Scalable reasoning on RDF documents with Hadoop and HBase

AKA how to make scale http://sindice.com indexing pipeline

Michele Catasta
Google Indexing pipeline
Google Indexing pipeline
(c) Michael Kleber@GOOG
Why leave information implicit?

RDFS and OWL inference

ABox and TBox - STATEFUL algorithms

in-memory model

SPOF - deployment hell
foaf:Person rdfs:subClassOf dbpedia-owl:Park
HBase as the ontology network storage

simple key-value model

data **denormalization**

Pareto principle - 80/20

with proper caching, it does scale

simplicity/optimality tradeoff
Work in progress: a Subject-Context index on HBase

saves computation at indexing time

no denormalization involved

complex update logic

ColumnValueFilter vs Secondary indexes
Credits:
- Renaud Delbru
- Robert Fuller
Q&A
DI2 (Data Intensive Infrastructure) - DERI
http://di2.deri.ie/
michele.catasta@deri.org
pirroh@freenode (#hadoop #hbase)