

웹 유휴자원을 이용한 자율적 분산 컴퓨팅 환경

이기준*, 정채영**

*조선대학교 대학원 전산통계학과

**조선대학교 자연과학대학 전산통계학과

e-mail: cholee@shinbiro.com

Autonomous Distributed Computing Environment Using Web idle resource

Kee-Jun Lee*, Chai-Yeoung Jung**

*Dept. of Computer Science and Statistics, Graduate School, Chosun Univ.

**Dept. of computer Science and Statistics, College of Natural Sciences, Chosun Univ

요약

본 연구에서는 인터넷에 연결된 유휴자원을 자율적으로 묶어 하나의 분산시스템을 구축하고 이를 이용하여 사용자에게 병렬 프로그래밍 시스템을 설계, 구현하는 방법에 관하여 기술하였다 제안된 자율적 웹 분산 컴퓨팅 환경은 인터넷 컴퓨팅 구축에 있어서 필수적인 동적자원의 중계, 작업의 분배방법을 제시한다. 또한 제안된 시스템 구축방안은 지니 기술을 이용함으로써 자율적 웹 분산환경구축이 가능하였고 실험을 통하여 인터넷의 유휴자원을 이용함으로써 적은 비용으로 높은 생산성을 얻을 수 있는 가능성을 제시하였다

1 서론

최근 컴퓨터 하드웨어 기술의 급속한 발달과 더불어 인터넷의 발전속도는 가히 상상을 초월하는 수준에까지 이르렀다. 초기의 인터넷 접속의 목적은 단순한 정보의 수집과 배포였지만 이제는 인터넷에 연결된 수천만의 컴퓨터의 자원을 활용하는 분야로 확장되었다. 따라서 시스템 자원활용을 위하여 애플릿(Applet)과 같은 플랫폼 독립적인 기술을 바탕으로 한 안전한 웹 분산 시스템의 구축에 관한 연구가 활발히 진행되고 있다. [1][4][5]

본 논문에서는 인터넷에 연결된 유휴자원을 자율적으로 묶어 하나의 분산시스템을 구축한 후 사용자에게 병렬 프로그래밍 환경을 제공하여주는 시스템을 설계하고 구현하는 방법에 관하여 연구하였다. 이러한 웹 분산 컴퓨팅 환경구축과 관련된 대표적인 연구로는 Javalin[1], ParaWeb[2], Charlotte[3] 등이 있으며 이들은 웹브라우저를 이용하여 사용자로 하여금 병렬 분산 환경을 제공하며 플랫폼 독립적인 시스템 구축을 위하여 Java[6]를 사용하였다. 그러나 이들은 각각 병렬 프로그램 작성의 어려움과 확장성, 각 작업간

의 공유메모리를 통한 통신을 지원하지 못한다는 단점을 지니고 있다. 따라서 본 연구는 이러한 방법의 문제점들을 해결하기 위하여 지니기술을 이용하여 인터넷에 연결된 이기종간의 확장성 및 자율적인 시스템구성과 효율적인 웹 컴퓨팅환경 구축방법에 대한 기술한다.

본 논문의 구성은 2장에서 제안한 자율적 웹 분산 시스템의 구성방법 및 구현에 대하여 설명하고, 3장에서 제안된 시스템의 구현과 성능평가 및 고찰을, 4장에서 결론을 맺는다.

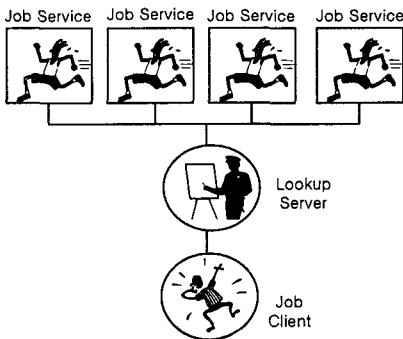
2. 자율적 웹 분산 컴퓨팅 시스템

기존의 웹 분산 구축환경은 웹브라우저를 이용하여 사용자가 직접 참여하는 방식이었다. 이러한 환경은 병렬프로그램을 작성할 때 수행환경, 참여하는 시스템의 수, 플랫폼의 사양등을 사전에 알 수가 없고 또한 웹에 연결된 수많은 자원들을 효율적으로 이용할 수 없다는 단점을 지니고 있다. 제안한 시스템은 지니기술(Jini Technology)[7][8]을 이용하여 동적 컴퓨팅 자원을 최적으로 활용할 수 있도록 지원함과 동시에 병

렬처리 작업을 위한 스케줄링, 시스템 오류에 대한 동적처리를 지원한다. 본 절에서는 제안한 시스템에 대한 시스템구성 및 동작방식을 설명한다.

2.1 시스템의 구성

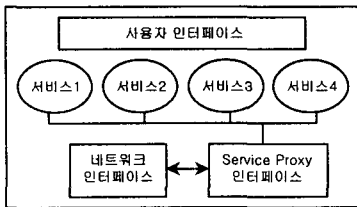
제안한 시스템은 역할에 따라 모두 세 가지의 어플리케이션으로 구분된다. 먼저 분산처리문제에 대한 해결방법을 제공하여주는 Job Service와 분산작업을 요구하는 Job Client, 그리고 Job Service와 Job Client 사이의 작업을 연계해주는 Lookup Server로 구성된다.



[그림 1] 시스템의 구조

2.1.1 Job Service

Job Service는 Job Client에 의해 요구되어진 병렬처리 문제를 해결할 수 있도록 자신의 자원을 제공하는 시스템이다.



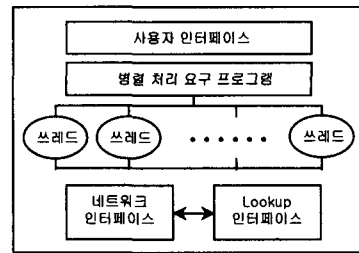
[그림 2] Job Service

Job Service는 사용자 인터페이스, 서비스, Service Proxy 인터페이스, 네트워크 인터페이스로 구성이 된다. Service Proxy 인터페이스는 Lookup Server에 Job Service가 제공할 수 있는 서비스를 등록하는 부분으로 Job Client가 요구하는 서비스를 제공할 수 있는 사용자 프로그램을 등록시키는 일을 담당한다.

Service부분은 Job Client가 작업을 요청하였을 때 실행되는 병렬프로그램 코드블록이다. 사용자 인터페이스를 통한 서비스 허용요청이 있을 때 Service Proxy 인터페이스를 통하여 Lookup Server에 등록된다. Service Proxy등록을 위하여 먼저 Job Service는 Multicast 방식으로 Discovery작업을 수행한 후 Lookup Server의 위치를 파악하여 Proxy를 등록한다.

2.1.2 Job Client

Job Client는 Job Service에게 병렬작업을 요청하였는 주체로 병렬프로그램이 작동할 수 있도록 [그림 3]과 같은 구조를 갖는다.



[그림 3] Job Client

Job Client는 병렬처리 요구 프로그램, 네트워크 인터페이스, Lookup 인터페이스로 구성되어 있다. 병렬처리 요구 프로그램은 사용자가 Job Service에게 작업을 분담요청할 병렬프로그램이며 수행 시 쓰레드를 발생시켜 각 Job Service에게 분산 작업을 요구한다.

Lookup 인터페이스는 병렬작업을 위하여 수행할 서비스의 목록을 수집한다. 만일 선정된 작업이 수행되었을 때 Lookup 인터페이스는 Unicast 방식과 Multicast방식을 이용하여 Lookup Server의 위치를 파악하여 작업수행이 가능한 Job Server의 목록을 수집한다.

2.1.3 Lookup Server

Lookup service는 네트워크에 연결되어 있는 독립된 호스트에 존재하며 Job Client에게 Job Service를 중재하는 역할을 수행한다. 그리고 Job Service들은 Lookup service에 자신의 서비스 객체와 자신을 설명하는 속성들을 등록한다. 또한 Job 클라이언트는 Lookup service에서 원하는 서비스 프록시 객체와 속성을 찾고 서비스 프록시 객체를 다운로드를 하여 사

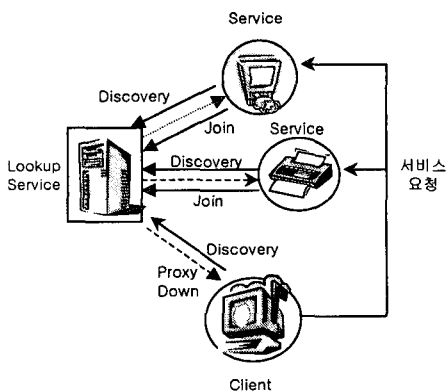
용한다.

2.2 시스템의 동작방식

제안한 자율적 웹 분산 시스템을 구성하기 위한 지니기술(Jini Technology)[7][8]과 함께 공유메모리 방식을 이용한다.

2.2.1 지니(Jini)

지니는 Sun에서 제안한 제안한 네트워크 PnP 기술로 네트워크에 접속된 기기나 소프트웨어들을 동적으로 상호작용을 가능케하는 런타임 인프라스트럭처 기술이다. 지니는 이용자(Client)와 제공자(Service), 관리자(Lookup)로 구성되어지며 각각의 구성요소들은 Discovery, Join, 그리고 Lookup 프로토콜을 이용하여 상호 통신하며 이용자와 제공자는 기본적으로 자바 RMI(Remote Method Invocation)를 이용하여 상호작용한다 지니 시스템의 수행과정은 먼저 서비스 제공자는 Discovery 프로토콜을 이용하여 서비스 관리자를 찾고, Join 프로토콜을 이용하여 서비스 관리자에게 서비스의 서비스와 상호작용을 하는 직렬화된 자바 객체인 프록시 객체(proxy object)를 등록한다. 서비스 이용자는 Discovery 프로토콜을 이용하여 서비스 관리자를 찾고 Lookup 프로토콜을 이용하여 자신이 원하는 서비스를 검색하여 프록시 객체를 다운로드 받아 서비스와 상호작용한다.



[그림 4] Jini의 수행구조

2.2.2 공유메모리

공유메모리는 병렬 작업을 수행중인 Job Service들이 접근할 수 있는 가상 어드레스 공간으로 병렬 프로그램이 수행될 때 각 Job Service들에게 공유메모

리의 일정부분을 할당하여 상호작업 및 데이터전달이 가능토록 구성되어 있다. 각 Job Service들은 Job Client의 공유메모리에 접근하여 필요한 데이터를 얻어 연산을 수행한 후 결과값을 기록한다.

3. 구현 및 고찰

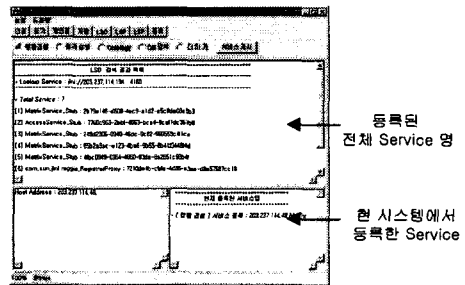
제안된 시스템을 구현하기 위하여 Java1.2.2와 Jini1.0.1을 이용하였고 Job Client 1대, Lookup Server 1대, Job Service 7대를 이용하여 실험하였다. 실험에 사용된 System의 사양은 다음과 같다.

[표 1] 실험에 사용된 시스템 사양

Lookup	Job Client	Job Service	Job Service	Job Service	Job Service	Job Service	Job Service	Job Service
PIII500	pII200	p166	p233	pIII600	pII233	pII433	p166	pII433
64MB	32MB	16MB	32MB	128MB	32MB	64MB	16MB	64MB
Win98	Win98	Win98	Linux	WinNT	Win98	Win98	Win98	Linux

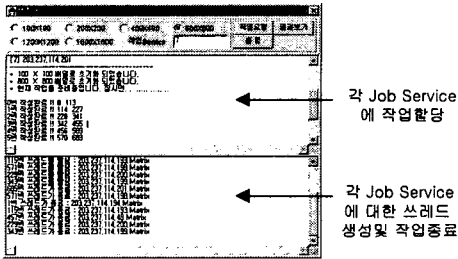
Lookup Server에는 코드의 다운로드를 위해 웹서버(web Server) Rmid, Lookup Service가 구동되어 비스들을 주선하는 역할을 한다

각 Job Service에는 Service Application이 실행되어지고 이들은 Multicast 방식에 의하여 서비스 내용을 Lookup에 등록(Join)한다. 아래의 [그림5]는 Job Service의 실행화면이다.



[그림 5] 구동된 Job Service

Lookup에 등록되어진 서비스는 총 7가지로 이들은 분산작업을 위하여 각각의 주어진 서비스명을 통하여 호출되어진다. 위 [그림 5]의 Job Service 시스템은 "203.237.114.48.Matrix"라는 서비스명을 갖는다. Job Client는 이들 각각 등록된 서비스목록을 참조하여 분산작업을 할당하고 해당 Job Service들에게 작업을 요청한다. 아래의 화면은 작업을 요청하여 실행되어진 내용을 보여준다.



[그림 6] Job Client의 작업실행

실험은 800×800 행렬에 대한 곱셈연산시간을 측정하였다. Job Client는 작업에 참여하는 Service에 비례하여 연산작업을 분배하였고, 분산 작업에 참여하는 Job Service의 수를 변경해가며 측정하였다. 아래의 [표 2]는 측정된 전체작업시간과 연산에 소요된 연산시간을 나타낸다.

[표 2] 전체작업시간과 연산시간

Job Service 수	1대	2대	4대	7대
전체작업시간	873ms	763ms	735ms	650ms
연산시간	632ms	437ms	273ms	104ms

위의 [표 2]에서 실제 연산시간은 작업에 참여한 Job Service의 수에 비례하여 6배이상 연산시간이 감소하였지만 작업을 위해 준비하는 시간을 포함한 전체작업시간에서는 약 20%이상의 효과만을 얻을 수 있었다. 이는 실제 연산시간에 비하여 참여 Job Service에 대한 Discovery, Lookup과정에서 많은 시간이 소요됨을 알 수 있다.

4. 결론

본 논문에서는 웹에 연결된 각종 유휴자원을 이용하여 자율적 웹 분산 환경을 구축하기 위한 방안을 연구하였다. 제안한 분산환경 구축방법은 지니 기술을 이용함으로써 자율적 웹 분산환경구축이 가능하였고 실험을 통하여 적은 비용으로 높은 효과를 볼 수 있는 가능성을 제시하였다. 이후의 연구과제로 전체연산시간에 대한 감축을 위하여 Job Service 준비시간을 단축시킬 수 있는 방안과 함께 다양한 웹 유휴자원활용에 대한 연구가 수행되어야 하리라 사료된다.

참고문헌

- [1] B. Christiansen, P. Cappello, M. Ionescu, "Javelin : Internet-Based Parallel Computing Using Java" 1997 ACM Workshop on Java for Science and Engineering Computation, pp30-40, June 1997.
- [2] T. Brecht, H. Sandhu, M Shan, and J. Talbot "ParaWeb : Towards World-Wild Supercomputing" 7th ACM SIGOPS European Workshop, pp181-188 Sep. 1996.
- [3] A. Baratloo, M, Karaul, Z. Kedem, and P. Wychkoff, "Charlotte:Metacoputing on the Web", 9th International Conference on Parallel and Distributed Computing Systems, pp151-159, Sep, 1996,
- [4] A. S. Grimshaw, Wm. A. Wulf, and the Legion team, "The Legion Vision of a WorldWild Virtual Computer." Communications of the ACM, 40(1), January 1997,
- [5] K. M. Chandy, B. Dimitrov and H. Le, "A World-Wild Distributed System Using Java and the Internet" InProceedings of the Fifth IEEE International Symposium on High Performance Distributed Computing, Syracuse, NY Aug. 1996
- [6] David Flanagan, "Java in a Nutshell, A Desktop Quick Referecne for Java Programmers", O'Reilly & Associates, Inc, 1996.
- [7] S. Oaks, H. Wong, "JINI in a Nutshell", O'Reilly& Associated, Inc March, 2000.
- [8] W. Keith Edwards, "Core JINI" Prentice Hall PTR, 1999.