International Journal of Advanced Computer Technology (IJACT)

ISSN:2319-7900

# GRID MONITORING SYSTEMS: A SURVEY

Harneet Arora, SGGSWU, Fatehgarh Sahib; Priyanka Arora, GNDEC, Ludhiana; Kamaldeep Kaur, GNDEC, Ludhiana

## Abstract

The Grid monitoring process forms the eternal part of any grid environment. It helps in collecting the information related to the state and functionality of the grid resources at every instant of time. This collected information is further utilized for managing the grid resources and preventing the occurrence of undue failures. Grid monitoring is a widely researched area. A number of grid systems have been designed to cater the different requirements of a grid environment. This paper provides an overview of the grid monitoring process and its architecture. It also discusses the various grid systems that are available by classifying them into different categories based on the characteristics of a grid environment in which it is used.

## Introduction

A Grid environment is composed of highly distributed and heterogeneous resources which are dynamic in nature. In case of a failure it becomes very difficult to find out where the fault has occurred. As such it is quite essential to maintain the track of all the grid resources. The process of grid monitoring helps in collecting and maintaining the information related to the current and the past status of the grid resources [1]. This information helps the user in checking the availability of the resources, their characteristics, finding the faults in case of a failure. Therefore this process plays a vital role in resource selection, scheduling, data replication and performance analysis of a grid system [2].

The monitoring process can be categorized into four phases:

(i) *generation of events:* In this phase sensors enquire the elements present in a system

(ii) *processing of generated events:* The collected data is filtered according to the criterion specified

(iii) *distribution of events:* The processed information is distributed to the interested parties

(iv) *presentation (consumption) of events*: The events are presented to the parties in a proper order so that it becomes easier for the users to draw conclusions.

This process should also satisfy a set of requirements depending upon the system on which it is used. These requirements include: (i) *scalability* (ii) *portability* (iii) *extensibility* etc. The process of grid monitoring along with the set of requirements has been specifically structured by the Global Grid Forum into the Grid Monitoring Architecture. A number of grid systems have been designed based on this architecture. The paper describes the grid monitoring architecture.

## Grid Monitoring Architecture

The Global Grid Forum has described architecture for implementing the grid monitoring architecture. As shown in fig 1, this architecture includes a number of components like:

*(i)* *Producer:* It is a process which generates the events.

*(ii)* *Consumer:* It is a process which receives the events generated by the producer.

*(iii)* *Registry*: It is a directory service that allows producers to publish the set of events that are generated and allows consumers to access the published events.

*(iv)* *Interactions*. The interactions between the producers and consumers are categorized into three: (a) *Subscribe/Publish:* It is the set of interactions that are initiated by the producer. (b) *Query/ Response*: It is the set of interactions that are initiated by the consumer followed by a response from the producer. (c) *Notification*: It is the interaction initiated by the producer without any further interaction.

*(v)* *Republisher:* It is the combination of both producer and consumer as single entity which is used for aggregating, summarising and filtering the data.

*(vi)* *Schema Repository:* It is the event schema which is the collection of different events.

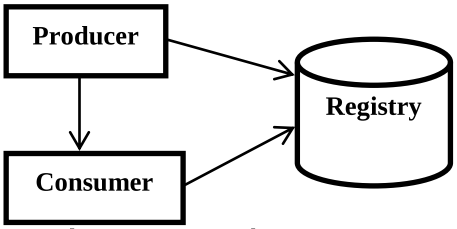


Fig 1: The Grid Monitoring Architecture.

## A. Taxonomy of Monitoring Systems

The grid monitoring systems can be categorized into four depending upon the characteristics of the Producer and Republisher. These categories are:

**i) Level 0:** In level 0 systems the events flow from sensors to consumers in an online modeor offline mode as shown in fig 2. Level 0 systems are also known as self- contained systems because these systems do not exhibit their functionalities to producers. The different level 0 systems that are available are:

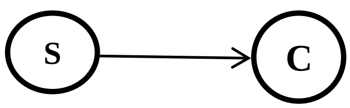


Fig 2: The flow of events in level 0 systems.

(a) *GridICE:*GridICE was developed to facilitate the grid administrators by providing them the information regarding the entire virtual organization, individual organizations and resources. It has a centralized structure in which the main server extracts the information about the status of the resources present in the entire grid environment. The GridICE has its own sensors which collect the information and store them in a database. This stored information is then used for making the performance analysis of the grid environment [4].

(b) MapCenter: MapCenter is used to check the availability and distribution of services by the administrator. It periodically monitors only the availability of grid resources and the services provided by them. This collected information can be presented in different logical views as required by the administrator [5].

**ii) Level 1**:In level 1 system the sensors are either implemented separately or on the same machine as the producers. The architecture of Level 1 systems is shown in fig 3.

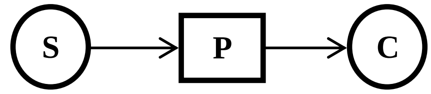


Fig 3: The flow of events in level 1 system

These systems are also known as Producer only Systems. The level 1 system that is available is:

*a) Autopilot:* Autopilot was developed for the systems which have to adapt to changing policies. Its functionality is implemented via four separate components that are sensors, actuators, clients and distributed servers. The sensors are used for reading the information about the resources whereas actuators are used for writing the information. Both the sensors and actuators store their characteristics in a property list. This property list is used by the clients. The collected information is then summarized and aggregated as per the requirement [6].

**iii) Level 2**: In level 2 systems in addition to producers there is a republisher which performs a fixed set of tasks as shown in fig 4. These systems are also known as Producer and Republisher Systems. Some of the level 2 systems are:

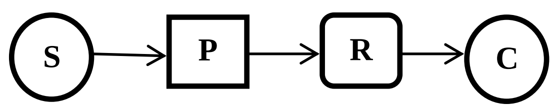


Fig 4: The flow of events in level 2 systems.

(a) *CODE based monitoring systems:* The code system has centralized republisher which is used for organization wide administration. It has four components namely observers, actors, managers and directory service. The observers control the sensors and hence act as the producers. Actors manages the all the daemons. Managers acts as the consumers and directory services acts as the registry [7]. The other grid monitoring systems with a centralized republisher is GridRM, Hawkeye etc.

ISSN:2319-7900

(b) *JAMM:* JAMM stands for Java agents for monitoring and management. This system has distributed republishers. In JAMM a sensor manager is used per host in order to handle dynamic activation of hosts. The sensors act as the producers. The gateways act as the republishers and the consumers collect the information from the sensors by making use of registry [8]. The other systems that have distributed republishers are HBM, Remos etc.

(c) *NWS:* The Network Weather Service is particularly used for scheduling and dynamic resource allocation. In NWS, the sensors collect both the active and passive information about the resources available. It includes information like CPU usage, memory availability etc. A forecasting process then consumes the information collected by the sensors for performance analysis [9].

**iv) Level 3:** In level 3 systems a number of republishers are present which can be arranged in any hierarchy. These systems are also known as Hierarchy of Republishers. The level 3 systems that are available are:

*(a) Gangila:* In Gangila, a set of arbitrarily arranged republishers collect the information from the lower level data sources and provide this information on demand to the higher level republishers. This system does not have a registry as such the information regarding the producers and the republishers should be known beforehand [11].

*(b) RGMA*: The Relational Grid Information Service Research. In RGMA different set classes of producers are used. The database producers collect the static information whereas the stream producers collect the dynamic information. The producers declare their relation using SQL queries. A consumer is the entity which uses the 'select query'. A global schema is used which stores all the relations. A Republisher is a set of queries which provides a relational view of the entire system [12].

The other Level 3 systems are: MonALISA, Paradyn etc.

# Conclusion

A large number of grid monitoring systems are available that address all the requirements of a grid monitoring system. These systems have different set of specifications and can be employed depending upon the need and the availability of resources in a grid environment.

# Acknowledgments

The authors are thankful to IJACT Journal for the support to develop this document.

# References


[1] IanFoster and Carl Kesselman (Ed.),"The GRID 2 Blueprint for a New Computing Infrastructure", Elsevier, USA, 2004.

[2] SerafeimZanikolas andRizosSakellariou, "A taxonomy of grid monitoring systems", Elsevier, 2004.

[3] B. Tierney, R. Aydt, D. Gunter, W. Smith, M. Swany, V. Taylor and R. Wolski, "A Grid Monitoring Architecture", GWDPerf-16–3, Global Grid Forum, August 2002.

[4] S. Andreozzi, N. De Bortoli, S. Fantinel, A. Ghiselli, G. Tortone andC. Vistoli, "GridICE: A Monitoring Service for the Grid", Third Cracow Grid Workshop, Cracow,Poland, October 27–29, 2003, pp. 220–226

[5] F. Bonnassieux, R. Harakaly and P. Primet, "MapCenter: an opengrid status visualization tool", 15th International Conference on Parallel and Distributed ComputingSystems, Louisville, USA, 2002.

[6] R.L. Ribler, J.S. Vetter, H. Simitci and D.A. Reed, "Autopilot: adaptivecontrol of distributed applications", 7th IEEE Symposium on High-Performance DistributedComputing, 1998, pp. 172–179.

[7] W. Smith, "A Framework for Control and Observation in DistributedEnvironments", Tech. Rep. NAS-01–006, NASA AdvancedSupercomputing Division, NASA Ames Research Center,July 2001.

[8] B. Tierney, B. Crowley, D. Gunter, J. Lee and M. Thompson, "Amonitoring sensor management system for grid environments",Cluster Comput. Vol. 4 (1), 2001, pp. 19–28..

[9] R.Wolski, N. Spring and J. Hayes, "The network weather service: adistributed resource performance forecasting service for metacomputing",J. Future Generation Comput, 1999, pp.757–768.

ISSN:2319-7900

[10] M.L. Massie, B.N. Chun and D.E. Culler, “Ganglia DistributedMonitoring System: Design, Implementation, and Experience”,Parallel Computing, Vol. 30, 2004, pp. 817–840.
[11] A. Cooke, A.J.G. Gray, L. Ma, W. Nutt, J. Magowan, M. Oevers,P. Taylor, R. Byrom, L. Field, S. Hicks, J. Leake, M. Soni,A.Wilson, R. Cordenonsi, L. Cornwall, A. Djaoui, S. Fisher,N.Podhorszki, B. Coghlan, S. Kenny and D. O’Callaghan, “R-GMA:an information integration system for grid monitoring”, 10th International Conference on Cooperative Information Systems, 2003.

# Biographies

**HARNEET ARORA,** received the M.E. degree in Computer Science & Engineering from the University Institute of Engineering & Technology, Panjab University, Chandigarh, in 2011, the B.Tech. Degree in Computer Science & Engineering from CEC, Landran, Punjab Technical. Currently, She is an Assistant Professor of Computer Science & Engineering at Sri Guru Granth sahib World University, Fatehgarh Sahib. Her teaching and research areas include Wireless Sensored Networks, Grid Computing. Professor may be reached at harneet159@gmail.com

**PRIYANKA ARORA,** received the M.E. degree in Computer Science & Engineering from the University Institute of Engineering & Technology, Panjab University, Chandigarh, in 2011, the B.Tech. Degree in Computer Science & Engineering from Guru Nanak Dev Engineering College, Punjab Technical. Currently, she is an Assistant Professor of Information Technology at Guru Nanak Dev Engineering College, Ludhiana. Her teaching and research areas include Grid Computing.

**KAMALDEEP KAUR,** received the M.E. degree in Computer Science & Engineering from the University Institute of Engineering & Technology, Panjab University, Chandigarh, in 2011, the B.Tech. Degree in Computer Science & Engineering from Guru Nanak Dev Engineering College, Punjab Technical. Currently, she is an Assistant Professor of Computer Science & Engineering at Guru Nanak Dev Engineering College, Ludhiana. Her teaching and research areas include Data mining, Grid Computing.