

유비쿼터스 컴퓨팅 환경에서의 온톨로지 기반 상황 인식 미들웨어

정 현 만*

Context-Aware Middleware based on Ontology in Ubiquitous Computing Environment

Jung Heon Man*

요 약

유비쿼터스 컴퓨팅 환경에서 상황 인식 기반의 서비스를 제공하기 위해선 동적인 상황 관리 기술과 상황 추론 기술, 그리고 상황 모델링 기술이 필요하다. 본 논문에서는 유비쿼터스 컴퓨팅 환경에서 상황 변화에 대한 사용자의 요구사항을 능동적으로 반영하고, 동적인 응용 적응성을 지원하는 계층적 온톨로지 기반 상황 관리 모델을 제안하고 이를 기반으로 상황 인식 미들웨어를 설계한다. 또한, 상황 인식 서비스 구현을 위해 다양한 컨텍스트 발견, 획득, 해석, 추론을 효과적으로 지원하며 사용자의 서비스 실행 시 발생할 수 있는 상황 충돌을 해결하기 위한 방법을 제시한다. 이 논문에서 제시한 계층적 온톨로지 기반 상황 인식 미들웨어는 유비쿼터스 환경에서 요구되는 다양한 상황 인식 서비스의 개발 및 운용을 효과적으로 지원 할 수 있다.

Abstract

To support service based on context-aware in ubiquitous computing environment, there are required context managing, context reasoning and context modeling technologies. In this paper, we propose a hierarchical ontology-based context management model and design a context-aware middleware based on this model for supporting active application adaptability and reflecting users' requirements dynamically in contextual changes. It also provides efficient support for inferencing, interpreting, acquiring and discovering various contexts to build context-aware services and presents a resolution method for context conflict which is occurred in execution of service. The proposed middleware can support the development and operation of various context-aware services, which are required in the ubiquitous computing environment.

- ▶ Keyword : 상황 인식 미들웨어(Context-Aware Middleware), 상황 인식 서비스(Context-Aware Service), 상황 온톨로지(Context Ontology), 상황 모델(Context Model), 유비쿼터스 컴퓨팅(Ubiquitous Computing)

1. 서론

유비쿼터스 컴퓨팅은 일상 생활 속에 편재해 있는 컴퓨팅 자원을 이용하여 사용자가 언제 어디서나 동적인 서비스를 받을 수 있는 환경을 제공하며 조용한 컴퓨팅(calm computing)[1], 보이지 않는 컴퓨팅(invisible computing)[2], 사라지는 컴퓨팅(disappearing computing)[3] 등의 용어는 유비쿼터스 컴퓨팅에 관한 사용자 인터페이스 관점을 잘 설명해 주고 있다.

유비쿼터스 환경에서는 기존 컴퓨팅 환경에서의 사용자와 컴퓨터간의 대화형 상호작용이 아닌 물리적인 환경, 상황(context)등을 시스템이 인식하고 이를 기반으로 사용자와의 상호 작용을 지원하는 상황 인식 기술이 필수적인 요소로 자리 잡고 있다. 또한 상황 인식 서비스는 다양한 상황 정보를 수집 및 해석을 통해 인식하고, 추론과정을 거쳐 기존의 수동적인 서비스에서 벗어나 사용자 명령 없이도 자동으로 실행되는 지능형 서비스를 지원하며, 각 사용자에게 맞춰진 개인화된 서비스 등을 제공한다. 상황 인식 서비스는 의료, 교육, 재난, 구호, 쇼핑 등 사회 전 분야에 걸쳐 응용될 수 있어 사회 전반에 걸쳐 많은 영향을 줄 것이다[4,5].

기존의 상황 인식 서비스 기반 기술 중 상황에 대한 체계적 관리를 위한 온톨로지 기술과 사용자 맞춤형 서비스 제공을 위한 추론 기술에 대한 연구는 어느 정도 진행되고 있으나 사용자 사이에 발생하는 서비스 충돌 해결에 대한 연구는 부족한 실정이다[4,5,6,7].

본 논문에서는 상황 인식 미들웨어의 설계 및 구현 시 필요한 기반 기술에 대해 알아보고 계층적 온톨로지 기반의 상황인식 미들웨어를 설계한다. 제안한 모델은 사용자의 행동 및 환경의 변화에 따른 센서 입력 데이터를 바탕으로 온톨로지를 구성하고 상황 정보를 추론하여 서비스를 실행한다. 추가적으로 사용자 사이의 서비스 지원 과정에서 발생할 수 있는 상황 충돌(Context Conflict)을 해결할 수 있도록 설계하였다.

2. 관련 연구

이 장에서는 유비쿼터스 컴퓨팅에서 필수적으로 요구되는 상황 인식과 관련된 기술과 기존 연구에서 이루어진 상황에 대한 정의와 모델링 방법, 상황인식 서비스 인프라에 대해서 기술한다.

2.1 상황 인식 개요

상황에 대한 다양한 정의가 있지만 사용자와 유비쿼터스 컴퓨팅 환경 사이의 관계와 연관지어지는 사용자 주위의 상황이나 상태(circumstance) 또는 객체(object)들에 대한 정보를 통칭한다[4]. 상황의 본질적인 정의는 "세계(Real World)에 존재하는 실체(Entity)의 상태를 특징화하여 정의한 정보"라고 정의할 수 있으며, 여기서 실체란 인간, 장소 또는 사람과 서비스간의 상호 작용을 의미한다고 할 수 있다.

Schilit와 Theimer는 그들의 연구에서 "상황 인식"이란 용어를 처음으로 사용하였는데, 그 연구에서는 상황을 장소, 사람이나 사물들을 구별 짓는 특징인 아이덴티티(identity), 사람이나 사물들을 포함하는 환경의 변화 등으로 설명한다[5]. Dey는 상황을 사용자가 속해 있는 환경 내에서 사용자의 감정적인 상태, 주의력, 위치와 방향, 날짜와 시간, 사람과 사물

등으로 정의한다[6].

위와 같이 관점에 따른 상황 정의에 약간의 차이가 있으나 일반적인 상황 정보는 사용자 상황, 물리적 환경 상황, 컴퓨팅 시스템 상황, 사용자-컴퓨터 상호 작용 이력, 기타 상황으로 분류할 수 있으며 사용자의 현재 상황에 따라 적절한 정보 혹은 서비스를 제공하기 위해 상황을 이용하는 것을 상황인식(Context-Awareness)이라 한다[4,7].

2.2 상황 정보 모델링 기술

상황 모델링은 상황 정보에 대한 높은 수준의 추상적 개념을 제공하기 위해 필요하며 센서와 액추에이터(actuator)의 추상화를 제공함으로써, 개발자가 다양한 하드웨어 장치와 인터페이스하는 부담을 줄일 수 있다. 이러한 상황 모델링 기법은 각 시스템에서의 상황 정보 교환을 위해 사용되는 데이터 구조에 관한 스키마에 의해 다음의 네 가지로 분류된다(4,5,6,7).

키-값(Key-Value) 모델은 상황 정보를 모델링하기 위한 가장 간단한 데이터 구조이다. 환경 변수로서 상황 정보(예, 위치 정보)의 수치(value)를 어플리케이션에게 제공하는데, 키-값 쌍의 형식을 사용하여 상황을 모델링하며 Key-Value 모델은 관리하기 쉽지만, 효과적인 상황 검색 알고리즘을 위한 복잡한 구조화가 부족하다.

마크업 스키마(Markup Scheme) 모델링 기법은 속성과 내용을 갖는 마크업 태그로 구성된 계층적 데이터 구조이다. 특히, 마크업 태그의 내용은 항상 다른 마크업 태그에 의해 재귀적으로 정의될 수 있다. 이러한 종류의 상황 모델링 기법에서는 전형적으로 프로파일이 대표적이다. 프로파일들은 항상 XML과 같은 모든 마크업 언어의 슈퍼클래스인 SGML에서 유도된 직렬화에 기반한다.

객체 지향 모델링 기법은 유비쿼터스 환경에서 상황의 동적 특성에 관한 문제점들을 해결하기 위해서, 캡슐화와 재사용이라는 객체 지향의 주요 장점을 적용한다. 상황 처리에 관한 자세한 내용은 객체 수준에서 캡슐화되므로, 다른 컴포넌트들에게 은닉되고 상황 정보의 접근은 특정 인터페이스를 통해서만 제공된다.

온톨로지 기반 모델링은 온톨로지는 개념과 상관 관계를 기술하는 도구로 상황정보를 표현하고 공유하기 위한 어휘 및 용어를 제공하며 상황 정보의 다양성 때문에 도메인 기반 온톨로지(domain specific ontology)를 정의하고 있다. 이 모델은 유비쿼터스 컴퓨팅 환경에서 상황 지식의 공유와 재사용을 지원하며, 또한 계층적 상황 온톨로지 모델에서는 상위 계층의 온톨로지를 이용하여 도메인에 맞는(domin-specific) 하위 계층의 온톨로지를 생성해 낼 수 있다.

2.3 상황 정보 센싱 기술

유비쿼터스 환경에서 사용자의 위치 정보는 상황 인식 서비스에서 중요한 상황 정보로 위치 인식 시스템을 인식 가능 영역에 따라 분류하면 다음과 같이 세 가지로 분류할 수 있다.

매크로 위치인식 시스템은 가장 광범위한 위치인식 가능 영역을 제공하며, 현재 위치기반 서비스(Location Based Service)를 위해 GPS(Global Positioning System)와 이동통신망 기반 위치인식 시스템이 활용되고 있다.

마이크로 위치인식 시스템은 무선 환경의 제한으로 매크로 위치인식 시스템이 커버하지 못하는 실내나 지하 또는 건물 밀집지역 등에서 위치인식을 제공하며, 위치인식을 위해 사용되는 매체에 따라 적외선 기반 시스템(Active Badge), 초음파 기반 시스템(Active Bat, Cricket), 무선랜 신호 기반 시스템(RADAR, 3D-iD), 영상 기반 시스템(Easy Living) 등이 있다.

Ad hoc 위치인식 시스템은 무선 센서 네트워크와 같이 위치인식을 위한 추가적인 하드웨어의 도입이 어려운 환경에서 위치정보를 가지는 레퍼런스 노드와의 무선 링크 연결성만으로 위치를 계산한다. 언제, 어디에서나 그리고, 다양한 응용에 적합한 위치 정보를 수집하기 위해서는 여러 위치인식 시스템간의 통합이 요구된다.

위에서 언급한 상황 정보 모델링 및 센싱 기술 외에 상황 정보 융합 및 추론기술(Context fusion & Reasoning), 상황 정보 교환 기술 (Context exchange), 상황인식 서비스 설명 및 발견기술(Context-aware service description language)등이 상황 인식 서비스 인프라에서 제공되어야 하는 중요한 기술이다.

2.4 상황 인식 서비스 미들웨어

현재 상황 인식과 관련된 연구는 상황 인식 미들웨어와 미들웨어를 이용한 서비스 즉 응용으로 분류할 수가 있다.

상황 인식 서비스 미들웨어와 관련된 연구에서는 상황인식 응용 개발에 필요한 공통 기능을 응용 레벨에서 분리하여 미들웨어 형태로 개발자에게 제공하는 방안이 연구되고 있으며, 개발자에게는 응용과 관계된 기능에만 집중할 수 있도록 하고, 사용자에게는 일반화된 응용 서비스를 제공한다.

이 절에서는 상황 인식 미들웨어와 응용 서비스와 관련된 대표적인 연구에 대해 요약하였다.

(1) 상황 인식 미들웨어

Maryland Univ.의 CoBrA(Context Broker Architecture)[7]는 유비쿼터스 환경에서 상황 정보의 생성, 관리, 배포를 위한 아키텍처로 공간상에 존재하는 모든 컴퓨팅 객체(computing entity)들을 위한 상황 공유 모델을 관리하는 broker agent 개발을 목표로 하고 있다. Policy language를 통해 사용자 및 기기의 상황 정보에 대한 접근 관리를 제공하며 상황을 RDF와 OWL[8]을 기반으로 온톨로지로 구성하여 지식 공유 및 기본적인 추론이 가능하도록 설계 하였다.

Singapore Univ.의 SOCAM(Service Oriented Context-aware Middleware)[9]은 유비쿼터스 환경에서 제공되어지는 다양한 상황의 상호의존적 개념에 대한 정의 및 모바일 환경에서의 상황인식 서비스 제공을 위해 구현되었으며 OWL 기반의 온톨로지를 사용하여 의미기반 상황 표현과 다양한 형태의 상황에 대한 추론, 지식 공유, 상황의 분류와 상호 의존성과 관련된 문제를 다루고 있다.

Georgia Tech.의 CASA(Context-Aware Security Architecture)[10]는 이동컴퓨팅 환경에서 다양한 네트워크 및 서비스, 장치와의 접속 제어 및 권한 부여와 같은 보안 서비스를 투명하게 제공하기 위하여 사용자 생체정보 및 인식 기술 또는 Active Badge 등과 같은 센서를 이용하여 사용자의 ID, 위치, 역할 등을 인식하고 사용자를 인증하는 상황인식 보안 서비스 인프라를 제공한다. 정책 설정을 위한 GPD(L(Generalized Policy Definition Language)과 사용자 인식, 시간, 장소 등에 따라 적절한 권한 부여를 위해 GRBAC(Generalized Role-Based Access Control) 모델을 제안하고 있다.

Illinois Univ.의 Gaia[11]는 응용이 다양한 상황정보를 얻고 추론할 수 있게 해주며, 상황 처리를 위해 논리 추론과 기계 학습 방법이 폭넓게 활용되며, 서로 다른 유비쿼터스 컴퓨팅 환경뿐만 아니라 이종 에이전트간의 시맨틱한 상호 운용성을 보장하기 위해서 DAML(Darpa Agent Markup Language)로 기술된 온톨로지를 사용한다. Gaia상의 모든 에이전트는 CORBA 환경 위에서 작동되며, 이를 위해 CORBA의 다양한 미들웨어 서비스를 활용하고 있다.

Lancaster의 GUIDE[12] 프로젝트는 영국 Lancaster를 방문한 여행객의 효율적인 여행 서비스 제공을 위해 여행객에게 펜기반 테블릿 PC를 제공하여 여행 가이드를 하는 위치기반 서비스 시스템이다.

GATECH의 ContextToolkit[13]은 상황 인식 미들웨어의 출발점으로 상황과 사용자의 입력 값을 분류하고 상황을 사용자, 장소, 상태, 시간 등 4개의 상황 타입으로 정의하였다. 특징은 센서로부터 취득한 상황 정보를 표현 및 통합을 한 후 애플리케이션에게 제공한다.

(2) 상황 인식 서비스

Microsoft의 Easy Living[14, 15]은 사용자 신원, 위치, 대상물 인식 정보를 이용하여 정보 가전기기를 제어하는 지능형 가정환경을 구축한다. 객체와 사람, 공간사이의 물리적 관계의 정보를 상황정보화 하여 사용함으로써 상황을 추론하고 사용자가 원하는 서비스를 제공한다.

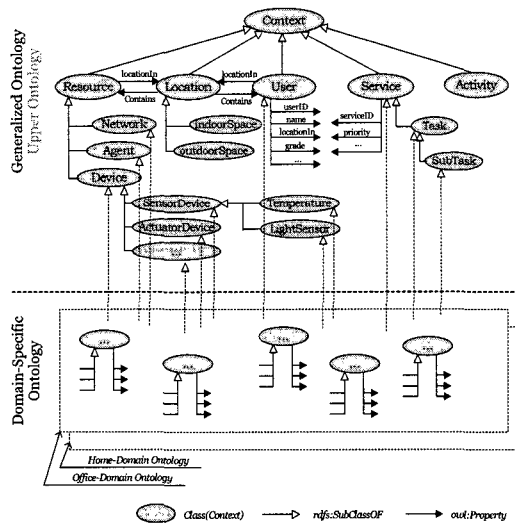
Georgia Tech. Aware Home Research Initiative(AHRI)의 AwareHome[16][17]은 가정내 사용자의 상황(누가, 어디에서, 무엇)을 파악하는 연구로 실내에서의 위치 인식을 위해 RFID와 마루 메트를 이용했으며, 혼자 사는 노인들의 위치 정보와 활동 상태 정보를 파악하여 병원이나 보호자에게 알려주는 스마트 홈의 모델을 제시하였다

Musex[18]는 박물관에서 관람하고 있는 어린이를 대상으로 실시한, 학습자 위치 인식에 맞춰 학습자 부근의 전시물과 관련된 콘텐츠를 RFID가 부착된 PDA(Personal Digital Assistants)에게 퀴즈 형태로 제공하고 사용자의 흥미를 유발시키도록 하는 연구이다.

현재 상황 인식 서비스에 대한 다양한 연구가 진행되고 있으나 사용자와 환경의 상황을 반영한 사용자간에 발생하는 서비스 충돌 해결 및 불확실하고 모호한 상황 데이터의 처리에 대한 연구가 부족한 실정이다.

3. 상황 인식 모델 및 미들웨어 설계

온톨로지를 기반으로 상황을 모델링함으로써 쉬운 추론 및 패턴 분석, 사람과 기계, 기계들간 원활한 커뮤니케이션, 확장성 및 상황정보 공유의 용이성, 다양한 응용 서비스 관리의 편리함을 얻을 수 있으며 계층적 상황 온톨로지 모델을 구성하여 상위 계층의 온톨로지를 상속받아 도메인에 적합한(domin-specific) 하위 계층의 온톨로지를 생성해 낼 수 있다. 이 장에서는 2-계층 상황 온톨로지 관리 방법을 제안한다.



[그림 1] 2-계층 상황 온톨로지 구성

3.1 온톨로지 기반 상황 모델

온톨로지는 개념(concept)과 관계(relation)들로 구성된 사전으로서 특정 도메인에 관련된 객체들을 계층적 구조로 표현하고 추가적으로 이를 확장할 수 있는 추론 규칙을 포함한다. 유비쿼터스 환경에서는 모든 객체들과 객체의 속성들을 특정 종류의 도메인 안에 계층적으로 표현해야 하고, 이벤트 상황에 따라서 속성 값들을 변화시키는 규칙이 필요하므로 온톨로지를 사용하여 유비쿼터스 환경내의 모든 객체들을 계층적으로 구성, 관리해야 한다.

그림 1은 이 논문에서 설계한 2-계층 상황 온톨로지를 표현한 것이다. 미들웨어상에서 관리되는 상황 정보는 상위 계층의 일반화된(generalized) 온톨로지 상황 정보와 도메인별 특성에 맞게 상위 온톨로지를 상속하여 별도로 정의되는 하위 도메인 상황 정보로 구성되며, 이들은 웹 온톨로지 언어인 OWL로 온톨로지를 정의한다. 서비스가 미들웨어에 새롭게 배치되는 경우에는 미들웨어는 하위 도메인 상황 정보에 정의하여 상황 정보를 운용할 수 있어야 한다.

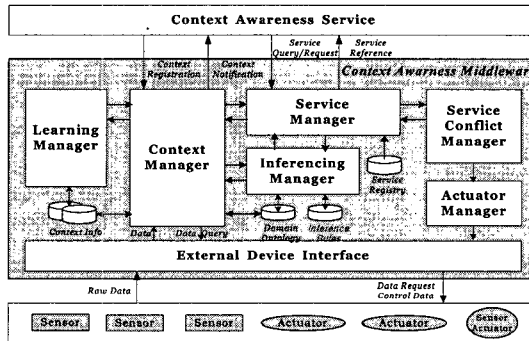
모든 온톨로지는 Context 클래스에서 상속받아서 만들어지며 상위 온톨로지에는 최상위 클래스인 Context 클래스에서

상속받은 자원(Resource), 위치(Location), 사람(User), 서비스(Service), 행위(Activity) 클래스와 도메인별 상황 정보는 상위 클래스에서 상속받은 여러 하위 클래스로 구성된다.

각 상황 클래스는 해당 상황을 설명하기 위한 다수의 속성을 포함할 수 있으며, 이 속성은 다른 클래스와의 관계(relation)를 나타낼때도 사용되며, 다른 상황을 프로퍼티로 포함할 수도 있다. 예를 들어서, 홈네트워크 응용 온톨로지에서 Location 클래스로부터 상속받은 indoorSpace의 Room 상황과 Resource의 하위 클래스인 Device 클래스로부터 상속받은 Temperature 상황이 있는 경우 Temperature 상황은 Room 상황의 프로퍼티로 이용될 수 있다.

3.2 상황 인식 미들웨어 설계

본 절에서는 온톨로지 기반 상황 인식 관리 모델을 기반으로 상황 인식 미들웨어를 설계한다. 상황 인식 미들웨어는 센서로부터 받은 데이터를 이용해서 상황 정보를 관리, 조합, 학습, 추론하여 이 정보를 원하는 상황 인식 서비스에 이들을 전달하는 역할을 담당한다.



(그림 2) 상황인식 미들웨어 구성도

설계한 미들웨어는 상황 관리자(Context Manager), 서비스 관리자(Service Manager), 추론 관리자(Inferencing Manager), 서비스 충돌 관리자(Service Conflict Manager), 학습 관리자(Learning Manager), 액츄레이터 관리자(Actuator Manager), 외부장치 인터페이스(External Device Interface)로 구성되며 전체 구성도는 그림 2와 같다.

(1) 상황 관리자

상황 관리자는 기본 상황을 생성하는 상황 생성기(context generator)와 기본 상황을 저장, 관리하고 복합 상황 추론을 위해 추론 관리자와 상호 작용을 하는 온톨로지 관리 모듈(ontology management module), 그리고 상황 인식 서비스와 공급/등록 방식으로 이벤트 전달을 하는 이벤트 브로커(event broker), 그리고 상황 저작자와 상황 인식 서비스로부터 전달받은 명령을 해석하기 위한 번역기(translator)로 구성된다.

저작자는 2-계층 상황 온톨로지를 이용해서 도메인에 필요한 상황 온톨로지를 OWL 파일로 정의하고 이를 번역기에 전달한다. 상황 생성기는 사용자의 명령이나 위치이동 등과 같은 환경(센서)으로부터 얻어진 입력 데이터와 데이터베이스에 저장되어 있는 상황 정보를 입력으로 하여 OWL parser를 이용해서 적절한 기본 상황을 생성, 저장 관리하고 사용자 서비스를 실행하기 위해 액츄레이터 관리자를 통해 외부 디바이스 제어 명령을 출력한다. 기본 상황 정보는 온톨로지 관리 모듈을 통해서 이벤트 브로커에게 전달되어서 상황 인식 서비스에 전달되고, 복합 상황 추론을 위해서 Jena를 이용하여 Fact인 nTriple형태로 변환된 뒤 추론 관리자에 전달된다.

(2) 서비스 관리자

서비스 관리자는 미들웨어에 연결된 상황 인식 서비스들간의 상호 작용을 지원하기 위해서 서비스 검색 및 등록, 조합, 삭제 기능을 제공하며 다중 사용자의 서비스 요구를 처리한다. 사용자로부터 서비스에 대한 요청이 들어오면 서비스의 종류와 사용자 프로파일, 환경의 상황 정보에 따라 서비스를 제공하기 위한 계획을 세우고 서비스 객체를 생성한 후 SubTask들로 Task를 구성한다. Task가 서비스에 등록되면 서비스 충돌 관리자를 통해 충돌 여부를 확인하고 Task를 실제 액추레이터의 동작을 관리하는 액추레이터 관리자에 등록한다.

서비스는 Task의 묶음으로 한 명의 사용자가 하나 이상의 서비스 객체를 가지며, Task는 일정한 목표를 가지고 있는 서비스 단위로 하나 이상의 Task 또는 가장 작은 서비스 단위인 SubTask로 구성된다. 예를 들어 'TV 시청'이라는 서비스는 'TV Power_on', 'Light Turn_off'라는 Task와 'TV 채널 11', '소리크기 15'라는 SubTask로 구성된다. 서비스 팩토리(Service Factory)는 서비스 객체 및 SubTask, Task 객체를 생성하고 상태 변화를 관리한다.

(3) 추론 관리자

기존 추론엔진이 단순한 상황 브로커나 온톨로지에 기반한 상황 쿼리 엔진(Context Query Engine) 형태인 것에 비해 추론 관리자는 온톨로지 추론을 위해서 Jena API를 이용하며 내부적으로 OWL 상황 온톨로지 파일을 파싱한 후 온톨로지 관리 모듈을 통해 추론 관리자에서 룰 기반의 추론 시스템인 JESS(Java Expert System Shell)로 복합 상황 추론을 한다. 이때, 기본 상황 생성기에서 발생한 기본 상황을 Fact의 형태로 추론엔진에 추가 및 삭제하며, 상황 저작자가 정의한 규칙들과 온톨로지를 이용해서 온톨로지 추론과 사용자 규칙 기반 추론을 통해서 상황 정보의 일관성 검사와 상황 관계성을 도출하고 룰에 따른 최적 서비스 검출 및 서비스 충돌 검사를 보조한다. 추론관리자는 상황 인식 서비스에서 현재 컨텍스트 정보를 요청했을 때, 그 정보가 존재하지 않을 경우 추론을 통해 새로운 정보를 생성한다.

(4) 서비스 충돌 관리자

사용자에 대한 서비스 제공 시에 발생하는 Task, subTask의 충돌을 관리하며 계층적인 서비스 상황 온톨로지와 사용자 등급, 공간별 소유자(Owner), 공간별 사용자 우선순위(priority), 서비스/Task 우선순위, 디바이스 우선순위를 조합한 우선순위 기반의 서비스 룰을 구현하여 충돌을 해결한다.

서비스 충돌은 한 공간에서 다중 사용자가 같거나 유사한 Task들을 동시에 제공받으려 할 때 발생하는 다중 사용자간의 서비스 충돌과 여러 Task가 한 사용자에게 의해 동시에 발생할 때 일어나는 Task 충돌(취침서비스 + TV보기 서비스), 동시에 하나의 기기를 서로 다른 서비스가 제어하려 하거나 또는 비슷한 환경을 제어(온도를 올리는 방법 : 보일러 ON, 에어컨 OFF, 창 OPEN/CLOSE)하려고 하는 기기 사이의 충돌인 SubTask 충돌로 분류된다.

(5) 학습 관리자

학습 관리자는 동적으로 변화하는 환경에서 사용자의 행동 패턴 및 프로파일을 기반으로 사용자에게 적절한 서비스를 제공하기 위해서 신경망(neural network), 결정트리(decision tree), 연합룰 찾기(association rule finding) 등을 적용 시켜서 기계학습을 수행하며 액추레이터 관리자는 충돌관리자를 통해 서비스 충돌의 위험이 없는 서비스가 등록이 되면 서비스를 수행하기 위해 외부 디바이스 인터페이스 모듈로 제어 신호를 보낸다.

4. 결론

본 논문에서는 유비쿼터스 컴퓨팅 환경에서의 상황 인식 미들웨어를 설계 및 구현하는데 필요한 기술들에 대해 살펴보고 최근 연구되고 있는 상황인식 미들웨어와 서비스 응용에 대한 연구에 대해 검토하고 2-계층 온톨로지 기반의 상황인식 미들웨어를 제안했다.

제안한 온톨로지 기반 상황 모델은 일반화된 상위 상황 온톨로지를 상속받아 응용에 적합한 하위 온톨로지를 생성할 수 있으며 동적 상황 변화에 대해 유연하게 인식하고 변경할 수 있다. 또한, 사용자와 환경의 상황 변화에 따라 적절한 서비스를 선택실행하고 서비스 지원 과정에서 발생할 수 있는 사용자간의 상황 충돌 해결을 지원할 수 있도록 다양한 우선순위에

기반한 서비스 충돌 관리자를 설계하였다

이 논문에서 제시한 2-계층 온톨로지 기반 상황 인식 미들웨어는 홈 네트워크, 스마트 오피스 등의 실내 환경 및 다양한 유비쿼터스 환경에서 동적 상황 인식 미들웨어로 적용될 수 있을 것으로 보인다.

향후 연구 방향은 정확하지 않은 센싱 데이터로 인해 발생하는 불확실(imperfect)하고 모호한(uncertain) 상황에 대한 추론 및 처리에 대한 연구이다.

참고문헌

- [1] M. Weiser and J. S. Brown, The Coming Age of Calm Technology, In P. J. Denning & R. M. Metcalfe (Eds.), Beyond Calculation: The Next Fifty Years of Computing, pp. 75-85, 1998.
- [2] A. D. Norman, The Invisible Computer: Why Good Products Can Fail, The Personal Computer Is So Complex, and Information Appliances Are the Solution, MIT Press, 1998.
- [3] J. Wejchert, "The Disappearing Computer," Information Document, IST Call for proposals, European Commission, Future and Emerging Technologies, 2000.
- [4] Dey, A.K., et al. "A conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications" anchor article of a special issue on Context-Aware Computing, Human-Computer Interaction (HCI) Journal, Vol. 16, 2001
- [5] B. N. Schilit, N. Adams, and R. Want, "Context-aware computing applications," Proceedings of the Workshop on Mobile Computing System and Applications, pp. 85-90, 1994.
- [6] A. K. Dey, "Supporting the Construction of Context-Aware Applications," Dagstuhl seminar on Ubiquitous Computing, 2001.
- [7] Chen, Harry, Tim Finin, and Anupam Joshi. "An Intelligent Broker for Context-Aware Systems." Adjunct Proceedings of Ubicomp 2003, Seattle, Washington, USA, October 12-15, 2003.
- [8] OWL : <http://www.w3.org/2004/OWL/>
- [9] T. Gu, H.K. Pung and D.Q. Zhang "A Middleware for Building Context-Aware Mobile Services," In Proceedings of IEEE Vehicular Technology Conference(VTC), 2004.
- [10] Michael J. Covington, Prahlad Fogla, Zhiyuan Zhan, and Mustaque Ahamad, "Context-aware Security Architecture for Emerging Applications," Security Applications Conference(ACSAC), 2002.
- [11] M. Roman, C. K. Hess, R. Cerqueira, A. Ranganathan, R. H. Campbell, and K. Nahrstedt. "Gaia: A Middleware Infrastructure to Enable Active Spaces", In IEEE Pervasive Computing, pp. 74-83, Oct-Dec 2002.
- [12] K. Cheverst, N. Davies, K. Mitchell, A. Friday: Experiences of Developing and Deploying a Context-Aware Tourist Guide: The GUIDE Project. Proc. of MOBICOM 2000, Boston, ACM Press (2000)
- [13] A. K. Dey and Gregory D. Abowd. "The Context Toolkit: Aiding the Development of Context-Aware Applications", In the Workshop on Software Engineering for Wearable and Pervasive Computing, Limerick, Ireland, June 6, 2000.
- [14] S. Shafer, B. Brumitt, and B. Meyers, The EasyLiving Intelligent Environment System". CHI

Workshop on Research Directions in Situated Computing, April 2000

- [15] <http://research.microsoft.com/easyliving>
- [16] K. Cory, R. Orr, G. Abowd, C. Atkeson, I. Essa, B. MacIntyre, E. Mynatt, T. Starner, and W. Newstetter, "The aware home: A living laboratory for ubiquitous computing research" . In the Proceedings of the Second International Workshop on Cooperative Buildings, 1999.
- [17] <http://www-static.cc.gatech.edu/fce/ahri>
- [18] Yatani, Koji, Masanori Sugimoto and Fusako Kusunoki. "Musex: A System for Supporting Children's Collaborative Learning with PDA's." Second IEEE International Workshop on Wireless and Mobile Technologies in Education. 2004

저자소개



정헌만

1996년 2월 서울산업대학교 전자계산공학과
 2001년 2월 인하대학교 전자계산공학과 공학석사
 2004년 2월 인하대학교 컴퓨터정보공학과 박사과정 수료

관심분야 상황인식, 시맨틱웹, 웹서비스, 유비쿼터스 센서네트워크