Data Management

The ATLAS Distributed Data Management System, called DQ2, makes the experiment data available to scientists on the Worldwide LHC Computing Grid.

Almost 90 Petabytes of data, in 2.5 billion files, collected in 400’000 datasets, at almost 900 storage systems, distributed all over the world.

To make sure that we don’t lose track of this massive amount of data, we need to account it over time. However, such simple aggregate plots as shown above are unfortunately not sufficient.

### Step 3: Make analysis available

Finally, it must be possible to access the results. This is provided in two ways, a separate CSV file per requested summary per day, and as a table in HBase. Pig writes both outputs directly, and there is no intermediary step necessary. The flexible, non-relational tables of HBase allow to store the summaries in a compact and efficient way. Lookup is immediate.

### Step 2: Run the analysis

Pig uses an algebraic language that describes a data-pipeline, e.g., use all rows from a CSV dump where column 3 must end with string AOD, and aggregate the values from columns 4, 6 and 9. Pig then creates MapReduce jobs, which run on Hadoop.

Pig is extremely efficient, because it can create an algebraic minimum possible access path to the data, which is important when the dimensionality of a problem is high. That way, superfluous access to the data is inhibited. Current execution time for the summaries per day is at 8 minutes.

### Step 1: Prepare data for analysis

First, we need to extract the relevant data from the DQ2 tables in Oracle and write it into HDFS, the Hadoop Distributed File System. A daily job reads all necessary DQ2 tables and writes the result into a dump on HDFS, which can be mounted as a POSIX filesystem via FUSE. This data is a simple CSV file with 25 columns, with one file per day. This allows the recreation of historical summaries, as old data is preserved. The size of the dump is current about 3GB uncompressed for 10 million rows.

Accounting

**帐·户·记·录**

The system of recording and summarizing business and financial transactions and analyzing, verifying, and reporting the results. (Merriam-Webster, 2012)

Our business is data, so our accounting needs to work on data attributes like:

- What is the occupancy at groups of sites? e.g., all SCRATCH at UKTIER2s
- Distributed between datatypes? e.g., RAW, ESD, NTUP
- Matching some ATLAS physics definitions? e.g., reprocessing campaigns
- Of a particular tag? e.g., h405_r120
- And how much of it is temporary user data? e.g., can be deleted

The dimensionality of this problem by number of attributes is greater than 25, which presents a serious combinatorial, and therefore time delay, problem. So up to now, we only supported 3 dimensions: site, project and datatype.

Extending the accounting for arbitrary selections of attributes poses a hard problem that cannot right now be solved by "just aggregating data on the fly". For this reason, a new smart and scalable accounting system was developed.

### The new accounting system backend

**Apache HBase** is an open-source, distributed, versioned, column-oriented store modeled after Google’s BigTable. HBase supports random read/write access and can host very large tables, with billions of rows times millions of columns.

**Apache Pig** is a platform for analyzing large data sets that consists of a high-level language for expressing data analysis programs, coupled with infrastructure for evaluating these programs. The salient property of Pig programs is that their structure is amenable to substantial parallelization, which in turns enables them to handle very large data sets.

**Apache Hadoop** is a framework that allows for the distributed processing of large data sets across clusters of computers with MapReduce. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

**Cloudera** provides an enterprise-ready, commercial Distribution for Hadoop. Cloudera integrates the most popular projects related to Hadoop into a single package, which is run through a suite of rigorous tests to ensure reliability during production. The Cloudera distribution has shown to be of very high quality, and well documented.

**Puppet** is an IT automation software that helps system administrators manage infrastructure throughout its lifecycle, from provisioning and configuration to patch management and compliance. Currently, 12 nodes are provisioned in the CERN Computing Centre, which are managed by Puppet. This includes host configuration, software installation, and software configuration.

**Oracle** is a relational database management system, with advanced features like clustering, partitioning, hot backup, data warehousing, and much more. DQ2 stores its transactional and relational data in the central Oracle Database 11g at CERN. This is the core data that needs to be accounted.

Mario Lassnig, Vincent Garonne, Gancho Dimitrov, Luca Canali for the ATLAS Collaboration

CHEP 2012