Supporting service management data composition in grid environments

Vitalian A. Danciu, Nils gentschen Felde
Munich Network Management Team
Ludwig-Maximilians-University of Munich
Oettingenstr. 67, 80538 Munich, Germany
{danciu, felde}@mnm-team.org

Abstract

Service management issues commonly encountered in IT infrastructures appear greatly amplified when observed in grid environments. Specifically, the creation of views on services – grid services as well as other services provided by infrastructures supporting the grid – becomes more difficult. In this paper we suggest the application of a generic service monitoring architecture to the grid environment. For this purpose, the Relational Grid Monitoring Architecture (R-GMA) is employed as a means of extracting data relevant to the grid. The aggregation of this data into a service view is facilitated by the composition layers of the architecture.

Keywords

Grid, Monitoring, Service Management

1. Motivation

The emerging grid environments make use of a multitude of monitoring tools that provide a huge amount of information about the grid regarding resources, user data, applications and sometimes even jobs. However, a service view on grid monitoring data is not yet available.

Monitoring architectures being deployed at present correlate and aggregate events originating from resources to higher level information records, allowing decisions leading to management actions. Examples include the *IBM WebSphere Studio Application Monitor* [6], *HP OpenView* [1] or the *Grid Network Management Solution* offered by Clear & Sorrento. To protect investments (financial and time wise) managing services within the grid on basis of existing monitoring tools and frameworks is an important issue. The products and efforts mentioned above neither focus on providing a service view nor integrate monitoring data from similar tools. However, they can be integrated into an umbrella monitoring architecture addressing these issues.

In order to support service management data composition it is necessary to retrieve relevant information about the underlying system and especially lower level information about the resources involved. This information may be sufficient to trigger automated management actions targeting the effected resources themselves or sending alert messages

* Parts of this work have been funded by the German Federal Ministry of Education and Research under contract 01 AK 800 B.

to the administrator. However, management of grid services in a broader scope requires aggregated and correlated information that constitutes combined status-data of underlying resources, often from multiple domains.

In this paper we apply the generic architecture described in [3] to grid environments. The aim is to facilitate deployment of well established grid monitoring tools like *gLite*. The upper layers of the architecture provide the facilities required for aggregation of the monitoring data gathered. We rely on the *Relational Grid Monitoring Architecture* (R-GMA) as a basis for the extraction of grid monitoring data. The approach is not limited to the R-GMA setup. Alternative monitoring systems such as the *Monitoring and Discovery Service* (MDS) of the *Globus Toolkit* (GT) [4] or the *Unicore* monitoring components could be used, if applicable.

2. The Relational Grid Monitoring Architecture

The *Grid Monitoring Architecture* (GMA) [7] is a *Global Grid Forum* (GGF) standard regarding the design of monitoring architectures in the grid. An implementation of and extension to this standard is described within the so-called *Relational Grid Monitoring Architecture* (R-GMA) [2]. Among other improvements, it introduces the concept of the *republisher* that provides a bus-like dissemination of monitoring data. Figure 1 illustrates the R-GMA schematically as needed in this context. The generic producer–consumer role model defined in GMA is combined with a directory service where producers and consumers have to register as depicted on the right hand side of the figure.

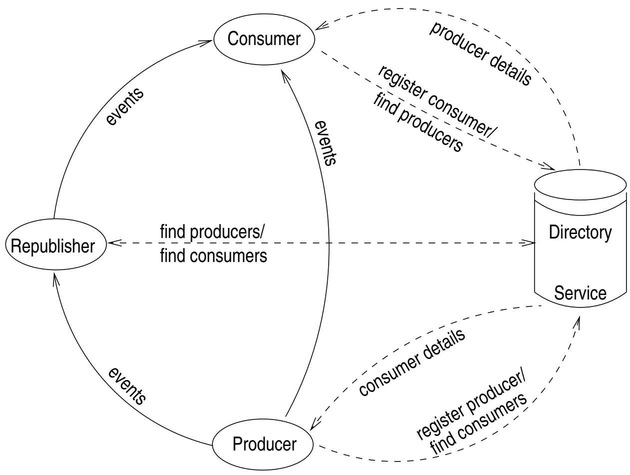

Figure 1 The R-GMA Roles

Consumers are for example grid management components using the information provided by the producers. They may retrieve a list of the producers available along with their corresponding capabilities from the directory service and register directly with the producers that satisfy their needs. The producer will provide monitoring data to its registered consumers.

The republisher added in R-GMA provides the ability to multicast information gathered by the producers. The monitoring information is thus forwarded to the consumers which have subscribed the information from the producer, as well as offered to applications that are not parts of the R-GMA model.

3. Using R-GMA to support service management data composition

This section gives an overview of our approach to service management data composition with an underlying grid monitoring architecture. In our case, using the R-GMA republisher to connect the grid monitoring subsystem with higher level service management data composition enables the aggregation and correlation of events in order to enhance and/or automate service management procedures.

Figure 2 shows the architecture as described in [3] deploying the R-GMA components as information providing agents. In the following, we describe the different layers and components of the resulting, integrated architecture in detail.

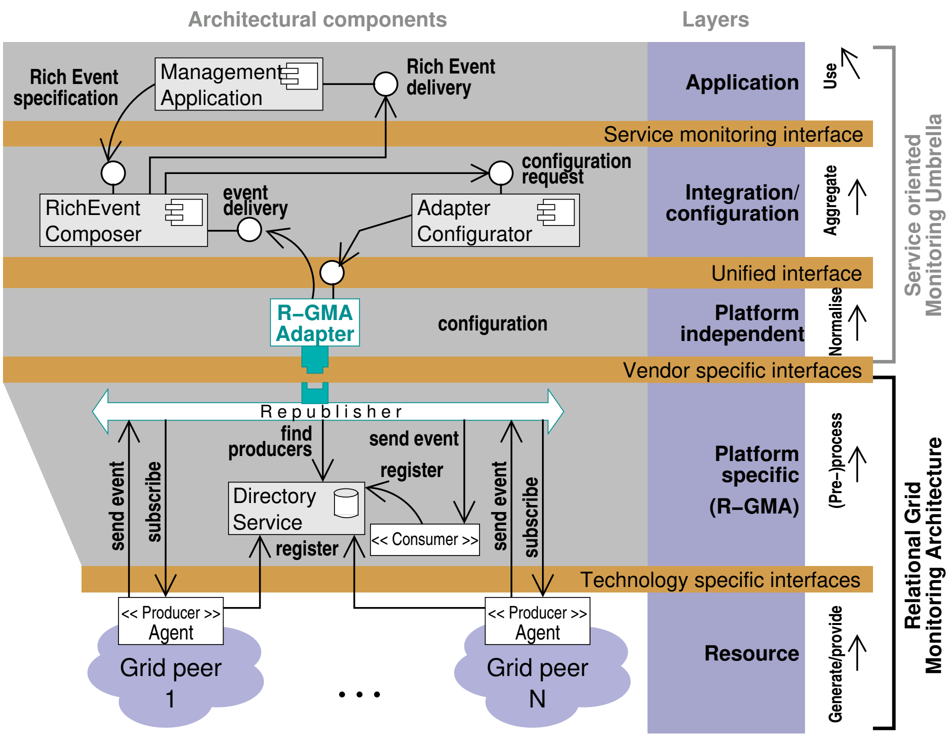

Figure 2 Integration of R-GMA into a generic, service oriented architecture

Resource layer The lowest layer – the resource layer – consists of sensors that gather information about monitored resources. These sensors are resource specific and form a very heterogeneous variety of information providers. In our case, the sensors are realized by (R-)GMA agents in the "Producer" role (compare figure 1). The information provided by each agent is very resource specific and hardly any pre-processing is performed by the agents themselves.

Platform specific layer The platform specific layer contains the grid specific monitoring system. Here, we use the R-GMA as described in section 2. This layer comprises a directory service and components in the "Consumer" role (e.g. a gLite resource broker [2]) which use the data collected by the sensors of the resource layer.

The R-GMA performs grid specific operations upon the information provided by the producer of the resource layer. This results in aggregated information about the status of the grid infrastructure and its components. However, aggregation is performed on a per-attribute basis. While this lowers the data volume transported to upper layers, it does not take into account the needs of service management regarding aggregation of information items from different sources. In our setup, these needs are addressed by the Integration/Configuration layer, specifically by the RichEvent Composer component (see the description of that layer for details).

Platform independent layer The first step towards a homogeneous service view is a technology independent layer that hides the heterogeneity of underlying tools and interfaces. This layer consists of adapters that provide the information of the monitoring systems connected.

In our case, the platform independent layer includes an adapter for R-GMA, connected to the republisher component. It connects the grid monitoring environment to the higher level service management layers for data composition as shown in figure 2. Thus, from an R-GMA model perspective, it occupies a Consumer role.

Beside this adapter, other adapters connecting several different monitoring subsystems may be included as [3] points out. An umbrella architecture combining multiple monitoring systems with different, among each other incompatible interfaces can be achieved in this manner.

Integration and configuration layer Aggregation across different monitoring data types originating from different resources is performed in this layer. The *RichEvent Composer* component is configured with a specification of aggregates (rich events) to be produced, as well as the triggering conditions (e.g. points in time or value thresholds) for the relaying of aggregated and named data records.

Adapter configuration is performed centrally in the *Adapter Configurator* component, in response to configuration requests. These requests originate at the composer in accordance with the requirements of a service monitoring task.

Beside the grid-specific usage of resources, other data sources – encapsulated by their specialized adapters – may be included in rich event generation. However, this is outside the scope of this work.

Application layer Requirements regarding monitoring information are specified by an administrator operating a management application. *RichEvents* originating from the integration and configuration layer are made available to the management application. Their structure can be formalized using a declarative language as the one suggested in [3]. The reaction to receiving an event depends on the managment application and on the purposes of the administrator; a plausible option is the evaluation of managment policies, possibly resulting in the execution of management operations.

4. Further work

Service monitoring in grids poses new challenges in that it combines inter-organizational monitoring issues with the need for a service view on gathered information.

In this work, we leveraged an architecture aimed at supporting service management and integrated grid-specific low-level monitoring components into its lower layers. While this approach is a first step towards a monitoring architecture designed to provide higher-order monitoring information in a grid context, a lot of work has still to be done. The next step in our work is to enlarge the scenario and involve multiple different grid monitoring systems. This implies an investigation on the impact of involving information providers of different capabilities and information of different levels of granularity onto the overall system and its impact on the quality of information derived. A study about the scalability of the system (especially, the components of the integration layer) seems necessary when considering large scale scenarios.

Acknowledgment

The authors wish to thank the members of the Munich Network Management (MNM) Team for helpful discussions and valuable comments on previous versions of this paper. The MNM Team directed by Prof. Dr. Heinz-Gerd Hegering is a group of researchers of the University of Munich, the Munich University of Technology, the University of the Federal Armed Forces Munich, and the Leibniz Supercomputing Center of the Bavarian Academy of Sciences. Its web-server is located at `http://www.mnm-team.org`.

References

[1] Hewlett-Packard Company. HP OpenView, May 2006.

[2] A.W. Cooke, A.J.G. Gray, W. Nutt, J. Magowan, M. Oevers, P. Taylor, R. Cordenonsi, R. Byrom, L. Cornwall, A. Djaoui, L. Field, S.M. Fisher, S. Hicks, J. Leake, R. Middleton, A. Wilson, X. Zhu, N. Podhorszki, B. Coghlan, S. Kenny, D. OrsquoCallaghan, and J. Ryan. The Relational Grid Monitoring Architecture: Mediating Information about the Grid. *Journal of Grid Computing*, 2(4):323–339, December 2004.

[3] V. Danciu and M. Sailer. A monitoring architecture supporting service management data composition. In *Proceedings of the 12th Annual Workshop of HP OpenView University Association*, number 972–9171–48–3, pages 393–396, Porto, Portugal, July 2005. HP.

[4] I. Foster. Globus Toolkit Version 4: Software for Service-Oriented Systems. In *IFIP International Conference on Network and Parallel Computing*, pages 2–13. Springer-Verlag LNCS 3779, 2005.

[5] I. Foster and C. Kesselmann, editors. *The Grid 2: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, 2004.

[6] IBM. WebSphere Studio Application Monitor, May 2006.

[7] B. Tierney, R. Aydt, D. Gunter, W. Smith, V. Taylor, R. Wolski, and M. Swany. A grid monitoring architecture. Technical report, Global Grid Forum, January 2002.

Biographies

Vitalian A. Danciu studied computer science at the Ludwig-Maximilian-University in Munich, Germany (LMU), where he received his diploma degree (M. Sc.). Since joining the MNM team in 2003, he is a Ph. D. candidate at LMU while working as a research and teaching assistant. His research interests include policy-based, process-oriented management, as well as management of mobile systems.

Nils gentschen Felde studied computer science at the Rheinische Friedrich-Wilhelms-University Bonn, Germany, where he received his diploma degree (M. Sc.). Since joining the MNM team in 2005, he is a Ph. D. candidate at LMU while working as a research and teaching assistant. His research interests include grid computing, grid monitoring, as well as IT security.