

COMPUTING GRID APPLICATIONS FOR THE ATLAS LHC EXPERIMENT AND BEYOND

R.W.L. Jones, for the ATLAS Collaboration and GridPP
1 University of Lancaster, Lancaster, LA1 4YB, England

Key words to describe the work: User Interface, Installation Tools, Data Management, Grid Deployment, Simulation, Monitoring, GridPP.

Key Objectives: Long term, to deliver a worldwide computational Grid for the ATLAS experiment, handling several petabytes of stored data each year. Immediate objectives: Automated code and environment packaging interfaced with an automated push/pull distribution, installation and verification mechanism; GUI and API interfaces to EDG and PPDG/GryPhyn/iVDGL middleware and the experimental software; a Monte Carlo production system based on Grid tools; working intercontinental prototypes, with integrated monitoring, data replication and bookkeeping.

Motivation for the work (problems addressed): At the most conservative estimates, ATLAS will produce over 1Pb of data per year requiring 1-2M SpecInt95 of cpu to process and analyse, and to generate large Monte Carlo datasets. The collaboration is worldwide, and only Grids will allow all collaborators must have access to the full datasets. Atlas must develop an intercontinental distributed computing and data Grid with a user interface to shield the user from the Grid middleware and the distributed nature of the processing; we must develop automated production systems using the Grid tools; and we must provide tools that automatically get the required experimental software and run-time environment to remote sites to avoid the problems of chaotic and multi-site management. Bookkeeping, replication and monitoring are also required.

Even at start-up, when the data rate will be lowest, the ATLAS LHC experiment will be producing over 1Pb of data per year and will require about 1.6M SpecInt95 to process this and the matching Monte Carlo data. The collaboration has members in six out of seven continents, which all require access to the full datasets. Clearly, only a computation and data Grid can provide the required solution.

Generic middleware can solve many of our problems, but there are many aspects where we need to provide our own user layer. In addition, we must test the viability of the middleware on offer, and we can also generate tools that will be of use to others. ATLAS is engaged in a series of Data Challenges of increasing size and complexity over the next five years. These will generate increasingly large Monte Carlo samples for physics and detector studies, which are testing Grid tools as they become available. The same exercises allow us to develop and test our computing model and guide the distribution of experimental resources.

The worldwide nature of the ATLAS collaboration means we have involvement in many Grid projects, and are developing prototype applications within these. ATLAS has an important role in maintaining cohesion between projects, which is essential for our worldwide Grid to succeed.

The original ATLAS computing model was formed when the view of the solution was hierarchical, with many tiers. Our current thinking still sees a distinct

Tier-0 at CERN, where the data are produced. The old 'Tier-1' facilities are now seen as truly regional facilities, which dissolve into a 'cloud' shared by the whole collaboration. In particular, only about 1/3 of the full dataset will reside in any regional facility, allowing multiple copies around the world and saving on storage. As a consequence, the regional facilities must be open to collaborators worldwide. This has cost-sharing implications, but also places requirements on the middleware, which must allow accounting not only by virtual organisation, but also by region, and by group within an experiment. In a similar way, it must be possible to set priorities for groups and activities within a given virtual organisation through the middleware tools and accounting.

The cloud model is likely to continue out to smaller facilities, and we are already seeing distributed facilities at a sub-national level, with ScotGrid in the central belt, and the two large computing surfaces in Liverpool and Lancaster (the latter also hosting a mass storage system); the concept of a centralised facility in local regions is already looking outdated.

ATLAS has identified several key components for its Grid system to work, which must be developed in-house. Where possible, collaboration is sought with other experiments, and this has been especially successful with LHCb, with the help of GridPP.

A clear need is for a User Interface that is aware of the Grid middleware and also of the experimental

software framework, and which hides the complexities from the user. It must also be able to work in a non-Grid environment. ATLAS and LHCb have defined a project called GANGA to provide such an interface. It builds on an existing prototype developed within ATLAS called GRAPPA, which uses an XCAT web portal and Active Notebook technologies, and on existed scripting interfaces. GANGA must be a robust and adaptable replacement, and so will be component-based. One possibility under investigation is to link components using a PYTHON bus. Whichever approach is the final solution, standard interfaces to external middleware are essential. The main focus of the current development is a clear capture of the software services to be captured.

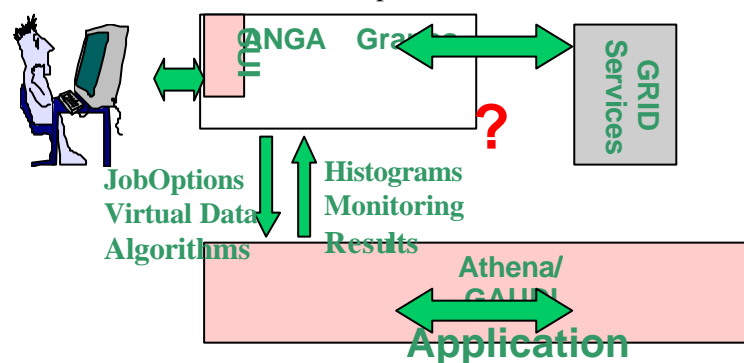


Figure 1: GANGA is an interface between the User, the Grid services and the experimental framework. There is still residual direct interaction between the framework and some Grid services.

Another clear common requirement is the need for tools to install experimental software and run-time environments automatically on remote fabrics that declare themselves available. Relying on by-hand installation at each site is not scalable and can lead to an incoherent environment. As both ATLAS and LHCb use CMT for the management of their software and runtime environment, CMT is being adapted to automatically produce rpms of packages, and also to expose their external and inter-dependencies. These are then presented in a format understood by PACMAN, which has both push and pull methods for software installations from defined caches. PACMAN can also test if installations are already present; work is ongoing on dealing with version policies and advertising software installations via MDS.

While file replication tools are under development elsewhere, ATLAS has an immediate need for such services. It has therefore developed MAGDA, a data

and replica management service based on MySQL servers. This is already in use in the Data Challenges, and will incorporate tools like GDMP as they become mature. This tool is also interfaced to a MySQL-based metadata service, AMI, and also to Virtual Data Catalogues.

All of these tools are being proven on various test beds. ATLAS has two major test beds for Grid middleware, the EDG test bed and the US ATLAS test bed. Interoperability has been achieved, but it still requires by-hand intervention to overcome certification issues. However, applications and middleware have been transferred between both domains. In addition, ATLAS has a major presence in smaller Grid test beds, such as NorduGrid, which are also developing middleware and architectural designs. Monitoring tools are important to understand the system performance, and ATLAS uses both EDG tools and also the US GridView tool.

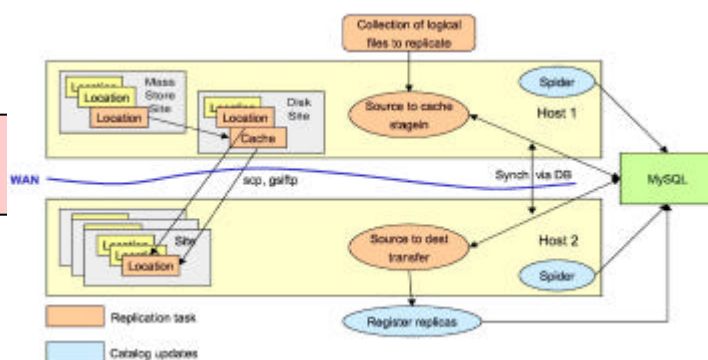


Figure 2: The MAGDA Grid Data Manager

The Presentation will describe all of the above features: the User Interface developments, the Installation Tools requirements and prototype solutions, the bookkeeping and data management prototypes, the monitoring and the deployment on the major test beds. The emphasis will be on those activities with a major contribution from GridPP, but the intercontinental nature of the collaboration will also be stressed.

- GridPP: <http://www.gridpp.ac.uk/>
- GRAPPA: <http://uatlas.physics.indiana.edu/grappa>
- CMT: <http://www.cmts.org/>
- Pacman: <http://physics.bu.edu/~youssef/pacman/>
- MAGDA: <http://atlassw1.phy.bnl.gov/magda/info>