# Handling RDF data with tools from the Hadoop ecosystem

Paolo Castagna | Solution Architect, Cloudera

7 November 2012 - Rhein-Neckar-Arena, Sinsheim, Germany

## How to process RDF at scale?

Use MapReduce and other tools from the Hadoop ecosystem!

## Use N-Triples or N-Quads serialization formats

- One triple|quad per line
- Use MapReduce to sort|group triples|quads by graph|subject
- Write your own `NQuads{Input|Output}Format` and `QuadRecord{Reader|Writer}`
- Parsing one line at the time not ideal, but robust to syntax errors (see also: `NLineInputFormat`)

NQuadsInputFormat.java, NQuadsOutputFormat.java, QuadRecordReader.java, QuadRecordWriter.java and QuadWritable.java

## N-Triples Example

```
<http://example.org/alice>  <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xmlns.com/foaf/0.1/Person>  .
<http://example.org/alice>  <http://xmlns.com/foaf/0.1/name>  "Alice"  .
<http://example.org/alice>  <http://xmlns.com/foaf/0.1/mbox>  <mailto:alice@example.org>  .
<http://example.org/alice>  <http://xmlns.com/foaf/0.1/knows>  <http://example.org/bob>  .
<http://example.org/alice>  <http://xmlns.com/foaf/0.1/knows>  <http://example.org/charlie>  .
<http://example.org/alice>  <http://xmlns.com/foaf/0.1/knows>  <http://example.org/snoopy>  .
<http://example.org/bob>  <http://xmlns.com/foaf/0.1/name>  "Bob"  .
<http://example.org/bob>  <http://xmlns.com/foaf/0.1/knows>  <http://example.org/charlie>  .
<http://example.org/charlie>  <http://xmlns.com/foaf/0.1/name>  "Charlie"  .
<http://example.org/charlie>  <http://xmlns.com/foaf/0.1/knows>  <http://example.org/alice>  .
```

# Turtle Example

```
@prefix :        <http://example.org/>  .
@prefix foaf:    <http://xmlns.com/foaf/0.1/>  .

:alice
     a           foaf:Person ;
     foaf:name   "Alice" ;
     foaf:mbox   <mailto:alice@example.org>  ;
     foaf:knows  :bob ;
     foaf:knows  :charlie ;
     foaf:knows  :snoopy ;
     .

:bob
     foaf:name   "Bob" ;
     foaf:knows  :charlie ;
     .

:charlie
     foaf:name   "Charlie" ;
     foaf:knows  :alice ;
```

## RDF/XML Example

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns="http://example.org/" >
  <rdf:Description rdf:about="http://example.org/alice">
    <foaf:knows rdf:resource="http://example.org/snoopy"/>
    <foaf:knows rdf:resource="http://example.org/charlie"/>
    <foaf:knows rdf:resource="http://example.org/bob"/>
    <foaf:mbox rdf:resource="mailto:alice@example.org"/>
    <foaf:name>Alice</foaf:name>
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://example.org/bob">
    <foaf:knows rdf:resource="http://example.org/charlie"/>
    <foaf:name>Bob</foaf:name>
  </rdf:Description>
  <rdf:Description rdf:about="http://example.org/charlie">
    <foaf:knows rdf:resource="http://example.org/alice"/>
    <foaf:name>Charlie</foaf:name>
  </rdf:Description>
</rdf:RDF>
```

# RDF/JSON Example

```
{
  "http://example.org/charlie" : {
    "http://xmlns.com/foaf/0.1/name" : [ {
      "type" : "literal" ,
      "value" : "Charlie"
    }
     ] ,
    "http://xmlns.com/foaf/0.1/knows" : [ {
      "type" : "uri" ,
      "value" : "http://example.org/alice"
    }
     ]
  }
   ,
  "http://example.org/alice" : {
    "http://xmlns.com/foaf/0.1/mbox" : [ {
      "type" : "uri" ,
      "value" : "mailto:alice@example.org"
    }
     ] ,
    "http://xmlns.com/foaf/0.1/name" : [ {
      "type" : "literal" ,
      "value" : "Alice"
    }
     ] ,
    "http://www.w3.org/1999/02/22-rdf-syntax-ns#type" : [ {
      "type" : "uri" ,
      "value" : "http://xmlns.com/foaf/0.1/Person"
  ...
```

## Convert RDF/XML, Turtle, etc. to N-Triples

- RDF/XML or Turtle cannot be easily splitted
- Use `WholeFileInputFormat` from the "*Hadoop: The Definitive Guide*" book to convert one file at the time
- Many small files can be combined using `CombineFileInputFormat`, however in case of RDF/XML or Turtle things get complicated

## Validate your RDF data

- Validate each triple|quad separately
- Log a warning with line or offset in bytes of any syntax error, but continue processing
- Write a separate report on bad data: so problems with data can be fixed in one pass
- This can be done with a simple MapReduce job using N-Triples|N-Quads files

## Counting and stats

- MapReduce is a good for counting or computing simple stats
  - How properties and classes are actually used?
  - How many instances of each class?
  - How often some data is repeated across datasets?
  - …

StatsDriver.java

## Turtle and adjacency lists

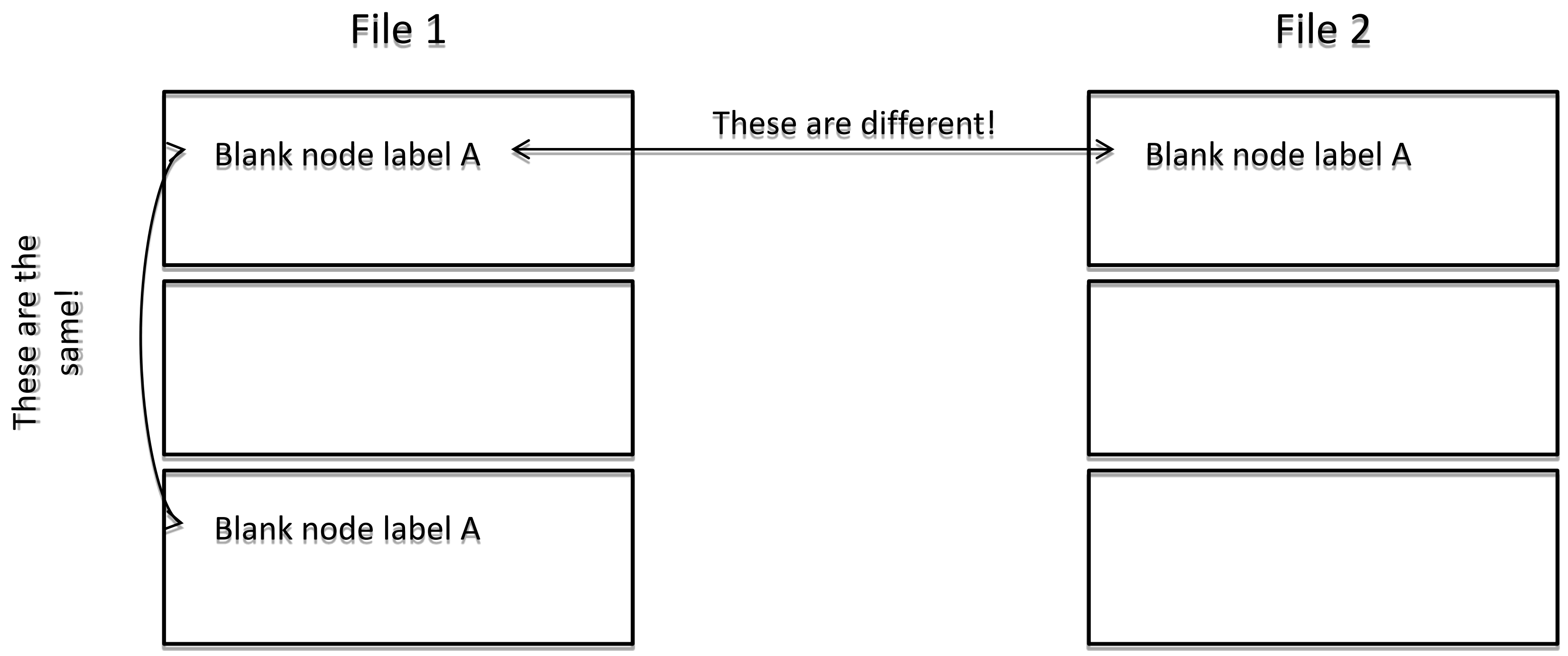
```
<http://example.org/alice> <http://xmlns.com/foaf/0.1/mbox>
<mailto:alice@example.org>; <http://xmlns.com/foaf/0.1/name> "Alice";
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xmlns.com/foaf/0.1/Person>; <http://xmlns.com/foaf/0.1/knows>
<http://example.org/charlie>, <http://example.org/bob>,
<http://example.org/snoopy>; . <http://example.org/charlie>
<http://xmlns.com/foaf/0.1/knows> <http://example.org/alice> .

<http://example.org/bob> <http://xmlns.com/foaf/0.1/name> "Bob";
<http://xmlns.com/foaf/0.1/knows> <http://example.org/charlie>; .
<http://example.org/alice> <http://xmlns.com/foaf/0.1/knows>
<http://example.org/bob> .

<http://example.org/charlie> <http://xmlns.com/foaf/0.1/name> "Charlie";
<http://xmlns.com/foaf/0.1/knows> <http://example.org/alice>; .
<http://example.org/bob> <http://xmlns.com/foaf/0.1/knows>
<http://example.org/charlie> . <http://example.org/alice>
<http://xmlns.com/foaf/0.1/knows> <http://example.org/charlie> .
```

## Apache Giraph

- Subset of your RDF data as adjacency lists (eventually, using Turtle syntax)
- Apache Giraph is a good solution gor graph or iterative algorithms: shortest paths, PageRank, etc.

https://github.com/castagna/jena-grande/tree/master/src/main/java/org/apache/jena/grande/giraph

# Blank nodes


File 1
File 2
These are different!
Blank node label A
Blank node label A
Blank node label A
These are the same!

## Blank nodes

```
public MapReduceAllocator (JobContext context, Path path) {
    this.runId =
context.getConfiguration().get(Constants.RUN_ID);
    if ( this.runId == null ) {
        this.runId = String.valueOf(System.currentTimeMillis());
    }
    this.path = path;
}

@Override
public Node create(String label) {
    String strLabel = "mrbnode_" + runId.hashCode() + "_" +
path.hashCode() + "_" + label;
    return Node.createAnon(new AnonId(strLabel)) ;
}
```

MapReduceLabelToNode.java

# Inference

- For RDF Schema and subsets of OWL, inference can be implemented with MapReduce:
  - use `DistributedCache` for vocabularies or ontologies
  - perform inference “as usual” in the map function
- WARNING: this does not work in general
- For RDFS and OWL ter Horst rule sets:
  - Urbani J., Kotoulas, S., ...
    *“WebPIE: a Web-scale Parallel Inference Engine”*
    Submission to the SCALE competition at CCGrid 2010

InferDriver.java

## Apache Pig

- If you use Pig with Pig Latin scripts, write Pig input/output formats for N-Quads
- PigSPARQL, an interesting research effort:
  - Alexander Schätzle, Martin Przyjaciel-Zablocki, ...
    *"PigSPARQL: Mapping SPARQL to Pig Latin"*
    3th International Workshop on Semantic Web Information Management

NQuadsPigInputFormat.java

## Storing RDF into HBase

- How to store RDF in HBase?
- An attempt inspired by Jena SDB (RDF over RDBMS systems):
  - V. Khadilkar, M. Kantarcioglu, ...
    *"Jena-HBase: A Distributed, Scalable and Efficient RDF Triple Store"*
    University of Texas at Dallas - Technical report (2012)
- Lessons learned:
  - storing is "easy", quering is "hard"
  - Linked Data access pattern: all triples for a given subject

https://github.com/castagna/hbase-rdf

## Building (B+Tree) indexes with MapReduce

- tdbloader4 is a sequence of four MapReduce jobs:
  - compute offsets for node ids
  - 2 jobs for dictionary encoding (i.e. URL → node ids)
  - sort and build the 9 B+Tree indexes for TDB

https://github.com/castagna/tdbloader4

## Jena Grande

https://github.com/castagna/jena-grande

Fork me on GitHub

- Apache Jena is a Java library to parse, store and query RDF data
- Jena Grande is a collection of utilities, experiments and examples on how to use MapReduce, Pig, HBase or Giraph to process data in RDF format
- Experimental and work in progress

# Other Apache projects

- Apache Jena – http://jena.apache.org/
- Apache Any23 – http://any23.apache.org/
  - a module for Behemoth[1]?
- Apache Stanbol – http://stanbol.apache.org/
- Apache Clerezza – http://incubator.apache.org/clerezza/
- Apache Tika – http://tika.apache.org/
  - an RDF plug-in for Tika? Or, Any23 should be that?
- Apache Nutch – http://nutch.apache.org/
  - a plug-in for Nutch (or leverage Behemoth) which uses Any23 to get RDF datasets from the Web?
- …

1 https://github.com/digitalpebble/behemoth

cloudera®

Ask Bigger Questions