

분산 컴퓨팅 환경에서의 동적 자원 관리를 위한 적응형 이동 컴퓨팅 모델

정은주^o 정동원 백두권
고려대학교 컴퓨터학과
{violetto^o, withimp, baik}@software.korea.ac.kr

AMC: An Adaptive Mobile Computing Model for Dynamic Resource Management in Distributed Computing Environments

Eunju Jeong^o Dongwon Jeong Dookwon Baik
Software System Laboratory, Dept. of Computer Science, Korea University

요 약

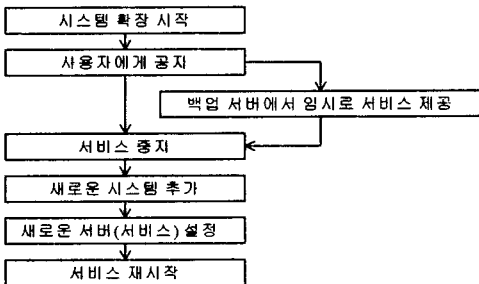
시스템의 구조는 확장 또는 부하 균형을 위해 재구성 및 재구조화가 필요하다. 이러한 상황에서, 다수의 새로운 시스템이 기존의 시스템에 추가되며 이전의 시스템이 담당하고 있던 역할의 일부분이 새로 추가된 시스템으로 분산된다. 이러한 자원 관리는 몇 가지 문제를 야기시키는데, 가장 큰 문제점은 일부 시스템 또는 전체 시스템의 재구조화를 위해 일정기간 동안 서비스 제공이 중단된다는 점이다. 이 논문에서는 동적인 자원관리가 요구되는 변화하는 환경에서 자동적으로 새로운 시스템을 추가하고 적응적으로 시스템의 부하를 재조정할 수 있는 이동 컴퓨팅 모델을 제안한다.

1. 서론

이 논문에서는 서버 시스템을 사용자에게 특정 서비스를 제공하는 시스템 그룹으로 정의한다. 메일 서비스, FTP 서비스, 웹 서비스, 지도 서비스 등의 응용 서비스들은 하나 또는 여러 개의 시스템에 분산되어 존재할 수 있다.

특수한 경우, 하나의 문제를 여러 개의 작은 문제로 나누고, 각 문제를 담당하는 서버 시스템으로 분산시킬 수 있는 서버 시스템을 구축할 수 있다. 서버는 하위 서버 시스템으로 분산한 문제의 결과들을 모아서 그들로부터 최종 결과를 생성한다. 때때로 하위 서버 시스템은 그들 간의 상호협력에 의해 협업이나 통신을 한다. 위에서 언급한 모든 경우들은 분산 컴퓨팅 환경하에서 이루어지며 각각의 경우에 대해서는 향후에 보다 상세하게 서술할 것이다.

이 논문에서 제안하는 시스템 모델의 목적과 필요성을 명확하게 기술하기 위하여 실질적인 시스템 관리의 한 예를 가정해 보자. 예를 들어, 관리자가 성능 향상과 부하 균형을 위해 새로운 시스템을 추가하고자 할 때, 기존의 시스템 확장 처리과정은 다음과 같다.



<그림1> 일반적인 확장 처리과정

그림 1에서는 확장시 가능한 두 가지 방법을 표현하고 있다. 첫 번째는 서버 확장을 위해서 서비스를 중지하고 새로운 시스템을 추가하는 경우이다. 두 번째는 서버의 내용을 임시로 저장한 백업 서버에서 서비스를 제공하게 한 후, 기존의 서버를 중지시키고 새로운 시스템을 추가하는 경우이다. 이들이 지니는 문제점들을 정리하면 다음과 같다.

- 사용자에게 일정기간 동안 서비스를 제공할 수 없으므로 서비스의 신뢰성 저하
- 백업 서버 시스템의 관리를 위한 추가 비용 소모
- 시스템 자원의 낭비
 - : 예를 들면, 몇몇 서버 시스템은 사용자의 감소로 인해 사용하지 않을 수 있기 때문에 필요치 않은 관리 비용이 요구된다.
- 동적 또는 실시간 시스템의 최적화 및 재구조화 불가능

앞에서 서술한 문제점을 해결하기 위해 적응형 이동 컴퓨팅(Adaptive Mobile Computing, AMC) 모델을 제안한다. 이 모델은 서버 시스템 자원의 최적화 및 재구조화를 위한 프레임워크를 제공한다.

2. 이동 에이전트 패러다임과 분산 컴퓨팅 타입

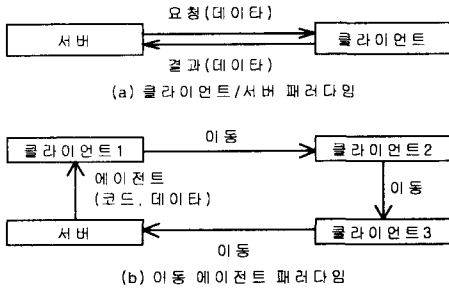
2.1 이동 에이전트 패러다임

이동 에이전트 패러다임은 전통적인 클라이언트/서버 패러다임으로부터 진화되었다. 더 일반적이고 유사한 개념은 이동 객체이며 연산 집합과 데이터를 함께 캡슐화하여 원격 노드에서 다른 노드로 전송할 수 있다[1][2]. 전통적으로 분산 시스템 서비스는 클라이언트/서버 패러다임을 사용해서 구성되는데, 이 패러다임은 서버에서 모든 연산처리가 이루어지므로 대부분의 부하가 서버에 집중되는 단점을 가진다.

이동 에이전트 패러다임은 전통적인 패러다임(클라이언트/서버 패러다임)의 문제를 해결할 수 있는 많은 장

점을 지닌다. [3]에서는 이동 에이전트의 장점에 대하여 정의하고 있다.

그림 3은 이동 에이전트 패러다임과 클라이언트/서버 패러다임의 특징을 도식화해서 보여준다[5].

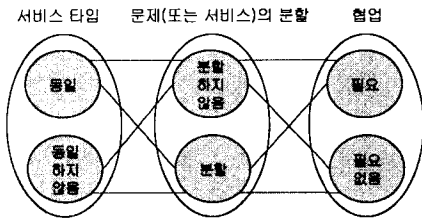


<그림 2> 클라이언트/서버 패러다임과 이동 에이전트 패러다임

이동 에이전트 패러다임의 특징은 이 논문에서 제안한 모델의 핵심으로서, 프로그램의 이동성을 이용하여 서버 시스템의 자동적인 재구조화 및 확장을 가능하게 한다.

2.2 분산 컴퓨팅 타입

분산 컴퓨팅 타입을 정의하기 위하여 서비스 타입, 분할 및 협업 요인을 이용하였으며, 그림 3은 이들 간의 관계를 표현하고 있다.



<그림 3> 분산 컴퓨팅 요인

표 1은 그림 3의 분산 컴퓨팅 요인들의 의미를 표로 정리하여 나타낸 것이다.

<표 1> 분산 컴퓨팅 요인의 의미

구분	의미	
서비스 타입	동일하지 않음	하위 서버 시스템이 다른 작업을 처리
	동일	하위 서버 시스템이 동일한 작업을 처리
문제(또는 서비스)의 분할	분할하지 않음	하나의 작업(또는 문제)을 여러 개의 하위 작업으로 나누지 않음
	분할	하나의 문제를 여러 개의 문제로 나누어 처리한 후, 최상위 서버 시스템에서 모든 결과를 모아서 최종 결과를 만듦
협업	필요없음	하위 서버들의 상호협력 및 통신 필요없음
	필요	하위 서버는 상호협력 및 통신을 하면서 주어진 문제의 해결책을 찾음

표 2는 그림 3에 도식화 한 세 가지 요인(서비스 타입, 분할 및 협업)들 간의 관계를 표로 정리하여 나타낸 것으로, 각 타입의 특징은 다음과 같이 정의할 수 있다.

<표 2> 분산 컴퓨팅 분류

타입	서비스 타입	서비스(또는 작업)의 분할	협업
1	동일	분할하지 않음	필요없음
2	동일	분할	필요없음
3	동일	분할하지 않음	필요
4	동일	분할	필요
5	동일하지 않음	분할하지 않음	필요없음
6	동일하지 않음	분할	필요없음
7	동일하지 않음	분할하지 않음	필요
8	동일하지 않음	분할	필요

- 타입 1 : 단일 서비스 제공
- 타입 2 : 단일 서비스 제공. 태스크와 문제 분할
- 타입 3 : 단일 서비스 제공. 다른 원격 시스템에 위치한 분산 서비스 간의 협업을 위해 필요
- 타입 4 : 단일 서비스 제공. 태스크(또는 문제)를 나누고, 분산된 하위 태스크(또는 하위 문제, 서비스) 간의 협업을 위해 필요
- 타입 5 : 다양한 서비스 제공
- 타입 6 : 다양한 서비스 제공. 태스크와 문제 분할
- 타입 7 : 다양한 서비스 제공. 다른 원격 시스템에 위치한 분산 서비스 간의 협업을 위해 필요
- 타입 8 : 다양한 서비스 제공. 태스크(또는 문제)를 나누고, 분산된 하위 태스크(또는 하위 문제, 서비스) 간의 협업을 위해 필요

위에서 언급한 내용을 기반으로 이 논문의 목적인 시스템의 자동적인 재구조화 및 관리를 위한 시스템 모델을 제공하기 위해서 앞에서 기술한 이동 에이전트 패러다임을 이용한다.

3. 적응형 이동 컴퓨팅(AMC) 모델

3.1 AMC 모델의 구조

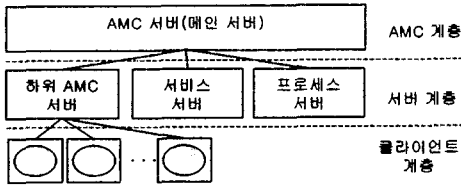
AMC(Adaptive Mobile Computing, 적응형 이동 컴퓨팅) 모델은 AMC 계층, 서버 계층, 클라이언트 계층의 3계층 구조로 이루어진다. AMC 계층은 메인 서버 시스템으로 전체 시스템과 그들의 서비스를 제어하는 가장 중요한 역할을 수행한다. 그 역할은 다음과 같다.

- 내부와 외부의 요인들을 모니터링해서 진화를 할 것인지 아닌지 결정
- 진화할 때 시스템을 재구조화. 이 때, 에이전트를 생성하고 그들을 하위 시스템으로 이동시킴. 이러한 에이전트는 서버 프로그램과 간단한 프로그램으로 구성
- 하위 시스템간의 통신을 위한 협업 공간(통신 공간) 제공

AMC 계층의 메인 서버 시스템은 모든 요청을 처리할 수 없을 때 몇 개의 다른 시스템에 태스크(서비스, 처리 과정)를 분산한다. 하위 서버 시스템들로 구성된 두 번째 계층이 서버 계층이다. 때때로 서버 계층의 에이전트는 AMC 서버에 의해 주어진 하위 문제를 해결할 수 있는 서버 또는 간단한 프로그램이라 할 수 있다. 마지막 계층인 클라이언트 계층은 서버 측의 문제 해결 허용범

위 초과시 형성된다. 이 상황에서는 필요한 프로그램과 데이터를 일시적으로 클라이언트 시스템 안에 이동시켜서 요청한 문제를 해결한다. 모든 요청은 클라이언트 시스템의 자원을 사용해서 처리함으로써 부하 분산 효과를 얻을 수 있다.

그림 4는 AMC 모델의 3 계층 구조를 보여준다.



<그림 4> AMC 모델의 3계층

표 3은 AMC 모델의 특성을 정리하여 표로 나타낸 것이다.

<표 3> AMC 모델의 특성

구분	특성
지속성 (Continuance)	서비스가 지속적으로 사용자에게 제공
지능성 (Intelligence)	AMC 계층의 서버 시스템은 다양하고 변화하는 환경을 감시하고 인식해서 지능적으로 다른 시스템에게 태스크 분배
이동성 (Mobility)	연산과 데이터는 메인 서버 시스템으로부터 하위 시스템으로 이동하여 주어진 서비스를 제공하거나 특정 문제를 해결한다. 이는 이동 에이전트 패러다임에서 가장 중요한 특성[2]
동적 확장성 (Dynamic Extensibility)	서비스를 지속적으로 제공하면서 시스템의 확장 및 재구조화를 할 수 있음
최적화 (Optimization)	클라이언트 시스템을 하위 서버 시스템으로 사용함으로써 서버 시스템들의 부하를 분산시키고 전체 시스템 성능 분산[3]

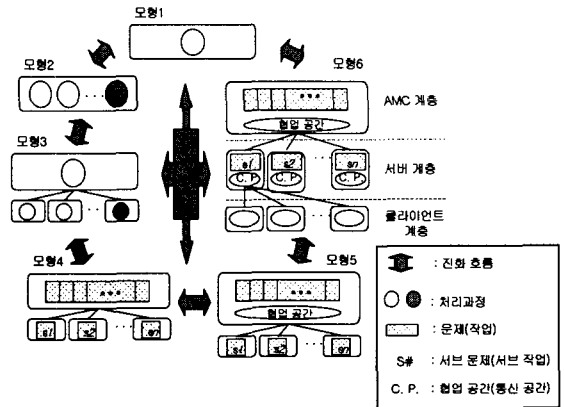
3.2 AMC 모델과 진화론

AMC 모델은 다윈의 진화론과 매우 유사하다. 다윈의 진화론은 "환경(자연)은 개체들 중에서 환경에 적합하고 우수한 개체를 선택하여 번식이 가능하게 하고, 열등한 개체들은 도태시킨다."라는 자연 선택이론을 바탕으로 한다[4]. 진화론은 생물체가 환경 변화에 자신의 구조를 자율적으로 진화시키며 적응한다는 것이다. 이 논문에서 제안하는 AMC 모델은 이러한 특성을 지니는 자율적인 시스템 구조를 제공한다. 즉, 진화이론처럼 환경에 따라 시스템의 구조를 동적으로 변화시킴으로써 변화된 환경에 최적화된 형태로 발전해 나간다.

3.3 AMC 모델 기반 시스템의 진화 모형

AMC 모델의 기본적인 개념은 다윈이 만든 진화론을 기반으로 하고 있으며, 시스템의 자율적이고 자동화된 구조화를 위해 이동 에이전트 패러다임을 사용한다. AMC 모델기반 시스템은 내부와 외부의 요인 즉, 환경에 따라 점차적으로 진화한다. 그림 5에서는 AMC 모델기반 시스템의 가능한 진화 모형과 각각의 구조의 예를 보여준다.

그림 5에서, 임의의 시스템은 시스템 구조가 확장 또는 축소될 수 있다. 예를 들어, 사용자의 수요가 많아 시스템이 모형2에서 모형6으로 확장될 수도 있고, 서버의 성능 향상이나 사용자의 감소로 모형6에서 모형2로 축소될 수도 있다. 이는 진화론의 자연선택이론처럼 필요한 개체는 번식시키고, 쓸모없는 개체는 도태시키는 것과 같은 개념이다.



<그림 5> AMC 모델 기반 시스템의 일반적인 진화 단계

4. 결론 및 향후 연구 방향

현재, 서버 자원 관리는 자동적 또는 동적으로 처리되지 않고 수동적으로 확장 및 재구조화를 해야 하기 때문에 많은 문제점을 지닌다. 이러한 문제점을 해결하기 위해서 내부와 외부의 환경에 따라 시스템이 진화할 수 있는 AMC 모델을 제안하였다.

AMC 모델은 분산 컴퓨팅, 이동 에이전트 패러다임, 다윈의 진화론 등을 기반으로 한다. 진화론 및 이동 에이전트 패러다임의 적용은 AMC 모델 기반 시스템에 자율성, 이동성, 확장성, 적응성, 지속성, 최적화 등의 많은 장점을 제공한다. 즉, 변화하는 환경에 따라 시스템 구조가 자동적·자율적으로 진화할 수 있도록 한다.

향후 연구과제로는 보다 다양하고 세부화된 요인 정의 및 연구가 요구되며, 시스템 구조의 자율적인 진화를 위한 진화 알고리즘에 대한 연구가 요구된다.

참고문헌

- [1] E. Jul, H. Levy, N. Hutchinson, and A. Black, "Fine-Grained Mobility in the Emerald System", ACM ems. Transactions on Computer Systems, Vol. 6, No. 1, 109-133, February 1988
- [2] R. Gray, D. Kotz, G. Cybenko and D. Rus, "Mobile agents: Motivations and state-of-the-art systems", Dartmouth college, TR2000-365, 19, April 2000
- [3] D. B. Lange and M. Oshima, "Seven Good Reasons for Mobile Agents", Vol. 42, No. 3, Communications of the ACM, March 1999
- [4] 다윈/이민재, "종의 기원", 을유문화사, 2001
- [5] R. S. S. Filho, "The Mobile Agents Paradigm", ICS221-Software Engineering Final Paper (Winter 2001)