

GCaaS: A Light-weight Grid Computing Web Application

류샤오, 우균
부산대학교 전자전기컴퓨터공학과
e-mail : {liuxiao,woogyun}@pusan.ac.kr

GCaaS: A Light-weight Grid Computing Web Application

Xiao Liu, Gyun Woo
Dept. of Electronic and Computer Engineering, Pusan National University

Abstract

Recent global and cloud computing render the Internet and Web application to be a paramount field since it is uncomplicated to access and less time and space limitation. On the other hand, a growing number of computations using grid computing techniques indicates the requirements and quantities of large-scale computations are becoming foremost progressively. Therefore, that will be much practicable if there is a Web-based service that could provide Grid computing functions. In this paper, Several similar Web-based cloud and parallel computing systems will be discussed and a model of Web application termed GCaaS which supports grid computing services will be introduced.

1. Introduction

Grid computing provides outstanding performance at large-scale computations and complex calculations. Grids started off in the mid-90s to address large-scale computation problems using a network of resource-sharing commodity machines that deliver the computation power affordable only by supercomputers and large dedicated clusters at that time [1]. There are several of complicated fields such like DNA computing, volcanic activity, climate change, and graphics computation, which need be solved accurately. Since the applications in these fields cannot be processed at only one processor or computer, a grid computing system, which consists with several computers or processors, is able to solve problems together is necessary.

Nevertheless, it is a fairly tedious work to deploy a grid computing system [2]. Besides, there are few kinds of friendly user interface for accessing existed grid computing systems, which means it is laborious that try to using grid computing technologies as non-professional-grid-computing developers. However since there is a growing number of large-scale computing usages [3], it is necessary to build an anyone-easy-to-use grid computing system for those people who need deal with large-scale computational problems.

In this paper, we will introduce a new system model for supporting grid computing application on Web termed GCaaS. GCaaS stands for Grid Computing as a Service. This model presents a concise structure which includes a background multiple task processing framework, a task distributing mechanism, a friendly responsive design Web front-end user interface.

We will discuss several related and similar systems in the second section, the third section will illustrate the details of our GCaaS model. Furthermore, an experiment implementation system will be discussed termed GCaaS-1.0

to verify the practicability and feasibility of GCaaS. Ultimately, the future work will present the plan to develop this system with more convenient functions and the summary with several issues from development procedure will be exhibited either.

2. Related Work

2.1. XtermWeb

XtermWeb is a cloud computing model for parallel computing. This system approximately separated as root server, workers LAN and results collector. The server distributes jobs to the workers which connected by LAN and workers return results to collector when jobs have been complete [4]. XtermWeb is based on a model of distributed system and able to append or detach workers groups. Another feature of XtermWeb has its own communication protocol to support stable and secure data transmission.

The XtermWeb seems like an initial model of Web based parallel computing service and also emerged several problems and disadvantages such like non-friendly UI and since XtermWeb is a complex distributed system it is ambitious to setup for a non-professional developer.

2.2. GridSim

GridSim is a toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing [5]. GridSim could simulate different types of resources in a grid network and with no limit on the number of task jobs.

However, the heterogeneous architecture of GridSim system courses the obstacle of system portability, it means GridSim could not be redeployed conveniently to another computer when building a new grid computing environment.

3. System Architecture

In this section, the system architecture of GCaaS will be presented by its consisted modules. The background processing module of GCaaS was developed based on an open source parallel computing framework named JPPF [5], to provide system accessibility for multiple platform we chose an open source responsive design Web framework called Webplate [6]. Since the JPPF framework runs at Java environment we built a JSP server for our Web application. The next three subsections will illustrate the deployment of grid computing network, processing procedure of system back-end and the Web front-end of GCaaS.

3.1. Grid Computing Network Deployment

The JPPF framework provides a main processing module of GCaaS for application jobs and since it could support parallel computing that is possible to run multiple jobs at a same time. We optimized the existed JPPF framework to suit GCaaS and Figure 1 illustrated the optimized JPPF architecture of GCaaS.

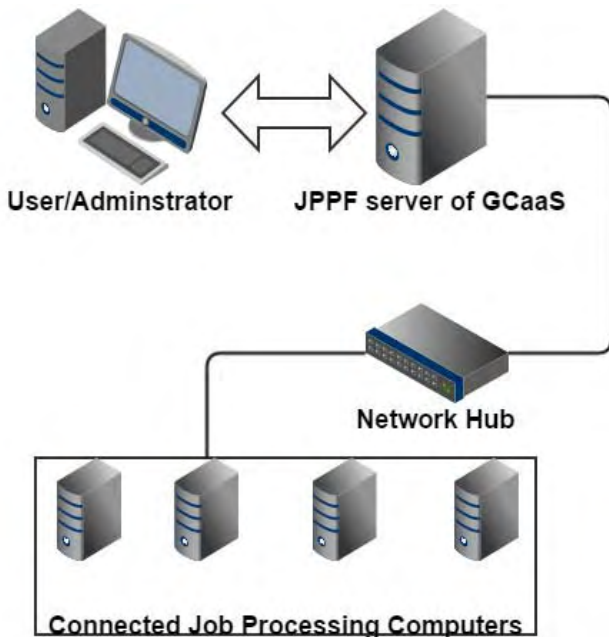


Figure 1 JPPF architecture of GCaaS

General users and system administrator could access the JPPF server of GCaaS to add Application Jobs (AJ) or manage the connected Job Processing Computers (JPC) through a network hub program. Several JPCs could construct a JPC group and since parallel computing is available in JPPF framework the JPC group is able to process complicate AJs synchronously. Furthermore, the JPPF server of GCaaS could also be considered as a JPC and participate the AJ processing procedure.

3.2. Processing Procedure of System Back-end

GCaaS is able to add an extra external computer as a JPC when it connected the grid network and it also is able to add several JPCs as a JPC group to process AJs with existed JPC group together. In the other hand, it is certainly able to

reduce or disconnect JPC or JPC group and without changing system configuration. The excellent expandability leads GCaaS could suit almost any of situations whatever the computable resources are much or less. Figure 2 presented the procedure of GCaaS system during processing an Application Job.

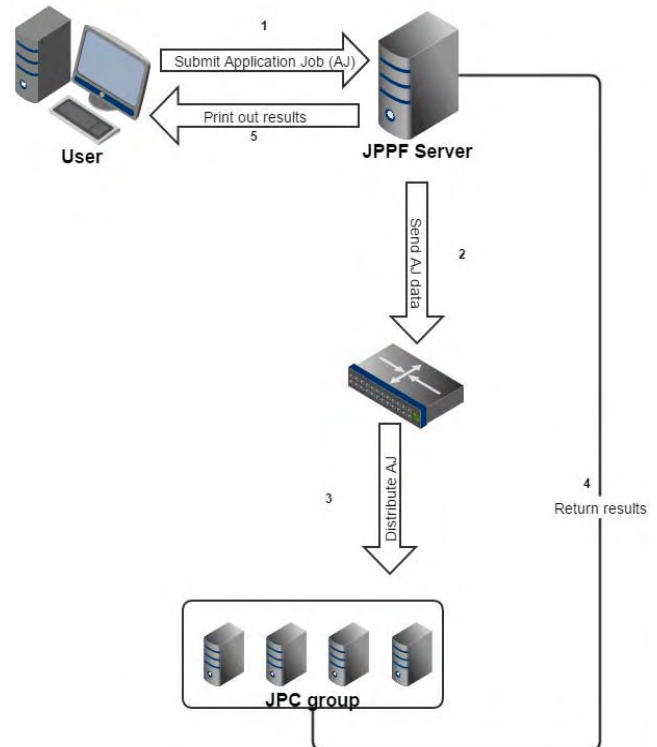


Figure 2 AJ processing procedure

As shown as Figure 2, the user sends AJ (Application Job) to JPPF server and JPPF server transmit AJ to network hub program, the hub will distribute AJ to connected JPC or JPC (Job Processing Computer) group. JPC will return processed results to JPPF server when AJ was solved and JPPF server will print out the result to user. JPPF server could serve as a JPC to process AJ if there is no connected JPC exists or all the JPCs are busy.

3.3. Responsive Web Front-end Design

To provide accessibility of GCaaS system for different platforms such as PC, smart-phone and tablet we designed a responsive Web front-end. Responsive Design. New web standards are making responsive design practical, allowing designers to create a single website that can adapt layout and content to viewing contexts across a spectrum of digital devices and the result should be a more satisfying experience for any user [7]. There are plenty of Responsive Web Design frameworks such as Groundwork CSS [8], MUELLER Grid System [9], Cool Kitten [10], etc. We chose a concise Responsive Web Design framework named Webplate to build a friendly Web front-end UI of GCaaS system. Webplate is an open source Responsive Design framework based on Apache License 2.0 [6]. Figure 3 presented the main page of GCaaS, users and system administrator could access the GCaaS Web page and login to the system, adding new Job Processing Computer (JPC) or submit a new Application Job

(AJ) is available during this website. Figure 4 is the view of accessing main page of GCaaS system through a smartphone and all the functions are able to be used when using any of non-PC browsers.

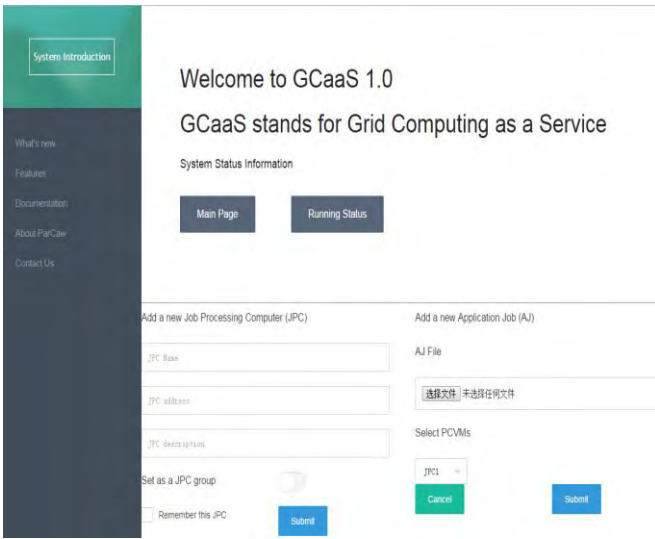


Figure 3 Main page of GCaaS System



Figure 4 Main page view in smartphone

4. Features of GCaaS system

In this section, we will discuss all the features of GCaaS from its mobility, easy-portability, light-weight property, security and flexibility.

4.1. Mobility

Since GCaaS supports Responsive Web Design fairly well the management and use of system will not be limited at traditional PCs, people can access the GCaaS system at anywhere in anytime by any of platforms without system performance changes. The mobility provides a much more

convenient approach than other similar systems.

4.2. Easy-portability

GCaaS does not depend on an installation procedure before starting running it The JPPF server could be set-upped by running a short shell command and it is also handy to add a Job Processing Computer (JPC) that just start a processing program meanwhile the connection with JPPF server will be connected by network hub program automatically. Not only that, to optimize the information storage space of Job Processing Computer (JPC) and Application Job (AJ) we built a database system to storing those data. Figure 5 presented the GCaaS architecture of the connection of the processing system and database. That will be much easier when porting GCaaS system to another environment because including such as system configuration information, network information, JPC information, AJ data and results all of data just need to insert a new database without reconfiguring and restoring.

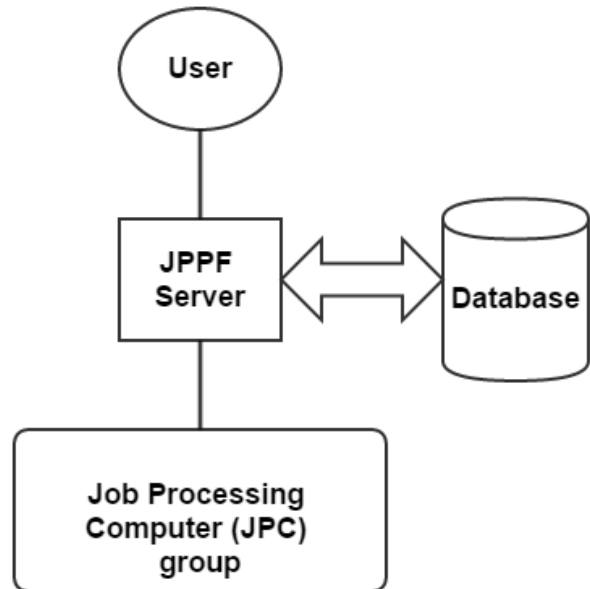


Figure 5 Connection of Job processing system and Database

4.3. Light-weight Performance

The whole scale of the GCaaS system is under 50 MBs including a JPPF server, Job Processing Computer (JPC) and database configuration files. This kind of system scale saves lots of computer and storage resources but still could provide grid computing services [11].

4.4. Security

To protect the Job Processing Computer (JPC) and stored data from intent attack, we use a firewall to avoid those job processing related modules being exposed straightforwardly on the Internet. Figure 6 demonstrated the full structure of GCaaS system with all system components. The protection of JPPF server was also be considered by adding a user authentication module.

4.5. Flexibility

As we discussed at section 3.2, GCaaS system is available at both of expanding or reducing Job Processing Computer (JPC) to suit different computation requirements and

environments, it is quite efficient to save unnecessary resources and spaces. Figure 7 illustrated the device status page of GCaaS, user is supported to manage all the devices including the JPPF server such as stop/run, connect/disconnect and delete. As shown as Figure 7, both of machine 1 and machine 2 are connected in the same JPC group and that will be an independent JPC if the JPC Group prints 0 like machine 2 and 3.

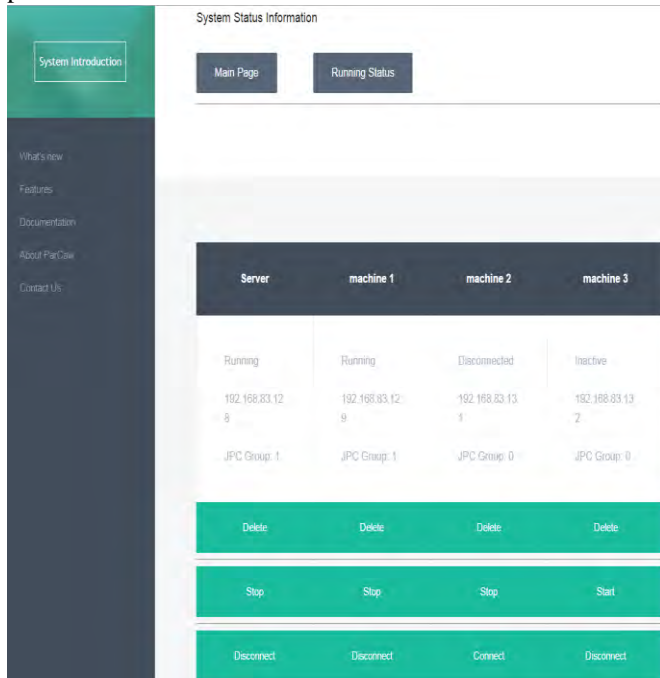


Figure 7 Connecting status of GCaaS system

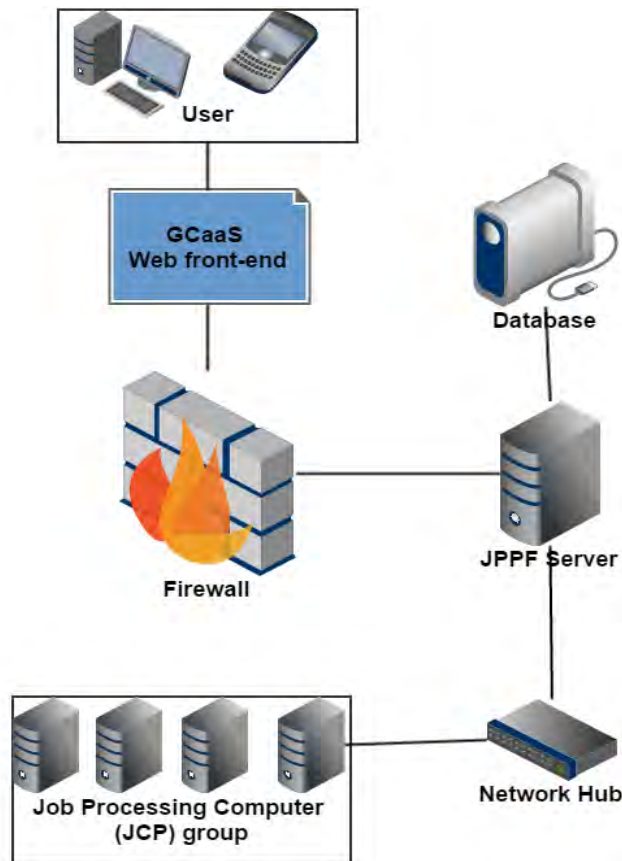


Figure 6 Full architecture of GCaaS system

5. Conclusion & Future Work

Grid computing provides an outstanding performance at large-scale computations. In the other hand, since could computing and SaaS represent be the general trend we put these two technologies together and built a new system model for provide grid computing services on a Web application named GCaaS. By discussing the several similar existed systems and introducing the architecture of GCaaS the feasibility and practicability were verified. The Features of GCaaS presented several advantages of GCaaS and the structure of GCaaS system with full components was also be discussed. We will keep developing and optimizing GCaaS system to become a veritable powerful grid computing Web application.

Acknowledgment

This work was supported by BK21PLUS, Creative Human Resource Development Program for IT Convergence.

*Corresponding should be addressed to Gyun Woo (Email: woogyun@pusan.ac.kr)

Reference

- [1] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in Grid Computing Environments Workshop, 2008. GCE'08. Ieee, 2008, pp. 1–10.
- [2] R. Sen. (2008) Developing parallel programs [Internet], <https://msdn.microsoft.com/en-us/library/cc983823.aspx>.
- [3] N. Hemsoth. (April 15, 2013) Future challenges of large-scale computing [Internet], http://www.hpcwire.com/2013/04/15/future_challenges_of_large-scale_computing/.
- [4] G. Fedak, C. Germain, V. Neri, and F. Cappello, "Xtremweb: A generic global computing system," in Cluster Computing and the Grid, 2001. Proceedings. First IEEE/ACM International Symposium on. IEEE, 2001, pp. 582–587.
- [5] J. org. (2014) Jppf home [Internet], <http://www.jppf.org/>.
- [6] Chrishumboldt. (2014) Get Webplate [Internet], <http://getwebplate.com/>.
- [7] Gardner B S. Responsive web design: Enriching the user experience [J]. Sigma Journal: Inside the Digital Ecosystem, 2011, 11(1): 13-19.
- [8] groundwork (2014) Say hello to responsive design made easy [Internet], <https://github.com/workgroundcss/workground>.
- [9] muellergridsystem (2015) MODULES [Internet], <http://muellergridsystem.com/>.
- [10] cool-kitten (2013) a parallax scrolling responsive framework [Internet], <https://github.com/jalxob/cool-kitten>.
- [11] Volz, Raphael, Daniel Oberle, and Rudi Studer. "Implementing views for light-weight web ontologies." Database Engineering and Applications Symposium, 2003. Proceedings. Seventh International. IEEE 2003.