Job optimization in ATLAS TAG-based Distributed Analysis

Computing in High Energy and Nuclear Physics
Prague | Czech Republic | 21 - 27 March 2009

US ATLAS

the Problem

ATLAS is projected to collect over one billion events/year during the first few years of operation without counting the events produced by simulation. The same events are available at different stages of processing: RAW, ESD (Event Summary Data), AOD (Analysis Object Data), DPD (Derived Physics Data).

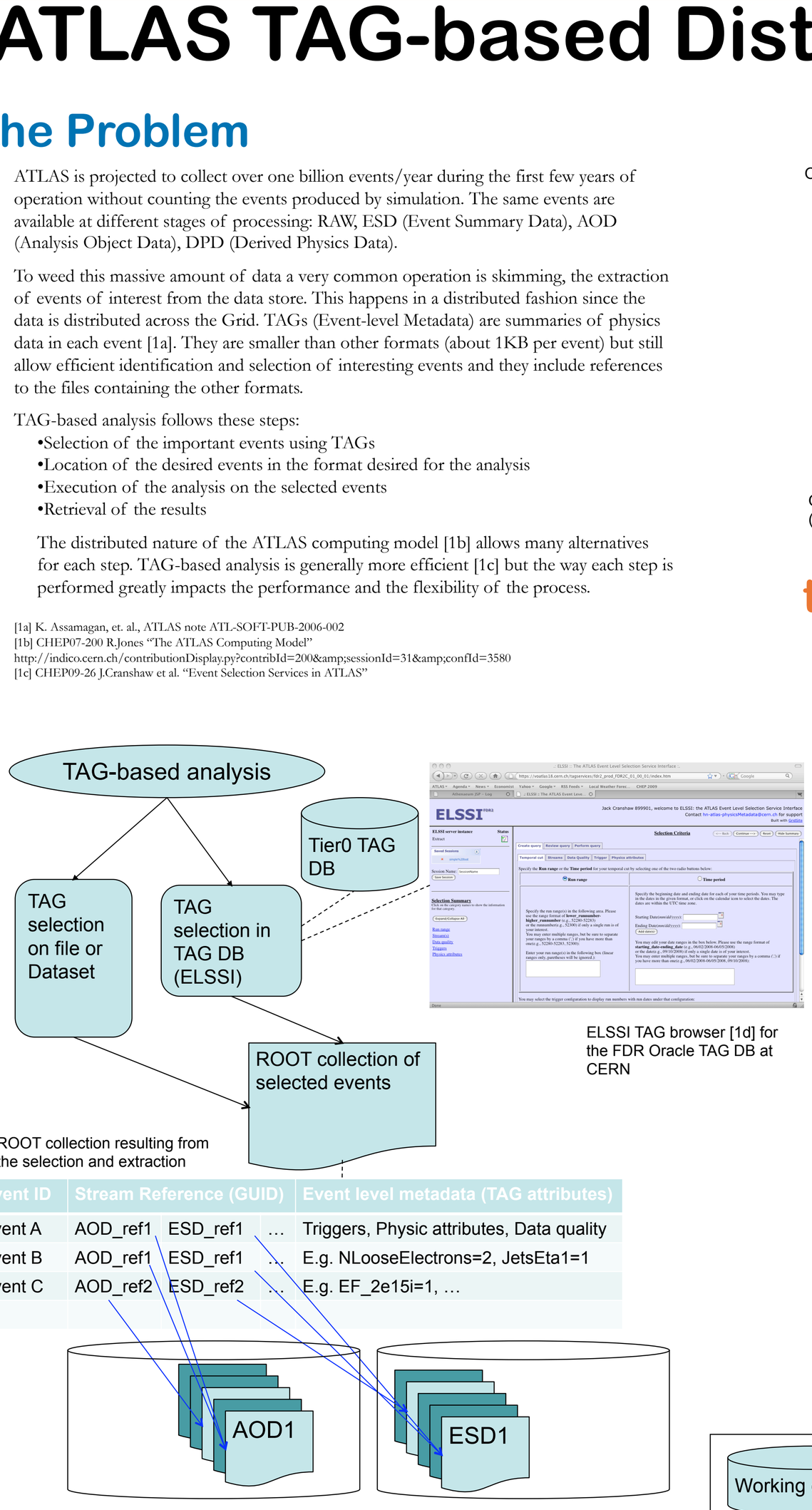
To weed this massive amount of data a very common operation is skimming, the extraction of events of interest from the data store. This happens in a distributed fashion since the data is distributed across the Grid. TAGs (Event-level Metadata) are summaries of physics data in each event [1a]. They are smaller than other formats (about 1KB per event) but still allow efficient identification and selection of interesting events and they include references to the files containing the other formats.

TAG-based analysis follows these steps:

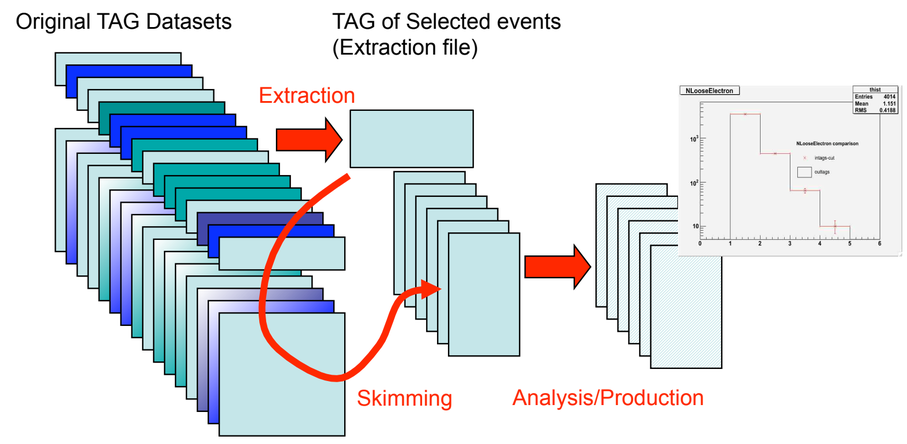
- Selection of the important events using TAGs
- Location of the desired events in the format desired for the analysis
- Execution of the analysis on the selected events
- Retrieval of the results

The distributed nature of the ATLAS computing model [1b] allows many alternatives for each step. TAG-based analysis is generally more efficient [1c] but the way each step is performed greatly impacts the performance and the flexibility of the process.

[1a] K. Assamagan, et. al., ATLAS note ATL-SOFT-PUB-2006-002
[1b] CHEP07-200 R.Jones "The ATLAS Computing Model"
http://indico.cern.ch/contributionDisplay.py?contribId=200&sessionId=31&confId=3580
[1c] CHEP09-26 J.Cranshaw et al. "Event Selection Services in ATLAS"



ELSSI TAG browser [1d] for the FDR Oracle TAG DB at CERN



the TAG-based Analysis Process

Event selection. TAG information is stored in the different technologies supported by LCG POOL[2a] (ROOT files, relational DB, …) and it is accessed using POOL command line utilities, the Athena framework or the interactive Web frontend ELSSI[2b]. The result of the selection is a ROOT[2c] collection containing the desired events (Extraction file).

File location. Depending on the desired data format the File unique IDs (GUID) of the files containing the selected events can be extracted from the result of the selection. Then the files must be located on the Grids used by ATLAS and the local path at the site is required for the processing.

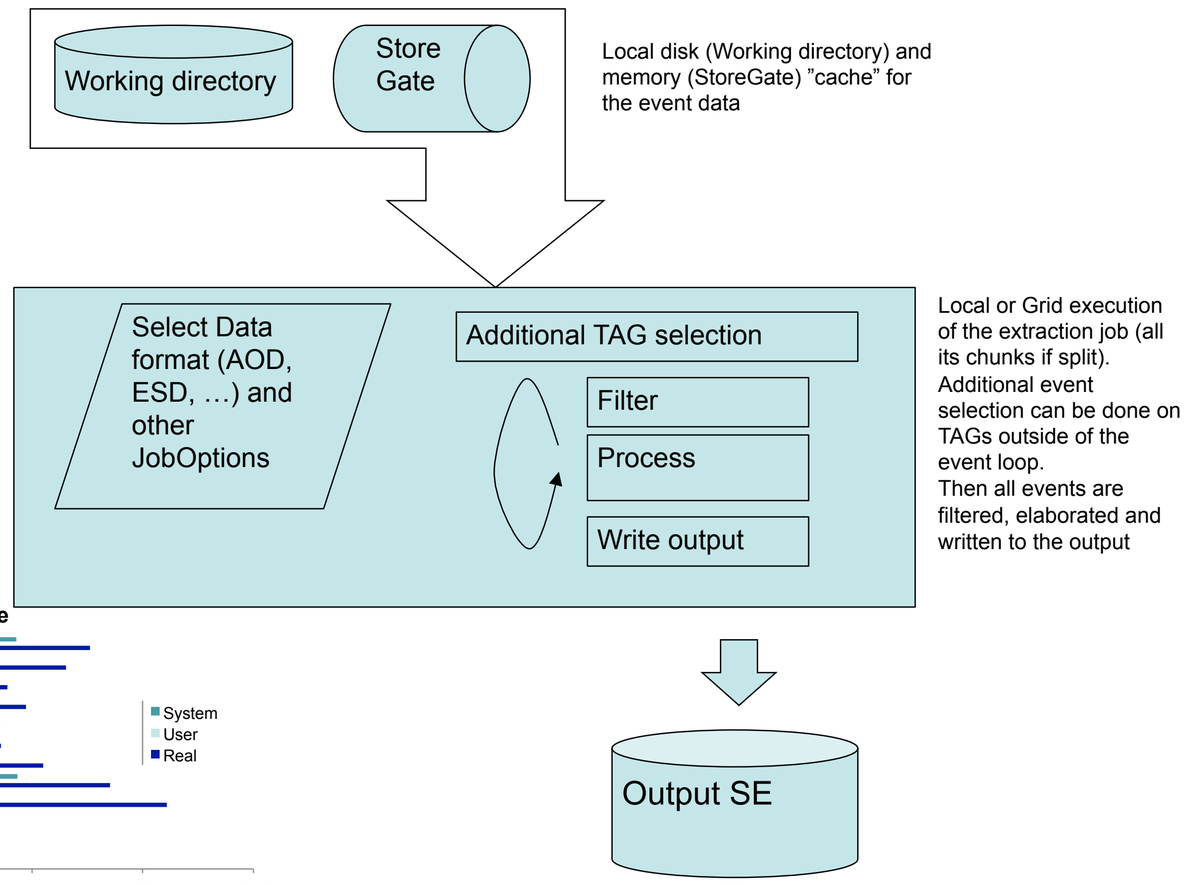
Pathena[2d] has been modified to isolate only the files containing the selected events in the desired format and find them locally to prepare the input file catalog for Athena[2e] (PoolfileCatalog.xml). A JSON-based Web Service has been developed to translate GUIDs into file names and find their Dataset and location across multiple Sites and Grids. This is using the information stored in DQ2[2f], ATLAS distributed data manager, and the local file catalogs at the sites.

Analysis execution. Included here are user productions, like DPD productions. It is ATLAS policy not to automatically move data for analysis jobs. Sometimes the job is split on a file or event basis and executed where input files are available (both the Extraction file and the referenced events, e.g. AOD, ESD, …). Panda jobs normally copy the input files to the working directory. POSIX file systems, as well as dCache, Castor and xRootd, also allow direct access. Different combinations of direct access or copy of the input files were tested and compared. As a result current TAG-based analysis in Panda copies the TAG files to the working directory and accesses directly into the SE the referenced events.

Pathena job splitting has been improved to take into account which files of the input Dataset include selected events (avoiding to run on files that do not contain selected events)

Result retrieval. dq2-get, a DDM command, allows retrieval of all the output produced by the whole analysis even if it resides on different sites.

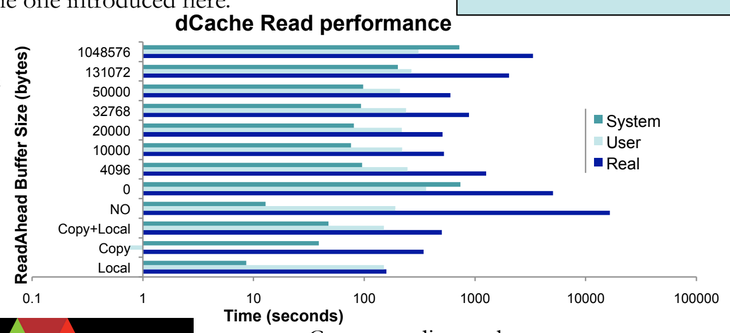
[2a] POOL http://pool.cern.ch/
[2b] CHEP09-26 J.Cranshaw et al. "Event Selection Services in ATLAS"
[2c] R. Brun and F. Rademakers. "ROOT - An Object Oriented Data Analysis Framework", Nucl. Inst. & Meth. in Phys. Res. A 389 (1997) 81-86. See also http://root.cern.ch/
[2d] Pathena (Athena on PanDA) https://twiki.cern.ch/twiki/bin/view/Atlas/PandaAthena
[2e] Athena https://twiki.cern.ch/twiki/bin/view/Atlas/AthenaFramework
[2f] DQ2 https://twiki.cern.ch/twiki/bin/view/Atlas/DistributedDataManagement



the Tests

Most of the tests have been running on the Chicago cluster of the ATLAS Midwest Tier2. The cluster has several multicore hosts in a PBS queue. A gigabit Ethernet interconnects the nodes and a dCache [3a] storage element with about 200TB.
Jobs have been submitted locally or on the Grid using Pathena [3b]
The chart below shows the effect of dCache ReadAhead buffer: the simple enabling of the buffer improves data access speed in an analysis job accessing the file in the SE, even if most of the data buffered is also discarded (efficiency is always below 3%). The length of the buffer is not really important as long as it is enabled. When accessing all the events the performance is comparable with local copy + local execution. In the charted tests almost 100% of the events are selected. In a selective skim direct access is even better. Other tests showed the inefficiencies of different ways to split the job (e.g. TAG unaware splitting on a per file/event basis) compared to the one introduced here.

[3a] dCache http://www.dcache.org/
[3b] Pathena https://twiki.cern.ch/twiki/bin/view/Atlas/PandaAthena



THE UNIVERSITY OF CHICAGO

Argonne NATIONAL LABORATORY

Corresponding author:
Marco Mambelli (University of Chicago)
marco@hep.uchicago.edu

Other Authors:
Jack Cranshaw (Argonne National Laboratory)
Tadashi Maeno (Brookhaven National Laboratory)
David Malon (Argonne National Laboratory)
Marcin Novak (Brookhaven National Laboratory)

BROOKHAVEN NATIONAL LABORATORY

DEPARTMENT OF ENERGY · UNITED STATES OF AMERICA