Chapter 14
상황인식 처리 기술

오양가 (139144103)

목차

- 개요
- 상황인식 서비스 요소 기술
- 상황정보의 정의 및 표현
- 상황정보 전달을 위한 미들웨어

1. 개요

- 유비쿼터스 시대의 응용 및 서비스
  - 센서를 통해 수집된 상황 정보를 인식, 해석, 추론과 같은 처리 과정을 거친 후, 사용자에게 상황에 적절한 서비스를 제공

- 상황 인식 및 처리 기술
  - 크게 환경 또는 사용자와 관련된 빛, 온도, 소리, 움직임 등의 정보를 검출하는 **센서 기술**
  - 센싱된 정보를 공유할 수 있게 하는 **센서 네트워크 기술**
  - 공유된 정보를 상위의 상황 정보로 **추론하고 가공하는 지능**
  - 그 상황 정보에 따라 사용자에게 적절한 서비스를 제공하는 **사용자 인터페이스 기술**로 집약

- 센서부분을 제외한 상황인식 처리 기술
  - **상황인식 서비스 요소 기술**
  - 최근 활발한 연구가 진행되고 있는 **온톨로지 기반의 상황 정보 정의 및 표현 방법**
  - 상황 정보 전달을 위한 **미들웨어**에 대해 언급

상황인식 서비스 인프라에 대한 요구 사항 분류

- 기능적 요구 사항(Functional Requirements)
  - 상황 정보 수집, 상황 정보 저장 및 관리, 상황 정보 신청 및 배달, 상황 정보 분해 및 융합 등이 포함
- 비기능적 요구 사항(Non Functional Requirements)
  - 확장성, 모듈성, 보안성, 이동성, 인터페이스 적절성, 결함 포용성, 서비스 품질, 발전 가능성, 플랫폼 간 호환성(Cross platform) 등이 있음

상황인식 서비스 인프라관련 연구에서 공통적으로 요구되는 요소기술

① 상황 정보 수집 기술

- 온도, 습도와 같은 **환경적 상황정보**와 체온, 혈압 등 **사용자와 관련된 정보들**은 단말에 부착된 센서를 통해 직접 수집
- 수집된 상황 정보들은 **상황 정보 모델**에 따라 내부에 저장 과정을 거쳐 추론을 위한 기초 자료가 됨
- 다수의 상황정보는 시간이 지남에 따라 변화하므로, 주기적인 **폴링(polling)**이나 **전파(broadcasting)**기법에 의해 파악.
- **조명의 밝기**는 감광성 반도체 소재, **가속도와 진동**은 가속도계, **인접 객체 감지**는 수동형 적외선 센서, **소리**는 마이크로폰, **기후 정보는** 온도계 및 습도계를 이용하여 센싱

② 상황정보 모델링 기술(Context modeling)

- 센서와 장치(actuator)에 대한 추상화를 제공해야 하며, 상황정보들이 서로 교환될 수 있게 표준화된 모델링 기술이 필요
- 상황 정보의 표현 방법은 스트랭(Strang)과 린호프-포피엔(Linnhoff-Popien)은 6가지로 구분

1) 키값 (Key-value) 기반 모델

  - 가장 간단한 형태의 모델링 방식으로 정보를 나타냄
  - 많은 상황 인식 시스템의 서비스 발견(Discovery) 프로토콜들은 키값(Key-value)으로 나타내어진 값에 대하여 단순한 문자 비교의 방식으로 필요한 서비스를 발견

2) 마크 업 기반 모델

  - 태그, 속성 그리고 내용을 계층구조로 나타내어 재귀적 형태를 가지고 있음

3) 그래픽 기반 모델

- 통합 모델링 언어(Unified Modeling Language; UML)와 같은 강력한 그래픽 기반의 기능을 이용해서 표현

4) 객체지향 기반 모델

- 유비쿼터스 컴퓨팅 환경의 동적 상황을 객체지향 기술을 이용하여 추상화
- 장점: 새로운 타입의 상황 정보의 추가 및 인스턴스 업데이트 등이 분산된 시스템에서 용이 함

5) 로직 기반 모델

- 사실(Fact), 표현(Expression), 규칙(Rule)의 정형화된 표현을 사용
- 장점: 상황 정보는 사실이라는 형식으로 나타내고, 규칙을 통해서 사실이나 표현을 추론

6) 온톨로지 기반 모델
  - 상호 관계성 및 부분적인 상황 정보를 쉽게 표현
  - 최근 다양한 상황 인식 프레임워크는 주로 온톨로지 기반의 모델을 채택
  - 자원 기술 프레임워크(Resource Description Framework; RDF), 온톨로지 웹 언어(Ontology Web Language; OWL) 등의 온톨로지 표준 언어로 선언적인 표현
  - 온톨로지 웹 언어에 기반한 시멘틱 웹 연구와 연관하여 활발한 연구

- **온톨로지(Ontology)**라는 용어
  - 철학에서 현실에 대한 개념과 관계를 연구하는 분야에 사용
  - 개념과 관계를 정리하여 주석 정보를 제공하고 다른 사람 또는 컴퓨팅 에이전트 등과 공유하는데 많이 사용
  - 그 외에도 타 도메인의 지식을 재사용, 도메인 지식의 상속, 부분 지식의 결합 및 컴퓨터를 이용하여 방대한 지식베이스에 대해서 정형화된 분석을 가능하게 함

2. 상황인식 서비스 요소 기술

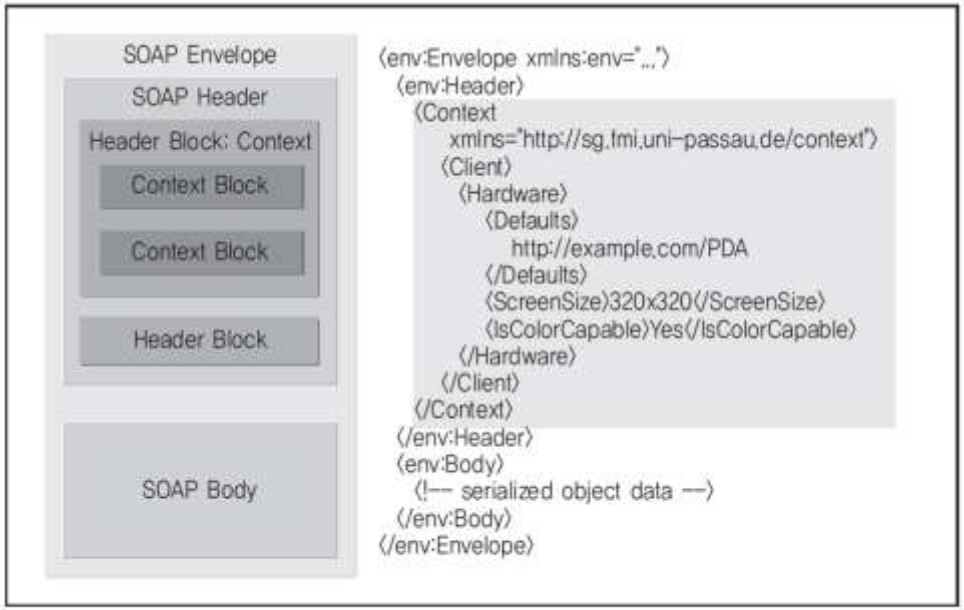
	분산 조합	부분 확인	표현되는 정보의 질	불명확한 정보 대처	표현의 정규도	응용 가능성
키값 모델	✕	✕	△	△	△	○
마크업 기반 모델	○	◎	✕	✕	○	◎
그래픽 기반 모델	△	✕	○	✕	○	○
객체지향 기반 모델	◎	○	○	○	○	○
로직 기반 모델	◎	✕	✕	✕	◎	△
온톨로지 기반 모델	◎	◎	○	○	◎	○

③ 상황정보 융합 및 추론 기술(Context fusion & reasoning)

- 센싱 데이터를 융합하여 상위 상황정보를 유도하기 위해 확률적인 메커니즘을 제공하고, 계층적 상황정보를 기반으로 지능적인 추론 방법도 제공.
- 온톨로지를 이용한 상황지식의 공유와 재사용 지원 및 높은 추론 기능을 제공함에 따라 온톨로지를 이용한 추론 시스템도 활발히 연구.

④ 상황정보 교환 기술(Context exchange)

- 센서, 장치 및 객체와의 상호작용을 지원하기 위해 이벤트 기반 또는 폴링 기반의 통신 메커니즘을 제공



[그림 15-1] SOAP 메시지에 포함된 상황 정보

2. 상황인식 서비스 요소 기술

⑤ 상황정보 툴킷 기술(Context toolkit)

- 일종의 비주얼 프로그래밍 툴로써, 센서로부터의 정보를 상황정보화하는 상황정보 위젯, 여러 개의 상황정보 위젯으로부터 정보를 취합하는 서버, 그리고 상황정보 위젯이나 서버로부터의 상황을 다시 한번 취합하여 고수준의 상황으로 처리하는 해석기로 구성됨.

  (위젯 :하드웨어 센서를 위한 공통의 인터페이스를 제공하는 소프트웨어 컴포넌트)

⑥ 상황인식 서비스 기술 언어(Context-aware service description language)

- WSDL(Web Services Description Language)은 웹 서비스에 의해 제공되는 메소드, 메시지, 바인딩 등을 정의하고 서비스를 설명하기 위해 사용되는 또 다른 형태의 XML 언어

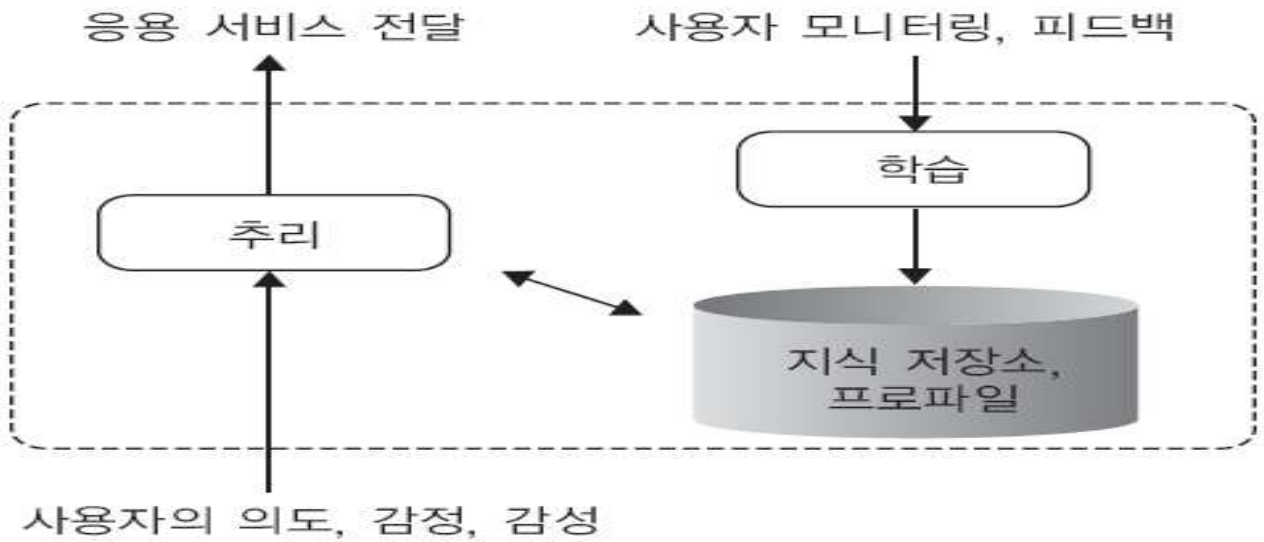
⑦ 상황인식 서비스 구조 기술(Context-aware service infrastructure)

- 서비스 구조는 미리 정의된 공통 데이터 형식과 네트워크 프로토콜로 구성.
- 서비스 구조의 가장 좋은 예는 인터넷.
- TCP/IP 표준 하부 프로토콜과 DNS, DHCP, Telnet, FTP, HTTP 등과 같은 상위 서비스 프로토콜로 서비스 구조가 구성되어 있어, 인터넷상에 새로운 장치와 새로운 서비스가 투명하게 추가.
- 상황인식 서비스를 위한 서비스 구조가 제공되면, 센서나 서비스, 장치 등이 다른 구성요소에 영향을 주지 않고, 동적으로 추가 가능.

2. 상황인식 서비스 요소 기술

⑧ 지능형 에이전트 기술

- 사용자의 단순한 의도뿐만 아니라, 감정, 감성을 고려하여 전체적인 상황을 자율적으로 판단하여 사용자에게 적합한 서비스를 제공하는 것이 목표.
- 사용자의 정보와 일치하는 서비스 정보를 매핑하여 지식 저장소에 저장하고, 이를 기반으로 사용자의 요구가 있을 때 그에 알맞은 서비스를 제공.



[그림 15-2] 지능형 에이전트 개념도 [참고자료: ETRI, 2004]

⑨ 상황정보 관리 기술

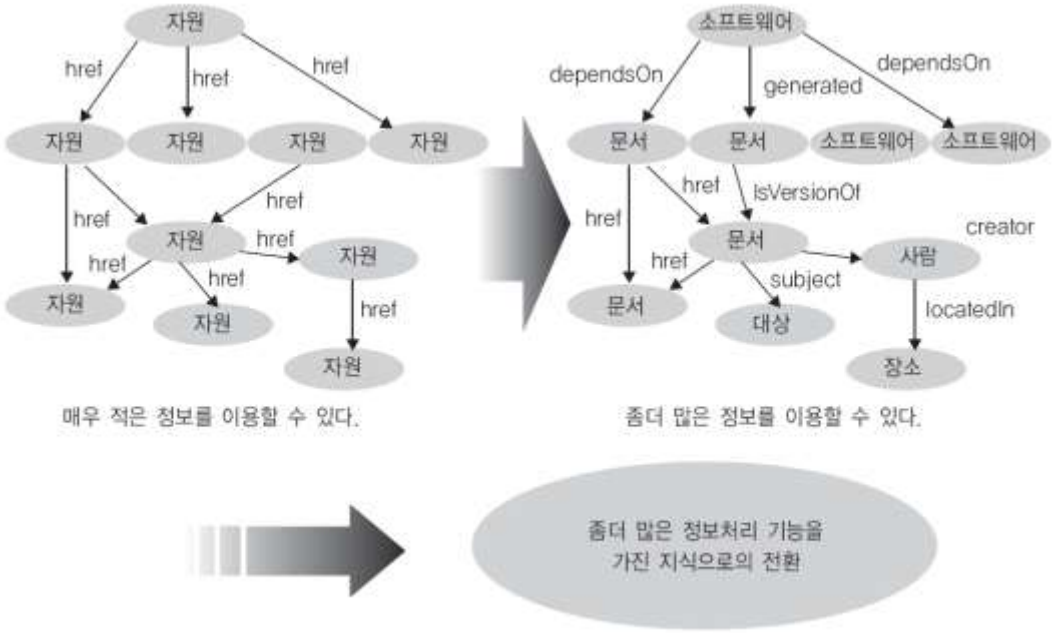
- 응용이 필요로 하는 상황정보를 지능적으로 조합하여 효율적으로 제공.
- 상황정보 툴킷에서 서버와 해석기(interpreter)를 이용.
- 서버는 상황정보 위젯들로부터 필요한 다양한 상황을 가져와서 취합하고, 해석기는 이들 상황을 다른 형태나 의미로 변환하여 상위 레벨의 응용에게 제공.
- 임의대로 추가되고 제거되는 물리적인 상황 센서들과 다양한 응용군에 대해서 플랫폼이 지속적으로 지원하는 기능

3. 상황 정보의 정의 및 표현

- 상황정보 정의
  - 사용자에게 좀더 정확한 서비스를 제공하기 위하여 필요
- 상황정보 표현
  - 수집된 상황 정보가 한 개인에게 국한 되는 것이 아니기 때문에 상황 정보에 대하여 전반적으로 표현함으로써 다수의 사용자에게 정보를 제공
- 시맨틱 웹 (Semantic Web)
  - 상황정보를 정의하고 표현하기 위해 사용되는 가장일반적인 방법.

시맨틱 웹(SEMANTIC WEB)

- '컴퓨터가 정보의 의미를 이해하고 의미를 조작할 수 있는 웹'
- 단순히 문서의 의미적 내용을 컴퓨터가 알지 못하는 수준을 월드 와이드 웹이라 한다면, 정보와 지식을 처리할 수 있는 웹의 환경을 시맨틱 웹이라 할 수 있음.



[그림 15-3] 월드와이드웹에서 시맨틱 웹 변화 [참고자료: 54]

시맨틱 웹을 지원하는 언어 (1/2)

시맨틱 웹 환경에서 지식을 표현하기 위한 언어

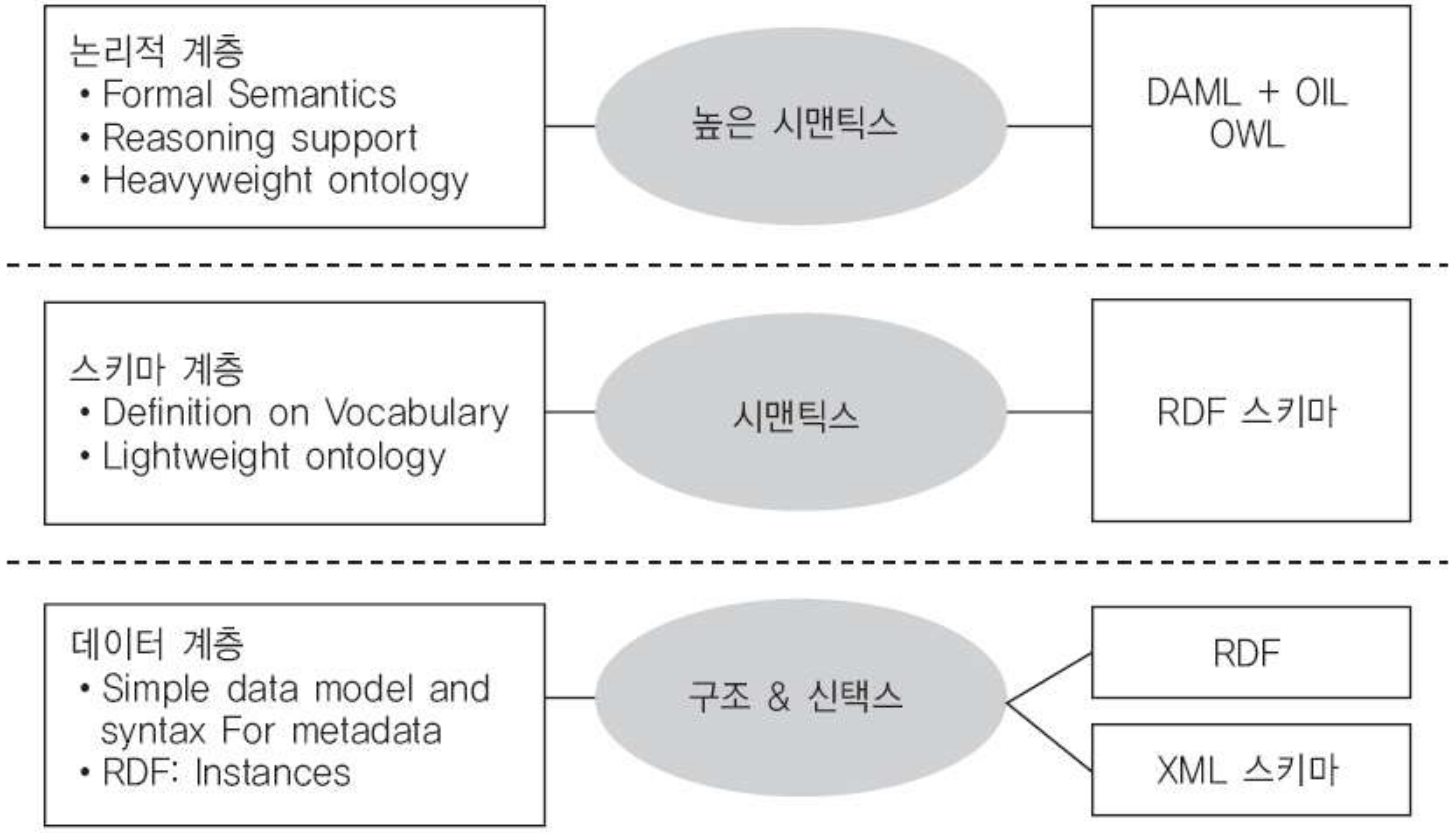
- ① 자원 기술 프레임워크(Resource Description Framework: RDF)
  - 주어-술어-목적어 형식의 문법을 사용
- ② 자원 기술 프레임워크 스키마(Resource Description Framework Schema: RDFS)
  - RDF를 기반으로 하여 대상 자원이 속하는 클래스와 속성에 대한 정의를 가능케 하는 스키마 언어
- ③ DARPA 에이전트 마크 업 언어(DARPA Agent Markup Language:DAML)
  - XML에 기반을 두고 있지만, 객체 및 객체들 간의 관계를 묘사하기 위한 XML보다 훨씬 더 많은 기능을 가지며, 의미론을 표현하고, 웹 사이트들 간에 보다 높은 차원의 상호 운용성을 구축하도록 설계

시맨틱 웹을 지원하는 언어 (2/2)

- ④ 온톨로지 추론 계층(Ontology Inference Layer:OIL),
  - 온톨로지는 제한된 단어에 명세를 제공하기 위하여 사용되는데, 이런 온톨로지를 위한 웹 기반 표현과 추론 계층을 위하여 제안됨.

- ⑤ 온톨로지 웹 언어(Ontology Web Language:OWL)
  - 문서에 포함된 정보를 응용에서 자동 처리하고자 할 때 활용하는 언어임.
  - OWL을 이용하면 임의의 어휘를 구성하는 용어(term)의 의미와 용어들 간의 관계를 명시적으로 표현.

지식 표현 언어의 계층화 구조



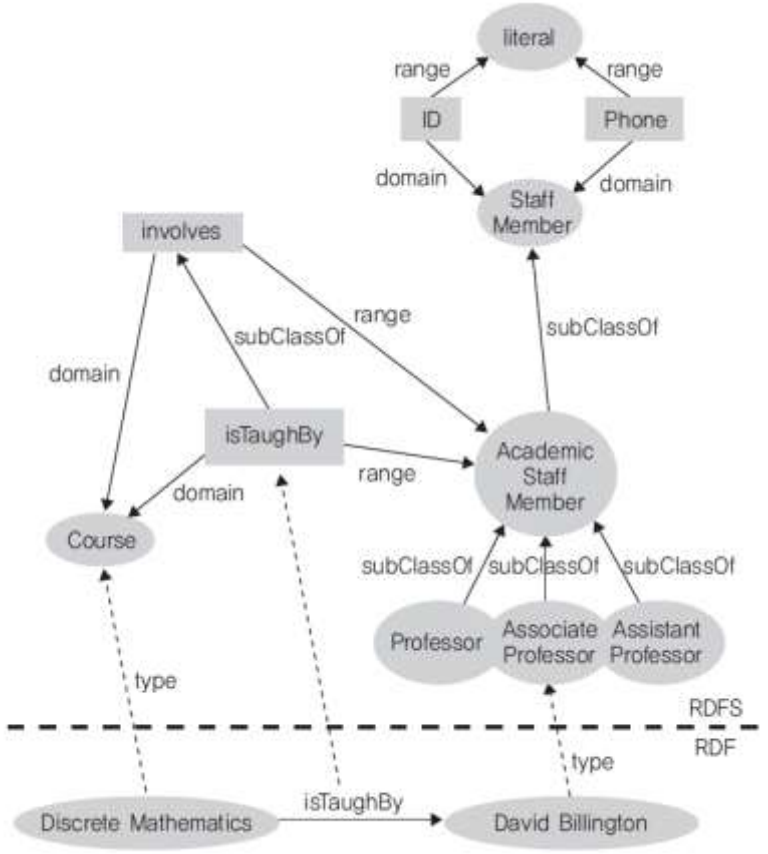
[그림 15-4] 지식 표현 언어의 계층화 구조 [참고자료: 54]

3. 상황 정보의 정의 및 표현

❖ 실제 적용 예
- 상황정보: "교과는 교수가 강의한다"라는 문장으로 표현.
- 클래스(타원형)
  - "교과(course)"
  - "교수(Academic Staff Member)"
  - "학교 구성원(Staff Member)"
- 속성(사각형):"강의(isTaughtBy)"
- 서브클래스: subClassOf 화살표로 표현

❖ 상황 정보
- "이산 수학(Discrete Mathematics)교과는 David Billington 교수가 강의한다"

➔ 최종적으로 클래스 및 속성의 인스턴스로 표현되고 있음
- 즉, "이산 수학(Discrete Mathematics)"은 "교과(Course)" 클래스의 인스턴스,
- "David Billington 교수"는 "부교수" 클래스의 인스턴스,
- 두 클래스 사이의 속성 "isTaughtBy"는 "강의한다"라는 인스턴스.



[그림 15-5] RDF와 RDFS 계층 [참고자료: 54]

4. 상황 정보 전달을 위한 미들웨어

- 유비쿼터스 컴퓨팅 환경의 주요 구성요소
  - 상황인식 시스템에서는 다양한 정보를 수집하는 센서와 사용자에게 적절한 서비스를 제공하는 애플리케이션 사이에 중간 매개체 역할을 하는 상황인식 미들웨어(Context Awareness Middleware)가 존재
  - 상황인식 미들웨어는 상황인식 애플리케이션과 센서들 사이에 위치하여 다양한 정보를 수집·가공한 후 이러한 정보들을 애플리케이션에 제공하고, 사용자가 필요로 하는 서비스를 제공

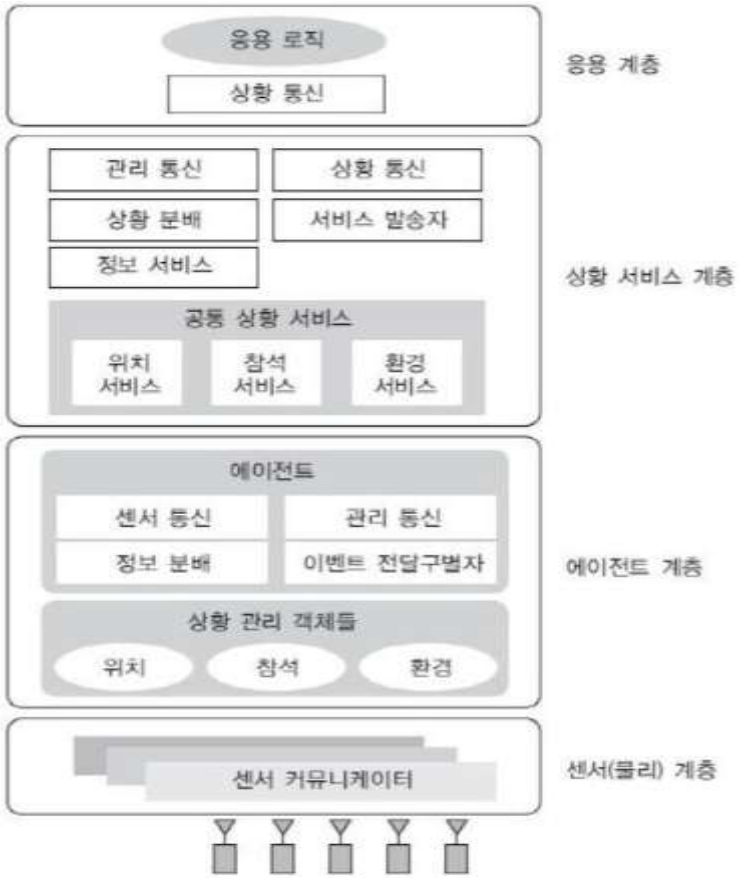
- 상황인식을 위한 미들웨어가 제공해야 하는 서비스
  - 여러 센서들로부터의 상황정보의 수집 및 다른 에이전트들에게 상황정보의 전달.
  - 저수준 정보 수집부터 고수준 추론.
  - 에이전트의 추론 및 러닝 메커니즘의 사용 지원.
  - 에이전트에게 상황에 따라 적절한 행동들을 수행하게 하는 능력.
  - 서로 다른 에이전트들 간의 구문과 의미적 내부 소통 지원.

4. 상황 정보 전달을 위한 미들웨어

❖ 상황정보 전달을 위한 미들웨어 아키텍처

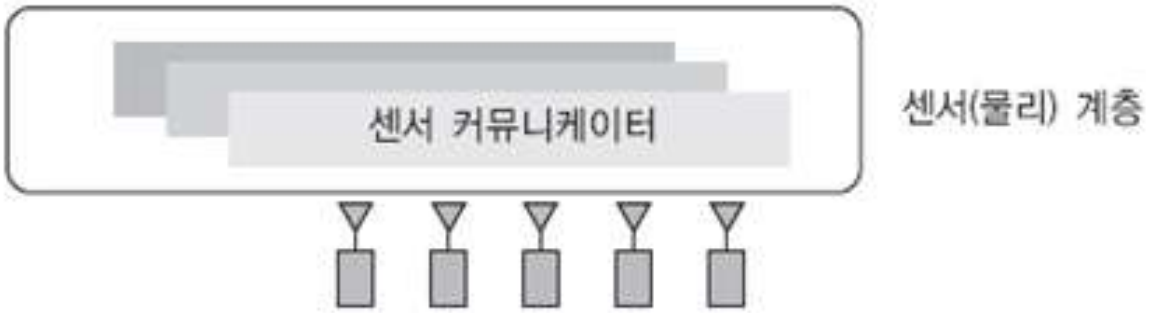
- 상황정보 전달을 위한 미들웨어는 환경 정보를 센싱하는 센서로부터 센싱 정보를 받아 이를 저장한 후 필요한 응용에게 전달.

- 즉, 상황 정보 전달 미들웨어를 통해 하나의 센서가 수집한 정보를 여러 응용이 공유.



4. 상황 정보 전달을 위한 미들웨어

- 센서(물리) 계층
  - 센서와의 통신은 각 센서 제작사들의 인터페이스 방식에 의존적이므로 이를 최소화하기 위해 센서 통신(Communication) 모듈을 둠
  - 센서 의존적인 인터페이스를 사용하여 주기적으로 센서의 정보를 획득
  - 물리적으로 하나의 센서 커뮤니케이터는 하나의 센서에 연결
  - 센서 의존적인 저수준의 인터페이스는 표준 응용 인터페이스로 래핑(Wrapping)
  - 센서 커뮤니케이터가 센서 인터페이스를 사용하여 센서로부터 정보를 가져오고 그 정보를 다시 표준 인터페이스를 사용하여 에이전트에게 전달
  - 표준 인터페이스는 두 가지 메시지 형태를 지닌다.
    - SQL과 같은 질의를 위한 **요구/응답(Request/Response) 메시지**
    - **비동기적 통보(Asynchronous Notification)** 메시지



4. 상황 정보 전달을 위한 미들웨어

- 에이전트 계층
  - ① 상황 관리 객체(CMO)와 ②에이전트(Agent)로 구성
  - 4개의 서브 모듈 - 센서 통신, 관리 통신, 정보 분배, 이벤트 전달 구별자로 구성

에이전트 계층

[그림 15-7] 상황 정보 전달을 위한 미들웨어 아키텍처-에이전트 계층 [참고자료: 54]

4. 상황 정보 전달을 위한 미들웨어

① 상황 관리 객체(CMO)

- 상황 툴킷의 상황 위젯과 유사
- 센서의 정보를 추상화 (cf. 에이전트는 프로세싱 로직과 중앙 저장을 담당)
- 속성값은 상황에 대한 추상화된 상태이며, 속성값이 변할 때마다 이벤트를 발생시킴

- 실세계의 많은 객체들을 센서 정보의 내용에 따라 세 가지 유형으로 모델링
  ① **참석(Presence)** 상황 관리 객체: 어떤 대상이 어느 위치에 있는지를 표현
  ② **위치(Location)** 상황 관리 객체: GPS와 같이 특정 객체의 위치를 나타냄
  ③ **환경(Environment)** 상황 관리 객체: 객체를 둘러싸고 있는 환경의 모든 상태 (예: 방안의 온도)를 표현

[표 15-1] 상황 관리 객체들 [참고자료: 54]

관리 객체 유형	속성	공지
참석 상황 관리 객체	ID, 객체 ID, 영역 ID, 시간	속성값 변경
위치 상황 관리 객체	ID, 좌표(x, y, z), 시간	속성값 변경
환경 상황 관리 객체	ID, 온도, 습도, 잡음, 풍속, 등	속성값 변경

4. 상황 정보 전달을 위한 미들웨어

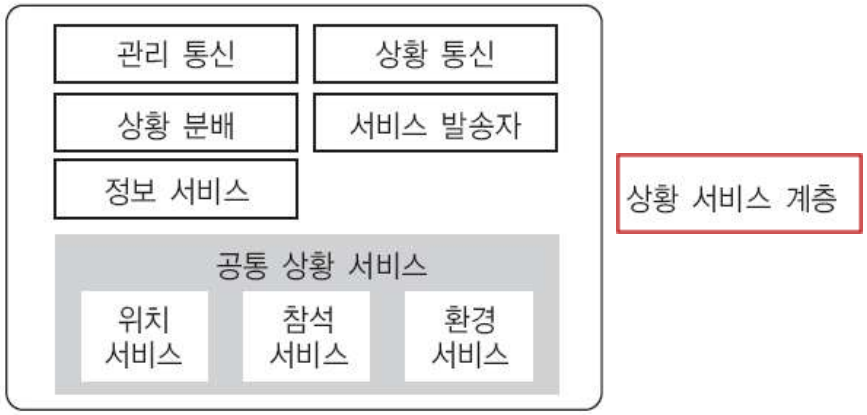
② 에이전트

- 센서 통신(Sensor Communication) 모듈
  - 정보를 얻기 위해 에이전트 내 센서 통신은 센서와 직접적으로 연결되어 있는 **센서 커뮤니케이터**로부터 **표준 인터페이스(질의나 비동기 이벤트)**를 통해서 **센서 정보를 받음**
- 관리 통신(Management Communication) 모듈
  - **에이전트 계층**과 상위계층인 **상황 서비스 계층**과의 통신을 담당
  - **SNMP, WBEM 등 표준 네트워크 관리 프로토콜**이 사용됨
- 정보 분배(Information Distribution) 모듈
  - 센서로부터 받은 정보를 **상황 관리 객체에 분배**하는 역할을 맡고 있음
- 이벤트 전달 구별자(Event Discriminator) 모듈
  - 상황 관리 객체로부터 발생 가능한 모든 이벤트 중에서 **실제 관리자에게 전달되어야 할 이벤트**를 구별

[그림 15-7] 상황 정보 전달을 위한 미들웨어 아키텍처-에이전트 계층 [참고자료: 54]

4. 상황 정보 전달을 위한 미들웨어

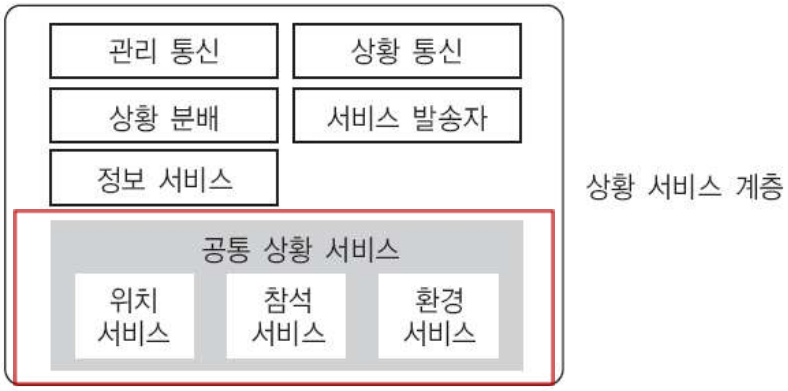
- 상황 서비스 계층
  - 응용 계층과 에이전트 계층에 대한 인터페이스가 존재
  - 에이전트 계층에 대해 관리자 역할을 수행하며, 에이전트로부터 센서 정보를 가져옴
  - 요구/응답 형태의 폴링 방식이나 신청(Subscribe)/발행(Publish) 형태의 이벤트 공지 방식으로 정보를 획득
  - 낮은 수준의 정보를 얻은 후엔 이를 저장하고 필요한 응용에게 서비스를 제공



[그림 15-8] 상황 정보 전달을 위한 미들웨어 아키텍처-상황 서비스 계층 [참고자료: 54]

4. 상황 정보 전달을 위한 미들웨어

- 각 모듈의 역할
  - 공통 상황 서비스(Common Context Service) 모듈
    - 상황 관리 객체와 일치되는 **참석, 위치, 환경** 서비스로 구성
      - » 각 응용에서 이 서비스를 이용하기 위해서는 **'신청** (Subscribe)' 메시지를 보내면, 해당 응용에게 서비스를 전달한다**(발행**(Publish)**)**
      - » **'요구'/ '응답'** 메시지를 사용할 수 있음



[그림 15-8] 상황 정보 전달을 위한 미들웨어 아키텍처-상황 서비스 계층 [참고자료: 54]

4. 상황 정보 전달을 위한 미들웨어

- 관리 통신(Management Communication) 모듈
  - 관리자로서, **에이전트 계층의 에이전트**와 통신을 담당하는 인터페이스 역할을 함
  - **에이전트**로부터 폴링/이벤트를 통해 정보를 받음
- 상황 통신(Context Communication) 모듈
  - **상황 서비스 계층**과 **응용 계층** 사이의 인터페이스 역할을 함
  - 상황 정보 요구 및 전달을 위해서 HTTP와 유사한 **상황 기반의 전송 프로토콜(ConteXt Transfer Protocol: CXTP)**이 사용될 수 있음
- 상황 분배(Context Distribution) 모듈
  - 에이전트로부터 넘겨온 정보는 관련된 **공통 상황 서비스**로 전달되어야 함
  - 에이전트 계층의 **상황 관리 객체 형태**는 바로 **서비스 형태**와 일치
  - **참석 상황 관리 객체**로부터 넘겨온 정보는 공통 상황 서비스의 **참석 서비스 모듈**로 전송
- 서비스 발송자(Service Dispatcher) 모듈
  - **응용**이 상황 서비스를 요청할 때 해당하는 **공통 상황 서비스에게 그 요청을 넘겨주는 역할**을 함
- 정보 서비스(Information Service) 모듈
  - 객체 ID에 대한 모든 정적인 정보 서비스를 제공하여 **저수준의 정보**를 **고수준의 정보로 변환**시킴

4. 상황 정보 전달을 위한 미들웨어

- 상황정보 전달을 위한 미들웨어 아키텍처
  - 응용계층
    - 응용 로직과 상황 통신 모듈 부분으로 구성

[그림 15-9] 상황 정보 전달을 위한 미들웨어 아키텍처-응용 계층 [참고자료: 54]

    - 상황 통신 모듈
      - 상황 기반의 프로토콜(CXTP))을 이용하여 **상황 서비스 계층**으로부터 상황 정보를 요청하고 받는 역할을 함
      - 응용의 메시지 요구 형태
        - » "subscribe presence Kugsang" 메시지는 Kugsang ID의 참석 서비스를 신청한다는 것을 의미

4. 상황 정보 전달을 위한 미들웨어

- 미들웨어를 통한 상황 정보 전달 시나리오
  - 센서는 환경 정보를 센싱하여 미들웨어에게 전달하고 상황 정보 전달을 위한 미들웨어는 이를 필요한 응용에게 전달
  - 센서는 먼저 센서 커뮤니케이터와 센서 벤더(Vendor) 의존적인 인터페이스를 사용하여 센싱 정보를 전달
  - 센서 커뮤니케이터는 질의 또는 이벤트라는 공통 센서 인터페이스를 통해 에이전트에게 전달
  - 에이전트는 센싱 정보를 상황 관리 객체로 바꾸어 관리 통신 프로토콜(SNMP)을 통해 상황 서비스 계층의 관리자에게 전달
  - 상황 서비스 계층에서는 넘겨온 센싱 정보를 센싱 정보 서비스 등을 수행하여 고급의 상황 정보로 변환하고 이를 저장
  - 자신이 필요한 상황 정보를 상황 기반의 전송 프로토콜(CXTP)을 통해 상황 서비스 계층에게 알린 후, 상황 정보가 적절한 서비스를 수행한 후 응용에게 전달

감사합니다