

웹기반 협력 학습을 위한 멀티에이전트간의 통신에 관한 연구

이철환 · 한선관

인천교육대학교 컴퓨터교육과

chlee56@mail.inue.ac.kr · fish@compedu.incheon.ac.kr

요약

본 연구는 웹 기반 협력학습을 위한 시스템에서 학습자의 학습을 돕기 위한 멀티에이전트간의 통신에 관한 연구이다. 우선, 에이전트 시스템에 대한 전반적인 내용과 특징을 고찰하였고 에이전트 상호간의 통신인 KQML에 대하여 살펴보았다. 또한 협력학습을 위한 에이전트 기반의 시스템 구조와 에이전트간의 상호 통신 방법을 제시하였다. Java 언어를 이용하여 협력학습 시스템을 설계 구현하였으며 실험을 통하여 에이전트간의 통신에 의한 협력학습 시스템의 효율성을 고찰하였다.

A Study of Communication between Multi-Agents for Web Based Collaborative Learning

Chul-Hwan Lee · Sun-Gwan Han

Inchon National University of Education, Dept. of Computer Education

Abstract

The purpose of the paper is communication between multi-agents for student's learning at web based collaborative learning. First, this study investigated the whole contents and characteristics of an agent system and discussed KQML, communication language between multi-agents. Also, we suggested architecture of an agent based system for collaborative learning and interaction method between agents using KQML. We designed and implemented collaborative learning system using Java programming language, and we also demonstrate the efficiency of collaborative learning system by communication between multi-agents through experiments.

1. 서론

오늘날 웹을 기반으로 하는 인터넷 환경의 발달로 교육의 환경에 커다란 변화를 가져오고 있다. 특히 정보와 지식의 폭발적 증가로 이를 검색하고 습득하기 위한 기술과 방법들을 요구하고 있다. 기존의 교육 환경에서 이러한 어려움을 극복하는 방법으로 가상 공간 속에서 다양한 학습자들이 협력적인 학습을 통해 지식의 공유와 학습의 시너지 효과를 찾는 방법이 활발하게 연구가 진행되고 있다.

90년대 중반에 들어서면서 에이전트에 대한 관심과 필요성이 증대되고 있는데, 그 이유 중에 두드러진 것은 인터넷의 폭발적인 이용 증가로 인하여 좀더 사용자에게 정보를 쉽게 접할 수 있도록 편리함을 제공하는 기능이 요구되었기 때문이다. 즉, 인터넷의 정보 홍수 속에서 원하는 정보를 정확하게 원하는 시간에 얻기란 쉬운 일이 아니며, 따라서 이러한 작업을 대신해주는 에이전트의 역할이 점점 커지고 있다. 교육 부문에서도 이러한 에이전트의 적용과 역할이 절실히 요구되고 있는 실정이다. 또한 국가 시책으로 추진되고 있는 초고속 정보통신망과 연계하여 교육적인 정보 검색뿐 아니라 온라인 쇼핑 등의 전자상거래나 메시징과 같은 이동 컴퓨팅 분야에서도 에이전트의 필요성은 대두되고 있다[12].

에이전트를 한마디로 표현하면 사용자를 대신해서 사용자가 원하는 작업을 자동적으로 해결하여 주는 소프트웨어라고 할 수 있다. 지능형 에이전트는 인공지능 분야에서 오래 전부터 연구되어 온 개념으로, 사실상 인공지능의 최종 목표가 사람과 유사하게 생각하고 행동하는 지능을 가진 에이전트의 개발이라고 보아도 크게 벗어나는 표현은 아닐 것이다.

여러 에이전트의 분산 협동 처리를 위해서는 에이전트간 통신이 필수적이다. 에이전트간 통신의 목적은 정보나 작업처리의 공유와 교환에 있다. 즉, 모든 정보를 포함하고 어떠한 일도 혼자서 처리할 수 있는 완전한 에이전트를 개발하는 것은 불가능하므로 자신이 가지고 있지 않는 정보를 접근하거나 자신이 해결하지 못하는 작업을 처리하기 위해 다른 에이전트에게 도움을 청하는 단계가 반드시 필요하게 되는데, 이 도움을 청하는 방법으로 에이전트간 통신을 이용하는 것이다. 에이전트간 통신을 위한 가장 큰 걸림돌은 각 에이전트가 가지고 있는 이형질성(heterogeneity)이다[8]. 즉, 각각의 에이전트는 서로

다른 사람에 의해, 서로 다른 시기에, 서로 다른 플랫폼을 바탕으로, 서로 다른 목적으로 개발되었기 때문에 이들간의 통신을 위해서는 상호 이해 가능한 언어와 프로토콜이 필요하다.

이러한 노력의 대표적인 예가 DARPA의 Knowledge Sharing Effort의 일환으로 개발한 KQML (Knowledge Query Manipulation Language)이라고 할 수 있다[10].

인터넷상에서 학습자들은 자신이 원하는 협력학습 시스템을 찾기란 매우 어려우며 또한 기존의 협력학습 시스템은 학습자 중심이 아닌 운영 시스템 중심으로 비계획적인 방법으로 진행하여 학습의 효율성을 떨어뜨리며 학습자의 학습 연계성 및 학습자의 능력 등을 고려하지 않아 결손학습을 가져오게 된다.

따라서 본 논문은 2장에서 이러한 협력 학습의 문제점을 보완하고 지능적이고 자율적인 학습을 가능하게 하는 에이전트 시스템에 대한 전반적인 내용과 특성 등을 고찰하여 보고, 3장에서 협력학습에 맞는 에이전트 기반의 시스템을 설계하여 그 방안을 제시하였다. 또한 KQML을 이용하여 멀티 에이전트간의 상호 통신을 통하여 적절한 학습 에이전트 서버를 찾아 학습자들이 원하는 시간에 학습을 가능하게 해 줄 수 있는 시스템을 구현하였다. 4장에서는 실험으로 에이전트간의 상호 통신을 통해 본 시스템의 효율성을 입증하고 결론 및 향후 연구에 대하여 고찰하였다.

2. 이론적 고찰

2.1. 에이전트의 정의

에이전트란 사용자의 컴퓨터 업무를 도와주거나 대신해 주는 프로그램으로 정의 내릴 수 있는데 사전적 의미로 대행자, 대리자, 영업사의 대표, 특히 돌아다니는 세일즈맨 등을 뜻하며, AI적인 의미에서는 분산 환경 하에서 작업을 수행하는 지적인 특성을 갖는 응용프로그램으로, 사회과학적 의미에서는 하나의 일을 상호 협력해서 해결하는 응용프로그램으로 본다. 또한 HCI적인 의미에서 보면 사람의 생각을 인지해서 사람의 요구사항을 처리해 주는 프로그램이라고 할 수 있다[11].

따라서 에이전트는 사용자나 다른 시스템의 제어를 받지 않고도 자신의 행동과 상태를 스스로 제어

할 수 있고, 주변 상황을 인지하여 변화가 발생하였을 경우 이에 대처할 수 있으며, 에이전트들간의 통신 수단을 통하여 다른 에이전트와 상호 협력하는 등의 특성을 포함하고 있다. 이처럼 에이전트는 지능적인 성격을 가지고 있으며, 상호 독립적인 프로그램의 개념보다는 상호 보완적인 존재로서 각각의 에이전트가 갖는 단점들을 극복하고 협력함으로써 복합적인 목표를 달성할 수 있다.

이러한 에이전트의 접근법은 전자 우편을 관리하고 인터넷상의 과도한 정보를 여과하는 프로그램 등의 지원 도구로, 시간 소비적이며 반복적으로 발생하는 업무를 대행하는 개인 비서 차원의 도구 등으로 적용되어 왔다[2].

에이전트에 대한 정의는 절대적인 것은 아니지만 어떤 소프트웨어가 에이전트인지 아닌지를 나름대로 판단하는데 좋은 지표가 될 수 있다. 물론 판단 대상이 에이전트다 또는 아니다라고 단정짓는 것은 좋은 접근 방향이 아닌 것 같고, 다만 위에서 정의된 항목에 얼마나 많이 부합되는가에 따라 에이전트의 역할을 좀 더 충실히 한다고 말할 수 있겠다.

에이전트의 정의에서 나타나 듯이 기본적으로 에이전트라고 불리기 위해서 필요한 몇 가지 특성을 추출해낼 수 있는데 대표적으로 자율성 (autonomy), 지능 (intelligence), 이동성 (mobility), 사교성 (social ability) 등을 들 수 있다[1].

자율성(autonomy)은 에이전트와 다른 일반 소프트웨어를 구별해주는 가장 핵심이 되는 특성이라고 볼 수 있는데, 사용자나 다른 에이전트의 직접적인 지시나 간섭 없이도 스스로 판단하여 행동하는 성질을 의미한다. 일반 프로그램이 수동적인 것에 비해 에이전트는 자율성을 가짐으로써 능동적으로 작업 수행을 진행하게 되며, 이를 위해 에이전트는 수행 동작이나 내부 상태변화 등에 대한 제어권도 가지고 있다. 자율성은 사용자로 하여금 상위단계 목적 (high-level goal)에 집중을 하게 하고 그 목적을 달성하기 위한 세부 질 차등은 에이전트가 맡게 된다.

지능성(intelligence)은 지식 베이스와 추론 능력을 갖추고 사용자의 의도를 파악하여 계획 (planning) 을 세우고 학습 (learning) 을 통하여 새로운 지식을 스스로 터득하는 성질로 인공지능에서 많이 연구된 결과를 근거로 한다. 지능은 사실상 자율성과 밀접한 관계를 가지게 되며, 지능을 바탕으로 에이전트는 같은 작업이라도 계획과 경험을 통해 더

나은 효과를 기대할 수 있다.

이동성(mobility)은 사용자가 요구한 작업을 현재의 호스트에서 수행하지 않고 실제 그 작업을 처리하는 호스트로 이동시켜 수행함으로써 수행의 효율을 높이고 네트워크 부하를 줄이는 효과를 가져온다. 이동성은 기존의 클라이언트/서버의 개념과는 판이하게 다른 개념으로 서버의 내용을 클라이언트를 통해 전송 받아 정보를 얻거나 작업수행을 하는 것이 아니고 클라이언트가 필요로 하는 작업을 위해 에이전트를 서버로 보내어 수행을 시킨다. 이동성은 특히 인터넷의 보급으로 컴퓨터 네트워크를 통해 제공되는 정보의 수가 급증하면서 그 중요성이 강조되고 있으며 원격 통신 시에도 통신 라인이 항상 접속되어 있을 필요가 없기 때문에 무선 이동 통신을 위한 작업수행 환경에서 큰 효과를 낼 수 있다.

사교성(Social Ability)은 에이전트간 통신 능력을 의미한다. 즉, 하나의 에이전트로는 처리하지 못하는 작업의 수행을 위해 다른 에이전트의 도움을 필요로 할 때 에이전트간 메시지 교환에 의존하게 된다. 에이전트 통신 언어 (agent communication language) 를 이용한 에이전트간 통신은 메시지 전달 (message passing) 이나 공유 메모리 (shared memory) 방법을 이용할 수도 있고 다른 에이전트의 메소드 (method) 를 불러 수행하기도 한다.

이러한 기본적인 특성 외에도 에이전트가 가지는 성질은 여러 가지 있는데 그 중에서 환경 변화에 대하여 반응할 수 있는 반응성 (reactivity), 틀린 정보를 주고받지 않는 정직성 (veracity), 그리고 반드시 목적을 달성하는 방향으로 작업을 수행한다는 이성적 행동 (rationality), 단순히 환경에 응답하는 것이 아니고 자신의 목적을 성취하기 위한 행동들을 자발적으로 제시하는 선행성(Pro-activeness) 등이 있다.

2.2. 에이전트의 구성요소와 유형

하나의 에이전트는 기본적으로 에이전트 엔진 (engine), 영역지식 (domain knowledge), 통신 모듈 (communication module) 등으로 구성된다. 에이전트 엔진은 에이전트 생성과 작업수행, 에이전트 종료 등의 일련의 작업을 조정하기 위한 제어 지식과 추론 능력 등을 가진다. 영역 지식은 특정 응용 분야의 작업 수행에 필요한 지식으로써 에이전트의 역할을 특

정지어주는 부분이다.

에이전트는 생성 시에 자신의 영역지식과 관계된 작업능력을 공개함으로써 다른 에이전트와의 작업 공유를 시도한다. 에이전트 통신은 다른 에이전트와의 메시지 교환을 담당하는데 대부분 자신이 해결하지 못하는 문제에 대하여 다른 에이전트의 도움을 청하는데 많이 이용된다[7].

에이전트 통신의 주된 목적이 다른 에이전트와의 협동을 통한 문제 해결이라고 할 때 몇 가지 문제점이 발생한다. 즉, 다른 어떤 에이전트가 존재하는 지, 또는 다른 에이전트가 해결할 수 있는 능력이 무엇 인지를 알고 있어야 한다는 것이다. 그렇지 않고는 누구에게 도움을 청할 지 판단할 수 없기 때문이다.

또 다른 문제점은 모든 에이전트끼리 서로 연결되어 있어야 하기 때문에 발생하는 네트워크 사용의 비용과 성능 저하를 들 수 있다. 이런 문제점을 해결하기 위한 접근 방향이 바로 멀티 에이전트 시스템이다.

분산 환경에서의 에이전트 구조하에서는 크게 3가지 유형의 에이전트가 존재하는데, 인터페이스 에이전트(Interface agent), 작업 에이전트(Task agent), 그리고 정보 에이전트(Information agent)가 그것이다[12].

또한 에이전트를 그 규모별로 분류하면 다음과 같이 3가지로 분류할 수 있다[11].

단독 에이전트(single agent)는 보조 에이전트와 같이 하나의 에이전트로서 고유 분야에서 사용자에게 서비스 제공하는 것으로 보조 에이전트 시스템 (Assistant Agent System), 사용자 인터페이스 에이전트 시스템(User Interface Agent System), 지능형 에이전트 시스템(intelligent Agent System)등이 이에 속한다.

협력 에이전트(cooperating agent)는 몇 개의 에이전트간에 상호 협력을 통하여 서비스 제공하는 에이전트로 다중 에이전트 시스템(Multi-Agent System)이라고도 한다.

에이전트 사회(society of agent)는 협력 에이전트 개념보다 더 광의의 집단을 지칭하며, 이동 에이전트 시스템(Mobile Agent System)이 이에 속하며 다중 에이전트 시스템들간의 연결 공동체라고 보면 이해가 쉽다.

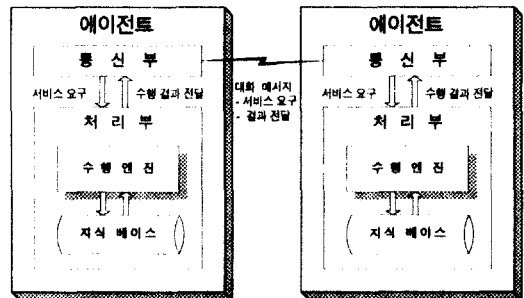
현재 연구와 개발이 활성화 된 부분은 다중 에이전트 시스템, 이동 에이전트에 중점을 두고 있으며

많은 시스템들이 상용화되어 사용 중이다.

2.3. 멀티 에이전트 시스템과 KQML

멀티 에이전트 시스템은 하나의 에이전트로 해결하지 못하는 복잡한 문제의 해결을 위하여 여러 에이전트간의 협동이 필요하게 되었고, 이를 효과적으로 수행하기 위해 제안되었다. 멀티 에이전트 시스템은 여러 응용 에이전트 외에 조정자 에이전트 (coordinating agent, facilitator)라는 중재자를 통해 메시지의 전달과 각 에이전트의 제어를 수행하게 된다[6]. 이 경우 모든 응용 에이전트의 통신 메시지는 조정 에이전트를 통해 다른 에이전트로 전달된다. 따라서 각 응용 에이전트는 조정 에이전트와만 연결된다.

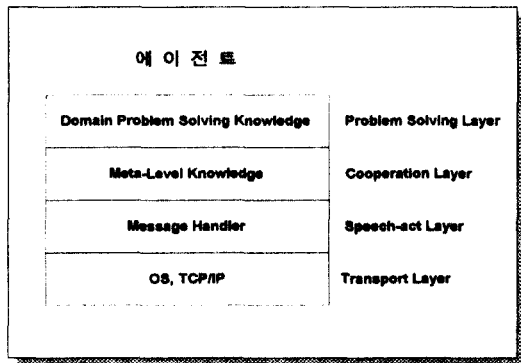
각 에이전트가 어떤 처리 능력이 있는지에 대한 정보는 응용 에이전트 생성시에 조정 에이전트로 등록되며, 조정 에이전트는 이를 바탕으로 통신 메시지를 해당 응용 에이전트에게 보낸다. 응용 에이전트는 다른 에이전트가 어떤 일을 할 수 있는 지에 대하여 알 필요가 없고, 단지 필요시 통신 메시지를 조정 에이전트에게 보내어 맡기면 된다. 이와 같이 조정 에이전트는 응용 에이전트의 수행 능력을 통해 에이전트의 실체를 알 수 있는 에이전트 내임 서버의 역할을 수행하는 것이다. 멀티 에이전트의 일반적으로 연합 에이전트 구조 (federated agent architecture)를 가지며, 하나의 멀티 에이전트 시스템에 하나의 조정자와 여러 에이전트가 계층구조를 이루고, 조정자는 다른 멀티 에이전트 시스템의 조정자와 메시지를 주고받는다[7].



[그림 1] 멀티 에이전트의 구조

멀티 에이전트에서 발생하는 중요한 문제는 각

응용 에이전트의 이형질성이다. 물론 멀티 에이전트 기반구조와 응용 에이전트들을 처음부터 같이 개발했다면 이러한 문제가 없겠지만 기존의 응용 프로그램을 바탕으로 에이전트 기능을 추가한 형태의 에이전트인 경우는 기존 응용 프로그램의 특성에 따라서 다른 형태를 지니게 된다. 이와 같은 이형질의 응용 에이전트간의 통신을 위해서는 표준화된 메시지 형태와 전달 프로토콜이 있어야 한다[3].



[그림 2] Layered Model of an Agent

대표적인 에이전트간 통신언어로 미국 국방성의 DARPA(Defense Advanced Research Projects Agency)산하 Knowledge Sharing Effort 그룹에서 만든 KQML(Knowledge Query And Manipulation Language)이 있다[5]. KQML은 화행 이론(Speech-act Theory)에 기반을 두고 일차 술어 논리(First-Order Logic) 형태를 갖는 메시지에 대한 문법적 구조와 어휘를 제공할 뿐만 아니라 통신 규약을 포함하고 있으며, 에이전트나 프로그램이 정보나 지식을 교류할 수 있도록 기반 구조를 제공하고 있는데, KQML의 통신 메시지는 통신 계층 (communication layer), 메시지 계층 (message layer), 내용 계층 (content layer)의 3단계로 나누어지고, 메시지 계층에서 표현된 언어의 종류와 영역 온톨로지(domain ontology) 정보를 기술해주면 실제 내용 계층에는 서로 다른 형태로 표시되었더라도 서로 이해할 수 있고 통신이 가능하게 되는 것이다[10].

다음은 3가지 계층을 모두 나타내는 KQML 메시지의 간단한 예를 보여주는데, learner라는 이름을 가진 에이전트가 학습 정보를 제공하는 learning-server라는 이름을 가진 에이전트에게 학습시간을

묻는 KQML 메시지이다.

```
(ask-one
 :sender learner
 :receiver learning-server
 :reply-with message-id01
 :language LPROLOG
 :ontology Collaborative-Learning
 :content ((Subject Math)
           (Time learning ?T)))
```

[그림 3] KQML 메시지의 형태

:receiver, :sender, :reply-with 인자는 통신 단계에서의 정보를 기술하며, 수행문의 이름과 :language, :ontology 인자는 메시지 단계의 정보를 기술하고, :content 인자는 내용 단계를 정의한다. :receiver, :sender 인자에는 메시지를 송수신하는 에이전트의 이름을 기술하고, :reply-with 인자에는 메시지의 일련 번호를 기술한다.

:ontology 인자를 통해서 두 에이전트는 학습 시간에 대해서 메시지를 교환하는 것을 알 수 있으며, :language 인자는 :content 인자를 구성하는 언어를 기술하고, :content 인자는 LPROLOG 문법의 수학 학습 시간에 대한 질의어라는 것을 알 수 있다.

3. 협력 학습 시스템과 멀티 에이전트

3.1 협력 학습 시스템

웹 기반의 학습 시스템은 다양한 정보와 지역적인 문제를 극복할 수 있음에도 불구하고 활용 측면에서 많은 문제점을 지니고 있다.

학습자가 효율적인 학습 정보를 찾기가 매우 힘들다는 것이 가장 큰 문제점으로 지적된다. 웹 기반 학습을 제공하는 다양한 업체와 개인 홈페이지들은 학습의 내용과 방법 등을 감추고 가입을 권유하여 이익의 대상으로 보고 있기 때문에 각각 떨어져 있고 예측하기 어려운 학습 정보들이 인터넷의 장점인 다양한 정보와 지식의 공유를 막고 있다. 물론 다양한 검색 엔진이나 교육용 포털 사이트들이 많이 존재하여 학습자들이 필요한 학습 홈페이지에 접근할 수 있지만 검색엔진에서 찾은 내용이 정말 자신의 학습에 도움이 되는지 직접 접근해야 알 수 있기 때문에 학습을 하기도 전에 흥미를 잃어 학습의 효과를 감

소시키고 있다. 따라서 이를 자율적이고 지능적으로 찾아 줄 수 있는 새로운 접근이 필요하다.

다음으로 지적되고 있는 문제가 다양한 학습자들이 동시에 협동 학습할 수 있는 환경이 부족하다는 점이다. 대부분의 교육용 콘텐츠들은 정적인 교육의 내용만을 제시하고 있기 때문에 현장 교실에서 볼 수 있는 다양한 학습의 방법을 적용할 수가 없다는 점이 문제가 되고 있다. 일반적으로 웹에서 많이 사용되는 채팅(Chatting)프로그램의 경우 학습에서 그 효용성이 매우 뛰어나도 불구하고 잡담과 시간 보내는데 주로 사용되고 있다. 이는 같은 주제와 흥미를 가진 학습자들이 모일 수 있는 환경이 조성되지 않았기 때문이며, 같은 주제와 흥미를 가진 학습자들이 같은 홈페이지에 동일한 시간에 접속하여 협력할 수 있는 시스템을 구축하여야 한다.

다음으로 웹 콘텐츠가 자주 변하고 사라지며 생겨난다는 것이다. 복사(Mirroring)의 방법으로 기존의 자료를 보관할 수 있지만 이것은 개인 학습자들에게는 그렇게 쉬운 일은 아니며 이를 관리해줄 수 있는 특수한 학습 서버가 필요하며 이러한 학습자들을 서버와 연결해 줄 수 있는 지능적인 시스템이 필요하다[13].

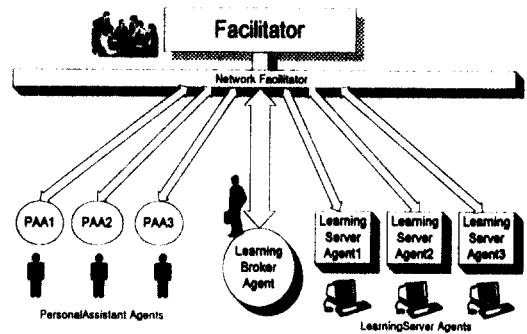
이러한 문제를 적절하게 해결하여 줄 수 있는 시스템이 바로 멀티 에이전트를 이용한 시스템이며, 에이전트의 구조로 볼 때 전체 에이전트를 관리하고 조정해 줄 조정자 에이전트(Facilitator)가 필요하며, 개인 학습자 에이전트(Personal Assistant agent)와 학습 서버를 찾아 줄 학습 브로커 에이전트(Learning Broker agent), 학습을 제공하는 학습서버 에이전트(Learning Server agent)가 필요하게 된다.

이러한 에이전트간의 통신 언어는 지적인 특성으로 인하여 KQML의 구조가 가장 효율적이며 다음절에서 이 시스템의 기본 구조를 제안, 설계한다.

3.2 협력학습 시스템의 기본 구조

본 연구에서 설계한 지능형 멀티에이전트 기반 협력학습 시스템은 [그림 4]와 같이 에이전트 사회의 원활한 유지를 위한 조정자(Facilitator)와 개인 협력 학습 시간 결정과 학습서버를 찾아줄 개인 비서 에이전트(Personal Assistant Agent; PAA), 학습을 제공하는 학습 서버 에이전트(Learning Server Agent)와 학습 서버와 학습자를 연결해 주는 학습 브로커

에이전트(Learning Broker Agent)로 구성되며 각 에이전트의 기능은 다음과 같다.



[그림4] 협력학습 시스템의 구조

3.2.1 조정자(Facilitator)

KQML을 통신언어로 사용하는 멀티에이전트 시스템에서는 조정자라는 특별한 에이전트가 필요하다. 이 에이전트는 에이전트간 네트워크 환경과 수행문을 용이하게 전달하는 것을 관리한다. 멀티에이전트 시스템에서 새로 생긴 에이전트는 조정자에게 등록을 해야 한다. 이것은 조정자가 관리하고 있는 에이전트 사회에 새로운 에이전트의 생성을 알리는 것이다. 이로써 새로 생긴 에이전트가 제공할 수 있는 서비스를 다른 에이전트에게 알리고 새로 생긴 에이전트가 원하는 서비스를 다른 에이전트로부터 제공할 수 있다.

본 연구에서 설계한 멀티에이전트 시스템에서 개인 비서 에이전트는 에이전트의 이름과 주소를 조정자에게 등록하고 사용자의 협력 학습을 대행하는 작업과 학습자의 학습 일정에 대한 정보를 제공할 수 있다는 것을 에이전트 사회에 알려야 하며, 학습 서버 에이전트도 마찬가지로 에이전트의 이름과 주소를 조정자에게 등록하고 학습자에게 협력학습이 가능하다는 것을 에이전트 사회에 알려야 한다.

3.2.2 개인 비서 에이전트(Personal Assistant Agent)

개인 비서 에이전트는 학습자의 정보를 이용해 학습자의 일반적인 학습 일정을 파악하고, 학습자의 요구에 따라 자율적이며 능동적으로 협력학습이 가능한 시간과 서버를 찾아 준다. 가장 먼저 조정자에게 등록을 하여 자신의 존재를 알리고 필요한 정보를 주고받게 된다. 학습자의 요구에 적절한 서버가 없을 경우 사용자의 선행 지식이나 경험을 추론 엔

진을 이용하여 가장 적합한 학습 서버를 찾아 주거나 학습자의 시간에 맞도록 학습 시간을 변경하게 된다.

3.2.3 학습서버 에이전트(Learning Server Agent)

실제적인 학습 내용을 가진 학습 서버로 자신의 일정을 가지고 학습자들을 모집하게 되며 학습자의 모집은 브로커 에이전트에게 위임하여 다양한 학습자들이 협력학습에 참여하면 다양한 학습 방법을 통해 협력학습을 하게 된다.

3.2.4 학습 브로커 에이전트(Learning Broker Agent)

가장 먼저 조정자에 등록하여 조정자에 등록되어 있는 학습서버 에이전트를 확인하게 되며, 개인 학습자 에이전트를 모집하여 다양한 학습자들의 협력학습에 대한 여러 가지 요구를 받게 되어 적절한 학습 서버를 찾아 학습자와 연결시켜 주는 역할을 하게 된다. 즉, 학습자들은 전문적인 학습서버를 찾기가 힘들기 때문에 가장 효율적인 학습 서버를 대신 찾아서 연결해 주는 역할을 한다.

3.3 멀티 에이전트간의 협력 학습 시나리오

현실의 세계에서 볼 때 어떤 학습자가 컴퓨터 학습을 받고자 한다. 이 학습자는 지금 웹기반 협력학습이 어디에서 이루어지고 있는지 모르고 있으며, 또한 어떤 내용으로 언제 학습을 하는지도 모르고 있다. 다만 학습자가 원하는 시간과 장소에서 수준에 맞는 다양한 학습자와 적절한 학습 내용을 가지고 웹기반 협력학습을 하려고 한다.

또한 학습을 서비스하는 교사의 입장에서도 학습에 대한 내용과 방법만 알고 있으며, 교사 개인적으로 학습자들을 모집하는 방법을 모르며, 모집할 시간 또한 부족하다.

바로 이러한 학습자와 교사를 연결해 주는 것이 바로 학습 중개자이다. 현실세계의 예를 들면 교육청에 등록되어 있는 학원이나 학교, 교사들을 파악하여 두었다가 학습자들의 협동학습에 대한 요구를 받게 되면 적절한 학원이나, 학교, 교사들을 연결시켜 주는 역할을 한다. 즉, 학습자가 학습에서 원하는 요구 사항을 중개자에게 알려 주면, 중개인은 다양한 학원, 학교, 교사에게 학습을 원하는 학습자의 이름, 위치와 전화번호 등을 전달하게 된다.

이 과정이 끝나게 되면 학원이나 학교, 교사들이

학습자의 요구 사항을 파악하여 학습 시간과 내용 등을 학습자에게 전달한다.

그러면 학습자는 자기에게 제시된 여러 곳의 정보를 토대로 가장 적합한 협력 학습 장소를 선택하여 학습을 예약한다.

마지막으로 학습자와 연결된 학습 장소(학교, 교사)에서 학습에 변동이 있을 때 수정된 사항을 전달하면 된다.

에이전트의 입장에서 볼 때 지금까지의 과정을 살펴 보면 다음과 같다.

- 1) 학습자는 개인 비서 에이전트에게 협력학습을 할 수 있도록 학습 서버를 찾아달라고 요청한다. 이때 개인 비서 에이전트는 학습자에게 기초적인 학습 자료를 요구하게 되는데, 이때 학습자는 학습과목과 학습 시간, 학습 수준 등에 대한 요구 사항을 입력한다.
- 2) 요구 사항을 입력받은 개인 비서 에이전트는 협력 학습을 할 수 있도록 학습 브로커 에이전트에게 적절한 학습 서버를 찾아 달라고 부탁을 한다.
- 3) 학습 브로커는 조정자에 등록되어 있는 모든 학습 서버에게 학습자가 요구한 사항들을 전달하게 된다.
- 4) 학습자가 요구한 내용에 부합되는 학습 서버들은 이제 직접 개인비서 에이전트에게 자신의 학습 내용과 학습의 조건들을 제시하게 된다.
- 5) 다양한 학습 서버에게서 받은 여러 자료를 가지고 개인비서 에이전트는 가장 적합한 서버를 추출하여 채택된 학습 서버에게 협력학습 시간을 예약한다.
- 6) 학습서버 에이전트가 협력 학습을 불가능하게 한 다거나 다른 이유로 학습이 변경되었을 때 수정된 내용을 전달하거나, 학습자가 협력학습의 내용을 변경하고자 할 때 변경 메시지를 보낸다. 다음은 본 연구에서 이용한 KQML의 예를 들어본다.

3.3.1 학습에 관계된 지식(Learning Ontology)

학습 지식(Learning Ontology)은 학습자 에이전트와 학습 브로커 에이전트, 학습 서버 에이전트에 의해 협력 학습에 대한 선택적인 내용을 묻는데 사용된다. 다음의 내용들은 학습 지식(Learning Ontology)에 의해 사용되어진 수행문(Performatives)들이다.

- ask-all (asking collaborative learning options)
- tell (response of ask-all)
- sorry (response of ask-all)

insert (update collaborative learning information)
error

1)협동 학습(Collaborative learning)을 묻는 KQML

```
(ask-all
:language anACL
:ontology CollaborativeLearning
:content (
:Nation KOR
:Subject Math
:theme Figure
:learningDate "12/01/99"
:startTime "16:00"
:learningTime "2-3 hour"
:learningLevel mid
:expirationTime 300))
```

2)협동 학습(Collaborative learning)을 묻는 내용에 대한 응답 KQML

```
(tel :language anACL
:ontology CollaborativeLearning
:content(
:Nation KOR
:Subject Math
:theme Figure
:learningDate "12/01/99"
:startTime "15:50"
:learningTime "3-4 hour"
:learningLevel mid ))
```

3) 실패하였을 경우 응답 KQML

```
(sorry :language anACL
:ontology CollaborativeLearning
:content (
:errmsg "no Collaborative lreaning found"))
```

4)협력학습에 대한 추가 사항 및 변경 사항 KQML

```
(insert :language anACL
:ontology CollaborativeLearning
:content(
:Nation KOR
:Subject Math
:theme Figure
:learningDate "12/01/99"
:startTime "16:20"
:learningTime "2-4 hour"
:learningLevel mid ))
```

3.3.2 학습 브로커 정보(Learning Broker Ontology)

학습 브로커 정보(Learning Broker Ontology)는 개인 학습 에이전트와 학습 브로커 에이전트에 의해 협력 학습의 선택 사항을 묻는데 사용되는 정보이다. 학습 브로커 정보(Learning Broker Ontology)는

다음 두 가지의 KQML 정보의 수행문으로 확장할 수 있다.

```
broker-all
recruit-one
```

1)학습 브로커에게 :content에 있는 수행문에 대한 응답을 찾아달라고 부탁하는 KQML

```
(broker-all
:sender personalAssistant
:receiver learningBroker
:reply-with id0
:language KQML
:ontology learning-broker
:content (
ask-all
:sender personalAssistant
:receiver learningBroker
:reply-with id1
:language anACL
:ontology CollaborativeLearning
:content (
:theme Figure
:startTimearrival "16:00"
:expirationTime 30 )))
```

2)학습 브로커 에이전트에게 :content에 있는 수행문에 응답할 수 있는 학습 서버 에이전트를 찾아서 그 학습 서버 에이전트가 개인 학습자 에이전트에게 응답하라고 부탁하는 KQML

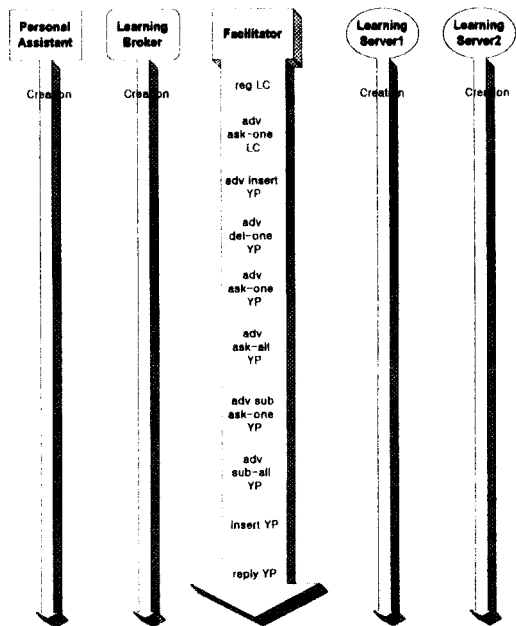
```
recruit-one
(recruit-one
:sender personalAssistant
:receiver learningBroker
:reply-with id0
:language KQML
:ontology learning-broker
:content (
subscribe
:sender personalAssistant
:reply-with id0
:language KQML
:ontology kqml-ontology
:content (
ask-all
:sender personalAssistant
:reply-with id0
:language anACL
:ontology CollaborativeLearning
:content (
:Nation KOR
:Subject Math ))))
```


3.4 멀티 에이전트간의 상호 작용 다이어그램

다음의 <표 1>은 멀티 에이전트간의 상호 메시지의 종류를 정리한 내용이며, 아래의 메시지를 가지고 다음과 같이 4단계로 협력학습을 시도한다.

<표 1> 멀티 에이전트간의 상호 메시지

메시지	축약형	의미
advertise	adv	자신을 알리고 다른 에이전트의 접근을 허용한다.
broker-all	brk-all	모든 receiver에게 원하는 지식이나 정보를 요청한다.
forward	fwd	새로 발생한 메시지를 보내달라고 요청한다.
LifeCycle Ontology	LC	각 에이전트의 생명 주기와 정보들의 지식 내용
recommend-one	rec-one	질문에 답할 수 있는 모든 에이전트를 receiver에게 끈는다.
register	reg	자신의 심볼이름을 Facilitator에게 등록한다.
subscribe	sub	지식이나 정보가 갱신되면 알려달라고 요청한다.
learning broker ontology	LB	학습을 브로커하는 생명 주기와 정보들의 지식 내용
learning ontology	LOI	협력학습을 하는 생명 주기와 정보들의 지식 내용
YellowPage ontology	YP	서비스를 제공하는 홈페이지의 지식 내용



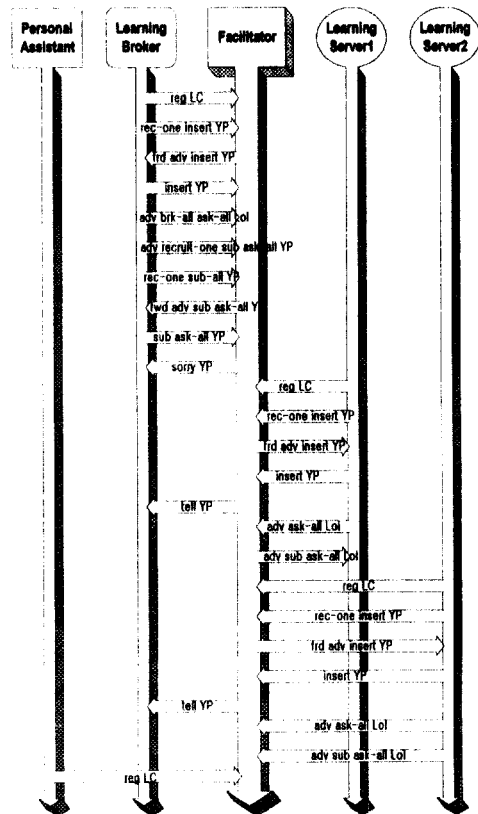
[그림 5] 생성(Creation) 단계

3.4.1 단계: 학습 에이전트 생성(Creation) 단계

[그림 5]와 같이 가장 먼저 Facilitator가 생성(Creation)이 되면 다른 에이전트들이 자신에게 등록(Register)하여 조정, 중재할 수 있도록 수행문(Performative)을 통보(Advertise)하고 조정자에게 메시지를 전달하거나 지식을 삽입하도록 허용한다.

3.4.2 단계: 등록/홍보(Register/Advertise) 단계

다음 [그림 6]과 같이 각각의 에이전트가 자기 자신을 알리기 위해 Facilitator에 등록(Register)을 하게 된다. 등록된 에이전트는 다른 에이전트들이 쉽게 자신에게 접근할 수 있도록 홍보(Advertise)한다.



[그림 6] 등록/홍보(Registration/Advertisement) 단계

우선 학습 브로커 에이전트가 학습자와 협력학습 서비스를 하는 시스템의 중재자 역할을 하기 위해 조정자(Facilitator)에 등록을 하고 자신에게 지식이

나 정보가 있으면 알려달라고 조정자에게 요청하게 된다.

다음으로 협력 학습 서버 에이전트 2개가 조정자 (Facilitator)에 등록을 하며 역시 협력학습을 하는 학생들을 모집할 수 있도록 조정자에게 요청을 한다. 마지막으로 개인 보조 학습 에이전트가 등록을 하게 된다.

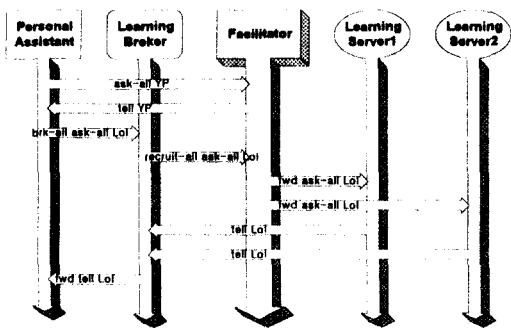
3.4.3 3단계: 협력 학습 중개(Broker) 단계

등록과 홍보가 성공적으로 이루어진 다음에 [그림 7]과 같이 개인 학습 보조 에이전트가 협력 학습을 위해 조정자에게 협력학습 서버를 연결 해 줄 수 있는 학습 브로커를 요청하게 되면 조정자는 자신에게 등록되어 있는 학습 브로커들을 검색하여 학습자에게 브로커의 위치를 통보해 준다.

학습 브로커의 위치를 파악한 학습자 에이전트는 브로커에게 자신의 요구에 맞는 학습 서버 에이전트를 연결해 달라고 요청을 하게 된다.

이를 요청 받은 학습 브로커 에이전트는 조정자에게 학습 서버를 찾아달라고 요청을 하게 되고 조정자는 자신에게 등록되어 있는 학습 서버를 검색하여 학습 브로커에게 연결되도록 도와준다.

학습 브로커는 검색된 모든 학습 서버를 개인 학습 에이전트에게 통보한다.



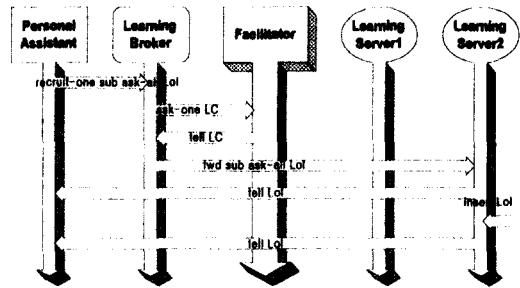
[그림 7] 중개(Broker) 단계

3.4.4 4단계: 참여 확인(Subscribe) 단계

학습 서버의 위치를 파악한 개인 학습 서버는 협력 학습을 위해 자신의 요구 사항(학습 대상, 요일, 시간, 학습 과목, 주제, 수준)을 학습 브로커 에이전트에게 전달하면 조정자에 의해 학습자의 요구에 적합한 학습 서버를 검색하여 연결하게 된다.

본 다이어그램에서는 학습 서버 2번이 적합한 협력 학습 시스템으로 결정되어 학습자 에이전트에게 협력학습에 참여하도록 확인(Subscribe)하게 된다.

만일 협력 학습 서버에 중대한 문제가 발생하거나 지식, 정보의 내용이 갱신, 변경되는 경우에는 개인 학습자 에이전트에게 변경된 정보를 전달하게 된다.



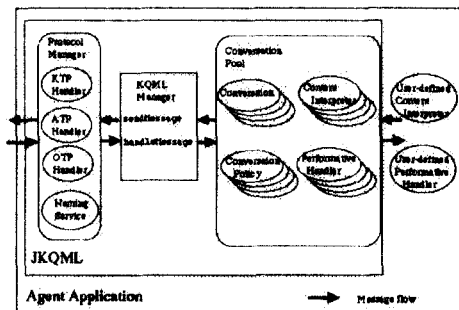
[그림 8] 참여 확인(Subscribe) 단계

4. 실험 및 고찰

4.1 협력학습 멀티에이전트 시스템의 구현

본 연구에서 설계하고 구현한 협동 학습을 위한 멀티에이전트는 객체 지향 언어인 Java를 통해 구현하였다. 추론 엔진의 패턴 매칭을 위해서 전문가 시스템 셸인 CLIPS의 Java 버전인 Jess(The Java Expert System Shell)를 사용하였으며, 에이전트간의 메시지 교환을 위해 KQML 소프트웨어인 JKQML(Java based KQML API)을 사용하였다.

JKQML은 다음과 같은 구조를 가지고 있으며 협력 학습을 하기에 적합한 구조와 기능을 제공하고 있다.



[그림 9] JKQML의 3 레이어 구조

4.2 협력학습 멀티에이전트 시스템의 실험과 고찰

협력 학습을 위한 실험 환경은 Sun Unix Server 를 기반으로 Facilitator가 구동되었으며 학습자 에이전트와 학습 서비스 에이전트의 경우 LAN환경에서 Windows 95, 98의 운영체제를 탑재한 Client 컴퓨터와 Windows NT에서 구동하였다.

멀티에이전트의 한 구성을 이루고 있는 조정자(Facilitator)는 JKQML에서 기본적으로 제공하고 있으며, 본 연구에서는 개인 학습 에이전트와 학습 브로커 에이전트와 학습 서비스 에이전트를 구현하고, 이 에이전트들을 조정자(Facilitator)에 등록하여 실험을 하였다.

학습 브로커 에이전트는 [그림 11]과 같은 인터페이스를 제공하고 있으며, 학습 서비스 에이전트는 [그림 12]와 같은 화면이다.

[그림 10] Facilitator

조정자 에이전트의 인터페이스는 [그림 10]과 같고 멀티에이전트 시스템에 등록된 모든 에이전트의 이름과 주소를 보여주고 에이전트가 제공할 수 있는 서비스를 보여준다.

[그림 12] Learning Server Agent

[그림 13]에서 보는 바와 같이 학습보조 에이전트가 자신의 요구를 입력하여 전송하게 되면 브로커 에이전트가 학습 서비스 에이전트를 검색하여 적절한 학습 서비스 에이전트를 학습자 에이전트에게 알려준다.

[그림 13] Personal Assistant Agent

적절한 학습 서비스 에이전트를 찾은 학습자 에이전트는 협력학습에 참가하겠다는 메시지를 Subscribe버튼을 클릭하여 협력학습을 예약하며 협력학습의 내용이 변경되었을 때 변경된 내용을 학습자 에이전트에게 전송하게 된다.

[그림 11] Broker Agent

본 연구에서 실험한 내용은 다음과 같다. 학습자는 협력학습을 다음과 같은 내용으로 하고자 한다.

학습 환경 한국어(Language: Kor)
 학습 과목: 수학(Subject: Math)
 학습 내용: 도형(Content: Figure)
 학습 날짜: 1999년 12월 1일(Date: 12/01/99)
 학습 시간: 16시 정각(Time: 16:00)
 학습 가능 시간: 2-3시간(Duration: 2hour)
 학습 수준: 중간(Level: mid)
 확인 시간: 3초(Expired time: 30 sec)

위의 요구 사항을 가지고 협력 학습을 하고자 하며, 학습 서비스 에이전트는 6개를 구동하여 적합한 정보를 찾고자 한다.

[그림 15] 결과의 변경 확인

5. 결론

본 연구에서는 분산 원격 협력 학습 환경에서의 멀티 에이전트 시스템의 적용과 이러한 멀티 에이전트간의 효율적인 정보 전달을 위한 통신 언어인 KQML의 적용에 대한 내용을 가지고 협력 학습 시스템을 설계 구현하였다. 학습 서비스 에이전트가 협력 학습을 주관하여 다양한 학습 방법을 제공하고 있으며 이를 학습 브로커가 다양한 학습자들을 모집하여 협력 학습의 중개자 역할을 하였으며, 개인 학습자의 협력학습을 쉽게 돕고자 개인 학습자 에이전트가 자율적이고 지능적으로 학습자의 요구사항을 받아 적합한 학습 서비스 에이전트를 찾아내었다.

이러한 에이전트간의 원활한 정보 교환과 통신을 위해 KQML을 사용하였으며, 협력 학습에 맞는 시나리오를 설계, 구현하여 실험에 적용하여 최적의 협력 학습의 환경을 제공하였다.

본 연구에서 구현한 시스템은 효율적인 원격 분산 협력 학습을 위한 기본 구조를 제공하고 있으며 실제적인 협력 학습의 방법에 대해 다양하게 접근할 필요가 있으며, 기존의 웹기반 교육 응용 프로그램을 통합하여 현재 다르게 진행되고 있는 연구와 접목시키고, 보다 효과적인 협력 학습 시스템을 구축하는 연구가 진행되어야 하겠다.

[그림 14] 참여 확인 인터페이스

검색 결과와 요청한 질문에 대한 응답은 다음 [그림 14]와 같이 학습자의 요구에 가장 적합하게 제시되었다.

또한 [그림 15]와 같이 학습 서비스 에이전트의 협력학습 내용이 변경되었을 때, 변경된 내용이 학습자 에이전트에게 올바르게 전송되는 것을 알 수 있다. 이전에 서버가 제공하는 수학 도형의 협력학습 시간이 16:00시 정각이었던 것이 서버의 사정이나 협력 학습 참여자의 요청에 의해 16:10분으로 변경된 결과를 학습자에게 전송한 화면이다. 변경에 대해서 기존 학습자의 협력학습에 대한 학습의 가능여부를 재전송할 수 있으며, 재전송하지 않을 경우 변경된 시간에 협력학습을 하는 것으로 확인하게 된다.

참고문헌

[1] Bradshaw, J. M., An Introduction to Software Agents. In Software Agents, ed J. M. Bradshaw. Menlo Park, Calif.: AAAI Press. 1997.

[2] C. Petrie. The Secretaries Nightmare, Proceedings of the CAIA Workshop on Coordinated Design and Planning, 1994.

[3] FIPA '97 Draft Specification: part 2 Agent Communication Language., 1997.

[4] J. S. Rosenschein and G. Zlotkin, Rules of Encounter: Designing Conventions for Automated Negotiation among Computers, The MIT Press, Apr. 1994.

[5] Labrou, Y and Finin, T., A Proposal for a new KQML Specification. TR CS-97-03, 1997.

[6] Labrou, Y., Semantics for an Agent Communication Language. Ph.D. diss., University of Maryland at Baltimore County. 1996.

[7] Micheal Knapik & Jay Johnson, Developing Intelligent Agents for Distributed Systems, McGraw-Hill, 1998.

[8] Micheal N. Huhns & Larry M. Stephens, Multiagent systems and Society of agents, The MIT Press, 1999.

[9] T. Finin, J. Weber, and et al, Specification of the KQML Agent-Communication Language,

[10] The ARPA Knowledge Sharing Initiative External Interfaces Working Group, Draft Specification of the KQML Agent-Communication Language Plus Example Agent Polices and Language, *Technical Report of ARPA Knowledge Sharing Effort Group*, June 15, 1993.

[11] Wooldridge, M. and Jennings, N. R., Intelligent Agents : Theory and Practice, *Submitted to Knowledge Engineering Review*, Oct. 1994.

[12] Wooldridge, M., Jennings, N. R., Intelligent Agents: Theory and Practice, *Knowledge Engineering Review*, 10(2), pp. 115-152, Jun. 1995.

[13] SunGwan Han, GeunSik Jo. Web based Intelligent Tutoring System for Collaborative Learning, *Webnet99*, 1999.

이 첩 환

1977년 인천교육대학교 졸업
 1989년 연세대학교 교육대학원 졸업(교육학 석사)
 1988년 한국방송통신대 전자계산학과 졸업
 1993년 미국 피츠버그대 정보공학대학원 졸업 (정보공학 석사 MSIS)
 1994년 미국 피츠버그대 사범대학원 교육공학과 컴퓨터교육 전공 졸업(교육학 박사 Ed.D)
 1989-1991년 미국 피츠버그대 사범대학원 교육공학과 연구 조교
 1991-1993년 미국 펜실바니아주 컴퓨터 교육 연구소 (ITEC Center) 연구원
 1994-현재 인천교육대학교 컴퓨터 교육과 교수
 연구 분야: 컴퓨터 교육, 멀티미디어, 웹기반 교육

한 선 관

1991년 인천교육대학교 졸업
 1995년 인하대학교 전자계산교육과 졸업(석사)
 1998-1999년 인하대학교 전자계산공학과(박사과정)
 1999년~현재 인천교육대학교 컴퓨터교육과 강사
 연구 분야: 에이전트 시스템, 전문가 시스템, 지능형 교육 시스템, 인공 지능, 원격 협력 학습

<URL>

<http://www.alpha-works.ibm.com/formula/jkqml>
<http://www-ksl.stanford.edu/knowledge-sharing/papers/kqml-overview.ps>
<http://www.cs.umbc.edu/~jklabrou/publications/tr9703.ps>
<http://www.cs.umbc.edu/kqml/kqmlspec.ps>