ISWC 2017 Tutorial:
Semantic Data Management in Practice

Part 2: Storage and Querying

Olaf Hartig

Linköping University

✉ olaf.hartig@liu.se

@olafhartig

Olivier Curé

University of Paris-Est Marne la Vallée

✉ olivier.cure@u-pem.fr

@oliviercure

LINKÖPING UNIVERSITY

UNIVERSITÉ PARIS-EST MARNE-LA-VALLÉE

Goals

- Achieve an initial understanding of the RDF database management ecosystem

- Understand differences between 7 identified production-ready stores

Overview

- RDF storage
- Seven production-ready RDF stores
- Ontology Based Data Access
- Demo
- APIs

RDF Storage

- Although most production-ready RDF stores support ACID properties, they are best considered as
  - **OLAP** (online analytical processing)
  - **not OLTP** (On line transaction processing)

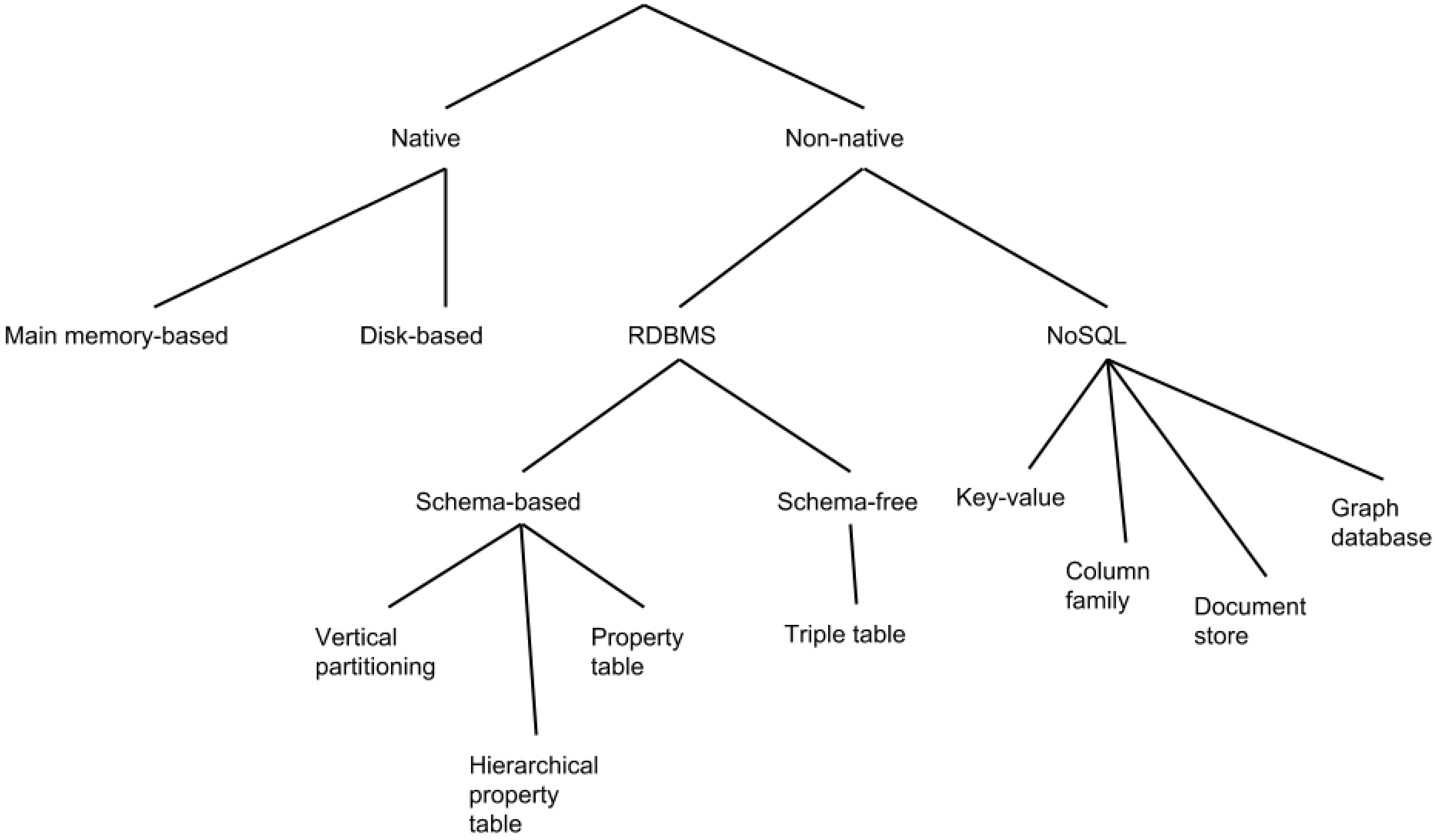
- This implies that updates are performed in batch
  - Mainly due to reasoning (see Section 5)

RDF Storage

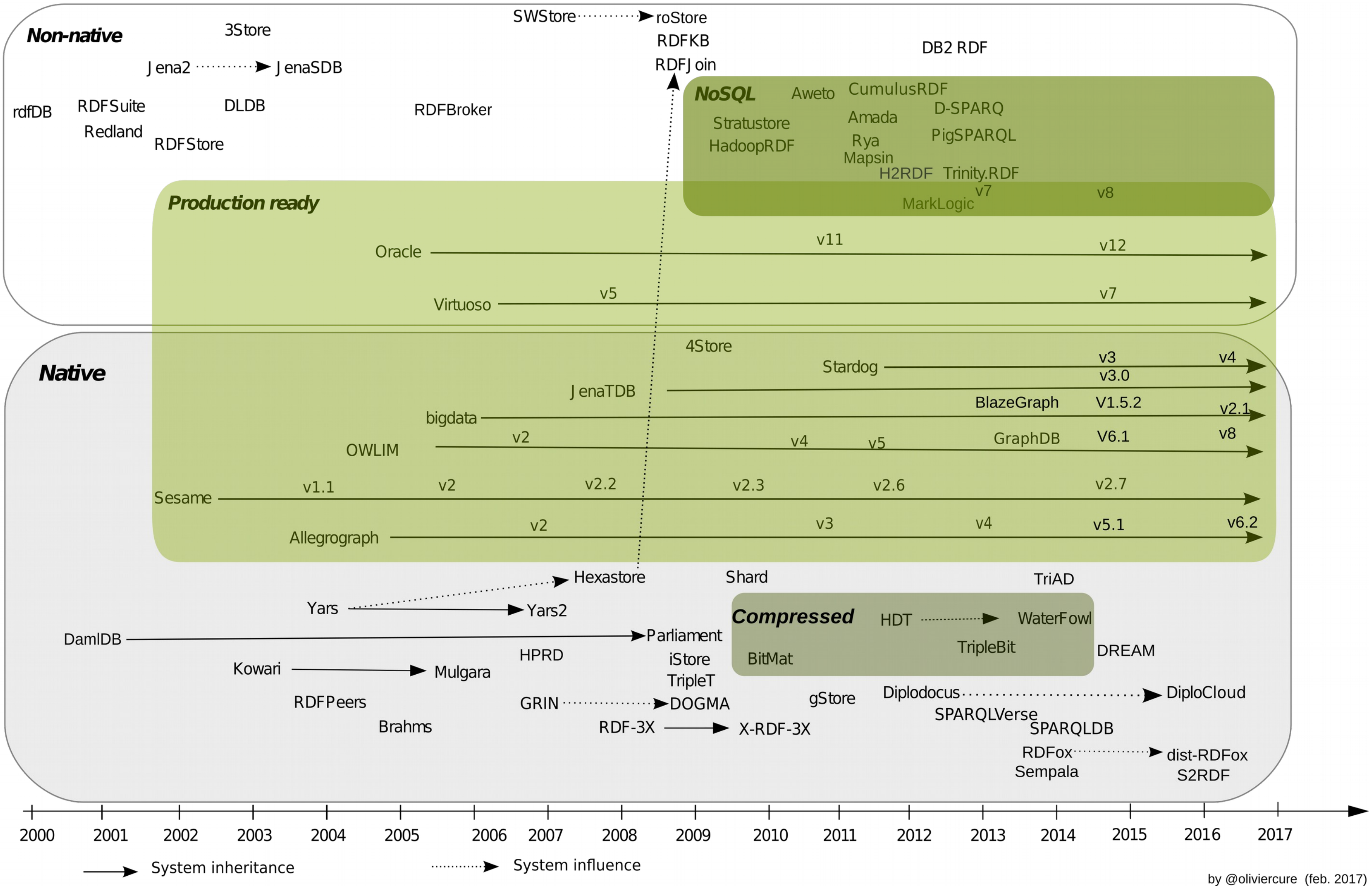
- RDF is a logical data model and thus does not impose any physical storage solution

- Existing RDF stores are either
  - based on an existing DataBase Management System,
    - relational model, e.g., PostgreSQL
    - NoSQL, e.g., Cassandra
  - Designed from scratch, e.g., as a Graph store

RDF Stores Taxonomy



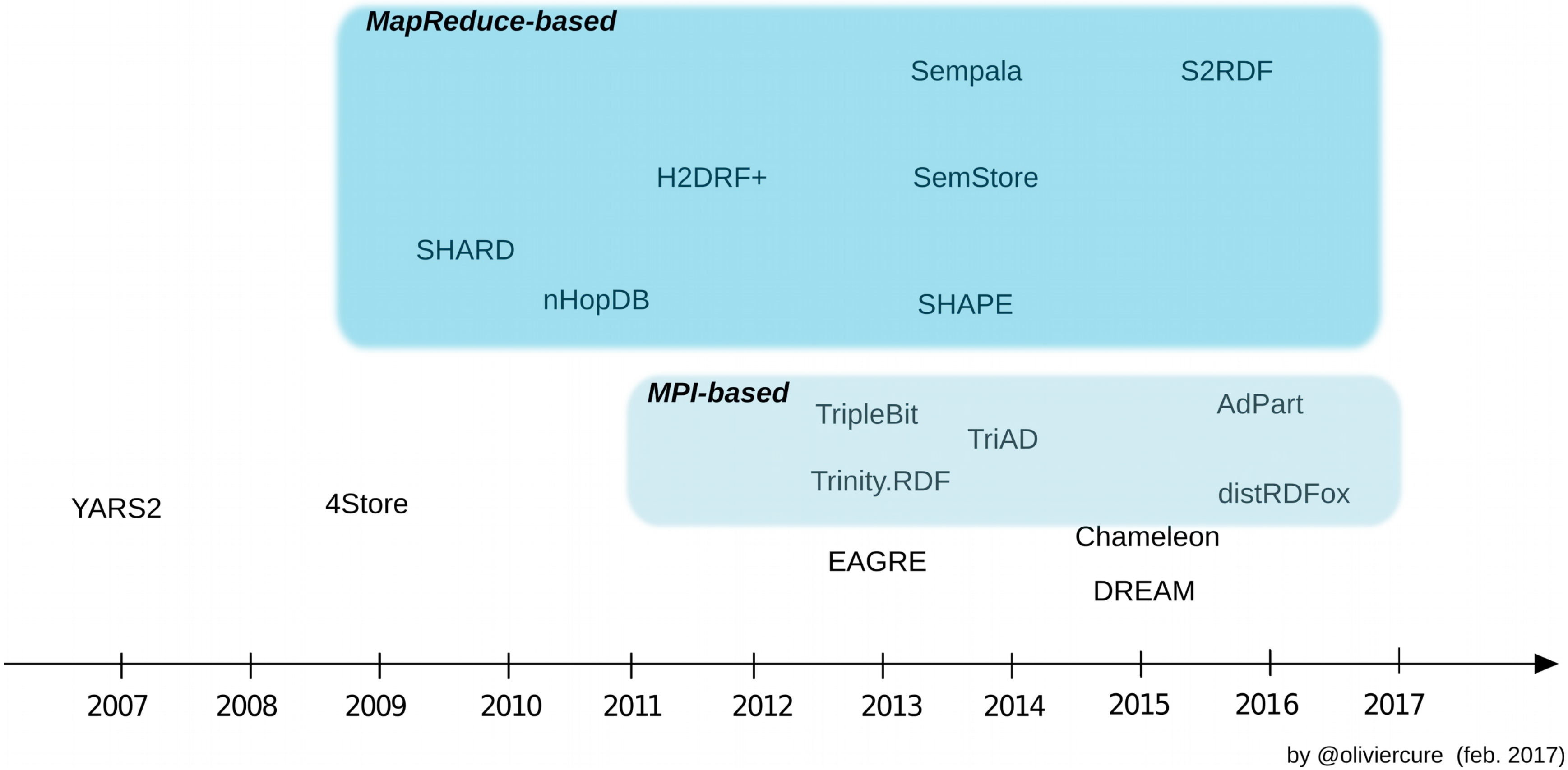
RDF Store Ecosystem



by @oliviercure (feb. 2017)

RDF Distributed data management

- RDF storage is part of Big data
- Distribution of RDF triples over a cluster of machines

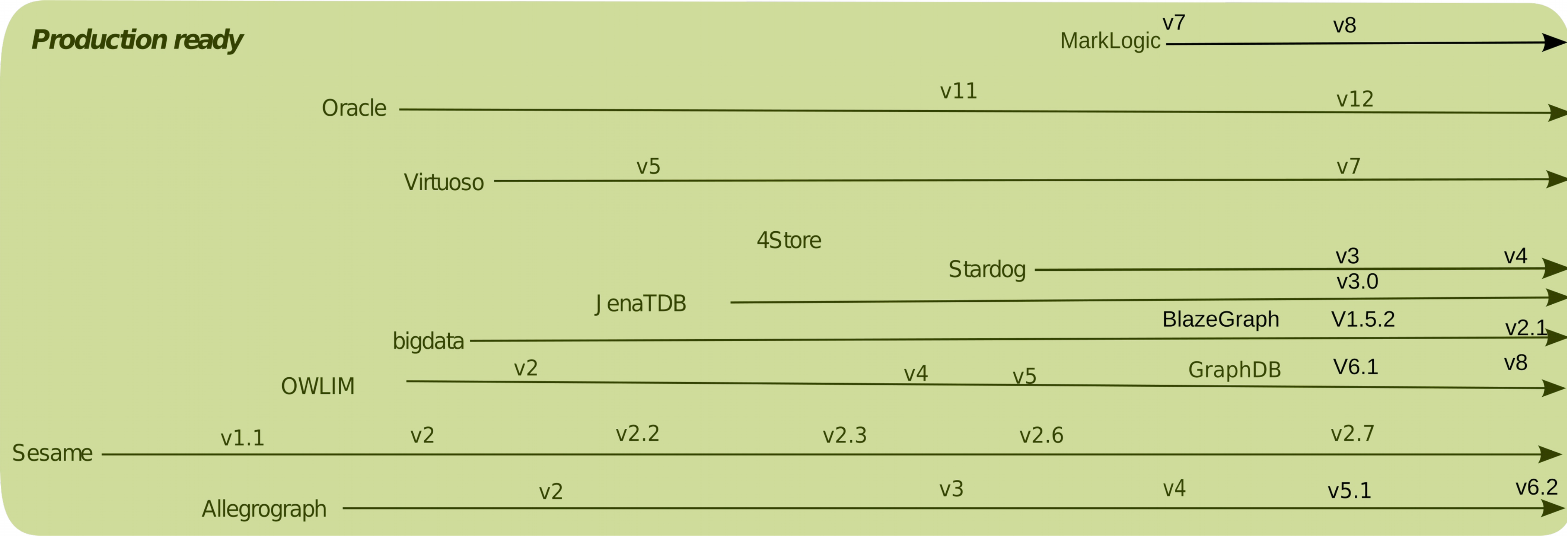


by @oliviercure (feb. 2017)

Overview

- RDF storage
- Seven production-ready RDF stores
- Ontology Based Data Access
- Demo
- APIs

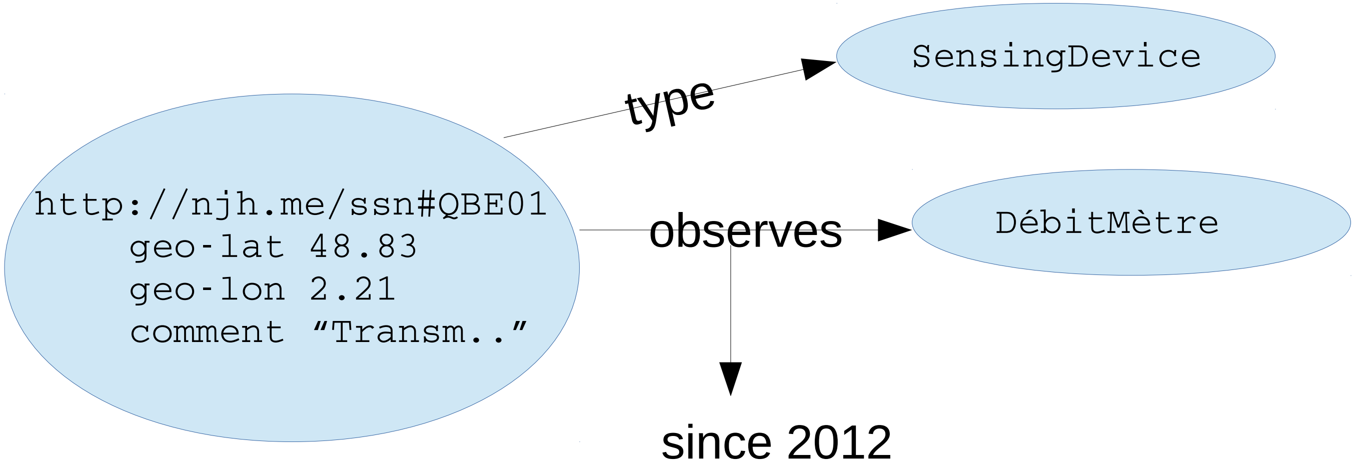
7 Production-Ready Systems



- They all guarantee
  - **ACID** transactions
  - **Replication** (mostly Master-Slave, some Master-Master)
  - **Partition** (Range, Hashing)

Data Models and Querying

- Some of these systems support other data models
  - **XML** for MarkLogic and Virtuoso
  - **Property graph** for GraphDB, BlazeGraph and Stardog



Data Models and Querying

- Some of these systems support other data models
  - **XML** for MarkLogic and Virtuoso
  - **Property graph** for GraphDB, BlazeGraph and Stardog
  - **Relational** for Virtuoso and Oracle
  - **Document** for MarkLogic

- Hence other query languages than SPARQL (v1.1) can be supported
  - **Gremlin** for property graph, **Xquery** for XML, **SQL** for relational, Prolog

License

- Some of these systems have free editions but with some feature or use limitations:
  - MarkLogic’s dev license is free for up to 1TB and 10 months max
  - Stardog: community (10DB max with 25M triples/DB, 4 users), dev (no limits but 30 day trial)
  - Allegrograph: free and dev have restrictions of 5M and 50M respectively
  - Virtuoso and GraphDB: free but no clustering and no replication
  - Blazegraph: free for a single machine
- All systems have commercial editions (Oracle is commercial only)

Summary of production-ready systems

Triple store	Full-text search	Cloud-ready	Extra features
Allegrograph	Integrated + solr	AMI	
Blazegraph	Integrated + solr	AMI	Reification done right
GraphDB	Integrated + solr + elacticsearch (ent.)	AMI	RDF ranking
MarkLogic	Integrated	AMI	With Xquery, Javascript
Oracle	Integrated		Inline in SQL
Stardog	Integrated + Lucene	AMI	Integrity constraints, Explanations
Virtuoso	Integrated	AMI	Inline in SQL

Overview

- RDF storage
- Seven production-ready RDF stores
- Ontology Based Data Access
- Demo
- APIs

OBDA (Ontology Based Data Access) Alternative

- Relevant when you have an existing (relational) database and want to reason over it using an ontology
- The ontology models the domain, hides the structure of the data sources and enriches incomplete data
- The ontology is connected to the data sources via mappings that relate concepts and properties to SQL views over the sources
- Queries, expressed in SPARQL, are translated into the sources query language (usually SQL)
- State of the art is Ontop

Overview

- RDF storage
- Seven production-ready RDF stores
- Ontology Based Data Access
- Demo
- APIs

Demo

- With Blazegraph (v2.1.4)
  - Website: https://www.blazegraph.com/
  - Download: https://sourceforge.net/projects/bigdata/files/bigdata/2.1.4/blazegraph.jar/download
  - Start: java -server -Xmx4g -jar blazegraph.jar
  - http://localhost:9999/blazegraph
- And an extract of our sensor database instantiating the Semantic Sensor Network ontology

Overview

- RDF storage
- Seven production-ready RDF stores
- Ontology Based Data Access
- Demo
- APIs

Available APIs

- Two popular Java APIs to process and handle RDF data and SPARQL queries are:
  - RDF4J (formerly Sesame)
  - Apache Jena
- They both
  - provide a JDBC-like API and REST-like API
  - storing, querying and reasoning capabilities