

# ATLAS DISTRIBUTED ANALYSIS DEMO

Johannes Elmsheuser

Ludwig-Maximilians-Universität München, Germany

25 Sep 2006/LHCC comprehensive review



- ① ATLAS DISTRIBUTED ANALYSIS
- ② TASK: ACCESS DATA IN DISTRIBUTED ENVIRONMENT
- ③ INTRODUCTION TO THE TOOL
- ④ HOW TO PREPARE A JOB TO RUN ON THE GRID
- ⑤ EXAMPLE OF REAL ACTIVITY
- ⑥ CONCLUSIONS

- ① ATLAS DISTRIBUTED ANALYSIS
- ② Task: Access data in distributed environment
- ③ Introduction to the tool
- ④ How to prepare a job to run on the Grid
- ⑤ Example of real activity
- ⑥ Conclusions

# ATLAS DISTRIBUTED ANALYSIS

- ATLAS data analysis foreseen on Tier2 sites of three GRIDs: LCG, Nordugrid, OSG
- Data from Production System is currently consolidated by DDM-operations team on 3 sites: CERN, Lyon, BNL
- LCG tool for distributed analysis: GANGA
- OSG production/analysis system tool: Panda
  
- ATLAS Analysis model implemented by PAT team foresees Athena analysis of AODs/ESDs and interactive use of Athena-aware-ROOT tuples

- ① ATLAS Distributed Analysis
- ② **TASK: ACCESS DATA IN DISTRIBUTED ENVIRONMENT**
- ③ Introduction to the tool
- ④ How to prepare a job to run on the Grid
- ⑤ Example of real activity
- ⑥ Conclusions

# TASK DESCRIPTION

- Plot different kinematic distributions of:
  - Signal:  $H \rightarrow WW \rightarrow \mu\nu\mu\nu$  (PYTHIA,  $m_H = 170 \text{ GeV}$ )
  - Main background:  $Z/\gamma^* \rightarrow \mu\mu$  (PYTHIA,  $60 \text{ GeV} < m_{\mu\mu}$ )
- Start from local working area with ATHENA code reading AODs and producing ROOT-tuples with essential information
- Datasets are located on the LCG-GRID
  - Wrap up working area, send to dataset location, produce output ROOT-tuples, retrieve output dataset ROOT-tuples
  - Most of these tasks are managed by: **GANGA**
- Finally use small ROOT-tuples to produce kinematic distributions

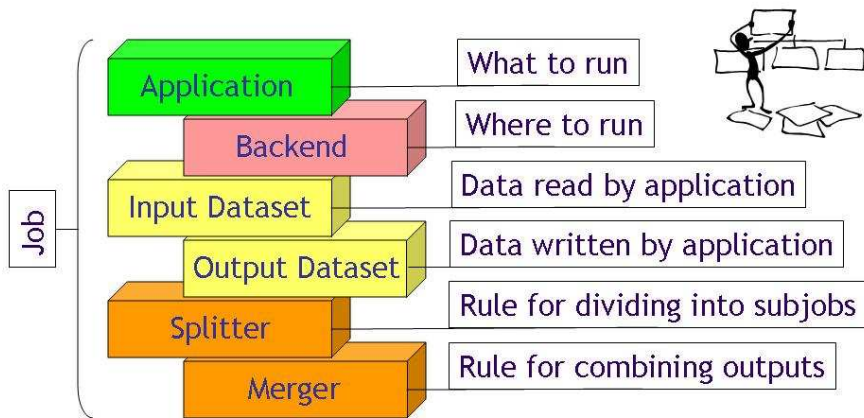
- ① ATLAS Distributed Analysis
- ② Task: Access data in distributed environment
- ③ **INTRODUCTION TO THE TOOL**
- ④ How to prepare a job to run on the Grid
- ⑤ Example of real activity
- ⑥ Conclusions

- Ganga is a user tool for job definition and management
- 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 JobTransforms, and for DQ2 data-management system
- Component architecture readily allows extension
- Python framework
  
- Development team: F.Brochu (Cambridge), U.Egede (Imperial), J.Elmsheuser (München), K.Harrison (Cambridge), H.C.Lee (ASCC), D.Liko (CERN), A.Maier (CERN), J.T.Moscicki (CERN), A.Muraru (Bucharest), V.Romanovsky (IHEP), A.Soroko (Oxford), C.L.Tan (Birmingham) and contributions past and present from many others

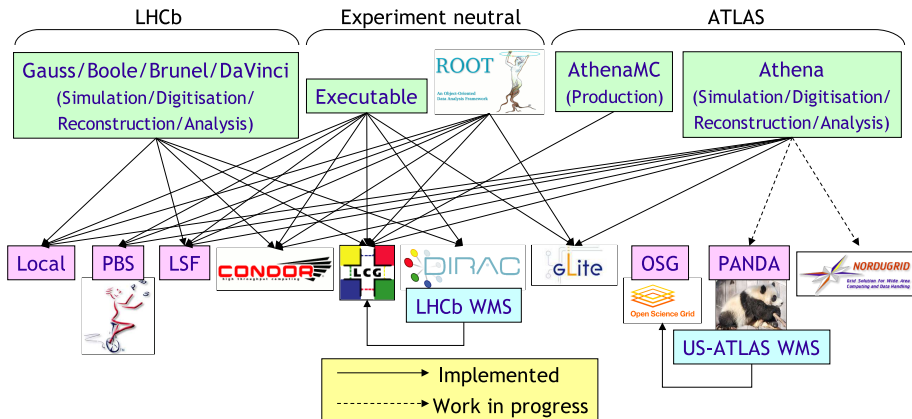


# GANGA II

- Ganga is based on a simple, but flexible, job abstraction
- A job is constructed from a set of building blocks, not all required for every job



- Ganga simplifies running of ATLAS (and LHCb) applications on a variety of Grid and non-Grid backends



- ① ATLAS Distributed Analysis
- ② Task: Access data in distributed environment
- ③ Introduction to the tool
- ④ HOW TO PREPARE A JOB TO RUN ON THE GRID
- ⑤ Example of real activity
- ⑥ Conclusions

- Data is organized in datasets, list of files
- AMI Metadata:
  - <http://ami3.in2p3.fr:8080/AMI/>
  - query different production steps and generator options
- Dataset browser:
  - <http://gridui02.usatlas.bnl.gov:25880/server/pandamon/query?overview=dslist>
- Dashboard (still under construction):
  - DDM monitoring:
  - Job monitoring: <http://arda-dashboard.cern.ch/atlas>

- Declaring input data:
  - Ganga has DDM/DQ2 integration
  - Select dataset name and specify as input dataset parameter to Ganga
  - Either submit to specific site of dataset location or let Ganga do it automatically during submission
- Splitting parameter:
  - Define to number of parallel sub-jobs as input parameter to Ganga
- Declaring output data:
  - Specify names of output files
  - Stored on remote storage element and optional registered into DDM/DQ2
- Source code:
  - Working area is wrapped up together with jobOptions and re-compiled on remote GRID host

- ① ATLAS Distributed Analysis
- ② Task: Access data in distributed environment
- ③ Introduction to the tool
- ④ How to prepare a job to run on the Grid
- ⑤ **EXAMPLE OF REAL ACTIVITY**
- ⑥ Conclusions

# JOB DEFINITION (I)

- Job definition from command line on local desktop:

```
athena AnalysisSkeleton_jobOptions.py
```

- Job definition from command line to the GRID:

```
ganga athena \  
  --inDS csc11.005320.PythiaH170ww11.recon.A0D.v11004107 \  
  --outputdata AnalysisSkeleton.aan.root \  
  --split 3  
  --ce ce106.cern.ch:2119/jobmanager-lcglsf-grid_2nh_atlas \  
  AnalysisSkeleton_jobOptions.py
```

## JOB DEFINITION (II)

Job definition within GANGA IPython shell

```
j = Job()
j.name='5145'
j.application=Athena()
j.application.prepare()
j.application.option_file='$HOME/athena/testarea/11.0.5/InstallArea
j.splitter=AthenaSplitterJob()
j.splitter.numsubjobs = 10
j.inputdata=DQ2Dataset()
j.inputdata.type='DQ2_LOCAL'
j.inputdata.dataset='csc11.005145.PythiaZmumu.recon.AOD.v11004103'
j.outputdata=DQ2OutputDataset()
j.outputdata.outputdata=['AnalysisSkeleton.aan.root']
j.backend=LCG()
j.backend.CE='ce-fzk.gridka.de:2119/jobmanager-pbspro-atlasS'
j.submit()
```



- csc11.005320.PythiaH170wwll.recon.AOD.v11004107
  - $H \rightarrow WW \rightarrow \mu\nu\mu\nu$ ,  $m_H = 170$  GeV
  - PYTHIA 6.3,  $\sigma = X$  pb
  - 19 files, 3 jobs
  
- csc11.005145.PythiaZmumu.recon.AOD.v11004103
  - $Z/\gamma^* \rightarrow \mu\mu$ ,  $60 \text{ GeV} < m_{\mu\mu}$
  - PYTHIA 6.3,  $\sigma = 1497$  pb
  - 106 files, 10 jobs

# MONITORING WITH THE IPYTHON SHELL

```
elmsheus@lxplus014:~ - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe

Ganga.GPIDEv.Lib.Job : INFO killing job 34
Ganga.GPIDEv.Lib.Job : INFO killing job 34.0
Ganga.GPIDEv.Lib.Job : INFO job 34.0 status changed to "killed"
Ganga.GPIDEv.Lib.Job : INFO killing job 34.1
Ganga.GPIDEv.Lib.Job : INFO job 34.1 status changed to "killed"
Ganga.GPIDEv.Lib.Job : INFO killing job 34.2
Ganga.GPIDEv.Lib.Job : INFO job 34.2 status changed to "killed"
Ganga.GPIDEv.Lib.Job : INFO job 34 status changed to "killed"
Out [2]: 1

In [3]:

In [3]:execfile('/afs/cern.ch/user/e/elmsheus/split/5320_cern.py')
Ganga.GPIDEv.Lib.Job : INFO submitting job 35
Ganga.GPIDEv.Adapters : INFO submitting job 35.0 to LCG backend
Ganga.GPIDEv.Adapters : INFO submitting job 35.1 to LCG backend
Ganga.GPIDEv.Adapters : INFO submitting job 35.2 to LCG backend

In [4]:

In [4]:jobs
Out [4]: Statistics: 25 jobs
-----
# id status name subjobs application backend backend.actualCE
# 6 completed Executable LCG ce-fzk.gridka.de:2119/jobmanager-pbspro-atlas
# 7 new Executable LCG
# 8 new Executable LCG
# 9 completed Executable LCG grid10.lal.in2p3.fr:2119/jobmanager-pbs-atlas
# 10 completing Executable LCG ce-fzk.gridka.de:2119/jobmanager-pbspro-atlas
# 12 completed Athena LSF
# 13 new Executable Local
# 14 completed Athena LSF
# 15 completed 2 Athena LCG
# 16 completed Athena LSF
# 17 completed athena_lcg_dq2_test1, GRIDNA, DQ2_LOCAL, 7233 Athena LCG ce-fzk.gridka.de:2119/jobmanager-pbspro-atlas
# 18 new athena_lcg_dq2_test2, LYON, DQ2_LOCAL, 5300 Athena LCG
# 19 completed athena_lcg_dq2_test2, LYON, DQ2_LOCAL, 5300 Athena LCG cclgccl102.in2p3.fr:2119/jobmanager-bqs-shor
# 20 completed Athena LSF
# 22 submitted 5320 3 Athena LCG
# 24 submitted 5320 10 Athena LCG
# 26 completed 5145_short 2 Athena LCG
# 27 completed 5300 3 Athena LCG
# 28 new 5300 Athena Local
# 29 completed 5300 3 Athena LCG
# 30 killed 5145 10 Athena LCG
# 31 completed 5145_short 5 Athena LCG
# 32 running 5300 3 Athena LCG
# 34 killed 5320_cern 3 Athena LCG
# 35 submitted 5320_cern 3 Athena LCG

In [5]:
```

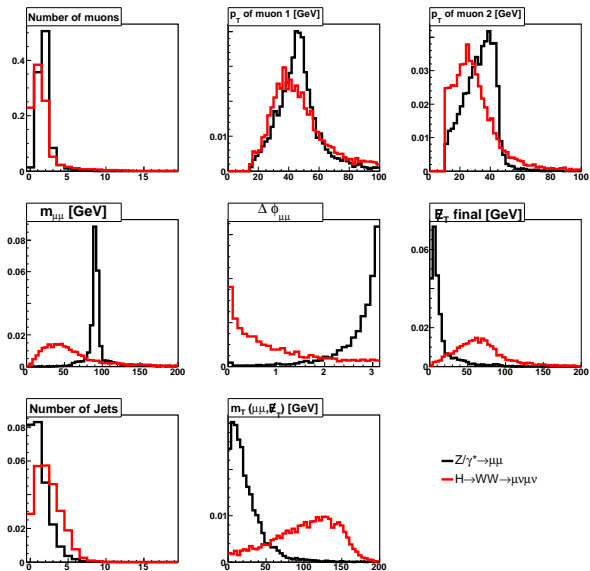
# MONITORING WITH THE GUI

The screenshot shows the Ganga GUI interface. At the top, the window title is 'GANGA <@tplus014.cern.ch>'. Below the title bar, there are menu options: 'Job', 'Folders', 'View', and 'Help'. A toolbar contains icons for file operations and checkboxes for 'Scriptor', 'Log', and 'Job Builder'. The main area is divided into two panes. The left pane, titled 'Jobs', contains a table with columns: 'id', 'status', 'name', 'application', 'exe filename', and 'backend'. The right pane, titled 'Job Details', shows the configuration for a specific job.

id	status	name	application	exe filename	backend
6	completed		Executable	closese.sh	LCG
7	new		Executable	closese.sh	LCG
8	new		Executable	closese.sh	LCG
9	completed		Executable	closese.sh	LCG
10	completing		Executable	closese.sh	LCG
12	completed		Athena	*NVA*	LSF
13	new		Executable	echo	Local
14	completed		Athena	*NVA*	LSF
15	completed		Athena	*NVA*	LCG
16	completed		Athena	*NVA*	LSF
17	completed	athena_lcg_dq2_test1, GRIDKA, DQ2_LOCAL, 7233	Athena	*NVA*	LCG
18	new	athena_lcg_dq2_test2, LYON, DQ2_LOCAL, 5300	Athena	*NVA*	LCG
19	completed	athena_lcg_dq2_test2, LYON, DQ2_LOCAL, 5300	Athena	*NVA*	LCG
20	completed		Athena	*NVA*	LSF
22	submitted	5320	Athena	*NVA*	LCG
24	submitted	5320	Athena	*NVA*	LCG
26	completed	5145_short	Athena	*NVA*	LCG
27	completed	5300	Athena	*NVA*	LCG
28	new	5300	Athena	*NVA*	Local
29	completed	5300	Athena	*NVA*	LCG
30	failed	5145	Athena	*NVA*	LCG
31	completed	5145_short	Athena	*NVA*	LCG
32	running	5300	Athena	*NVA*	LCG
34	submitted	5320_cern	Athena	*NVA*	LCG

```
Job Details
Job {
  status = 'submitted',
  name = '5320_cern',
  inputdir = '/afs/cern.ch/user/e/elmshesus/gangadir/workspace/Local/34/input/'
  outputdir = '/afs/cern.ch/user/e/elmshesus/gangadir/workspace/Local/34/output/'
  outputsandbox = [],
  id = 34,
  inputdata = DQ2Dataset (
    tagdataset = '',
    type = 'DQ2_LOCAL',
    names = [],
    match_ce = False,
    dataset = 'csc11.005320.PythiaH170wll.recon.AOD.v11004107'
  ),
  merger = None,
  inputsandbox = [],
  application = Athena (
    atlas_release = '11.0.5',
    max_events = None,
    options = None,
    user_setupfile = File (
      name = '',
      subdir = ''
    ),
    option_file = File (
      name = '/afs/cern.ch/user/e/elmshesus/athena/testarea/11.0.5/PhysicsAni
      subdir = ''
    ),
    user_area = File (
      name = '/afs/cern.ch/user/e/elmshesus/gangadir/workspace/Local/UserAr
      subdir = ''
    )
  ),
  outputdata = DQ2OutputDataset (
    output = [],
    datasetname = ''
  )
}
```

# RESULTS



- ① ATLAS Distributed Analysis
- ② Task: Access data in distributed environment
- ③ Introduction to the tool
- ④ How to prepare a job to run on the Grid
- ⑤ Example of real activity
- ⑥ CONCLUSIONS

- Demonstrated Distributed Analysis of ATLAS/Athena jobs with GANGA on LCG
- GANGA:
  - very flexible setup and easy extendable through plugin design
  - Job submission, Monitoring and Output retrieval almost fully automatic
- Future Improvements:
  - User view: bulk submission, better Athena job error recovery and book-keeping
  - Developer view: robust data access on all possible sites