An Extensible Scientific Computing Resources Integration Framework Based on Grid Service

Binge Cui¹, Xin Chen¹, Pingjian Song², and Rongjie Liu²

¹ College of Information Science and Engineering, Shandong University of Science and Technology, 266510 Qingdao, China
² First Institute of Oceanography, S.O.A. Xianxialing Road 6#, 266510 Qingdao, China
cuibinge@yahoo.com.cn

Abstract. Scientific computing resources (e.g., components, dynamic linkable libraries, etc) are very valuable assets for the scientific research. However, due to historical reasons, most computing resources can’t be shared by other people. The emergence of Grid computing provides a turning point to solve this problem. The legacy applications can be abstracted and encapsulated into Grid service, and they may be found and invoked on the Web using SOAP messages. The Grid service is loosely coupled with the external JAR or DLL, which builds a bridge from users to computing resources. We defined an XML schema to describe the functions and interfaces of the applications. This information can be acquired by users by invoking the “getCapabilities” operation of the Grid service. We also proposed the concept of class pool to eliminate the memory leaks when invoking the external jars using reflection. The experiment shows that the class pool not only avoids the PermGen space waste and Tomcat server exception, but also significantly improves the application speed. The integration framework has been implemented successfully in a real project.

Keywords: Grid Service; Integration Framework; Legacy Application Encapsulation; Class Pool; Reflection.

1 Introduction

With the rapid development of the Internet technology, more and more applications can be accessed through the Web, e.g., Google Documents, Picasa Web Albums, Google Maps and Earth, etc. Moreover, an exciting interactive Web application, Mashup [1], has emerged. It draws upon content retrieved from external data sources to create entirely new and innovative services. Web applications have brought too much convenience for the people. For example, most applications do not need to install the client, thus it is very easy to deploy and maintain them. However, many existing applications weren’t designed for use in the Web, which were called legacy applications as usual [2]. In order to exert the greatest value of the legacy applications and provide computing services for more people, these applications should be encapsulated and published in a simple and friendly way [3].

© Springer-Verlag Berlin Heidelberg 2009
Reflection is a powerful technique and can enable applications to perform operations which would otherwise be impossible. However, reflection also has drawbacks, such as performance overhead, security restrictions and exposure of internals. Because reflection involves types that are dynamically resolved, certain Java virtual machine optimizations cannot be performed. As a result, reflective operations have slower performance than their non-reflective counterparts. Moreover, because reflection needs to reload the external classes every time, and the PermGen space [4] that used to store the classes and class descriptions will not be released by the GC (garbage collection) mechanism, reflective operation swallows up the limited PermGen space very soon.

Our research motivation comes from a remote sensing data integration and sharing project. In this project, there are a variety of remote sensing images that need to be processed and shared. The processing algorithms, such as metadata extraction algorithm and cloud cover detection algorithm, have been made into EJB or COM components. These components have been proved to be stable and reliable after a long trial. The customer wants to reuse these computing resources in the project to protect the existing investment and shorten the development cycle.

2 Application Integration Framework Based on Grid Service

In order to integrate the legacy applications, we designed a legacy applications integration framework based on Grid service, as shown in Figure 1.

![Figure 1. Legacy Application Integration Framework Based on Grid Service](image)

The Grid service has embedded several kinds of adapters, such as the adaptor for EJB component, COM component, dynamic linkable library, etc. Grid service itself does not achieve any specific function. When received an operation request from the user interface, Grid service will query the applications registry, and invoke the corresponding legacy applications using the adaptors. After the applications finished their tasks, Grid service will receive the results and return them to the user interface. In a word, Grid service provides an abstract and common invoke interface to shield the differences among the various legacy application interfaces.