

광역 분산 처리 환경에서의 모바일 에이전트 시스템 분석

방정원*

The Analysis of Mobile Agent System for the Wide Distributed Processing

Bang, Jung Won *

요 약

인터넷 기술과 웹 기반 응용 프로그램의 발전으로 광역 분산처리에 대한 요구가 기하급수적으로 증대되고 있다. 이러한 광역 분산처리 환경의 요구사항에 적용할 수 있는 모델로 모바일 에이전트를 들 수 있다. 그러나 현재의 모바일 에이전트 모델은 여러 가지 기술적 제약점들을 가지고 있다. 모바일 에이전트가 광역 분산 처리 환경에서 사용되기 위하여 필요한, 모바일 에이전트 시스템의 구조와 사용되는 프로그래밍 패러다임과 사용 기술 등에 있어서의 문제점들을 분석하였다.

Abstract

Due to the advances of the Internet technologies and web-based applications, the demand for wide distributed processing is increased. Mobile Agents have been proposed as a model to meet with the requirement for such environment. But current mobile agent system has the technical restriction. The requirement of mobile agent for the wide distributed processing is analysed in mobile agent system architecture, programming paradigm and technologies.

▶ Keyword : 모바일 에이전트, 에이전트 시스템, 에이전트 프로그래밍

1. 서론

인터넷 기술과 웹 기반 응용 프로그램의 팽목할만한 발전으로 인해, 현재는 광역 분산처리에 대한 요구가 기하급수적으로 증대되고 있다. 이러한 광역 분산처리 환경의 요구사항에 적용할 수 있는 모델로 모바일 에이전트를 들 수 있다[1]. 앞으로 거의 모든 인터넷 사이트들은 모바일 에이전트 형태를 제공하게 될 것으로 보인다. 그러나 현재의 모바일 에이전트 모델은 여러 가지 기술적 제약점들을 가지고 있다.

모바일 에이전트는 컴퓨터 네트워크에서 사용자를 나타내는 프로그램이며, 스스로 네트워크내의 한 노드로부터 다른 노드로 이동할 수 있고, 사용자의 요구에 따라 간단한 작업을 수행할 수 있다[2].

모바일 에이전트는 이를 지원하는 에이전트 서버를 필요로 한다. 에이전트 서버는 모바일 에이전트에 리소스와 서비스를 제공하며, 모바일 에이전트를 수행할 수 있는 환경을 제공하고, 프로그래머에게는 에이전트를 프로그래밍 할 수 있는 기본적인 명령들을 제공한다.

에이전트는 다른 에이전트나 에이전트 서버와 상호 작용하도록 디자인 되었다. 예를 들어, 에이전트는 에이전트 서버의 리소스를 사용하거나, 에이전트 서버에 특별한 서비스를 요청하거나, 다른 에이전트들과 함께 업무를 수행하기 위해서는 에이전트 서버의 허가를 받아야 한다. 적용업무 프로그램들이 필요로 하는 기능이 여러 가지로 다양하므로, 에이전트는 적용업무 도메인에 따라 다른 업무들을 수행하는 여러 가지 형태로 만들어진다. 적용업무의 도메인에 따라 다양한 요구사항들은 적용업무에 따라 특수한 서비스나 리소스를 제공하는 적용업무 별 특별 에이전트 서버를 만들어 냈다.

적용업무에 따른 다양성이나 여러 가지 사용자 요구사항으로 인해 다양한 특징을 갖는 에이전트 시스템이 존재하지만 크게 보면 다음과 같은 두 가지로 분류할 수 있다. 자바 베이스 시스템과 비 자바 베이스 시스템. 잘 알려져 있고 쉽게 접근할 수 있는 검증된 메카니즘이기 때문에 자바가 에이전트 구성도구로 널리 채택되었다. 그러나 불행하게도 자바는 표준으로 사용되기에는 치명적인 몇 개의 단점을 가

지고 있다. 가장 중요한 문제는 에이전트 쓰레드의 실행 상태를 파악할 수 없다는 점이다. 이러한 문제점은 자바 가상 머신에 일반적인 자바 가상 머신과는 호환 되지 않는 변경을 필요로 하게 되었다. 에이전트 시스템에 필요로 하는 기본 기능들을 지원하지 못한다면 표준으로 쓰일 수 없기 때문이다. 효율적인 모바일 에이전트 응용프로그램의 개발을 위해서는 적당한 모델과 기반구조를 필요로 한다. 이 논문에서는 모바일 에이전트가 광역 분산 처리 환경에서 사용되기 위하여 필요한, 모바일 에이전트 시스템의 구조와 사용되는 프로그래밍 패러다임과 사용 기술 등에 있어서의 문제점들을 논의하고자 한다.

II. 모바일 에이전트 시스템 구조

에이전트 시스템의 구조는 에이전트 서버, 통신 프로토콜, 프로그래밍 명령, 에이전트에 자원과 에이전트 구조를 제공하는 메카니즘으로 이루어져 있다. 이것이 에이전트 시스템의 기반구조이다. 각 구성원들을 만들고 이들을 모바일 에이전트 구조를 구성하기 위해 통합시키는 여러 가지 방법들이 제안되었다. 대부분의 방법들은 설계자가 자신만의 에이전트 시스템을 구성할 수 있도록 블록과 같이 작용하도록 디자인 되었다. 이 장에서는 제안된 방법들의 주된 개념과 중요한 아이디어들에 대하여 살펴본다.

1. 모빌리티

모빌리티는 시스템의 각 항목들이 수행되는 위치가 바뀔을 말한다. 여기서 각 항목들이라 함은 데이터, 실행 제어문, 프로그램 코드, 수행 환경 등을 말한다. 전형적으로 Remote Evaluation, Code On Demand, 모바일 에이전트의 3가지 모빌리티 패러다임이 있다[3]. 이것은 이동되는 항목들의 종류에 따라 분류된다. 모든 패러다임이 프로그램 코드의 일부가 한 노드에서 다른 노드로 보내져 그곳에서 수행되어지는 코드 이동을 포함하기 때문에 코드 모빌리티라고도 부른다. 코드를 보낸 노드가 수행을 시작시키면 Remote Evaluation이라 하며, 받은 노드가 수행을 시작시키면 Code On Demand라 부른다. 모바일 에이전트는 다시 강약의 2가지 패러다임으로 나누인다.

패러다임은 코드뿐만 아니라 모바일 에이전트 변수들의 값들로 구성된 데이터 상태, 스택과 프로그램 카운터로 구성된 수행상태까지도 고려한다(4). 모바일 코드 패러다임은 코드, 자원, 프로세서가 어떻게 분포되어 있는가에 따라 설명된다. 프로세서는 추상적인 머신으로 코드를 수행하고 수행 상태를 보존하는 역할을 한다. 모바일 에이전트 패러다임이 코드와 데이터상태만을 이동시키면 약 모빌리티, 수행상태까지도 이동시키면 강 모빌리티로 분류한다. 자바는 연속적인 메카니즘으로 인해, 내부 쓰레드 구조를 추출하는 것이 어렵다. 따라서 Aglets, Odyssey, Java2go 등의 많은 자바 기반 시스템들이 약 모바일 에이전트 시스템이다.

그렇다면 항상 강 모빌리티가 필요하고, 강 모빌리티가 약 모빌리티보다 우월한가? 보고된 바에 의하면 대부분 모바일 에이전트 적용업무들의 요구사항들이 약 모빌리티만으로도 만족 될 수 있다고 한다(5). 강 모빌리티는 단지 특별한 적용업무 분야에서만 필요로 하는데 이런 경우는 반드시 정지된 곳에서 다시 시작해야하는 로드 발란싱과 같은 분야이다. 강 모빌리티가 좀 더 완벽하기는 하지만 약 모빌리티가 더 구현하기 쉽고, 위험요소가 적으며, 대부분의 에이전트 적용업무에 사용 가능하다. 따라서 모바일 에이전트 시스템을 디자인할 때, 적용업무의 특성을 잘 살펴보고 다른 컴포넌트들을 디자인하기 전에 모빌리티의 범위를 결정해야 한다.

2. 코디네이션

에이전트의 끊임없이 이동하는 성향으로 인해, 에이전트나 에이전트 서버 같은 서로 다른 구성원들과 간의 행동에 관한 관리가 필요하다. 코디네이션은 이러한 두 그룹간의 구성원들 사이에서 일어난다. 여러 개의 모바일 에이전트가 협력하여 업무를 수행하는 적용 업무는 에이전트를 코디네이팅 함으로 한 장소를 여러 번 방문하지 못하게 하는 등의 메카니즘이 필요하다. [6]번의 자료에 따르면, 공간적 시간적 성격에 따라 코디네이션 모델을 4개의 카테고리로 구분한다. 만일 통신을 시작한 쪽이 상대방의 이름을 알아야 한다면, 공간적으로 커플링 되었다고 분류한다. 통신이 동기화되어 있다면, 임시적으로 커플링 되었다고 분류한다. 즉, 이것은 수신에이전트와 송신 에이전트가 동시에 통신상에 있어야 함을 의미한다. 경매 시스템의 제한 시간 아래서 에이전트 간에 계약을 하거나 경합을 하는 것은 이러한 좋은 예가 된다.

참여하는 에이전트의 이름을 인식하고, 에이전트 간에 기간을 동기화하는 것은 적용업무에 따라 다르다. 따라서 에

이전트 시스템이 코디네이션 모델을 구축하는 확정된 시나리오를 갖는 것이 때때로 문제 시 되기도 한다. 왜냐하면 그것은 시스템을 일반화를 떨어뜨리는 것이 되기 때문이다. 더 효율적인 에이전트 시스템의 표준화 문제를 다루기를 원한다면, 새로이 디자인된 시스템은 다른 모델들과 융합하는 한 가능하면 일반적이어야 한다.

III. 에이전트 프로그래밍 시스템

모바일 에이전트 시스템이 보편화 되고 표준화되면, 혼합적인 분산 시스템 환경에서 프로그래밍하고 사용하기 쉽게 될 것이며, 존재하는 가능한 많은 에이전트들과 함께 일할 수 있을 것이라는 것은 분명하다. 이러한 요구사항을 만족시키기 위해서는 현재 사용되는 언어 중에 가장 적합한 프로그래밍 언어가 선택되거나, 에이전트를 위한 새로운 언어가 디자인 되어야 한다. 이에 필요한 기존의 프로그래밍 언어의 고려사항들과 프로그래밍 명령들을 살펴보기로 하자.

1. 기존 프로그래밍 언어의 고려사항

몇몇의 에이전트 시스템은 에이전트 코드를 위하여 Tcl, Python, Perl과 같은 스크립트 언어를 쓴다. 스크립트 언어들은 상대적으로 간단하고 에이전트를 프로그래밍하기가 용이하다. 그러나 이러한 언어들은 모듈화 하거나 캡슐화 하기가 쉽지 않고, 성능이 나쁘다(2). 현재는 대략 55%의 에이전트 시스템이 에이전트 프로그래밍언어로 자바 언어를 사용하고 있다고 보고 되어 있다. 자바가 가장 많이 사용되는 우수한 언어이지만 모바일 에이전트 시스템을 위해 자바를 사용하는 것은 몇 가지 단점을 지닌다.

첫째, 자바는 에이전트를 위해 디자인 된 언어가 아니라 는 것이다. 에이전트 언어는 에이전트를 제어하거나 모바일 리티나 에이전트 간에 커뮤니케이션에 있어 여러 가지 패러다임을 지원해야 한다. 자바와 같은 일반적 목적의 언어는 에이전트 프로그래밍 환경에 필수 불가결하지만, 그것에 의해 지원되지 않는 많은 사용되지 않는 프로그래밍 명령들이나 기능을 가지고 있기 때문에 여러 가지 예측 치 못한 오류들이 발생할 수 있다. 또한 자바 가상 머신이 요구되는 기능을 전혀 지원하지 않는다면 자바 가상 머신에 대한 수

정은 불가피하다. 더욱이 자바는 객체 지향 패러다임을 지원하기 때문에 이것은 어떤 에이전트 어플리케이션에서는 적합하지 않을 수도 있다.

둘째, 에이전트 시스템의 각 항목들은 에이전트의 범용 메소드의 이름을 정확히 알아야 하는데, 이로 인해 프로그램이 유연성이 없고 프로그래밍 작성이 어렵게 되었다. 에이전트 프로그래밍에서 에이전트 이름을 알고 특히 범용 에이전트 시스템의 메소드 이름을 유지하는 것은 쉽지 않다.

셋째, JDK의 버전이 에이전트 프로그래밍 시스템의 이식성과 호환성에 있어 중요한 역할을 한다. 어떤 시스템의 경우 JDK특정 버전 이전에서만 사용가능 하기 때문이다.

Knabe는 에이전트 프로그래밍 언어로서의 4가지 요구사항을 지적한다. 그것은 코드 매니플레이션(명명하기와 이식성)과 분산 환경에서의 혼합성의 지원, 성능, 보안이다(7).

기존의 프로그래밍 언어를 이러한 요구사항을 만족하도록 수정하는 것보다는 새로운 에이전트 프로그래밍 언어를 개발하는 것이 바람직하다.

2. 프로그래밍 명령

Karnik와 Tripathi는 에이전트 프로그래밍 명령들을 다음과 같이 분류한다(2).

- 기본 에이전트 관리 : 생성, 발송, 복제, 이식
- 에이전트와 에이전트간의 통신 및 동기화
- 에이전트 모니터 및 제어 : 상태 조회, 에이전트 호출과 소멸
- 장애처리 : 체크포인팅, 예외 처리, 감사 등
- 보안관련 : 암호화, 서명 등

이러한 명령들은 시스템 수준에서 구조적으로 이러한 명령들을 지원하도록 구현된 에이전트 서버 구조에 의존한다. 그래서 설계자는 구문 해석과 컴파일을 제외한 에이전트 프로그래밍 시스템의 기능적 특징들 대부분이 에이전트 서버 모듈과 직접적으로 연관되어 있고, 에이전트 서버 모듈에 포함됨을 알아야 한다. 에이전트 관리 명령들 다음으로, 에이전트 통신이 시스템에서 가장 중요하다. 따라서 이 두개의 기본 명령들이 에이전트 시스템의 최소 기능을 구성하고 있다. 우리는 이 둘을 공통 클래스 명령들이라 하고 나머지 세 개를 하이 클래스 명령이라 한다.

에이전트 시스템의 명령수가 증가함에 따라, 디자인과 구현에 있어 시스템의 복잡정도가 기하급수적으로 증가될 것이다. 이러한 진보된 기능들을 필요로 하지 않는 많은 어플리케이션 영역들의 경우, 하이 클래스 기본명령들은 에이전트 시스템에 첨가되지 않을 것이다. 따라서 설계자는 그들

의 사용 그룹에 따라 프로그래밍 기본 명령들을 레고와 같이 다룰 수 있는 블록으로 모듈화할 필요가 있다. 그러나 이것은 하이 클래스 기본 명령들을 방지하라는 것은 아니다.

IV. 분산 환경에 따른 요구사항

1. 이식성과 확장성

모바일 에이전트는 널리 쓰이기 때문에 모바일 에이전트 코드는 반드시 이식성이 있어야 한다. 높은 이식성은 에이전트가 여러 다른 운영체제를 가지고 있는 혼합적 환경에서 수행될 수 있음을 말한다. 그러므로 대부분의 에이전트 시스템은 에이전트 코드를 수행하는 이식 가능한 가상 머신을 제공하는 프로그래밍 언어에 기반을 두고 있다(8).

이상적으로는 이식성은 플랫폼에 제한 받지 않는 표준화된 모바일 코드를 사용함에 따라 얻어질 수 있다. 자바의 바이트 코드와 같은 중간단계의 코드를 적용하는 것도 한 가지 대안 방법이 될 수 있다. 코드 베이스 형식을 사용하는 것은 이식성을 증대 시킨다. 그러므로 대부분의 에이전트 시스템은 에이전트 코드를 수행하는 이식 가능한 가상 머신을 제공하는 프로그래밍 언어에 기반을 두고 있다(10).

모바일 에이전트와 관련된 단체는 이행되는 에이전트의 코드나 상태가 암호화된 형식뿐만 아니라, 수행 환경 즉 가상 머신까지도 표준화 하려는 시도를 하여야 한다(11).

2. 보안

에이전트 간에 또는 에이전트와 에이전트 서버간의 활동이 쉽게 일어난다. 이러한 활동 가운데 보안의 문제가 발생할 수 있다. 악의를 가진 에이전트는 의도적이든 우연이든 간에 다른 에이전트들을 공격하고 때로는 에이전트 상태 값들을 변경할 수도 있다. 보안과 관계된 필요성들은 다음과 같다(2).

- 에이전트의 프라이버시와 일치성
- 에이전트와 서버의 인증
- 인증절차와 접근의 제어
- 계속하고 차지하고 지불하는 메카니즘

설계자는 어떠한 공격으로부터도 엔티티를 보호할 수 있는 적절한 보호 메커니즘을 제공해야 한다. 우리는 또한 에이전트와 서버를 보호할 필요가 있다. 그러므로 보안특성은 통신보안과 서버 자원 보호, 에이전트 보호 등을 포함한다. 대부분의 에이전트 시스템은 서버 자원 보호 메커니즘을 가지고 있으나 다른 보안 기능들도 모두 갖추고 있지 않다는 사실은 눈여겨 볼만하다.

3. 성능

모바일 에이전트와 기존의 패러다임들 간의 성능 비교가 수행되었다[12]. 그 결과 모바일 에이전트가 항상 최상의 해결책 만은 아니라는 것을 보여준다. 그러나 통신 시스템이 아주 빠르거나, 네트워크가 사용가능 하지 않거나 모바일 성향으로 인해 사용자가 임시로 시스템에 연결이 되지 않을 때는 가장 잘 적용할 수 있다.

모바일 에이전트는 때때로 기존의 전통적인 방법들보다 태스크를 수행하는 데 오래 걸린다. 왜냐하면 네트워크 트래픽을 피하기 위한 시간 절약이, 느린 수행이나 이행 오버헤드보다 적기 때문이다. 자바의 Just-In-Time의 컴파일 덕에, 모바일 코드는 원래 컴파일 된 코드만큼 빠르게 수행될 수 있다. 또한 이행 오버헤드를 줄이려는 연구가 계속되고 있다.

현재 에이전트를 계획하는 것은 에이전트 시스템의 성능에 영향을 준다는 것이 알려져 있다. Moisumi는 어떻게 에이전트가 효과적으로 그들의 업무를 수행하기 위해 네트워크를 돌아다니며 시간을 어떻게 사용하는지를 연구하였다 [9]. 에이전트의 수와 실행시간은 모바일 에이전트 플래닝의 두 가지 중요한 성능요소이다. 에이전트는 계획된 노드들을 돌아다니는 동안 네트워크 대역폭을 소모하게 된다. 잘못 스케줄된 에이전트 여정은 더 높은 라우팅 경비로 인해 수행 시간을 더 길게 만든다. 한 태스크를 위해 만들어진 에이전트 숫자가 많을수록 전체 라우팅 비용은 증가한다. 이러한 오버헤드를 줄이고 좀 더 좋은 성능을 내기 위해 에이전트들은 보내지기 전에 적절히 스케줄 되어야 한다.

V. 결론

이 논문에서는 모바일 에이전트가 광역 분산 처리 환경에서 사용되기 위하여 필요한, 모바일 에이전트 시스템의 구조와 사용되는 프로그래밍 패러다임과 사용 기술 등에 있어서의 문제점들을 살펴보았다. 대부분의 에이전트 시스템은 적용업무의 특성에 따라 특성화 되었으며, 55% 정도의 에이전트 시스템이 자바 언어를 사용한다. 이러한 모바일 에이전트 시스템이 급증하는 인터넷 환경 하에서 효율적으로 사용되려면, 본문에서 제기한 문제점들을 충족하여야 하며, 이와 더불어 표준화되고 일반화된 시스템을 필요로 하므로 이에 대한 연구가 있어야 한다.

참고문헌

- [1] Proc. of the 2nd International Workshop on Mobile Agent, Stuttgart, Lecture Notes in Computer Science, No. 1477, Springer-Verlag, September 1988.
- [2] Karnik N.M. and Tripathi A., Design Issues in Mobile Agent Programming Systems, IEEE Concurrency 52-61, 1998
- [3] Picco G.P., Carzaniga A. and Vigna G., Designing distributed applications with mobile code paradigms. Proc. of 19th International Conference on Software Engineering, July 1997.
- [4] Ghezzi C. and Vigna G., Mobile Code Paradigm and Technologies: A Case Study, International Workshop on Mobile Agents, April 1997.

- [5] Cabri G., Leonardi L. and Zambonelli F., Weak and Strong Mobility in Mobile Agent Application, Proc. of the 2nd International Conference and Exhibition on the Practical Application of Java, April 2000.
- [6] Cabri G., Leonardi L. and Zambonelli F., How to Coordinate Internet Applications based on Mobile Agents, Proceeding of the 7th IEEE workshop on Enabling Technologies, June 1998.
- [7] Knabe F., An overview of Mobile Agent Programming, Proceeding of the 5th LOMAPS Workshop on Analysis and Verification of Multiple-Agent Languages, 1996.
- [8] Silvia L. M., Soares G. and Martins P., The Performance of Mobile Agent Platforms, 1st International Symp. on Agent Systems and Applications, October, 1999
- [9] Moissumi K., Mobile Agent Planning Problems. Phh thesis, Dartmouth College, 1998
- [10] Thorn T., Programming Language for Mobile Code, ACM Computing Surveys, Polytechnic of Milano, 1999
- [11] Kotz D. and Gray R. S., Mobile Agents and the Future of the Internet, ACM Operating Systems review, 33(3):7-13, August, 1999.
- [12] Puliafito A., Riccobene S. and Scarpa M., An Analytical Comparison of Client-Server, Remote Evaluation and Mobile Agents Paradigms, 1st International Symp. on Agent Systems and Applications, October, 1999

저 자 소 개

방 정 원

1984년 서울대학교 계산통계학과
 1984~1995 한국 아이비엠(주)
 1997년 한국과학기술원 정보 및
 통신공학과
 1997~현재 청강문화 산업대학
 컴퓨터소프트웨어과 교수