



KR0000270

KAERI/TR -1591/2000

에이전트의 발전 및 응용에 관한 연구

A Study on Development and Applications of Agents

2000 년 6 월

한국 원자력 연구소

Korea Atomic Energy Research Institute

**Please be aware that all of the Missing Pages in this document were
originally blank pages**

제 출 문

한국원자력연구소장 귀하

본 보고서를 “에이전트의 발전 및 응용에 관한 연구” 보고서로 제출합니다.

2000 년 6 월

과 제 명 : 중대사고 위해도 완화전략 분석 종합전산코드 개발

주 저 자 : 박선희

공 저 자 : 손재민

요 약

에이전트(Agent)의 연구는 초기의 인공지능 분야의 한정된 연구에서 고도의 분산 복합 시스템을 설계하여 구현하는 실용적인 소프트웨어 개발에 이르기까지 다양하게 적용되고 있다. 인공지능의 세분화된 연구 분야 중 분산 인공지능이라는 새로운 연구 분야가 형성되면서 에이전트를 토대로 문제를 해결하려는 시도가 시작되었다. 이와 관련하여 사용자가 원하는 작업을 쉽게 수행할 수 있도록 편리함을 제공해 주는 지능형 시스템(Intelligent System)에 대한 요구가 증대되어 왔으며, 최근에 와서 이 문제에 대해 인터넷에의 접근이 시도되면서 사용자가 원하는 작업을 자동적으로 대신 해결하고 처리하여 주는 다양한 형태의 에이전트가 개발되었다.

본 논문에서는 이러한 에이전트와 관련된 시스템들의 기본 개념과 연구 방향 및 그 응용에 대해 분석하였으며, 에이전트가 등장한 배경, 필요성, 기본 개념 및 특성 등 에이전트의 개요와 구성 요소에 대하여 면밀하게 분석하였다. 그리고 현재 까지 수행되어 온 에이전트에 대한 국내외의 연구 현황을 분석하여, 지능 에이전트의 개발과 관련된 특성에 대해 기술하였다. 또한 에이전트가 실생활에 적용될 수 있는 분야에 대하여 보편화된 에이전트의 응용 및 사례에 대해 연구하였고, 특히 지능 에이전트의 기능 및 서비스에 대해 분석하였다.

에이전트 기술의 적용 분야는 웹 사이트(Web Site)에서 정보의 검색 뿐 아니라 인터넷 정보의 가공, 전자상거래, 산업체 및 의료 분야 등과 같이 점차 다양해지고 있다. 본 연구는 이러한 다양한 에이전트에 대해 전반적인 개념을 분석하고 비교해 봄으로써 지능 에이전트에 대한 이해와 향후 이 분야의 연구에 대한 방향 정립에 도움이 되고자 한다.

ABSTRACT

An agent study is variously applied from the restricted research of an early AI field to the practical SW development of the design and implementation of a high level, distributed hybrid system. Among the detailed research fields of AI, making a new research field called Distributed AI, the trial of resolving a problem based on an agent, has begun. Related to it, the requirement of an intelligent system to offer convenience to easily do a job which the user wants, with the recent trial of an approach for internet agents of various types, were developed to be performed automatically and the agent anticipate the user's need.

In this paper, the basic concepts regarding an agent system, research direction and applications were analyzed and outlined. The consisting elements were also analyzed, such as background, needs, basic concepts and properties of the agent. The agent's research status was recently analyzed, and the development and related property of an intelligent agent were written. Application and the instance of a generalized agent about applicable fields in the real world were studied, functions and services about intelligent agents were analyzed.

The applicable field of an agent technique becomes various not only in information retrieval on a web site, but also in the processing of internet information, and in the electronic commerce, industry and medical fields. Through analyzing and comparing the general concept of various agents, this study will aid in establishing a future direction of research in this field.

목 차

1.	서 론.....	1
2.	에이전트의 개념.....	3
2.1	에이전트의 기본 개념.....	3
2.1.1	특징.....	3
2.1.2	지식기반 시스템.....	5
2.2	에이전트의 구성 요소.....	8
2.3	에이전트 언어.....	10
2.4	에이전트의 분류와 특성.....	11
2.4.1	논리 구성 및 반응성에 따른 분류.....	12
2.4.2	이동성에 따른 분류.....	13
2.4.3	다른 에이전트와의 관계에 따른 분류.....	15
2.4.4	사회성에 따른 분류.....	17
3.	에이전트의 발전.....	19
3.1	에이전트의 발전 배경.....	19
3.2	에이전트의 연구 동향.....	19
3.2.1	에이전트의 구성 방식에 관한 연구 동향.....	20
3.2.2	에이전트의 기능에 관한 연구 동향.....	21
3.3	에이전트 표준화의 동향.....	24
4.	에이전트의 응용.....	27
4.1	숙고형 에이전트 분야.....	27
4.2	협력 에이전트 분야.....	32
4.3	보조 에이전트 분야.....	38
4.4	멀티 에이전트 분야.....	41
5.	결론.....	45
	참고문헌	46

그림 목 차

그림 2-1. 에이전트의 구조	9
그림 2-2. ICL의 구조	11
그림 4-1. 구인구직 에이전트 시스템의 구조	28
그림 4-2. 결재 관리 에이전트 시스템의 구조	29
그림 4-3. 일정 관리 에이전트 시스템의 구조	30
그림 4-4. 논문수집 에이전트 시스템의 구조	35
그림 4-5. 불건전 기사 검색 에이전트 시스템의 구조	36
그림 4-6. 카타로그 에이전트 시스템의 구조	37
그림 4-7. 웹 에이전트의 구조	38
그림 4-8. 진료 예약 에이전트 시스템의 구조	44

표 목 차

표 3-1. 관점에 따른 에이전트의 분류	12
------------------------------	----

1. 서론

에이전트는 단어의 뜻인 “대리인”이 의미하듯이 사용자가 원하는 작업을 대신 수행한다는 의미로 사용된다. 즉, 사용자가 원하는 작업을 대신 수행하기도 하고, 나아가서 사용자가 원하는 취향을 학습하여 자율적으로 사용자의 일을 처리하는 등의 의미로 사용된다.

에이전트에 관한 연구는 인공지능 분야의 사실상의 주된 연구 과제였다고 할 수 있다. 인공지능 분야의 초창기에 연구된 지능 시스템이 좀 더 구체적인 연구로 발전되어 지식 표현, 전문가 시스템, 기계 학습 등으로 세분화되었으며, 70년대 후반에 분산 인공지능이라는 연구 분야가 형성되어 분산 인공지능 문제들을 에이전트를 토대로 해결하려는 시도가 시작되었다. 80년대와 90년대를 지나면서 에이전트라는 용어가 기존의 컴퓨터 연구 분야에서 사용되기 시작하였고, 90년대에 들어 인터넷과 웹의 급속한 확장과 함께 웹 정보환경의 에이전트의 출현을 맞게 되었다. 최근에 와서 인터넷과 웹의 확장으로 웹 정보환경하의 에이전트가 출현하게 되었다. 이러한 에이전트는 사용자를 대신하여 주어진 일을 수행하는 대행자나 중개자 역할을 하며 더 나아가 개인 비서 역할의 에이전트를 개발하려는 연구도 이와 관련이 있다.

에이전트는 특성상 정보 검색 뿐 아니라 정보의 여과 등과 같은 인터넷 정보가공을 위해 쉽게 적용될 수 있는 기술이다[1,2]. 또한 다중 에이전트를 이용한 공정 스케줄링, 협력 설계 등의 제조 분야, 자율적 에이전트를 활용한 항공관제 분야, 지역적 분산성으로 인해 에이전트 시스템의 활용이 적합한 물류 분야 등 산업체 응용은 물론, 정보관리, 전자상거래, 업무절차관리 등의 상업적 응용과 환자상태 감시, 진료 시스템 등의 의료분야에서도 에이전트가 활용될 수 있다.

에이전트 시스템에 대한 연구의 목적은 사용자의 다양한 요구 사항을 만족시키기 위해 사용자의 요구 사항을 분석하여 적응력 있는 시스템을 구축하려는 데에 있다. 따라서, 에이전트 기술은 단순히 사용자의 요구 사항을 분석하는 것 뿐만 아니라, 사용자의 기호를 학습하여 사용자에게 보다 정확한 정보를 제공하기 위한 지능 에이전트로 발전하고 있다[3].

이러한 발전과 함께 최근에 와서 관심과 이용이 증대되고 있는 에이전트에 대한 연구 현황, 기술 개발 현황을 분석함으로써 향후의 연구 계획인 원자력 자료 정보화에의 활용을 위한 기초 단계로서 에이전트 분야의 연구 방향을 정립하는데 기여하고자 연구를 수행하게 되었다.

본 논문의 내용은 다음과 같다. 1 장에서는 연구의 필요성에 대해, 2 장에서는 에이전트의 기본 개념, 구성 요소, 언어 및 분류와 특성에 대해, 3 장에서는 에이전트의 발전에 대해, 4 장에서는 에이전트의 응용에 대해 기술하였고, 마지막으로 5 장에서는 결론을 기술하였다.

2. 에이전트의 개념

2.1 에이전트의 기본 개념

에이전트란 사전적 의미로는 '대행자', '대리인'이라는 뜻을 가지며, 컴퓨터 분야에서는 사용자를 대신해서 사용자가 원하는 작업을 자동적으로 해결하여 주는 소프트웨어라고 할 수 있다[4]. 에이전트의 개념에는 소프트웨어적인 것 뿐만 아니라 로봇(Robot)과 같은 하드웨어적인 것도 포함된다.

하지만 에이전트라는 용어는 연구 분야에 따라 다르게 정의되기도 한다. 인공지능(Artificial Intelligence) 분야에서는 에이전트를 분산 환경에서 수행하는 지적인 특성을 가진 프로그램이라고 하는 반면, 휴먼 인터페이스(Human Interface) 분야에서는 사람의 생각을 인지해 사람이 요구하는 사항을 처리하는 프로그램이라고 정의하기도 한다[5].

에이전트 분야는 80년대 말 인공지능과 분리되어 독립적인 연구 주제로 대두되면서 더욱 활발히 연구되고 있다. 기존의 인공지능은 하나의 독립된 에이전트로서 주로 인지적 측면에서 에이전트의 지식 표현, 추론 등의 연구 개발이 이루어졌고 전문가 시스템(Expert System)이나 지식 기반 시스템 (Knowledge-Based System) 등의 응용 시스템이 개발되었다. 하지만 점점 더 다양해지고 복잡해지는 요구를 해결하기에는 단독 에이전트의 한계가 있었고, 그것을 해결하기 위해 분산 협력 처리의 개념이 접목되기 시작했으며 이는 에이전트와 밀접한 관계가 있다.

2.1.1 특징

에이전트 시스템은 동적이고 능동적이며 유연한 시스템으로 이전의 소프트웨어와 비교해 볼 때 문제 해결의 질적 향상과 성능 향상을 가져왔다. 많은 연구들이 에이전트 시스템이 무엇이냐는 질문에 대한 직접적인 대답보다는 에이전트 시스템이 가져야 하는 주요 특징들을 열거하여 에이전트 시스템을 간접적으로 규정하고 있다. 에이전트의 여러 가지 개념을 중심이 되는 세가지로 요약해 보면 자율성, 협

력성, 적응성이라고 할 수 있으며, 그 주요 특징들은 다음과 같다[1,6].

- 자율성(Autonomy) : 사용자 또는 다른 에이전트의 지시 없이도 스스로 행동하는 성질, 스스로 판단하여 목적을 달성하기 위해 일련의 작업을 수행할 수 있는 것을 말하며, 이 특징은 에이전트를 일반 프로그램과 구별하여 특징 지워 주는 가장 두드러진 성질이다. 사용자로 하여금 상위 수준의 목적에 집중하게 하고 그 목적을 달성하기 위한 하위 세부 절차 등은 에이전트 시스템이 처리할 수 있도록 한다. 이 특징은 에이전트가 자율적이고 목적 지향적인 프로세스라는 것을 나타내는데, 자율성에는 환경의 변화에 수동적으로 대처하는 반응성(Reactivity) 외에도 자신의 목적을 가지고 능동적으로 작업 수행을 추구하는 목적 지향 행동(Goal-Oriented Behavior, Proactive Behavior)이 포함되기도 한다. 이 자율성을 위해 지능(Intelligence) 이라는 특징이 추가되는데, 이는 지식 베이스를 가지고 추론 및 계획을 할 수 있게 할 뿐만 아니라 작업을 수행하면서 새로운 사실에 대해 알아낸 것을 다음 작업에 적용할 수 있게 하는 학습 기능 또한 추가될 수 있다.
- 협력성(Cooperation) : 표준 언어나 프로토콜(Protocol)을 이용하여 다른 에이전트와 공통된 목적을 달성하려는 성질을 말한다. 협력성은 에이전트가 작업의 수행을 위해 다른 에이전트나 사용자와 협력한다는 것을 나타내는데, 표준 언어나 프로토콜을 이용하여 다른 에이전트와 공통된 목적을 달성하는 성질이다.
- 적응성(Adaptability) : 학습할 수 있는 능력을 말하며, 에이전트가 어떤 환경 안에 위치하고 그 환경의 특성을 인지하여, 환경 변화에 반응하고, 환경이나 사용자에게 맞도록 적응하여 수행 효율을 높이는 성질을 말한다.
- 사회성(Social Ability) : 에이전트의 통신 언어를 이용하여 에이전트들 사이의 통신을 통해 정보교환 등으로 서로의 문제 해결을 위해 도움을 주고 받을 수 있다.
- 반응성(Reactivity) : 하나의 에이전트 시스템이 그것이 놓여 있는 환경의 변화를 감지하고 이에 반응하는 능력을 말한다. 즉, 물리적인 실세계, 그

래픽 인터페이스(Graphic Interface)를 통해 접속하는 사용자, 다수의 에이전트 시스템들, 인터넷, 또는 이 모든 것들의 조합들에 대해 변화를 감지하고 이에 따라 적절한 반응을 취하는 것을 말한다.

- 이동성(Mobility) : 사용자가 요구한 작업을 현재의 시스템에 국한되어 수행하지 않고, 에이전트라는 객체 자체가 그 작업을 수행할 수 있는 다른 시스템으로 이동하여 수행하는 성질을 말한다.

2.1.2 지식기반 시스템

인공지능의 논제인 지식의 표현, 추론, 학습 등에 대하여 살펴 보고, 지능 에이전트를 구성하기 위해 어떻게 조합될 수 있는가를 살펴보도록 한다.

(1) 지식의 표현

인간은 세상으로부터 보고, 듣고, 만지고, 느끼고, 냄새를 맡는 등 경험을 통해 지식을 습득한다. 이러한 지식을 컴퓨터에 어떻게 저장할 것인가 하는 문제가 지식의 표현인데, 이것을 간단히 말하면 기호이다. 컴퓨터에 지식을 저장하는 일반적인 접근 방법은 다음과 같은 3가지로 거론된다[7].

첫째, 순차적(Procedural) 표현이다. 상수나 변수 등을 사용하여 사실을 코드화하는 것 뿐만 아니라, 그러한 사실들을 사용하고 관리하는 연산의 순서를 정의하는 것이다.

둘째, 관계형(Relational) 표현이다. 지식은 어떤 항목에 대한 정보의 집합이며, 특유한 속성과 값을 정의하는 필드(Field) 또는 열(Column)의 집합으로 표시된다.

셋째, 계층적(Hierarchical) 표현이다. 지식의 또 다른 형태로 상속되는 지식이 있는데, 이는 관계를 중심으로 하여 객체(Object)의 종류(Kind) 또는 클래스(Class) 사이의 속성을 공유하는 것이다. 객체 상속은 지식의 표현을 간결하게 해주고, 추론을 가능하게 해준다.

실세계에 대한 지식을 표현하는 것들의 집합을 지식베이스라고 한다[8]. 대량의 지식베이스상에서 추론을 하는 프로그램을 지식기반 시스템이라고 한다. 에이전트에 필요한 지식의 양, 즉 지식베이스의 크기와 추론 알고리즘의 정교함은 에이전트가 가지는 자율성과 이동성의 정도에 영향을 주는 중요한 요소이다.

(2) 추론

If-then 규칙은 지식 표현의 가장 좋은 표현이며, 인간이 만들기 쉽고 이해하기 쉽다. 이러한 규칙은 추론 시스템에 의해 관리된다. 추론 시스템은 새로운 사실을 작업 영역에 단순히 추가하기만 하는 단조형(Monotonic)과 새로운 증거가 제공될 때 사실을 철회할 수 있는 비단조형(Nonmonotonic)이 있다.

추론의 방법에는 전방 추론과 후방 추론의 두가지가 있다. 전방 추론은 규칙을 사용하는 데이터 중심의 추론이다. 작업영역에 있는 사실들을 가진 초기 집합으로 시작되며, 선택(Match) 단계에서는 규칙 중 대립되는 집합들은 선택하고, 대립 분석 단계에서는 하나의 규칙을 선택한다. 행위(Act) 단계에서는 규칙을 제거하고 작업 영역에 새로운 사실을 추가한다. 이러한 과정은 대립 집합이 없을 때까지 반복된다. 후방 추론은 규칙을 사용하는 목표 중심의 추론이다. 목표로부터 출발하며, 규칙을 통해 알고리즘은 결과-절로부터 선행-절까지를 연결하여 규칙이 참이 되는 것을 증명하려고 한다.

(3) 학습

지능의 중심이 되는 요소는 경험으로부터 학습하고 배우는 능력이다. 대부분의 복잡한 지식의 표현과 추론 알고리즘은 지능 에이전트가 접하게 되는 이전의 모든 상황에 대해 알 수 있는 방법은 없다. 그러므로 환경에 변화하여 적응하거나 경험을 통해 작업이 좋아지도록 하는 능력은 어떠한 소프트웨어 시스템에 대해서도 중요하게 구별지어지는 것이다. 학습은 어떤 반복적인 일을 수행하기 위해 복잡한 환경에서 사용되거나 에이전트가 불확실한 상황에 적응해야만 할 때 아주 유용하게 사용된다.

지능 에이전트에 대한 정의는 다른 과학의 연구 방향에 대한 효과에 종속적이라고도 하고, 실제 응용에서 연구되는 사항들을 반영하는 것이라고도 한다[9]. 또한 에이전트가 어떤 작업을 수행하기 위해서는 많은 지능을 필요로 하기 때문에 지능 에이전트라고 하기도 한다. 지능 에이전트는 다른 에이전트로 메시지를 보내거나 응용 프로그램 인터페이스 또는 시스템 서비스를 호출하여 동작을 취하도록 하는 작동체이다. 일반 프로그램들은 지정한 작업을 수행하고 그 결과만을 사용자에게 제공하기 때문에 지능 에이전트가 아닌 것은 일반 소프트웨어 프로그램이라고 간주

할 수 있다.

지능 에이전트를 구축하는 전통적인 접근 방법은 그들을 일종의 지식 기반 시스템(Knowledge-Based System)의 한 특수 유형으로 보는 방법이며, 이러한 패러다임을 기호주의 인공지능(Symbolic AI)이라고 한다. 기호주의 인공지능 패러다임은 물리적 기호 체계 가설을 기초로 하며, 이 가설은 일련의 기호 구조들을 처리할 수 있는 하나의 물리적 기호 체계가 지능적 행위를 나타낼 수 있는 충분한 수단이 된다고 본다[10].

지식의 표현, 추론 및 학습은 인공지능의 기술이다. 지능 에이전트의 기능과 요구 사항은 그러한 인공지능의 기술이 사용되어야만 한다. 에이전트 시스템에 지적 능력을 부여하는 방법 중의 하나로 인공지능에서의 전통적인 접근 방법 중 지식 기반 접근 방식을 채택하여 에이전트 시스템에 지적 능력을 부여한다. 이러한 방법을 통해 구현된 지적 능력을 갖는 에이전트 시스템을 지능 에이전트 시스템이라고 한다.

계획은 지능 에이전트의 기본적인 요구 사항이다. 계획은 부프로그램 또는 계획의 일부 사이에서 상호작용이 있기 때문에 어렵다. 계획 알고리즘은 3 가지 형태 즉, 선형, 비선형, 계층의 3 가지 형태로 연구된다. 계획은 문제 해결의 몇가지 측면을 포함하고 있다. 우선 큰 문제를 작은 것으로 분해하여 부 프로그램을 쉽게 해결하도록 한다. 그리고, 이전의 단계에서 수행한 작업을 하지 않도록 부 프로그램의 해결 순서에 대해 강제사항을 고려한다. 마지막으로 해결을 위해 진보한 것처럼 상태를 계속 추적하도록 한다.

지능 에이전트는 에이전트 중에서 학습 능력이나 추론 능력, 계획 능력과 같은 지능적인 특성을 갖는 에이전트를 말한다. 지능 에이전트가 갖는 특성을 살펴보면 다음과 같다.

- 학습 기능(Learning) : 사용자의 프로그램 사용 경향을 파악하여 같은 작업을 반복 수행하지 않도록 지원한다.
- 추론 기능(Reasoning) : 사용자가 원하는 작업에 대해 기존의 처리 방법이나 다음 시스템에 있는 에이전트의 경험과 지식을 바탕으로 하여 작업의 처리 방법을 파악하고 그에 따라 문제를 해결한다.

- 계획 기능(Planning) : 여러 에이전트가 협력하여 하나의 작업을 처리하기 전에 에이전트간의 통신과 에이전트가 수행하여야 할 작업을 어떤 방식으로 진행할 것인가에 대해 계획하고 그 계획에 따라 통신하고 작업을 수행한다.

학습 기능은 각 개인마다 브라우저를 사용하는 것을 감시하면서 사용자가 선호하는 문서의 종류를 분류한 후 선호도를 학습한 결과를 개인의 프로파일에 기록하도록 한다. 이에 대한 연구는 인공지능과 정보검색 분야에서 이미 제시된 많은 이론들을 사용하고 있다. 사용자가 웹에 연결되어 있는 동안 에이전트로부터 현재의 웹 문서에 대해 정보 변화 감시를 추천받게 되며 사용자가 추천된 URL 을 받아들이나 아니냐에 따라 피드백(Feedback)을 보내게 된다. 이러한 직접적인 피드백 외에 암시적인 피드백이 있는데, 사용자가 정보 변화 감시를 위해 URL 을 등록시키는 행위 자체를 학습하는데 이용하기도 한다[2].

2.2 에이전트의 구성 요소

에이전트는 일반적으로 처리 모듈, 지식 모듈, 통신 모듈로 구성되어 있으며, 그 구조는 그림 2-1 과 같다.

(1) 처리 모듈

처리 모듈은 문제를 해결하기 위해 구체적인 작업을 정의한 기구들이 여기에 해당한다. 단위 작업의 문제에 대한 해결, 협력을 위한 제어, 동작의 제어를 담당한다.

(2) 지식 모듈

문제 해결에 필요한 지식을 저장하고 있으며, 지식은 제어를 위한 지식, 협력을 위한 지식 등 여러 종류가 있을 수가 있다. 또한 지식을 관리하기 위한 기구가 포함되며, 추론이나 학습 등과 연결될 수 있다.

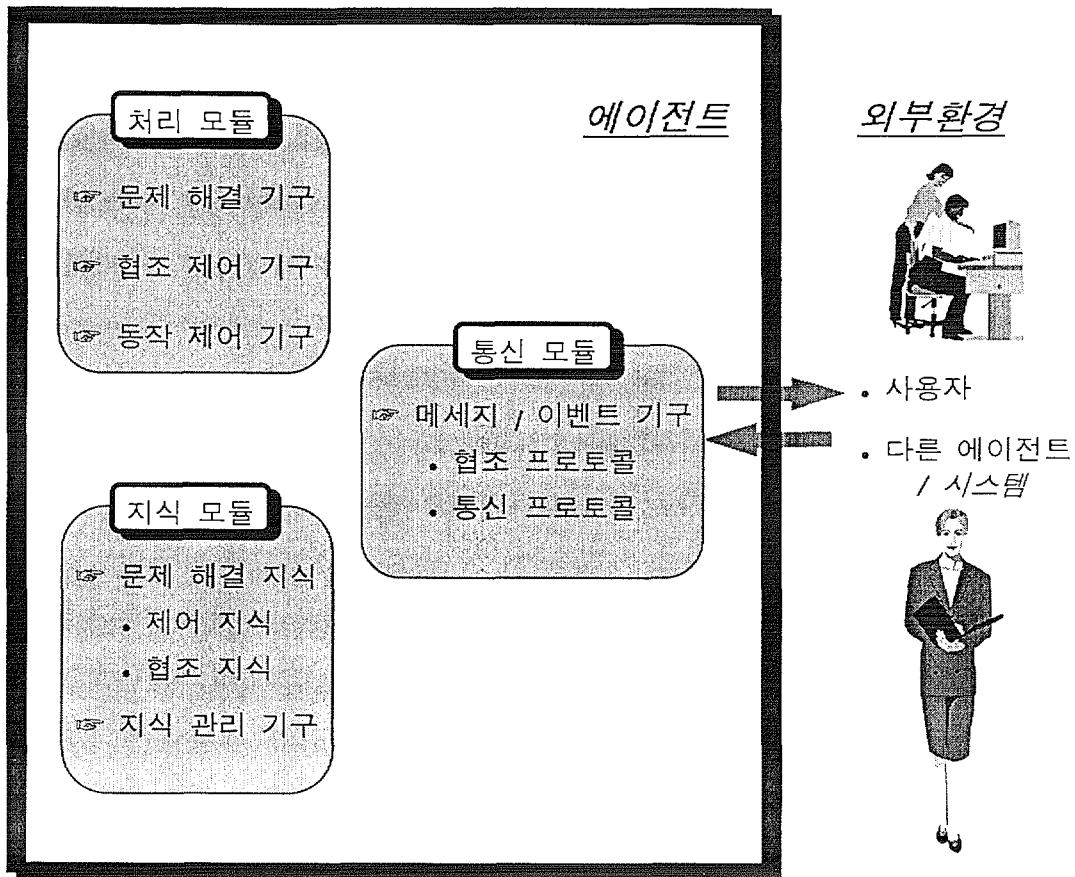


그림 2-1. 에이전트의 구조

(3) 통신 모듈

사용자 또는 다른 에이전트간의 메시지와 이벤트 등에 대한 통신을 위하여 제어를 담당한다. 사용자와 에이전트가 정보를 서로 주고 받기 위해서는 통신 규약이 필요하다. 또한 분산된 네트워크 환경에서는 다양한 응용 에이전트들이 존재하며, 이 에이전트들은 서로 다른 플랫폼(Platform), 서로 다른 프로그래밍 언어(Programming Language), 서로 다른 문제 해결 방법 등을 통해 구현된다. 이로 인해 에이전트끼리의 통신을 위해 내부적인 통신 규약이 필요하다. 이에 대한 연구로는 미 국방부의 'Knowledge Sharing Effort' Project 의 일환으로 수행되는 ACL(Agent Communication Language)이나 SRI(Stanford Research Institute)의 ICL(Inter-Agent Communication Language)등이 있다.

에이전트 기술의 표준화를 목적으로 하는 국제조직인 FIPA(Foundation for Intelligent Physical Agents)가 1996년 9월 정식으로 설립되어 활동을 시작하였다. FIPA는 기본적으로는 에이전트의 내부 구조와 구현 방법은 표준화의 대상으로 하지 않는다. 에이전트의 관리 기술, 에이전트 간의 통신 언어, 에이전트와 기존 소프트웨어와의 통합 등 이 세 가지 요소의 기술을 대상으로 하여 표준화를 하고 있다.

2.3 에이전트 언어

에이전트들간의 인터페이스가 일정하지 않기 때문에, 이를 처리하고 에이전트들간의 협력 작업을 수행하기 위하여 에이전트들간의 교류를 위한 언어가 필요하다.

(1) KQML

에이전트 사회에서의 대표적인 에이전트 통신 언어는 KQML(Knowledge Query and Manipulation language)로, ARPA(Advanced Research Projects Agency) KSE(Knowledge Sharing Effort)의 한 연구 그룹인 External Interfaces Working Group에 의해 제시되었다[11]. 지식 베이스들에 대한 응용 시스템을 설계할 때, 개발자들은 KQML을 사용함으로써 다음과 같은 점들로부터 자유로울 수 있다.

- 지식 베이스(기능성)의 위치와 이를 필요로 하는 곳의 위치
- 지식 표현 방법, 요구 방법 및 전달 방법
- 직접적인 지식 전달 메카니즘(Mechanism)

이로 인해 기존 지식 베이스의 응용 시스템이나 새로운 지식 베이스는 기존의 시스템에 쉽게 연결될 수 있으며, 그 기능성들을 이용할 수 있다.

KQML은 정보 교환에 관련된 통신 상황에 중점을 둔 언어이며, 통신의 내용이 되는 정보의 형태나 의미에 독립적이다. 또한 기존의 소프트웨어들에 랩핑(wrapping)되어 쉽게 에이전트화 될 수 있다. 특히, 단순한 구문 위주의 프로토콜 이상이다. 이는 KQML이 자신을 이용할 통신 상황에 대한 이해와 이에 기반한 에이전트 시스템의 구조에 대한 이해도 포함하고 있음을 의미한다.

KQML은 내용 계층(Content Layer), 메시지 계층(Message Layer), 통신 계층(Communication Layer)으로 구성되어 있다. 내용 계층에서는 내용 메시지를 자신의 언어로 구성한 후 KQML 메시지 안에 싸서 메시지 계층으로 보내게 된다. 메시지

계층은 내용 계층에 기술된 내용에 부가적인 정보를 포장하여 메시지를 생성해 내는 계층이다. 통신 계층은 KQML 에서 가장 외부에 있는 계층으로 내용 계층, 메시지 계층을 거친 메시지를 통신에 관련된 부가 정보인 송신자, 수신자, 메시지 ID 등을 추가하여 패키지(Package) 형태로 만든다.

(2) ICL

에이전트들간의 교류를 위한 언어로 ICL(Inter-Agent Communication Language)이 있다. ICL 은 에이전트들간의 상호 교류를 위해 약속된 프로토콜과 메시지 형태이다[12]. 그림 2-2 에서와 같이 ICL 은 내용 계층(Content Layer), 프로토콜 계층(Protocol Layer), 래퍼 계층(Wrapper Layer)으로 되어 있다. 내용 계층은 에이전트간에 교류되는 메시지의 내용을 나타낸다. 프로토콜 계층은 에이전트들간의 약속된 규칙을 나타낸다. 래퍼 계층은 에이전트가 그 내용을 보고 해석될 수 있는 메시지인가 판단하며, 어느 에이전트가 보낸 것인가를 알아내도록 한다.

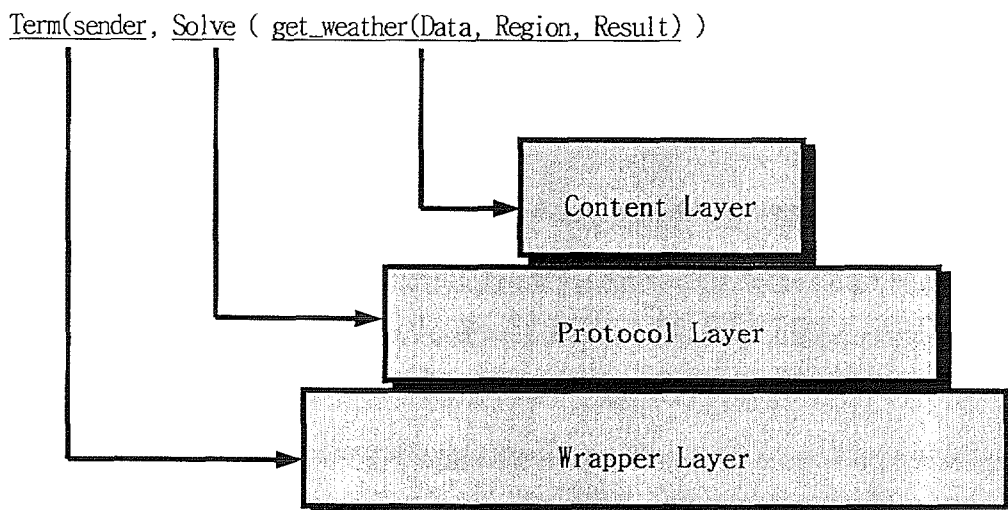


그림 2-2. ICL 의 구조

2.4 에이전트의 분류와 특성

에이전트는 관점에 따라 여러 가지로 분류될 수 있다. 논리 구성적 관점에 따

라 숙고형 에이전트(Deliberative Agent), 반응형 에이전트(Reactive Agent), 혼합형 또는 복합형 에이전트(Hybrid Agent)로 분류되기도 하고, 에이전트와 그것을 둘러싸고 있는 환경과의 관계를 의미하는 반응성의 기준에 따라 숙고형 에이전트(Deliberative Agent)와 반응형 에이전트(Reactive Agent)로 분류될 수 있다. 네트워크를 통한 이동성 유무의 관점에 따라 이동형 에이전트(Mobile Agent)와 정적 에이전트(Static Agent)로 분류된다. 다른 에이전트들과의 관계인 자동성(Autonomy), 협력(Cooperation), 학습(Learning) 능력의 유무에 따라 협력 에이전트(Agent), 인터페이스 에이전트(Interface Agent), 협력 학습형 에이전트(Collaborative Learning Agent), 보조에이전트로 분류될 수 있다. 한 에이전트가 자신의 행동을 계획할 때 다른 에이전트들의 존재를 가정하고 그들과의 상호작용을 고려하는가 하는 사회성의 기준에 따라 단일 에이전트(Single Agent)와 멀티 에이전트(Multi Agent)로 분류하기도 한다. 이 내용을 표로 정리하면 표 3-1 과 같다.

표 3-1. 관점에 따른 에이전트의 분류

관점	에이전트의 분류
논리 구성 및 반응성	숙고형 에이전트, 반응형 에이전트, 복합형 에이전트
이동성	이동 에이전트, 정적 에이전트
다른 에이전트와의 관계	협력 에이전트, 인터페이스 에이전트, 협력 학습형 에이전트, 보조 에이전트
사회성	단일 에이전트, 멀티 에이전트

에이전트에 대한 몇가지의 특성을 살펴 보고자 한다.

2.4.1 논리 구성 및 반응성에 따른 분류

(1) 숙고형 에이전트

기호주의 인공지능 패러다임을 기초로 한 에이전트들을 일반적으로 숙고형 에이전트(Deliberative Agent)라고 한다. 숙고형 에이전트는 기호를 이용하여 명시적으로 표현한 월드 모델(World Model)을 가지며, 패턴 매치(Pattern Match)와 기호 처리에 바탕을 둔 논리적 추론(Logical Reasoning)을 통해 자신이 취해야 할 행동

(Action)을 스스로 결정해 간다. 이러한 숙고형 에이전트들은 이러한 이론적 배경에도 불구하고 복잡하고 변화가 심한 현실적인 문제에 신속히 대응하는데는 많은 어려움을 갖는 것으로 알려져 있다. 일부 연구자들은 숙고형 에이전트가 갖는 문제점을 제기하며 그 대안으로서 환경 변화에 민감하게 대응하는 반응형 에이전트를 제안하였다.

지능 에이전트는 숙고형 에이전트의 시스템의 한 종류로 지식의 양과 질이 지적 능력의 핵심이라고 할 수 있는 시스템이다[6]. 이 에이전트는 핵심 구성 요소라고 할 수 있는 지식 베이스를 유지하고 있으며 약간의 초기 지식을 갖고 있다. 호출될 때마다 두 가지의 일을 수행한다. 우선 인지한 바를 지식 베이스에게 알려주고, 다음으로 어떤 행위를 하여야 하는지 지식 베이스에게 물어본다. 질문에 대한 답을 구하기 위해 어떤 행동이 더 바람직 한지를 증명하기 위한 추론(Reasoning)이 전개된다. 지능 에이전트는 지식 베이스를 바탕으로 새로운 사실을 추가하거나 유도하기 위해 추론 메커니즘이 필요하며, 이것이 지능 에이전트의 중요한 구성 요소이다.

2.4.2 이동성에 따른 분류

(1) 이동 에이전트

이동 에이전트는 네트워크 에이전트 또는 순회(Itinerant) 에이전트라고도 말하며, 프로그램 자체가 네트워크를 통해 이동해 가며 수행되는데, 이형의 네트워크에서 한 시스템에서 다른 시스템으로 이동할 수 있는 프로그램이다. 즉, 에이전트가 수행을 시작한 하나의 시스템에 묶여 있지 않는 에이전트인 것이다[13]. 고정 에이전트와 비교해 볼 때 서로 다른 시간에 서로 다른 시스템에서 수행될 수 있는 경우인데, 기존의 메시지 전달 방식과는 달리 직접 작업을 수행할 수 있는 소프트웨어 객체를 전달하는 방식을 이용한 것이다.

이와 유사한 예로 자바 애플릿(Java Applet)을 들 수 있는데 애플릿은 웹 브라우저가 요구할 때 서버가 코드를 보내주어 브라우저 내에 있는 JVM(Java Virtual Machine)이 코드를 수행하는 반면, 이동 에이전트는 자신의 판단에 의해 이동하는 것이 다르다.

이동 에이전트는 이질적인 네트워크에서 자신의 제어에 의해 시스템 사이를 이동하고, 각 시스템의 다른 에이전트와 상호 동작하거나 자원을 이용하여 맡겨진 임무를 수행하고, 수행이 끝난 경우에는 처음 시작한 곳으로 되돌아 온다. 에이전트가 인터넷을 돌아다니며 수행되기 위해서는 코드 뿐만이 아니라 상태도 이동되어야 한다. 코드는 이동될 모든 시스템에 똑같은 형식으로 동작되어야 하며 상태는 이동 코드의 수행 위치, 중간 경과 등을 계속 유지하여야 하므로 지속성(Persistence)이 있어야 한다.

이동 에이전트를 언어 측면에서 분류하면 자바에 기반한 이동 에이전트 시스템과 자바에 기반하지 않은 이동 에이전트 시스템으로 분류할 수 있다.

이동 에이전트는 이벤트가 발생했는지 등의 상태를 독립적이고 자의적으로 볼 수 있고(Watch), 인터넷에서 원하는 정보가 어디에 있는지 이동하면서 찾을 수 있고(Search), 여러 가지 서비스를 조화롭게 처리할 수 있다.

이동 에이전트는 에이전트 구현 언어가 스크립트(Script) 언어로 작성되고 인터프리터(Interpreter)로 수행된다는 것이 특징이다. 또한 자신을 다른 컴퓨터로 이동시키는 명령이 있으므로 그 명령을 만나면 다른 서버로 이동할 수 있다. 그 외에도 자신의 판단에 따라 이동하는 능력 뿐 아니라, 동일한 에이전트를 복제해 다른 시스템으로 보내고 그들이 가져온 결과를 모아 복합적인 결과를 만들기도 한다.

이동 에이전트는 이동 컴퓨팅 분야에 효과적인 기술이다. 이동 컴퓨팅이란 사용자가 노트북이나 PDA(Personal Digital Assistant)를 들고 다니면서 사용하는 환경으로, 이런 컴퓨터는 이동성 때문에 다른 컴퓨터와의 통신이 취약하다. 따라서, 클라이언트/서버(Client/Server) 모델은 컴퓨터가 계속 연결되어 있어야 하므로 이동 컴퓨팅과는 거리가 있다.

이동 에이전트의 응용은 전자 상거래(Electronic Commerce), 그룹 공동 작업(Group Collaboration), 이벤트 모니터링(Event Monitoring), 작업 흐름 자동화(Workflow Automation), 정보 검색(Information Retrieval), 망 관리(Network Management) 및 이동 컴퓨팅(Mobile Computing) 등이다. 이동 에이전트의 응용 분야는 새로운 응용 분야를 개척하는 것이 아니고 기존의 분산 응용을 이동 에이전트를 통하여 효율적이고 편리하게 제공할 수 있다는 것이다. 또 에이전트가 이동하기

때문에 보다 확장성 있는 응용을 작성할 수 있다.

하지만 이동 에이전트는 사용자 요구에 따라 에이전트가 컴퓨터 사이를 이동하므로, 사용자는 서비스를 제공받기 위해 지속적으로 연결 상태를 유지할 필요가 없다. 단지 나중에 두 컴퓨터를 연결해 처리된 결과를 가져오기만 하면 된다.

반면에 이동 에이전트가 갖는 가장 큰 문제점은 보안이다. 자신의 컴퓨터에서 수행되는 다른 사람의 이동 에이전트는 바이러스처럼 나쁜 영향을 끼칠 수 있다. 또 다른 문제점으로는 수많은 서버 중에서 자신이 원하는 서비스를 제공하는 서버를 어떻게 효율적으로 찾는가와 이동 에이전트 서버가 있는 시스템으로만 이동이 가능하다는 것이다. 이동 에이전트는 스크립트로 구성되기 때문에 자신을 수행시켜 줄 인터프리터가 없으면 무용지물이 된다. 즉, 자바 애플릿이 자바 인터프리터가 없는 웹 브라우저에서는 아무 역할도 할 수 없는 것과 마찬가지로 이동 에이전트의 성공 열쇠는 이동 에이전트 서버가 얼마나 많이 설치될 수 있는가에 달려 있다.

2.4.3 다른 에이전트와의 관계에 따른 분류

(1) 인터페이스 에이전트

아이콘이나 메뉴의 사용 등 그래픽 사용자 인터페이스(GUI)의 개발로 많은 편리함을 가져왔지만 그것만으로는 사용자의 불편함을 완전히 해소하지는 못한다. 사용자는 자신이 원하는 작업에 대한 내용만 기술하고 나머지는 컴퓨터가 알아서 해 주기를 바란다. 이 역할을 인터페이스 에이전트가 하는 것이다[4].

사용자가 컴퓨터를 편리하게 쓸 수 있도록 지원하는 것을 사용자 인터페이스 에이전트라고 한다. 예를 들어, 사용자가 키보드(Keyboard)나 마우스(Mouse)의 조작 없이 대화를 통하여 사용자가 원하는 작업을 처리해 줄 수 있다면 컴퓨터에 익숙하지 않은 사용자에게도 현재보다는 훨씬 편리해질 것이다. 이것이 바로 인터페이스 에이전트의 궁극적인 목적인 사용자 인터페이스를 고도화하는 것이라고 할 수 있다.

사용자 인터페이스에 관한 에이전트의 역할은 크게 보아 3 가지로 말할 수 있다. 첫째, 사용하기 쉬운 인터페이스를 제공하여 반복적인 작업에 대한 편리성을 제

공한다. 둘째, 학습을 통해 사용자의 습성을 파악하여 적용시키는 학습 인터페이스 에이전트이며 주로 데스크탑 응용에서 발생하는 사용자의 처리 유형을 반영하게 된다. 셋째, 멀티미디어 인터페이스를 제공하는 인터페이스이며, 음성이나 필기체 입력 등을 가능하게 하여 자연스러운 인터페이스를 지원한다.

인터페이스 에이전트는 사용자를 관찰하면서 개별적인 응용 프로그램을 처리하는 방법에 대해 그때의 상황과 처리 방법으로 나누어 학습해 두었다가 똑같은 상황이나 비슷한 상황이 발생하면 사람 대신 그 일을 처리해 주는 기계 학습을 채용한 프로그램이다. 이러한 기계 학습 방법에는 몇가지 문제점이 있다. 에이전트를 학습시키는데 많은 시간과 사용자의 반복적인 일의 처리가 있어야 하며 에이전트를 학습시키는 동안에는 인터페이스 에이전트의 도움 없이 사용자가 일을 직접 처리하여야 한다. 또 에이전트가 학습하지 못한 상황이 발생한 경우에 에이전트는 어떠한 대처도 할 수 없다는 것이며 이런 경우에 처음부터 사용자가 다시 훈련을 시켜야 한다.

이러한 문제점을 해결하기 위하여 협력 인터페이스 에이전트에 대해 연구하기 시작했다. 이는 협력 에이전트 시스템의 한 분야로, 한 에이전트가 자신의 경험이 부족하여 처리할 수 없는 일이 주어진 경우에 다른 에이전트에게 어떤 처리를 하여야 하는지를 물어보아 처리하게 된다. 이와 같이 협력 인터페이스 에이전트 개념을 도입하면 독자적으로 처리할 수 없는 일들이 에이전트간의 협력을 통해서 처리될 수 있는 장점이 있다.

(2) 보조 에이전트

보조 에이전트는 사용자의 작업을 돕는 역할을 한다. 어떤 문제를 해결하는 블랙박스과 같은 것으로 사용자를 대신해 일을 처리해 주는 프로세스라고 말하기도 한다. 이와 같은 견해의 예로는 네트워크에서 자신이 원하는 자료를 찾아 주는 Archie 나 뉴스 그룹 프로그램인 Free Agent 를 에이전트라고 부르는 경우를 들 수 있다.

기존 프로그램은 단지 사용자 지정한 작업을 수행하는 프로그램인 반면 보조 에이전트는 능동적인 특성을 갖고 사용자의 작업을 대행한다. 따라서 몇몇 사람은 보조 에이전트의 능동적인 특성을 반영하기 위해 로봇이라 부르기도 한다. 로봇에

대한 사용 예로는 사용자의 지식을 기반으로 전자우편을 정리해주고 자동으로 답장 해 주는 MailBot 이나 FTP, Archie, Netfind 등과 같은 인터넷 관련 작업을 도와주는 소프트봇(Softbot), 그리고 웹에 있는 정보를 찾아주는 웹 에이전트 등을 들 수 있다.

(3) 협력 에이전트

에이전트들은 사용자가 정해 주거나 경험에 의해 알아낸 목적을 단독으로 달성하지 못하는 경우가 많다. 단독의 에이전트가 가질 수 있는 경험은 한정되어 있지만 사용자의 요구는 다양할 뿐만 아니라 예측을 할 수 없기 때문이다. 이러한 것을 해결하기 위해 에이전트간의 정보 교류가 필요하게 되었다. 이러한 방식을 반영한 것이 협력 에이전트(Collaborative Agent)이다[3].

협력 에이전트 시스템은 에이전트의 특징을 바탕으로 다른 에이전트와 함께 적용되고 있으며, 특히 정보 관련 에이전트쪽에서 많이 이용되고 있는데 이것은 협력 에이전트가 에이전트간 지식 수준의 정보 공유를 하기 때문으로 보여진다.

하지만 협력 에이전트는 실제로 구현하기가 쉽지 않다. 협력 에이전트 시스템을 평가할 수 있는 기준이나 평가 방법이 존재하지 않아 시스템 개발 후 시스템의 성능을 분석할 수 없다. 협력 에이전트가 어떻게 학습할 것인지를 결정해야 하고, 에이전트간의 정보를 공유하기 위해 서로간에 이해할 수 있는 표현 방법을 사용하는 것이 필요하고, 같은 공동체에 있는 에이전트의 유지와 접촉할 에이전트를 선택하는 방법에 대한 연구도 더 필요하다. 이것은 에이전트가 속할 공동체를 잘못 선택하면 사용자가 관심도 없는 정보를 제공할 수도 있고, 또 접촉할 에이전트의 선택에 따라 사용자에게 제공할 정보의 수준이나 사용자의 만족도가 달라지게 된다.

2.4.4 사회성에 따른 분류

(1) 멀티 에이전트

멀티 에이전트는 분산 환경에서 상호 협력을 통해 작업을 수행하는 컴퓨터 프로그램을 말한다. 일반적으로 하나의 에이전트는 하나의 작업을 수행한다. 그래서 어떤 복잡한 문제를 해결하기 위해서는 여러 에이전트들이 서로 협력하여 작업을 할 경우가 필요하다.

멀티 에이전트는 여러 응용 에이전트와 이를 응용 에이전트 사이의 중재자 역할을 하는 조정자(Coordinator) 또는 촉진자(Facilitator)로 구성된다. 조정자는 한 에이전트로부터 받은 메시지를 다른 에이전트에게 전달하며 각 에이전트의 제어를 담당한다. 물론 조정자 없이 각각의 응용 에이전트들 사이의 직접적인 메시지 전달도 가능하나, 여러 문제점으로 인해 조정자를 둔다[14].

조정자가 있는 멀티 에이전트 시스템에서 모든 에이전트의 통신 관련 메시지는 조정자를 통해서 다른 에이전트로 전달된다. 에이전트끼리 대화한다는 것은 정해진 언어 규약에 따라 메시지를 주고 받음을 의미한다.

에이전트는 다른 에이전트에게 서비스를 요청하기 위해 정해진 언어 규약에 따라 요구 사항을 메시지 형태로 바꾼 후 해당 에이전트에게 전달한다. 다른 에이전트로부터 서비스 요청을 받은 에이전트는 그 메시지를 분석해 내부에서 처리할 수 있는 형태로 변환해서 이를 처리한다. 에이전트는 그 결과를 다시 메시지 형태로 바꿔 이를 요청한 에이전트에게 전달한다.

멀티 에이전트의 가장 큰 장점은 독립적인 응용 프로그램의 집합으로는 해결할 수 없는 보다 복잡한 서비스를 다른 에이전트와의 협력을 통해 제공할 수 있다는 점이다. 이 외에 자신이 필요로 하는 에이전트를 시스템에 붙임으로써 새로운 서비스에 대한 시스템의 확장이 용이하다는 장점이 있다.

3. 에이전트의 발전

3.1 에이전트의 발전 배경

에이전트는 인공지능 분야의 사실상의 주된 연구과제였다고 볼 수 있으며, 인공지능의 최종 목표는 사람과 유사하게 생각하고 행동하는 지능을 가진 에이전트의 개발이라고 볼 수 있다. 하지만 에이전트 분야가 인공지능과 분리되어 독자적인 연구 주제로 대두된 것은 80년대 말부터이며 그 후 에이전트에 대한 독자적인 학술 대회가 개최되고 관련 제품이 출시되면서 더욱 활발한 연구가 진행되었다.

에이전트가 인공지능과 분리되어 독자 노선을 걷게 된 배경에는 여러 가지가 있을 수 있지만, 그 중 분산 협동 처리와 에이전트간의 통신 개념이 대두되면서 부터라고 할 수 있다. 인공지능 자체는 하나의 독립된 에이전트로 에이전트의 지식 표현과 추론 등에 대한 연구가 이루어졌고, 이를 바탕으로 전문가 시스템이나 지식 베이스 시스템 등의 응용 시스템이 개발되었다. 하지만 점점 다양해지고 복잡해지는 사용자의 요구를 해결하기에는 한계가 있었으며, 이의 해결 방법으로 다른 에이전트의 도움을 받아 처리하는 분산 협동 처리의 개념이 도입되어 분산인공지능이라는 새로운 연구 분야가 형성되었다. 이것은 다양한 분산인공지능 문제들을 에이전트라는 추상적 단위와 에이전트간의 상호작용을 토대로 하여 해결하기 위해 시도되었다.

에이전트라는 용어는 데이터베이스나 운영체제, 네트워크 등 기존의 컴퓨터 연구 분야에서 사용되기 시작했으며, 기본적으로는 컴퓨터 환경이나 네트워크 환경에 구애 받지 않고 동일한 기능을 제공하는 프로세스를 의미하였다. 최근에 와서는 인터넷과 웹의 급속한 진보와 함께 웹 환경하의 소프트웨어 에이전트가 출현하게 되었다. 이러한 에이전트는 사용자를 대신하여 주어진 일을 수행하는 대행자 또는 중개자의 역할을 한다.

3.2 에이전트의 연구 동향

인공지능 분야가 처음 소개된 이후로 에이전트에 대한 연구가 계속되어 왔다. 최근에 이르러 에이전트는 인공지능 분야는 물론, 데이터 통신(Data

Communication), 소프트웨어 공학(Software Engineering), 데이터베이스, 병행 시스템(Concurrent System), 로보틱스(Robotics), 사용자 인터페이스 설계(User Interface Design) 등 다양한 분야의 연구자들과 개발자들로부터 폭넓게 관심을 받는 연구 주제가 되었다. 소프트웨어 에이전트(Software Agent) 기술은 분산 자원의 공유와 관리, 또는 더 나은 인간-기계 인터페이스(Man-Machine Interface) 제공을 가능케 하는 기술로서 더욱 관심 받는 연구가 되었다.

에이전트 기술을 이용한 시스템의 구축이나 실험 또한 여러 응용 분야에서 수행되고 있다. 이 장에서는 에이전트의 구성 방식, 에이전트의 기능에 따른 에이전트의 연구 현황을 살펴 보기로 한다.

3.2.1 에이전트의 구성 방식에 관한 연구 동향

국내외 연구 방향은 에이전트의 구성 방식에 따라 크게 세가지로 나누어 볼 수 있다. 단일 에이전트(Single Agent), 하이브리드 에이전트(Hybrid Agent), 그리고 분산환경 및 멀티 에이전트(Distributed Environment and Multi Agent)에 관한 연구이다.

첫째, 단일 에이전트는 에이전트의 특성들 중 어느 하나에 초점을 둔 에이전트이다. 즉 인터페이스 에이전트는 사회성에, 이동 에이전트는 이동성에, 적응형 에이전트는 적응성에, 심사 숙고형 에이전트는 목표 지향성에, 반응형 에이전트는 반응성에 각각 초점을 맞춘 경우라고 할 수 있다.

둘째, 하이브리드 에이전트는 단일 에이전트들에 대해 장점은 최대화하고 약점은 최소화할 수 있도록 몇가지의 특성을 함께 보유한 에이전트이다. 예로 심사숙고형 에이전트로 하여금 장기적인 목표에 대한 계획 부분을 다루고 반응형 에이전트는 신속한 반응을 요구하는 것에 대해 빠른 반응 시간을 제공하도록 하여 하나의 에이전트로 구성한 경우이다.

셋째, 분산 환경 및 멀티 에이전트 아키텍처는 단일 에이전트 또는 하이브리드 에이전트가 협력하여 복잡한 문제를 해결할 수 있도록 한다[8]. 멀티 에이전트는 단일 에이전트가 사용자의 요구 사항에 대한 모든 작업을 수행하는데 있어서의 로드를 극복하기 위하여 단순한 작업을 수행하는 에이전트들을 통합하여 복잡한 작업을

수행하도록 하는 방법이며 여러 응용 분야에서 적용되고 있다.

하지만 하이브리드 에이전트에 대해 몇가지의 문제점이 지적될 수 있다. 우선 하이브리드 아키텍처가 기반하고 있는 이론이나 원칙이 아직도 불분명하다는 것인데, 이로 인해 대부분의 문제에 대응할 수 있도록 하이브리드 에이전트를 설계하는 과정에서 임시 방편의 방법이 사용되기 쉽다. 또한 하이브리드 에이전트가 특정 응용 분야에만 한정적으로 적용될 수 있도록 설계되는 경우가 많은데, 이것은 응용 분야만을 고려 대상으로 한다면 오히려 장점이 되기도 한다.

3.2.2 에이전트의 기능에 관한 연구 동향

에이전트의 기능에는 문서 변화 감지, 해석, 할당, 보조자 역할 및 계획 기능이 있다.

웹 에이전트는 인터넷상에서 보다 효율적인 검색을 수행할 수 있도록 도와주는 기능을 가지고 있다. 웹의 내용에 변경이 있을 때 사용자에게 자동적으로 통보하는 기능 자체에 대한 연구와 웹에 산재한 많은 정보들을 사용자 측면에서 효율적으로 접근하거나 관리하기 위한 연구들이 많이 있다. 그렇지만 여러 기능을 통합하여 제공하는 시스템은 드물고 하나의 기능을 달성하기 위한 연구들이 대부분이다[2]. 각 기능에 대한 연구를 살펴보고자 한다.

문서 변화의 감지 기능은 웹의 내용에 변경이 있을 때 사용자에게 알려 주는 것으로 개인 웹 로봇을 통해 이 기능을 구현하기도 한다. 웹 페이지의 내용 전체, 링크 부분 중 사용자로부터 입력 받은 키워드를 포함한 URL, 웹 페이지의 헤드라인 등에 대해 변경 유무를 판단하여 메일로 사용자에게 알리는 방법이 있다. 또, URL 을 입력 받거나 키워드를 입력 받아 내용이 달라졌는지 비교하고 달라진 경우 사용자에게 통보하여 달라진 내용을 확인하도록 하는 방법이 있다. 그리고, 사용자가 원하는 정보에 대해 변화가 있는지의 감시를 위해 웹 브라우저 상단에 메뉴가 항상 띄워져 있어서 URL 을 등록하도록 하는 방법이 있는데, URL 등록을 위해 별도로 웹 사이트를 방문하지 않아도 되며, 웹에서 들어오는 정보들을 걸러서 적절한 형태로 웹 브라우저에게 보내주게 된다.

해석의 기능은 여러 에이전트가 중간 처리 결과를 주고 받으며 협조적으로 문

제를 해결하고 최종적으로 하나의 결과를 도출해 낸다.

할당의 기능은 여러 가능성으로부터 에이전트간의 조정에 의해 그때 그때 해결 방법이 주어지며, 여러 개의 해가 있을 수 있다. 네트워크 부하를 분산시키거나 관리하고 진단하는 분야에서 활용될 수 있는데, 그 예로 지연 검색 에이전트가 있다. 이것은 네트워크 사용량의 과다로 인한 검색 속도의 저하될 때의 문제를 해결하기 위한 것으로서, 네트워크 부하가 적은 시간대를 찾아 자동 검색을 수행한다. 이를 위해서는 지연 검색 URL 테이블의 정보가 필요하다. 각 사용자마다 가지고 있는 지연 검색 URL 테이블에는 시간 정보와 URL 리스트 정보가 있으며, 남은 시간을 확인하여 정해진 시간 단위로 지연 검색 URL 테이블에 있는 URL 을 대상으로 네트워크 속도를 점검하고 조건이 맞으면 검색을 시작한다.

보조자 역할의 기능은 다른 측면에서의 접근 방법으로 사용자를 대신하여 인터넷을 활용하여 작업을 수행하고 사용자로 하여금 효과적인 선택을 할 수 있도록 하기 위한 것이다. 이에 관한 연구들을 크게 두가지로 분류해 볼 수 있는데, 웹 브라우징 에이전트(Web Browsing Agent)와 백그라운드 에이전트 (Background Agent)이다.

웹 브라우징 에이전트는 사용자가 웹 브라우저를 사용하는 동안에 효과적인 선택을 하도록 도와주는 기능을 수행한다. 일반적인 검색 엔진은 사용자가 직접 질의한 내용과 관련이 있는 웹 사이트들을 링크로 구성한 페이지만을 결과로 준다. 하지만 웹 브라우징 에이전트는 직접적인 상호작용 없이도 사용자가 평소에 접근하는 URL(Uniformed Resource Locator)을 자동적으로 프로파일(Profile)에 실시간 갱신(Real Time Update)하며, 이를 근거로 하여 사용자가 정보를 검색할 때 관련 사이트들을 추천한다. 또한 사용자가 뉴스 그룹에 머무는 동안 뉴스 그룹의 기사들을 대상으로 하여 사용자에게 관심이 될 만한 기사들을 추천한다.

백그라운드 에이전트는 사용자와 에이전트간에 직접적인 연결이 없는 상황에서도 계속 일을 처리하여 사용자에게 결과를 알려 주는 기능을 수행한다. 사용자에게 잠재적으로 관심이 될 만한 웹 사이트의 문서들을 수치로 표현되는 관심도와 함께 사용자에게 보여 주고 사용자의 반응을 통해 사용자의 선호도를 학습한다. 지정된 시간에 구동되며 지정된 시간에 사용자에게 URL 을 제시한다. 예로, 유즈넷

(Usenet) 기사를 대상으로 하여 사용자와 시스템간의 상호작용을 통해 스스로를 훈련시켜 사용자의 관심 분야가 포함된 기사들을 주기적으로 사용자에게 제공한다.

이외에 위의 두가지 기능을 포함한 에이전트가 있다. 이것은 사용자가 접속하고 있는 현재의 웹 문서에 대해 사용자의 관심이 될 만한 사이트임을 알리고, 정보 변화 감시를 위해 URL 등록을 추천하는 웹 브라우징 에이전트의 기능과 사용자가 웹에 연결되어 있지 않은 경우에도 에이전트가 자동으로 정보 변화 감시와 지연 검색 등의 작업을 수행하는 백그라운드 에이전트 기능을 동시에 가진다.

계획이라는 것은 어떠한 것이 정의된 순서대로 수행된다면, 정의된 계획의 목표를 만족시키도록 실세계의 현재 상태를 바꾸는 일련의 행동들이다[15]. 계획은 초기 상태로부터 목표 상태에 이르기까지 일련의 동작들을 생성하는데, 각 에이전트는 독립적인 목표를 가지고 있으며, 목표간의 분쟁이 발생하면 교섭과 조정에 의해 해결한다. 분산 항공관제가 그 예가 될 수 있다.

계획의 기능을 활용한 에이전트인 계획 기반 에이전트는 인공지능의 계획 시스템(Planning System)에 관한 연구로 지능 에이전트의 이론과 구조에 밀접히 연관된 분야이다[10]. 일반적으로 계획 시스템은 주어진 목표(Goal)들을 달성할 수 있는 일련의 구체적인 행동(Action)들과 그들의 실행 순서(Order)를 자동으로 구해준다. 계획 시스템(Planning System) 또는 계획기(Planner)를 핵심 모듈로 하는 에이전트 시스템들을 계획 기반 에이전트 시스템(Plan-Based Agent System) 또는 계획 에이전트 시스템(Planning Agent System)이라 부른다.

초기의 계획 시스템들은 대부분 선형 계획 방식(Linear Planning)을 사용하였다. 이 방식은 달성하고자 하는 목표들이 서로 독립적이며 어떤 임의의 순서에 따라 하나씩 달성해 가면 결국 전체 목표들을 모두 달성할 수 있다는 “선형 가정(Linear Assumption)”을 기초로 하고 있다. 이 방식은 계획의 일부로 포함될 새로운 동작들이 언제나 한쪽 끝으로만 삽입되도록 하여 계획이 선형적으로 확장되어 간다. 이러한 선형 가정은 목표들간에 상호작용(Interaction)이 심한 복잡한 현실 문제들을 풀 때는 많은 후진(Backtracking)을 유발하게 되어 결과적으로 계획 시스템의 효율과 성능이 저하된다. 이러한 선형 가정에서 탈피하여 다른 방법이 시도되기도 하였는데, 계층적 계획 시스템(Hierarchical Planning)과 비선형 계획 방식(Nonlinear

Planning, 부분 순서 계획 방식이라고도 함)이 그 시도의 일환으로 계획의 효율을 높이려는 접근 방법이다.

1980 년대에 이르러 네트워크와 분산 시스템의 발달로 인해 서로 분산되어 있는 여러 에이전트들이 공동의 목표나 작업을 위해 함께 협력하여야 할 필요가 높아졌다. 이것을 다루기 위한 것에는 분산 계획 방식(Distributed Planning), 멀티 에이전트 계획 방식(Multi Agent Planning)이 있다.

최근에 와서는 이동 로봇, 공정 제어, 작업 스케줄링과 같은 환경 변화가 심한 실시간 응용(Real-Time Application) 분야에 계획 시스템의 도입이 늘어남에 따라 반응형 계획 방식(Reactive Planning)에 관한 연구가 진행되고 있다.

3.3 에이전트 표준화의 동향

여러 에이전트들 간의 분산 협동 처리를 위해서는 에이전트간 통신이 필수적이며, 이는 정보나 작업 처리의 공유와 교환에 그 목적이 있다. 에이전트간 통신의 가장 큰 장애는 각 에이전트가 가진 이형질성(Heterogeneity)이다. 각각의 에이전트는 각기 다른 개발자에 의해 서로 다른 플랫폼을 바탕으로 서로 다른 목적으로 개발되었기 때문에 서로간의 통신을 위해서는 상호 이해가 가능한 언어와 프로토콜이 필요하다. 이의 대표적인 것이 KQML 이라고 할 수 있다.

에이전트 기반의 응용 프로그램들간의 상호이용성(Interoperability)을 최대화하기 위하여 에이전트 기술의 표준화를 목적으로 FIPA(Foundation for Intelligent Physical Agents)라는 국제조직이 1996 년에 정식으로 설립되었다. 이 회합에서는 에이전트 기술의 표준화에 대해 국제 규모의 표준화 조직의 필요성, 표준화의 기본 방침, 에이전트의 기본 기능 등에 대하여 논의 되었다. 그 이후의 회합에서는 연구 회의 형태로 개최되었으며, 수차례의 회합을 거치면서 에이전트 표준안이 다듬어졌다. (Internet URL : <http://drogo.cse.stet.it/fipa/>) FIPA 는 FIPA97 이라고 하는 7 개 파트의 사양을 이미 개발하였고, FIPA98 이라는 새로운 version 도 개발하였다.

7 개 파트의 사양은 크게 두가지 부분으로 나누어 볼 수 있는데, 요소 기술에 대한 표준 부분과 응용을 실현하는 방법을 나타내는 참고 부분으로 되어 있다.

표준 부분에는 3 개 파트가 있다. 첫번째는 에이전트를 관리하기 위한 참고 모

델을 규정하는 파트인 에이전트 관리(Agent Management)이다. 여기에서 에이전트를 관리하기 위하여 에이전트를 정의할 필요가 있는데, FIPA 에서는 에이전트를 FIPA-ACL (Agent Communication Language)을 이해하는 실행 주체라고 정의하고 있다. 에이전트가 활동하는 기반을 AP(Agent Platform)이라고 하며, AP 는 아래의 3 개의 에이전트로 구성된다.

- 1) 에이전트 환경 모두를 관리하는 AMS(Agent Management System)
- 2) 에이전트간의 통신을 매개하는 ACC(Agent Communication Channel)
- 3) 에이전트에 대한 검색 서비스를 제공하는 DF(Directory Facilitator)

FIPA 는 이들의 구현은 관여하지 않으며 각 파트가 가져야만 하는 기능과 인터페이스를 규정하고 있다.

두번째로 FIPA 의 대부분을 차지하는 파트로 에이전트간의 통신을 위한 언어를 규정하는 에이전트 교류 언어(Agent Communication Language)가 있다. 이 규정에는 대화 이론을 바탕으로 고안된 메시지군과 의미, 구문 규칙이 포함된다. 에이전트는 이러한 메시지를 통하여 서로 통신을 한다. FIPA 의 표준화 대상은 메시지 패싱의 구현 모델이 아니라 메시지 그 자체의 의미와 구문 규칙이다.

세번째로는 에이전트가 기존의 소프트웨어 자원(라이브러리, 데이터베이스, 디바이스 등)을 적극적으로 이용하기 위하여 에이전트와 소프트웨어의 통합 참조 모델 및 온토로지(Ontology)를 규정하는 파트인 에이전트/소프트웨어 통합 (Agent/Software Integration)으로 되어 있다. 이 모델의 중심 요소는 소프트웨어에 관한 검색 서비스를 제공하는 ARB(Agent Resource Broker)와 에이전트와 소프트웨어의 교환 서비스를 제공하는 래퍼(Wrapper) 에이전트이다. 온토로지는 행위 (Action), 술어, 서비스 기술의 3 개 영역으로 되어 있으며, 이 표현에 이용되는 어휘와 문법을 규정하고 있다.

참고 부분에는 4 개의 응용을 선정하여 이를 실현하기 위한 기술의 표준화를 한다는 방법으로 합의하였다. 첫번째로는 Personal Travel Assistant 이며, 에이전트가 항공권을 구입, 호텔 예약, 여행의 계획, 대급 지불 등 전자화된 여행 관계 서비스를 사용자의 대리로서 수행하는 응용에 대해 기술하고 있다.

두번째로는 Personal Assistant 이며 에이전트가 회의 일정의 예약, Directory 서

비스, 정보 관리, 여행 계획 등을 수행하여 주는 전자비서 응용에 대해 기술하고 있다.

세번째로는 **A/V Entertainment and Broadcasting** 이며, 에이전트가 시청자인 사용자의 선택 프로그램을 자율적으로 수집하여 제시하는 응용을 기술하고 있다. 참고 모델은 사용자 도메인, 프로바이더 도메인, 사용자 그룹 도메인 등 3 개의 도메인으로 구성되어 있다.

네번째로는 **Network Management and Provisioning** 이며, 에이전트가 사용자에게 맞추어 네트워크 자원 관리를 하는 통신 시스템을 위한 동적 VPN(Virtual Private Network)에 대하여 기술하고 있다. 이 파트의 목적은 이용할 수 있는 여러 가지 서비스 프로바이더나 네트워크 서비스 중에서 가장 좋은 조건의 네트워크 서비스를 사용자에게 제공하기 위한 것이다.

FIPA 의 7 개 파트의 사양을 살펴 보았다. 기존의 표준화 과정은 이미 실증된 기술 가운데서 보편적인 것을 표준 사양으로 선택하는 것이 보통이다. 하지만, FIPA 에서는 종래의 보편적인 기술을 선택하는 방법이 아닌 기존의 기술을 참고하면서 새로운 기술의 사양을 만드는 방법을 취하고 있다.

에이전트 기술의 성공적인 보급을 위해서는 다른 환경이나 다른 종류의 에이전트간 원활한 통신이 필수적이며, 에이전트간의 표준 통신 언어, 표준 온토로지, 기존 소프트웨어와의 인터페이스 등에 관한 국제적 표준화가 필요하다. FIPA 를 통한 표준 사양은 앞으로 에이전트 기술의 보급에 많은 영향을 줄 것으로 보여진다.

4. 에이전트의 응용

4.1 속고형 에이전트 분야

(1) InfoFinder

앤더슨 컨설팅 연구실에서 개발한 지능 웹 에이전트인 InfoFinder 가 있다 [16,17]. 이것은 브라우징 과정의 관심도를 기반으로 학습이 이루어진다는 점에서 뒤에서 설명될 Webwatcher 와 같지만, Webwatcher 가 비감독 학습(Unsupervised Learning) 방식을 이용하는데 비해, InfoFinder 는 관심 문서에 대한 사용자의 직접적인 관심 여부와 관심 영역을 입력받아 사용자의 관심을 학습하도록 하여 사용자가 자신의 관심도를 직접 표현하는 감독 학습(Supervised Learning) 방식을 이용한다.

사용자가 웹 브라우저를 사용하여 온라인 문서를 읽을 때 관심 문서에 대해 InfoFinder 의 메뉴 중 관심 여부를 나타내는 아이콘을 이용하여 사용자의 관심과 문서에 대한 관심 분야를 직접 지정하며, 관측된 문서를 사용자의 관심 분야 영역 별로 저장한다. 감독 학습 방식은 사용자의 관심 영역을 보다 정확히 표현할 수 있는 장점이 있는 반면 사용자에게 자신의 관심도를 항상 표시하게 하는 불편을 수반한다. 이 에이전트는 사용자의 관심도를 만들기 위해 관심 영역에 대한 키워드를 학습하게 되며, 이렇게 학습된 영역별 키워드는 사용자가 관심 영역에 대한 문서를 검색할 때 이용할 수 있다.

(2) 구인/구직 에이전트

에이전트의 기본 개념에 협상 개념을 포함한 지능 에이전트가 있다. 이 분야에는 프로토콜, 이슈(Issue), 추론 모델의 세 분야에 대해 주로 연구되고 있으며, 자동화된 협상을 위해 협상 모델이 필요하다. 포항공과대학교에서 구축한 구인 구직을 위한 에이전트 시스템인 EARTS-II 는 분산 환경에서 협상을 통해 능동적이고 적응성 있는 멀티미디어 데이터 서비스를 위한 지능 에이전트이다[5,18,19].

시스템은 에이전트를 생성하여 파견하고 에이전트가 가져온 결과에 따라 서비스 받을 수 있도록 해주는 구인/구직 시스템과 파견된 여러 에이전트들 사이에서 구인과 구직이 적절하게 이루어지는 작업 시장으로 구성되어 있으며 그림 4-1 과

같다.

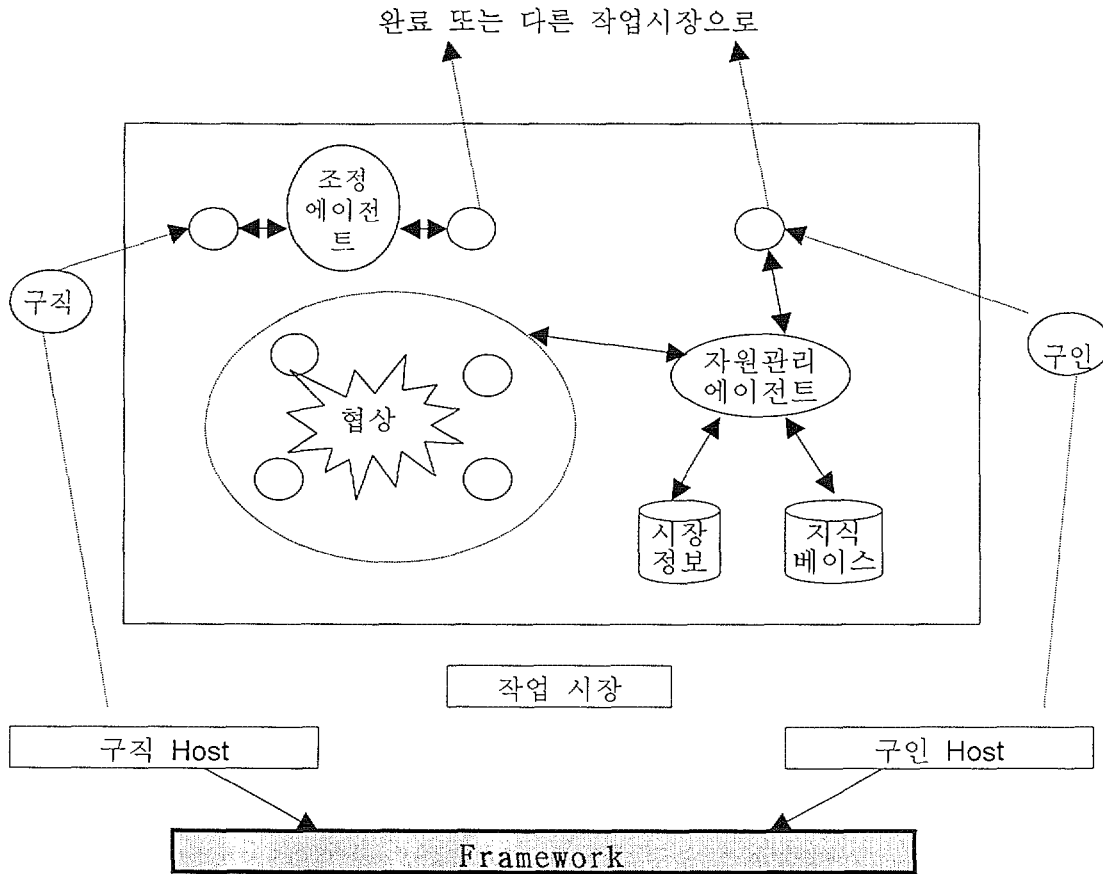


그림 4-1. 구인/구직 에이전트 시스템의 구조

사용자가 원하는 조건을 제시하면 구인/구직 시스템은 지능 에이전트를 생성하고, 지능 에이전트가 다른 에이전트와의 협상을 통해 서로의 조건에 맞는 적당한 결과를 제안하며, 제안된 상대와는 멀티미디어 데이터 서비스를 통하여 사이버 공간에서 구인 구직이 이루어진다.

작업 시장에서는 생성된 여러 개의 구인/구직 에이전트들이 협상을 통해서 구인/구직이 일어 나게 되는데, 이를 위해 조정 에이전트, 자원 관리 에이전트 등이 있다. 또한 에이전트들이 좋은 결정을 내기 위해 자신이 속해 있는 영역에 대한 지식을 갖고 있는 지식 베이스와 다른 직업 시장에 대한 정보를 갖고 있는 데이터베이스가 있다.

이 에이전트에서 사용된 협상 알고리즘은 연결 함수, 만족 함수, 그리고 교차 연산, 변종 연산을 적용하여 구현한 것이다. 이것은 구인 구직 뿐만이 아니라 미아 찾기, 이산가족 찾기 등 자신이 가진 효용성을 최대로 하려는 양자간의 관계에서 최적의 해를 찾는 여러 분야에 적용할 수 있다.

(3) 결재 관리 에이전트 시스템

한국전자통신원에서 구축한 결재 관리 에이전트는 결재, 출장, 구매 등과 같은 업무 흐름에 대한 워크플로우(Workflow) 멀티 에이전트 시스템을 구성하는 하나의 에이전트이며, 전형적인 지능 에이전트 시스템이다[6].

전체 구성은 조정 에이전트, 사용자 인터페이스 에이전트, 그리고 데이터베이스, 결재, 출장 등에 관한 응용 에이전트들로 구성되며 그림 4-2 와 같다.

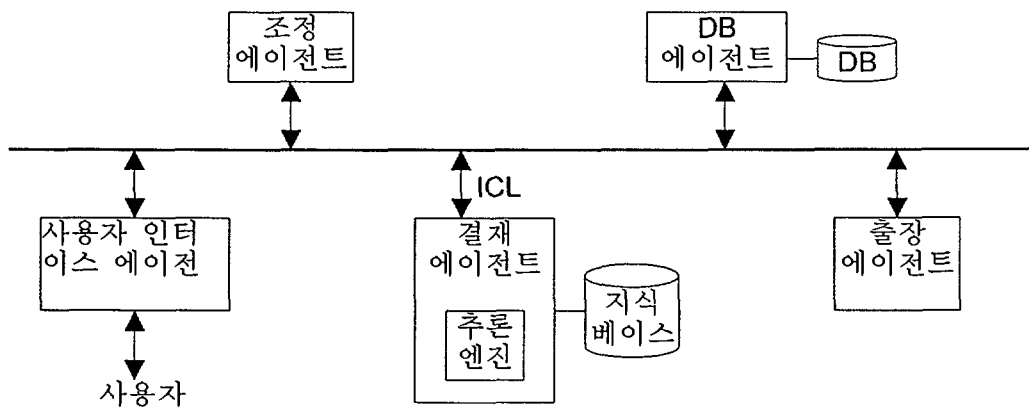


그림 4-2. 결재 관리 에이전트 시스템의 구조

조정 에이전트는 응용 에이전트들의 기능과 특성에 대한 메타 지식을 갖고 있으며, 응용 에이전트들간의 명령, 지식, 데이터의 상호 교류를 위한 조정 기능을 수행한다.

사용자 인터페이스는 사용자와 관련된 작업을 처리해 준다.

응용 에이전트는 특정 영역의 작업을 전문적으로 처리하는 기능을 갖는다. 이 에이전트들은 네트워크를 통하여 ICL(Interagent Communication Language)이라는 에

이전트 통신 언어로 통신한다. 응용 에이전트의 하나인 결재 에이전트는 결재에 필요한 지식 베이스를 갖고 있으며 다른 에이전트로부터 질문 받으면 추론을 통해 질문에 대한 답을 생성하고, 그 답을 요청한 에이전트에게 보내 준다. 추론 중에 조직 정보, 개인 일정 정보와 같은 정보를 다른 에이전트에게 요청하여 받을 수 있도록 확장할 수 있다.

(4) 일정 관리 에이전트 시스템

이화여자대학교에서 구축한 일정관리 에이전트는 일정관리를 위한 개인 정보를 처리하고, 스케줄 조정을 위한 의사 소통 과정을 대행하는 지능 에이전트이다 [20].

이 시스템은 추론 모듈, 자원관리 모듈, 통신 모듈로 구성된 에이전트와 사용자의 입력을 받는 인터페이스 모듈로 구성되어 있으며 그림 4-3 과 같다. 이 에이전트의 특징은 지식 확장을 위해 에이전트 사이의 정보의 교환이며, 이를 위해 통신 기법을 적용하였다.

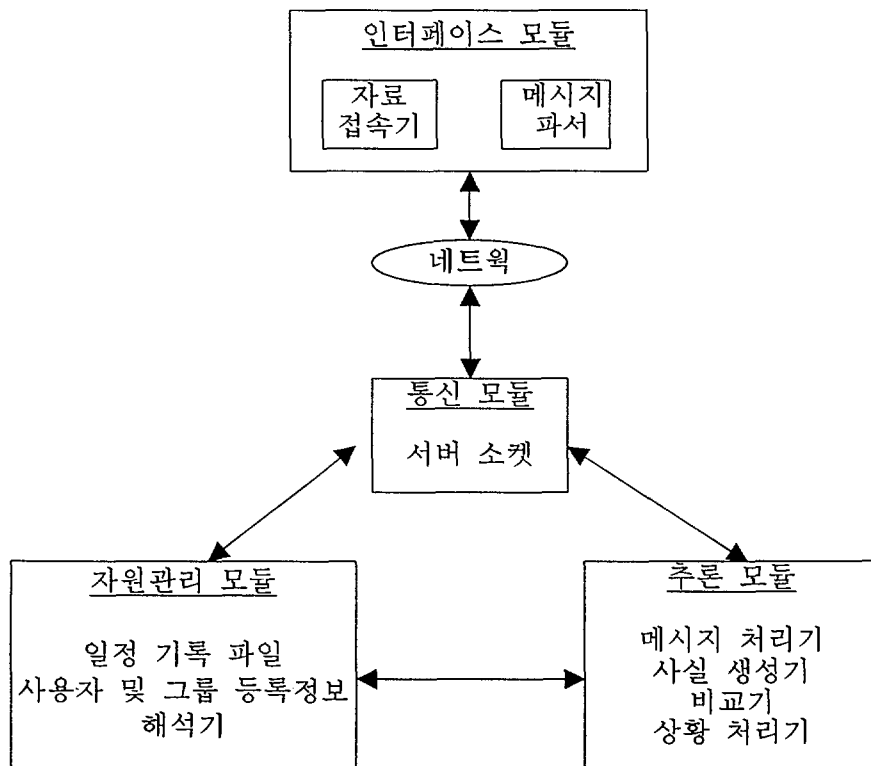


그림 4-3. 일정 관리 에이전트 시스템의 구조

사용자는 인터페이스 모듈을 통해 등록하고, 일정 내용을 입력하게 되는데, 이를 토대로 다른 사용자가 스케줄 요청을 할 때, 추론 모듈에서 사용자의 일정과 선호도를 고려하여 상황을 판단하고 해석기를 이용한 메시지 교환 과정을 통해 조정을 하게 된다. 추론 모듈은 주어진 지식을 바탕으로 상황을 판단하며, 사용자가 기대하는 행위를 예측하여 사용자에게 기대되는 사고와 관련된 행위를 산출한다. 자원관리 모듈은 에이전트가 사용자의 일정 정보, 사용자 및 그룹의 등록 정보, 에이전트에 대한 정보 및 해석기 등을 포함하고 있으며, 자원의 추가와 삭제, 관련 처리를 한다. 통신 모듈은 자료 형식이 다른 단위 프로그램들 사이에 데이터를 교환하기 위해 필요하다.

일정 관리 에이전트는 사용자와의 상호 작용이 가능하고, 에이전트간의 메시지 교환을 수행하며, 사용자의 편의를 위해 웹 인터페이스를 활용하였다.

(5) WiseWire

WiseWire 웹 에이전트는 WiseWire 사에서 만든 지능 웹 에이전트로 인터넷에서 정보를 검색할 때, 사용자의 기존 검색을 분석하여 얻은 프로파일을 이용하여, 사용자들에게 각 개인이 원하는 정보를 검색할 수 있도록 한다[21]. WiseWire 웹 에이전트는 Business & Finance, Arts & Humanities, Computers & 인터넷과 같은 일반적인 주제들을 위주로 사용자 검색 기능을 제공하며, 각 주제에는 하위에 더욱 전문화된 다수의 주제들을 포함하고 있다. 사용자가 관심을 가지는 주제를 선택하면 WiseWire 는 주제와 관련된 문서를 제공하며 사용자는 제안된 문서들에 대해 관심 정도와 관심 분야에 따라 평가하고 이러한 선택과 평가를 기반으로 각 사용자의 관심을 학습한다. WiseWire 가 사용자의 관심을 학습하기 위해 사용하는 속성은 사용자의 관심을 0 부터 10 까지 세분화한 관심도와 문서의 내용, 관련성, 저자, 소스, 검색 날짜에 대한 정보이다. WiseWire 는 사용자들에게 문서에 대한 속성들을 입력 받고 이를 학습하여 사용자의 관심 주제에 대하여 특성화(Personalize)시킨다. WiseWire 는 사용자의 관심에 대한 변화를 직접 피드백하도록 하고 있다. 즉, 사용자가 자신의 관심도에 대한 정보에 직접 접근하여 관심 문서에 대해 삭제하거나 문

서에 대한 학습을 수행하며 학습할 주제나 문서에 대하여 사용자가 긍정적 힌트(Positive Hint)와 부정적 힌트(Negative Hint)를 주도록 한다. 이러한 관심 정보와 비관심 정보를 기반으로 사용자의 관심(Profile)을 기계 학습을 이용하여 학습하고, 이렇게 학습된 사용자의 관심을 이용하여 적응력 있는 검색을 수행한다. WiseWire 는 각 사용자에 대한 관심뿐 아니라, 일군의 그룹에 대한 학습(Collaborative Learning)을 수행한다. 즉, 비슷한 관심을 보이는 사용자들의 가상 공유 그룹을 생성하고 그룹 내의 모든 사용자들의 관심 문서에 대한 학습 결과를 이용하여 그룹 내의 관심 주제와 관심 정보에 대하여 재학습한다. 그룹 내의 여과(Collaborative Filtering)를 이용하여 그룹 내의 한 사용자가 학습한 내용을 그룹 내의 다른 사용자가 이용할 수 있도록 하여 검색 효율을 높였다.

4.2 협력 에이전트 분야

(1) 정보 검색 분야의 응용

에이전트 기술은 분산 인공지능, 컴퓨터 통신, 소프트웨어 공학 등의 여러 분야가 종합되는 새로운 분야이며, 지식 기반 시스템이나 분산 인공지능에 대한 전통적 기술들은 전통적인 기술들만으로는 잘 해결되지 않았던 실세계의 문제들의 일부를 풀기 위하여 에이전트 시스템 기술로 응용될 수 있다[6].

정보 검색 분야에서는 동적으로 변화하는 분산 정보 네트워크상에서 필요한 정보를 다수의 에이전트가 협력하여 검색하고 수집한다. 정보 자원이 대규모이고 다양화되고 동적으로 변화하고 있어 검색을 위해 많은 시간과 노력이 필요하기 때문에 정보 검색 분야에서 에이전트의 도입은 절실하다고 할 수 있다. 정보 검색의 각 분야별로 살펴 보고자 한다.

① 정보 처리 에이전트

정보를 처리하는 에이전트는 크게 두가지 종류로 나눌 수 있는데[1], 하나는 원하는 정보를 찾아주는 정보 검색(Information Retrieval 또는 Information Searching) 에이전트이고, 다른 하나는 사용자의 취향에 맞도록 정보를 가공하고 걸러주는 정보 여과(Information Filtering) 에이전트이다. 정보 검색은 원하는 검색어에 대한 수

많은 정보 사이트를 자동으로 찾아주는 역할을 수행한다. 이것은 흔히 검색 엔진을 연상할 수 있지만 보통의 검색 엔진 자체만으로는 자율성이나 협력성, 적응성 등의 성질을 만족하지는 못하므로 에이전트로 보기에 는 미흡하다.

정보 여과와 정보 검색은 인터넷 상의 정보를 사용자에게 제공해 주는 것은 비슷하지만 약간 다른데, 정보 검색이 사용자의 질의에 따라 정보를 찾아주는 (Finding) 것이라면, 정보 여과는 사용자의 프로파일에 따라 필요 없는 정보를 삭제 하는(Removing) 것이다.

정보 검색 에이전트는 다수의 사용자가 다수의 정보 소스에 대해 원하는 정보를 요청하고 결과를 얻게 되는데, 이 과정에서 에이전트는 중개 역할을 하게 된다. 사용자가 원하는 것이 무엇이고 원하지 않는 것이 무엇인지를 기억하여 적응이 가능하도록 하는 기능을 수행하고, 사용자와 정보 소스간의 통신을 지원하고 결과 문서의 라우팅 역할을 담당하고, 각 정보 소스에 특징적인 수행을 책임지는 등의 역할을 한다. 이 에이전트의 예로는 SAIRE, Distributed Intelligent Search, Restina, CARROT 이 있다[1].

정보 여과 에이전트의 정보 여과 개념은 끊임없이 유입되는 정보 중에서 필요한 것이 무엇이고 필요하지 않은 것이 무엇인지를 판단하여 필요 없는 것은 무시하는 것이다. 정보 여과의 과정은 사용자의 프로파일이 중요한 역할을 하는데, 여기에는 사용자가 관심을 가지는 사항에 대한 정보가 포함된다. 정보 여과의 과정은 얻어진 정보에 대해 사용자의 프로파일과 비교하여 관심이 있는 정보만을 걸러서 저장한 후 사용자는 걸러진 정보만 볼 수 있게 한다. 사용자는 걸러진 정보를 본 후 그것이 실제 자신의 원하는 것이었는지를 알려주게 되고 이 과정을 거치면서 사용자 프로파일을 재구성할 수 있다[1].

정보 검색도 마찬가지이지만 정보 여과의 과정은 수반하는 여러 문제점을 안고 있다. 첫째, 단어 선택 문제이다. 관심도를 단어로 표현한다고 할 때 같은 관심 분야라 하더라도 사람마다 선택하는 단어가 다를 수 있고, 시간 경과에 따라 다른 형태로 표현할 수 있다. 둘째, 문서가 구조화되어 있지 않거나 구조화되어 있다 하더라도 일부만 되어 있는 문제이다. 일반 텍스트의 나열인 내용을 통해 여과 작업을 하는 것은 어려운 일이다. 또 사용자가 얻는 정보의 종류는 다양하고 각각 다른

구조를 가지고 있기 때문에 이를 모두 고려하는 작업이 중요한 쟁점이기도 하다. 셋째, 정보 여과 에이전트를 훈련시켜야 한다는 것이다. 사용자 프로파일은 처음부터 사용자의 의도를 완벽하게 나타낼 수 없기 때문에 점진적으로 만족스러운 형태로 재구성하여야 하며, 이를 위해 정보여과 에이전트를 훈련시켜야 하며 이것은 시간을 요하는 작업이다.

다수의 정보 여과 에이전트가 제시되어 있다. 어떤 정보를 대상으로 여과 작업을 하느냐에 따라 분류될 수 있으며, 크게 웹 문서 여과 에이전트, 상용 뉴스 여과 에이전트, Usenet 뉴스 여과 에이전트로 나눌 수 있다.

② 정보 변화 감시 에이전트

정보 변화 감시 에이전트의 기능은 점점 더 빠르게 변화하는 정보들을 사용자가 개별적으로 검사하는 수고를 덜고, 변화된 사실을 자동으로 감지하여 사용자에게 통보하는 역할을 수행한다. 이것과 관련한 주요 기능을 3 가지의 살펴 볼 수 있다. 첫째는 특징 추출의 기능이다. 대부분의 사용자는 중요한 사항에 대해 변경되었을 때 통보 받기를 원하며, 중요하지 않은 변화 요인에 대해서는 민감한 반응을 나타내지 않는다. 이를 위해 웹 문서의 내용을 대표할 만한 단어들을 선별하는 기능이 필요하다. 이것은 웹 문서에서 키워드를 추출하여 얻은 용어들을 기준으로 선별하게 된다. 둘째는 변화 측정의 기능이다. 변화 감시를 위해서는 새로운 정보를 가져와서 기존의 내용과 비교해야 한다. 즉, 문서간에 달라진 점이 있는 지를 파악하여야 한다. 셋째는 사용자에게 통보하는 기능이다. URL 의 내용이 변경되었다고 판단될 때 메일로 사용자에게 통보해 준다.

③ 정보 통합 에이전트

정보 통합 에이전트는 인터넷에서 제공되는 여러 개의 정보 사이트에서 사용자가 원하는 정보를 추출하여 하나의 형태로 제공하는 기능을 수행한다[1]. 이것이 필요한 이유는 다수의 정보 소스를 사용자가 일일이 접근하여 검색하는 노력을 줄여주고 사용자에게 불필요하다고 판단되는 것을 걸러 주는 점 등이 있기 때문이다.

(2) 논문 수집 에이전트 시스템

한국전자통신연구원에서 구축한 논문 수집 에이전트 시스템은 정보 검색을 위한 에이전트 시스템이다[22]. 사용자를 대신하여 메일링 리스트로부터 메일을 분석하고 메일내에 포함된 URL 을 추적하여 웹상에서 개인 가상 도서관을 구축하여 준다.

시스템은 전자우편 감시 및 여과 에이전트, 정보 수집 에이전트, 사용자 인터페이스 에이전트, 개인 가상 도서관, 지식 베이스로 구성되어 있으며 그림 4-4 와 같다.

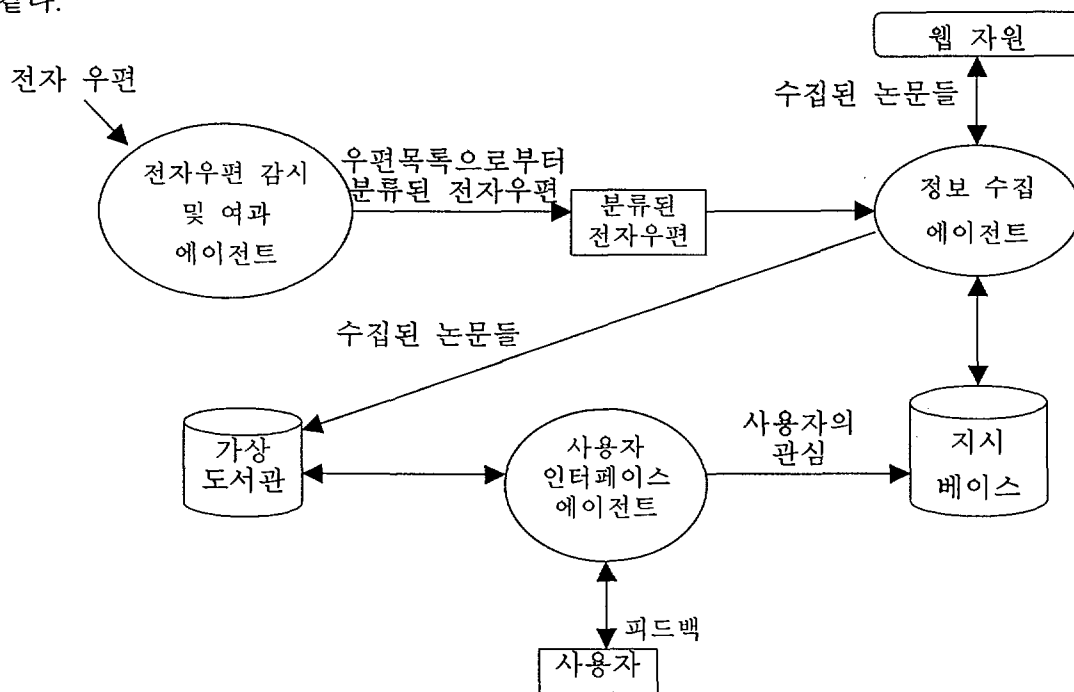


그림 4-4. 논문수집 에이전트 시스템의 구조

초기에는 메일링 리스트를 등록하기 위해 사용자로부터 내용을 입력받으며, 사용자를 대신하여 에이전트들이 수신된 메일을 토대로 사용자의 선호도에 맞는 논문과 논문에 관한 정보를 수집한다. 사용자는 사용자 인터페이스 에이전트를 이용하여 개인 가상 도서관의 자료에 대한 열람은 물론, 검색, 삭제 등의 작업을 할 수 있으며, 수집된 자료들에 대한 피드백을 줄 수 있다.

(3) 불건전 기사 검색 에이전트 시스템

충북대학교에서 개발한 불건전 기사 검색 에이전트 시스템은 에이전트 기법을 이용해 BBS(Bulletin Board System) 내의 방대한 양의 기사를 검색하여 불건전한 정보를 자동으로 검색하기 위한 정보 검색 에이전트이다[23].

이 시스템은 4 개의 모듈과 3 개의 데이터베이스로 구성되어 있다. BBS 연결 모듈은 BBS 서버에 접속하고 데이터를 수신해 BBS 구조 분석 모듈에 전달한다. BBS 구조 분석 모듈은 BBS 연결 모듈로부터 받은 데이터를 분석하여 데이터베이스에 저장하거나 데이터를 불건전 정보 검색 모듈로 전달한다. 불건전 정보 검색 모듈은 BBS 구조 분석 모듈로부터 받은 데이터에 대해 불건전성 여부를 판별한다. 사용자 인터페이스 모듈은 사용자로부터 필요한 데이터를 입력받아 검색 결과를 보여주고 삭제나 경고 등의 조치를 할 수 있도록 한다. 검색 안내 데이터베이스는 검색을 위한 키워드, 기사 분석을 위한 정보, 탐색 규칙에 대한 정보로 구성되어 있으며 그림 4-5 와 같다.

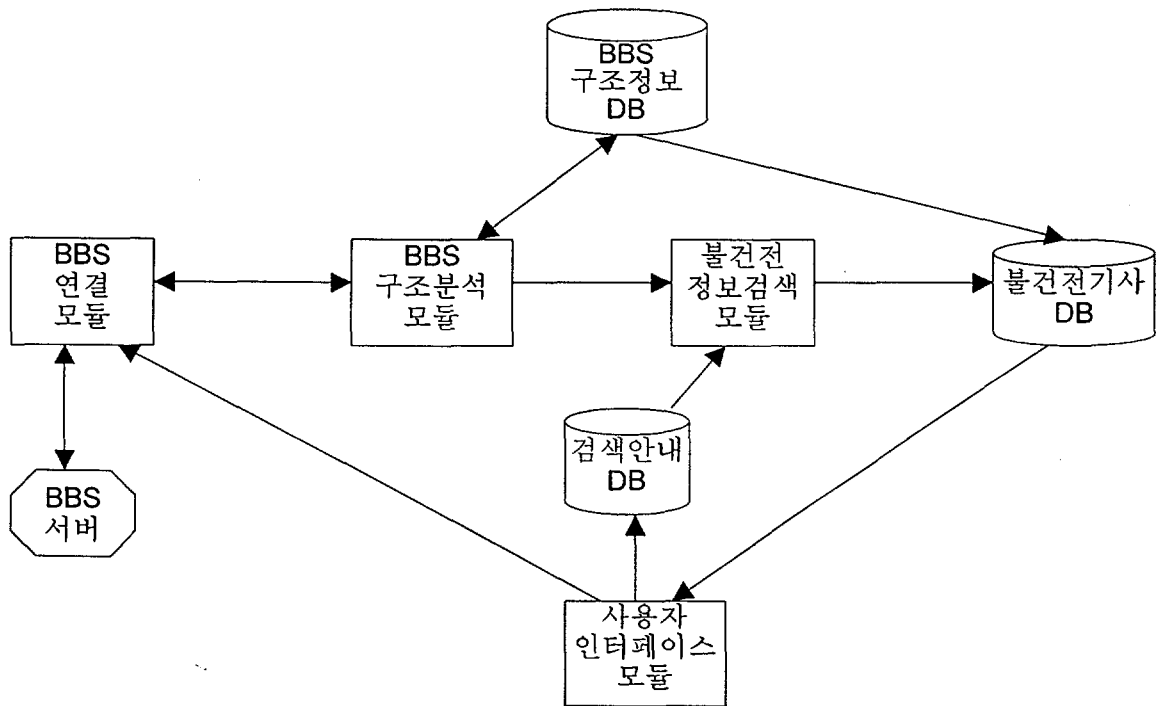


그림 4-5. 불건전 기사 검색 에이전트 시스템의 구조

BBS 구조 정보 데이터베이스는 불건전 정보 검색을 위해 BBS 내의 기사를 읽어 올 때 사용하는 기사의 위치 정보를 저장한다. 불건전 기사 데이터베이스는 불건전 정보를 저장한다.

기사 검색은 기사의 제목과 내용에서 불건전성에 관련된 키워드를 기반으로 하여 찾는다. 키워드 검색만으로는 부족하므로 불건전성 판정을 위해 제목의 키워드와 화일 확장자를 이용한 탐색 규칙을 사용한다.

입력된 정보를 바탕으로 에이전트 시스템은 불건전 정보를 포함한 기사를 자동으로 검색하여 사용자에게 알리는 등의 적절한 조치를 취한다.

(4) 카타로그 에이전트 시스템

중앙대학교에서 카타로그 에이전트 시스템은 사례 기반 추론을 이용하여 전자상거래의 효율을 높이기 위한 협력 에이전트이다[24].

시스템은 카타로그 에이전트, 검색 에이전트, 사례 기반 데이터베이스로 구성되어 있으며 그림 4.6 과 같다. 카타로그 에이전트는 사용자에게 신상 정보를 요구하고 기존의 사례를 이용하여 사용자에게 보여주는 기능을 한다. 사례 기반 데이터베이스는 사용자가 입력한 물품의 종류에 따라, 개인의 신상에 따라 그룹화하여 구축되도록 하였다. 검색 에이전트는 사용자가 신상 정보와 물품을 입력하면 사례

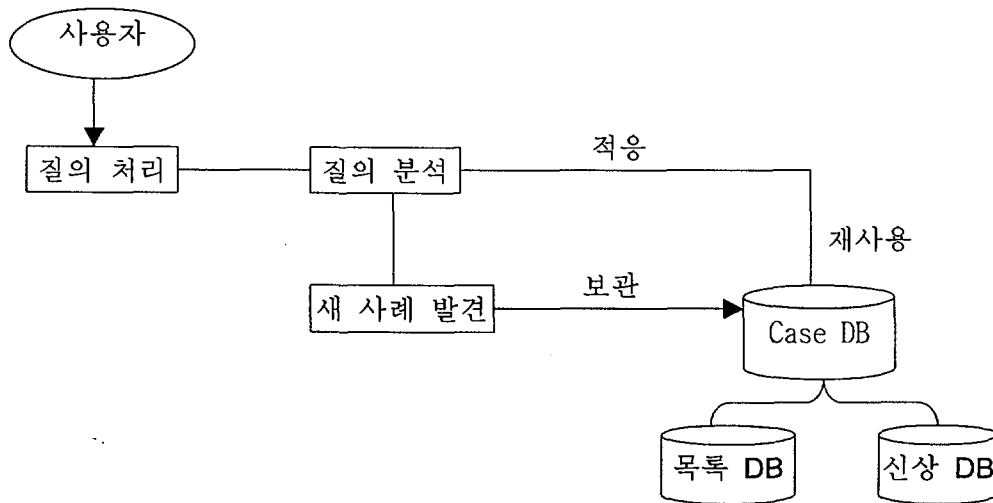


그림 4-6. 카타로그 에이전트 시스템의 구조

기본 데이터베이스로부터 신상이 비슷한 사용자가 원했던 종류의 물품을 찾아 사용자에게 결과로 보여준다.

카타로그 에이전트 시스템은 사용자에게 좀 더 편리한 안내를 할 수 있고 구매 효과도 높일 수 있어 기존의 전자상거래 시스템을 보완한 에이전트이다.

4.3 보조 에이전트 분야

(1) 웹 에이전트

웹 에이전트는 에이전트 기술을 웹과 연결하여 사용자가 좀 더 편리하게 웹을 사용할 수 있도록 한다[25]. 웹 에이전트의 구조를 그림 4-7에 나타내었다. 웹 에이전트는 기존의 인터넷과 관련하여 주어진 기능을 수행하기 위해 에이전트가 활동하는 하드웨어 기반 구조가 필요하며, 하드웨어 기반 구조 위에 웹 서버가 구축된다. 이러한 기반 위에 에이전트 서버가 조성되며, 에이전트 서버는 에이전트가 생성되어 작업을 수행할 수 있도록 기본적인 공통 기능을 지원한다. 에이전트 서버에 의하여 생성된 에이전트는 적절한 우선순위에 따라 실행된다.

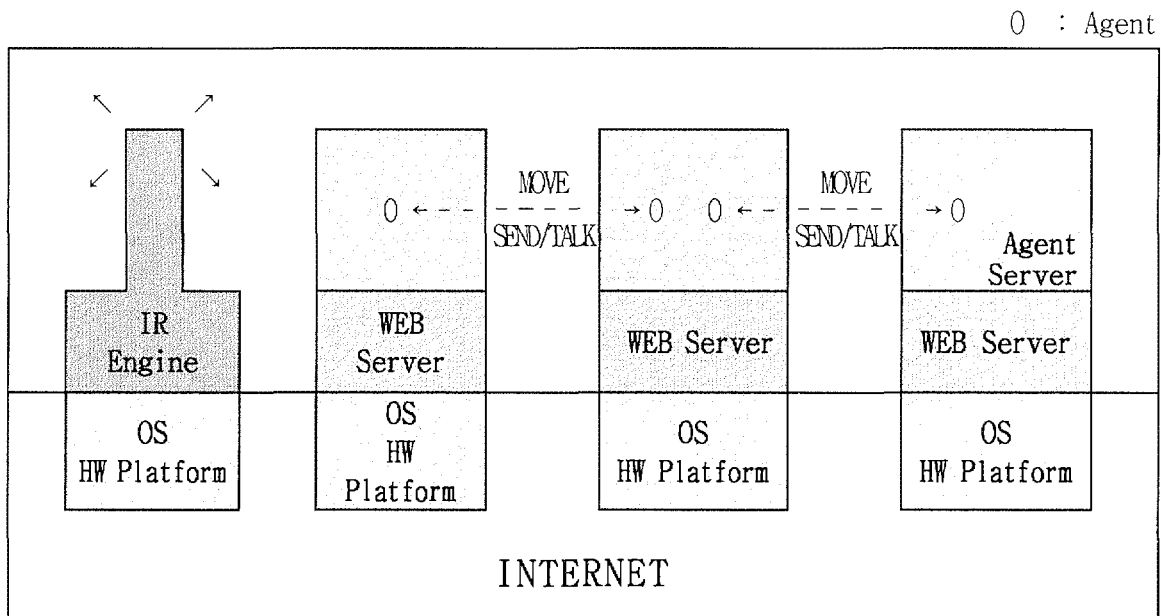


그림 4-7. 웹 에이전트의 구조

웹 에이전트에서는 사용자측의 브라우저에서 시행 명령이 있어야 정보 검색

등 일련의 동작이 야기되며, 그 작업이 끝나면 다시 새로운 명령을 기다리는 수동적인 체제를 갖고 있다[26].

현재 이용되고 있는 웹 검색 시스템은 일반적인 구조를 가지고 있어서, 각 개인의 다양한 요구 사항을 고려할 수 없고, 이로 인하여 불필요한 정보를 많이 검색하여야 하는 단점을 보완하고자 한다. 이를 위해서는 기존의 검색 시스템의 단점을 극복하고 사용자 관심도를 학습, 관리하도록 하여, 웹 검색 시스템을 활용하는 사용자에게 가장 적절한 정보만을 가져오고, 시간이 지남에 따라 사용자의 관심도를 학습할 수 있는 지능 웹 검색 에이전트를 구축하는 것이다.

정보 자원들이 여러 개 존재할 경우 어느 곳에 질의를 줄 것인가 하는 문제에서 출발하여 다중 자원 접근 문제를 해결하기 위해 신경망 메카니즘을 이용한 웹 정보 검색 에이전트가 연구되었다. 즉 사용자의 질의에 대한 피드백으로부터 정보 자원들에 대해 학습하고 학습된 지식을 바탕으로 새로운 질의에 대해 사용자가 원하는 정보 자원을 찾아내 그로부터 정보를 검색한다.

여기에서 좀더 진보한 신경망 기반 멀티 에이전트 웹 정보검색에 대한 연구가 있다[27]. 검색 도구라고 하는 전형적인 정보 검색 에이전트는 정보들에 대한 색인을 단일 플랫폼에 데이터베이스로 구축하고 TF×IDF 알고리즘에 기반한 벡터 공간 모델(Vector Space Model)[28]과 같은 전통적인 모델을 사용한다. 이 방법은 많은 사용자들에 의해 검색이 빈번히 이루어지는 환경에서는 심한 병목현상과 접근 지연 현상을 일으킬 수 있다.

이러한 문제의 해결을 위해 메타 검색 서비스라고 하는 분산 웹 정보 검색 에이전트가 있다. 이것은 정보 검색 작업을 각자의 색인 데이터베이스를 갖고 있는 다수의 검색 도구들에 분산시키는 방법을 사용한다. 사용자의 질의를 여러 검색 도구들에 동시에 브로드캐스트(Broadcast)하고 제공되는 결과들을 사용자에게 웹 문서 형태로 제시한다. 그러나 대부분의 경우에 검색 도구들은 주제별로 구축되어 있어 웹 정보에 대한 색인들이 주제별로 클러스터링 (clustering)됨에 따라 무분별한 브로드캐스트는 불필요한 네트워크 자원의 접근과 이에 따른 많은 통신비용을 발생시킨다.

사용자 프로파일 기반 개인 웹 에이전트에 대한 연구도 활발하다[25,16]. 사용

자가 브라우징한 직접적인 관심 문서들을 대상으로 관심도를 표시하여 정확한 학습 입력 값이 구축된다. 또한 관심 영역의 특성을 고려한 키워드 추출 기법과 특성 추출 기법을 적용하여 중요 키워드의 정확도를 높인다. 이때 관심 영역에 속하는 키워드 가중치를 설정하여 중요도를 높인다. 추출된 키워드들은 학습 시스템을 이용하여 결정 트리 형태로 최적화 되고 소수의 중요 키워드를 추출하는데 이용되어 사용자의 선호성을 학습함으로써 사용자 프로파일을 구성하고 관리하며, 관심이 될만한 사이트를 추천한다. 기존 웹 에이전트는 관심 문서에 대한 학습을 한번 수행하는 것에 비해 사용자 프로파일 기반 개인 웹 에이전트에서는 재귀적 학습 방법이 이루어진다.

(2) Webwatcher

카네기 멜론 대학(CMU)에서 구축한 웹 에이전트인 **Webwatcher** 는 사용자 브라우징 행위 정보를 기반으로 관심 문서를 구축한다[16,29,30]. 웹 브라우저 상위에서 실행하면서 사용자의 행동을 모니터링하여, 개인 사용자에게 적응력을 부여한 에이전트이다. **Webwatcher** 는 사용자의 컴퓨터 내부에서 데몬으로 실행되면서, 사용자가 보는 웹 브라우저와 사용자의 행동을 관측하고, 이를 기반으로 사용자의 관심 문서들을 모으고, 모아진 관심 문서들을 분석하여 사용자의 관심 파일을 만들며, 사용자가 정보 검색시에 이 관심 파일을 사용하여 사용자의 관심 문서를 예측하여 사용자에게 제안하게 된다.

Webwatcher 의 내부는 크게 세부분으로 되어 있다. 첫째는 사용자가 검색하는 웹 페이지를 관측하는 모니터링 부분으로 프록시(Proxy)를 이용하였으며, 이것을 사용자의 컴퓨터 내부에 두어 사용자가 현재 검색하고 있는 문서의 위치(URL)를 관측하고 사용자가 보고 있는 문서에 대한 행동, 즉, 웹 페이지를 저장하거나 북마킹하거나 프린트하는 등의 행동을 모니터링한다. 둘째는 모니터링한 결과인 문서들을 분석하여 사용자의 관심 영역을 측정하는 부분으로, 관찰된 문서를 분석하기에 앞서 각 문서에 대한 중요 단어를 추출함으로써 특징 추출(**Feature Selection**)을 수행하고 관심 문서와 문서에 대해 선택된 단어를 이용하여 문서를 사용자의 관심 영역으로 분류하며 주요 키워드를 추출하여 사용자의 프로파일을 작성한다. 셋째는 사

용자의 프로파일을 이용하여, 사용자가 웹 브라우저에서 검색을 수행할 때 각 문서에 대한 링크중에서 사용자의 관심과 가장 유사한 정보 즉, 관심 문서를 사용자에게 제안한다. 이때 Webwatcher 는 에이전트가 추천한 링크에 대한 사용자의 반응과 사용자가 원하는 정보를 획득했는지의 성공 여부를 모니터링한다.

이렇게 하여 형성된 문서에 대한 피드백 정보에 기계 학습을 적용하며 지속적인 학습을 통해 사용자의 관심에 대해 더욱 명확해지고 사용자의 주요 관심 변화를 파악할 수 있다. 하지만 이것은 사용자 관심 문서를 시스템이 자동으로 추측하기 때문에 사용자에게 편리하기는 하지만 학습을 위한 관심 정보의 신뢰성이 낮다는 단점이 있다.

(3) Webby

Webby 는 IBM 도쿄 연구실에서 개발한 웹 에이전트의 일종으로 웹 브라우저를 모니터링하고 사용자의 선호도를 측정하여 그 결과를 CGI 서비스로 보여주는 웹 에이전트이다[31]. Webby 의 기능은 웹 브라우저를 사용하여 필요한 정보를 얻고자 하는 사용자에게 사용자가 선호하는 문서에 대한 선호도와 이 문서를 다시 참조하고자 할 때의 편의를 제공한다. Webby 는 웹 서버의 서비스 중 프록시 서버 서비스의 기능을 이용해서 사용자에게 대한 웹 브라우저의 행위를 모니터링하며, 프록시 서버를 이용하여 사용자가 방문하고자 하는 문서에 대한 위치(URL)를 추출해 낼 수 있다. 추출된 문서 위치는 간단한 Indexing 작업을 통해서 사용자의 컴퓨터에 저장되고, 이 정보는 사용자의 관심을 추출해내는 기초적 자료가 된다. 사용자의 관심을 추출해내는 방법은 단순히 문서에 대한 방문 횟수를 이용하는 것인데, 현재까지 사용자가 방문한 웹 페이지에 대한 문서의 수치에 대해 백분율로 관심도를 산출하며 이러한 사용자 방문 문서에 대한 백분율을 이용하여 사용자의 관심 분야를 예측한다.

4.4 멀티 에이전트 분야

(1) 협력 여과 에이전트

정보의 여과에 대한 연구와 여러 에이전트간의 정보 공유에 대한 연구에서 출

발한 것이 협력 여과 에이전트인데, 하나의 에이전트로서는 처리하기가 어렵고 복잡한 문제에 대해 이를 해결하기 위해 다른 에이전트의 도움을 받거나 도움을 줄 수 있다는 개념에서 연구되었다.

네트워크 기술의 발전으로 많은 사람들이 정보의 공유를 위한 장치들을 쉽게 접할 수 있게 되었으며, 이것은 다수의 사람들이 자신이 가지고 있는 정보를 쉽게 공유할 수 있음을 의미한다. 정보의 종류가 다양해지고 정보의 양도 증가할수록 사용자가 필요로 하는 정보를 찾기 위한 시간과 노력은 그만큼 많아지게 된다. 이것은 앞으로 정보의 양이 증가할수록 더욱 심각한 수준까지 이르게 될 것이다. 이러한 정보의 부담을 해결하기 위하여 여과에 대한 연구가 시작되었다.

여과는 일반적으로 사용자가 원하는 정보를 찾는 작업(Filtering In)과 사용자가 원하지 않는 정보의 제거(Filtering Out)의 두가지 의미를 갖고 있다. 이와 관련된 많은 응용 프로그램들은 여과를 수행하기 위하여 문서나 정보를 선택할 때 사용자가 요청한 정보에 해당하는 키워드의 Matching 을 통해 이루어지는 인지적 (Cognitive, Content-Based) 여과 방법을 많이 사용한다.

여과 시스템은 다음과 같은 몇가지의 일반적인 특징을 가진다.

- 정보 여과에서 다루는 문서나 정보의 종류는 비구조적(Unstructured) 또는 준구조적(Semi-Structured) 데이터에 대한 정보 시스템이다.
- 주로 문자 정보(Textual Information)를 다룬다.
- 많은 양의 데이터를 다룬다.
- 여과 응용 프로그램은 원격지 자원(Remote Source)으로부터 브로드캐스트 되는 데이터의 흐름으로부터 정보나 문서를 얻는다.
- 여과는 사용자 프로파일을 기반으로 하여 여과 작업을 수행한다.

이러한 특징이 있는 대부분의 여과에서 핵심이 되는 부분이 사용자 프로파일의 생성, 유지 및 관리이다. 에이전트의 개념을 여과에 도입하여 사용자로부터의 피드백과 사용자의 관심도를 분석하고 학습하여 사용자 프로파일을 더욱 정교하게 만들어 가게 되며, 사용자 프로파일은 여과의 정확성과 사용자의 만족도를 높이게 된다. 하지만 이러한 여과 에이전트는 몇가지의 문제점을 갖고 있다.

- 사용자 프로파일을 사용자의 선호도를 반영할 정도로 구축하고 이를 기반

으로 여과를 수행하기 위해서는 많은 학습시간이 요구된다.

- 사용자 프로파일이 사용자에게 대해 개인화되어 있지 않은 경우에는 효율적인 여과를 기대할 수 없고 단순한 키워드 Matching 을 통한 인지적 여과에 의존할 수 밖에 없다.
- 새로운 사용자나 새로운 분야에 대해 정보를 요구하는 경우, 즉 사용자 프로파일이 개인화되어 있지 않고 사용자의 선호도를 반영할 정도로 구축되어 있지 않은 경우에는 여과 에이전트는 처음부터 학습을 다시 해야 하는 문제가 발생한다.

이러한 문제들은 한 사용자를 위해 에이전트가 존재하기 때문에 발생하는 것이다. 이러한 에이전트가 다른 에이전트와 사용자의 프로파일을 기반으로 정보를 공유한다면 이러한 문제는 어느 정도 해소될 수 있을 것이다. 이러한 관점에서 협력 여과 에이전트가 연구되기 시작했다.

협력 여과 에이전트는 같은 주제에 대해 흥미를 가지는 공동체 (Community)내에 속해 있는 다른 사용자가 여과하고자 하는 문서나 정보에 대해 보여준 간접적인 피드백 즉, 문서나 정보를 저장하거나 관련된 키워드 등을 사용자가 입력하는 빈도수 등의 행동이나 직접적인 피드백 즉, 문서나 정보에 대한 유용한 정도를 숫자를 통해 등급을 결정하는 행동을 통해서 이 문서나 정보를 여과할 것인지를 결정하는 프로그램이라고 할 수 있다.

(2) 진료 예약 에이전트 시스템

이화여자대학교에서 구축한 진료 예약 에이전트 시스템인 ARMETI 는 종합병원에서 발생하는 진료 예약상의 문제를 자동화하고 지능적으로 해결하기 위한 멀티 에이전트 시스템이다[32].

순수 자바 코드로 구현된 진료 예약 시스템은 사용자 인터페이스 에이전트를 기본으로 하여 4 개의 기능 에이전트와 브로커로 구성되어 있으며 그림 4-8 과 같다. 기능 에이전트는 개인 일정 관리 에이전트, 의사 일정 관리 에이전트, 환자의 진료 예약을 관리하는 진료 예약 에이전트, 환자의 진료 상황과 입원실 예약을 담당하는 진료 에이전트가 각각의 기능을 수행하는 에이전트들이다. 브로커는 기능 에이전트

들이 요구에 맞는 작업을 수행하기 위하여 단위 에이전트들을 연결하는 'Match-Making' 역할을 한다. 각각의 에이전트는 각기 다른 특정한 일을 하지만 데이터베이스로 연결되어 있으며, 에이전트들간의 통신을 위해 상호 협력한다. 시스템은 사용자와 상호 교류를 위해 사용자 인터페이스 에이전트를 사용한다.

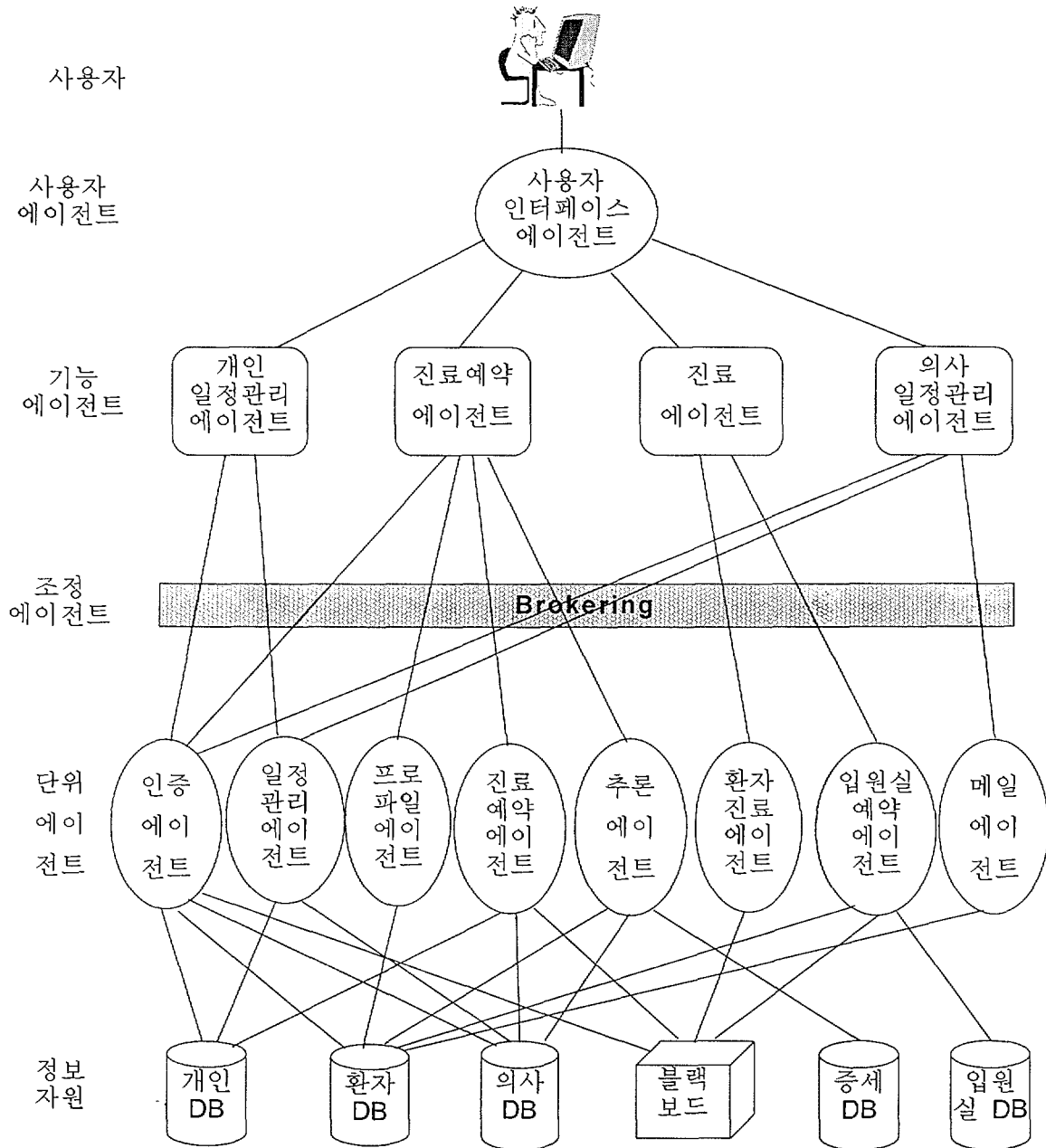


그림 4-8. 진료 예약 에이전트 시스템의 구조

5. 결론

본 논문에서는 사용자가 원하는 작업을 대신 수행하는 컴퓨터 프로그램인 에이전트에 대하여 살펴 보았다. 에이전트는 인공지능 분야에서 연구되기 시작하여 점차 세분화되어 독자적인 분야로 연구되고 있다. 이러한 에이전트에 대한 개념과 특성에 대해 살펴 보았으며, 발전 배경, 연구 동향과 응용에 대하여 살펴 보았다.

에이전트 기술의 성공적 보급을 수행하기 위해서는, 다른 환경이나 다른 종류의 에이전트 간의 유연한 정보 교환이 필수적이라고 할 수 있다. 이와 관련하여 에이전트에 기반한 응용 시스템의 효율적인 환경 구축을 위해서는 상호 이용성이 문제로 제기된다. 그러나 각각의 에이전트가 서로 다른 플랫폼을 바탕으로 서로 다른 목적으로 개발되었기 때문에 에이전트간의 이형질성이 과생되었으며, 이로 인해 에이전트간의 정보 교환에 많은 어려움을 주고 있고 표준화 작업 또한 지연되고 있다. 이를 해결하고 에이전트의 상호 이용성을 확보하기 위해서는 지속적으로 발전하는 네트워크의 기술에 대한 빠른 접목과 더불어 에이전트 간의 표준 통신 언어, 표준 온토로지, 기존의 소프트웨어와의 범용 인터페이스 등에 관한 국제적 표준화가 좀 더 이루어져야 한다. 또한 에이전트의 요소들에 대한 평가 기준, 단일 에이전트의 성능 평가 기준, 에이전트간의 비교 기준 등에 대한 연구가 필요하다.

1993년 Mosaic 이라는 웹 브라우저가 개발된 후 약 7년여만에 인터넷 정보의 홍수 시대를 맞이하게 된 추세를 감안할 때 앞으로 정보가 더욱 다양해지는 상황에서 에이전트는 좋은 해결책이 되어 줄 것이다. 그러기 위해서 에이전트는 사용자를 대신하여 작업을 자동적으로 수행하며, 다른 에이전트들과의 정보교환을 통하여 보다 효율적인 문제해결 방안을 스스로 수립하고, 계속적으로 변화하는 인터넷 환경에 스스로 적응하도록 개발되어질 것이다. 또한 에이전트는 정보검색과 전자상거래와 같은 네트워크 전반에 걸친 분야 뿐 아니라 전화메세지, 전자우편, 팩스 등의 온라인 메시지를 지능적으로 연결하고 정보의 여과 작업을 수행하는 메시징 분야에도 활용될 것으로 전망되며, 더 나아가서 향후에는 에이전트의 개발 방향에 맞추어 인터넷의 정보형태에도 영향을 줄 것으로 예상된다.

향후 이 연구를 바탕으로 원자력 자료의 정보화에 에이전트 시스템을 활용할 계획으로 있다.

참고문헌

- [1] 최중민, "인터넷 정보 가공을 위한 에이전트", 정보처리학회지 Vol.4 No.5, September 1997.
- [2] 김태훈, 최중민, "사용자 편의의 인터넷 정보검색을 위한 지능형 웹브라우저 에이전트", 정보과학회 논문지(B) 25 권 7 호, 1998.
- [3] 양재영, 최중민, "협동 에이전트 시스템", 전자공학회지 Vol.26 No.1, 1999.
- [4] 최중민, "에이전트 개요와 연구방향", 정보과학회지 15 권 3 호, 1997.
- [5] JiSook Chang, Sunhgmin Kim, Jin-Sook Choi, Jeon-Young Lee, "EARTS: Multi-agent Job Hunting System", HCI '98 학술대회 발표논문집. pp462-467, 1998.
- [6] 장혜진, "지식 기반 에이전트 시스템", 정보처리학회지 Vol.4 No.5, September 1997.
- [7] Joseph P. Bigus, Jennifer Bigus, "Constructing Intelligent Agents with Java", Wiley, 1997.
- [8] 최중민, 김준태, 심광섭, 장병탁 공역, "인공지능 - 지능형 에이전트를 중심으로", 사이텍미디어, 2000.
- [9] Walter Brenner, Ruediger Zarnekow, Hartmut Wittig, "Intelligent Software Agents", Sprinter, 1998.
- [10] 김인철, "계획 기반 에이전트 시스템", 정보처리학회지 Vol.4 No.5, September 1997.
- [11] 노현철, 이근배, 이종혁, "KQML 에 기반한 멀티 에이전트 통신 환경", 정보과학회지 15 권 3 호, 1997.
- [12] 민병의, 박상규, 장명욱, 이광로, 황승구, "분산 환경 기반 개방형 에이전트 구조", 정보과학회지 15 권 3 호, 1997.
- [13] 김평중, 윤석환, "이동 컴퓨팅을 위한 이동 에이전트 시스템", 정보처리학회지 Vol.4 No.5, September 1997.
- [14] 백순철, 최중민, 장명욱, 박상규, 임영환, "이형 분산 환경에서 에이전트들간의

이형성을 극복하기 위한 멀티 에이전트 기반 구조", 정보과학회, 제 2 권, 제 1 호, 1996

[15] Flavio Moreira de Oliveira, "Advances in Artificial Intelligence", Springer, 1998.

[16] 소영준, 박영택, "사용자 프로파일 기반 웹 에이전트", 정보과학회 논문지(B) 27 권 3 호, 2000.

[17] Bruce Krulwich, Chad Burkey, "The InfoFinder Agenbt: Learning User Interest through Heuristic Phrase Extraction", AgentSoft Ltd, Andersen Consulting LLP, 1995.

[18] 장지숙, 김성민, 최진숙, 이진영, "분산 환경에서의 이동 에이전트를 이용한 구인-구직 시스템", 한국정보과학회 학술발표논문집, 제 25 권 1 호, 1998.

[19] 김성민, 이동하, 장지숙, 최진숙, 이진영, "분산환경에서 능동적이고 적응성이 있는 지능형 에이전트간의 협상", 정보과학회논문지(B) 26 권 1 호, 1999.1

[20] 김희정, 박승수, "지능형 에이전트를 기반으로 한 일정 관리 시스템", 한국정보과학회 학술발표논문집, 제 24 권, 1 호, 1997.

[21] WiseWire, <http://www.urlwire.com/newsarchive/091797A.HTM>

[22] 권혁찬, 이재흠, 손수호, 유관중, "논문수집을 위한 에이전트 시스템", 한국정보과학회 학술발표논문집, 제 25 권 1 호, 1998.

[23] 손봉기, 남기범, 이진명, "BBS 의 불건전 기사 검색을 위한 에이전트 시스템의 설계", 한국정보과학회 학술발표논문집, 제 26 권 1 호, 1999.

[24] 김선욱, 이재필, 김기태, "사례기반추론을 이용한 Catalog Agent 의 설계", 한국정보과학회 학술발표논문집, 제 26 권 1 호, 1999.

[25] 백혜정, 박영택, 윤석환, "사용자 관심도를 이용한 웹 에이전트", 정보처리학회지 Vol.4 No.5, September 1997.

[26] 신봉기, 김영환, "웹 에이전트", 정보과학회지 15 권 3 호, 1997.

[27] 최용석, 유석인, "신경망 기반 멀티 에이전트 웹 정보검색 시스템", 정보과학회 논문지(B) 26 권 5 호, 1999.

[28] G. Salton and M. McGill, Introduction to Modern Information Retrieval, MacGraw-Hill, New York NY, 1983.

- [29] R. Armstrong, D. Freitag, T. Joachims, T. Mitchell, "WebWatcher: A Learning Apprentice for the World Wide Web", 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, March 1995.
- [30] 이대령, 박영택, "정보추출을 이용한 기계학습기반의 웹에이전트", 한국정보과학회 학술발표논문집, 제 23 권, 1 호, 1996.
- [31] Webby, http://www.ibm.com/ibm/hci/exhibits/intelligent_interfaces/webby.html
- [32] 한혜란, 박승수, "웹 상에서 에이전트를 기반으로 한 진료 예약 시스템의 설계 및 구현, 한국정보과학회 학술발표논문집, 제 26 권 1 호, 1999.

서 지 정 보 양 식					
수행기관보고서번호	위탁연구기관보고서번호	표준보고서번호	INIS 주제코드		
KAERI/TR-1591/2000					
제 목 / 부 제	에이전트의 발전 및 응용에 관한 연구				
주 저 자	박 선 희 (한국원자력연구소, 열수력안전연구팀)				
공 저 자	손 재 민 (한국원자력연구소, 하나로이용기술개발팀)				
발행지	대 전	발행 기관	한국원자력연구소	발행일	2000년 6월
면 수	57p	도 표	유 (0), 무 ()	크 기	A4
참고사항					
비밀여부	공개 (0), 대외비 (), _____ 급 비밀		보고서종류	기술보고서	
위탁연구기관			계약번호		
초 록	<p>에이전트(Agent)의 연구는 초기의 인공지능 분야의 한정된 연구에서 고도의 분산 복합 시스템을 설계하여 구현하는 실용적인 소프트웨어 개발에 이르기까지 다양하게 적용되고 있다. 인공지능의 세분화된 연구 분야 중 분산 인공지능이라는 새로운 연구 분야가 형성되면서 에이전트를 토대로 문제를 해결하려는 시도가 시작되었다. 이와 관련하여 사용자가 원하는 작업을 쉽게 수행할 수 있도록 편리함을 제공해 주는 지능형 시스템(Intelligent System)에 대한 요구가 증대되어 왔으며, 최근에 와서 이 문제에 대해 인터넷에의 접근이 시도되면서 사용자가 원하는 작업을 자동적으로 대신 해결하고 처리하여 주는 다양한 형태의 에이전트가 개발되었다.</p> <p>본 논문에서는 이러한 에이전트와 관련된 시스템들의 기본 개념과 연구 방향 및 그 응용에 대해 분석하였으며, 에이전트가 등장한 배경, 필요성, 기본 개념 및 특성 등 에이전트의 개요와 구성 요소에 대하여 면밀하게 분석하였다. 그리고 현재까지 수행되어 온 에이전트에 대한 국내외의 연구 현황을 분석하여, 지능 에이전트의 개발과 관련된 특성에 대해 기술하였다. 또한 에이전트가 실생활에 적용될 수 있는 분야에 대하여 보편화된 에이전트의 응용 및 사례에 대해 연구하였고, 특히 지능 에이전트의 기능 및 서비스에 대해 분석하였다.</p> <p>에이전트 기술의 적용 분야는 웹 사이트(Web Site)에서 정보의 검색 뿐 아니라 인터넷 정보의 가공, 전자상거래, 산업체 및 의료 분야 등과 같이 점차 다양해지고 있다. 본 연구는 이러한 다양한 에이전트에 대해 전반적인 개념을 분석하고 비교해 봄으로써 지능 에이전트에 대한 이해와 향후 이 분야의 연구에 대한 방향 정립에 도움이 되고자 한다.</p>				
주제명 키워드	에이전트, 인공지능, 분산 인공지능, 에이전트의 구성 요소, 에이전트 언어, 에이전트의 분류, 인터넷, 정보 검색, 웹 사이트				

BIBLIOGRAPHIC INFORMATION SHEET					
Performing Org. Report No.	Sponsoring Organization Report No.		Standard Report No.	INIS Subject Code	
KAERI/TR-1591/2000					
Title/Subtitle	A Study on Development and Applications of Agents				
Main Author	Park Sunhee (KAERI, Thermal-hydraulic Safety Research)				
Author	Sohn Jae Min (KAERI, HANARO Utilization Technology Development Team)				
Pub. Place	Taejon	Pub. Org.	KAERI	Pub. Date	June 2000
Page	57p	Fig. & Tab.	Yes (0), No ()	Size	A4
Note					
Classified	Open (0), Outside (), _____ Class Doc.		Report Type	Technical Report	
Sponsoring Org.			Contract No.		
Abstract	<p>An agent study is variously applied from the restricted research of an early AI field to the practical SW development of the design and implementation of a high level, distributed hybrid system. Among the detailed research fields of AI, making a new research field called Distributed AI, the trial of resolving a problem based on an agent, has begun. Related to it, the requirement of an intelligent system to offer convenience to easily do a job which the user wants, with the recent trial of an approach for internet agents of various types, were developed to be performed automatically and the agent anticipate the user's need.</p> <p>In this paper, the basic concepts regarding an agent system, research direction and applications were analyzed and outlined. The consisting elements were also analyzed, such as background, needs, basic concepts and properties of the agent. The agent's research status was recently analyzed, and the development and related property of an intelligent agent were written. Application and the instance of a generalized agent about applicable fields in the real world were studied, functions and services about intelligent agents were analyzed.</p> <p>The applicable field of an agent technique becomes various not in information retrieval on a web site, but in the processing of internet information, and in the electronic commerce, industry and medical fields. Through analyzing and comparing the general concept of various agents, this study will aid in establishing a future direction of research in this field.</p>				
Subject Keywords	Agent, Artificial Intelligent, Distributed AI, Component of agent, Classification of agent, Information retrieval, Web site				