

변형 P2P 기반 시스템을 활용한 이동 에이전트 시스템에 관한 연구

(A Study of an Mobile Agent System Based on Hybrid P2P)

이 석 희*, 양 일 등**, 김 성 열***

(Seok-Hee Lee, Il-Deung Yang, and Seong-Ryeol Kim)

요 약 최근 협업을 위한 그리드와 클라우드 컴퓨팅은 사회적 이슈가 되고 있다. 이러한 협업 네트워크 시스템의 기본은 P2P 시스템이다. P2P 시스템은 기존 클라이언트/서버 기술과는 달리 서로의 정보를 교환하기 위해 형태적인 분류에 따라 그 목적과 기능이 나뉜다. 이러한 목적과 기능에 따라 사용자는 정보와 자료의 검색, 원격 프로그램 제어 기능 등을 위한 통합 서비스를 제공한다. 대부분의 P2P 시스템은 클라이언트/서버 형태를 취하고 있지만 확장성과 관리적인 측면에서의 단점을 극복하기 위해 혼합 형태를 적용하는 시스템이 늘고 있다. 그리고 최근 분산 컴퓨팅은 사용자에 대한 서비스적인 측면의 기능을 제공하기 위해서 다양한 요구를 적절하게 수용할 수 있는 다양한 형태의 이동 에이전트 기술이 필요하다. 본 논문에서는 이동 에이전트에서 필요로 하는 원격자원에 대한 접근을 위한 실행 및 접근 제어와 에이전트의 관리 기능과 신뢰성을 향상시키기 위해 적합한 이동 에이전트 시스템을 제안하고 설계하였으며 개발하였다.

핵심주제어 : 이동 에이전트, 자원 예약, 변형 P2P

Abstract Recently a grid and cloud computing collaboration have become a social issue. These collaborative network system, the P2P system based on this system. Distinguished from the client/server systems, P2P systems in order to exchange information, its purpose and functions are divided according to the morphological Category. In accordance with the purposes and functions of information and data retrieval, remote program control and integration services for the offers. Most P2P systems client/server scalability, and management takes the form, but to overcome the disadvantages in terms of applying the mixed-mode system is increasing. And recently the distributed computing aspects of the service to users in order to provide suitable to accommodate the diverse needs of various types of mobile agent technology is needed. In this paper, as required by the mobile agent access to a remote resource access control and agent for the execution and management capabilities and improve the reliability of the mobile agent system designed to suggest.

Key Words : Mobile Agent, Resource Reservation, Hybrid P2P

1. 서 론

* 한국교통대학교 의료정보공학과, 제1저자

** 청주대학교 컴퓨터정보공학과, 교신저자
(netause@gmail.com)

*** 청주대학교 컴퓨터정보공학과, 제2저자

최근 인터넷을 이용한 응용 기술인 클라우드 컴퓨팅은 많은 사회적 관심을 받고 있다. 이러한 이유는 IT서비스를 받기 위한 통합 자원 인프라 기술의 우수성과 동시에 사용자 접근성에서의 편리함이라고 할

수 있다. 이렇게 사용자에게 편리한 서비스를 제공하기 위한 논리적인 가상화를 위한 기술은 기존에도 다양한 분야에서 원격지의 자원을 편리하게 제공하기 위해서 발전하였다.

이러한 기술로는 클러스터링, P2P(Peer to Peer), 그리드, 분산 객체 기술 등이 있는데 이러한 시스템들 중에서 P2P 시스템은 분산 시스템에 적용하기가 용이하다[1].

일반 사용자들은 네트워크 및 컴퓨팅 환경이 발전함에 따라 사용자들은 자신의 정보를 직접 자신들이 서비스할 원하게 되었고 이러한 의도로 개발된 P2P 시스템은 클라이언트/서버(C/S)형태에서의 서버에 대한 자원 사용량의 부하와 역할을 줄이기 위한 목적과 모든 컴퓨터에서 자원을 제공할 수 있는 서버 형태가 가능하도록 개발이 되었다.

다양한 목적과 의도에 맞게 설계된 P2P 시스템은 여러 형태로 응용되어 개발되어지고 수많은 네트워크 관련 시스템의 기반 기술로 자리 잡게 되었다. 응용 P2P 시스템의 개발을 통해 분산처리과 인프라를 제공하기 위한 프로그램이 사용자를 위해 서비스 형태로 제공되고 이는 사용자의 입장에서 특정한 서비스를 제공 받는 시스템으로 사용되고 있다.

서비스를 제공하는 형태의 소프트웨어로 많은 전문가 시스템에 사용되는 에이전트 개념이 도입되며 이동 에이전트와 프로그램에 대한 관심도 높아지고 있는 실정이다.

이에 본 논문에서는 사용자 입장에서의 목적과 분류에 따른 협업 서비스를 제공 받기 위한 에이전트 시스템을 제안하고 이 시스템에서의 서비스를 관리하기 위한 서비스 관리자와 이러한 작업을 관리하기 위해 필요한 요소 기술로 자원관리자와 이를 시스템에서 제공받기 위한 접근 방법으로써의 서비스 접근 제어 기법과 접근 관리자를 제안 하였다.

본 논문의 2장에서는 P2P 시스템과 변형 P2P 시스템의 내용을 다루고 에이전트와 이동 에이전트에 대해 기술하였으며 기존의 운영체제와 시스템에서 사용되는 에이전트관리자와 자원관리자 그리고 접근제어 관리자에 대하여 다루었다. 3장에서는 제안하는 이동 에이전트의 기반이 되는 변형 P2P 시스템과 이동 에이전트 시스템의 기능과 동작과정 그리고 제안된 에이전트 서비스 관리자와 자원관리자, 접근제어관리자의 구체적인 내용을 기술하였으며 4장에서는 제안하

는 에이전트의 기능에 대한 기존 에이전트와의 비교표를 통하여 평가하였고 마지막으로 5장에서는 결론과 향후 과제에 대해 기술하였다.

2. 관련 연구

논문에서 제안된 에이전트 시스템과 관리자의 구성은 아래 그림과 같다.



<Fig 1> 제안 에이전트 시스템의 구성도

제안된 에이전트 시스템은 기반 시스템을 변형 P2P 시스템으로 개발되었다. 기반 시스템은 연결성을 유지하고 신뢰성을 제공하기 위한 시스템으로써 기존의 C/S 형태의 P2P와 계층 형태의 P2P를 변형하여 설계하고 제안하였다. 이러한 변형의 P2P 시스템에서 서비스를 제공하기 위한 기법으로 서비스 이동 에이전트를 제안하였으며 설계하였다.

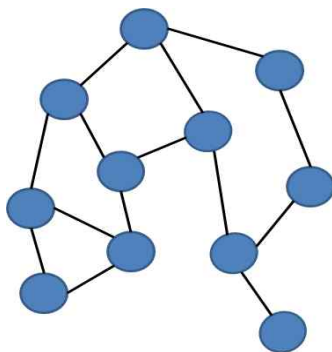
관련 연구로는 제안된 시스템에 필요한 기반 시스템인 P2P 시스템과 상위 서비스 계층의 에이전트 시스템과 에이전트 시스템에서 협업을 위해 필요한 에이전트 관리자와 자원관리자, 그리고 접근제어관리자를 제안하기 위하여 기존의 시스템들과 내용을 기술하였다.

2.1 P2P와 변형 P2P

순수 분산 P2P 형태의 대표적인 모델인 Gnutella 네트워크는 2003년 3월경에 저스틴 프랭클(Justine Frankel), 톰 페퍼(Tom Peper)에 의해 음악파일을 공

유하기 위해서 아래와 같은 그물망형태로 만들어졌다 [1,2].

이러한 Gnutella는 그물망형태로 순수 P2P형태로써 연결성에 있어서 장점을 지니고 있고 서버가 존재하지 않기 때문에 C/S 형태가 지닌 단점인 서버의 부하를 줄일 수 있지만 사용자가 원하는정보를 찾기 위한 방법으로 논리적인 라우팅을 사용하고 메시지를 연결된 모든 컴퓨터에게 전달하기 때문에 수많은 트래픽을 발생시키는 단점을 지닌다[3,4].



<Fig 2> Gnutella의 형태

이러한 단점을 극복하기 위한 방법으로 서로 다른 기본형태의 조합과 응용을 통해 개발된 것이 변형(Hybrid) P2P 시스템이다. 이러한 시스템의 예로는 C/S 형태와 계층형태를 조합하여 만들어진 E-Donkey를 들 수 있다. 이러한 응용 형태는 최근의 개발 시스템에서만 볼 수 있는 것은 아니다. 1979년 미국의 듀크 대학교 톰 스루스코트(Tom T.)와 짐 엘리스(Jim E.)가 유닉스 공동체와의 정보교환을 위해 연결하고 이를 통해 스티브 벨로빈(Steve B.)이 처음 개발한 유즈넷(Usenet)시스템도 도메인을 포함한 계층형태에 각 도메인에 여러 사람이 공유할 수 있는 C/S 형태를 취한 변형의 서비스라고 볼 수 있다. 이렇게 많은 목적과 기능에 따라 P2P 시스템의 형태도 변형된 형태로 사용되고 있다.

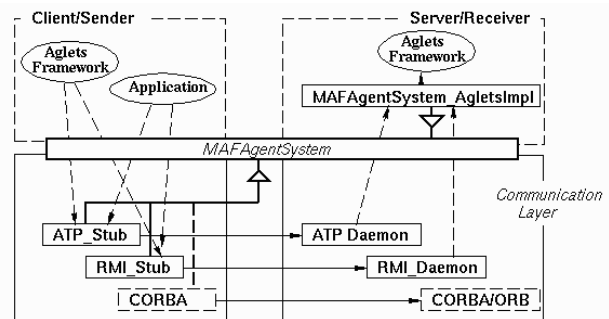
2.2 에이전트와 이동 에이전트

에이전트란 누군가를 위하여 무엇인가를 대신해주는 컴퓨터 프로그램을 통칭한다. 이러한 에이전트는 지속적으로 환경에서 지각된 것과 내부 지식을 바탕으로 추론하며, 그 결과 행동하여 환경에 영향을 미친

다. 구체적인 학술적 정의에 따르면 에이전트란 특정 환경 내에 위치하여 에이전트의 설계된 목적을 만족시키기 위하여 자율적으로 유연하게 행동할 능력이 있는 컴퓨터 시스템이다[5].

이러한 에이전트 시스템은 90년대에 들어서 인터넷과 웹의 급속한 성장과 더불어 이동 에이전트로 발전하게 되었다. 이동 에이전트는 기존의 에이전트 개념에서 역할 수행을 위해 에이전트가 다른 시스템으로 이동할 수 있는 개념이 추가된 것이다. 이러한 이동 에이전트의 기능을 수행할 수 있는 시스템을 자바의 클래스와 적절하게 적용하여 개발한 IBM의 Aglet이 대표적이라 할 수 있다.

Aglet은 클래스를 하나의 에이전트로 볼 때 이러한 클래스가 서로 다른 시스템에서 네트워크를 통해서 필요에 의해 옮겨 다닐 수 있고 이를 통해 서비스를 제공할 수 있다는 개념에서 개발되었다. ATP(Agent Transfer Protocol)를 이용하여 에이전트의 상태를 4개 역할로 구분하여 수행하고 이를 통해 서비스를 제공하도록 개발되었다[6]. Aglet의 통신 체계와 역할은 다음 그림과 같다.



<Fig 3> Aglet 시스템의 계층 구성도

2.3 에이전트 관리자와 자원 관리자

에이전트 관리자는 각각의 기능을 지닌 이동형 에이전트에 대한 관리 기능으로써 에이전트 소프트웨어에서 보다 운영체제의 프로그램 관리에서 사용이 되어왔다. 윈도우 운영체제에서는 프로그램에 대한 설치와 삭제 및 관리를 위하여 제어판에서 프로그램 관리 기능을 제공하며 설치된 프로그램의 실행에 대하여 작업관리자(Task Manager)를 두어 프로그램의 활성 상태에 대한 정보와 프로세스에 대한 정보 및 제어

기능, 그리고 중앙처리장치, 메모리, 네트워크 등 자원에 대한 사용량도 모니터링 할 수 있도록 제공하고 있다.

이러한 기능은 프로그램에 대한 관리의 차원과 운영체제로써의 편의성과 관리적인 차원에서 서비스로 제공되고 있으며 유닉스와 리눅스 시스템에서도 프로세스에 대한 제어와 상태에 대하여 제공되는 명령어와 기능들을 제공하고 있다. Aglet에서는 이동된 에이전트 소프트웨어가 보이며 에이전트에 대한 상태가 표시되는 기능을 제공한다.

이러한 기존의 프로그램과 에이전트에 대한 관리 기능과 자원에 대한 상태 및 제어 기능은 이동형 에이전트를 이용한 협업 시스템에서는 서비스 제공자와 에이전트들 사이에 필요한 구체적인 관리 기능 개선이 추가적으로 필요하다.

2.4 에이전트 접근 관리자

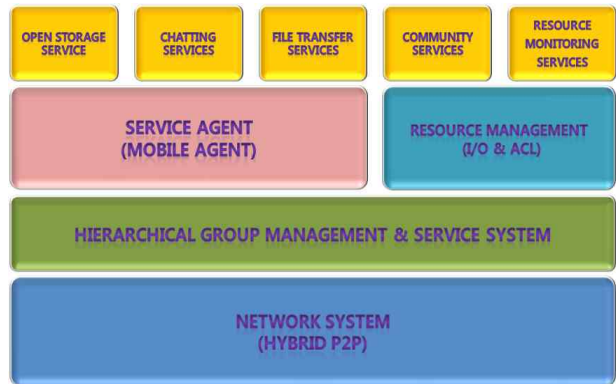
이동 에이전트의 보안은 전달된 에이전트가 컴퓨터에 대해 공격을 가할 수 있고 정보를 유출할 수 있기 때문에 필수 요소이다[7]. 접근 관리자의 기능은 Aglet에서도 에이전트의 보안성에 따른 접근 계정 기능으로 사용되었다[8]. 이러한 접근 방법과 기술은 기존의 유닉스와 리눅스 시스템에서도 사용되었다. 전체의 파일 시스템에서 하나의 파일에 대한 사용자의 접근을 접근 모드로 제공하고 여러 사람이 함께 사용될 수 있는 다중 사용자에게 대한 계정관리를 제공하여 관리자와 권한사용자가 이를 설정하고 그에 따라 접근의 여부를 판단할 수 있는 시스템이다.

하지만 접근 계정 및 제어 기능은 분산처리와 협업을 하기 위한 시스템에는 기존의 운영체제에서 제공하는 파일 접근 수준의 기능만을 제공해서는 보안 요소로써 부적합하며 이를 에이전트 자원관리자와 함께 제공하기 위한 기능도 제공되어야 한다.

3. 제안 시스템

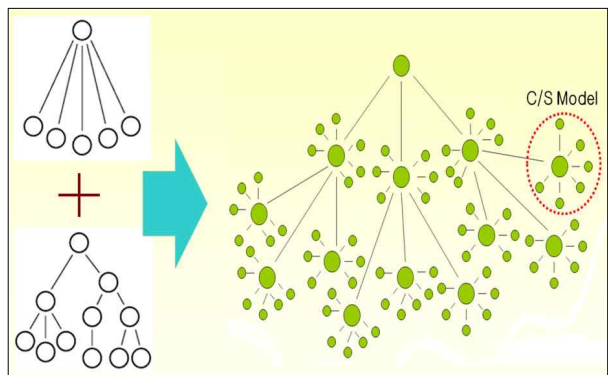
3.1 제안 시스템의 구성

시스템을 이루고 있는 플랫폼의 구조는 아래 그림과 같다.



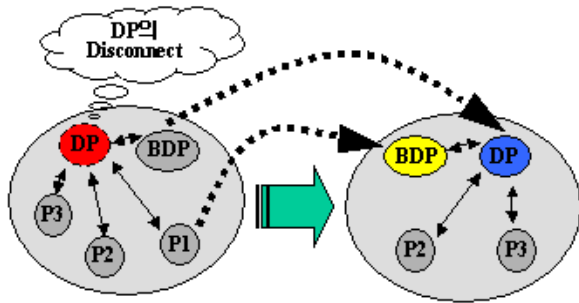
<Fig 4> 제안 시스템의 구조도

네트워크의 연결성에 대한 신뢰성 제공 및 관리를 위한 기반 시스템으로 변형 P2P 시스템을 사용하였다. 적용된 변형 P2P는 기존의 중앙 집중 형태와 계층 형태를 아래와 같이 변형한 형태를 취하고 있다.



<Fig 5> 변형 P2P의 적용 형태

변형 P2P를 적용한 기반 시스템에 다양한 서비스를 제공하기 위한 서비스 에이전트가 계층의 노드마다 C/S 형태로 연결성을 유지하는 그룹 형태로 존재한다. 그리고 그룹에서 연결성을 제공하기 위한 수단으로 그룹 내의 서버 에이전트 역할을 하는 DP(Designated Peer)와 호스트 에이전트 중 서버 에이전트의 보조 역할을 하는 BDP(Backup Designated Peer)를 지정하여 아래 그림과 같은 자동 역할 위임 기능인 재 선출 기법을 사용하여 연결성을 유지하고 서비스를 제공할 수 있도록 설계 하였다.



<Fig 6> 자동 재 선출 과정

이를 통해 사용자에게 원하는 그룹에서의 서비스 제공과 분산처리 및 자원제공 서비스 등을 제공하기 위한 접근 및 그룹 관리와 연결 유지 기능을 제공한다.

3.2 제안 에이전트 시스템의 동작 과정

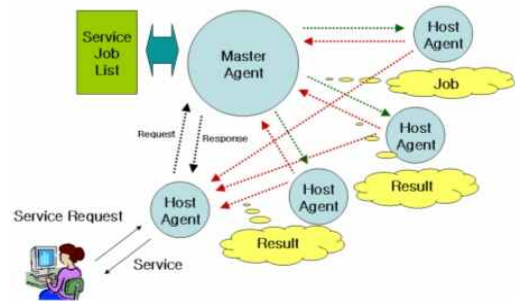
기본 시스템을 통해 서비스를 위한 에이전트 시스템은 그룹에서 이루어지며 개발 언어는 자바를 사용하였다. 클래스의 형태로 이동되며 이는 기존의 에이전트 시스템인 Aglet과 유사하지만 서비스가 에이전트와 함께 제공되며 동작과정에 있어서 기본 시스템의 정보와 자원을 활용하는 측면에서 장점을 지니고 있다.

통신을 담당하는 기능은 메시지 처리 방법을 지닌 비동기식 메시지 전달방식을 활용했다. 이러한 방식은 기본 시스템인 P2P 시스템의 그룹 관리에 관한 프로토콜과 에이전트 기반의 통신 프로토콜에 대한 통합의 기능을 중심으로 설계되었다.

에이전트 통신 시스템은 데이터를 전달하기 보다는 제어 정보와 관리 및 절차에 필요한 정보를 전달하기에 용이하다. 이러한 방법으로 전달될 자바의 클래스 기반의 서비스가 전달되고 실행되며 제어되는 시스템이다.

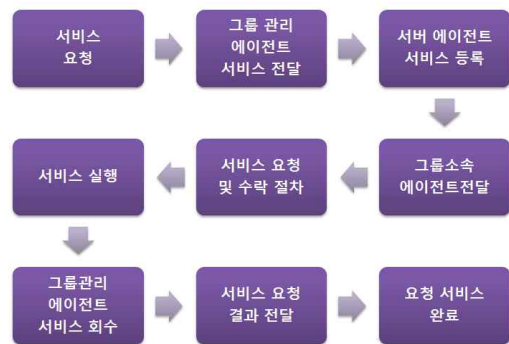
서비스 에이전트 시스템의 동작은 그룹에 소속된 컴퓨터에서 이루어지며 서비스를 요구하기 위해 에이전트를 이용한 서비스 목록이 제공되고 이를 통해 원하는 서비스를 요청하게 된다. 이러한 요청 서비스는 그룹에서의 서버에 해당하는 서비스 관리 에이전트로 전달되며 이로써 그룹 내의 서비스로 등록이 되며 이를 통해 접속된 그룹 소속의 컴퓨터들에게 전달되고

분산처리 및 자원 활용에 대한 서비스의 실행이 이루어진다.



<Fig 7> 제안 이동 에이전트의 구성도

이렇게 전달된 그룹 내의 여러 컴퓨터들이 처리된 결과를 다시 그룹 내에서의 서버 에이전트가 수집하고 회수하며 이를 서비스 요청했던 컴퓨터로 전달되는 아래 그림과 과정을 지니고 있다.



<Fig 8> 에이전트의 서비스 처리 흐름

3.3 이동 에이전트 서비스 관리자

제안된 에이전트 시스템은 에이전트 연결 및 접속 관리와 서비스 실행 및 전송 관리 그리고 자원 및 정보 관리와 접근 관리로 세분화 할 수 있다.

서비스에 대한 실행 및 전송 관리는 자바의 클래스와 패키지 및 개발환경에 대한 정보를 포함하여 서비스의 서버와 클라이언트로 구분되어 전달되며 이는 서비스를 제공한 에이전트가 전달하는 형태로 제공되고 사용자는 전달된 서비스의 내용을 잔달 받을 수 있으며 허가 할 수 있는 형태로 제공된다. 이는 현재의 인터넷의 추가 기능을 담당하는 ActiveX 기술처럼

서비스의 요청을 받은 컴퓨터의 허가에 의해 설치될 수 있으며 임의로 삭제 할 수 있고 필요에 의해 요청 받은 서비스를 반대로 요구 할 수도 있다.

서비스에 대한 제어는 서비스 기능을 포함하고 있는 에이전트의 코드 내에 존재하며 이를 위한 자원 관리자와의 통신도 이루어져야한다.

3.4 이동 에이전트 자원 관리자

자원관리자는 원격에서의 에이전트에 대한 자원의 활용 및 사용현황 정보를 제공하기 위해 만들어졌다. 하지만 날로 보안과 해킹에 대한 우려를 고려하여 에이전트의 고유 기능이 외의 접근에 대한 자원의 현황과 예약에 대하여 다루어지고 있다.

자원관리자의 기능은 크게 자원 제공과 자원 현황 정보 제공 기능으로 구분할 수 있다. 자원에 대한 제공은 컴퓨터에 대한 원격지에서의 자원을 의미하며 입·출력장치와 중앙처리장치와 주 기억장치, 내·외부 저장장치에 대한 활용 자원을 의미한다. 이를 제공하기 위해서는 할당 정책과 예약 및 활용에 대한 정보를 서비스하기 위한 공용 스키마 설정이 필요하다.

이를 통해 서비스 에이전트는 서비스를 제공 받기 위한 스키마를 미리 제공하여 원격지의 사용자로부터 허가를 받을 수 있는 신뢰성을 제공할 수 있다.

자원 현황정보 제공에 대한 기능은 개발 코드에 부여되는 기능으로 제공되어야하며 이를 위하여 공용 스키마에 대한 정책과 방법에 따라 실시간으로 제공되어야 할 자원에 대한 접근이 발생하는 시점의 모든 개발코드에 추가적으로 정보제공에 관련된 코드가 기록되어야한다. 이를 통해 신뢰성을 제공 받을 수 있으며 이에 대한 접근의 제어를 위해 이동 에이전트의 기능과 역할에 따른 접근 관리자와의 통신이 필요하다.

3.5 이동 에이전트 접근 관리자

컴퓨터에서 터미널의 접속에 대한 기능의 제한을 위해 권한을 부여하기 시작했고 이는 파일에 대한 접근을 전제로 읽기, 쓰기, 실행에 대한 권한으로 부여되었다. 이후로 원격 제어 및 분산처리, 에이전트 등의 시스템에서의 프로그램 실행은 실행 불가에 대한 기준으로 적용되었다. 하지만 이동 에이전트 시스템에서

는 이러한 에이전트의 적합성과 이를 허용했을 때에 발생 할 수 있는 문제를 고려하여 본 논문에서는 이러한 문제를 에이전트의 접근에 대한 구체적인 제어 구분 정보를 아래와 같은 표로 제안하여 이를 설계하였다.

<Table 1> 제안 자원 접근 권한 분류표

기본 실행	• CPU, Memory, Network 접근가능
외부 자원 접근	• HardDisk, External Devices
초과 자원 접근	• 에이전트 스키마 정의 내용 참조
Admin 권한	• Registry 및 Program, Data, Devices 관리 권한

시스템에 대한 자원 중 중앙처리장치(CPU)는 향후 이동 에이전트의 병렬처리를 위한 기능과 네트워크의 대역폭은 향후 이동 에이전트의 다중 경로와 다중 전송 방법의 기반을 위한 예약으로 설계하였다. 이는 앞서 설계한 자원 관리자와의 통신과 제어에 대한 정보의 공유로써 활용된다. 제안된 접근 관리자는 그 정보를 XML(Extended Markup Language)을 활용하여 아래와 같이 설계하였다.

<Table 2> 서비스 에이전트 정보 스키마

```

// 상기 내용 생략
<internalresource>
  <cpu>
    <max> 50 </max>
    <min> 0 </min>
  </cpu>
  <mem>
    <max> 100000 </max>
    <min> 1000 </min>
  </mem>
  <network>
    <max> 100000 </max>
    <min> 0 </min>
  </network>
  <harddisk>
    <enable> 1 <enable>
  </harddisk>
// 이하 내용 생략

```

설계된 접근 자원 예약 시스템 하드디스크에 대한 예약은 논리적으로 하드디스크를 지정한 용량만큼 할당하여 공유 및 분산 정보의 형태를 파일 형태로 예약한 공간에 기록하여 활용하기 위한 방법으로 설계되었고 메모리에 대한 예약은 내부 배열 형태로 크기

만큼 생성자로 할당하여 외부에서 접근 가능한 메모리로 활용할 수 있도록 설계하였다.

4. 제안 시스템의 평가

4.1 형태적 비교

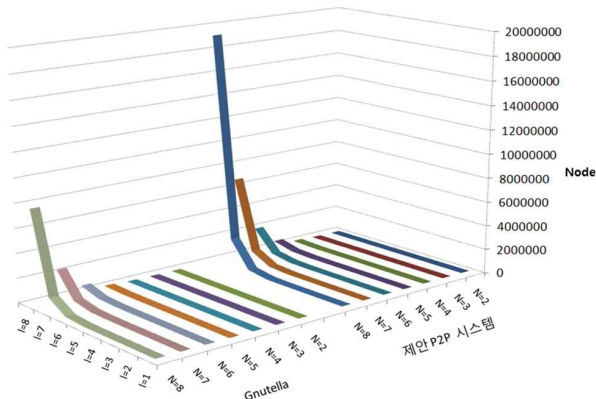
제안된 에이전트의 구성 형태는 변형 P2P에 의해 형성되었다. 이 변형의 P2P 시스템에 대한 확장성을 계산하기 위하여 순수 P2P 방식 중 확장성이 높은 그룹망형태를 지닌 Gnutella를 사용하였다.

아래 표와 같은 변수의 정의와 함수식을 이용하였고 N은 노드가 인접한 노드들과 연결되어 있을 때의 다른 연결된 노드들의 수를 의미하며 1 계층 구조나 연결 구조에서의 깊이를 측정하는 단계를 의미한다.

<Table 3> 시스템 확장성과 트래픽 계산 함수식

분석에 사용된 변수의 정의와 함수식	
N	연결된 다른 노드수
1	계층 구조에 있어서의 Level 깊이
$F(n, x, y)$	$f(n, x, y) = \text{Sum}[(n^l) + f(n-1, x, y)], l = x \rightarrow y]$
$h(n, l, s)$	$h(n, l, s) = s * f(n-1, x, y) + s$

노드 x로부터 노드 y로 가는데 걸리는 Level의 깊이를 l이라고 할 때, l만큼의 Level로 노드 x로부터



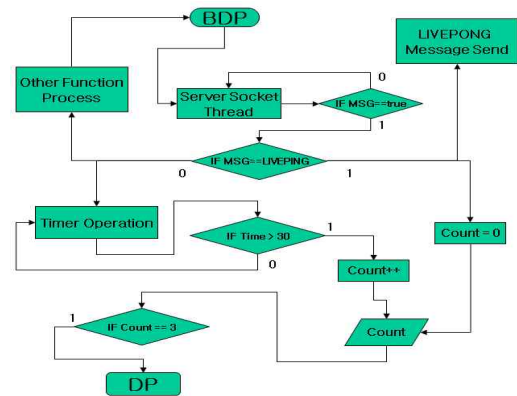
<Fig 9> 기반 시스템의 확장성 비교

터 인접할 수 있는 노드들의 최댓값을 $F(n, x, y)$ 식으로 작성하였고 여기서 1은 에이전트와 에이전트간의 거리를 의미하며 N은 에이전트에 접근한 인접 에이전트에 대한 개수를 의미한다. 포괄적으로 node는 에이전트 전체의 시스템에서 수용할 수 있는 노드의 수를 의미한다. 계산 결과에 따른 Gnutella와의 비교 그래프는 <그림 9>와 같다.

이러한 기반시스템은 접근성 측면에서 Aglet이 접근 가능한 에이전트 서버의 노드만 접근 가능한 것을 보완하였으며 P2P 시스템 중 확장성에 장점을 지닌 Gnutella의 확장성과 비교를 통해 기반시스템이 에이전트 노드 추가와 연결에 필요한 확장성이 높다는 것을 증명하였다.

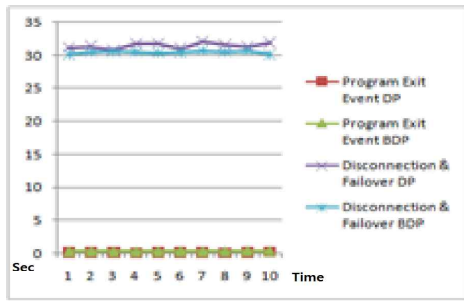
4.2 안정성 비교

제안된 시스템은 기존에 에이전트에서는 제안되지 않은 서비스 기반 자체의 연결성을 자체 복구하는 자동 역할 위임 기능으로 설계되었다. 에이전트 서비스의 연결성을 유지하고 제공하는 것이 중요한 만큼 연결성을 유지하고 복구하기 위한 방법을 앞서 제안 시스템의 구조에서 언급한 DP와 BDP의 역할 재위임 기법으로 아래 그림과 같이 설계하였다.



<Fig 10> 역할 재위임 순서도

프로그램과 네트워크의 이상상태에 따른 이벤트가 발생했을 때 서비스를 복구할 수 있는 서버의 재위임 시간을 측정하여 아래 그림과 같이 그래프로 나타내었다.



<그림 11> 역할 재위임 소요 시간

위의 그래프와 같이 DP와 BDP가 물리적인 연결성을 잃어버린 경우 3초를 넘게 되는데 기본적으로 연결성을 재확인하고 검사하는 시간으로 3초를 지정했기 때문에 나타난 결과이다.

4.3 정성적 비교

기존의 모바일 에이전트는 ATP(Agent Transfer Protocol)을 활용하며 제안 시스템은 자체적으로 아래 표와 같은 프로토콜로 설계하였고 <그림 7>에서와 같이 MA(Master Agent)와 HA(Host Agent)로 구분하고 서로의 상태에 대한 정보와 제어 정보로 나누어 설계하였다.

<Table 4> 제안 에이전트 프로토콜

HA → MA				
SLR (Service List Request)	ID		IP	
SLA (Service List Answer)	Sign			
SL (Service List)	Agent ID	Agent Description	...	End List
ATR (Agent Transfer Request)	Agent ID	Agent Name	ID	IP
ATA (Agent Transfer Answer)	Agent Name	Agent Size	Schema Info	Schema Size
SAT (Service Agent Transfer)	Base File Transfer Protocol Use			
SR (Service Request)	Agent ID	Agent Name	ID	IP
SA (Service Answer)	Sign			
MA → HA				
SRR (Service Run Request)	Agent Name	Agent Description	Request ID	Request IP
SRA (Service Run Answer)	Sign			
SL (Service List)	Agent ID	Agent Description	...	End List
ATR (Agent Transfer Request)	Agent ID	Agent Name	ID	IP
ATA (Agent Transfer Answer)	Agent Name	Agent Size	Schema Info	Schema Size
SAT (Service Agent Transfer)	Base File Transfer Protocol Use			
SR (Service Request)	Agent ID	Agent Name	ID	IP
SA (Service Answer)	Sign			
State Access & Control				
AMR (Agent Move Request)	Agent ID	Agent Name	ID	IP
AMA (Agent Move Answer)	Agent ID	Agent Name	ID	IP
AM (Agent Move)	Agent ID	Agent Size	...	End
AME (Agent Move End)	Agent ID	Agent Name	AgentMoveEnd	
AE (Agent Excute)	Agent ID	Agent Name	Schema Info	Schema Size
ASR (Agent State Request)	Agent ID	Agent Name	ID	Flag
ASA (Agent State Answer)	Agent ID	Agent Name	State Information OR Result	
AQ (Agent Quit)	Agent ID	Agent Name	ID	IP
AR (Agent Return)	Agent ID	Agent Name	ID	IP

추가적으로 Aglet에는 없는 보안성을 고려하여 응

용 RSA 암호화 기법을 활용한 프로토콜 전송기법을 활용하였으며 쌍방 공개키 교환기법으로 프로토콜 보안을 강화하였다. Aglet의 접근 관리는 단순하게 이동 에이전트가 이주할 수 있는지에 대한 여부에 따라 이루어졌다[8].

이러한 관리는 차후 협업 및 분산처리 업무에 일반 사용자의 허용과 결합될 수 없으며 이에 제안 접근 제어 관리자를 포함한 형태의 제안 에이전트 시스템의 장점과 특징에 대하여 대표적인 이동 에이전트인 Aglet과 비교 분석하면 다음의 표와 같다.

<Table 5> 제안 시스템과 Aglet과의 비교표

구 분	Aglet	제안 이동 에이전트
접근성 측면	독립적 소프트웨어로 접근이 어려움	그룹을 제공하는 기반 P2P 시스템으로 접근이 용이함
확장적 측면	서버/클라이언트 형태로 제한적임	혼합 계층형태로 확장이 용이함
제어적 측면	실행의 여부에 따라 결정되는 형태임	실행 조건의 세분화된 정보에 의해 접근
관리적 측면	에이전트 수행 시의 자원 정보 제공이 없음	자원 관리자가 에이전트 역할에 따른 자원을 관리함
보안적 측면	프로토콜 보안성 측면 취약	응용 RSA기법으로 프로토콜 보안성 강화

5. 결론 및 향후 연구 과제

본 논문에서는 변형 P2P를 활용한 서비스 관리 이동 에이전트에서의 에이전트 관리자와 자원 관리자, 그리고 접근 제어 관리에 대한 내용과 방법을 제안하고 설계하였다.

많은 자원을 분산 처리하기 위한 노력과 원격에서 자원을 활용하기 위한 기술은 날로 발전하고 있으며 특히 최근 에이전트의 확장을 위한 네트워크의 응용과 접목에 대한 연구도 제안되고 있다[13]. 또한 협업과 실시간 통합 서비스를 위한 그리드 시스템과 원격지의 잉여 자원에 대한 통합과 서비스의 제공을 위한 클라우드 컴퓨팅에 대한 연구가 각 응용 분야에서 활발하게 연구되어 지고 있다.

홈 네트워크와 자동화는 유비쿼터스에 대한 기술로 점차 이루어지고 있으며 사무실 환경과 커뮤니티에

대한 사용자들의 요구로 점차 모든 일들에 대한 원격지의 처리에 대한 기술은 필요로 할 것이며 특정 그룹에 대한 서비스가 아닌 통합과 협업에 대한 개념의 서비스 제공은 여러 분야에서 활용될 것이다.

제안 내용에서 에이전트에 대한 관리자는 현재의 컴퓨터 운영체제에서 관리하는 프로그램과 프로세스에 대한 관리와 흡사하지만 설치한다는 개념과 전달된다는 의미에서 차이가 존재하며 이는 자원에 대하여 자바의 기술처럼 독립적일 수 있다면 향후 같은 형태로 발전이 될 것이다.

앞으로 구체적으로 자원관리를 위한 실시간 자원 접근 정보를 위해 언어와 개발환경 차원에서의 기능적인 개발 요소가 필요하며 사용자 입장에서의 접근 제어 관리의 편리함을 위한 사용자 인터페이스에 대한 연구와 개발이 계속되어야 한다.

참 고 문 헌

- [1] Andy Oram, "Peer-to-Peer - Harnessing the Power of Disruptive Technologies", O'Reilly, September, 2001.
- [2] Nelson Minar, "Distributed Systems Topologies : Part 1", http://www.openp2p.com/pub/a/p2p/2001/12/14/topologies_one.html, December 14, 2001.
- [3] Clip2 Distributed Search Solutions , "The Gnutella Protocol Specification v0.4. Distributed Search Services," <http://dss.clip2.com>, September 2000.
- [4] Jordan Ritter, "Why Gnutella Can't Scale. No Really.", <http://www.darkridge.com/~jpr5/doc/gnutella.html>, February, 2001.
- [5] 이광형, 조충호, "인공지능개론", 홍릉과학출판사, 2000년 9월.
- [6] Danny B. Lange, Mitsuru Oshima, Mobile Agent with Java: The Aglet API, <http://www.moe-lange.com/danny/wwwj.pdf>
- [7] Mitsuru Oshima, Guenter Karjoth, and Kouichi Ono, "Aglets Specification 1.1", Draft 0.65, September 1998.
- [8] uca Ferrari, "The Aglets 2.0.2 User's Manual", September 18, 2004. <http://aglets.sourceforge.net>
- [9] chiro Satoh, "Selection of Mobile Agents", to appear in Proceedings of 24th IEEE International Conference on Distributed Computing Systems (ICDCS'2004), IEEE Computer Society, March 2004.
- [10] Bhandaru N. and Croft W., "An architecture for supporting goal-based cooperative work", in Gibbs S. and Verrjn-Stuart A., eds., Multi-User Interfaces and Applications, pp 337 - 354, Elsevier Science Publishers B.V., North-Holland, 1990.
- [11] http://en.wikipedia.org/wiki/EDonkey_network#Features
- [12] Brian Kantor, Phil Lapsley, "Network News Transfer Protocol" RFC 977. February, 1986.
- [13] A.K. Sharma, Atul Mishra, Vijay Singh, "An Intelligent Mobile-Agent Based Scalable Network Management Architecture For Large-Scale Enterprise System", International Journal of Computer Networks & Communications (IJCNC), Vol.4, No.1, January 2012.
- [14] 오동, 김승, 박순철, "이동 에이전트 엔진을 이용한 웹 기반 정보 검색 시스템의 설계 및 구현", 한국산업정보학회, Vol.4, No.4, pp.79-87, 1999.
- [15] 정종진, "분산 환경 하에서 지능형 에이전트 기반의 자동 스케줄링을 위한 제약만족 기법의 응용", 한국산업정보학회, v.12, no.1, pp.87-94, 2007.



이 석 회 (Seok-Hee Lee)

- 정회원
- 청주대학교 컴퓨터정보공학과 공학사
- 청주대학교 전산정보공학과 공학석사
- 청주대학교 컴퓨터정보공학과 공학박사
- 한국교통대학교 보건생명대학 의료정보공학과 외래교수
- 관심분야 : 클라우드 컴퓨팅, 이동 에이전트, 분산처리



양 일 등 (Il-Deung Yang)

- 학생회원
- 청주대학교 컴퓨터정보공학과 공학사
- 청주대학교 대학원 컴퓨터정보공학과 공학석사
- 청주대학교 대학원 컴퓨터정보공학과 박사과정
- 관심분야 : 영상처리, 휴먼 네트워크



김 성 열 (Seong-Ryeol Kim)

- 정회원
- 숭실대학교 전자계산학과 공학사
- 숭실대학교 대학원 전자계산학과 공학석사
- 숭실대학교 대학원 전자계산학과 공학박사
- 청주대학교 이공대학 컴퓨터정보공학과 교수
- 관심분야 : 컴퓨터 네트워크, 컴퓨터 보안, 분산 객체 시스템

논문접수일 : 2012년 06월 04일
 1차수정완료일 : 2012년 08월 20일
 게재확정일 : 2012년 08월 30일