

지능형 의료 정보 제공을 위한 멀티 에이전트 플랫폼

최원기

모빌씨앤씨 연구소
(choiwk@mcnc.co.kr)

김일곤

경북대학교 컴퓨터학과 부교수
(ukkim@knu.ac.kr)

의료 정보 시스템은 많은 지식과 의료 인력과의 연계가 필요하기 때문에, 멀티 에이전트 시스템의 활용 영역으로 적합하다. 본 논문에서는 지능형 의료 정보 제공을 위한 멀티 에이전트 플랫폼의 설계 및 구현에 대해 기술한다. 멀티 에이전트 시스템을 이용하여 지능형 의료 영상 정보를 제공하기 위해 FIPA(Foundation for Intelligent Physical Agent)가 제안하는 에이전트 플랫폼 모델을 따르고 있다. 윈도우 기반에서 컴포넌트 활용을 위해 COM(Common Object Model)과 XML(Extensible Markup Language)로 인코딩된 ACL(Agent Communication Language)을 메시지 교환에 이용하고 있다. 의료 영역의 특성상 의료 전문 인력과의 연계가 아주 중요한 요소이기 때문에, 의료 인력을 하나의 휴먼 에이전트로 개념화하여 시스템에 연결하고, 소프트웨어 에이전트와 휴먼 에이전트들이 지식을 공유하기에 적합한 물리적 기반을 제공하여 지능형 의료 정보 서비스를 용이하게 한다.

1. 서 론

멀티 에이전트 시스템은 자치성을 가진 에이전트들이 모여 서로 지식을 교환하며 자신의 목표를 추구하는 시스템이다[1]. 멀티 에이전트 시스템은 동적이고 고도의 자치성을 요구하는 영역에 적합하다[2]. 의료 영역은 고도의 전문 지식을 가진 전문 의료 인력과의 연계가 필요하고, 진료 행위 역시 복잡하기 때문에 이들을 연결해 주는 자치 시스템을 필요로 한다. 또, 의료 영역에서 정보는 의료 지식의 전달을 필요로 하기 때문에, 지식을 주고 받는 멀티 에이전트의 시스템의 활용 영역으로 적합하다.

본 논문에서는 의료 영역 중 현재 활발하게 전산화가 진행중인 의료 영상에서 지능형 의료 정

보 서비스를 제공하는 지능형 의료 정보 멀티 에이전트 플랫폼에 대해 기술한다. 에이전트 플랫폼은 에이전트들이 활동할 수 있는 물리적 기반을 제공해주며, 에이전트들이 사용하는 언어, 통신 방법, 에이전트들 간의 관계를 결정짓는다. 본 플랫폼은 FIPA(Foundation for Intelligent Physical Agent) 규격을 근간으로 구성되어 있으며, 의료 영역에서는 필수적인 전문 의료 인력과의 연계를 지원한다.

본 논문의 구성은 다음과 같다. 2절에서는 지능형 의료 정보 멀티 에이전트 플랫폼의 근간을 이루는 FIPA 규격에 대해 살펴보고, 3절에서는 본 플랫폼이 어떻게 구성되어 있는지에 대해 기술한다. 4절에서는 XML로 인코딩된 ACL이용과 그 장점, 그리고 XML을 이용한 사람과의 연동에 대해 살펴

본다. 본 플랫폼에서는 사람과의 직접적 연결을 위해 WAP(Wireless Application Protocol)을 활용하고 있으며, 5절에서는 본 플랫폼의 특성과 다른 플랫폼과의 기능적 비교에 대해 논하고 마지막으로 결론, 향후 과제에 대해 기술한다.

2. FIPA 규격

FIPA는 다양한 회사들과 연구소들이 모여서 만든 에이전트 표준화 기구이다. FIPA97, FIPA98, FIPA2000[3]을 비롯해 꾸준히 표준화 작업을 진행하고 있다. FIPA2000에서는 특히 에이전트 메시지와 관련된 규격들이 많이 제시되었다. 본 절에서는 FIPA의 규격들중에서 에이전트 관리, 에이전트 통신 언어, 휴먼-에이전트 상호작용에 대한 부분들을 살펴보기로 한다.

(1) 에이전트 관리

에이전트들은 플랫폼의 MTS(Message Transport System)을 통해 메시지를 교환한다. 에이전트 플랫폼은 에이전트들이 동작할 수 있는 물리적 기반을 제공해준다. 에이전트 플랫폼에는 AMS(Agent Management System), DF(Directory Facilitator)의 관리를 위한 에이전트들이 존재한다. 이전 규격에는 ANS(Agent Name Server), ACC(Agent Communication Channel)이 존재하였는데 규격의 발전에 따라 사라졌다. AMS는 에이전트 관리를 담당하고 있으면 DF는 옐로우 페이지(yellow page) 역할을 담당하고 있다. MTS는 플랫폼내의 에이전트들만이 아니라 다른 플랫폼에 있는 에이전트간의 메시지 전달을 담당하고 있다. 에이전트 플랫폼간의 통신은 IOP

(Internet Inter Orb Protocol)을 이용하도록 규정되어 있다.

에이전트들은 AID(Agent ID)를 가지고 MTS는 메시지들을 이 AID를 보고 해당 에이전트에게 라우팅하도록 되어 있다. AID는 에이전트 이름을 가지며 추가적으로 전송 주소, 이름 resolution 서비스등의 패러미터가 존재한다. 에이전트 이름은 일반적으로 에이전트의 실제 이름과 홈 에이전트 플랫폼 주소의 쌍을 '@'으로 연결해 사용한다. 전송 주소는 에이전트와 접촉할 수 있는 물리적 주소이며 여러 종류의 전송 주소가 있을 경우 이들을 모두 기술해 줄 수 있다. AID는 그림 1과 같이 구성된다.

```
(agent-identifier
 :name wonki@giant.knu.ac.kr
 :addresses
 (sequence dcom://giant.knu.ac.kr/acc))
```

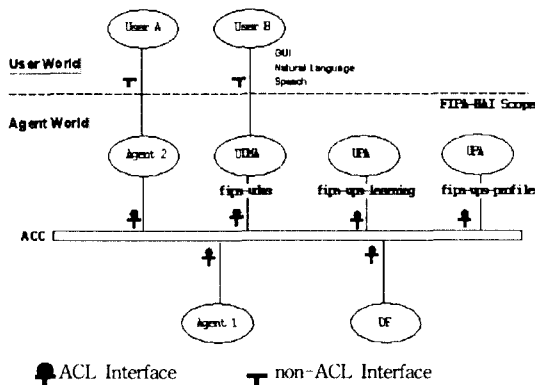
<그림 1> Agent ID 예

(2) 에이전트 통신 언어

ACL(Agent Communication Language)은 에이전트간의 통신에 이용되는 언어이다. FIPA에서는 메시지 계층(Message Layer)과 내용 계층(Content Layer)으로 나누어 내용 전달에 사용되는 언어는 에이전트 임의로 사용하고, 전달에 이용되는 계층만 따르면 메시지가 전달될 수 있게 해준다. 메시지 계층은 KQML(Knowledge Query Manipulation Language)과 비슷한 형태이다. KQML의 Performative와 같은 역할을 하는 Communicative Act를 가지고 있으며 메시지 전송에 필요한 파라미터들을 가지고 있다.

(3) 휴먼-에이전트 상호 작용

FIPA에서는 사람과 에이전트간의 상호 작용에 대한 규격을 제시하고 있다[5]. 규격에서는 그림2과 같은 휴먼-에이전트 상호 작용 모델을 제안하고 있다. 이 모델에서는 사용자(휴먼)가 UDMS(User Dialog Management Service)를 제공하는 에이전트(UDMA: User Dialog Management Agent)와 대화를 통해 사용자가 다른 에이전트들과 정보를 교환하는 형태이다. FIPA는 UDMS외에도 사용자 개인 프로필(User Personalization profile)을 획득하고 관리하는 UPS(User Personalization Service)를 정의하고 있다. 이 두 서비스를 위한 온톨로지, 프로토콜, 사용자와 에이전트 대화에 사용되는 액션들의 의미를 정의하고 있다.



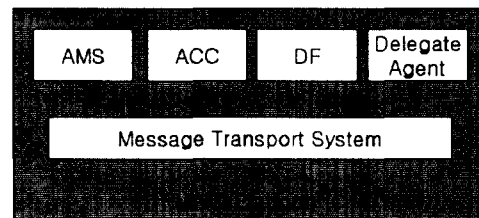
fipa-udms : FIPA user dialogue management service ontology
 fipa-ups-profile : FIPA user personalization profile
 fipa-ups-learning : FIPA user personalization learning ontology

<그림 2> Human-Agent Interaction Reference Model

에이전트 입장에서는 사용자는 다른 에이전트와 크게 다르지 않다. 다만 UDMS를 제공하는 서비스를 통해 접촉하는 점만이 다르다. 하지만 에이전트와 사람이 사용하는 언어(자연어)는 매우 달라서 많은 문제점을 가진다. 그래서 자연어와 ACL을 동일한 방법으로 표현하는 언어의 필요성이 대두되었고, 이를 위해서 SKDL (Structured Knowledge Description Language)¹⁾를 소개하고 있다.

3. 지능형 의료 정보 멀티 에이전트 플랫폼

지능형 의료 정보 멀티 에이전트 플랫폼은 FIPA의 에이전트 관리 모델을 따르며, 그림3과 같이 구성되어 있다. 플랫폼에 있는 내부 에이전트는 FIPA 규격에 나오는 AMS, DF 외에 ACC와 위임 에이전트(Delegate Agent)가 존재한다. ACC는 이전 FIPA 규격에 포함되어 있었던 에이전트로 다른 플랫폼간의 메시지 전송과 같은 특수한 형태의 메시지 전송을 담당하던 에이전트이다. 위임 에이전트는 사람을 대신하는 역할을 한다.



<그림 3> 지능형 의료 정보 멀티 에이전트 플랫폼

1) SKDL은 필요한 특성과 간략한 형태만 기술되어 있다.

```
//연결을 위한 인터페이스
interface ConnectionInterface : IUnknown
{
    HRESULT connect(
        [in, ref] IUnknown *thisAgent,
        [in, ref] ConnectionContext *context,
        [out, retval] MessageInterface **mInt);
}
//메시지 전송을 위한 인터페이스
//에이전트, 플랫폼 둘 다 가짐
interface MessageInterface : IUnknown
{
    HRESULT sendMessage([in] BSTR m);
}
```

<그림 4> 플랫폼의 COM Interface

지능형 의료 정보 멀티 에이전트 플랫폼에서 에이전트간의 통신에 사용되는 메시지는 XML로 인코딩[4]된 형태를 사용하며, FIPA의 CA와 파라미터의 의미를 그대로 따른다. 에이전트와 플랫폼의 연결은 COM을 사용한다. 에이전트는 플랫폼에 연결되어 메시지를 다른 에이전트에게 보내는 데 이 때 사용되는 인터페이스는 그림 4와 같다.

AMS는 이름처럼 에이전트 관리하는 역할을 한다. 에이전트가 플랫폼의 서비스를 제공받기 위해서는 먼저 AMS에 등록해야 한다. 등록시에는 자신을 기술하는 내용(AMS Agent Description)을 보내야 하고 그 내용은 표 1과 같으며, FIPA2000 규격과는 약간의 차이가 있다. state에서는 이동 에이전트를 고려하지 않기 때문에 Transit이 없고, delegate²⁾, race라는 추가 항목이 있다. name 항목 역시 FIPA에서는 필수는 아니지만 본 플랫폼에서는 반드시 필요한 항목이다. Style-Sheet 항목은 등록시에 에이전트

가 보내지 않고, put-style을 하게 되면 AMS가 직접 채운다. AMS가 처리하는 연산들은 표3에 나타나 있다. 본 플랫폼에서 추가된 연산은 put-style로 에이전트가 자신이 사용하는 언어의 XML Style-Sheet를 넣어주는 것이다. AMS는 웹 서버에 받은 내용을 기반으로 XML 스타일 시트(Style Sheet)를 생성해 준다. 에이전트는 자신이 사용하는 언어(:language) 별로 복수의 스타일 시트를 등록할 수 있다.

AMS는 연결된 에이전트의 관리를 위해 주기적으로 ping 메시지를 보낸다. 본 플랫폼의 에이전트들은 반드시 이 메시지에 응답을 보내야 한다. AMS는 3번 이상 응답하지 않는 에이전트는 연결이 끊어졌다고 판단하고 해당 에이전트를 삭제한다.

DF는 에이전트가 자신이 제공하는 서비스들을 광고하는 옐로우 페이지 역할을 한다. DF에 자신을 광고하기 위해서는 AMS와 마찬가지로 자신의 능력을 기술해서 등록해야 한다. AMS/DF 등록 항목들은 표 1, 2와 같다. 본 플

<표 1> AMS Agent Descriptions

파라미터	의미	유무
Name	에이전트 ID	Require
Ownership	소유자	Option
Race	종류(software, human)	Require
State	라이프 사이클(waiting, suspended, initiated)	Option
Delegate	직접적 연결이 불가능할 때 대신 메시지를 받을 에이전트 ID	Option
Style-Sheets	스타일 시트 페이지 주소들	Option

2) 이전 FIPA 규격에는 있었던 항목이다.

<표 2> DF Agent Descriptions

파라미터	의미	유무
Name	에이전트 ID	option
Services	지원하는 서비스들	option
Protocol	지원하는 프로토콜들	option
Ontology	알고 있는 온톨로지들	option
Language	내용 계층에 사용하는 언어들	option

<표 3> Operations in AMS and DF

Function	의미	지원
Register	등록	AMS/DF
Deregister	삭제	AMS/DF
Modify	변경	AMS/DF
Search	질의	AMS/DF
Get-description	플랫폼 프로파일 ³⁾ 을 질의	AMS
Put-style	에이전트의 XML Style Sheet를 등록	AMS
Put-Service-page	에이전트의 서비스 페이지를 등록	DF
Get-service-page	등록된 서비스 페이지를 가져온다	DF

플랫폼에서 DF는 소프트웨어 에이전트만이 아니라 사람도 위임 에이전트를 통해 등록할 수 있다. DF가 제공하는 연산은 표 3에 있다. Put-service-page와 get-service-page는 에이전트의 서비스를 사람이 이용할 수 있게 해주는 XML 문서를 등록하고 가져오는 연산이다. service-page는 에이전트가 제공하는 서비스를

사람이 브라우저를 통해 요청할 수 있도록 하기 위해 만들어진 것이다. 위임 에이전트는 service-page를 이용해 사람이 서비스를 직접 요구할 수 있게 해준다.

에이전트 그림에서 나오는 인터페이스를 이용해 플랫폼에 연결한 뒤 MTS를 통해 메시지를 전송할 수 있다. MTS는 receiver의 에이전트 ID를 보고 해당 에이전트에게 전송한다. 만약 receiver가 물리적으로 MTS와 연결되어 있지 않다면 AMS에 있는 delegate 항목에 등록된 에이전트에게 대신 전달한다. 이 과정을 거치고도 MTS가 전송할 수 없는 메시지는 ACC로 전송되고 ACC는 메시지를 분석하여 해당하는 에이전트로 전송한다. ACC로 메시지가 포워딩되는 경우는 대개 다른 플랫폼에 있는 에이전트에게 메시지를 보내는 경우이다. 본 플랫폼에서는 ACC가 inter-platform 통신을 담당하고, 사람도 소프트웨어 에이전트와 동일하게 취급한다. 이를 위해서는 사람과 에이전트 플랫폼의 중간에서 매개체 역할을 하는 존재로서 위임 에이전트(Delegate Agent)가 존재한다. 위임 에이전트는 웹 서버와 연동되어 사람이 접근할 수 있는 인터페이스를 제공해준다. 사람도 에이전트와 마찬가지로 에이전트 플랫폼의 서비스를 제공받기 위해서는 AMS에 등록해야 하는데, 위임 에이전트가 사람의 아이디를 이용해서 AMS에 사람을 에이전트로 등록하고 그 사람의 능력들을 DF에 등록해 광고할 수 있다. 앞으로 이렇게 등록된 사람을 휴먼 에이전트라 지칭하고 에이전트는 인공으로 만들어진 에이전트를 지칭한다. 위임 에이전트는 휴먼 에이전트 등록시에 자신을 휴먼 에이

3) FIPA에 Agent Platform Profile이 정의되어 있다. Profile에는 주소, 지원하는 전송 프로토콜들, Mobile 지원 여부 등이 있다.

전트의 대리자로 AMS에 등록한다. 이후에 만약 어떤 에이전트가 휴먼 에이전트에게 메시지를 보내게 되면 앞에서 기술한 MTS의 동작에 의해 위임 에이전트로 메시지가 전송되고 MTS는 웹 서버를 통해 휴먼 에이전트가 접속되었을 때 메시지를 전달하거나 푸쉬(push)를 이용해 바로 전달할 수 있다.⁴⁾ 휴먼 에이전트의 접속 환경은 인터넷 익스플로러와 같은 XML 브라우저와 WAP 브라우저 환경으로 제한한다. 브라우저의 구분은 웹 서버를 통해 쉽게 할 수 있다. XML을 이용하는 방법은 4절에 기술되어 있으며 WAP 이용은 5절에 기술되어 있다. 위임 에이전트는 COM의 로컬 서버 형태로 구현되어 오토메이션 인터페이스를 통해 ASP(Active-Server page)와 연동되어 웹 서비스를 제공해준다. 환경의 구분은 웹 서버가 브라우저 정보를 알려주기 때문에 쉽게 할 수 있다. WAP 환경에서는 여러 제약 사항 때문에 단순 통보만을 지원하고, XML 브라우저는 좀더 다양한 형태의 상호 작용이 가능하다. XML 브라우저의 경우 두 가지 형태로 구분할 수 있다. 하나는 에이전트로부터 휴먼 에이전트가 메시지를 받고 응답을 보내는 경우이고, 나머지 하나는 휴먼 에이전트가 에이전트의 서비스를 요구한 경우이다. 에이전트로부터 메시지를 받는 경우는 위임 에이전트는 다음과 같이 동작한다.

- ① 에이전트에게서 받은 메시지를 분석하고 저장한다.
- ② AMS에게서 메시지를 보낸 에이전트의 스타일 시트 주소를 가져온다. 없다면 not-understood 메시지를 보낸다.
- ③ 가져온 스타일 시트 주소와 보낸 메시지를 이용해 XML문서를 생성하고, 메시지의 CA와

프로토콜에 따라 선택할 수 있는 행동들을 버턴 형태로 문서에 추가한다. 만들어진 문서는 브라우저를 통해 휴먼 에이전트에게 보여진다.

- ④ 휴먼 에이전트는 필요한 입력을 채우고 자신이 원하는 행동을 선택한다.
- ⑤ 휴먼 에이전트가 선택된 행동과 입력을 이용해 응답 메시지를 만들어 이를 메시지를 보낸 에이전트에게 보낸다. 물론 응답할 필요가 없다면 메시지를 생성해 보내지 않는다.

휴먼 에이전트가 에이전트에게 먼저 메시지를 보내는 경우는 에이전트의 서비스를 요청할 때이고 위임 에이전트는 다음과 같이 동작한다.

- ① DF에게서 휴먼 에이전트가 필요로 하는 에이전트를 검색한다.
- ② 검색된 에이전트를 휴먼 에이전트에게 보여주고 에이전트와 원하는 서비스를 선택하게 해준다.
- ③ 선택된 에이전트의 서비스 페이지를 DF에게서 가져오고, 스타일 시트를 AMS에게 가져와 서비스 요청 XML 문서를 생성한다.
- ④ 생성된 문서에서 휴먼 에이전트는 입력이 필요한 항목을 채우고 서비스를 요청한다.
- ⑤ 입력과 서비스 요청 페이지를 이용해 메시지를 생성해 메시지를 에이전트에게 보낸다. 이때 메시지의 Conversation-id, reply-to와 같은 파라미터는 위임 에이전트가 임의로 생성한다.

위 과정들을 살펴보면 휴먼 에이전트의 입력을 필요로 하는 경우가 있다. 사용자의 입력을 받는

4) push는 아직 구현되어 있지 않다.

화면은 스타일 시트를 이용하면 되지만, 입력을 받아 들이는 일은 위임 에이전트가 해주어야 한다. 이를 위해서는 입력이 필요한 부분을 구분해주는 공통의 XML 태그가 필요하다. 본 플랫폼에서는 <VARIABLE> 태그를 이용해 이를 구분한다. <VARIABLE> 태그의 이용에 따라 내용 계층 언어의 설계에 약간의 제약이 따르게 된다.

4. ACL에서 XML이용

ACL에 XML을 이용한다는 것은 FIPA의 정의된 CA와 파라미터들의 의미를 그대로 유지하면서 인코딩 방법으로 XML을 사용한다는 것을 의미한다. XML로 인코딩된 ACL의 장점은 다음과 같다[6]. 인코딩된 메시지의 예는 그림 5에 있다.

- XML로 인코딩 된 경우 파서 개발이 용이하다. XML 파서에 DTD 정보만 넘겨주면 된다. 따라서, 새로운 ACL 구문이 추가되거나 변경되어도 쉽게 이를 지원할 수 있다.
- ACL의 XML화는 에이전트를 좀 더 웹 환경과 친숙하게 만들어, 웹 기술과 에이전트의 결합을 용이하게 해줄 것이다.
- XML의 링크 기능을 이용하게 되면 ACL 메시지의 content에 사용되는 온톨로지를 공유하게 쉽게 해준다. 온톨로지를 단순히 명시하지 않고, 온톨로지가 정의된 곳을 링크하게 되면 여러 에이전트들이 온톨로지를 쉽게 공유할 수 있다.
- 링크 기능은 위에 기술한 것 외에도 다양한 형태로 이용될 수 있다. 예를 들자면 메시지의 송신자(sender)를 그 에이전트의 정보가 있는 파일의 위치를 링크하게 되면 수신자는 송신자를 좀 더 잘 이해하게 된다.

위에 열거된 장점들은 ACL 메시지 계층을 XML로 인코딩되었을 때의 장점들이다. 내용 계층을 XML로 인코딩할 경우, 위에 열거된 장점들 외에 또 하나의 장점으로 XML은 html과 달리 내용 중심으로서 뷰가 분리되어 있어 여러 가지 형태로 디스플레이가 가능하다는 것이다[7]. XML 문서는 스타일 시트를 지정할 수 있으며, 이 스타일 시트에 따라 동일한 문서가 여러 가지 형태로 디스플레이 될 수 있다. XML의 이 특징은 멀티 에이전트 시스템과 휴먼 에이전트를 연동시키는데 많은 장점을 준다. 휴먼 에이전트와의 연동에서 가장 어려운 점은 에이전트 간에 이용되는 ACL을 사람이 쉽게 이해할 수 없기 때문이다. 따라서 에이전트가 사람에게 메시지를 전달할 때는 SKDL과 같이 자연어에 가까운 형태를 사용하고자 하는 시도들이 있다. 하지만 현재까지 특별한 해결방법이 없는 자연어 처리를 통한 접근 방식은 현실적으로 해결책을 제시하지 못할 것으로 판단된다. 본 논문에서는 자연어 형태의 접근 방법 대신 인공 언어로 표현된 지식을 적절한 표현 양식으로 사람이 쉽게 이해할 수 있는 형태로 꾸미는 방법을 제안한다. 데이터베이스에 들어있는 복잡한 데이터들도 이들이 어떤 의미를 가지는지를 아는 응용 프로그램에 의해서 사람이 쉽게 이해할 수 있게 나타나는 것처럼 ACL도 잘 디스플레이한다면 사람이 쉽게 이해할 수 있을 것이다. 하지만 시스템을 구성하는 에이전트들은 아키텍처, 언어가 모두 동일할 수 없기 때문에 각 에이전트마다 자신의 ACL을 잘 디스플레이하는 인터페이스를 가지는 것은 많은 문제점이 있다. 각각의 에이전트에게 사용자 인터페이스를 구현한다는 것도 큰 오버헤드이고, 제각각으로 구현된 인터페이스를 연동하는 것은 매우 어려운 일이다. 웹 기술과 XML은 이와 같

은 문제점을 해결해준다. 웹 기술은 통일된 인터페이스를 제공해주고, XML의 스타일 시트는 에이전트에게 가장 효과적으로 자신의 ACL을 디스플레이하는 방법을 제공해 준다.

에이전트가 휴먼 에이전트에게 메시지를 보낼 때 사람이 이해할 수 있는 자연어에 가까운 메시지를 보내지 말고, 자신이 사용하는 XML 형태의 메시지를 보내고, 이에 덧붙여서 사람이 쉽게 알아볼 수 있게 디스플레이해주는 스타일 시트를 알려주는 것이다. 따라서 에이전트는 휴먼 에이전트에게 효과적으로 의미를 전달할 수 있다. 이와 같은 방법은 메시지를 받는 대상이 휴먼 에이전트 여부를 고려하지 않아도 되기 때문에, 에이전트의 부담을 줄여준다.

그림 5는 DICOM 에이전트가 휴먼 에이전트에게 의료 영상 판독을 질의하는 메시지의 예이다. dicom@gaint.knu.ac.kr는 AMS에 자신의 스타일 시트를 이미 등록해둔 상태이다. 위임 에이전트는 먼저 메시지 타입이 query이고, 프로토콜이 fipa-query인 것을 보고 <VARIABLE> 태그에 있는 내용을 휴먼 에이전트가 입력해야 한다는 것을 알고 그림 7과 같은 화면을 보여주게 된다. content를 등록된 스타일 시트와 결합하면 템플릿 형태로 디스플레이된다.

그림 7의 맨 밑에 있는 버튼들은 그림 6과 같은 fipa-query 프로토콜에서 query에 응답을 선택할 수 있게 해주는 것이다. 버튼들은 각각 not-understood, refuse, inform에 해당한다.

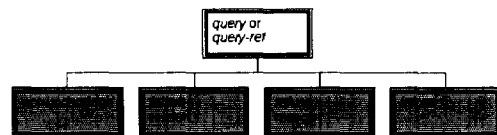
휴먼 에이전트가 일정 시간(reply-by) 동안 접속하지 않아 메시지를 보낼 수 없을 때, 위임 에이전트는 failure 응답을 보내게 된다. 예에서 나타나는 것처럼 XML과 스타일 시트의 활용을 통해 사람이 쉽게 에이전트에게 지식을 공유할 수 있다. 다른 CA들과 프로토콜들에 대해서도 각

상황에 맞는 화면을 보여주어 에이전트와 휴먼 에이전트의 통합 기능을 제공한다.

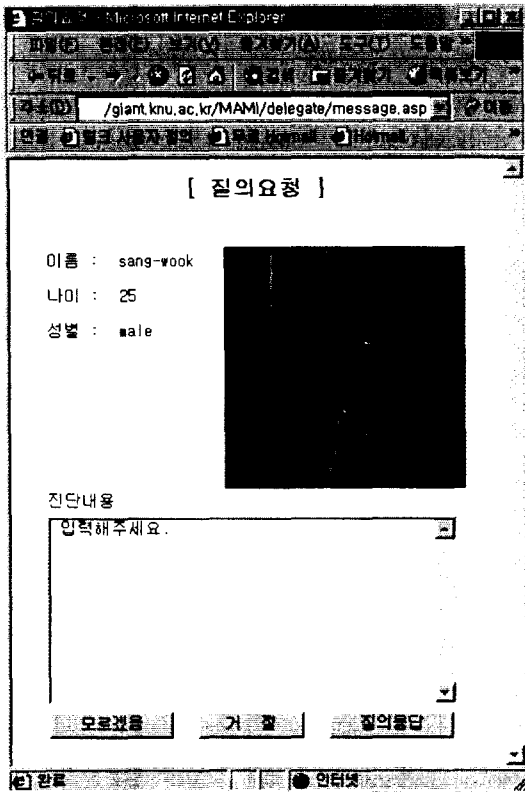
```

<?xml version="1.0"?>
<!DOCTYPE mami_acl SYSTEM "mami_acl.dtd">
<message>
<messagetype> query </messagetype>
<messageparameter>
<sender>
<agent-identifier>
<name> dicom@giant.knu.ac.kr </name>
<addresses> <sequence>
<addItem> dcom://giant.knu.ac.kr/acc </addItem>
</sequence> </addresses>
</agent-identifier>
</sender>
<receiver> wonki@giant.knu.ac.kr </receiver>
<language> dicom_diagnosis </language>
<protocol> fipa-query </protocol>
<reply-with> query-0011 </reply-with>
<reply-by> 30 </reply-by>
<content>
<dicom-image>
<patient>
<name> sang-wook </name>
<sex> male </sex>
<age> 25 </age> </patient>
</image link=
http://gaint.knu.ac.kr/dicom/c0001.dcm>
<diagnosis>
<VARIABLE> opinion </VARIABLE>
</diagnosis>
</dicom-image>
</content>
</messageparameter> </message>
    
```

<그림 5> XML로 인코딩된 메시지 예



<그림 6> FIPA Query Protocol



<그림 7> 질의 메시지 뷰 페이지

5. 시스템 평가 및 의의

지능형 의료 정보 멀티 에이전트 플랫폼과 다른 에이전트 플랫폼과의 기능적 비교를 표 4에 기술하였다[8, 9, 10]. 표에서 나타나듯이 아직 다른 플랫폼들에서는 전문 인력과 연동보다는 소프트웨어 에이전트 간의 연동에 더 초점을 맞추고 있다.

기존의 소프트웨어 패러다임에서는 사람은 사용자로서 소프트웨어는 사용자에 의해 조작되는 유틸리티(utility)로 사람에게 종속되어 사용되어

왔다. 에이전트라는 새로운 소프트웨어 패러다임에서는 사람의 조작에 의해 종속되는 소프트웨어가 아니라 독자적으로 동작하며 사람에게 서비스를 제공하거나 필요에 따라 사람으로부터 도움을 받는 - 지시를 받는 것이 아니라 - 관계로의 변환이 필요할 것으로 생각된다. 본 플랫폼에서는 이러한 패러다임 변화를 수용할 수 있게 하기 위해 전문 인력과 소프트웨어 에이전트 연동의 한 방법을 제시하고 있다. 에이전트 시스템이 동작하는 제한된 환경을 사람이 인식하기 쉬운 형태로 변환하고 이를 사람에게 전달하는 방법을 통해 사람을 하나의 휴먼 에이전트로 모델링하고 있다. 본 논문에서 제시한 방법론의 유용성은 다양한 영역에 걸쳐 실험되지 못하였지만, 시스템과 전문 인력(휴먼 에이전트)과의 유기적 결합 가능성을 보여준다.

<표 4> 다른 플랫폼과의 비교

	본 플랫폼	FIPA OS	Grass-Hopper
FIFA 규격	준수	준수	준수하지 않음
Protocol	COM	CORBA	CORBA TCP RMI
Mobile Agent	비지원	비지원	지원
ACL	XML encoded FIPA ACL	FIPA	
인력과의 연동	지원	지원하지 않음	지원하지 않음

6. 결론 및 향후 과제

지능형 의료 정보 멀티 에이전트 플랫폼은 FIPA 규격을 준수한 멀티 에이전트 플랫폼을 물리적 기반으로 구성하고, 에이전트간의 통신을 가능하게 해주며, 지식 교환을 용이하게 하기 위한 서비스들을 제공하고 있다. 메시지 교환에는 XML로 인코딩된 ACL을 사용한다. 사람을 휴먼 에이전트로 개념화하여 시스템을 구성하는 다른 소프트웨어 에이전트들과 연동은 본 플랫폼의 가장 중요한 특징이다. 이를 위해 위임 에이전트가 플랫폼에 있으며, 위임 에이전트는 WAP과 웹 환경을 통해 휴먼 에이전트와 접촉한다. 휴먼 에이전트와 지식교환을 위해 XML과 스타일 시트를 이용하는 방법을 제안하고 있으며, WAP을 통해 간단한 자연어로 휴먼 에이전트와 직접적으로 연결되어 있다. 본 플랫폼에서는 사람과 지식 교환을 통해 전문 의료 인력의 지식을 효과적으로 활용할 수 있게 해준다.

향후 과제로는 먼저 XML과 스타일 시트를 이용한 지식 교환 방법을 여러 영역에 적용해 효율성을 검증하는 것이다. 본 논문에서 그 가능성에 대해 살펴보았지만, 아직 일반적으로 사용했을 때 효율성은 검증되지 않았다. 그리고 다른 FIPA 플랫폼과의 상호 운용성을 지원해야 할 것이다. 또 SOAP(Simple Object Access Protocol)을 이용하는 에이전트 간 메시지 전달 메커니즘을 통해 웹과의 연동을 용이하게 하는 방법에 대한 연구가 필요하다.

참고문헌

- [1] Michael N. Huhns and Munidar P. Singh, Readings In Agents, Morgan Kaufmann Publishers Inc., p1-24, 1998.
- [2] Gerhard Weiss, Multiagent System, MIT Press, p.79-82, 1999.
- [3] FIPA, "FIPA Agent Management Specification", <http://www.fipa.org>, 2000.
- [4] FIPA, "FIPA ACL Message Representation in XML Specification", <http://www.fipa.org>, 2000.
- [5] FIPA, "FIPA 98 Part 8 Version 1.0: Human-Agent Interaction Specification", <http://www.fipa.org>, 1998.
- [6] Benjamin N. Grosz and Yannis Labrou, "An Approach to using XML and a Rule-based Content Language with an Agent Communication Language", Proceedings of the IJCAI-99 Workshop on Agent Communication Languages (ACL-99), 1999
- [7] Simon St. Laurent, Ethan Cerami, Building XML Application, McGraw-Hill, p128-149, 1999.
- [8] S. Poslad, P. Buckle, R. Hadingham, "Open Source Standards and Scaleable Agencies", Barcelona, Spain, June 2000.
- [9] S. Poslad, M. Calisti, "Towards Improved trust and security in FIPA Agent Platforms", Barcelona, Spain - June 2000.
- [10] IKV++, "Grasshopper Basics and concepts", <http://www.grasshopper.de>, 2000.

Abstract

A Multi-Agent Platform for Providing Intelligent Medical Information

Won-Ki Choi*
Il Kon Kim**

Medical domain is very applicable for multi-agent system because medical information systems need much knowledge and close relationship with medical staff. In this paper, we describe design and implementation of an intelligent medical multi-agent platform that provides medical images' information services. This platform supports a physical environment that medical agents can be deployed following FIPA(Foundation for Intelligent Physical Agent)'s agent management reference model. To use a variety of components on Windows, COM(Common Object Model) interfaces and XML(eXtensible Markup Language) for encoding ACL(Agent Communication Language) are used for multi-agent communications. Since many kinds of diverse and close relationships with medical staffs are essential, a medical staff is conceptualized as an agent and integrated with multi-agent systems. Also it provides an infrastructure applicable to share necessary knowledge between human agents and software agents in order to make intelligent medical information services easier.

* Mobile C & C

** Dept. of Computer Engineering, Kyungpook National University