany is: **F. Nietzsche**
 - Asking factual questions to a global KB can bring unexpected and strange results
 - We ask who is the most popular person, who qualifies as an entertainer
 - It uses a simple notion of popularity: RDFRank

The Modigliani Test for the Semantic Web

- ReadWriteWeb's founder Richard McManus:
“...**the tipping point for the Semantic Web** may be when one can ... deliver – using Linked Data – a comprehensive list of locations of original Modigliani art works ...”

http://www.readwriteweb.com/archives/the_modigliani_test_for_linked_data.php

The LDSR Query Passing the Modigliani Test

PREFIX fb: ... (check the query at <http://factforge.net/sparql>)

```
SELECT DISTINCT
  ?painting_l ?owner_l ?city_fb_con ?city_db_loc ?city_db_cit
WHERE {
  ?p fb:visual_art.artwork.artist dbpedia:Amedeo_Modigliani ;
    fb:visual_art.artwork.owners [
      fb:visual_art.artwork_owner_relationship.owner ?ow ] ;
    ff:preferredLabel ?painting_l.
  ?ow ff:preferredLabel ?owner_l .
  OPTIONAL { ?ow fb:location.location.containedby [
    ff:preferredLabel ?city_fb_con ] } .
  OPTIONAL { ?ow dbp-prop:location ?loc.
    ?loc rdf:type umbel-sc:City ;
      ff:preferredLabel ?city_db_loc }
  OPTIONAL { ?ow dbp-ont:city [ ff:preferredLabel ?city_db_cit ]
  }
}
```

Thank you!

We develop core semantic technology

Ontotext invested 200 person-years, partnered with 100 leading groups, created some of the most popular tools, and delivered multiple solutions.

We know what works and what doesn't

Ontotext set many benchmarks and advanced the frontiers of semantic databases.

We invented “semantic annotation” – linking text with data

Now we are prepared to

interlink your data, your content, and the web