國 立 交 通 大 學

資訊管理研究所

碩 士 論 文

一個在 CORBA 架構下的負載平衡演算法

A Loading-Balance Algorithm for Improving Efficiency of CORBA

研 究 生：紀 浩 仙

指導教授：羅 濟 群 博士

中 華 民 國 九十三 年 六 月

一個在 CORBA 架構下的負載平衡演算法

A Loading-Balance Algorithm for Improving Efficiency of CORBA

研 究 生：紀 浩 仙　　Student：Haohsien Chi

指導教授：羅 濟 群　　Advisor：Chi-Chun Lo

國立交通大學

資訊管理研究所

碩士論文

A thesis

Submitted to Institute of Information Management

College of Management

National Chiao Tung University

In Partial Fulfillment of the Requirements

For the Degree of

Master of Business Administration

In

Information Management

June 2004

Hsinchu, Taiwan, the Republic of China

中華民國　九十三 年 六月

中文摘要

傳統的分散式環境中，最常被提及的特性之一就是其負載平衡的特點。在以 CORBA 這個由 OMG 提出的架構中，定義且規範了許多分散式環境的標準。其中負載平衡就是一個重要的特性，因此一些 ORB 發行廠商也將這特性加以實做，多數的廠商是以一個額外的 agent 來處理 loading balance 的工作，但是我們發現這樣的架構下有一個根本性的問題：當這個 agent 失敗的時候，所有的工作將全部受到影響。於是我們提出了這個簡化的架構，雖然我使用的是 Visibroker 所發行的 ORB，但是屏除了它的目錄和負載平衡服務，純粹以應用層面的模式，來處理這兩個問題。最重要的一點，我們把這個負載平衡的權利放到 client 端來做，也就是說，只要 client 提出請求，且不是所有的服務提供者都失敗的情形下，這個系統都可以存活。

藉由模擬實驗中，我們將所提出的架構加以多次反覆測試；並比較其和原先 Visibroker 隱含的架構。希望得到：

1. 在這樣的架構下，系統的表現是否真的能提昇？我們以服務的回覆時間作為測量的基礎。
2. 相較於以發行的套裝軟體服務，這樣的架構會不會使得應用程式寫作者覺得更困難、更難以撰寫，抑或沒有顯著的差別？也就是說，隱含性的架構比較方便或是這樣的應用層面架構無顯著的差異；這裡我們將以質化分析來列舉優劣。

為了解決上述的問題，我們將想法加以實做，並用實驗模擬加以證明。

Abstract

In the traditional distributed systems, the most popular characteristic is loading-balance. In CORBA which is an OMG proposed architecture, this characteristic is also pointed out. In many published ORBs, many vendors used additional agents to handle this characteristic. But we found there will be a problem with this solution, that is, if this agent fails, the whole system will not work . So we proposed a simplified model. We put this characteristic to be implemented on client's side. That is , if the client stands alive and not all service providers fail, the whole system will still be alive with this charateristic.

By simulations and some experiments, we will repeatedly test the proposed model. And compare this model with VisiBorker which is a published ORBs. And hope to get the answers of the following questions：

1. In this proposed model, will the system's performance be improved？This problem can be divided into two parts：One is system's loading status, and the other is system's response time.（efficiency）
2. Compare to the published CORBA software, will this model become more complicated to program？That is , this model is implemented on application layer, and some software embedded them in underlying architectures. What's difference when coding or programming？

To solve the above problem, we implemented this idea and used results of many experiments to get the answers.

Table of contents：

List of figures：

List of tables