The BEANS software for fast and easy data analysis

Arkadiusz Hypki

Nicolaus Copernicus Astronomical Center, Warsaw, Poland

ahypki@camk.edu.pl

Astroinformatics 2013, Sydney

Outline

- The BEANS software
- Underlying technologies
- Pig Latin with examples

The BEANS software

Data analysis on Apache Hadoop

The BEANS software - motivation

- tool for **storing the data** from hundreds of MOCCA simulations
  - each simulation has ≈10 files, and ≈10 GBs
- easy tool to **managing the data** the data from different simulations
  - comparing, extracting, filtering, grouping...

The MOCCA code

- one of the most advanced codes for simulations of real-size star clusters
- based on Monte Carlo method (few simplifications in comparison to N-body codes)
- very fast
- agrees very well with N-body codes
- provides as much details about stars as N-body codes
- allows to test whole range of possible initial conditions (beyond capabilities of any N-body code currently)
- `http://www.moccacode.net/`

Underlaying technologies

Apache Cassandra + Apache Hadoop, Elastic Search, D3...

Underlying technologies - Apache Cassandra

Figure: `http://cassandra.apache.org/`

- NoSQL database
- it’s not network file system
- decentralized
- replicated
- scales linearly
- fault-tolerant
- tunable consistency
- integrated with MapReduce
- Cassandra users: Netflix, eBay, Twitter, Reddit, Cisco, OpenX, Digg, CloudKick....
- largest known Cassandra cluster has over 300 TB of data in over 400 machines

Underlying technologies - Apache Hadoop

Figure: Apache Hadoop Logo

- **Google** came up with the concept – used to reindex the web
- adopted instantly in the OpenSource community (**Apache Hadoop + HDFS**)
- Facebook instance: 21 PB of storage in a single HDFS cluster, 2000 machines

Underlying technologies - Apache Hadoop

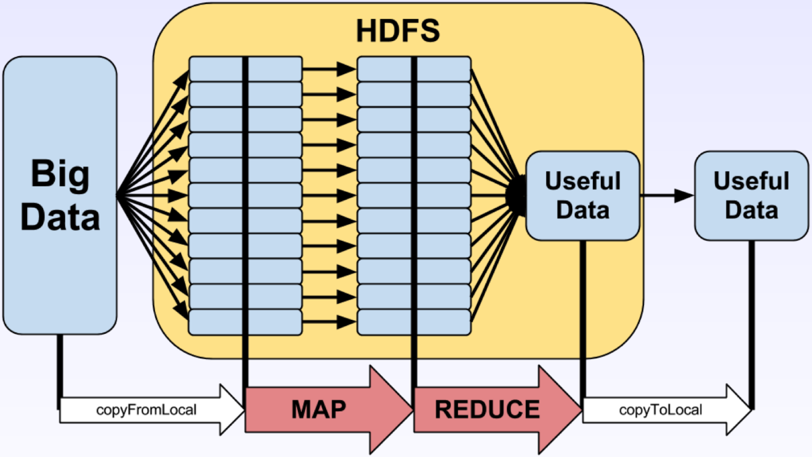

Figure: MapReduce: split/map and reduce; Perfect for embarrassingly easy parallel problems; Linear scalability; Works on commodity hardware

Underlying technologies – ElasticSearch, D3...

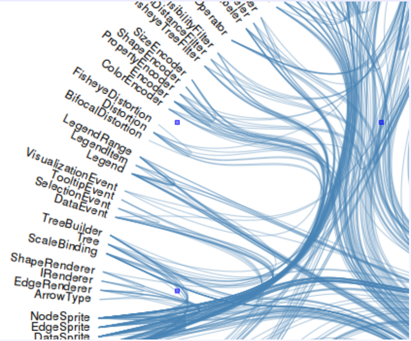

Figure: D3 example plot: Hierarchical Edge Bundling

- Elasticsearch: powerful open source search and analytics engine (`http://www.elasticsearch.org/`)
- D3: JavaScript library for making interactive, clean and powerful plots (`http://d3js.org/`)

Pig Latin

High level scripting language for Apache Hadoop

Pig Latin - Example 1. Parallel coordinates

```
rows = LOAD 'Harris catalogue/Harris3' USING Table();

STORE rows INTO 'plot1/TYPE parallel COLUMNS vr:vLSR:
c:rc:rh:muV:MV:th:rho0:lgTc:lgTh:massGn: bimod' USING
Plot();
```

Pig Latin - Example 2. Lines plot

```
r100 = LOAD ’MOCCA 600k rbar100/system/tphys,smt’ USING
Table();
r55 = LOAD ’MOCCA 600k rbar55/system/tphys,smt’ USING
Table();

X = UNION r100, r55;

STORE X INTO ’plot1/TYPE points COLUMNS tphys:smt TITLE
”Mass of the clusters” COLOR BY tbid’ USING Plot();
```

Pig Latin - Example 3. Filtering, grouping...

```
snap = LOAD ’600k snapshot/snapshot’ USING Table();

bss = FILTER snap BY type1.value == 10;

bssBinned = FOREACH bss GENERATE *, histogram(0.0, 2.0,
0.1, m1.value) as bin;

bssGr = GROUP bssBinned BY (time, bin);

bssGrCount = FOREACH bssGr GENERATE (’time’,
$0.time.value), (’bin’, $0.bin), (’count’, COUNT($1));


STORE bssGrCount INTO ’plot1/SPLIT BY time TYPE boxes
0.1 XRANGE 0; 10 COLUMNS bin:count TITLE ”bin vs. count”
’ USING Plot();
```

The BEANS software – Features

- OpenSource
- server + thin clients (laptop, desktop, phone, tablet, fridge... yes, fridge)
- UDFs defined in java, python, pearl, javascript, jython, ruby, groovy...
- piggy bank
- easy installation
- if underlying data changes → plots change
- sharing notebooks with URLs
- queuing jobs
- ....

The BEANS software – Remarks/limitations

- you can do the same with bash/python scripts + gnuplot but... BEANS simplifies complex queries
- not really tested on $\approx$100 TBs of data but... should work ($\approx$PBs don't know)
- web browsers cannot plot millions of points
- it's not a visualization software

The BEANS software – Best workflow

Best BEANS workflow:

1. ask question
2. write script(s)
3. examine plots
4. did you find the answer? If no, go to 1.

Thank you!