GridICE: Requirements, Architecture and Experience of a Monitoring Tool for Grid Systems

S. Andreozzi[*], E. Fattibene, G. Misurelli, G. L. Rubini, INFN-CNAF, Bologna, Italy
C. Aiftimiei[†], S. Fantinel[‡], INFN, Padova, Italy
G.Cuscela, G. Donvito, A. Pierro, INFN, Bari, Italy
N. De Bortoli, G. Tortone, INFN, Napoli, Italy

Abstract

Monitoring a Grid system is an essential activity for the support of operations and management of a Grid system. It must deal with the dynamics, diversity and geographical distribution of the resources available to virtual organizations, and the various levels of abstraction for modeling them. This paper presents the requirements and architecture of GridICE, a monitoring service for Grid systems. The experience in the context of production services is also described.

INTRODUCTION

Grid computing is concerned with the virtualization, integration and management of services and resources in a distributed, heterogeneous environment that supports collections of users and resources across traditional administrative and organizational domains.

One aspect of particular importance is Grid monitoring, that is the activity of measuring significant Grid resource-related parameters in order to analyze usage, behavior and performance of a Grid system. This activity can also help in the detection of fault situations, contract violations and user-defined events.

In [4], two main types of monitoring are identified: infrastructure monitoring and application monitoring. The former aims at collecting information about Grid resources and possibly maintain the history of observations in order to perform retrospective analysis. The latter aims at enabling to observe a particular execution of an application; the collected data can be useful for application development support or to visualize the behavior when running in a machine with no login right access as it is in Grid systems.

In this paper, we focus on the first category of monitoring systems, that is infrastructure monitoring. Firstly, we define terms and concepts relevant to our work. Then, a number of requirements are identified while exploring three real-life scenarios. After that, GridICE is presented as a monitoring service architecture satisfying these requirements. Particular attention has been paid to the different categories of consumers of monitoring information. Aggregation dimensions such as the Grid operators, Virtual Organization (VO) managers and site administrators have been considered in the design of the proposed system. Next, implementation details and experience results are reported. Finally, conclusions are presented together with directions for future work.

TERMS AND CONCEPTS

In this section, we provide the definition of terms and concepts that are related to the monitoring activity. In the context of Grid systems, the most referenced document is [20]. A later document provided extensions to the defined concepts together with a taxonomy of Grid monitoring systems [22]. It is worth to consider also the organization of concepts in other disciplines like the Sensor Web Enablement (SWE) activity [5].

In the context of Grid monitoring, we propose the following terms. An 'entity' is any networked and useful resources having a considerable lifetime (e.g. processors, memories, disk capacity, etc.). An 'attribute' is a characteristic of an entity. A 'measurement' is an instance of a procedure to assign numbers or other symbols to phenomena in such a way that relationships of the numbers or symbols reflect relationships of the attributes of the phenomena being observed. A measurement frequently involves an instrument or sensor, moreover it effectively binds a value to a time, location, and to the instrument or procedure used [5]. A 'measurement value' is an estimate of a value describing a natural phenomenon, which is characterized by its observable and may include other properties such as quality measures. A 'measurement unit' is a particular quantity, defined and adopted by convention, with which other quantities of the same kind are compared in order to express their magnitude relative to that quantity. A 'sensor' is a process monitoring an entity and generating observations. Sensors can be categorized in 'active sensors' (they interact directly with the entity which attribute) and 'passive sensors' (they perform the measurement without interacting with the entity, e.g., by reading log files). They can be also categorized in 'intrusive sensors' (they can sensitively affect the performance of the entity being monitored during the measurement process) or 'non-intrusive sensors' (run of the sensor does not affect the target entity).

In the monitoring activity of distributes systems, we can identify four main phases [22]: (1) generation, that is sensors enquiring entities and encoding the measurement values according to a schema; (2) distribution, that is transmission of the measurement values from the source to any interested parties; (3) presentation, that is the processing and abstraction of the received measurement values in order to enable the consumer to draw conclusions about the

[*] Contact author: sergio.andreozzi@cnaf.infn.it

[†] on leave from NIPNE-HH, Romania

[‡] also affiliated to INFN-LNL, Legnaro, Italy

operation of the monitored system; (4) processing, that is filtering or aggregation of the measurement values according to some predefined criteria; this can be performed during the whole monitoring activity.

REQUIREMENTS

There are different categories of users that can be interested in monitoring information of a Grid system. In this work, we mainly focus on Grid operators, Virtual Organization managers and site administrators. An analysis of their requirements shows that there are similarities among them so that they can benefit from an integrated tool offering different views targeted at their specific needs.

A first common aspect to the different users is the set of measurements to be performed. Typically, there is a wide number of base measurements that are of interest for all parties, while a small number is specific to them. What makes the difference is the aggregation criteria required to present the monitoring information. This aspect is intrinsic to the multidimensional nature of monitoring data. Example of aggregation dimensions identified in GridICE are: the physical dimension referring to geographical location of resources, the Virtual Organization (VO) dimension, the time dimension and the resource identifier dimension.

As an example, considering the entity 'host' and the measurement 'number of started processes in down state', the Grid operator can be interested in accessing the sum of the measurement values for all the core machines (e.g., workload manager, computing element, storage element) in the whole infrastructure, while the Virtual Organization manager can be interested in the sum of the measurement values for all the core machines that are authorized to the VO members. Finally, the site administrator can be interested in accessing the sum of the measurement values for all machines part of its site.

Another aspect that is common to all the consumers is being able to start from summary views and to drill down to details. This feature can enable to verify the composition of virtual pools starting from the aggregated view and focusing on the resource provided by each site. Another advantage is the ability to sketch the sources of problems.

A deeper analysis on the specific needs for each category of users was given in [1]. We summarize the set of identified requirements as follows: a Grid monitoring service should be able to (1) dynamically partition resources and service usage using three criteria: site ownership, operations domain, and virtual organization accessibility; (2) collect data in order to enable retrospective analysis; (3) deal with a large volume of data by carefully introducing reduction mechanisms; (4) collect both fine-grained and coarse-grained monitoring data; (5) help to detect fault situations and possibly prevent them; (6) provide general visualization and analysis functionalities; (7) rely on a common information model of the Grid resources; (8) adopt interfaces and protocols that are standard within the Grid community; (9) integrate with local monitoring systems, when available; and (10) track which machines are running the VO applications, the status and behavior of each machine, and the behavior of the software.

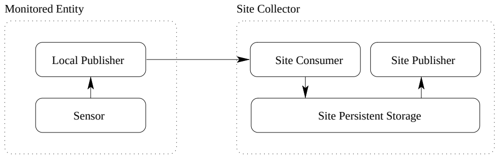

Figure 1: Components deployed in each administrative site

ARCHITECTURE

Following the taxonomy of Grid monitoring tools described in [22], GridICE can be considered as a second level monitoring system with a centralized republisher. This category of systems is characterized by the following components: (1) sensors performing the measurement process; (2) producers offering the measurements values by means of a program interface; (3) republisher consuming information from publishers and reorganizing them for being offered through a potentially different program interface; (4) consumers reading the monitoring information. These four logical components must be appropriately designed and integrated with other functionalities needed to deal with the dynamics, geographically-distribution and multi-institutional nature of a Grid system.

In Figure 1, we can see the logical components that are deployed in an administrative site. For each 'monitored entity', a set of 'sensors' must be configured and installed. In our context, sensors are programs that perform observations to compute a certain measurement value according to a predefined measurement schema [16]. The schema defined in GridICE is an extension of the GLUE Schema [2] where new classes and attributes have been defined for a more complete host-level characterization, Grid jobs related attributes and summary info for batch systems (e.g., number of total slots, number of worker nodes that are down).

A 'local producer' component takes care of exposing the results of the measurement process performed by the sensors. In each site, a 'site collector' performs the aggregation of the site monitoring data in a 'site persistent storage' that can act as a temporary cache or long-term repository for the monitoring data of the whole site. At this stage, the measurement values can be processed and transformed in order to offer new measurements by means of the 'site publisher'.

In Figure 2, we can see the republisher component. This is complex and performs several tasks. First of all, by means of a 'discovery' process, new sources of monitoring data are detected and a 'scheduler' component is configured in order to schedule 'consumer' processes that read the data from the 'site producers' and store them into the 'persistent storage'. The discovery process is necessary due

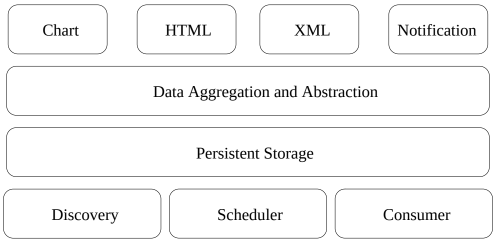

Figure 2: Central republisher architecture

to the dynamic and distributed nature of a Grid. It interacts with the Grid Information Service (GIS) to perform its task. The monitoring data inserted into the persistent storage can be processed with data reduction filters in order to limit the grown of the storage. The 'data aggregation and abstraction' component enables to perform aggregation and filtering over monitored data concerning the whole Grid system. Finally, there is a number of 'producer' components provided in different flavors, to deal with the diversity of final consumers (humans or applications), the delivery pattern (push or pull). Both the 'chart' and the 'HTML' producers are pull-based and transform the monitoring data in charts or HTML pages respectively. They are meant to be consumed by humans by means of browser programs. Also the 'Notification' is meant for humans even though a push-based delivery pattern is envisioned using publish/subscribe mechanism [3]. The 'XML' producer is a pull-based mechanism to offer the monitoring information to external applications.

In summary, as regards the distribution of monitoring data, the GridICE architecture can be considered a 2-level hierarchical model: the intra-site level is within the domain of an administrative site and aims at collecting the monitoring data at a single logical repository; the inter-site level is across sites and enables the Grid-wide access to the site repository. The former is typically performed by a fabric monitoring service, while the latter is performed via the Grid Information Service. In this sense, the two levels are totally decoupled and different fabric monitoring services can be adapted to publish monitoring data to GridICE, thought the proposed default solution is the CERN Lemon tool [14].

IMPLEMENTATION

The implementation status of GridICE is mature since it is included in production Grid systems since 2003 (i.e., LCG [15]). The initial design and implementation started in the European DataTAG project [10], while redesign for better usability, stability and reliability has been carried out during the EGEE project [7].

Considering the metering and first level of distribution components (see Figure 1), we have to discern among the GLUE Schema based measures and the extensions. Considering the former, sensors and producer were already available in the target Grid middleware (LCG). They consist of information providers containing the metering part and the transformation into the LDIF format [21], suitable for the adopted Grid Information Service (Globus Toolkit MDS 2.x [11]). Considering the latter, the adoption of the Lemon fabric monitoring tool provides the framework for local transportation (local producer, local consumer, local persistent storage). By means of a transformation adapter, data stored in the local repository are translated into LDIF format and injected in the GIS.

The second level of distribution is performed by adopting the Grid Information Service (GIS). The monitoring data aggregated at the site collector are transformed in order to be published by means the GIS. This choice implies the decoupling of the intra-site technology dependencies from the inter-site, thus enabling the republisher component to be independent from the internal site changes. The current Grid Information Service in the target production Grid is the Globus MDS 2.x, that is an LDAP-based technology. In the recent evolution of Grid middleware, the de-facto standard GIS is been replaced with different proposals, thus breaking a uniform interface. This poses new requirements to the republisher component, that is new adapters must be developed to deal with different interfaces and protocols for accessing monitoring data.

In the republisher component, the 'discovery' is a set of programs developed to detect and identify new sources of information. The 'consumer' is a program that can be invoked for each source of information identified during the discovery process. For each invocation, the published data is queried by means of the LDAP protocol and query language, the content is compared with the result of previous observations and data reduction mechanisms are applied. Finally, the data is inserted in the persistent storage. The 'scheduler' is based on Nagios [17] and enables to distribute and balance the workload due to the huge number of periodical invocations of consumers over the whole set of identified resources. The 'persistent storage' is implemented by means of the PostgreSQL [19], an open source, free, platform independent and data-intensive database management system. The 'data aggregation and abstraction' decouples from the specific persistent storage and enables to protect it by means of caching functionality. By means of PHP (PHP Hypertext Processor) [18], the data is extracted from the persistent storage, processed to create the necessary abstraction for the final consumer and encoded in an XML document. The uppermost layer is a set of four different publishers: the 'chart' presents the extracted data in a graphical form by means of JpGraph [13], the 'HTML' transforms the XML-based data into the HyperText Markup Language (HTML) by using an XSLT transformation, the 'XML' enables to access the monitoring data in a format suitable for further processing and, finally, the 'notification' offers a publish/subscribe mechanism for event-based push delivery (at the moment, only the e-mail format is supported) [3].

EXPERIENCE

The deployment activity covers the whole EGEE Grid with several server instances supporting the work of different Grid sub-domains (e.g., whole EGEE Grid domain, ROC domain, national domain). The republisher component for the whole EGEE Grid is currently monitoring around 200 Grid sites working in $24 * 7$. Other Grid projects have adopted GridICE for monitoring their resources (e.g., EUMedGrid [9], EUChinaGRID [8], EELA [6]).

As regards the user experience, GridICE has proven to be useful to different users in different ways. For instance, Grid operators have summary views for aspects such as information sources status and host status. They also rely on the notification capability to drive their attention to emerging problems. Site administrators appreciate the job monitoring capability showing the status and computing activity of the jobs accepted in the managed resources. VO managers use GridICE to verify the available resources and their status before to start the submission of a huge number of jobs. Finally, GridICE has been positively adopted in dissemination activities where it was used to provide a bird-eye view of a Grid system, but also to drill down to single resources.

CONCLUSION

Grid monitoring is a complex activity that involves both the infrastructure and user space. In this paper, we have focused on the first aspect by describing the requirements, architectural choices and implementation details of GridICE, a monitoring tool developed specifically for Grid systems. This tool deals with the multidimensional nature of monitoring data and presents abstractions for three main categories of users: Grid operators, site administrators and Virtual Organization managers.

While GridICE has reached a good maturity level in the EGEE project, many challenges are still open in the dynamic area of Grid systems. The short term plan is: to extend the discovery and second-level consumer to deal with the existence of different Grid Information Services technologies; to integrate the monitoring activity with site management information such is the planned downtime; extend the set of sensors in order to monitor the workload management service and the data transfer activities across Grid sites.

Long term plan envisions a major redesign of GridICE taking into account new capabilities available in the open source software arena like datawarehousing-related features for modern database management systems, role-based access model to monitoring data and adoption of recent evolution of Web interface design such is the Asynchronous JavaScript and XML (AJAX).

ACKNOWLEDGEMENTS

This research was partially funded by the European Union under contract number INFSO 508833.

REFERENCES

[1] S. Andreozzi, N. De Bortoli, S. Fantinel, A. Ghiselli, G.L. Rubini, G. Tortone and M.C. Vistoli. GridICE: a Monitoring Service for Grid Systems. In Future Generation Computer Systems Journal, Elsevier, 21(4):559-571, 2005.

[2] S. Andreozzi, S. Burke, L. Field, S. Fisher, B. Kónya, M. Mambelli, J.M. Schopf, M. Viljoen, and A. Wilson "GLUE Schema Specification - Version 1.2", December 2005.

[3] S. Andreozzi, N. De Bortoli, S. Fantinel, G.L. Rubini and G. Tortone. "Design and Implementation of a Notification Model for Grid Monitoring Events". In Proceedings of the International Conference on Computing in High Energy Physics (CHEP2004), Interlaken, Switzerland. 27 September-1 October 2004.

[4] B. Balis, M. Bubak, W. Funika, T. Szepieniec, R. Wismller, and M. Radecki. Monitoring Grid Applications with Grid-enabled OMIS Monitor. In Proceedings of the 1st European Across Grids Conference, volume 2970 of Lecture Notes in Computer Science, pages 230-239, Santiago de Compostela, Spain, February 2003. Springer-Verlag.

[5] S. Cox. Observations and Measurements. Draft Specification OGC 03-022r3 Version: 0.9.2. Open GIS Consortium Inc. Oct 2004.

[6] EELA, E-infrastructure shared between Europe and Latina America. http://www.eu-eela.org/

[7] Enapling Grid for E-sciencE project. http://www.eu-egee.org.

[8] EUChinaGRID initiative. http://www.euchinagrid.org/

[9] EUMEDGRID initiative. http://www.eumedgrid.org/

[10] European DataTAG Project. http://www.datatag.org.

[11] The Globus Team. Globus Toolkit 2.2 MDS Technology Brief, Draft 4, Jan 2003.

[12] GridICE website. http://grid.infn.it/gridice.

[13] JpGraph. http://www.aditus.nu/jpgraph/.

[14] Lemon, Fabric Monitoring Toolkit. http://cern.ch/lemon/.

[15] LHC Computing Grid project. http://www.cern.ch/lcg/.

[16] J. McGarry, D. Card, C. Jones, B. Layman, E. Clark, J. Dean, and F. Hall, "Practical Software Measurement: Objective Information for Decision Makers", 1st Edition, Addison-Wesley Professional, 2001.

[17] Nagios. http://www.nagios.org/.

[18] PHP. http://www.php.net/.

[19] PostgreSQL Database Management System. http://www.postgresql.org/.

[20] B. Tierney, R. Aydt, D. Gunter, W. Smith, M. Swany, V. Taylor, and R. Wolski. A Grid Monitoring Architecture, GGF Informational Document GFD.7, Jan 2002.

[21] W. Wahl, T. Howes, and S. Kille. Lightweight Directory Access Protocol v.3, RFC 2251, IETF, Dec 1997.

[22] S. Zanikolas and R. Sakellariou, "A Taxonomy of Grid Monitoring Systems", In Future Generation Computer Systems Journal, 21(1), January 2005.