BEANS – Interactive, Distributed Data Analysis of Huge Data Sets

Arkadiusz Hypki
2018.06.26

Astronomical Observatory Institute, Adam Mickiewicz University
BEANS is a web-based software for interactive distributed data analysis with a clear interface for querying, filtering, aggregating, and plotting data from an arbitrary number of datasets and tables.
Rysunek 1: BEANS main window with example Notebook
Why a new tool?

- limited choice of software for complex data analysis with gentle learning curve
  - especially for people already familiar with some bash scripting (e.g. AWK)
- to manage a huge number of scripts
- to manage a large number of datasets
  - e.g. >3k MOCCA models, ∼50 TBs of data in over 100k data files, billions(!) of rows (more models are coming!)
BEANS features

- central repository for storing huge amount of data
- platform to manage, filter and aggregate the data
- console and web interface
- written in a general form
  - it can be used in almost any field of research, or other open source projects
- data can be indexed(!)
  - rows can be found in < 1s
BEANS features

- data analysis in the form of notebooks
- distributed components: Apache Cassandra – database; Elastic – search engine; Apache Pig – high-level language for data analysis based on Apache Hadoop
  - all ready to handle > PBs of data
- interactive plots based on d3.js
- living notebooks: BEANS detects if underlying data has changed, if true, all other entries in notebooks are automatically reloaded
Two modes of work

Standalone

- everything embedded
- just type `java -jar beans.jar` and go to your browser
- sufficient if you have data which fits into one disk

Fully distributed

- Apache Cassandra, Elastic, Apache Hadoop are needed
- suitable for many TBs of data
Example – Parallel coordinates

rows = **LOAD** 'MOCCA/system' using UniTable();

rows = **FILTER** rows **BY** tphys - FLOOR(tphys, 100.0) < 50.0 AND tphys < 14000;

rows = **FOREACH** rows **GENERATE** tbid, nt, DSPARAM(DSID(tbid), 'zini') as zini, DSPARAM(DSID(tbid), 'fracb') as fracb, rchut2, r_h as rh, rtid, xrc, vc, u1, irc, tphys, FLOOR(tphys, 100.0) as t, tphys - FLOOR(tphys, 100.0) as diff;

rowsGrouped = **GROUP** rows **BY** (t, tbid);
Example – Parallel coordinates

\[
\text{rowsFlat} = \text{FOREACH rowsGrouped GENERATE group.tbid as tbid, group.t as t, FLATTEN(rows), MIN(rows.diff) as minDiff;}
\]

\[
\text{rowsFlat} = \text{FILTER rowsFlat BY tphys - t < minDiff + 0.001;}
\]

\[
\text{STORE rowsFlat into 'NAME "System every 1Gyr"' using UniTable();}
\]
Example – Parallel coordinates

Rysunek 2: An example of a plot Parallel coordinates based on MOCCA 100k models
BEANS is going to be a tool for data analysis as AMUSE is for numerical simulations

http://www.beanscode.net/