Performance of the NorduGrid ARC and the Dulcinea Executor in ATLAS Data Challenge 2

Robert Sturrock, University of Melbourne, Australia
Henrik Jensen, Josva Kleist, Ålborg University, Denmark
John Renner Hansen, Jakob Langgaard Nielsen, Anders Wäänänen, NBI, Copenhagen, Denmark
Daniel Kalici, University of Southern Denmark, Odense, Denmark
Arto Teräs, CSC, Helsinki, Finland
Helmut Heller, Leibniz-Rechenzentrum, Munich, Germany
John Kennedy, Günter Duckeck, Ludwig-Maximilians-Universität, Munich, Germany
Aleksandr Konstantinov, University of Oslo, Norway and Vilnius University, Lithuania
Jan-Frode Myklebust, University of Bergen, Norway
Farid Ould-Saada, Katarzyna Pajchel, Alex Read, Haakon Riiser, University of Oslo, Norway
Morten Hanshaugen, Sturle Sunde, USIT, University of Oslo, Norway
Andrej Filipčič, Matevž Tadel, Jožef Stefan Institute, Ljubljana, Slovenia
Leif Nixon, NSC, Linköping University, Sweden
Paula Eerola, Balázs Kónya, Oxana Smirnova, Lund University, Sweden
Jonas Lindemann, LUNARC, Lund University, Sweden
Lars Malinowsky, Nils Smeds, PDC, KTH, Stockholm, Sweden
Åke Sandgren, Mattias Wadenstein, HPC2N, Umeå University, Sweden
Tord Ekelöf, Uppsala University, Sweden
Christian Häberli, University of Bern, Switzerland
Mattias Ellert*, CERN, Geneva, Switzerland

Abstract

This talk describes the various stages of ATLAS Data Challenge 2 (DC2) in what concerns usage of resources deployed via NorduGrid's Advanced Resource Connector (ARC). It also describes the integration of these resources with the ATLAS production system using the Dulcinea executor.

ATLAS Data Challenge 2 (DC2), run in 2004, was designed to be a step forward in the distributed data processing. In particular, much coordination of task assignment to resources was planned to be delegated to Grid in its different flavours. An automatic production management system was designed, to direct the tasks to Grids and conventional resources.

The Dulcinea executor is a part of this system that provides interface to the information system and resource brokering capabilities of the ARC middleware. The executor translates the job definitions received from the supervisor to the extended resource specification language (XRSL) used by the ARC middleware. It also takes advantage of the ARC middleware's built-in support for the Globus Replica Location Server (RLS) for file registration and lookup.

NorduGrid's ARC has been deployed on many ATLAS-dedicated resources across the world in order to enable effective participation in ATLAS DC2. This was the first attempt to harness large amounts of strongly heterogeneous resources in various countries for a single collaborative exercise using Grid tools. This talk addresses various issues that arose during different stages of DC2 in this environment: preparation, such as ATLAS software installation; deployment of the middleware; and processing. The results and lessons are summarised as well.

* mattias.ellert@tsl.uu.se

ATLAS DATA CHALLENGE 2

The ATLAS [1] collaboration, in preparation for the startup date of its experiment at the Large Hadron Collider (LHC) [2] at CERN in 2007, is conducting a series of so called Data Challenges. These exercises are intended to verify the capability of the collaboration to handle the computational effort needed to analyse the data collected by its experiment. The Data Challenges have been designed to gradually increase in size to eventually reach a complexity equal to the computational effort foreseen to be needed when the experiment starts acquiring data. During 2004 the second of these exercises, ATLAS Data Challenge 2 (DC2), was carried out.

THE ATLAS PRODUCTION SYSTEM

In order to handle the task of ATLAS DC2 an automated production system [3] was designed. This production system consists of several parts: a database for defining and keeping track of the computing tasks to be done, the Don Quijote data management system [4] for handling the input and output data of the computations, the Windmill supervisor program that was in charge of distributing the tasks between various computing resources and a set of executors responsible for carrying out the tasks. By writing various executor the supervisor could be presented with a common interface to each type of computing resource available to the ATLAS collaboration. Executors were written to handle resources on the LHC Computing Grid [5], Grid 3 [6, 7], NorduGrid's ARC and various legacy batch systems [8]. During ATLAS DC2 the three Grid flavours carried out about one third of the total computational task each. The subject of this paper is the executor written for NorduGrid's ARC, called Dulcinea, and the part of DC2 that was carried out with it.

NORDUGRID'S ARC

The Advanced Resource Connector (ARC) [9] is a Grid middleware developed by the NorduGrid collaboration [10]. It is built on top of the libraries provided by the Globus toolkit [11]. An overview of the ARC middleware components can be found in Figure 1.

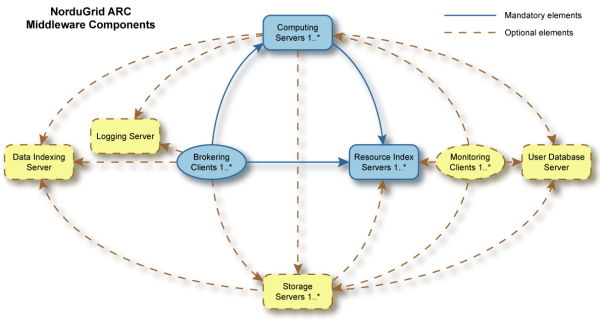

Figure 1: An overview of the ARC architecture showing the different components of the middleware.

NorduGrid's ARC has been deployed at a number of computing resources around the world (see Figure 2). These resources are running various Linux distributions and use several different local resource management systems (LRMS). Although various flavours of PBS are most common, there are sites running SGE, Easy and Condor as well. Using different LRMS specific information providers the different sites can present the information about their available resources in a uniform way in ARC's information system [12]. This information system is based on the Globus Monitoring and Discovery Service (MDS). The information provided is used by the brokering algorithm in the ARC user interface API [13] to find suitable resources

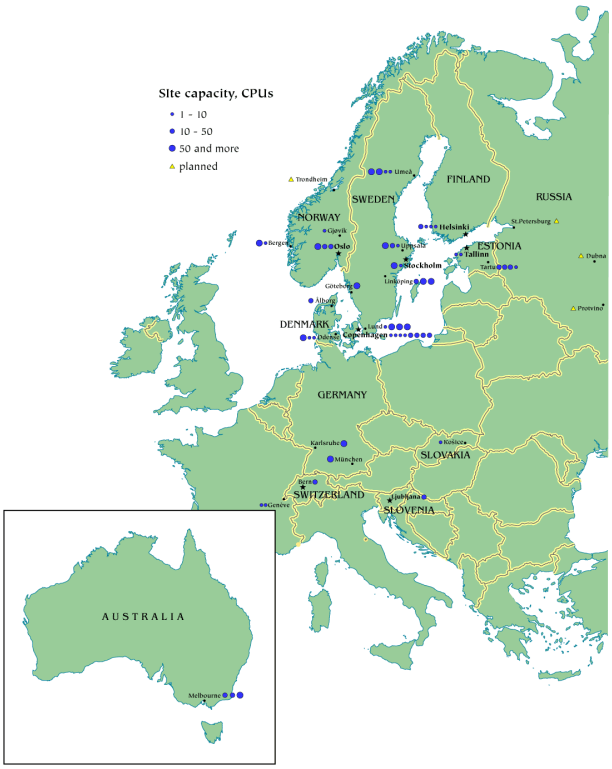

Figure 2: This map shows the geographical location of the ARC enabled sites.

for the tasks to be performed, as well as by various monitoring tools like the NorduGrid Grid Monitor (see Figure 3).

NorduGrid's ARC is fully integrated with the Globus Replica Location Service (RLS) [14]. This allows jobs sent to an ARC enabled site to specify input file locations as an RLS catalogue entry instead of a physical file location. A job can also, if desired, automatically register created output files in an RLS catalogue.

THE DULCINEA EXECUTOR

The ATLAS production system executor for NorduGrid's ARC, Dulcinea, was implemented as a C++ shared library. This shared library was then imported into the production system's python framework. The executor calls the ARC user interface API and the Globus RLS API to perform its tasks.

The job description received from the Windmill supervisor in form of an XML message was translated by the Dulcinea executor into an extended resource specification language (XRSL) [15] job description. This job description was then sent to one of the ARC enabled sites, selecting a suitable site using the resource brokering capabilities of the ARC user interface API. In the brokering, among other things, the availability of free CPUs and the amount of data needed to be staged in on each site to perform a specific task is taken into account.

The look-up of input data files in the RLS catalogue and

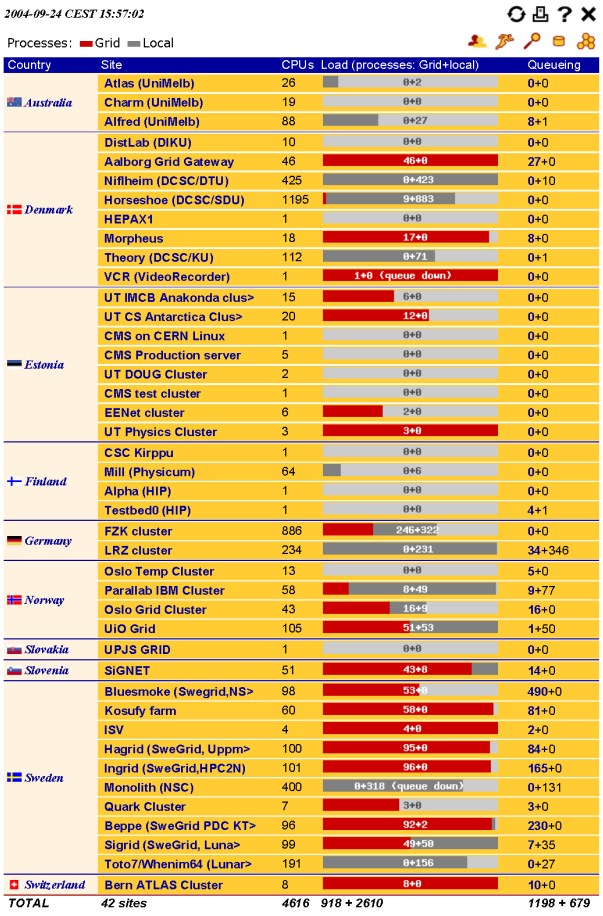

Figure 3: Snapshot of the NorduGrid Monitor at http://www.nordugrid.org/monitor/ showing the ARC enabled sites.

the stage-in of these files to the site is done automatically by the ARC Grid Manager [16]. The same is true for stage-out of output data to a storage element and the registration of these files in the RLS catalogue. The Dulcinea executor only has to add the additional RLS attributes needed for the Don Quijote data management system to the existing file registrations.

Also in other respects the Dulcinea executor takes advantage of the capabilities of the ARC middleware. The executor does not have to keep any local information about the jobs it is handling, but can rely on the job information provided by the Grid information system.

RUNNING DC2 USING DULCINEA

Due to the diversity of the ARC enabled resources in terms of which Linux distribution was installed on the various sites, the ATLAS software release was recompiled for various distributions. The ATLAS software release was then installed on the sites who wanted to participate in the ATLAS DC2 exercise. After testing the installation at each site the ATLAS runtime environment (RTE) was published in the site's Grid information system. By letting the Dulcinea executor request the RTE in its XRSL job description only sites with the proper software installation was considered in the brokering.

Of the computing resources connected through NorduGrid's ARC 22 sites had installed the ATLAS software needed to run DC2 (see Table 1). In total these sites had approximately 3,000 CPUs, of which approximately 700 were dedicated to ATLAS DC2.

During the first weeks of ATLAS DC2 we had severe problems due to the Globus RLS server hanging. This caused jobs to fail either at stage-in, because the location of input files could not be looked up in the catalogue, or at stage-out because the location of the output files could not be registered. Of these the first kind of failure was not that serious, since no CPU time was lost for these jobs, while the second kind was serious, because the CPU time for those jobs had been wasted. A few weeks into ATLAS DC2 a patch from the developers of the RLS was available that solved this problem.

Apart from this, most failures were due to site specific hardware problems. A storage element going down meant that all jobs requiring an input file stored on this particular storage element would fail at stage-in. This kind of failure did however not cause any wasted CPU time, since the job was not able to start. The Windmill supervisor automatically retried the job at a later time. If a site went down due to disk server, network or cooling problems the jobs run-

Table 1: In total 22 sites in 7 countries participated in DC2 through ARC. Together these sites had approximately 3,000 CPUs of which approximately 700 were dedicated to DC2.

	site	CPUs	DC2
AU	atlas.hpc.unimelb.au	26	30%
	charm.hpc.unimelb.au	19	100%
	genghis.hpc.unimelb.au	88	20%
CH	lheppc10.unibe.ch	8	100%
DE	atlas.fzk.de	884	5%
	lxsrv9.lrz-muenchen.de	234	5%
DK	benedict.aau.dk	46	90%
	morpheus.dcgc.dk	18	100%
	lscf.nbi.dk	32	50%
	fe10.dcsc.sdu.dk	1,200	1%
NO	grid.fi.uib.no	4	100%
	fire.ii.uib.no	58	50%
	grid.uio.no	40	100%
	hypatia.uio.no	100	60%
SE	hive.unicc.chalmers.se	100	30%
	sg-access.pdc.kth.se	100	30%
	bluesmoke.nsc.liu.se	100	30%
	farm.hep.lu.se	60	60%
	sigrid.lunarc.lu.se	100	30%
	ingrid.hpc2n.umu.se	100	30%
	hagrid.it.uu.se	100	30%
SI	brenta.ijs.si	51	100%

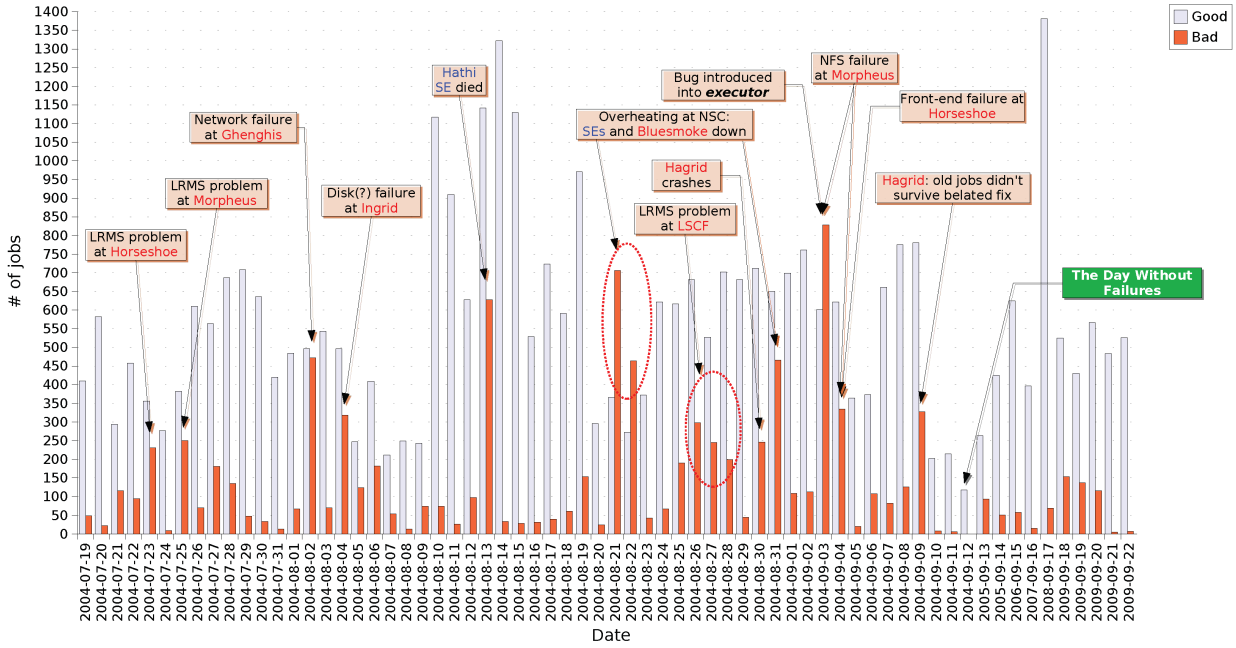

Figure 4: Distribution of successful and failed ATLAS DC2 jobs on the ARC enabled resources over time. Peaks in the failure distribution can often be traced to site specific problems as indicated in the graph.

ning on that site at the time were lost. Jobs queued but not yet running at the site may or may not survive depending on the circumstances. A summary of the analysis of the failures can be found in Figure 4.

SUMMARY

The ARC middleware and the Dulcinea executor provided a stable service for ATLAS DC2. More than 20 sites in 7 countries operated as a single resource and contributed approximately 30% of the total ATLAS DC2 production. Most of the resources were not 100% dedicated to the DC2 production.

The amount of middleware related problems were negligible, except for the initial instability of the RLS server. Most job failures were due to site specific hardware problems.

ACKNOWLEDGEMENTS

We thank the owners of the computing resources used during ATLAS DC2 for making them available to us for this exercise.

REFERENCES

[1] http://www.cern.ch/atlas/

[2] http://www.cern.ch/lhc/

[3] Luc Goossens *et al.*, 'ATLAS Production System in ATLAS Data Challenge 2', CHEP 2004, Interlaken, contribution no. 501

[4] Miguel Branco, 'Don Quijote — Data Management for the ATLAS Automatic Production System', CHEP 2004, Interlaken, contribution no. 142

[5] Davide Rebatto, 'The LCG-2 Executor for the ATLAS DC2 Production System', CHEP 2004, Interlaken, contribution no. 364

[6] Robert Gardner *et al.*, 'ATLAS Data Challenge Production on Grid3', CHEP 2004, Interlaken, contribution no. 503

[7] Xin Zhao *et al.*, 'Experience with Deployment and Operation of the ATLAS Production System and the Grid3+ Infrastructure at Brookhaven National Lab', CHEP 2004, Interlaken, contribution no. 185

[8] John Kennedy, 'The role of legacy services within ATLAS DC2', CHEP 2004, Interlaken, contribution no. 234

[9] Oxana Smirnova, 'The NorduGrid/ARC User Guide, Advanced Resource Connector (ARC) usage manual', http://www.nordugrid.org/documents/userguide.pdf

[10] http://www.nordugrid.org/

[11] http://www.globus.org/

[12] Balázs Kónya, 'The NorduGrid Information System', http://www.nordugrid.org/documents/ng-infosys.pdf

[13] Mattias Ellert, 'The NorduGrid Toolkit User Interface', http://www.nordugrid.org/documents/NorduGrid-UI.pdf

[14] Jakob Nielsen *et al.*, 'Experiences with Data Indexing services supported by the NorduGrid middleware', CHEP 2004, Interlaken, contribution no. 253

[15] Oxana Smirnova, 'Extended Resource Specification Language', http://www.nordugrid.org/documents/xrsl.pdf

[16] Aleksandr Konstantinov, 'The NorduGrid Grid Manager and GridFTP Server: Description and Administrator's Manual', http://www.nordugrid.org/papers.html