

유비쿼터스 네트워크 기반의 다중 에이전트 커뮤니케이션 구조

최정화^o 박영택

송실대학교 컴퓨터학과

cjh79@ailab.ssu.ac.kr^o, park@computing.ssu.ac..kr

A Structure for Multi-Agent Communication based on Ubiquitous Network

JungHwa Choi^o YoungTack Park

Dept. of Computer Science, Soongsil University

요 약

유비쿼터스 환경에서는 어떤 기기나 사물에 센서와 같은 컴퓨팅이 가능한 기기를 집어넣어 언제, 어디서든 커뮤니케이션이 가능하게 한다. 이러한 기기나 사물간의 커뮤니케이션은 시맨틱 웹의 의미적 연관성을 이용하여 해결할 수 있다. 컴퓨터가 이해할 수 있는 의미적 연결을 하기 위해서는 온톨로지와 같은 시맨틱 웹 환경요소를 이용하여 정보자원들 간의 의미적 연결이 필요하다. 이를 위해서 본 논문에서는 시맨틱 웹의 의미적 연관성을 이용하여 유비쿼터스 환경에서 컴퓨터가 스스로 정보자원들의 의미를 이해하고 동작할 수 있는 다중 에이전트 커뮤니케이션 구조를 제안한다. 본 논문에서 제안한 다중 에이전트 커뮤니케이션 구조는 에이전트들이 웹 온톨로지 언어를 기반으로 표현된 정보자원들 간의 연관성을 기반으로 자원의 의미를 이해하고, 기계들 사이에 커뮤니케이션이 가능하게 한다.

1. 서 론

물리공간에 고착돼 있던 공간개념을 뒤엎고 전자공간 속에 학교, 도서관, 쇼핑물을 집어넣은 개념이 현재의 웹이다. 그러나 여전히 물리공간에 남아 컴퓨터 속으로 들어갈 수 없는 자원은 많다. 유비쿼터스는 반대로 물리공간 속 모든 자원에 컴퓨터를 집어넣고 온라인 네트워크 상에 있으면 언제 어디서나 서비스를 받는 환경/공간을 의미한다.[1]

21세기 웹은 기계가 정보자원의 의미를 이해하고 이를 바탕으로 논리적 추론이 가능하게 하여 기계들 사이에 커뮤니케이션이 가능할 수 있는 웹으로 발전하게 될 것이다.[2] 이러한 구현이 1989년에 월드와이드 웹을 창안하였던 팀버너스 리(Tim Berners-Lee)가 제안한 차세대 웹이라 불리는 시맨틱 웹(Semantic Web)이다. 시맨틱 웹은 문서의 각 부분을 컴퓨터가 이해할 수 있는 형식으로 기술하여 복잡하게 얽혀져 있는 정보자원들 사이의 의미적 연관성을 통하여 다양한 정보를 보다 효과적으로 활용할 수 있게 해준다. 이러한 컴퓨터의 지능적인 처리가 가능하도록 웹 문서 내에 지식표현을 위한 온톨로지(Ontology)를 삽입하고 관계성과 추론규칙을 포함시킨다.[3]

본 논문에서는 웹에 표현된 문맥을 이해하기 위해 웹 온톨로지 언어인 OWL[4]을 사용한다. OWL은 DAML+OIL 웹 온톨로지 언어를 기반으로 하고 RDF/XML 구문 명세에서 정의된 RDF 트리플의 집합으로 구성되어있다. 또한 정의된 어휘를 사용하여 트리플을 위한 특정한 의미를 기술하여 온톨로지 언어 중에 가장 최근 버전으로 자원의 상호 운영성을 잘 반영한다.

유비쿼터스 환경이 구현되어 모든 사물에 힘이 깃들고 그것이 네트워크를 통해 서로 교신하게 될 때 실제적인 공간의 위치와 사용자 식별이 동시에 가능하게 된다. 또한 많은 자원들(동물, 식물, 시설 등) 하나하나에 주소를 부여하게 되어 다른 공간의

서비스 이용과, 사용자가 원하는 것을 자동으로 인터넷상에서 검색 및 제공하게 된다. 이러한 일은 인터넷 사용자 대신 에이전트가 컴퓨터나 인터넷상에 존재하는 정보 수집 및 분석을 수행하여 제공한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구에 대해 살펴보고 3장에서는 지능적인 서비스가 가능한 에이전트 커뮤니케이션 구조 및 메시지 흐름에 대해서 알아본다. 4장에서는 OWL 언어를 기반으로 작성된 온톨로지의 메시지 교환방식 설계에 대해 알아보고 마지막으로 결론과 향후과제를 논의한다.

2. 관련 연구

유비쿼터스 네트워크를 기반으로 생활의 모든 환경을 인터넷과 상호 융합하고자 하는 프로젝트는 이전에도 있어왔다. 사랑, 장치, 장소가 함께 하는 HP의 쿨타운(Cooltown)은 정보기간 또는 정보기기와 이용자에게 URL 생성 및 교환, 모든 개체(사람, 사물, 장소 등)가 웹 페이지를 보유하여 정보를 교환하는 방안을 제안하였다. 또한 URL 정보를 화면, 인쇄장치 등에 전송하여 각 구성요소를 상호 통합하기 위한 플랫폼으로 정보기간 간의 컴퓨팅으로 구현되기 때문에 정보기기의 호환성을 중시한다. 현재는 HP제품만 서비스가 지원되며 타 정보기기의 호환성을 위한 연구를 현재 진행하고 있다.[5]

이외 에이전트간에 메시지 요청 및 응답을 통해 사용자 요구를 예측하는 카네기 멜론 대학의 아우라(Aura)[6], 컴퓨팅 생활공간을 제공하는 MS의 이지리빙(Easyliving)[7] 등의 유비쿼터스 환경에 대비해 자원을 네트워크로 연결하고 자원의 감지 및 자동 처리하는 연구가 진행 중이다.

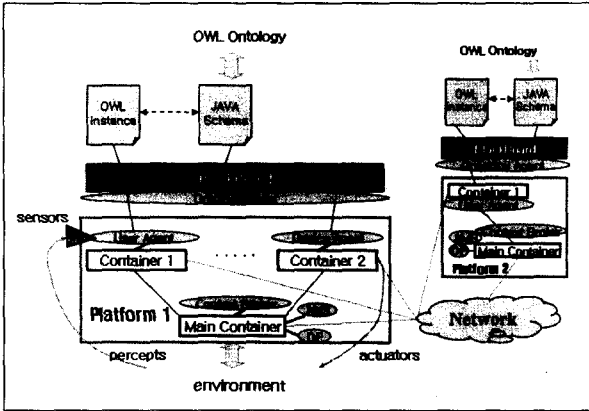
본 논문도 유비쿼터스 네트워크를 기반으로 하는 플랫폼 안에서 모든 자원의 컴퓨터를 에이전트로 보고 에이전트 간에 커뮤니케이션을 위한 시스템을 제안한다.

3. 다중 에이전트 커뮤니케이션 구조

3.1. 시스템 개요

지금까지 유비쿼터스 네트워크 기반의 연구는 자원에 네트워크가 되는 컴퓨터를 넣어 자원의 정보를 이해하고 웹 서비스를 함으로써 언제 어디서나 이용 가능한 네트워크 구축에 초점이 맞춰왔다.

물리공간에 존재하는 무수히 많은 자원들의 기능을 파악하지 못하고 정보를 주고받는다면 자원의 기능이나 사용자 프로파일 분석 시 판단상의 오류가 발생할 수 있다. 본 논문에서는 유비쿼터스 서비스를 사용자 요구에 맞게 서비스하기 위한 시맨틱 웹 기술 기반의 유비쿼터스 컴포넌트 소프트웨어 개발을 제안하며 시스템의 전반적인 흐름은 [그림1]과 같다.



[그림 1] 유비쿼터스 네트워크 기반의 시스템 구조 설계

[그림 1]에서와 같이 모든 자원은 네트워크로 연결되고, 유사 환경에 존재하는 자원들은 한 플랫폼으로 구성되며 그 안의 에이전트 간에 공유할 수 있는 블랙보드(Block Board)를 둔다. 또한 하나의 플랫폼은 특정기능의 공간(Space)을 컨테이너로 구분한다. 이 컨테이너에는 각기 다른 서비스를 하는 에이전트들이 존재하며 각 에이전트들은 블랙보드를 이용하여 커뮤니케이션 한다.

3.2. 시스템 구성 요소

각 자원에 존재하는 에이전트들을 네트워크로 연결하여 플랫폼을 구축하고 에이전트간의 메시지 교환방식을 설계 한다.

본 시스템은 플랫폼을 기반으로 여러 컨테이너로 구분되고 컨테이너에는 특정기능을 하는 에이전트들이 존재하게 된다. 다음은 이 각 구성요소들에 대해서 설명한다.

3.2.1 플랫폼

모든 자원은 네트워크 상에서 에이전트와 사용자를 이어주고 정보를 공유하는 플랫폼들로 구성된다.

3.2.2 컨테이너

하나의 플랫폼에는 하나의 메인 컨테이너가 존재한다. 메인 컨테이너는 플랫폼에 존재하는 모든 컨테이너를 관리한다. 그러기 위해 메인 컨테이너에는 두 가지 요소가 존재한다. 첫 번째 요소는 'White Page'로 에이전트를 생성 및 소멸하는 AMS(Agent Management Services)이고, 두 번째 요소는 'Yellow Page'로 에이전트가 제공하는 서비스 이름, 속성에 대한 명세서를 만들고 제어하는 DF(Directory Facilitator)이다.

각 컨테이너에는 특정기능을 수행하는 에이전트들이 존재하며, 본 논문에서는 크게 컨텍스트 브로커, 유저 에이전트, 디바

이스 에이전트, 온톨로지 에이전트로 구분한다.

3.2.3 컨텍스트 브로커

컨텍스트 브로커는 메인 컨테이너에서 물리공간의 사용자 인증과 전자공간의 에이전트와의 매개체 역할을 한다.

3.2.4 유저 에이전트

유저 에이전트는 사람을 추적하고 각 개인을 식별한다. 또한 사용자의 행동 및 그 상황에 맞는 다양한 정보에 근거하여 사용자를 대신하여 작업을 수행한다. 즉 자동적으로 상황을 판단 하면서 다양한 처리를 하게 된다.

3.2.5 디바이스 에이전트

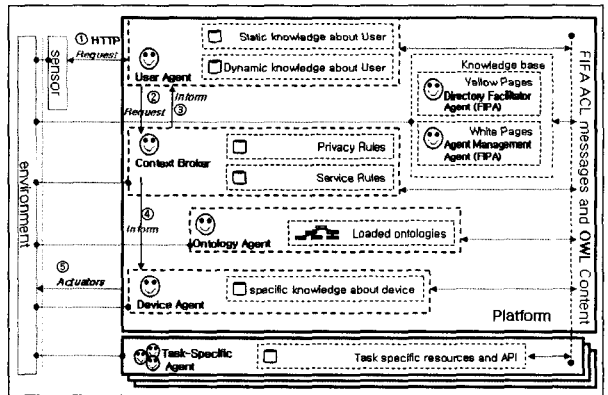
디바이스 에이전트는 장치(Device)기능에 따라 컨텍스트 브로커로부터 장치 작동 내역을 OWL형식으로 전달받아 작동한다. 디바이스 에이전트는 OWL 파일을 분석하여 실행 가능한 데이터형식으로 전환하고 장치를 작동한다.

3.2.6 온톨로지 에이전트

온톨로지 에이전트는 온톨로지를 보관하여 블랙보드(Black Board)를 구축한다. 또한 OWL 인스턴스(Instance)를 자바(Java) 스키마(Schema) 형식으로 변환 및 역변환 역할을 수행한다. 따라서 온톨로지의 추가적인 생성과 온톨로지 관리 및 사용자들의 프라이버시에 따른 정보를 제공하게 된다.

3.3. 메시지 흐름

유비쿼터스 네트워크로 연결된 플랫폼에서의 에이전트간의 메시지 흐름은 [그림 2]와 같다.



[그림 2] 에이전트 메시지 구조

[그림 2]에서와 같이 물리공간(Environment)으로부터 자원을 감지하여 에이전트들 간에 OWL 메시지를 통해 웹 서비스를 하게 된다. 예를 들어보면 유저 에이전트가 사람(User)을 감지하고 감지한 사람의 에이전트로부터 정보(Knowledge)를 받아 내어 컨텍스트 브로커에게 웹 서비스를 받아도 되는 사람인지 인증을 요청한다. 컨텍스트 브로커는 온톨로지를 통하여 그 사람에게 부여할 권한과 정보, 서비스내역을 유저 에이전트에게 알려준다. 또한 해당 서비스의 디바이스 에이전트에게 제공할 서비스를 OWL 인스턴스를 생성하여 전달한다. 디바이스 에이전트는 전송받은 OWL 인스턴스를 분석하여 장치를 작동하게 된다. 이와 같은 에이전트간의 메시지를 통해 장치가 스스로 적절한 서비스를 감지하여 사람의 개입 없이 다른 서비스 제공자에게 장치 자신의 역할을 등록하고 자동 서비스 하게 된다.

4. 메시지 교환 방식 설계

4.1. 인터페이스

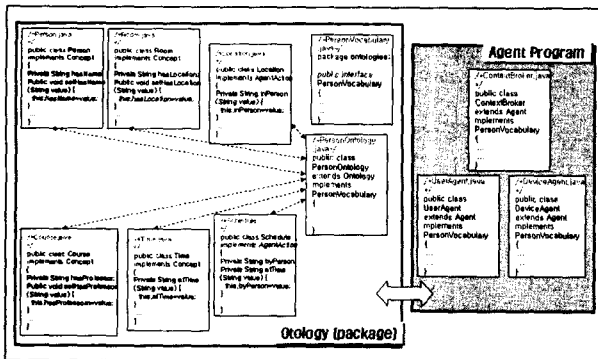
한 컴퓨터에서 에이전트들의 메시지 교환을 통해 다양한 플랫폼을 가로질러 다른 컴퓨터로 실행상태를 옮길 수 있다. 이러한 처리는 컴퓨터가 스스로 처리하게 되는데 시맨틱 웹의 기능을 사용함으로써 자동화된 기능이 수행되게 된다. 또한 온톨로지 사용으로 인한 사람들의 이름, 위치정보, 권한, 개인적 취향에 대한 정보를 공유하게 된다. 따라서 사람이 한 장소에서 다른 장소로 이동하면 다른 장소와 그 사람의 위치정보에 기입된 장소를 연결하는 에이전트가 존재하게 된다. 에이전트들은 일반적으로 특정 환경에서 일어나는 업무를 관리할 수 있는 프로그램이다. 따라서 방이나 복도를 걷는 사람이 가장 가까운 장치를 자동으로 이용하게 하는 기능들이 포함되어 있다.

4.2. OWL 컨텍스트 메시지 전송을 위한 온톨로지 구조

자동적 행위를 발생시키거나 행위에 대한 관계를 위해 OWL을 기반으로 온톨로지를 구축한다. OWL 속성(Attribute)을 이용하여 인스턴스 간의 특징을 파악하여 프라이버시(Privacy)를 보장한다. 주요 온톨로지의 구성은 다음과 같다.

- Physical Context 온톨로지 : Location, Time, Room
- Environmental Context 온톨로지 : Light, Sound
- Device Context 온톨로지 : Media Device 및 기타 Device
- Personal Context 온톨로지 : Person, Schedule

위와 같은 컨텍스트 온톨로지를 OWL형태로 표현한다. 또한 사용자와 유비쿼터스 디바이스 온톨로지를 계층적 구조로 설계하고 이를 에이전트가 활용할 수 있도록 시스템을 설계하였다. 이 시스템에서 작성한 온톨로지 구조는 [그림 3]과 같다.



[그림 3] 메시지 전송을 위한 온톨로지 구조

에이전트 간에 전송하게 되는 메시지 내용은 [그림 4]와 같이 OWL형식으로 인스턴스를 전송하게 된다. 메시지를 전송받은 디바이스 에이전트는 온톨로지를 기반으로 분석하여 장치를 작동하게 된다.

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://ailab.ssu.ac.kr/ontology/ubiProject/ontologies#"
  xmlns:rdfs="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://ailab.ssu.ac.kr/ontology/ubiProject/ontologies#">
  <Person rdf:ID="choi">
    <hasName>Jung-hwa </hasName>
    </Person>
    <Schedule rdf:ID="choi">
      <byPerson>choi </byPerson>
      <duringTime rdf:resource="2"/>
      <hasContent rdf:resource="#1 Project Seminar"/>
    </Schedule>
</rdf:RDF>
```

[그림 4] 디바이스 에이전트가 전송받은 OWL 인스턴스

본 시스템은 Java Class로 구성된 온톨로지를 이용하여 Person, Device, Operation등의 정보를 인식하고 수행된다. 모든 온톨로지 스키마가 정의된 PersonOntology 파일을 통하여 에이전트 프로그램은 온톨로지 기반의 메시지를 전달받게 된다. 온톨로지는 크게 Concept, Predicate, AgentAction 클래스로부터 상속받아 사용된다. Concept으로 정의된 스키마는 블랙보드에 들어갈 메타데이터(Meta Data)를 정의하고, 각 스키마의 인스턴스를 기능별로 구분하여 AgentAction 클래스에 스키마로 정의하여 에이전트 프로그램이 동작되게 된다. Predicate 클래스로부터 상속받은 클래스는 에이전트의 동작을 명시하게 된다.

4.3. 시스템 동작의 원리

이 시스템은 강력한 보안(Security) 기능과 풍부한 네트워크 관련 기능이 탑재되어있는 Java를 기반으로 특정 OS에 의존하지 않고 어떤 플랫폼에서도 애플리케이션을 실행할 수 있는 JADE (Java Agent Development Framework)[8]를 사용하여 구성하였다. JADE는 다중 에이전트 시스템 개발에 유용한 미들웨어(Middleware)로서 FIFA 표준에 따른 지능형 에이전트를 구현하기 위한 소프트웨어 개발 프레임워크이다.[9] 이 플랫폼을 이용하여 에이전트간의 네트워크가 형성되게 된다. 그리고 디바이스 간에 상호작용을 위해 FIFA 표준 온톨로지 언어인 OWL을 사용하여 에이전트 간에 메시지를 전송한다.

5. 결론과 향후 과제

본 논문에서는 유비쿼터스 서비스를 사용자 요구에 맞게 서비스하기 위해 시맨틱 웹의 의미적 연관성을 이용하였다. 또한 온톨로지를 활용하고 지능적인 서비스를 제공하기 위해 에이전트 간에 메시지 구조를 정의하였다. 이를 위해 네트워크를 통한 전달은 OWL을 기반으로 하고, 기능별로 에이전트를 구분하여 메시지에 따른 서비스를 하도록 시스템을 구현하였다.

향후 이 소프트웨어로 제어되는 기구와 센서 등에 대한 하드웨어 기술이 통합된다면 유비쿼터스 환경이 실현 가능하게 된다. 이 시스템을 위해서는 위치추적 및 예측을 위한 온톨로지에 명시된 데이터에 새로운 데이터가 들어오면 자동으로 인식하고 추론하는 연구가 요구된다. 또한 사용자 프라이버시와 보안을 위한 연구도 수행 되어야 할 것이다.

6. 참고 문헌

- [1] 김동환, 유비쿼터스 공간의 경제와 경영 전략
- [2] 노무라총합연구소 저, 박우경, 김의역/하원규 강역, 유비쿼터스 총서 3권 「유비쿼터스 네트워크와 신사회 시스템」 전자신문사, 2003.2.20.
- [3] Tim Berners-Lee, James Hendler and Ora Lassila, "The Semantic Web", Science American, May 2001.
- [4] OWL Web Ontology Language XML Presentation Syntax, W3C Note 11 June 2003.
- [5] Cooltown research/ A Web-Based Nomadic Computing System
- [6] Project Aura : Toward Distraction-Free Pervasive Computing, 2002.
- [7] EasyLiving : Technologies For Intelligent Environments
- [8] JADE Tutorial and Primer, Jean Vaucher and Ambroise Ncho Dep. K'informatique Universite de Montreal, September 2003.
- [9] Akio SASHIMA, Non-member, Koichi KURUMATANI, Non-member, and Noriaki IZUMI, "Physically-Grounding Agents in Ubiquitous Computing"