

OPEN ACCESS

AGIS: The ATLAS Grid Information System

To cite this article: Alexey Anisenkov *et al* 2012 *J. Phys.: Conf. Ser.* **396** 032006

View the [article online](#) for updates and enhancements.

You may also like

- [ATLAS Distributed Computing Monitoring tools after full 2 years of LHC data taking](#)
Jaroslava Schovancová
- [ATLAS Distributed Computing Monitoring tools during the LHC Run I](#)
J Schovancová, S Campana, A Di Girolamo et al.
- [Evolution of the ATLAS PanDA workload management system for exascale computational science](#)
T Maeno, K De, A Klimentov et al.



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

AGIS: The ATLAS Grid Information System

Alexey Anisenkov¹, Sergey Belov², Alessandro Di Girolamo³, Stavro Gayazov¹,
Alexei Klimentov⁴, Danila Oleynik², Alexander Senchenko¹ on behalf of the
ATLAS Collaboration

¹ Budker Institute of Nuclear Physics, Novosibirsk, Russia

² Joint Institute of Nuclear Research, Laboratory of Information Technology,
Dubna. Russia

³ CERN, ES Department, CH-1211 Geneva 23, Switzerland

⁴ Brookhaven National Laboratory, Upton, NY 11973, USA

Email: Alexey.Anisenkov@cern.ch

Abstract. ATLAS is a particle physics experiment at the Large Hadron Collider at CERN. The experiment produces petabytes of data annually through simulation production and tens petabytes of data per year from the detector itself. The ATLAS Computing model embraces the Grid paradigm and a high degree of decentralization and computing resources able to meet ATLAS requirements of petabytes scale data operations. In this paper we present ATLAS Grid Information System (AGIS) designed to integrate configuration and status information about resources, services and topology of whole ATLAS Grid needed by ATLAS Distributed Computing applications and services.

1. Introduction

The ATLAS experiment [1] produces petabytes of data per year that needs to be distributed globally to sites worldwide according to the ATLAS computing model [2]. The ATLAS computing model is based on a worldwide Grid computing infrastructure that uses a set of hierarchical tiers. It provides all members of the ATLAS Collaboration speedy access to all reconstructed data for analysis, and appropriate access to raw data for calibration and alignment activities. The root node of the complex tiered topology is the Tier-0, located at CERN itself. After a primary data processing at CERN in a Tier-0 facility, raw data from the detector and reconstructed data products are replicated to 10 Tier-1 centres (geographically spread across Europe, Asia and North America) according to the ATLAS data distribution policy. In addition to the data taken from the detector, simulated data is produced on Grid resources worldwide, and this data must also be replicated. Tier-1 and associated underlying tiers are grouped into a cloud. The ATLAS grid is composed by independent sub-grids:

- European Grid Infrastructure (EGI) [3]
- Open Science Grid (OSG) [4]

These two sub-grids form the Worldwide LHC Computing Grid infrastructure (WLCG). Figure 1 shows a schematic view of ATLAS Grid topology.

This immense volume of data must be handled by the computing with integrated information systems. ATLAS Grid Information System represents a general-purpose storage of different parameters, configuration, static and dynamic data needed to configure and operate the ATLAS Distributed Computing (ADC) [5,6] systems and services.

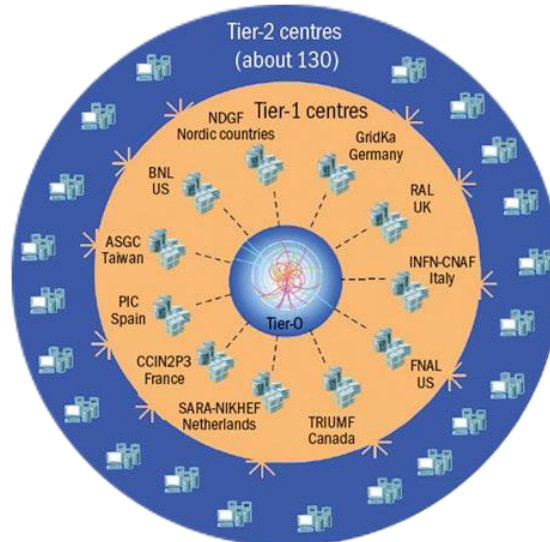


Figure 1. The topology of ATLAS computing grid

2. AGIS Requirements and Motivation

ADC applications and services require the diversity of common information, configurations, parameters and quasi-static data originally stored in different sources. Sometimes such static data are simply hardcoded in application programs or spread over various configuration files.

The difficulty faced by ADC applications is that ATLAS computing uses a variety of Grid infrastructures which have different information services, application interfaces, communication systems and policies. Therefore, each application has to know about the proper information source, data structure formats and application interfaces, security infrastructures and other low-level technical details to retrieve specific data. Moreover, an application has to implement communication logic to retrieve data from external sources that produces a lot of code duplications.

To satisfy ADC requirements a central information system should be designed and implemented to have a coherent approach to store and expose data. Such a central place should be a database based system with good scalability and convenient data access interfaces.

A centralized information system helps to solve information duplication issues and inconsistencies between data stored in many external databases, configuration files and even hard-coded in applications. A common information system also helps to solve synchronization issues, to reduce code duplication and simplify application code logic. AGIS is designed to cover all these requirements and ATLAS needs. By introducing its own caching mechanism AGIS helps to improve performance in data management.

Being an intermediate middleware system between a client and external information sources, AGIS automatically collects and keeps data up to date, caching information required by and specific for ATLAS, removing the source as a direct dependency for clients but without duplicating the source information system itself.

Also, using AGIS as one data repository will improve system maintainability and allow adding new classes of information centrally.

To properly operate the ATLAS computing infrastructure is crucial to centrally collect and organize various parameters like sites availability information, general ATLAS topology, site downtime information, resource description, data access permissions and many others. AGIS is

designed to integrate information about resources, services and topology of the ATLAS grid infrastructure from various independent sources including Berkeley Database Information Index (BDII), Grid Configuration Database (GOCDB) [7], the ATLAS data management system [8] and the ATLAS PanDA workload management system [9] and others.

3. AGIS Architecture

AGIS Architecture is based on the classic client-server computing model. Oracle Database Management System is chosen as a database backend. Since the system provides various interfaces like Application Programming Interface (API), Web interface and Command Line Interface to retrieve and manage data, no direct access to the database from the clients is required.

AGIS uses Django framework [10] as a high-level web application framework written in Python. Object Relation Mapping technique built in Django framework allows to operate and to access data in terms of high level models avoiding direct dependence of relational database system used. Figure 2 shows schematic view of AGIS architecture.

There is no full duplication of external information sources, since AGIS extracts only the information needed for ATLAS and stores data objects in a way convenient for ATLAS, and introduces additional object relations required by ADC applications.

All interactions with various information services are hidden. Synchronization of AGIS content with external sources is performed by agents which periodically communicate with sources via standard interfaces and update AGIS database content. For some types of information AGIS itself is the primary source.

The clients are able to update information stored in AGIS through the API. A python API and command line tools further help the end users and developers to use the system conveniently.

The client API allows users to retrieve data in XML or in JSON format, for instance all ATLAS topology can be exported in an XML feed.

WebUI is designed to be able to quickly view and change most of the information stored in the AGIS. The interface is focused on the use by ADC experts and does not require any special software development knowledge.

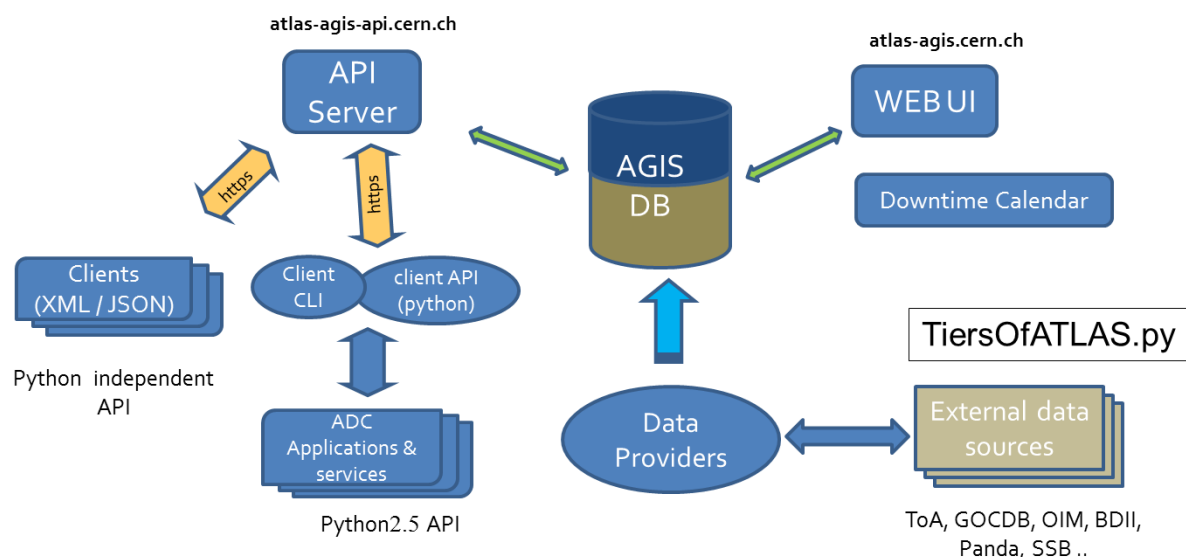


Figure 2. Schematic view of AGIS client-server architecture

To automatically populate the database a set of cron jobs runs on the main AGIS server. The Figure 3 illustrates typical data flow directions within the system while collecting data from external sources. One of the operation requirements recently addressed to the information system is the possibility to

track the changes and the functionality to fully reproduce the database content from previously saved state. To meet this demand the data populating approach has been improved by introducing intermediate level in the data collecting procedure. New approach assumes that data collected from external source is saved into a temporary storage before actually being inserted into the database. It allows to overwrite specific values and to control the data coming in AGIS. As the data interchange format the JSON structure is chosen.

The Figure 4 shows simplified class diagram of AGIS entities implemented.

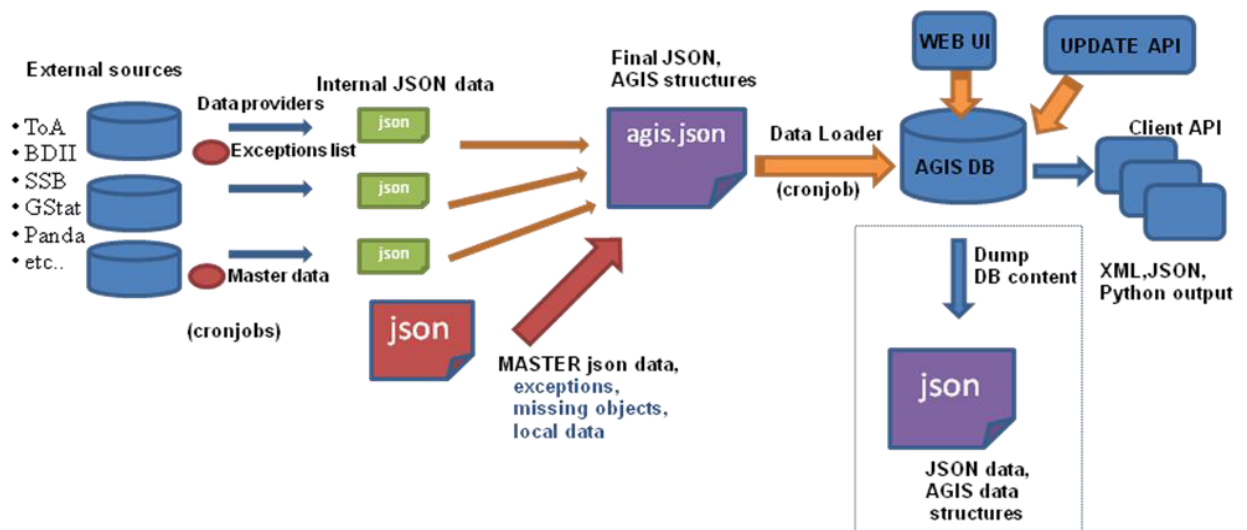


Figure 3. Schematic view of data flow in AGIS

4. Stored Information

There are two types of information stored in AGIS:

- data retrieved from external sources (TiersOfATLAS file, GOCDB database, BDII, myOSG [11], PanDA databases, WLCG Resource, Balance and Usage (REBUS) source, etc),
- data managed within AGIS.

In current implementation, AGIS stores all information concerning ATLAS topology: clouds, regional centers, sites specifics, such as geography, time zone, geo coordinates, etc. It also stores resources and services information including the description of Local File Catalog, File Transfer Server, Computing Element, Storage Resource Manager, Squid, Frontier services and others.

Key examples of information stored in AGIS are:

- ATLAS topology (clouds, centres, sites and sites specifics),
- site's resources and services information, status and its description,
- site information and configuration,
- data replication sharing and pairing policy,
- list of activities and its properties (Functional Tests, data distribution, etc.),
- global configuration parameters needed by ADC applications,
- user related information (privileges, roles, account info) for ADC applications ,
- downtime information about ATLAS sites, list of affected services,
- site blacklisting data.

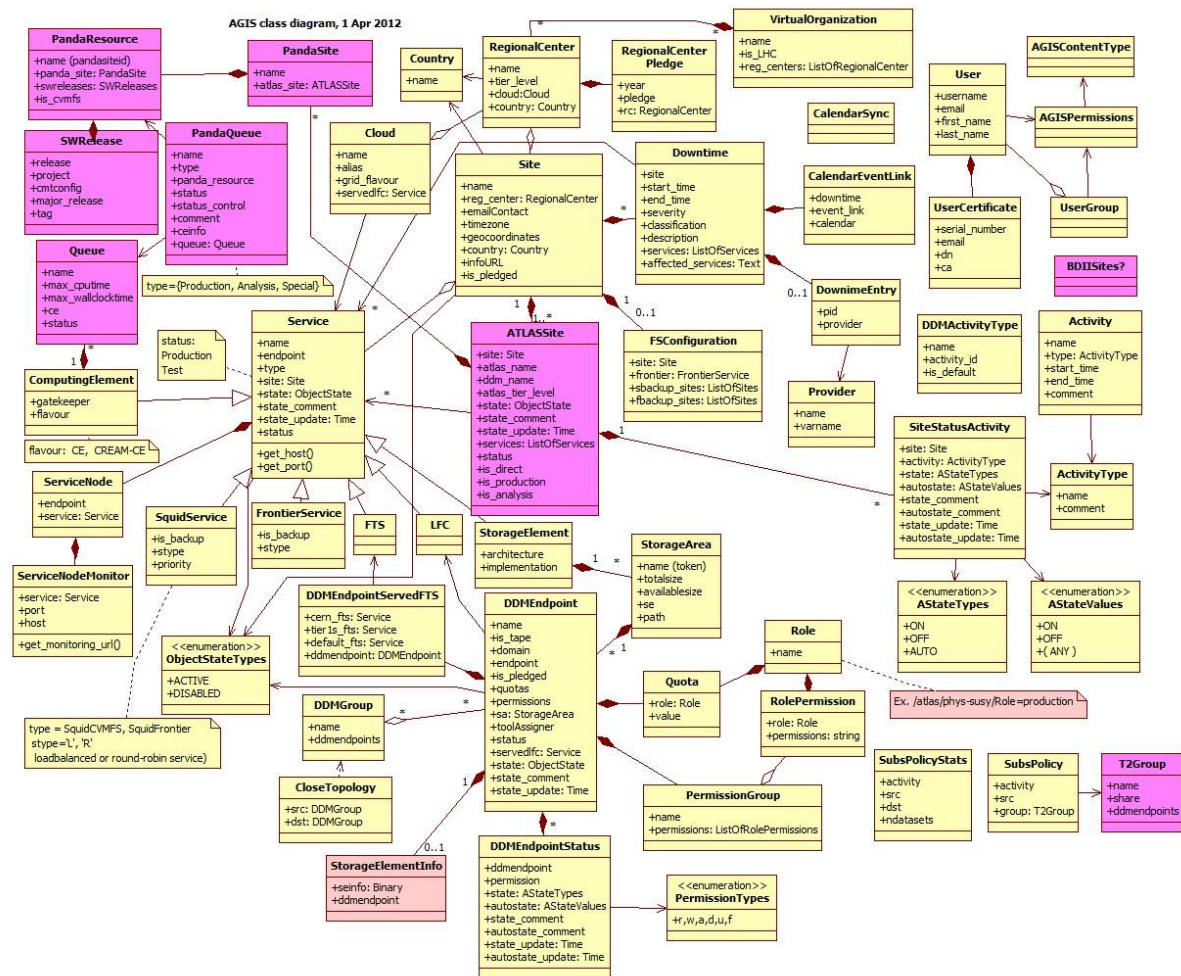


Figure 4. AGIS class diagram

5. Conclusion

AGIS has been developed to provide, in a single portal, the topology and resources information to configure the ATLAS computing applications. AGIS can be also used by interactive users which can have a global picture of the ATLAS topology and resources through the WEB. AGIS is currently in production and it is used by various ATLAS applications and services.

At the moment AGIS project is in stage of extending its functionality and improving reliability. Information stored in TiersOfATLAS file is migrated to AGIS database.

Many AGIS services, in particular the API functionality exposing ATLAS data in JSON format are starting to be actively used in the production applications. Web interfaces such as a site downtime calendar and ATLAS topology viewers are widely used by shifters and data distribution experts.

AGIS is continuously extending data structures to follow new demands of ADC needs. Recently implemented functionality to handle Frontier and Squid services data, to manage software installations data are the examples of last fulfilled requests from ADC users.

New external information sources will be supported in the near future. So, more ADC applications are expected to use AGIS as the main information system.

AGIS design and basic principles included into the architecture allows to use its core part for several HEP experiments. Russian Federation sites are considering AGIS as an information system to be used for RF cloud after two new Tier-1 facilities will be commissioned.

References

- [1] G. Aad et al., “The ATLAS Experiment at the CERN Large Hadron Collider,” JINST 3 (2008) S08003.
- [2] The ATLAS Computing Model, Adams, D ; Barberis, D ; Bee, C P ; Hawkings, R ; Jarp, S ; Jones, R ; Malon, D ; Poggioli, L ; Poulard, G ; Quarrie, D et al. - ATLAS Collaboration., ATL-SOFT-2004-007; ATL-COM-SOFT-2004-009; CERN-ATL-COM-SOFT-2004-009; CERN-LHCC-2004-037-G-085.- Geneva : CERN, 2005 - 37
- [3] European Grid Infrastructure (EGI), <http://www.egi.eu/>
- [4] Open Science Grid (OSG) Web Page, <http://www.opensciencegrid.org/>
- [5] ATLAS Distributed Computing, web page, <https://twiki.cern.ch/twiki/bin/view/Atlas/AtlasDistributedComputing>
- [6] A. Anisyonkov, A. Klimentov, D. Krivashin, M. Titov. ATLAS Distributed Computing, XXII International Symposium on Nuclear Electronics & Computing (NEC 2009) (Institute for Nuclear Research and Nuclear Energy, Varna, Bulgaria, 2009), pp. 4. September 7-14, 2009; Proceedings of the Symposium; E10, 11-2010-22, Dubna, 2010 - p. 45-49
- [7] Grid Configuration Database web site, <https://wiki.egi.eu/wiki/GOCDB>
- [8] M. Branco et al., “Managing ATLAS data on a petabyte-scale with DQ2,” J. Phys. Conf. Ser. 119 (2008) 062017
- [9] T. Maeno, “PanDA: Distributed production and distributed analysis system for ATLAS,” J. Phys. Conf. Ser. 119 (2008) 062036.
- [10] Django project web site, <http://www.djangoproject.com>
- [11] MyOSG project web site, <http://myosg.grid.iu.edu/about>