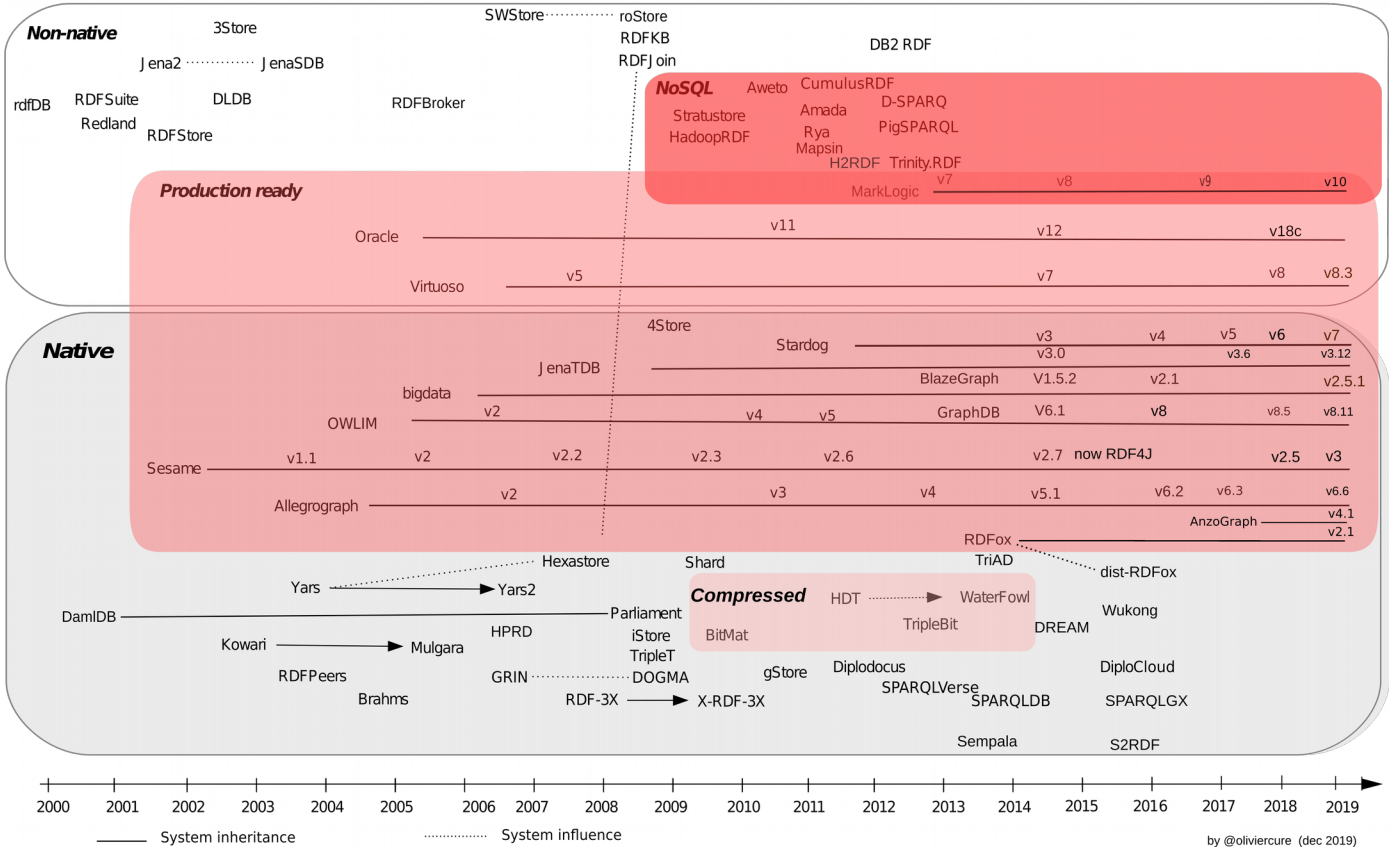
Analytical processing and reasoning in RDF stores

Olivier Curé

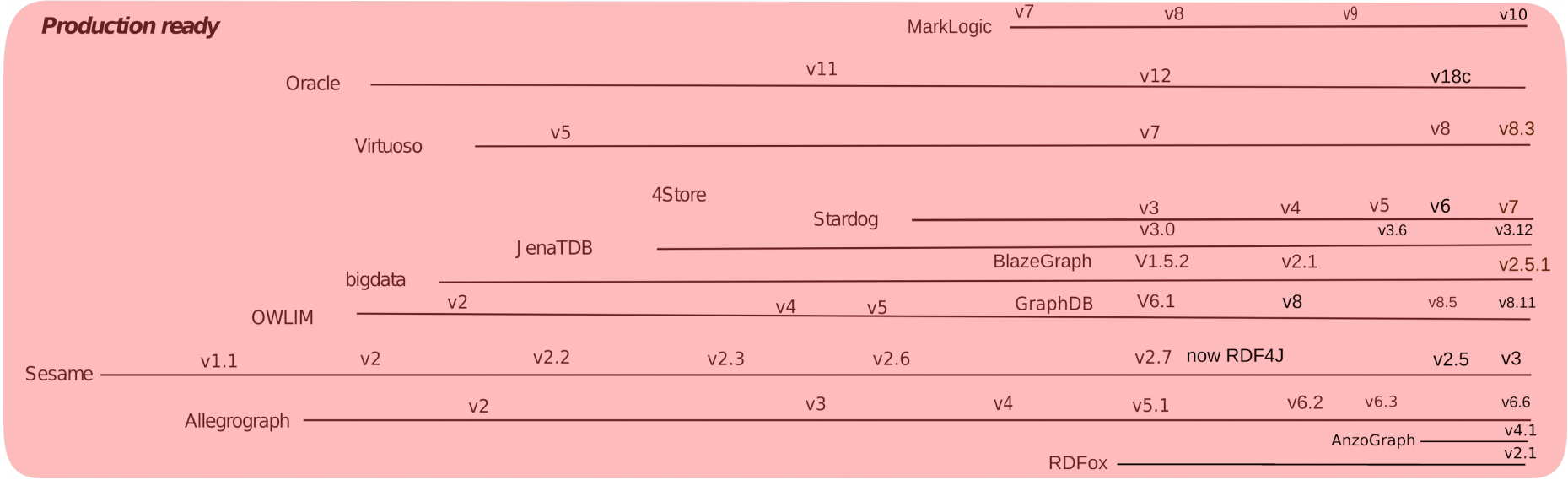
Université Gustave Eiffel

GASPARD MONGE INSTITUT

RDF database systems



RDF production-ready database systems



Common characteristics of these systems

- RDF data models (but property models is also supported)
- Querying with SPARQL 1.1 (but Gremlin is also supported)
- ACID transactions
- Reasoning
- Generally distributed
- Full-text search (using Solr or Elastic search)

Question addressed in this talk

- What is the right market for RDF stores?

Outline

1. Analytics in database systems
2. Reasons why RDF data management is OLAP
3. One system and several research directions
4. Conclusion

1. Analytics in database systems


The Economist
Obama the warrior
Misgoverning Argentina
The economic shift from West to East
Genetically modified crops blossom
The right to eat cats and dogs
FEBRUARY 27TH–MARCH 5TH 2010
Economist.com
The data deluge
AND HOW TO HANDLE IT: A 14-PAGE SPECIAL REPORT

Classification of database applications

- Two kinds of processing :
  - Transactional
    - Ability to collect and manage data in a concurrent and fault tolerant manner
  - Analytical
    - Ability to create information from data
    - Analyze operational data to create reports and support decision making

Classification of database applications

- Two kinds of processing :
  - Transactional -> OnLine Transactional Processing (OLTP)
    - Ability to collect and manage data
  - Analytical -> OnLine Analytical Processing (OLAP)
    - Ability to create information from data
    - Analyze operational data to create reports and support decision making
- Online means end-users expect fast answers to their queries

OLTP vs OLAP

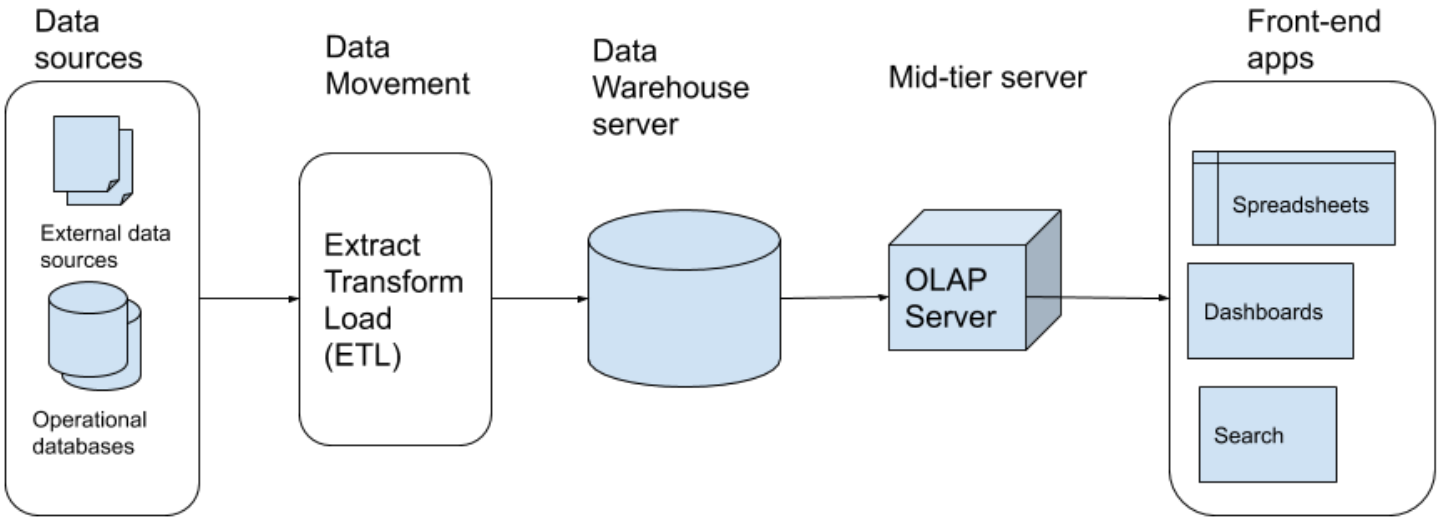
	OLTP	OLAP
Read pattern	Few tuples fetched by a key (index)	Large number of records are accessed
Write pattern	High frequency of small transactions	Low frequency of very large transactions, Bulk import, Data streams
Latency	Low	High
Dataset sizes	GB to few TB	TB to several PB
End-users	Many non specialized	Few specialized (Business analysts, Managers)

Kinds of analytics

- Descriptive analytics: what happened?
  - Did we run out of beer cans in store S last month?
- Diagnostic analytics: why did it happen?
  - Get contextual information that last month was the beginning of the soccer world cup
- Predictive analytics: what will happen?
  - Use external knowledge of coming sport events to know when to stock up beer cans in some stores
- Prescriptive analytics: how can we make it happen?
  - Sponsor sport events to sell more beer cans

Data storage

- “A Data Warehouse stores and manages data. OLAP transforms Data Warehouse data into strategic information” (OLAP Council)



Analytical processing

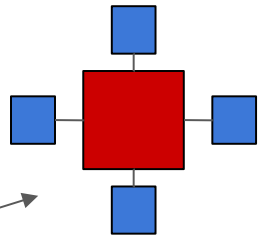
- Performed by OLAP and/or data mining components
- There are several kinds of OLAP systems
  - Multidimensional OLAP (MOLAP)
  - **Relational OLAP (ROLAP)**
  - Hybrid OLAP (HOLAP)
  - And the emerging Graph OLAP (GOLAP)

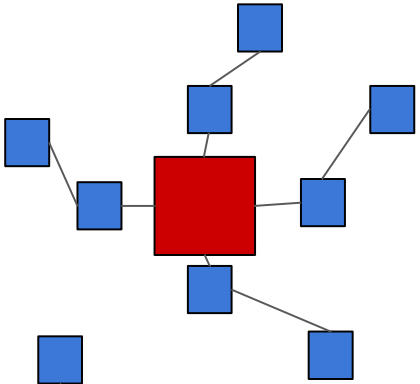
ROLAP - schema

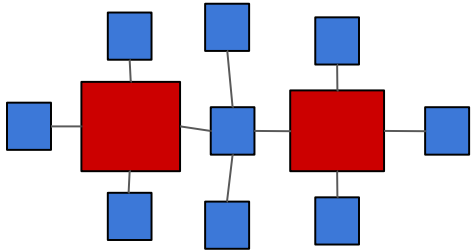
- Distinction between fact (e.g., item sales) and dimension (e.g., store,customer) tables

ROLAP - schema

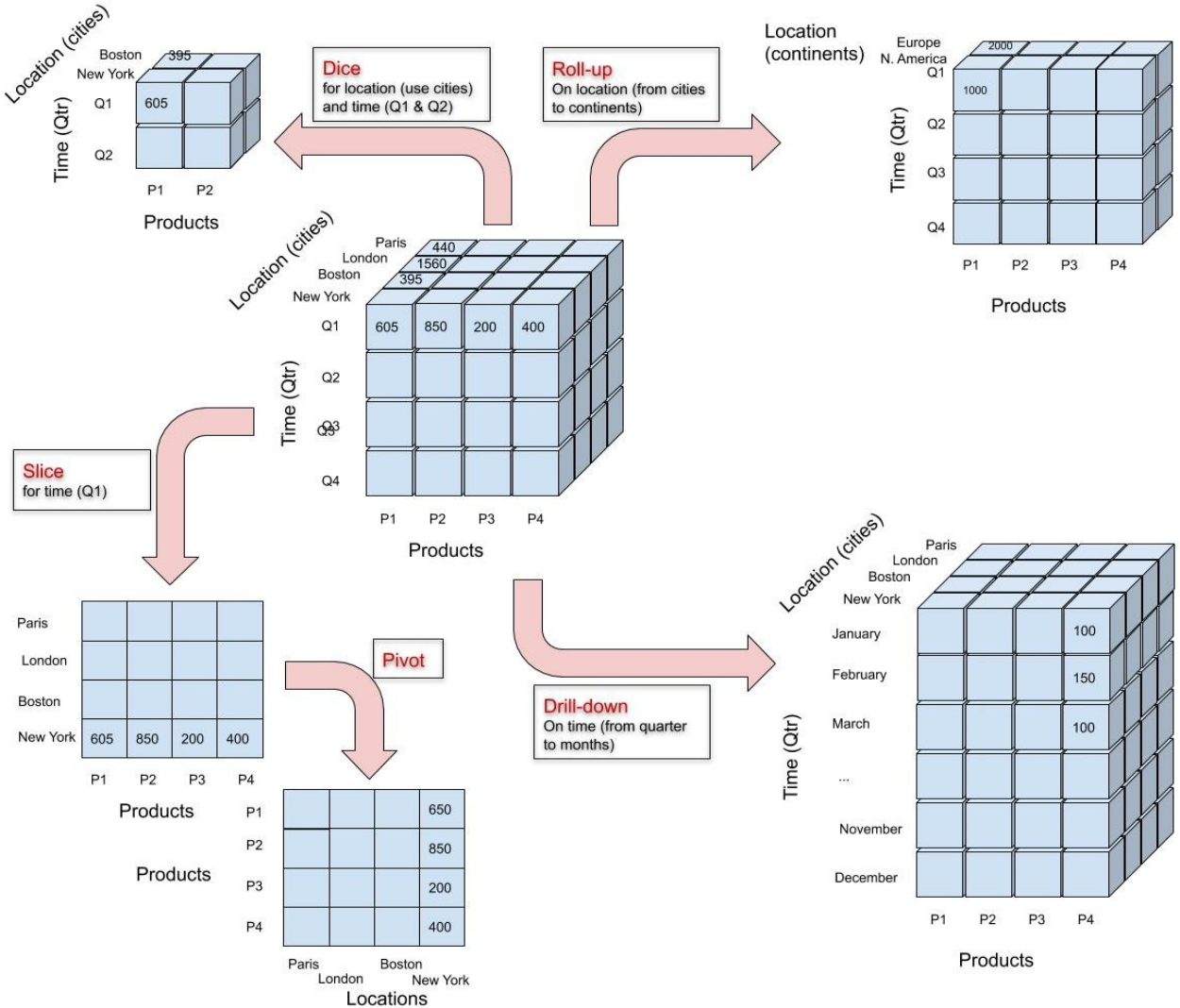
- Distinction between fact (e.g., item sales) and dimension (e.g., store, customer) tables
- Schemata
  - Star schema
  - Snowflake schema (multiple levels of dimension tables)
  - Constellation schema (multiple fact tables)







ROLAP - SQL extensions



OLAP - Views

	Virtual views	**Materialized views**
DB system	OLTP	Data Warehouse
Structure	Logical table	Logical table
Persisted in DBMS	No	Yes
Latency	High	Low
Cost of updates	Low	High

OLAP - Data storage

- Column-oriented storage
  - Stores tuples column-wise (not row-wise)
  - More efficient for select queries retrieving a subset of the tuples of some tuples
  - Limited on writing operations (have to access several files to insert/delete a tuple)

Row Store v. Column Store

Record #	Name	Address	City	State
0003623	ABC	125 N Way	Cityville	PA
0003626	Newburg	1300 Forest Dr.	Troy	VT
0003647	Flotsam	5 Industrial Pkwy	Springfield	MT
0003705	Jolly	529 S 5th St.	Anywhere	NY

Record #	Name	Address	City	State
0003623	ABC	125 N Way	Cityville	PA
0003626	Newburg	1300 Forest Dr	Troy	VT
0003647	Flotsam	Industrial Pkwy	Springfield	MT
0003705	Jolly	529 S 5th St.	Anywhere	NY

OLAP - Data storage

- Compression
  - More possibilities to compress data on columnar storage than row storage
  - Solutions
    - Run-length encoding
    - Delta encoding
    - Bit encoding

Original data

1	M
2	M
3	M
4	F
5	M
6	F
7	M
8	M
9	M

compression

Compressed data (value, offset,length)

1	(M,0,3)
2	(F,3,1)
3	(M,4,1)
4	(F,5,1)
5	(M,6,3)
6	
7	
8	
9	

Original data

Time	Temp
12:00	24.5
12:05	24.6
12:10	24.5
12:15	24.4
12:20	24.3

compression

Compressed data

Time	Temp
12:00	24.5
+5	+0.1
+5	-0.1
+5	-0.1
+5	-0.1

2. Reasons why RDF data management is OLAP

Use cases emphasized by prod-ready systems

- Curating data for advanced analytics
- ERP Integration
- Mainframe migration
- Managing regulated data
- Smart metadata management
- Advanced search and discovery
- Analytics/BI
- Fraud detection
- Recommendations

Use cases emphasized by prod-ready systems

- Curating data for advanced analytics
- ERP Integration
- Mainframe migration
- Managing regulated data
- Smart metadata management
- Advanced search and discovery
- Analytics/BI
- Fraud detection
- Recommendations

Study on large graph processing [1]

- Analytics is the task where end-users are spending the most hours (more than testing, debugging, maintenance, ETL and cleaning).

- Top graph computations performed: finding connected components, neighborhood queries, finding shortest paths, subgraph matching (i.e., SPARQL), ranking and centrality scores, reachability queries.

[1] Siddhartha Sahu et al. The Ubiquity of Large Graphs and Surprising Challenges of Graph Processing. PVLDB 11(4): 420-431 (2017)

Frequency of Linked Data updates

- In practice, Linked Data publishing is periodic bulk loading, e.g., DBPedia, Yago, health science
    - Periodically processing very large transactions makes more sense than supporting (very) frequently small transactions
    - Example: load a new version of DBPedia (billions of assertions every six months)

Features of the SPARQL query language

- SPARQL 1.0 (jan 2008) -> SPARQL 1.1 (march 2013)
    - 5 years to get update operations (INSERT and DELETE) in W3C recommendation !
    - It did not get that long to get update support in SQL, Cypher, Gremlin

Implementation aspect

- RDF Stores are mostly OLTP because it is easier to deal with ACIDity than efficient data partitioning
    - ACID is an engineering/programming challenge
    - Graph partitioning is a “theoretical” (NP-hard) problem

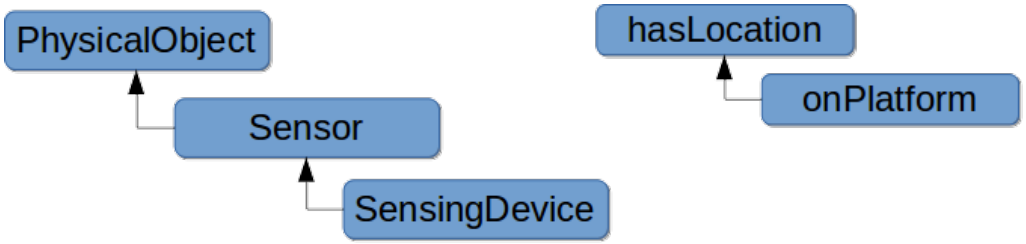
Issues with reasoning

- Two reasoning solutions
    - Materialization
    - Query rewriting

Materialization

- Make explicit all inferences in the store
- Pros:
    - Efficient query processing (no reasoning at query runtime)
- Cons:
    - Slow data loading
    - Data volume expansion
    - Tricky update management (Truth maintenance system)
- aka forward reasoning or saturation or closure

Materialization (2)



- Explicit facts:

```
QBE04 ssn:onPlatform Platform#2
QBE04 rdf:type SensingDevice
```

- The following facts would be added:

```
QBE04 rdf:type Sensor
QBE04 rdf:type PhysicalObject
QBE04 ssn:hasLocation Platform#2
```

Query rewriting

- Reformulate the original query such that all answers can be retrieved
- Pros:
  - No preprocessing overhead
  - No expansion of stored data volume
  - Easy update management
- Cons:
  - Slow query processing due to cost of reasoning at query runtime
- aka Backward reasoning or query reformulation

Query rewriting (2)

- Original query:

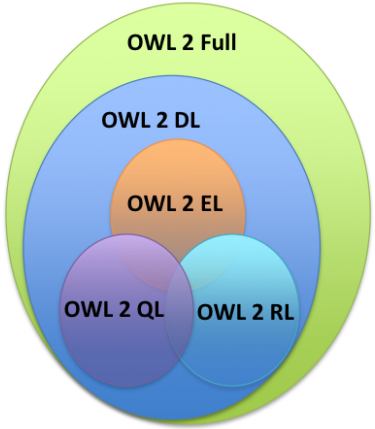
```
SELECT ?x WHERE { ?x rdf:type
PhysicalObject}
```

- Reformulated query:

```
SELECT ?x WHERE { ?x rdf:type
PhysicalObject}
UNION
SELECT ?x WHERE { ?x rdf:type Sensor}
UNION
SELECT ?x WHERE { ?x rdf:type
SensingDevice}
```

Reasoning in prod-ready RDF stores

Triple store	Materialization	Query rewriting
Allegrograph	OWLRL	RDFS++, Prolog
AnzoGraph	RDFS+	
Blazegraph	RDFS, OWL Lite	
GraphDB	RDFS, OWL Horst, OWLRL, OWLQL	
MarkLogic		RDFS, RDFS++, OWL Horst
Oracle	RDFS, OWLRL, OWLQL	
RDFox	OWL2RL	
Stardog	All OWL2	
Virtuoso		RDFS++



Reasoning

- It is hard to efficiently maintain the consistency of an RDF store in the face of of a high transaction rate and support for reasoning.
- With materialization, it may be impossible to support a high rate of transactions
- With query rewriting, query answering will be very slow

3. One system and several research directions

One system - AnzoGraph

- Originally Sparql City, acquired By Cambridge Semantics in 2016
  - Designed by an expert of OLAP systems (Applix, Netezza, Paraccel)
- Part of Anzo smart data lake and also available standalone
- Master-slave, shared-nothing
- RDFS+ inferences
- Rich library for SQL and excel-like functions, C++ API for UDF
- Extends SPARQL 1.1 with
  - OLAP ops : CUBE, ROLLUP, GROUPING SET
  - Named queries, views
- Graph algorithms: centrality, community detection, path finding

AnzoGraph

- Defined as both a Data Warehouse and a GOLAP system

- AnzoGraph can retrieve data from
  - Different external sources including OLTP RDF stores (partnership with MarkLogic
  - Anzo Smart Data Lake
- Commercial with 60 days trial version
- Sufficient to support descriptive and diagnostic analytics
- What about predictive and prescriptive analytics?

An ideal system

- Is distributed, fault tolerant
- Handles both batch and stream processing
- proposes reasoning services
- Performs description, diagnostic, predictive and prescriptive analytics
  - Such a system needs to support knowledge processing
  - Knowledge graph seems to be the ideal representation for this processing

No need to design such a system from scratch

- The Data Warehouse and MapReduce worlds are merging
  - Teradata, Cloudera/Hortonworks, MapR + Arcadia Data
- Frameworks such as Apache Spark or Apache Flink are good candidates to design an OLAP RDF store
  - Libraries available for
    - query processing (SparkSQL / Table API)
    - stream processing (Structure streaming / Datastream API)
    - Graph processing (GraphX / Gelly)
    - Machine learning (MLib / FlinkML)

Spark’s GraphFrame

- Equivalent of Spark’s DataFrame for distributed, fault tolerant graph processing
- Equipped with triplets API and graph pattern matching (Cypher-like)
- Supports view materialization, query optimization (via Catalyst)
- Able to manipulate graph algorithms (pagerank, connected components, triangle count, etc.)
- Easy to design an app that manipulates MLlib, SparkSQL, Structured Streaming
- Limits: no index (brute force), no incremental graph updates

4. Conclusion

- RDF database management is more an OLAP than an OLTP market
- But most RDF Stores are OLTP. Only AnzoGraph can be considered GOLAP
- More research and implementation are needed on issues pertaining to
  - Query processing
    - Integrating SQL-like OLAP operations and graph algorithms in SPARQL
  - Reasoning
  - Allowing prediction

Moving forward

- HTAP : Hybrid Transactional Analytical processing
  - Breaking the wall between OLTP and OLAP
  - Allows advanced analytics on real time transaction data
  - Challenges :
    - Being efficient for both OLTP and OLAP operations
    - Lack of experience on these systems
  - Technical solutions
    - One copy of the data
    - In-memory