

# 계층적 온톨로지 모델링을 이용한 컨텍스트 프레임워크의 설계

장인우\*, 우종우\*  
\*국민대학교 컴퓨터공학부  
e-mail : zx1018@kookmin.ac.kr

## A Design of Context Framework using Hierarchical Ontology Modeling

In-woo Jang\*, Chong-Woo Woo\*  
\*School of Computer Science, Kookmin University

### 요 약

유비쿼터스 컴퓨팅에 대한 연구의 일환으로 사용자의 상황이나 의도를 파악하여 최적의 서비스를 제공하는 방법에 대한 연구가 활성화되고 있다. 최근 이러한 연구의 일환으로 온톨로지를 이용한 상황인식 시스템이 제기되고 있는데, 이는 컨텍스트에 대한 다양한 표현력 및 뛰어난 확장성을 가지고 있기 때문이다. 본 논문에서는 기존 연구에서 부족했던 컨텍스트의 정형화된 표현 및 다양한 도메인에 대한 확장성을 보완한 온톨로지 기반의 Context-awareness Framework 을 설계하였다. 본 논문에서 제안하는 Framework 은 하위 컨텍스트를 사전처리 및 Fuzzy 를 이용하여 상위 컨텍스트로 변환하며 이를 하나의 통합된 컨텍스트인 상황(Situation)으로 표현함으로써 추론 및 서비스를 유연하게 제공할 수 있다.

### 1. 서론

유비쿼터스 컴퓨팅에서 Context-awareness 란 사람, 컴퓨터 기기, 환경 등이 상호 작용하여 시스템이 자율적으로 사람을 포함한 사물이 필요로 하는 행동을 찾아내는 것을 의미한다. 기계 스스로가 자율적 행위를 하기 위해서는 컴퓨터로 인지 가능한 모든 객체에 대한 상황 정보를 알아야 한다. 여기서 객체란 상황 정보를 가지고 있는 실생활 공간의 요소를 나타내며 크게 User, Physical, Location, Robot 등으로 나눌 수 있다. 이렇게 정의한 객체들의 상황에 대한 특징을 기술한 모든 정보를 컨텍스트라고 정의하며[3] 이를 통합한 Situation 을 인지하는 것이 Context-awareness 시스템의 핵심이라 할 수 있다. 이와 관련하여 기존에 연구되었던 시스템으로 ContextToolkit[6], CoBrA[4], SOCAM[5]등이 존재한다. 하지만 이러한 시스템들은 컨텍스트의 정형화된 표현이 가능하지 않고 다양한 도메인에 관한 확장성이 유연하지 못하다는 문제가 있다. 컨텍스트의 정형화된 표현이 필요한 이유는 개발된 시스템이 이종 시스템과의 호환 및 이식성 때문이며 확장성이 필요한 이유는 시스템에 새로운 장치 및 서비스를 추가하기 쉽도록 유연한 구조가 요구되기 때문이다.

본 논문에서는 이러한 점을 해결하기 위해 온톨로지 및 Fuzzy 를 이용하여 각 컨텍스트의 상태를 정의한 Semantic 한 데이터를 사용함으로써 정형화된 컨텍스트를 생성한다. 또한 상위 온톨로지 및 하위 온톨로지를 구축함으로써 도메인 정보를 추가 및 변환할 때 유연성을 가지도록 하였다. 또한 자체적인 추론 및 학습을 위하여 통합된 컨텍스트 형태의 Situation Representation 을 제공한다.

본 시스템의 궁극적인 연구목표는 유비쿼터스 환경에서 컨텍스트 정보를 기반으로 기계 스스로의 상황 인지에 따라 지능형 기기 및 지능형 로봇이 취해야 할 행동을 적합하게 수행하는 것이며 이를 위한 Context awareness Framework 을 구축하는 것이다. 이를 위하여 상위 온톨로지 및 도메인에 따른 하위 온톨로지를 구축하였으며 실제 컴퓨팅 디바이스를 사용하여 실험할 계획이다.

본 논문의 구성은 2 장에서 관련 연구, 3 장에서는 제안하는 온톨로지 모델과 Framework 에 대한 설계, 4 장에서는 구현 및 실험 그리고 5 장의 결론으로 구성하였다.

### 2. 관련 연구

#### 2.1 Context Awareness

기존 상황인식 시스템으로 CoBrA(Context Broker Architecture)[2], SOCAM(Service Oriented Context-Aware

“본 논문은 서울시 산학연 협력사업의 지원을 받아 연구 수행된 논문입니다.”

Middleware)[3], Context-Toolkit[4] 등을 들 수 있다. 이 시스템들은 다시 온톨로지 모델링을 사용한 시스템 (CoBrA, SOCAM)과 사용하지 않은 시스템 (ContextToolkit)으로 나눌 수 있다.

온톨로지를 사용한 시스템들의 장점은 컨텍스트 정보의 공유, 컨텍스트 정보의 재사용, 컨텍스트 정보의 추론 및 확장성과 상호 운용성을 들 수 있는데 이 때문에 최근 시스템들은 온톨로지를 사용하는 것이 필수적이 되고 있다. 기존에 연구되었던 시스템의 특성 및 한계는 다음과 같다.

1) Context-Toolkit

상황인식 미들웨어의 출발점이라 할 수 있으며 접속된 기기의 정보를 획득할 수 있다. 하지만 온톨로지를 사용하지 않았기 때문에 Context의 정형화된 표현이 불가능하며 이 때문에 이 기종 시스템간의 호환이 불가능하다.

2) CoBrA

온톨로지 기반의 상황인식 시스템으로써 정형화된 Context 표현이 가능하다. 하지만 Intelligent Meeting Room이라는 특정 도메인을 기준으로 온톨로지를 적용하였기 때문에 다양한 도메인에 적용할 수 있도록 확장성을 제공하지는 않는다.

3) SOCAM

온톨로지 기반의 Context 표현뿐 아니라 확장성을 고려한 CONON[5] 모델을 따르고 있기 때문에 효율적인 미들웨어 시스템이다. 하지만 온톨로지 추론을 위해서 OWL에 의존하기에 자체적인 학습 모듈 및 추론 엔진을 가질 수 없다는 단점을 가지고 있다.

3. 시스템 설계

3.1 Ontology Modeling

상황 인식을 위한 컨텍스트 모델링을 위한 방법은 크게 세가지가 존재한다. 이는 Application-Oriented Approach, Model-Oriented Approach, Ontology-Oriented Approach[1]이며 본 논문에서는 세 번째 방법을 사용한다. 온톨로지를 정의하기 위해 W3C에서 제안하는 여러 가지 언어 중에서 추론 및 표현력이 가장 좋은 OWL을 사용하는데 그 중에서도 OWL-DL을 사용한다.

<표 1> OWL-DL 표현

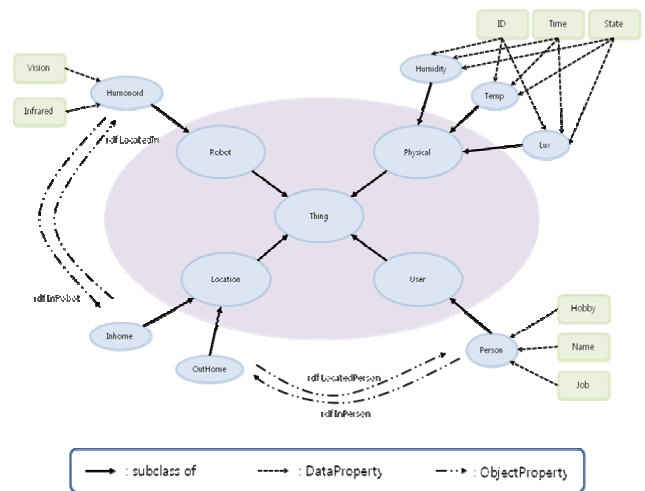
```

<owl:Class rdf:about="#Temp">
  <rdfs:subClassOf rdf:resource="#PhysicalContext"/>
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#RawData"/>
      <owl:minCardinality
        rdf:datatype="&xsd:nonNegativeInteger">1</owl:minCardinality>
    </owl:Restriction>
  </owl:equivalentClass>
  
```

본 논문이 제시하는 시스템에서 온톨로지 모델링은 아래 (그림 1)과 같다. 최상위 클래스인 Thing의

SubClass로 상위 온톨로지인 네 개의 주요 클래스가 존재한다. 주요 클래스인 Location, User, Physical, Robot은 도메인 독립적이며 각 Class의 특성에 따라 도메인에 맞는 하위 온톨로지를 지닌다. 본 논문에서는 위와 같이 구축된 상위 온톨로지를 기반으로 로봇을 이용한 스마트홈을 도메인으로 하여 하위 온톨로지를 작성하였다. 하위 온톨로지는 상위 온톨로지의 subclass로 각각의 속성값인 DataProperty 및 관계 속성인 ObjectProperty를 가지고 있다. 예를 들어 Physical 컨텍스트의 경우에는 subclass로 temp, humidity, lux가 있으며 각 subclass는 상태 정보를 위한 DataProperty로 Time, ID, State를 가진다. 또한 위치 정보인 Location 컨텍스트의 하위 온톨로지는 Robot 컨텍스트와 User 컨텍스트의 하위 온톨로지와 ObjectProperty를 이용하여 LocatedIn, Inrobot(Inuser)관계로 묶여 있다. 이렇게 상위 온톨로지 및 하위 온톨로지와의 상관 관계 및 연관 관계를 통하여 정형화된 표현 및 추론이 가능하다.

이와 같은 모델링은 CONON[7] 모델과 Schilit[3]이 정의한 컨텍스트를 결합한 것이며, 이 모델링의 목적은 정형화된 컨텍스트를 정확히 표현할 수 있을 뿐 아니라 다양한 도메인에도 확장이 가능하도록 하는 것이다. 이와 같이 구축되어 있는 온톨로지 모델링은 모델을 정의한 Schema와 Instance로 나눌 수 있으며 Schema에 따른 Instance를 생성함으로써 컨텍스트를 표현한다.

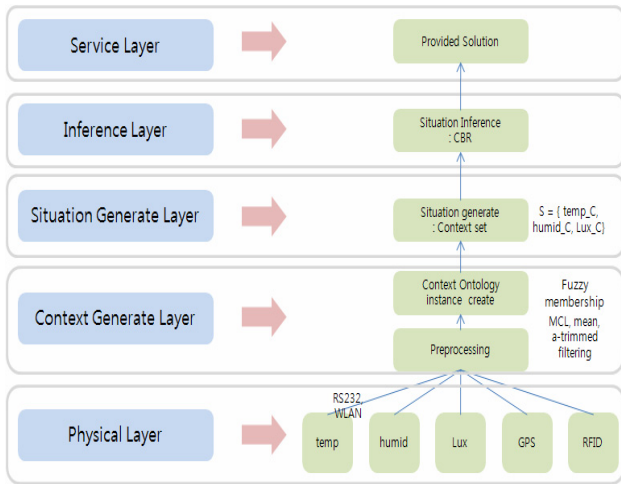


(그림 1) Ontology Model

3.2 전체 시스템 Framework

본 논문이 제시하는 Context-awareness Framework의 구조는 아래 (그림 2)와 같다. 기존 시스템의 경우 온톨로지를 활용하여 상황을 추론하는 반면 본 시스템에서는 추론엔진을 따로 구성하여 상황을 추론하는 모습을 제한한다. 이는 온톨로지를 이용한 추론이 한계가 있으며 또한 학습이 불가능하다는 단점을 극복하기 위함이다. 본 시스템의 구성은 크게 다섯 개의 Layer로 구성되어 있다. Physical Layer에서는 서버로

의 데이터 전송을 담당하고 Context Generate Layer 에서는 센서 디바이스에 따른 데이터의 가공 및 잘못된 데이터의 처리, 데이터의 계층화를 통해 Context Instance 의 생성을 담당한다. 또한 Situation Generate Layer 는 각 디바이스 및 로봇, 유저의 컨텍스트를 수집하여 통합된 컨텍스트인 벡터화된 Situation Representation 을 표현하며, Inference Layer 에서는 추론을, Service Layer 에서는 실제 지능형 기기 및 로봇의 행동에 대한 명령을 내리게 된다.



(그림 2) Context-awareness Framework

1) Physical Layer

Physical Layer 에서는 편재되어 있는 센서 디바이스로부터 데이터를 수집한다. 각 디바이스는 RS232 및 RF 신호를 통해 서버에게 데이터를 전송한다. 이 때 전송되는 데이터는 Numeric 한 raw Data 이다.

2) Context Generate Layer

Context Generate Layer 에서는 디바이스로부터 전송된 데이터를 Semantic 한 컨텍스트로 변환하는 과정을 거친다. 먼저 전송된 데이터는 디바이스 타입에 따라 Preprocessing 을 하는데, 이는 디바이스 데이터의 신뢰도를 높이기 위한 잘못된 정보의 보정이므로 필수적이라 할 수 있다. 이 단계가 끝나면 Fuzzy membership 함수를 통해 가공된 데이터를 컨텍스트로 변환하는데 Fuzzy 함수를 통해서 각 디바이스의 상태를 나타내는 이유는 Semantic 한 언어를 사용함으로써 현재 상황을 모니터링 하거나 추론하기에 적합한 컨텍스트[2][8]로 나타내기 위함이다.

3) Situation Generate Layer

Situation Generate Layer 에서는 디바이스의 컨텍스트를 수집하여 하나의 Situation 으로 변환한다. 실제 Context-awareness 는 이 Situation 을 인지하는 것으로 효율적인 추론 및 학습을 위해서는 Vector 화된 데이터가 필요하다. 이를 위해 본 논문에서는 Situation Representation 을 하나의 Vector 형태로 표현하며 이는 온톨로지로 구축된 Context Instance 를 추출하여 Vector 의 요소 요소로 변환한다. 즉 최종적으로 만들어지는 Situation Representation 은

$S = \{Temp\_c, Humidity\_c, Lux\_c, Distance\_c, \dots\}$  형태가 된다.

4) Inference Layer

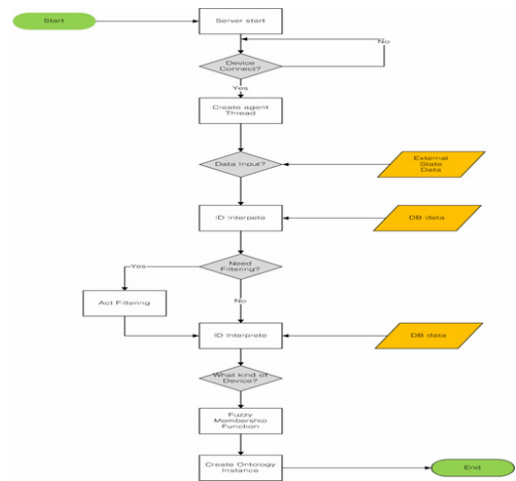
이 Layer 에서는 Situation Representation 을 기준으로 추론 및 학습을 하게 된다. Inference Engine 은 온톨로지 독립적이며 지금 설계 및 제작 중이기에 여기에서는 다루지 않는다.

5) Service Layer

이 Layer 에서는 Robot 의 지능적 행위뿐 아니라 AirConditioner 등의 지능형 기기를 위한 직접적인 명령을 실시한다.

4. 구현 및 실험

본 논문에서의 온톨로지 모델링은 Protege ' 4.0 을 사용하고 Instance 의 생성 및 컨텍스트의 추출을 위해서는 Jena2 를 사용한다. Protege' 4.0 은 온톨로지 모델링 툴로써 Class 선언 및 관계정의를 통하여 자동으로 OWL 파일을 생성할 수 있다. 이를 통해 앞에서 정의한 온톨로지 모델링을 구축하였으며 Jena2 의 경우 Instance 의 생성 및 관계 추론, DB 저장 등의 기능이 있으며 본 논문에서는 Context Instance 의 생성 및 Situation Representation 을 위한 도구로써 사용하였다.



(그림 3) 프로세스 순서도

Situation Generate Layer 까지의 흐름도는 위의 (그림 3) 과 같다. 처음 중앙 컴퓨터에서 서버 에이전트가 실행되면 디바이스의 접속을 기다린다. 사람, 로봇 혹은 공간에 장착되어 있는 디바이스가 작동되면 서버 에이전트에게 접속을 시도하게 된다. 서버 에이전트는 각 디바이스마다 하나의 스레드를 생성하여 관리하고 디바이스는 정해진 프로토콜에 따라 필요한 데이터를 전송한다. 이 때 서버 에이전트에서 생성된 스레드는 Agent DB 에 저장되어 있는 정보대로 Preprocessing 및 Fuzzy 화를 통해 데이터를 가공하며 정형화된 데이터는 Context Instance 로써 온톨로지에 등록된다.

Preprocessing 은 디바이스의 종류마다 처리하는 내용이 다르며 데이터의 보정 및 잘못된 데이터의 예외 처리를 담당한다. 특히 로봇 위치 데이터의 경우 신

뢰도가 높지 않기 때문에 MCL(Monte Carlo Filtering)를 수행하며 일반 데이터의 경우에도 디바이스의 고장 및 데이터의 보정을 위해서 Mean Filtering 및 예외처리를 실시한다. Fuzzy 화는 각 디바이스의 데이터 범위 및 특성에 따라 FuzzyMembership 함수를 작성하였으며 이를 통한 데이터의 정형화가 이루어진다. 아래 <표 2>는 FuzzyMembership 함수를 거친 데이터의 Semantic 한 형태를 보여주며 현재 컨텍스트 정보는 다섯단계로 작성되어 있다. 이는 정확한 컨텍스트를 표현할 수 없기에 추후 좀 더 세분화할 예정이다.

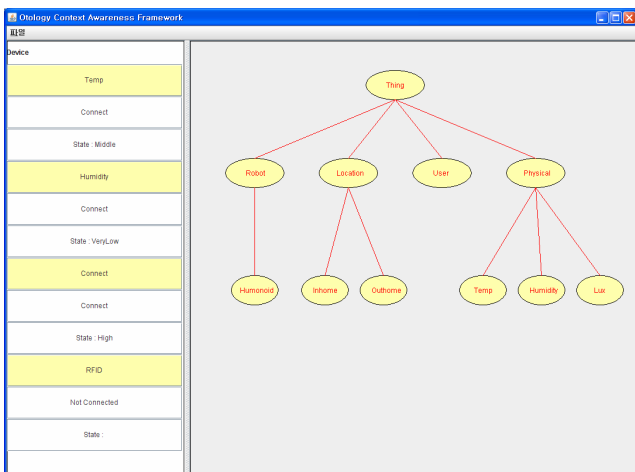
계속해서 스레드는 일정 시간마다 데이터의 분석을 통해 Context Instance 를 생성하며 이 때마다 컨텍스트의 통합 정보인 Situation 을 생성하게 된다.

<표 2> Semantic Context

	온도	습도	조도	거리
very low	매우 추움	매우 낮음	매우 어두움	매우 가까움
Low	추움	낮음	어두움	가까움
Middle	따뜻함	보통	보통	보통
High	더움	높음	밝음	먼
very high	매우 더움	매우 높음	매우 밝음	매우 먼

본 논문에서는 Context Framework 에 관한 실험으로서 실제 디바이스를 사용한다. 사용 디바이스는 온도, 조도, 습도를 측정할 수 있는 Zigbee, 입 출입을 체크하기 위한 RFID, 내부 위치를 측정하기 위한 GPS 가 된다. 또한 서비스를 수행할 Robot 으로 ROBONOVA 를 사용한다.

아래 (그림 4)는 본 논문에서 제안하는 시스템의 Prototype 모습이다. 온톨로지 모델을 로드하여 클래스의 계층관계 및 상관도를 오른쪽 화면에 보여준다. 또한 왼쪽 화면에서는 각 디바이스의 접속 현황 및 상태변화를 나타내며 이를 통해 각 디바이스의 정형화된 컨텍스트 정보를 확인할 수 있다.



(그림 4) System Prototype

### 5. 결론

본 논문에서는 보완된 Context-awareness Framework 을 제시했다. 이는 기존 컨텍스트의 정형화된 표현을 통한 상호 운영성[9]에 대한 이점을 가지며 상위 온톨로지의 구성으로 인한 다양한 도메인으로의 확장성이 뛰어나다는 장점도 가지고 있다. 또한 백터화된 Situation Representation 으로 인한 상황에 대한 추론 및 학습에 뛰어난 구조라 할 수 있다.

현재 Situation Generate Layer 까지 구축되어 있으며 향후 연구로는 Inference Layer 부터 구축할 예정이며, 추론 및 학습에 뛰어난 엔진 및 Service Layer 을 구축함으로써 모든 응용시스템에 활용이 가능한 Context-awareness System 으로 확대할 예정이다.

### 참고문헌

- [1] “유비쿼터스 환경의 상황인식 프레임워크를 위한 컨텍스트 온톨로지 정의” 2007.4 퍼지 및 지능 시스템학회 논문지 제 17 권 제 2 호
- [2] “Fuzzy OWL 을 이용한 사용자 Context 의 표현 및 추론” 2008.8 지능정보연구 제 14 권 제 1 호
- [3] Bill Schilit, Norman Adams, and Roy Want, “Context-aware Computing Applications,” In proc. Of IEEE zWorkshop on Mobile Computing Systems and Applications, SantaCruz, California, Dec. 1994, pp.85-90
- [4] Harry Chen, Tim Finin, and Anupam Joshi, “An Ontology for Context-Aware Pervasive Computing Environments,” Special Issue on Ontologies for Distributed Systems, Knowledge Engineering Review, Vol.18 pp.197-207, September 2003
- [5] T.Gu et al, “An Ontology Based Context Model in Intelligent Environments,” Proc. Of Communication Networks and Distributed Systems Modeling and Simulation Conference, 2004
- [6] A.K.Dey, G.D.Abowd, “The Context Toolkit; Aiding the Development of Context-Aware Application,” Human Factors in Computing Systems, pp.434-411, 1999
- [7] X.Wang, D.Zhang, T.Gu, H.Fung, “Ontology Based Context Modeling and Reasoning using OWL,” Proc. Of the Second IEEE Annual Conference on Pervasive Computing and Communication Workshop, 2004
- [8] Giorgos Stoilos, G. Stamou, N.Simou, S.Kollias, “Uncertainty and the Semantic Web”, IEEE Intelligent Systems, Vol.21, No.5(2006), 84-87
- [9] Thomas S, C.LP, and K.Frank. “CoOL : A Context Ontology Language to Enable Contextual Interoperability”, International Conference on Distributed Applications and Interoperable Systems, 2003