

DISTRIBUTED ANALYSIS IN ATLAS USING GANGA

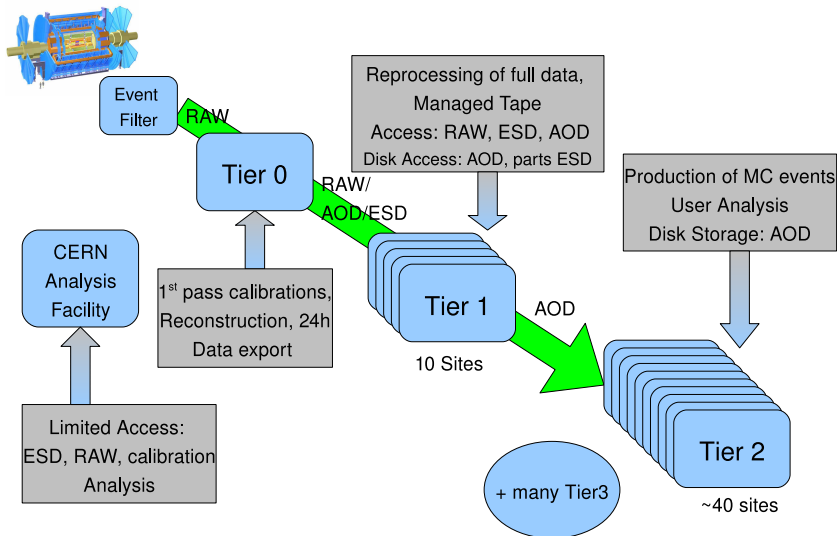
Johannes Elmsheuser

Ludwig-Maximilians-Universität München, Germany

24 March 2009/CHEP'09, Prague

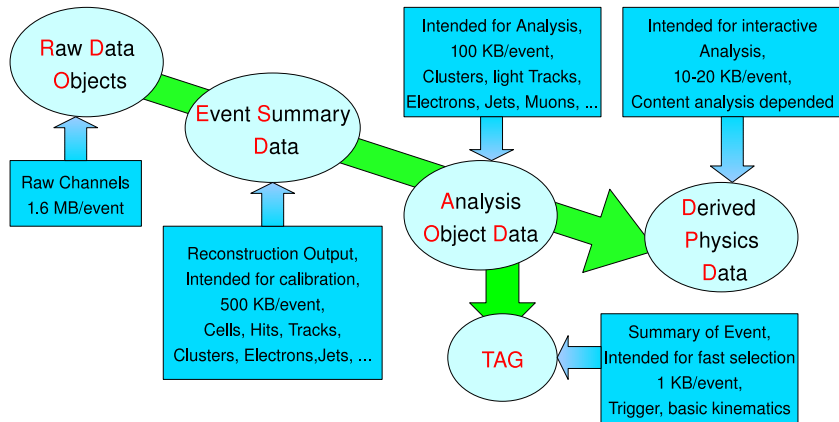


ATLAS DATA REPLICATION AND DISTRIBUTION



ATLAS EVENT DATA MODEL

Refining the data by: Add higher level info, Skin, Thin, Slim



- Heterogeneous grid environment based on 3 grid infrastructures:



- Grids have different middle-ware, replica catalogues and tools to submit jobs

⇒ Hide differences and complexity from the ATLAS user

DISTRIBUTED ANALYSIS MODEL

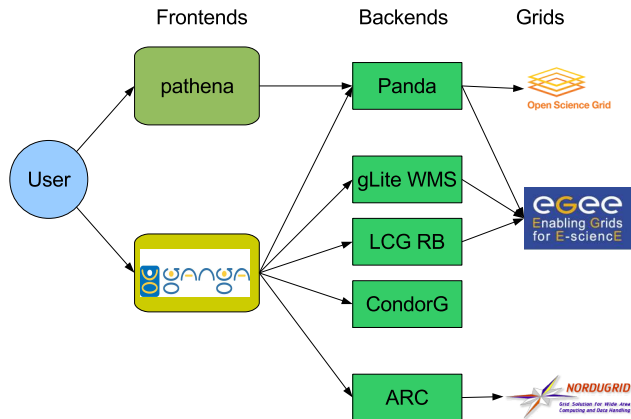
The distributed analysis model is based the ATLAS computing model

- Data is distributed to Tier1 and Tier2 facilities by default by the ATLAS Data Distribution system DQ2
 - available 24/7
 - Automated file management, distribution and archiving throughout the whole grid using a Central Catalogue, FTS, LFCs
 - Random access needs a pre-filtering of data of interest, e.g. Trigger or ID streams or TAGs (event-level meta data)
- user jobs are sent to the data
large input data-sets (several TBs)
- Results must be made available to the user
potentially already during processing
- Data is added with meta-data and bookkeeping in catalogues

SOME ANALYSIS WORK-FLOWS

- classic AOD/DPD analysis:
 - Athena user code sequentially processes large Monte Carlo or Data stream sample on the Grid
 - Produces ROOT tuple output which is further processed locally or on the Grid
- TAG plus AOD:
 - TAGs:
 - very small event summary
 - ROOT file or Database format
 - TAG pre-selection by seeking through AOD file
 - Further steps as above
- Small MC Sample Production:
 - Use Production System Transformation (Geant or Atlfast) to produce a small MC sample for special/official usage
- ROOT:
 - Generic ROOT application eventually with DQ2 access for e.g. Toy MC

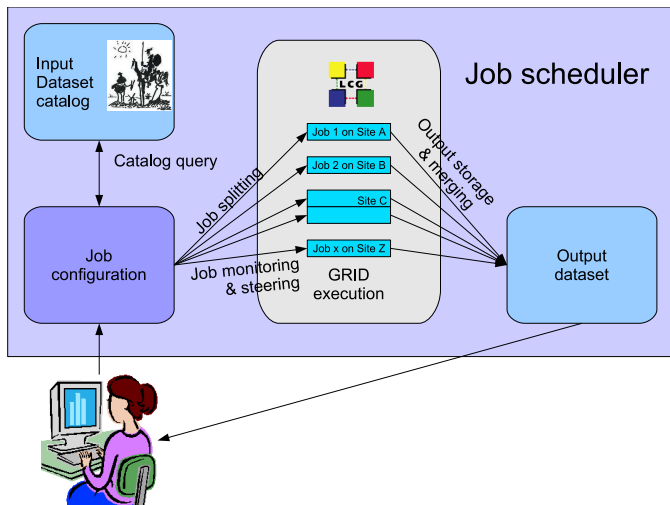
DISTRIBUTED ANALYSIS - CURRENT SITUATION



Data is centrally being distributed by DQ2 - Jobs go to data

DISTRIBUTED ANALYSIS

How to combine all different components: **Job scheduler/manager:**
GANGA



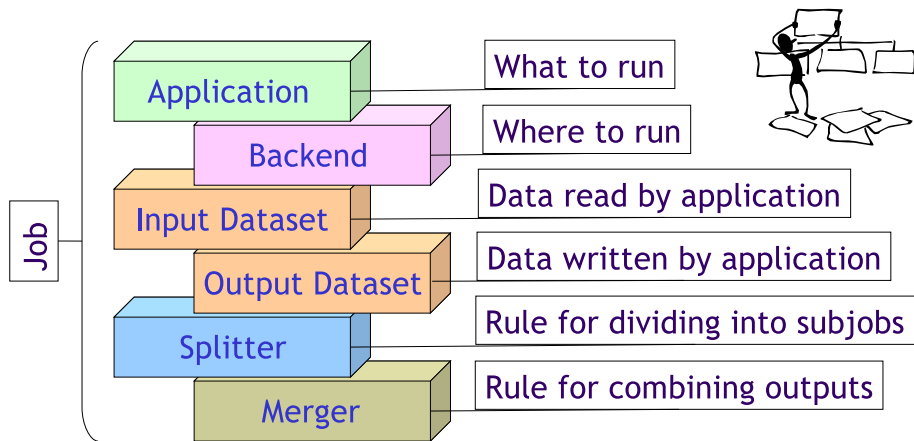
FRONT-END CLIENT: GANGA



- A **user-friendly** job definition and management tool.
- Allows simple switching between testing on a **local batch system** and large-scale data processing on distributed resources (**Grid**)
- Developed in the context of **ATLAS** and **LHCb** :
 - For **ATLAS**, have built-in support for applications based on **Athena** framework, for Production System **JobTransforms**, and for **DQ2** data-management system
- **Component** architecture readily allows extension
- Python framework
- GANGA is distributed under the GPL license
- For details see talk of D. van der Ster on Monday and A. Maier on Thursday

GANGA JOB ABSTRACTION

- GANGA simplifies running of ATLAS (and LHCb) applications on a variety of Grid and non-Grid back-ends



JOB DEFINITION USING ATLAS SOFTWARE

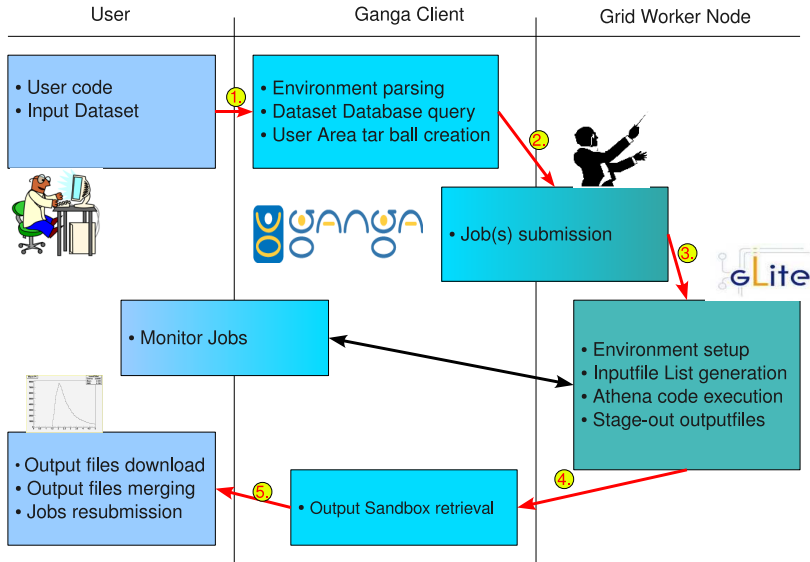
GANGA offers three ways of user interaction:

- Shell command line
- Interactive IPython shell
- Graphical User Interface

Job definition at command line for GRID submission:

```
ganga athena
  --inDS fdr08_run2.0052283.physics_Muon.merge.AOD.o3_f8_m10
  --outputdata AnalysisSkeleton.aan.root
  --split 3
  --lcg --cloud DE
  AnalysisSkeleton_topOptions.py
```

JOB WORK-FLOW: ATHENA ON LCG BACK-END



NEW IN GANGA 5

New in GANGA 5.0 and 5.1:

- GANGA 5.0.0: 13 June 2008
- GANGA 5.1.8 released: 6 March 2009
- 18 minor bug-fix and feature releases in between

GangaAtlas highlights:

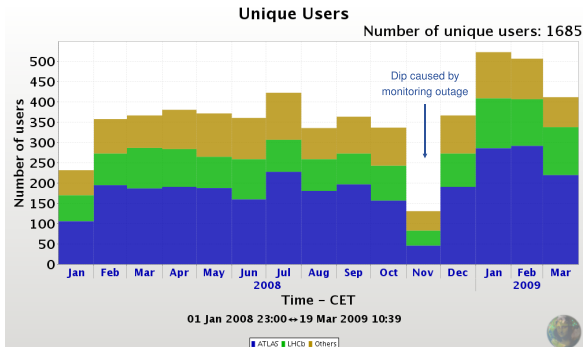
- GangaNG and GangaPanda: All 3 Grid flavours supported
- FileStager: background tread lcg-cp of input files
- Many improvements to DQ2 job splitter algorithm
- Many improvements of DQ2 integration - e.g. data-set/file tracer
- Add new work-flows: AthenaRootAccess
- Improved job statistics and reporting

Further Details:

- Poster about GangaPanda

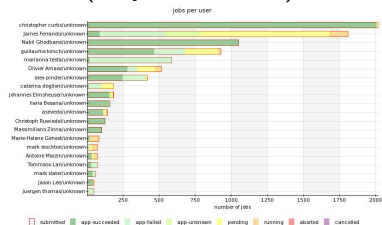
GANGA USAGE STATISTICS

- GANGA has been used by over 1500 users in total
- now approx. 150 ATLAS user per week. It is twice as much compared to one year ago.



NUMBER OF USER ANALYSIS JOBS

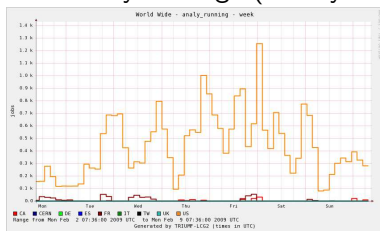
Dashboard view of GANGA usage (only WMS here):



~ 10k jobs per day

- Compare with up to ~ 100k finished daily production jobs
- Seeing an increased number of user in the last few months - but we expect many more !
- Testing system with daily functional tests: **GangaRobot**
- Need to test the DA system under high load: **HammerCloud**
- Further details: See „HammerCloud” talk on Thursday

Panda Analysis usage (mainly US):



Frequently asked questions or problems:

- Where is my data ?
- There is a problem with my special code configuration
- The job had problems with accessing the input data files
- The ratio of CPU and Wall-time is largely varying btw. 10% - 100% and depends on the site and user



Support:

- Started ATLAS wide user support mailing list for DA
- Shifters in EU and US time zone
- Hoping for user2user support
- Has developed to one of the busiest mailing lists in ATLAS



CONCLUSIONS AND SUMMARY

For the distributed analysis it is vital to have:

- Easy interface that does not scare off physicists
- A reliable and robust service of many components

What is working well so far:

- Analysis at a chosen number of sites
- Small scale MC production
- Automatic Standard Job Configurations

What works, but needs improvement:

- 'Blind' job submission
- Site availability and Input file access
- Exotic use cases

Homepage:

- <http://cern.ch/ganga>

Paper:

- GANGA: a tool for computational-task management and easy access to Grid resources (arXiv:0902.2685v1)