

# USER ANALYSIS IN ATLAS

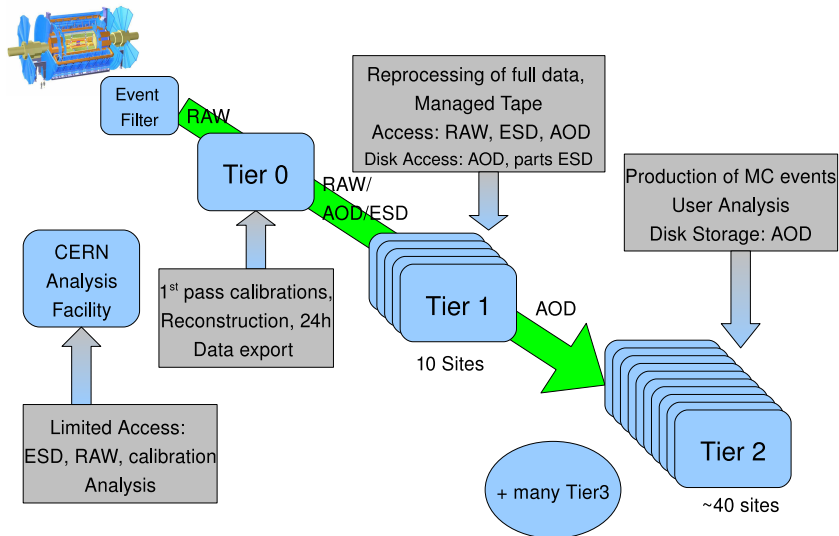
Johannes Elmsheuser

Ludwig-Maximilians-Universität München, Germany

10 February 2009/HGF Grid Worrkshop

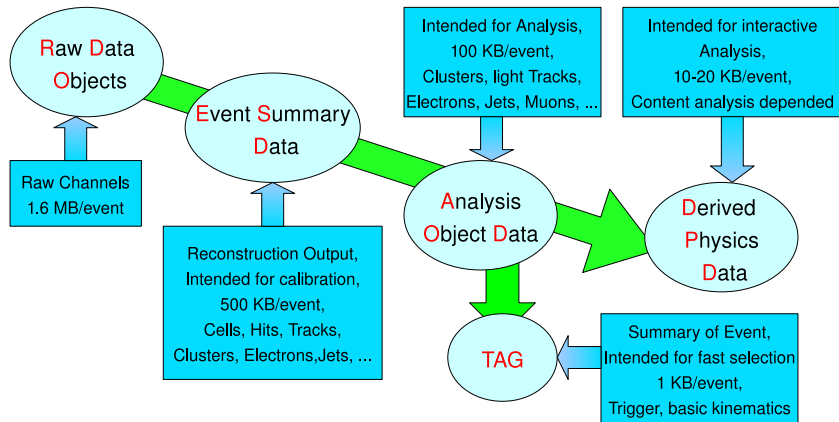


# ATLAS DATA REPLICATION AND DISTRIBUTION



# ATLAS EVENT DATA MODEL

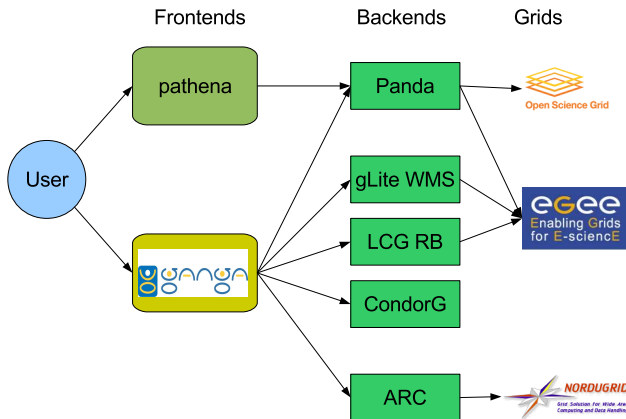
Refining the data by: Add higher level info, Skin, Thin, Slim



# ANALYSIS WORK-FLOWS - WHAT IS POSSIBLE ?

- Classic AOD/ESD/DPD analysis:
  - Athena and AthenaROOTAccess (sequential access)
  - A lot of MC processing, seeing now Cosmics and re-processed data access
- TAG plus AOD:
  - ROOT file format (seeking only particular events)
  - DB access with web front-end ELSSI (still under testing)
- Calibration:
  - Muon calibration, ID alignment (still under testing)
  - RAW and remote Database access (still under testing)
- Small MC Sample Production:
  - Default Production System Transformations (Geant or Atlfast)
- ROOT:
  - Generic ROOT application also with DQ2 access
- Generic Executable
  - Full support

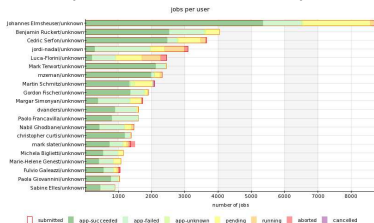
# DISTRIBUTED ANALYSIS - CURRENT SITUATION



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

# NUMBER OF USER ANALYSIS JOBS

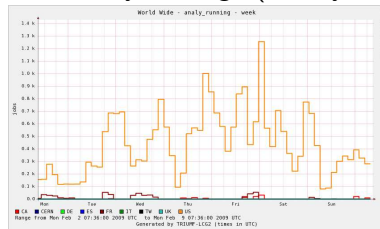
Dashboard view of Ganga usage  
(only LCG backend):



~ 70k jobs in last 7 days

- Compare with currently ~ 100k daily production jobs
- Seeing an increased number of user in the last few months - but we expect many more !
- Support is becoming large issue: Central ATLAS user support mailing list with shifters and experts

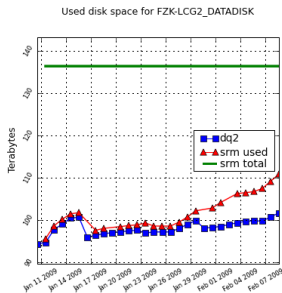
Panda Analysis usage (mainly US):



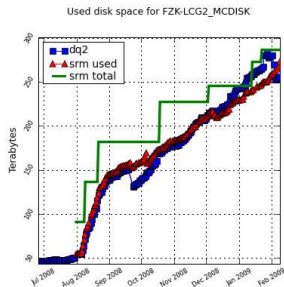
All centrally managed data is organized with DQ2

- ATLAS is distribution data with DQ2 system:
  - Cloud: Tier1 + many Tier1/2
  - Central DQ2 catalog at CERN and LFC catalog in every cloud
  - Data Movement via FTS
- Storage structure
  - Every site with set of different purposed space tokens
  - Production, Replication: MCDISK, DATADISK, PRODDISK, PHYS-GRROUPDISKs
  - Users: scratch area: USERDISK, Permanent: LOCALGROUPDISK
- Every Tier2/3 with DQ2 capabilities needs to have FTS endpoint and some space tokens

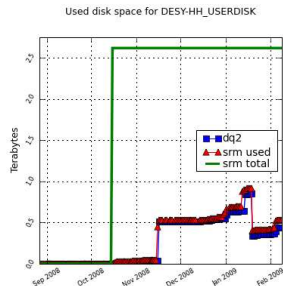
# SPACE TOKEN EXAMPLES



DATADISK at FZK



MCDISK at FZK



USERDISK at  
DESY-HH

Bookkeeping and Clean-up is a steady struggle and tedious jobs



# ANALYSIS STRESS TESTS - HAMMERCLOUD

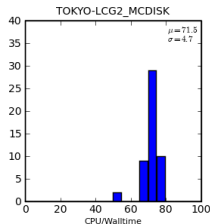
In November ATLAS started testing sites with analysis stress tests

- Purpose:
  - Systematically test different DQ2 enabled ATLAS sites with regular user analysis
- Differences to MC production:
  - More chaotic user analysis compared to organized MC production
  - User analysis puts a much higher load on the SE
- Analysis used so far:
  - AOD analysis based on UserAnalysis analyzing mainly muons
  - Athena 14.2.20, mc08\*AOD\*e\*s\*r5\*, some muon datasets
- Test procedure
  - Started with manual submission 50-200 jobs 5 people to IT and DE clouds using Ganga/glite WMS
  - Now: Automatic system with Ganga, MySQL DB for test scheduling, submission and reporting

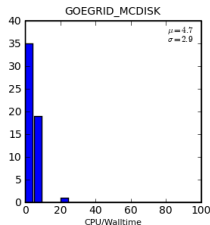
# ANALYSIS STRESS TESTS - HIGHLIGHTS

- Ganga code works reliable - small work-flow and speed improvements
- 200-300 analysis jobs easily saturates 1 GB switch between worker node and SE
- Testing different input access modes:
  - Posix I/O: rfio or dcap (RA\_BUFFER=32768)
  - FileStager: background thread 'lcg-cp' from local SE to worker node
- FileStager shows better CPU/Walltime ratio on several sites
- IFIC and Lisbon with 'file' access on Lustre with interesting result
- Data placement on pools sometimes not optimal
- First time NDGF with poor performance, requested too many input files at once

# HAMMERCLOUD - PLOTS

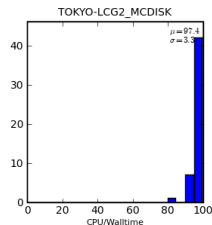


Tokyo, rfio input

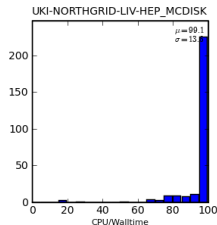


Göttingen

data on 1 pool



Tokyo, FileStager input



Liverpool, old CPUs

Event Rate is important number

# CONCLUSIONS AND SUMMARY

## 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

## Plans:

- Streamline code between the submission tools
- Scheduling: Push and/or Pull model
- Much more HammerCloud tests