The evolution of Grid Brokers: union for interoperability

ATTILA KERTÉSZ

Institute of Informatics, University of Szeged
MTA SZTAKI Computer and Automation Research Institute
keratt@inf.u-szeged.hu

Reviewed

Keywords: Grid Brokers, grid interoperability, Meta-brokering, matchmaking

Grid resource management is probably the research field most affected by user demands. Though well-designed, evaluated and widely used resource brokers, meta-schedulers have been developed, new capabilities are required, such as agreement and interoperability support. Existing solutions cannot cross the border of current middleware systems that are lacking the support of these requirements. In this paper we examine and compare different research directions followed by researchers in the field of Grid Resource Management, in order to establish Grid Interoperability. We propose a meta-brokering approach, which means a higher level resource management by enabling communication among existing Grid Brokers and utilizing them.

1. Introduction

Grid Computing has become a detached research field in the '90s and since then it has been targeted by many world-wide projects. Several years ago users and companies having computation and data intensive applications looked skeptically at the forerunners of grid solutions, who promised less execution times and easy-to-use application development environments by creating a new high performance network system of interconnected computers from all around the world. Research groups were forming around specific middleware components and different research branches have grown out of the trunk.

Many user groups from various research fields (biology, chemistry, physics, etc.) put their trust in grids, and today's usage statistics and research results show that they were undoubtedly right. Grid Computing is in the spotlight, several international projects aim at establishing sustainable grids (CoreGRID [1], LA Grid [2], Globus [3], etc.).

Nowadays research efforts are focusing on user needs: more efficient utilization and interoperability play the key roles. Grid resource management is probably the research field most affected by user demands. Though well-designed, evaluated and widely used resource brokers have been developed, new capabilities are required, such as agreement (Service Level Agreements, WS-Agreements [4]) and interoperability support. These two directions also depend on other grid middleware capabilities and services, and since they can hardly cross the border of these middleware solutions, they need revolutionary changes affecting the whole system. Solving these problems is crucial for the next generation of grids, which should rise up from the academic to the business world.

To achieve this, capabilities such as advance reservation and co-allocation need to become reality, but the currently used grid middleware solutions do not provide these services. Therefore, usually estimations and predictions are used in the scheduling process of the resource managers to overcome these lacking features and provide a more efficient schedule. (For example, Lőrincz et. al. monitor runtime information to determine the behavior of the job and use these additional data in scheduling [14]). Trying to enlarge the limitation borders, in this paper we are focusing on interoperability approaches in the field of Grid Resource Management.

The current solutions of grid resource management will not be able to fulfill the high demands of future generation grid systems, though several grid resource brokers [5] have been developed supporting different grid systems. Their main problem is that most of them cannot cross the borders of current grid middleware solutions, therefore the newly arisen problems need to be treated with novel research approaches. Nowadays grid systems have their own researchers and user groups. This means borders not only for the development but also for the interoperable utilization.

Figure 1. illustrates the current grid utilization: grid resources can be accessed either directly by using grid middleware components or through grid brokers that help finding a proper execution environment or through grid portals that provide a convenient user interface to grid services.

The need for interoperability among different grid systems has raised several questions and directions. The advance of grids seems to follow the way assigned by the Next Generation Grids Expert Group, which has been established by the European Commission. In their latest publication [6] they have pointed out that grid and web services are converging and envisaged a new hybrid architecture called SOKU (Service Oriented Knowledge Utility), which enables more flexibility, adaptability and advanced interfaces, therefore interoperability is evident and congenital in these systems.

Following these expert guidelines and the latest requirements of grid user groups, in this paper we propose a grid resource management solution that does not require major changes of the whole grid middleware and still provides interoperability.

2. Resource management and matchmaking in grids

When grids were born, resource management components of the middleware provided various interfaces to submit jobs, transfer files, query resource information, track job states and retrieve execution results. As grids and the number of users were growing, the dynamicity and heavy load made users unable to cope with manual resource selection. Automatic matchmaking between user requests and available resources came into view and resource brokers were born. This additional component contacts the Information System of the grid and schedules user requests to a proper execution environment, computing resource (most of the time *proper* means likely the fastest execution). In addition to contacting the resources, transferring the jobs, tracking the states and staging back the result files are also the tasks to be performed by the brokers. Describing the job requirements is done with a middleware-specific language (in general job description language).

This document needs to be submitted to the broker with the necessary input files and executable. This is the first time where interoperability problems appear. If users wanted to use different grid solutions, they need to use different description forms for the same requirements. Furthermore grids use different protocols to store resource information, transfer files, access resources, etc., though they implement the same methodology. Knowing these facts it is not surprising that users and developers have started to form separate user groups and developer communities around various grid solutions. Because of the same reasons resource brokers generally support one grid middleware and its job description language, therefore they are tightly coupled to that middleware.

Up to now most of the broker developers identified this separation and have started to redesign and extend their solutions for multi-middleware and multi-language support to provide a basic level of interoperability. Though carrying out these extensions take much time and still in progress nowadays, several solutions are ready to serve different user communities of different grids. The additional components for understanding other language descriptions and using other protocols make the extended brokers more and more robust and unmanageable. These redesigns are usually done for similar description forms and protocols, or for middleware solutions having common components. These observations show that broker extensions cannot be done for all the available middleware solutions, and the more grids an extended broker supports, the more failures can occur during its utilization.

Another possibility to enhance interoperability is the use of grid portals. These tools provide easy to use graphical interfaces to utilize various grid components. There are general purpose and specialized ones for supporting specific applications. An instance of the second approach is the Conflet framework (CONFigurable

Figure 1. The utilization of production Grids

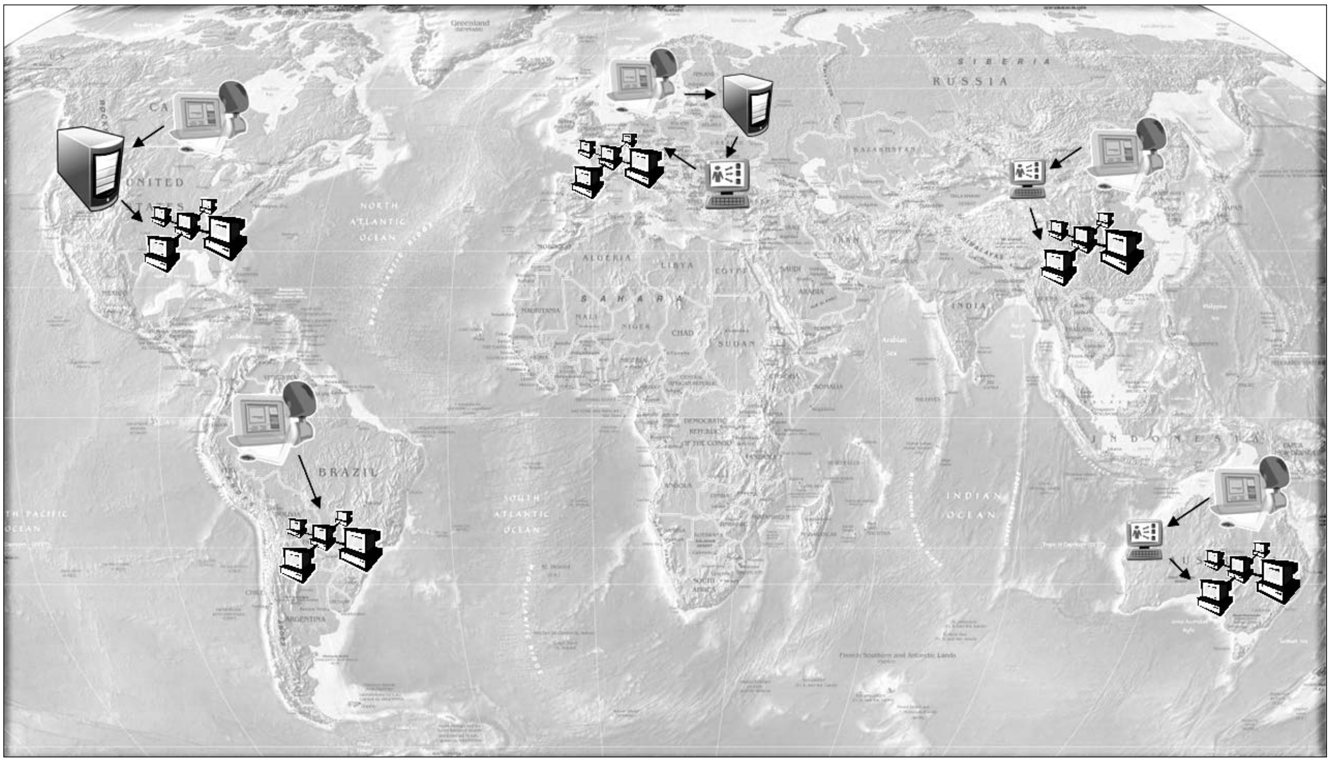
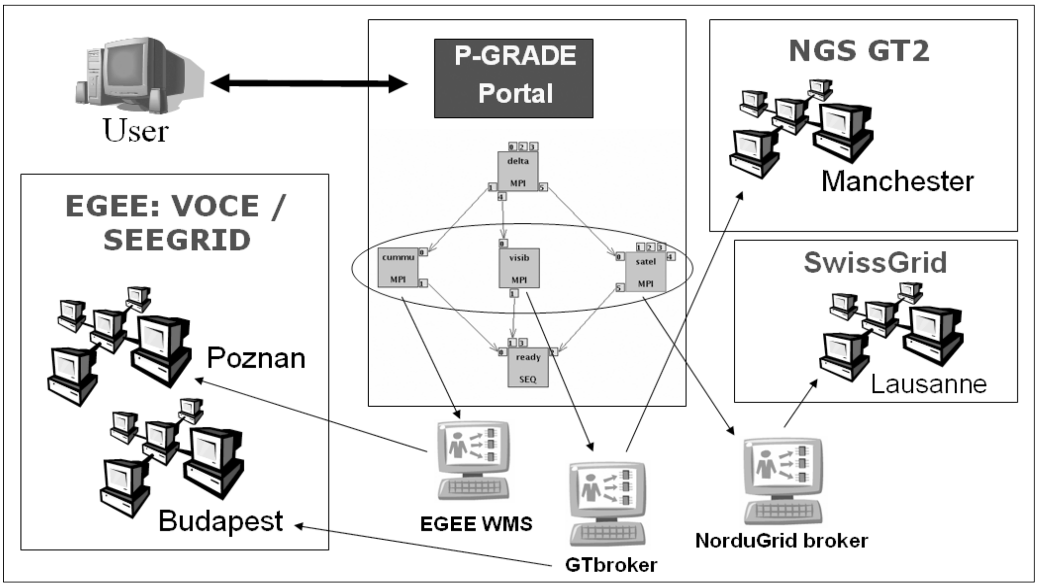


Figure 2.
Multi-grid and multi-broker support in the P-GRADE Portal



portLET [13]), which can be used to create specific portlets to one's application. Interfacing different brokers to portals is another option to extend interoperability and support more middleware solutions. Nevertheless these portals also attract other user communities and provide more computational power. In *Figure 2.*, we can see how the P-GRADE Portal [7] supports various production grids by interfacing different resource brokers.

The P-GRADE Portal is a workflow-oriented grid portal with the main goal to support all stages of grid workflow development and execution processes. It enables a graphical design of workflows created from various types of executable components, executing these workflows in Globus-based computational grids [3] relying on user credentials, and finally analyzing the monitored trace-data by the built-in visualization facilities. In the Portal box field of Fig. 2., the bigger boxes represent the executable files, jobs (delta, cummu, etc.), the smaller numbered boxes (ports) on their sides represent input and output files. Connecting these ports the user can create an application of dependent jobs, which together form a workflow. In the last step of the workflow edition the user can select brokers or resources to the jobs. During the execution these brokers take care of the execution of the jobs, or they are directly submitted to the manually selected resource. The disadvantage of this solution is the same as in the previous case: interfacing additional brokers requires modifications to the system.

3. The evolutionary step: unifying Grid Brokers

Facing with the problems stated in the previous section, several research groups turned their attentions to new solutions to establish interoperability. It has become obvious that keeping the same architecture would not bring interoperability in the near future; they need to wait for revolutionary changes in the whole middleware to enable a world-wide interoperable grid, which would take long years. The only way to achieve a higher level of interoperability in reasonable time is to unify brokers by enabling communication and data-flow among them.

One of the biggest grid research organizations is the OGF (Open Grid Forum), which has many research groups to share innovative ideas and standardize solutions in various fields of Grid Computing. The GSA-RG (Grid Scheduling Architecture Research Group [8]) is currently working on a project enabling grid scheduler interaction. They try to define common protocol and interface among schedulers enabling inter-grid usage. Implementing such an interface and using it by all the brokers would enable sharing different user jobs, workloads. Agreeing on a common interface and implementing it to the brokers definitely takes a long time.

The other similar approach enables communication among the same broker instances. Since in this case no negotiation is needed with other researchers and solutions, it is easy to agree on an interface and the implementation needs to be done only for their own solution. (Note that in this case different protocols will be created and used by different developers, again.) This approach is followed by the following projects: Koala [9], LA Grid [2] and Gridway [10].

Comparing these approaches we can see that all of them use a new method to expand current grid resource management boundaries. The interconnected domains are being examined as a whole, and they delegate resource information among broker instances managing different domains. Usually the local domain has preference and when it is overloaded, some jobs are passed to somewhere else – in this case the results should be passed back to the initial domain. Though this is a novel approach and all of them proved that they achieved better load balancing, the interoperability problem among different systems is still not solved.

The final solution lies in meta-brokering. This approach means a higher level brokering, which uses the existing resource brokers to reach different grids. Unlike existing brokers it uses metadata about the available broker capabilities and maps user requests to brokers not to resources. In order to achieve this we need to store and consume metadata about user jobs and resource brokers. The OGF has already developed a standard language for describing jobs - this is the JSDL (Job Submission Description Language [11]).

Regarding broker capabilities we designed a BPDL (Broker Property Description Language [12]) description format together with researchers from the Barcelona Supercomputing Center. Scheduling at the meta-brokering level requires additional metadata about the scheduling requirements of the users and scheduling properties of the brokers. Since the JSDL is lacking these attributes and the BPDL incorporates some of them, we decided to create a separate language called MBSDL (Meta-Broker Scheduling Description Language). Here we mention that the OGF-GSA-RG [8] has started to define an SDL (Scheduling Description Language) for enabling the aforementioned inter-broker communication, but they have not got too far, yet. We believe that MBSDL can be regarded as a contribution to their work. Once SDL becomes a standard, our system will be ready to use it.

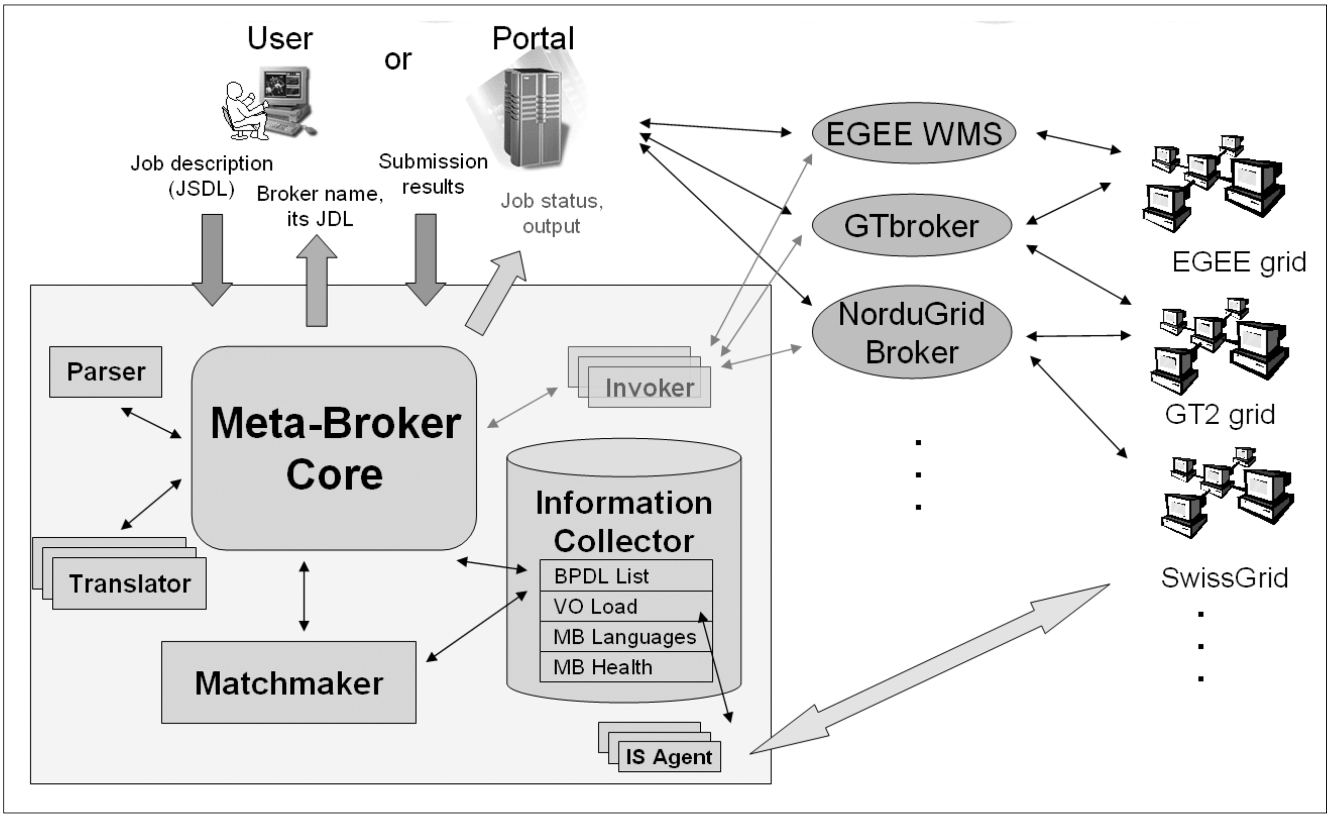
Using these tools we developed a meta-brokering service as a general web-service and named it as Grid Meta-Broker (shown in *Figure 3*). Together with BPDL and MBSDL metadata can be stored about resource brokers in the Information Collector (IC) component of the system. The users can specify their requirements with JSDL and MBSDL. Consuming these documents the MatchMaker component executes a scheduling algorithm to select a broker and a grid for the actual user job. So-called IS Agents are used to provide up-to-date information about the background grid load to help the MatchMaker skip grids with overloaded or unavailable resources. The next step is to translate the user request to the language of the selected broker and let the Invoker submit the job contacting the underlying broker. This component is responsible for tracking job states through the broker and retrieving the result files and logging information. The final step is to provide the results to the user and update the IC with broker performance data.

Another scenario can also be done, when instead of the Invoker the user or a portal contacts the selected broker and does the actual job submission. In this case they need to report the submission results themselves to the Meta-Broker.

4. Conclusions

We have learned several reasons why existing resource management systems cannot fulfill the newly arisen requirements of grid users. Providing bigger computation power and serving business-oriented investments requires a novel, higher level approach in grid resource management: we need to unify the separated grid islands and manage them together. Extending the current resource brokers with multi-middleware support or

Figure 3. The architecture of the Grid Meta-Broker



interfacing them by widely used grid portals can be a good starting point, but in the long run they become unmanageable and vulnerable.

More successful solutions have been developed by enabling inter-broker communication among specific broker instances operating in different domains of the same grid. Though it brings some level of interoperability, these brokering systems still cannot work together, do not have common interfaces.

The final solution for grid interoperability is the Grid Meta-Broker, which has been built on the latest standards of web and grid technologies taking into account the guidelines of NGG. The goals of meta-brokering are to use the widespread resource brokers to manage their own grids and to provide an intelligent way to unify these brokers and offer it to the users as a transparent multi-grid service.

In a unified, world-wide grid (WWG [15]) these Meta-Brokers will bridge the yet separated islands of grids and serve the whole user community in a fully interoperable manner.

Acknowledgement

This research work is carried out under the FP6 Network of Excellence CoreGRID funded by the European Commission (Contract IST-2002-004265).

References

[1] http://www.coregrid.net
[2] http://www.latinamericangrid.org/
[3] http://www.globus.org/
[4] http://www.ogf.org/documents/GFD.107.pdf
[5] A. Kertész, P. Kacsuk: A Taxonomy of Grid Resource Brokers, 6th Austrian-Hungarian Workshop on Distributed and Parallel Systems (DAPSYS 2006) in conjunction with the Austrian Grid Symposium 2006, Innsbruck, pp. 201-210, Austria, September 21-23, 2006
[6] Next Generation Grids Report: Future for European Grids: GRIDs and Service Oriented Knowledge Utilities – Vision and Research Directions 2010 and Beyond, December 2006 (NGG3).
[7] A. Kertész, G. Sipos, P. Kacsuk: Multi-Grid Brokering with the P-GRADE Portal, In post-proc. of the Austrian Grid Symp. (AGS'06), OCG Verlag, Austria, 2007.
[8] https://forge.gridforum.org/sf/projects/gsa-rg
[9] A. Iosup, D. H.J. Epema, T. Tannenbaum, M. Farrellee, M. Livny: Inter-Operating Grids through Delegated MatchMaking, In proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'07), Reno, Nevada, November 2007.
[10] T. Vazquez, E. Huedo, R. S. Montero, I. M. Llorente: Evaluation of a Utility Computing Model Based on the Federation of Grid Infrastructures, pp.372–381, Euro-Par 2007, 28 August 2007.
[11] http://www.ogf.org/documents/GFD.56.pdf
[12] A. Kertész, I. Rodero, F. Guim: Data Model for Describing Grid Resource Broker Capabilities, CoreGRID Workshop on Grid Middleware in conjunction with ISC'07 Conference, Dresden, Germany, 25-26 June 2007.
[13] D. Pasztuhov, I. Szeberényi: A new architecture of the Conflet system, Networkshop 2007, (in Hungarian). https://nws.niif.hu/ncd2007/docs/ehu/036.pdf
[14] L. Cs. Lőrincz, A. Ulbert, Z. Horváth, T. Kozsik: Towards an Agent Integrated Speculative Scheduling Service, 6th Austrian-Hungarian Workshop on Distributed and Parallel Systems (DAPSYS'2006), pp.211–222, Innsbruck, Austria, 21-23 Sept. 2006.
[15] P. Kacsuk, A. Kertész, T. Kiss: Can We Connect Existing Production Grids into a World Wide Grid?, 8th International Meeting High Performance Computing for Computational Science (VECPAR'08), Toulouse, France, 24-27 June 2008. (Submitted)

Author

Attila Kertész is a PhD student at the University of Szeged, Hungary and also a researcher at the Laboratory of Parallel and Distributed Systems of MTA SZTAKI Computer and Automation Research Institute, Hungary. He graduated as a program-designer mathematician, his research interests include grid brokering, scheduling and web services. He is participating two leading European projects S-Cube and CoreGRID Network of Excellence, and also a member of the CoreGRID Institute on Resource Management and Scheduling.