Solving a Problem in Grid Applications: Using Aspect Oriented Programming

Hyuck Han, Shingyu Kim, Hyungsoo Jung, and Heon Y. Yeom

School of Computer Science and Engineering,
Seoul National University,
Seoul 151-742, Korea
{hhyuck,sgkim,jhs,yeom}@dcslab.snu.ac.kr

Abstract. Aspect Oriented Programming (AOP) was introduced 10 years ago and many research projects have focused on broadening AOP and its target areas. However, few applications in the Grid computing world adopt AOP in comparison with very vigorous research of AOP. Therefore, we present a case study that covers a general networking problem in the Grid computing. AOP provides a novel solution of the problem without modifying existing source code. Aspects that we define are simple, intuitive and reusable. We believe that our implementation is very useful in developing other Grid computing software platforms, and AOP can be a powerful method in modularizing source codes and solving problems of software architectures.

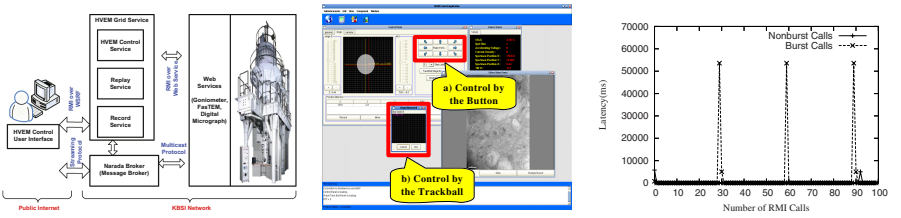
1 Introduction

Many scientists in e-Science [1] currently utilize computing resources as part of their research and will utilize more powerful computing resources across the Grid [2] infrastructure. They also have access to very large data sets and are able to perform real-time experiments through Grid applications. Distributed global collaborations over the Internet such real-time experiments require high bandwidth and low latency. However, firewalls in front of networks of research institutes often show unexpected long latency because many SOAP (Messaging Protocol of Grid computing) messages in a short period can be regarded as a DoS attack. Therefore, we devised a solution to overcome the problem. Then, we added a new module without any modification of legacy software by utilizing the Aspect Oriented Programming (AOP) [3] technique.

AOP is a new technology for separating crosscutting concerns that are usually hard to do in object-oriented programming (OOP). AOP complements OOP by allowing the developer to dynamically modify the static OO model to create a system that can grow to meet new requirements. Just as objects in the real world can change their states during their lifecycles, an application can adopt new characteristics as it develops.

In this paper we describe our AOP-based solution to overcome the firewall problem. Aspects that we define are simple, intuitive and reusable enough to be applied to other Grid applications.

Y. Shi et al. (Eds.): ICCS 2007, Part IV, LNCS 4490, pp. 705–708, 2007.



(a) System Architecture (b) User Interface (c) Latencies of RMI

Fig. 1. HVEM Grid System and its Problem

2 Overall System

2.1 Motivation

High-Voltage Electron Microscope (HVEM) allows scientists to see objects at a magnification greater than the actual sample, and we use HVEM that the Korea Basic Science Institute (KBSI)[4] operates. Figure 1(a) shows the architecture of the HVEM Grid System. Our system consists of three tiers - HVEM Control User Interface (HVEM Control UI), HVEM Grid Service and Web services which encapsulate Vendor-provided applications. XML messaging over HTTP is used to communicate between HVEM Control UI and HVEM Grid Service. KBSI operates firewalls between the public internet (HVEM Control UIs) and the KBSI network (HVEM Grid Service and legacy systems). Figure 1(b) shows HVEM Control User Interface. Part (a) calls HVEM Grid Service for each mouse click, and part (b) calls it continuously while trackball is moving (numerous and continuous calls in a short period). In Figure 1(c), lines noted as Nonburst Calls and Burst Calls correspond to latencies of RMI by part (a) and part (b) respectively. In case of Burst Call, there are some peaks exceeding 50 seconds at 28th, 57th and 90th call. This is due to the firewall which is located in front of HVEM Grid Service.

When the transport layer of SOAP is HTTP, the flow of each RMI call follows.

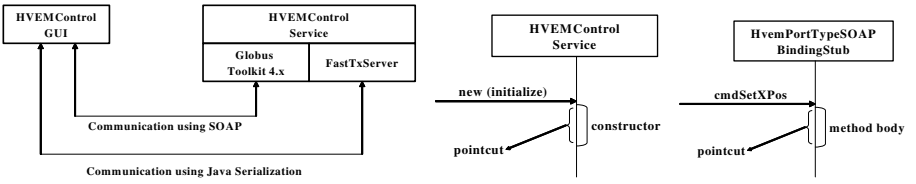
- Connect to host which runs Globus Toolkit
- Transfer XML-based SOAP messages to the corresponding connection
- Globus Toolkit invokes the requested method of the requested service.
- Globus Toolkit sends results of the method call to the connection
- Disconnect the connection

The burst of RMI calls by part (b) in Figure 1(b) suggests that the number of connection requests to the server host increases. The peaks in Figure 1(c) come from the feature that the firewall regards the burst as a DoS attack.

2.2 Improvements

Our primary goal of this study is to guarantee short and coherent latencies, but it is also important to minimize modifications of HVEM Grid Service and

HVEM Control User Interface. To achieve these goals, it is the best approach to add a new transport layer in the same level in which skeleton and stub are located in HVEM Grid Service and in HVEM Control User Interface.



(a) Improved System Architecture (b) Server-side pointcut (c) Client-side pointcut

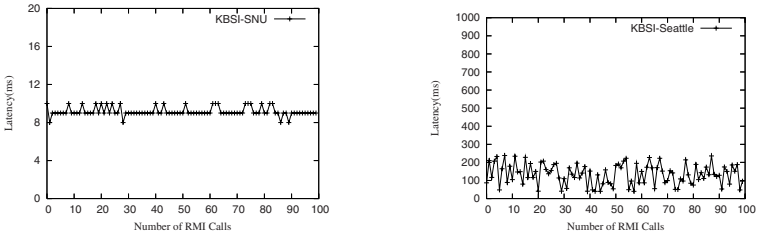
Fig. 2. Overall System

Figure 2(a) illustrates the improved system. In this system, communications between HVEM Control User Interface and HVEM Grid Service can use two ways. One is to use SOAP and HTTP and the other is to use Java Serialization. The former is for part (a) and the latter for part (b) in Figure 1(b). The server which utilizes Java Serialization is called FastTxServer.Unlike Globus Toolkit, the FastTxServer does not disconnect client after it responds to a request. This feature makes the firewall not to consider continuous calls as a DoS attack. To meet the goal of minimizing modifications, we capture execution points and add new functions in these points using AOP. In other words, we capture executions of HVEM Grid Service and HVEM Control User Interface, and replace them with desired methods. Figure 2(b) and 2(c) describe pointcuts in the program flow. In server-side, the constructor of HVEM Grid Service which captured as a pointcut is passed to the FastTxServer as argument. Then, the FastTxServer can execute a method through Java Reflection when a method call request is arrived at the FastTxServer. In client-side, entire methods, which generate the firewall problem, are captured as pointcuts. Then, invocations based on SOAP over HTTP are replaced with calls to the FastTxServer.

3 Evaluation

We evaluated our improved system using 3 PCs in Seoul National University (SNU), KBSI and Seattle. The machines were Pentium 4 2.80GHz in SNU and Seattle, and the machine was Pentium 4 3.0GHz in KBSI, running Linux 2.4.18. HVEM Grid Service server was in KBSI machine, and HVEM Control UIs were in SNU and Seattle. In case of Seattle, clients and the server are connected over the GLORIAD network[5].

Figure 3 shows latencies of RMI with optimization in stub and skeleton levels. Latencies were measured by calling RMI 100 times sequentially. The average latency was 10ms in SNU, and 120ms in Seattle. These short latencies are due



(a) Client in SNU (b) Client in Seattle

Fig. 3. Latencies of RMI with Improvement in Stub and Skeleton level

to the lack of XML translation in Java Serialization. In addition, firewalls of KBSI do not recognize any calls as DoS attacks. By this improvement, overseas users can use our HVEM Grid Service with a reasonable speed.

4 Conclusion

AOP is regarded as a powerful method for modularizing software and solving problems of software due to its attractive features. This article shows a case study that covers a firewall problem in Grid computing. We devised a AOP-based solution to overcome the problem without modification of legacy software or existing services and our results show that our improved system guarantees fast and coherent latencies.

Acknowledgment

The ICT at Seoul National University provides research facilities for this study.

References

1. Oxford e-Science Centre: (e-Science Definitions) `http://e-science.ox.ac.uk/public/general/definitions.xml`
2. Foster, I., Kesselman, C., Tuecke, S.: The anatomy of the grid: Enabling scalable virtual organizations. International Journal of SuperComputer Applications **15**(3) (2001)
3. Palo Alto Research Center: (The AspectJ(TM) Programming Guide) `http://eclipse.org/aspectj/`.
4. Korea Basic Science Institute: (KBSI microscopes & facilitates) `http://hvem.kbsi.re.kr/eng/index.htm`.
5. GLORIAD Korea: (GLORIAD-KR) `http://www.gloriad-kr.org/eng/index_eng.htm`.