Comparison of Global Computing with Grid Computing

Babak Behsaz, Pooya Jaferian, Mohammad Reza Meybodi
Department of Computer Engineering and Information Technology, Amirkabir University of Technology (Tehran Polytechnic), 424 Hafez ave., Tehran, Iran
{behsaz, jaferian, meybodi}@ce.aut.ac.ir

Abstract

In recent years, Global and Grid Computing emerge as two powerful technology trends. In this paper, we compare these two approaches of distributed computing. First, we present a definition for Global Computing that accentuates the key point in this trend. This key point distinguishes Global Computing from other trends and covers many such systems. Second, we contrast two approaches in general characteristics. Then, by comparing them in technical issues, we show that the key point in our definition of Global Computing is the main source of many technical differences between these methods. Finally, we present our opinion about the probable future of Global and Grid Computing.

1. Introduction

Dramatic progress of network technology has led to growing interests in distributed computing approaches in recent years. As a result, Grid Computing and Global Computing (GC) came out as appealing research areas. These approaches try to address the problem of utilizing idle distributed resources connected via a network. According to the current practical uses of Grid Computing, fulfilling requirements of a virtual organization is the key point of Grid Computing [1, 2].

Due to lack of a clear distinction between GC and Grid Computing, a need for comparison of these approaches has raised. This comparison can reveal motivations, strengths and weaknesses of each approach that may help better understanding and selecting of current methods to apply on real world problems. In this paper, we aim to address this issue from a practical perspective. This means we will compare GC and Grid Computing in the way they are in real systems, not in the way theoretically assumed to be. The only work that is to some extent close to ours has expressed some general similarities and differences between Peer-to-Peer and Grid Computing [3].

This paper is structured as follows. In the next section, we specify and fix the definition of GC. In section 3, we state general differences and similarities of GC and Grid Computing. These approaches will be compared in technical issues in section 4. Section 5 addresses the opinion of authors about probable future of Grid Computing and GC. Finally, last section discusses our concluding remarks.

2. Global Computing

To have a logical comparison of GC and Grid Computing, we should firstly specify what we mean by GC. In contrast with Grid Computing, there are few attempts to present a precise definition of GC and we observed this term is used in various meanings which are contrasting in some cases. Some systems considered GC as a computing approach for computing by computers distributed over the world like in [4, 5, 6]. Some considered it a method for computing in a wide area network as in [7, 8, 9]. Others assumed it an approach that can be used for computing in the Internet like in [10, 11, 12]. All these considerations have explicitly or implicitly assumed that the GC system will be used for computation in a network of computers scattered world wide. This common point leads to our definition of GC: "Global computing is a distributed computing approach for computing with a large collection of heterogeneous computers which are *essentially* scattered over the world and sharing various kinds of resources."

This definition is very similar to the definition of Grid Computing and some other computing approaches, but in fact, it has an important distinction with them. Firstly, we are emphasized that computing entities must be scattered over the world. Secondly, we have not posed any other constraints in this trend of computation like benefiting a (virtual) organization in Grid Computing. Therefore, despite vast intersection of

Proceedings of the Seventh International Conference on
Parallel and Distributed Computing,Applications and Technologies (PDCAT'06)
0-7695-2736-1/06 $20.00 © 2006 IEEE

GC with some similar technology trends, none of them cover GC or is covered by it. Thus, every effort for generalizing other similar computing technologies to cover GC will result in loss of efficiency at least in some cases. As we discuss in section 4, the mentioned key point causes some remarkable differences in technical issues (or even creates new technical issues) to reach high performance.

3. Comparing in General Characteristics

In this section, we compare GC and Grid Computing systems in general qualities. By general qualities, we mean characteristics which can be observed in current systems without going deep in their behind technical counter parts.

Knowledge of users: Typically, users of GC systems have less professional knowledge than users of Grid Computing Systems that commonly have an organizational relationship and some extent of knowledge as members of the organization.

Resources: On the one hand, most of the resources involved in a Grid Computing are more powerful and better connected than the common GC resources. On the other hand, number of resources of a GC system is much more than number of resources in a Grid.

Applications: Currently, successful applications on a GC system have two main characteristics. They have limited dependency to data transition and require great computation power.

Services and Standards: GC systems provide fewer and simpler services to their users. In contrast, many works are done for standardization of Grid technologies like OGSA [14]. Moreover, lack of standardization in GC has led to little generality of these systems and most of them are special-purpose.

Algorithms: Parallel Algorithms for GC systems are typically asynchronous and simple (usually, they just partition the problem space), because remarkable delay in communication makes a synchronous algorithm impractical.

4. Comparing in Technical Issues

To compare technical issues in Grid and GC systems, we should first define quality attributes that should be satisfied in each system. For Grid [2], the main quality attributes are performance, availability, and interoperability along with providing user with as much functionality as possible. In contrast, respecting our definition, main quality attributes of GC are scalability, usability, security, and fault tolerance. These differences have a significant impact on architecture and technical issues of each system.

Architecture: Currently, most GC systems are based on reference architecture of Grid and it can be seen that most of GC systems follow the five layer architecture proposed in [2] to some extents, but the ingredients of each layer differ in two systems according to the differences between quality attributes.

First, in the fabric layer, goal of Grid systems is to make many kinds of resources as transparent as possible. But in GC, doing this makes the system very large in size, and therefore it contradicts usability quality attribute of GC. Second, in the connectivity layer, Grid systems implement some authentication and encryption mechanisms. But due to the scale of the GC systems, implementation of such mechanisms is not feasible. Also in connectivity layer, GC systems should implement some fault tolerance mechanisms to manage high node volatility. Third, in resource layer, with respect to high performance and functionality, Grid embodies different techniques for resource monitoring, reservation, and managing QoS. In GC, due to lack of control on resources, simple resource monitoring and management techniques are implemented. Fourth, in collective layer, both systems implement scheduling, brokering, and directory services. But in Grid, other services such as data replication, workload management, diagnosis, etc. is implemented. These services have not been implemented on famous GC systems due to limitations of their lower layers. Finally, the application layer in two systems does not differ significantly.

Usability: In Grid, users are members of professional communities or (virtual) organizations. Therefore, they are typically forced to share their resources and also learn how to work with the Grid system in order to achieve the virtual organization's goal [3]. In contrast, in GC, because system requires as many resources as possible, it should attract more users even whom are less professional and have their own self interests. Therefore, usability is one of the most important attributes, which leads to three dominant properties. First, the client application of GC systems must be easy to use. Second, most parts (installation, configuration, etc.) of the system should work automatically. Third, the client application for these systems must be highly portable (e.g. by using java).

Security: This quality attribute is important in both systems. But, tactics for achieving security in GC differ from Grid. The virtual organization nature of Grid makes it possible to identify each participant and maintains moderate level of trust. Yet, in Global environment due to the scale and usability requirements, it's hard to identify each user. Therefore, absence of trust is the main security issue in GC. In famous Grids such as OGSA based Grids [16] the main security concerns are identification of users, privacy of

Proceedings of the Seventh International Conference on
Parallel and Distributed Computing,Applications and Technologies (PDCAT'06)
0-7695-2736-1/06 $20.00 © 2006 IEEE

data and enforcement of security policies which are addressed by using cryptographic algorithms and authentication and authorization mechanisms. In contrast, in GC, two major security issues exist as a result of untrusted environment. On the one hand, participant computers privacy and integrity should be maintained. In [10, 15], Sandboxing technique and in [11, 17] mechanisms such as software fault isolation and Secure Remote Helper Applications [17] are used to achieve this goal. On the other hand, some clients cannot tolerate that untrusted hosts access to their data and algorithms. To deal with this problem, [10] suggests using encrypted computing [18] and decomposition of pieces of data and code that should be executed in Global environments.

Incentive issues: In Grid Computing environments, the users' incentive is achieving virtual organization's goal. Therefore, there is usually no need for charging a user for using the Grid and reward a user for sharing his resources. Yet, in many GC environments, a user's self-interests should be satisfied. Therefore, a need for incentive mechanisms exists. The general idea for providing an incentive mechanism is to use economic models. For example, in SuperWeb [10], the need for economic model is addressed using an accounting module in the broker. Also, using market-based resource allocation in GC environments is suggested in [11]. Another idea is to attract users by fame [13], competition, etc.

Correctness: This issue is one of the aspects of fault tolerance. Comparing to Grid Computing environments, in GC, there is a doubt about the correctness of the result sent by the workers. There are two reasons for such behavior: First, economic incentive nature of many GC environments, raises the probability of cheating of the workers in generating results. Second, unknown and even non-standard computing resources of some workers (e.g. old Pentium processors), or changed applications may lead to generating faulty results from workers. In [10], some solutions for detecting such behavior are proposed. Also, [19] suggests methods for eliminating such workers in scheduling algorithms.

Resource allocation: In Grid environments, the availability quality attributes is satisfied. Therefore, we can suppose that the resources are available and reliable. In contrast, in Global environments, the resources are not available and reliable. As a consequence, allocation techniques in Global environments should deal with the problem of availability and reliability. Thus, the allocation strategies of Grid system do not address main GC concerns. For example, in contrast with Grid, scheduling techniques of GC should not put a high overhead on the system. Therefore, adaptive scheduling techniques [20, 21, 22, 23] and methods based on statistical modeling [24, 25, 26, 27, 28] which are introduced for Grid systems, are not applicable for Global systems. To deal with the problem of scheduling, most of the efforts are concentrated on finding easy to measure metrics for resource classification. For example, some simple metrics for availability and reliability is presented in [19]. In addition, some metrics for measuring performance of workers are introduced in [12].

5. Future of Global and Grid Computing

In previous sections, we compared GC with Grid Computing. Due to expressed differences, it seems that GC and Grid Computing are going to the different ways. But, the abilities of each system are so attractive that it will cause a gradual movement of these technologies toward each other. On the one hand, GC communities seek more sophisticated services and standards, and higher quality of service. On the other hand, Grid Computing communities want to reach scalability of GC systems and consequently, their large computational power. For example, in one of the ongoing GC systems, XtremWeb, every evolution of the system conducts it closer to Grid systems [15].

To reach the goal of combining the abilities of these two approaches, we predict some efforts will be made to obtain some general standards that unify necessary services of both approaches. These standards are general in the sense that they will relax limiting assumptions like world wide scalability in Global Computing systems or organizational control in Grids. After this standardization and bringing the combined approach to life, great efforts will be started to extend large volume of works in these two approaches to new combined approach.

To facilitate evolution process to combined general approach, we suggest a two layered framework. The first layer of this framework provides the interfaces required by unified services in combined approach. This is the layer that developers interact with for using the system. The second layer *separately* deals with implementation of the services for the case of world wide scalability and for the case of organizational control. We believe that implementation of most services must differ in each case, but this difference must be hidden from developers.

6. Conclusion

Global and Grid Computing are distributed computing approaches which utilize unused computer resources connected by a network. Although the main idea is similar, they are following it from different directions. GC systems are trying to refine their standards and Grids systems are seeking to reach more scalability. Due to more growth of Grid systems, it is sometimes assumed that Grid technology can be applied for GC tasks. But, due to the definition and consequent comparison we presented in this paper, we disagree with such opinions. Because the world wide scalability of GC systems leads to a great technical differences that needs new specialized methods to avoid loss of efficiency.

7. References

[1] L. Ferreira et al., *Introduction to Grid Computing with Globus*, IBM, New York, 2003.

[2] I. Foster, C. Kesselman, and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", *Intl. J. of Supercomputer Applications*, *15(3)*, 2001, pp. 200-202.

[3] I. Foster, and A. Iamnitchi, "On Death, Taxes, and Convergence of Peer-to-Pear Computing and Grid Computing", *2nd Intl. Workshop on P2P Systems*, 2003.

[4] J.M. Bahi, S.C. Vivier, and R. Cuturier, "Asynchronism for Iterative Algorithms in a Global Computing Environment", *In Proc. of the 16th IEEE Intl. Symposium on High Performance Computing Systems and Applications*, 2002, pp. 90-97.

[5] M. Sato, H. Tezuka, A. Hori, Y. Ishikawa, S. Sekiguchi, H. Nakada, S. Matsuoka, and U. Nagashima, "Ninf and PM: Comunication Libraries for Global Computing and High Performance Cluster Computing", *J. of Future Generation Computer Systems*, *13*, 1998, pp. 349-359.

[6] R. Wolski, D. Nurmi, and J. Brevik, "Models and Modeling Infrastructures for Global Computational Platforms", *In Proc. of the 19th IEEE Intl. Parallel and Distributed Processing Symposium*, 2005, pp. 224-231.

[7] K. Aida, A. Takefusa, H. Nakada, S. Matsuoka, and U. Nagashima, "A Performance Evaluation Model for Effective Job Scheduling in Global Computing Systems", *In Proc. of the 17th IEEE Intl. Symposium on High Performance Distributed Computing*, 1998, pp. 352-353.

[8] H. Nakada, M. Sato, and S. Sekiguchi, "Design and Implementations of Ninf: towards a Global Computing infrastructure", *J. of Future Generation Computer Systems, 15*, 1999, pp. 649-658.

[9] M. Sato, and K. Kusano, "NetCDF: a Ninf CFD componenet for Global Computing, and its Java applet GUI", *In Proc. of the 4th IEEE Intl. conference on High Performance Computing*, 2000, pp. 501-506.

[10] A.D. Alexandrov, M. Ibel, K.E. Schauser, and C.J. Scheiman, "SuperWeb: Towards a Global Web-Based Parallel Computing Infrastructure", *11th IEEE Intl. Parallel Processing Symposium*, 1997, pp. 100-106.

[11] P. Cappello, B.O. Chiristiansen, M.O. Neary, and K.E. Schauser, Market-Based Massively Parallel Internet Computing, *In Proc. of the IEEE Conference on Massively Parallel Programming Models*, 1997, 118-129.

[12] D. Kondo, H. Casanova, E. Wing, F. Berman, "Models and Scheduling Mechanisms for Global Computing Applications", *In Proc of the IEEE Intl. Parallel and Distributed Processing Symposium*, 2002, pp. 216-229

[13] S.M. Larsen et al., "Folding@Home and Genome@Home: Using distributed computing to tackle previously intractable problems in computational biology", *J. of Computational Genomics*, 2003.

[14] I. Foster, C. Kesselman, J. Nick, and S. Tuecke, "Grid Services for Distributed System Integration", *IEEE Computer Journal, 35(6)*, 2002, pp. 37-46.

[15] F. Cappello et. all, "Computing on large-scale distributed systems: XtremWeb architecture, programming models, security, tests and convergence with Grid", *Future Generation Computer Systems, 21*, 2005, pp. 417-437.

[16] P.Z. Kunszt, and L.P. Guy, "The Open Grid Services Architecture, and Data Grids", *In : Berman, F., Fox, G., Hey, T., (ed): Grid Computing: Making the Global Infrastructure a Reality*, John Willey & Sons Inc, 2003, pp. 385-408.

[17] I. Goldberg, D. Wagner, R. Thomas, and E. Brewer, "A Secure Environment for Untrusted Helper Applications: Confining the Wily Hacker", *In Proc. of the 1996 USENIX Security Symposium*, 1996

[18] J. Feigenbaum, "Encrypting Problem Instances - Or, ..., Can You Take Advantage of Someone Without Having to Trust Him?", *In Proc. of the 5th IACR Conf. on Crypto*, 1985, pp. 477-488.

[19] M. Taufer, P.J. Teller, D.P. Anderson, and C.L. Brooks, "Metrics for Effective Resource Management in Global Computing Environments", *In Proc. Of the 1st IEEE Intl. Conference on e-Science and Grid Computing*, Melbourne, 2005, pp. 204-211.

[20] E. Heymann, M.A. Senar, E. Luque, and M. Livny, "Adaptive Scheduling for Master-Worker Applications on the Computational Grid", *In Proc. of the First IEEE/ACM International Conference on Grid Computing (GRID'00)*, 2000, pp. 214-227.

[21] F. Berman, et al., "The GrADS Project: Software Support for High-Level Grid Application Development", *Int. J. High Performance Computing Applications, 15(4)*, 2001, pp. 327-344.

[22] S.S. Vadhiyar, and J.J. Dongarra, "Self Adaptivity in Grid Computing", *Concurrency and Computation: Practice and Experience, 17(2-4)*, 2004, pp. 235-257.

[23] F. Berman, et al., "New Grid Scheduling and Rescheduling Methods in the GrADS Project", *Int. J. of Parallel Programming, 33(2)*, 2005, pp. 209-229

[24] J. Brevik, D. Nurmi, and R. Wolski, "Automatic Methods for Predicting Machine Availability in Desktop Grid and Peer-to-Peer Systems", *In Proc. of the IEEE Intl. Symp. on Cluster Computing and the Grid*, 2004, pp. 190-199.

[25] D. Nurmi, J. Brevik, and R. Wolski, "Minimizing the Network Overhead of Checkpointing in Cycle-harvesting Cluster Environment", *In Proc. of the IEEE Intl. Conference on Cluster Computing (Cluster 2005)*, 2005.

[26] D. Nurmi, J. Brevik, and R. Wolski, "Modeling Machine Availability in Enterprise and Wide-area Distributed Computing Environments", *Technical Report CS2003-28 U.C. Santa Barbara CS Department*, 2003.

[27] D. Kondo, M. Taufer, C.L. Brooks, H. Casanova, and A.A. Chien, "Characterizing and Evaluating Desktop Grids: An Empirical Study", *In Proc. of the IEEE/ACM Intl. Parallel and Distributed Processing Symposium (IPDPS'04)*, 2004, pp. 17-25.

[28] D. Kondo, A.A. Chien, and H. Casanova, "Resource Management for Rapid Application Turnaround on Enterprise Desktop Grids", *In Proc. of the ACM Conference on High Performance Computing and Networking*, 2004.

Proceedings of the Seventh International Conference on
Parallel and Distributed Computing,Applications and Technologies (PDCAT'06)
0-7695-2736-1/06 $20.00 © 2006 IEEE