

# 유비쿼터스 멀티 서비스 개발에서의 일반적 상황모형 구축을 위한 상황모형 비교 평가방법론\*

박태환  
경희대학교 RCUBS 연구원  
(serveLord@khu.ac.kr)

권오병  
경희대학교 국제경영학부  
(obkwon@khu.ac.kr)

.....

유비쿼터스 시스템의 중요한 목적은 사용자의 상황을 빠르고 정확하게 인식하여, 선용적인 서비스를 사용자에게 제공해 주는 것이다. 또한 우수한 상황인식 시스템을 위해서는 상황모형이 시스템의 목적에 맞게 잘 설계되어야 한다. 특별히 유비쿼터스 시스템이 특정 도메인의 제약을 뛰어 넘어 통합 환경에서 서비스 제공하는 것을 지향하기 때문에, 기존의 단일 혹은 한정적 서비스만을 고려한 도메인 제한적 상황모형보다는 다중적 서비스에 활용될 수도 있도록 보다 더 일반적인 상황모형(generic context model)을 활용하는 것이 바람직할 것이다. 따라서 본 논문은 다중적 유비쿼터스 서비스에 적합한 상황모형을 이용하여 이질적인 유비쿼터스 환경에서 상황정보를 보다 잘 표현할 수 있도록 하는 일반적 상황모형을 제안하는 것을 목적으로 한다.

.....

논문접수일 : 2006년 07월      게재확정일 : 2007년 02월      교신저자 : 권오병

## 1. 서론

U-도시 등 복합적인 유비쿼터스 지능 공간 개발은 복수개의 서비스가 등장하고 동시에 사용될 것이다. 이러한 멀티 서비스를 고려할 때 더욱 일반적인 상황모형을 고려하는 것이 필요하다. 이때 상황(context)을 인식하는 것은 유비쿼터스 환경을 구축하는데 있어서 가장 기본적인 절차이면서도 가장 핵심적인 부분이다. 특정 상황에서 어떠한 개체들에 의미를 부여하여 유비쿼터스 시스템 개발에 쓸 것인지는 시스템 운용의 성과와도 직접 관련되어 있기 때문이다. 따라서 시스템 개발자는

어떤 서비스에서 유비쿼터스 환경을 구축할 것인지를 고려할 것이고 그 서비스에서 의미 있는 상황을 고려하여 설계하게 된다. 즉, 각 서비스에 따라 다른 상황모형을 사용하는 경향이 있다. 비록 현재 연구되고 있는 유비쿼터스 서비스들이 특정 도메인 및 서비스에 한하고 하지만, 궁극적으로는 멀티 서비스를 고려하여 생활 환경에서 어떠한 서비스의 출현도 허용할 정도로 충분히 일반적인(generic) 상황모형을 구축해야 할 것이다.

이에 따라 현재까지 지속되고 있는 많은 유비쿼터스 컴퓨팅 연구들은 여러 가지 모델링 방법으로 특정 도메인에 종속적인 서비스를 위한 상황모형

\* 본 연구는 한국IBM의 Celadon Project의 상황인식기술조사과제와 서울시 산학연 협력사업(과제번호: 10802)의 재래 시장 활성화를 위한 u-Market 개발사업의 지원에 의한 것임.

들을 설계하였고, 그것들을 기반으로 상황을 상황 인식 시스템을 구현하고 있다. 이렇게 만들어진 상황모형들 중 가장 많이 사용되어 온 것이 모두 여섯 가지 정도인데 그것은 키벨류 모형, 개체관계 모형, 마크업 스키마 모형, 객체지향 모형, 논리 모형 그리고 온톨로지 기반 모형들이다.

하지만 이러한 연구들은 왜 그러한 상황 모형을 가정하였는지에 대한 객관적인 근거를 제시하지 않고 있으며 당연히 다른 상황모형과의 비교를 하지 않았다. 그래서 이들을 비교 평가하려는 시도들도 진행된 바 있다(Strang and Linnhoff-Popien, 2004). 즉, 비교하는 상황모형들의 장단점이 무엇 인지를 분석한 다음, 유비쿼터스 시스템에 가장 적합할 것으로 보이는 상황모형을 선택하여 제안하였다. 하지만 역시 상황모형을 사용할 주체로 인간인 개발자에 초점이 맞추어져 있었기 때문에 상황모형이 에이전트 시스템과 얼마나 적합한 지에 대한 고려는 부족했다. 더욱이 상황모형들을 체계적인 비교 방법론을 제시하여 비교 평가하지 않고, 주관적인 판단으로 각 모형들을 비교 평가하여 유비쿼터스 컴퓨팅에 적절한 상황모형을 제안하고 있다는 문제점이 있었다. 따라서 상황모형을 사용할 주체들을 일반인, 개발자 및 에이전트 시스템으로 분류하고 각 사용 주체들에게 적합한 상황모형들을 제안하기 위해 객관적이고 체계적인 비교 평가 방법론을 개발할 것이 필요해졌다.

따라서 본 논문의 목적은 이제까지 제안된 상황모형들을 유형별로 분류한 다음, 이들을 몇 가지 기준으로 비교하여 유비쿼터스 시스템 에이전트가 상황을 표현하는데 가장 적절한 상황모형의 형태를 제안하고, 유비쿼터스 환경에서 의미 있는 상황개체로 쓰일 수 있는 요소들을 추출하여 그러한 요소들을 표현할 수 있는 일반적인 상황모형을 제안하는 것이다. 이를 위해 Strang와 Linnhoff-Po-

pien의 기존 연구를 기반으로 하여 여섯 가지 형태의 상황모형을 비교 평가에 상정하였다.

이를 위해 본 논문을 다음과 같이 구성하였다. 제 2장은 이제까지 제안된 상황모형들을 6가지로 분류해 보고(Strang and Linnhoff-Popien, 2004), 제 3장에서는 Strang과 Linnhoff-Popien이 분류한 상황모형들을 모형단순도와 모형표현도 등 두 가지 평가항목을 개발하여 비교 평가하였다. 제 4장에서는 유비쿼터스 시스템에 가장 적합한 상황모형을 선택하고, 유비쿼터스 환경에서 의미 있게 쓰일 수 있는 상황 개체들을 조사하여 일반적인 상황모형을 디자인하고, 제안한 상황모형을 바탕으로 하여 OWL-DL 파일을 생성하였다. 마지막으로 제 5장에서는 결론을 기술하였다.

## 2. 상황모형

지금까지 많은 연구자들이 상황에 대해 정의하고, 특정 도메인에 의존적인 혹은 일반적인 상황모형을 제시해왔다. 이제까지 제안된 상황모형은 다음과 같이 크게 6가지의 형태로 분류할 수 있다.(Strang and Linnhoff-Popien, 2004).

### 2.1 키 벨류(KV) 상황모형

이 모형은 가장 간단한 데이터 구조로 상황정보를 표현하는 상황모형이다(Schilit et al., 1994; Samulowitz et al., 2001). [그림 1]과 같이 이 상황모형은 상황정보를 나타내는 상황개체로 몇 개의 중요한 키 개체들만을 사용하였다. 초기의 상황모형들은 대부분 이를 바탕으로 해서 어플리케이션을 제작하였다. 초기에 사용자와 그 위치정보로만 상황모형을 디자인했으나, 점차 유비쿼터스 컴퓨팅

기술이 발전함에 따라 장치, 환경요소 등 다른 상황 개체들을 추가하는 형태로 이 상황모형은 발전되었다.

Name	Room	Distance
caps	35-2200	200ft
claudia	35-2108	30ft
perfector	35-2301	20ft
snoball	35-2103	100ft

(a)

Distance	Name	Room
20ft	perfector	35-2301
30ft	claudia	35-2108
100ft	snoball	35-2103
200ft	caps	35-2200

(b)

Name	Room	Distance
caps	35-2200	200ft
claudia	35-2108	30ft
perfector	35-2301	20ft
snoball	35-2103	100ft

(c)

Name	Room	Distance
caps	35-2200	200ft
claudia	35-2108	30ft
perfector	35-2301	20ft
snoball	35-2103	100ft

(d)

[그림 1] 키 밸류 상황모형의 예(Schilit et al., 1994)

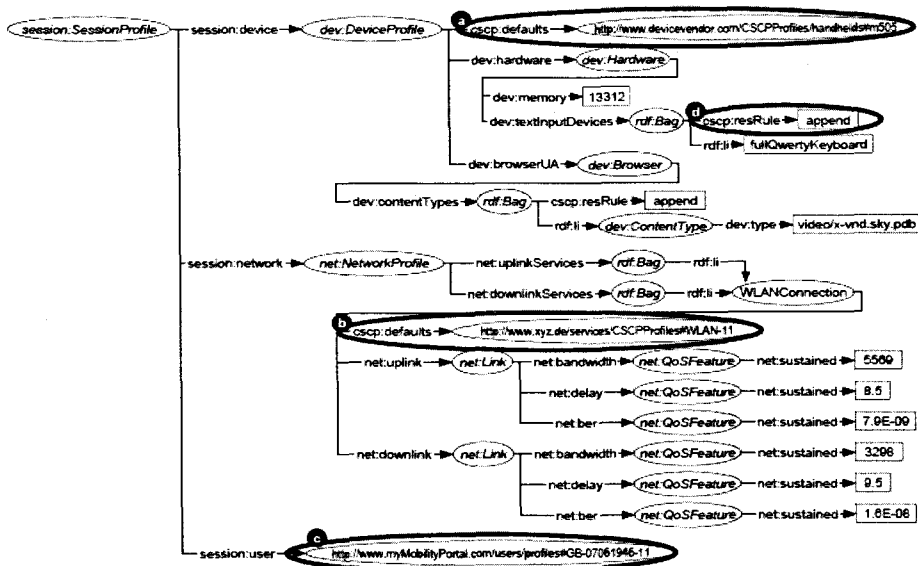
### 2.2 Markup Scheme(MS) 상황모형

이 모형의 특징은 [그림 2]와 같이 계층적인 데이터 구조가 속성과 내용으로 된 마크업 태그들로

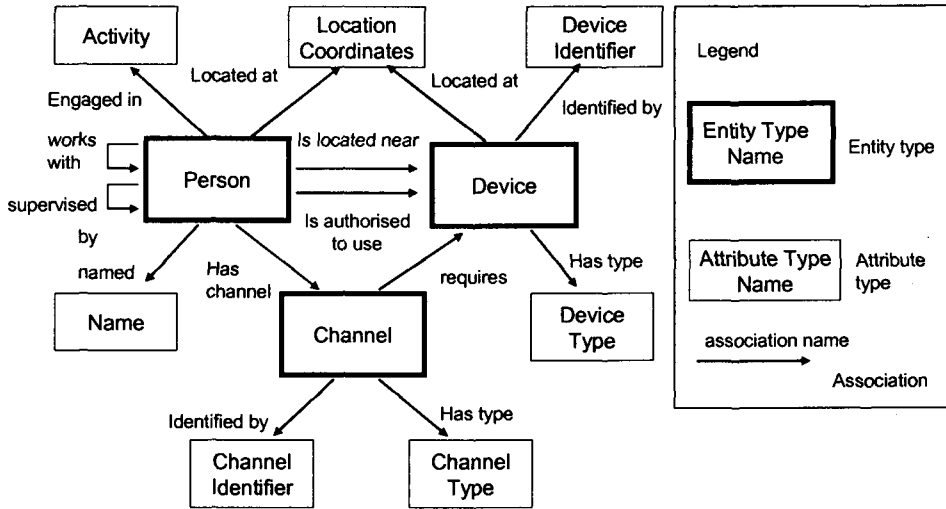
구성되어 있다는 것이다(Held et al., 2002). 이 상황모형은 간단하고, 유연하며, 구조화 되어 있고 편재적인 컴퓨팅에 적합하다.

### 2.3 개체관계(ER) 상황모형

매우 잘 알려진 일반적인 목적의 모형화 도구인 UML을 이용하여 UML diagram이라는 매우 강력한 그래픽 컴포넌트를 가지고 있다. 이 상황모형은 주로 상황 정보의 구조화를 기술하는데 쓰이며, 상황 모형으로부터 어플리케이션의 필요에 맞게 몇 가지 코드나 개체 관계 모형을 이끌어 내는 데 유용하다(Pascoe, 1998; Halpin et al., 2001; Henricksen et al., 2002; Indulska et al., 2003; Chtcherbina et al., 2003; Henricksen et al., 2003; Bauer et al., 2003; Henricksen et al., 2004). 표현의 예는 [그림 3]과 같다.



[그림 2] RDF로 표현한 프로필(Held et al., 2002)



[그림 3] 개체관계 상황모형의 예(Henricksen et al. 2004)

### 2.4 객체지향(OO) 상황모형

이 상황모형은 캡슐화(encapsulation), 재사용성(reusability), 상속성(inheritance) 등의 객체지향의 장점을 최대한 활용하도록 제작되었다(Schmidt et al., 1998; Cheverst et al., 1999; Schmidt et al., 1999; Hofer et al., 2002). 주로 동적인 유비쿼터스 환경에서 발생하는 문제를 해결할 때는 객체지향의 특징인 캡슐화와 재사용성을 사용하기 때문에 이

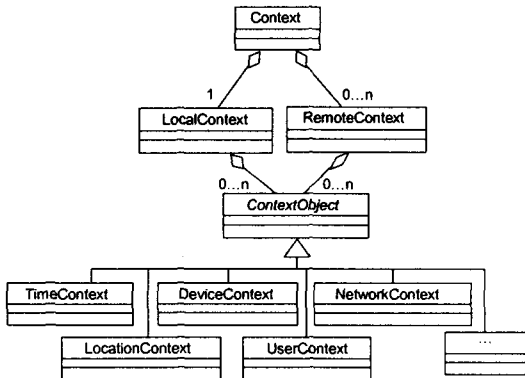
모형이 커다란 장점을 가진다. 객체지향 상황모형의 예로는 [그림 4]와 같다.

### 2.5 논리기반(LB) 상황모형

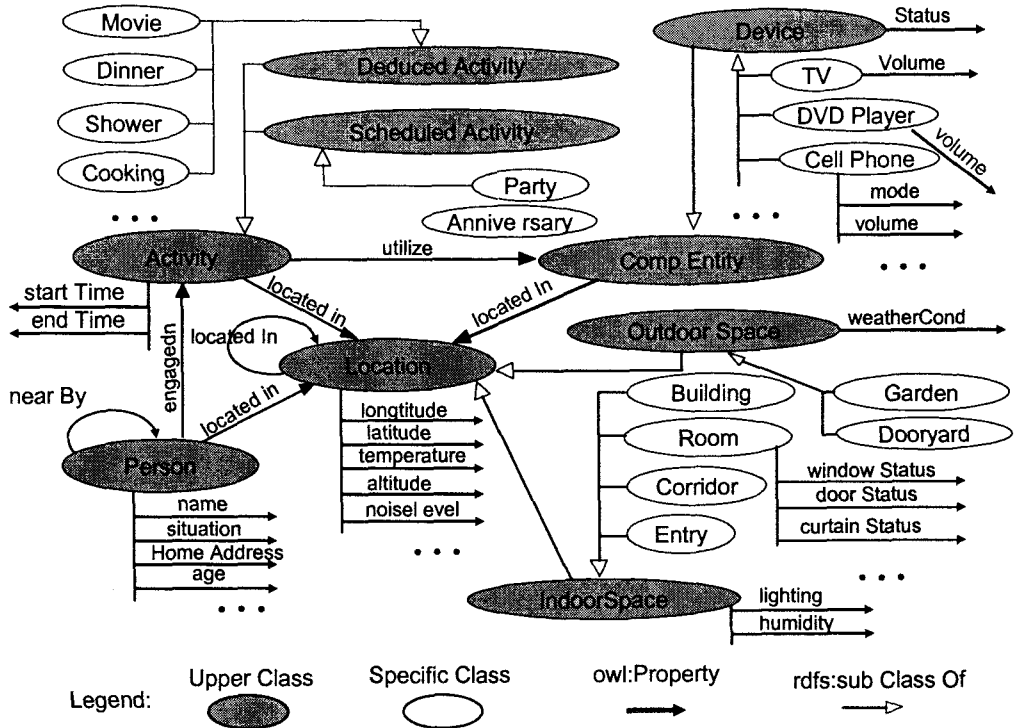
이 상황모형은 매우 높은 정형성을 갖고 있고, 전형적으로 사실, 표현, 규칙으로 상황을 정의한다(Gray et al., 2001). 따라서 이 모형 시스템에서는 앞에서 언급한 용어를 관리할 뿐만 아니라, 새로운 사실을 추가, 갱신, 제거할 수 있다.

### 2.6 온톨로지 기반(OB) 상황모형

온톨로지는 객체와 객체, 객체와 속성, 속성과 속성간의 상호관계성을 표현하는데 적합한 도구이다. [그림 5]에서 보이는 바와 같이, 이 상황모형은 컴퓨터에 의해 일상 생활을 데이터 구조화하고 부분적인 상황정보를 기술하는 데도 적합하다(Otzurk et al., 1997; Strang et al., 2003; Wang et al., 2004; Dogac et al., 2004; Preuveneers et al., 2004). 상황 온톨로지는 개체 속성 표현과 관계 정의



[그림 4] 객체지향 상황모형의 예(Hofer et al., 2002)



[그림 5] 온톨로지 기반 상황모형의 예(Wang et al., 2004)

시, 어플리케이션 개발자의 일을 간편하게 해주고, 간단히 새로운 상황 요소와 관계를 추가할 수 있으며, 특별한 상황 종류에 제한되지 않고, 서로 다른 상황에도 적용가능하다는 특징을 가지고 있다 (Korpipaa et al., 2003).

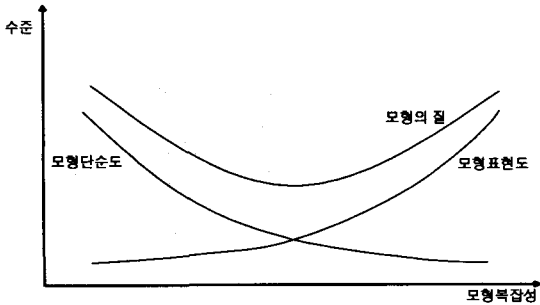
### 3. 상황모형 비교 평가

#### 3.1 모형 비교 평가에 대한 개요

모형의 질에 대한 비교 평가의 일반적인 접근은 모형의 복잡성에 따라 두 가지의 상충되는 차원인 모형단순도 또는 모형 이해가능성과 모형표현도

의 함으로 이루어져 왔다. 따라서 본 연구에서도 이러한 일반적인 평가 차원을 그대로 고려하였다.

[그림 6]에서 보는 바와 같이 모형의 질(수준)은 모형표현도와 모형단순도의 함으로 나타낼 수 있다. 하지만 모형의 단순도와 모형의 표현도는 서로 상반되는 성질을 가지고 있기 때문에, 어떤 쪽을 선택 해야 하는가는 그 모형을 사용하는 사용자에게 따라 달라지게 된다. 예를 들어, 초보자 같은 경우는 모형표현도 보다는 모형단순도가 높아야 더 좋은 모형이라고 느낄 것이다. 따라서 사용자를 크게 초보자, 전문가, 에이전트 시스템으로 나누고, 각각의 모형 사용 주체에 가장 적합한 상황모형을 다음과 같은 방법을 통해 선정할 것이다.



[그림 6] 모형단순도와 모형표현도로 측정된 모형의 질

### 3.2 평가를 위한 예제 시나리오

모형의 표현도와 단순도를 평가하기 위하여 사용된 시나리오는 다음 <표 1>과 같이 모두 여섯 가지이다.

이들 각각의 시나리오에 대해서 비교할 여섯 가지 상황모형을 [그림 7]에서 [그림 12]와 같이 작성하였다. 이중 시나리오 1에 대한 예만 보이면 다음과 같다.

<표 1> 평가에 활용된 시나리오

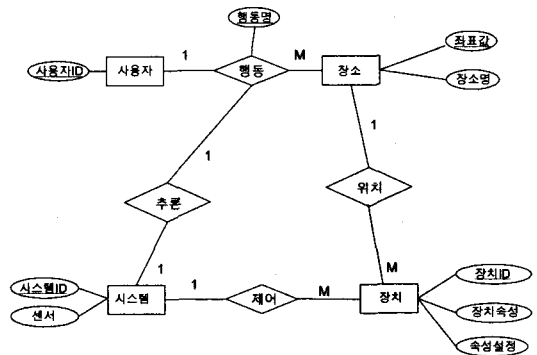
시나리오	시나리오	출처
시나리오 1	사용자가 침실에서 자고 있거나 욕실에서 샤워하고 있을 때는 걸려온 전화는 음성 메일 박스로 전해진다. 사용자가 부엌에서 요리하고 있거나 거실에서 TV를 보고 있을 때는 벨 소리의 음량이 더욱 커지게 된다. 사용자가 식당에서 가족들과 함께 식사를 할 때는 전화기는 진동 모드로 설정 된다.	Xiao Hang Wang et al., 2004
시나리오 2	장거리 비행을 한 Maria는 극동의 한 공항에 착륙했다. 이 곳의 출입국 관리소에서는 직원을 대신하는 장치를 사용해 출입국 검사를 한다. 즉, 출입국 관리 장치가 센서를 통해 출입국 장으로 들어오는 사람의 신원을 확인하고, 여권 확인 및 비자 업무도 처리한다. Maria는 그녀의 비자가 정리되어 있는 개인적인 소프트웨어 에이전트와 그녀의 신분 증명 정보를 가지고 있는 P-Com 이라는 장치를 통해 출입국 업무를 손쉽게 보고 나왔다. 공항을 나오자, 개인 소프트웨어 에이전트가 예약해 놓은 렌터카가 기다리고 있었고, 역시 개인 에이전트가 예약해 놓은 호텔로 향했다.	A. Dogac et al., 2004
시나리오 3	Bob은 Alice의 리포트를 모두 읽고 난 다음, 리포트에 대해 그녀와 함께 코멘트를 나누기 위해 그의 커뮤니케이션 에이전트를 통해 Alice와 접촉하도록 요청하였다. 그러나 Alice는 지금 한 학생과 회의를 하고 있어서, 그녀의 에이전트는 그녀를 방해할 수 없다는 결정을 내렸다. Alice의 에이전트는 Bob의 에이전트에게 연락하여 이 메일로 연락을 하거나, 30분 후에 만날 것인지를 묻는다. Bob의 에이전트는 그의	K. Henricksen et al., 2002
시나리오 3	스케줄을 체크하여 Alice의 에이전트가 추천한 시간이 적절하지 않음을 알고, Alice의 에이전트가 알려준 이메일 주소로 이메일을 전송한다. 잠시 후, Alice의 상관인 Charles는 그가 요청했던 리포트가 준비가 되었는지를 알아보기 위해 연락한다. Alice의 에이전트는 Charles의 요청이 긴급함을 알고 즉시, Charles에게 그녀의 사무실 전화로 연락할 것을 제안한다. Charles의 에이전트는 그가 지금 가지고 있는 핸드폰을 통해서 연락을 취하도록 한다.	K. Henricksen et al., 2002
시나리오 4	어시스턴트는 사무실 문에서부터 방문자와 상호작용을 하고, 사무실 소유자의 스케줄을 관리하는 에이전트를 일컫는다. 어시스턴트는 문 앞 매트에 위치한 두 개의 압력 센서에 의해 방문자가 접근하면 작동하여, 방문자의 신원확인과 사무실 소유자의 스케줄 상태 그리고 사무실 소유자가 방문자를 만나길 원하는 지의 여부를 바탕으로 다음 행동을 추천한다.	시나리오(G. Chen et al., 2000)
시나리오 5	Patrick은 그의 PDA를 가지고 거실로 들어선다. 거실에는 플라즈마 디스플레이가 있고, 음향 시스템도 있으며, 디지털 영상 프레임도 있다. Patrick이 그의 PDA를 켜고, 개인화 된 포털 웹 브라우저를 열었다. 포털 브라우저는 이미 Patrick의 신체 환경을 검사해 그의 개인화 된 음악과 비디오 영상 및 화면을 로컬로 보여준다. 스크린의 한 탭에서 그는 로컬 사운드 시스템에서 U2의 음악이 나오도록 선택한다. 그가 음악을 재생할 수 있도록 웹 폼을 받는다. Alex와 Philippe이 거실 안으로 들어오자 스테레오에서 음악이 들린다. Alex가 "이 음악은 내가 정말 좋아하는 U2 곡으로 알고 있는데, 지난 주에 Philippe의 집에 갔을 때 우리 같이 들었던 곡이잖아!"	P. Debaty et al., 2003

시나리오	시나리오	출처
시나리오 5	<p>라고 말한다. Patrick이 그의 PDA를 쳐다본다. 그의 포털 페이지에 또한 다음과 같은 하이퍼링크가 된 메시지가 보여진다. "사람들이 거실에 있다." "이 거실에서 이미 사용된 파일 연결" "이 사람들과 같이 있을 때 사용된 파일 연결". 그는 마지막 메시지를 고르고, Alex와 Philippe이 같이 있을 때 사용한 파일 중 가장 최근 것을 선택한다.</p> <p>그는 Alex가 언급한 곡을 빠르게 찾은 다음, 음향 시스템으로 그 음악을 듣는다. 이러한 사이에 Philippe은 자신의 PDA를 사용하여 디지털 영상 프레임을 제어하여 플라즈마 디스플레이에서는 그의 사진을 보여주고, 비디오에서는 지난 주에 갔던 U2 콘서트의 장면이 나오도록 한다. 그들은 모두 그 콘서트에 대한 이야기를 하고 있다.</p>	P. Debaty et al., 2003
시나리오 6	<p>Maria와 Jerry의 아들인 Robert는 비디오 게임을 하기 위해 그의 친구를 기다리고 있다. Robert의 친구는 새로 산 휴대용 DVD 플레이어를 가지고 도착한다. 그래서 그는 게임을 하기 보다는 영화를 보자고 Robert에게 제안했다. 또한 Robert에게 그가 가지고 온 디바이스에 있는 필름 중 하나를 볼 것인지 아니면, Robert의 집에 있는 영화 데이터베이스에서 필름 하나를 골라서 볼 것인지를 물었다. Robert는 친구의 DVD 플레이어를 사용하기 위해 그 장치가 게스트 장치인지와 홈 서비스에 사용되도록 허가되어 있는지를 물었다. 이 플레이어는 사용자의 정보가 올바르게 입력되어야만 사용할 수 있도록 되어 있었다. 첫 번째 정보는 사용자의 신체적 환경과 선호도에 따라 비디오가 작동하며, 두 번째로는 사용자의 연령을 체크하여 선택한 영화의 등급을 결정한다.</p>	S. B. Mokhtar et al., 2002

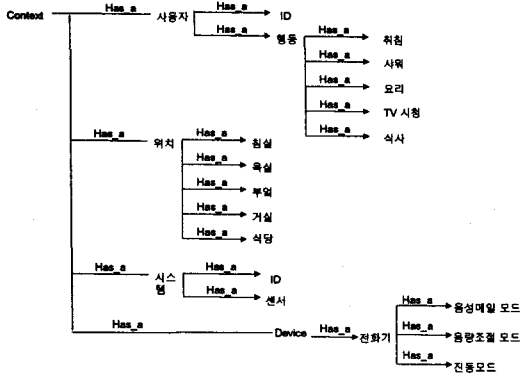
이와 같이 모형을 작성한 후에 각 모형에 대해서 표현도와 단순도를 측정하였다.

사용자	장소	장치
사용자	침실	전화기-음성메일 모드
사용자	욕실	전화기-음성메일 모드
사용자	부엌	전화기-음량조절 모드
사용자	거실	전화기-음량조절 모드
사용자	식당	전화기-진동 모드

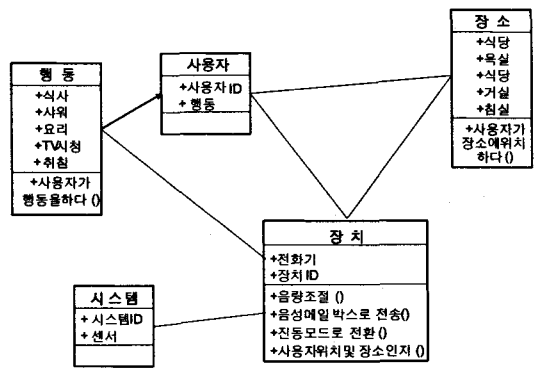
[그림 7] 시나리오 1에 대한 키-밸류 모형



[그림 9] 시나리오 1에 대한 개체 관계 모형



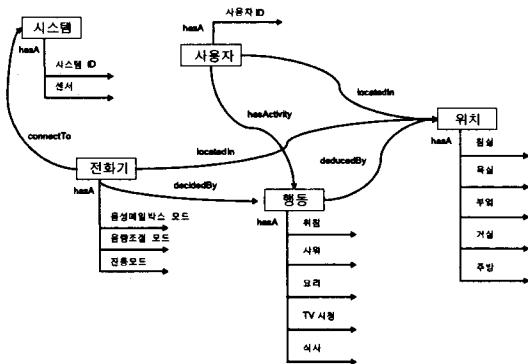
[그림 8] 시나리오 1에 대한 Markup Scheme 모형



[그림 10] 시나리오 1에 대한 객체지향 모형

사용자	기호	장소	기호	행동	기호	장치(전화기)
사용자	∩	식당	∩	식사	⇒	전동 모드
사용자	∩	목실	∩	샤워	⇒	음성메일박스 모드
사용자	∩	주방	∩	요리	⇒	음량조절 모드(음량小)
사용자	∩	거실	∩	TV시청	⇒	음량조절 모드(음량大)
사용자	∩	침실	∩	취침	⇒	음성메일박스 모드

[그림 11] 시나리오 1에 대한 논리기반 모형



[그림 12] 시나리오 1에 대한 온톨로지 기반 모형

### 3.3 모형의 표현도 비교

모형표현도는 상황모형이 상황을 기술할 때 모호하지 않되 모든 상황 정보들을 충분히 나타낼 수 있어야 하기 때문이다. 나아가 모형표현도를 측정할 때, 상황개체의 속성을 얼마나 자세히 표현할 수 있는가, 상황 개체들 간의 관계를 명확히 규정할 수 있는가, 상황 개체들 간의 논리적 연관성을 표현할 수 있는가, 논리적 표현과 계층적 표현이 가능한가 등의 항목으로 나눠 고찰하였다. 단, 여섯 가지의 시나리오를 표현한 각 상황모형은 2004년에 Strang과 Linnhoff-Popien이 비교 분석할 때 제시한 것들만을 대상으로 하였다. 다만 일부 인정된 확장형은 그것이 상용화되어 널리 사용되는 것

들이며 그 외에는 평가에서 고려하지 않았다.

<표 1> 유형별 모형표현도 비교표

상황모형 평가항목	KV	MS	ER	OO	LB	OB
개체 표현	○	○	○	○	○	○
속성 표현	○	○	○	○	○	○
개체간 관계표현		○	○○	○	○	○○
속성간 관계표현		○○			○	○○
논리표현		△ (RuleML)		△ (함수에서 표현가능)	○○	○
계층표현		○	△ (ECR)	○○		○○
행위(함수) 표현			△ (EER)	○○	○	○

먼저 키-밸류(KV) 모형은 개체 표현과 속성표현만을 할 수 있기 때문에 두 항목만 점수를 주었다. 두 번째로 Markup Scheme(MS) 모형은 개체 표현과 속성표현, 개체간 관계 표현, 속성간 관계 표현, 계층표현이 가능하고 논리 표현도 RuleML을 통해 제한적이거나 가능하므로 이들 항목에 일부 점수를 주었다. 특히 속성간 관계 표현에 있어서는 마크업 태그들을 사용하여 더욱 강점을 갖는다. 세 번째로 개체관계(ER) 모형은 개체표현, 속성표현, 개체간 관계 표현이 가능하고 계층표현은 개체관계모형의 확장형인 ECR(Entity Category Relation)모형을 통해, 행위(함수) 표현은 EER(Extended ER)을 통해 어느 정도 가능하다. 개체관계모형은 개체들 간의 관계를 나타내기 위해 개발되었기 때문에 개체간 관계 표현에 매우 적합하다. 네 번째로 객체지향(OO) 모형은 개체표현, 속성표현, 개체간 관계표현, 계층표현, 행위(함수) 표현이 가능하고 함수에서 논리표현도 가능하다. 이 모형은 특히 객체지향의 특징인 상속성을 바탕으로 계층표현에 적합하며, 함수를 사용하여 행위를 표현하므로 행위 표현에 더욱 적합한 것으로 인정된다.



다섯 번째로 논리기반(LB) 모형은 개체표현, 속성표현, 개체간 관계 표현, 속성간 관계 표현, 논리표현, 행위(함수) 표현이 가능한 것으로 알려졌다. 이 모형은 다른 모형과는 달리 언어적 표현을 많이 허용하므로 논리표현에 강점을 지닌다. 마지막으로 온톨로지 기반(OB) 모형은 객체지향의 클래스와 인스턴스의 개념과 유사한, 개념(concepts)과 사실(facts)로 상황정보를 표현한다. 또한 데이터 속성 표현 및 개체들 간의 관계 표현이 가능하고, 개체들간의 관계에 있어서 방향성과 언어적 표현을 병행하여 계층표현 및 논리표현이 우수하다. 이와 같은 평가에 근거하여 7점 만점으로 모형의 표현도를 측정하였다.

### 3.4 모형의 단순도 비교

두 번째 평가 차원인 모형단순도는 전술한 여섯

가지의 시나리오(X.H. Wang *et al.*, 2004; A. Dogac *et al.*, 2004; K. Henricksen ., 2002; G. Chen *et al.*, 2000; P. Debaty *et al.*, 2003; S. B. Mokhtar *et al.*, 2002)를 각 상황모형들로 표현하고, 각 모형들에 쓰인 개념(concept)과 값(value)의 수를 기초로 하여 산출해 내었다. 여기서 개념들이란 실제 상황 값을 체계적으로 조직하여 표현하기 위해 고려된 요소들로서 개체, 관계, 연결 흐름 등 값을 제외한 모든 요소들이 이에 해당한다. 단, 연결 흐름의 경우 단순한 연결관계는 0.5점의 가중치를 주었고 태그가 붙어서 설명을 해주는 연결관계는 1점의 가중치를 주었다. 그 결과 <표 2>는 6개의 시나리오를 각 모형으로 모델링 했을 때 각 모형이 가지고 있는 개념(개체, 관계, 속성)의 개수와 값의 개수를 각각 나타낸 것이다.

모형의 단순도 평가를 위해 상대적 모형 단순도

<표 2> 6개 시나리오에 따른 모형 별 개념과 값 비교표

		KV	MS	ER	OO	LB	OB
Number of Concept (Nc)	시나리오 1	3	29	18	19	22	27
	시나리오 2	3	17	15	11.5	15	16
	시나리오 3	3	17	17.5	11	16	18
	시나리오 4	3	18	17.5	10.5	10	17
	시나리오 5	3	24	30	18	13	31
	시나리오 6	3	24	31	18	10	30
Number of Value (Nv)	시나리오 1	15	16	9	22	20	16
	시나리오 2	9	10	6	14	9	10
	시나리오 3	9	10	11	13	12	12
	시나리오 4	6	11	11	12	4	11
	시나리오 5	9	13	14	18	8	17
	시나리오 6	6	14	16	18	4	17

와 상대적 모형 경량도라고 하는 두 가지 측정 항목을 개발하고 <표 2>의 값들을 토대로 하여 평가해 보았다. 상대적 모형파악도(Degree of Relative Model Grasp)는 다음 (1)과 같이 표현된다.

$$\text{상대적 모형파악도} = \frac{\text{모형파악도}}{\text{모형파악도 최대치}} \quad (1)$$

이때 모형파악도는 다음 (2)와 같이 개념의 수와 값의 수의 함수로 표현된다.

$$\text{모형파악도} = 1/\alpha^*N_C + (1-\alpha)^*N_V \quad (2)$$

단  $\alpha, 0 \leq \alpha \leq 1$ , 는 모형파악을 함에 있어서 단순한 값보다 개념이 얼마나 중요하게 영향을 주는가에 관한 가중치이다. 한편 모형경량도(Degree of Model Lightness)는 하나의 값을 표현하기 위하여 얼마나 많은 개념들이 활용되었는지에 대한 지표로서 다음 (3)과 같이 표현된다. 이는  $N_C/N_V$ 로 표현되는 overhead의 역수이다.

$$\text{모형경량도} = N_V/N_C \quad (3)$$

이러한 두 가지 척도를 기반으로 하여 모형단순도는 다음 (4)와 같이 구성된다.

$$\text{모형단순도} = \beta^* \text{모형파악도} + (1-\beta)^* \text{모형경량도} \quad (4)$$

단  $\beta, 0 \leq \beta \leq 1$ 는 모형 단순도에 모형파악도가 상대적으로 얼마나 더 영향을 주는지에 대한 가중치이다.

가중평균과 모형단순도, 상대적 모형 단순도, 모형경량도, 모형단순도를 구하여 <표 3>과 같이 모형 별로 정리하였다. 단  $\alpha$ 와  $\beta$ 는 공히 0.5로 하였다.

한편  $\alpha$ 값의 변화가 각 모형 유형별로 영향을 어떻게 주는지를 알아보기 위해  $\alpha$ 값을 증가시켜 가면서 상대적 모형 파악도의 값을 변화를 관찰하였다. 그 결과  $\alpha$ 민감도에 대해서 <표 4>와 같은 결과를 얻을 수 있었다. 이 결과에 의하면  $\alpha$ 민감도가 높은 것은 논리기반 모형이나 Markup Scheme 모형으로서 모형파악을 함에 있어서 단순한 값보다 개념이 얼마나 중요하게 영향을 주는가에 관한 일치된 견해가 없을 경우 가장 평가에 영향을 받을 것으로 보였다. 반면 객체지향 모형이나 온톨로

<표 3> 상황모형 유형별 모형단순도 비교표 ( $\alpha=0.5, \beta=0.5$ )

	KV	MS	ER	OO	LB	OB
$\alpha^*N_C + (1-\alpha)^*N_V$	6	16.916	16.333	15.416	11.916	18.5
모형파악도 $1/\{\alpha^*N_C + (1-\alpha)^*N_V\}$	0.166	0.0591	0.061	0.064	0.083	0.054
상대적 모형 파악도-①	1	0.354	0.367	0.389	0.503	0.324
Overhead	0.333	1.743	1.925	0.907	1.508	1.674
모형 경량도 (= 1/overhead)	3	0.573	0.519	1.102	0.662	0.597
상대적 모형 경량도-②	1	0.191	0.173	0.367	0.220	0.199
모형단순도 { $\beta \cdot \text{①} + (1-\beta) \cdot \text{②}$ }	1	0.272	0.270	0.378	0.154	0.261

<표 4> 모형 유형별  $\alpha$ 민감도 비교

$\alpha$	KV	MS	ER	OO	LB	OB
0.1	1.0000	0.6554	0.6783	0.5490	0.8796	0.6109
0.3	1.0000	0.4983	0.5618	0.5307	0.6761	0.5301
0.5	1.0000	0.3731	0.4528	0.5070	0.5106	0.4472
0.7	1.0000	0.2709	0.3508	0.4752	0.3735	0.3623
0.9	1.0000	0.1860	0.2550	0.4303	0.2581	0.2752
$\alpha$ 민감도	0.0000	0.1166	0.1058	0.0293	0.1546	0.0839

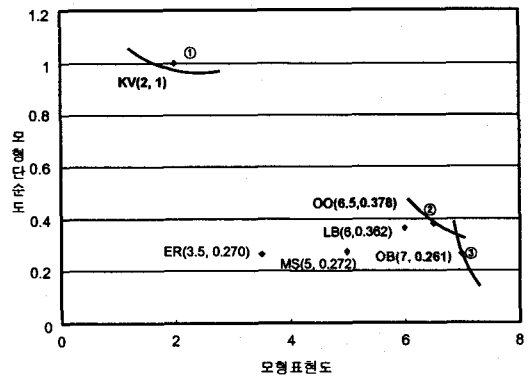
지 기반 모형들은 상대적으로 적은 영향을 받는 것으로 보여 이 유형들에 대한 모형단순도 평가는 비교적 안전한 것으로 보인다.

### 3.4 각 상황모형 유형별 종합평가

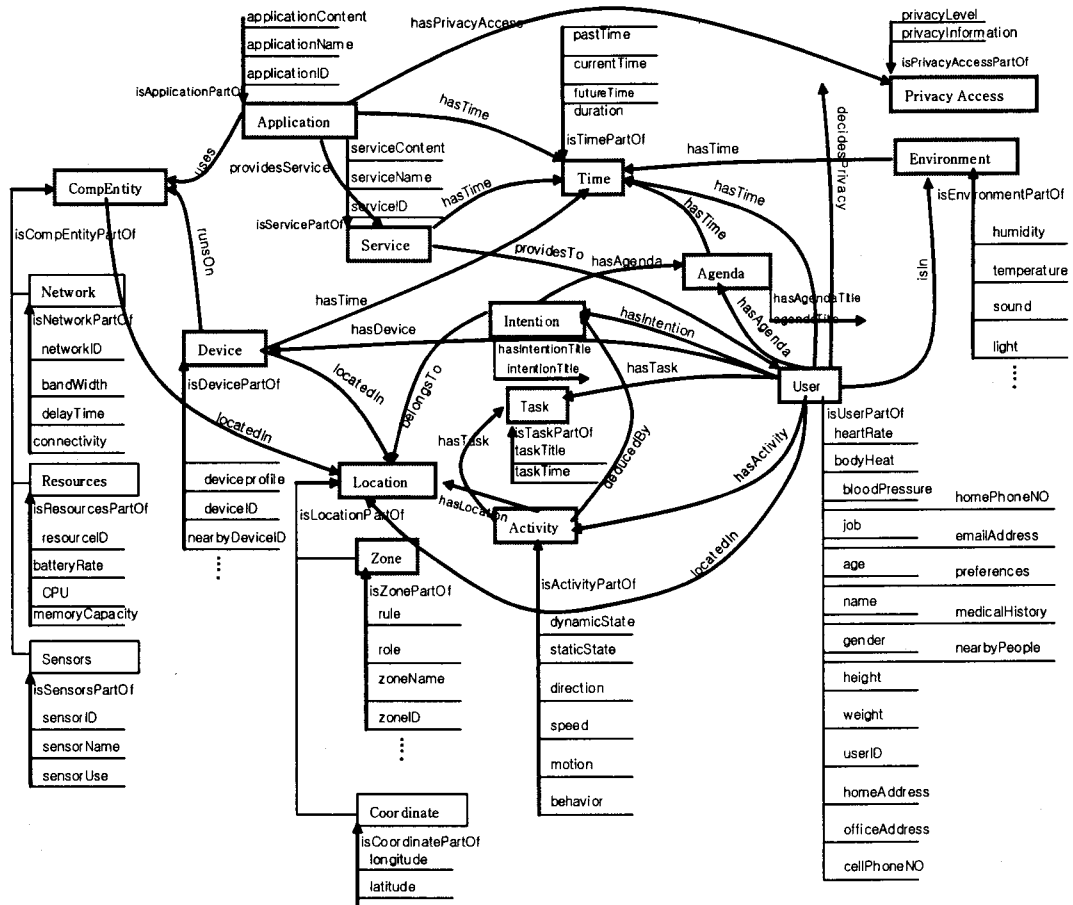
앞에서 실시한 모형표현도와 모형단순도 평가 결과에 의하여 각 상황모형 유형별로 종합적인 평가를 수행할 수 있었다. [그림 13]은 이상에서 알아본 각 상황 모형들의 모형단순도와 모형표현도를 도표화하여 나타난 것이다. 그래프 ①은 초보자들에게 더욱 적합한 모형임을 의미한다. 즉, 초보자들은 상황모형의 표현력보다는 얼마나 알기 쉽게 이해할 수 있는가를 더 중요하게 생각할 것이기 때문에 모형의 단순도는 높고 모형 표현도는 낮은 그래프 ①과 같은 상황모형이 적합할 것이다. 따라서 키-벨류 모형이 가장 적합하다고 볼 수 있다. 그래프 ②는 전문가들에게 더욱 적합한 모형임을 의미한다. 전문가들은 상황을 이해하기 위해서 모형의 단순도 보다는 얼마나 표현이 잘 되었는지를 중요시하기 때문이다. 따라서 모형단순도는 낮으나 모형표현도가 비교적 높은 객체 지향 모형이 전문가들에게 적합할 것이다. 마지막으로 그래프

③은 에이전트 시스템에 적합한 모형임을 의미한다. 에이전트 시스템은 유비쿼터스 환경에서 쓰일 수 있는 모든 상황을 인지하기 원하므로 될 수 있는 대로 표현력을 높이기를 원할 것이다. 따라서 모형단순도는 낮더라도 모형표현도가 더욱 높은 온톨로지 기반 모형이 에이전트 시스템에 가장 적합한 모형이라 할 수 있다.

이러한 평가가 가지는 의미는 상황모형의 우열을 논하는 것이 아니라 그 상황모형이 어떠한 용도로 사용되느냐에 밀접한 관련이 있다는 것이며, 다 나름대로의 의미를 가지고 있지만 에이전트들에 의하여 자율적이고 자동적으로 작동되는 상황



[그림 13] 각 상황모형의 모형단순도와 모형표현도



[그림 14] 온톨로지를 이용한 일반적 상황모형

인식 시스템을 구축하기 위하여는 온톨로지 기반 모형이 가장 적절할 수 있다는 점이다. 그러므로 본 연구에서 제안하려고 하는 일반적 상황모형의 용도가 에이전트 시스템들이 활용할 것임을 고려할 때 온톨로지 기반 모형에 의하는 것이 가장 적절할 것으로 판단하였다.

#### 4. 일반적 상황모형 제안

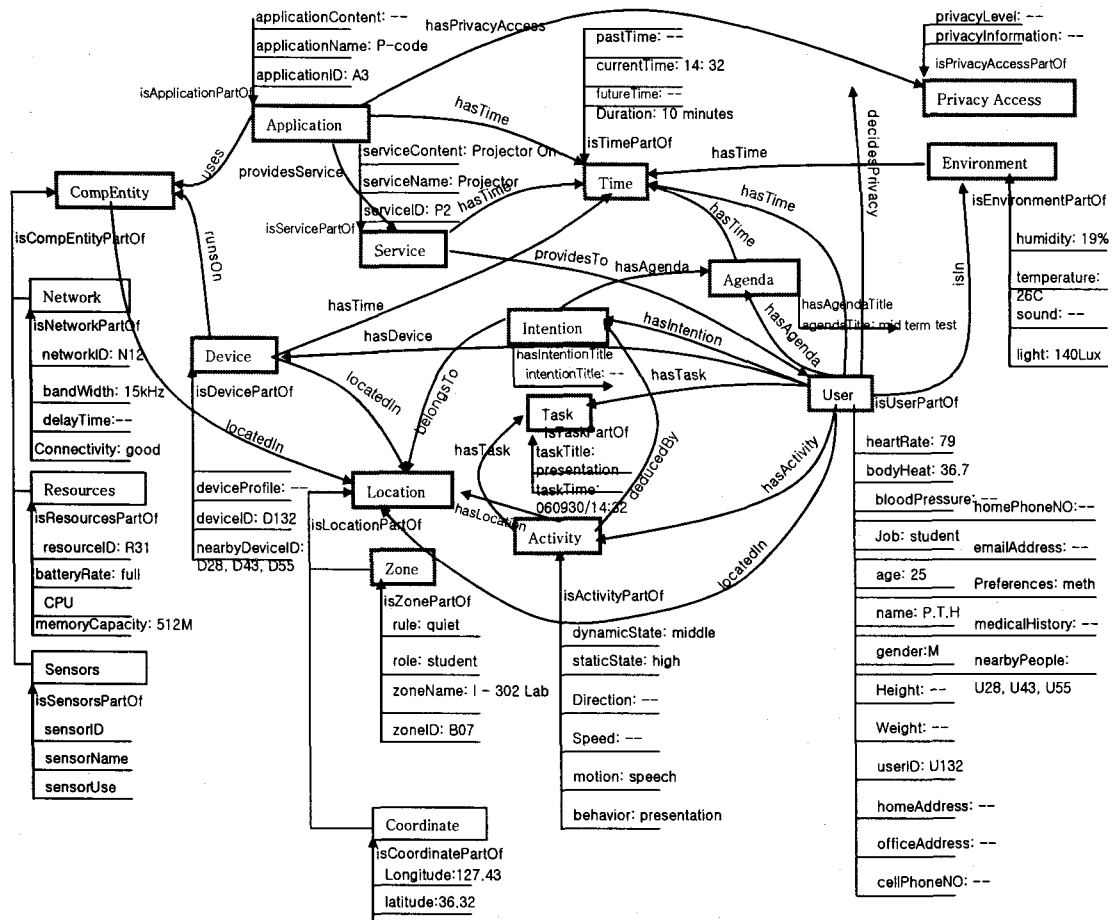
일반적 상황모형을 온톨로지 기반 모형으로 설

계하기 위해 최근 10년간 모바일 컴퓨팅이나 유비쿼터스 컴퓨팅 관련 문헌들을 조사하여 유비쿼터스 시스템에서 의미 있게 쓰일 수 있는 상황 개체들의 합집합을 만들어 각 개체들을 BNF 형식으로 정의하여 도표로 작성하였다. 자세한 정보는 부록 A를 참조할 수 있다. 이 상황 개체들 중 일반적인 유비쿼터스 컴퓨팅 환경을 표현할 수 있도록 하는 14 개의 주요 상황 개체들을 위와 같이 <표 5>로 정리하였다.

위의 14개 주요 상황개체는 다른 최근 10년간 조사된 다른 상황모형에서 쓰여진 상황 개체들을

<표 5> 14개의 주요 상황개체

Context Entities	Context Definition	Context Entities	Context Definition
Intention	* 사용자의 의도 *	CompEntity	* 일정한 구역 내에 존재하는 유비쿼터스 센서 네트워크 인프라 *
Identity	* user or person *	Device	* 사용자가 가지고 있는 장치 *
Task	* 사용자가 현재 하는 업무업무 *	Location	* 위치 *
Agenda	* 사용자의 일정 *	Zone	* 사용자가 위치한 공간 *
Activity	* 사용자가 한 행위(행동) *	Service	= * 사용자에게 제공되는 서비스 *
Application	* 서비스를 제공하기 위한 응용 프로그램 (소프트웨어) *	Environment	= * 사용자의 주변환경 요소 *
Time	* 시간 *	Privacy Access	* 시스템이 접근할 수 있는 사용자의 프라이버시 *



[그림 15] 캠퍼스 도메인 온톨로지 기반 상황모형

모두 포함하고 있어 다른 상황모형보다 일반성 면에서 더 우수함을 부록 A에 의해 확인할 수 있다.

본 연구에서는 동적인 유비쿼터스 환경에 적합한 상황인식 시스템을 만들기 위해서 상황인식에 쓰일 14개의 상황 개체들의 속성 및 개체들 간의 관계를 [그림 14]와 같이 온톨로지로 표현하였다. 이 상황모형은 일반적인 상황 모형이므로 유비쿼터스 환경에서 어떤 상황정보도 표현할 수 있도록 하는 것을 목적으로 한다.

위 일반적인 상황모형은 유비쿼터스 환경의 모든 상황을 표현할 수 있도록 설계되었으므로 특정

도메인에서 이 상황모형을 사용할 때에는 그 도메인에 맞도록 다음과 같이 개체와 속성들의 추가 및 삭제가 가능하다. 그 예로는 [그림 15]에 나타내었다.

더 나아가 본 논문에서 제안한 일반적 상황모형을 기초로 상황인식 시스템의 추론 엔진에 이용될 OWL DL 스키마 파일을 OWL 생성 프로그램인 Protégé 3.2 beta로 [그림 16]과 같이 생성하여 실제 유비쿼터스 서비스에 활용될 수 있도록 하였다. 본 온톨로지는 u-서울 과제에서 활용하고 있다.

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://www.owl-ontologies.com/Ontology1157525210.owl#"
  xml:base="http://www.owl-ontologies.com/Ontology1157525210.owl">
  <owl:Ontology rdf:about="">
    <owl:versionInfo rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
      >classified</owl:versionInfo>
  </owl:Ontology>
  <owl:Class rdf:ID="User"/>
  <owl:Class rdf:ID="Location"/>
  <owl:Class rdf:ID="CompEntity"/>
  <owl:Class rdf:ID="Resources">
    <rdfs:subClassOf rdf:resource="#CompEntity"/>
  </owl:Class>
  <owl:Class rdf:ID="Zone">
    <rdfs:subClassOf rdf:resource="#Location"/>
  </owl:Class>
  <owl:Class rdf:ID="Intention"/>
  <owl:Class rdf:ID="Coordinate">
    <rdfs:subClassOf rdf:resource="#Location"/>
  </owl:Class>
  <owl:Class rdf:ID="PrivacyAccess"/>
  <owl:Class rdf:ID="Task"/>
  <owl:Class rdf:ID="Device"/>
  <owl:Class rdf:ID="Application"/>
  <owl:Class rdf:ID="Agenda"/>
</rdf:RDF>
```

[그림 16] OWL DL 스키마로 표현된 일반적 상황모형

## 5. 결론

본 연구의 목적은 복수개의 유비쿼터스 서비스 구축을 위해 보다 일반화된 상황모형을 구축하기 위해 기존의 상황모형 유형들을 비교 평가하는 것이었다. 이를 위해 현재까지 제안된 상황모형에 대

해 포괄적으로 조사하였으며 조사한 상황모형의 특징을 알아보고 이들을 비교 평가하기 위하여 모형단순도와 모형표현도라는 두 가지의 평가 척도를 개발하였다. 연구 결과 온톨로지 기반 상황모형이 에이전트 시스템 구현에 가장 적절한 모형임을 확인하게 되었으며, 이에 따라 온톨로지 기반으로

국내의 한 재래시장의 유비쿼터스화를 위한 멀티 서비스 기반의 u-마켓 시스템에 실제적으로 사용될 일반적인 상황모형을 구축하였다. 개발된 OWL-DL 기반의 모형은 온톨로지 기반 추론기관의 검색 및 추론에 핵심으로 사용될 예정이다. 구축된 상황모형은 유비쿼터스 환경에서 상황개체로 정의될 만한 개체들을 기존의 문헌들을 종합하여 총괄적으로 반영한 것이다 (참조한 문헌별 상황 정보는 <부록 B> 참조). 이렇게 시스템 에이전트에 맞는 상황모형은 시스템의 오류를 낮추고, 성능을 향상시키는 결과를 가져올 것으로 기대된다.

## 참고문헌

- [1] Cheverst, K., Mitchell, K., and Davies, N, "Design of an object model for a context sensitive tourist GUIDE", *Computers and Graphics*, Vol.23(1999), 883~891.
- [2] Dogac, A., Laleci, G. B. and Kabak, Y. A, "Context Framework for Ambient Intelligence", *4th IEEE International Workshop on Distributed Auto-adaptive and Reconfigurable Systems*, Tokyo, 2004.
- [3] Gray P. and Salber D., "Modelling and using sensed context information in the design of interactive applications", *8th IFIP International Conference on Engineering for Human Computer Interaction*, Vol.2254 (2001), 317~336.
- [4] Gu, T. Wang, X. H. Pung, H. K. and Zhang, D. Q., "An ontology-based context model in intelligent environments", *Communication Networks and Distributed Systems Modeling and Simulation Conference*, (2004), 270~275.
- [5] Held, A., Buchholz, S. and Schill. A., "Modeling of context information for pervasive computing applications", *6th World Multi-conference on Systemics Cybernetics and Informatics (SCI)*, Orlando, 2002.
- [6] Henricksen, K., Indulska, J. and Rakotonirainy, A., "Modeling context information in pervasive computing systems", *1st International Conference on Pervasive Computing*, Vol.2414(2002), 167~180.
- [7] Henricksen, K., Indulska, J. and Rakotonirainy, A., "Generating context management infrastructure from high level context models", *Proceedings of the 4th International Conference on Mobile Data Management*, (2003), 1~6.
- [8] Henricksen, K. and Indulska, J., "A Software Engineering Framework for context-Aware Pervasive Computing", *2nd IEEE International Conference on Pervasive Computing and Communications (PerCom '04)*, (2004), 77~86.
- [9] Henricksen, K. and Indulska, J., "Modeling and Using Imperfect Context Information", *1st Workshop on Context Modeling and Reasoning (PerCom '04)*, (2004), 33~37.
- [10] Hofer, T., Schwinger, W., Pichler, M., Leonhartsberger, G., and Altmann, J. (2002). "Context-Awareness on Mobile Devices-the Hydrogen Approach", *36th Annual Hawaii International Conference on System Sciences (HICSS '03)*, (2003), 292a.
- [11] Indulska, J., Robinson, R., Rakotonirainy, A. and Henricksen, K. (2003). "Experiences in using cc/pp in context-aware systems", *4th International Conference on Mobile Data Management (MDM2003)*, (2003), 247~261.
- [12] Kaenampornpan, M., and O'Neill, E., "Modelling context: an activity theory approach", *2nd European Symposium on Ambient Intelligence*, (2004), 367~374.

- [13] Khedr, M. and Karmouch, A., "Negotiating Context Information in Context Aware Systems", *IEEE Intelligent Systems magazine*, Vol.19(2004), 21~29.
- [14] Kalian A., Gopalan S. and Sridhar V., "Hybrid Context Model based on Multilevel Situation Theory and Ontology for Contact Centers", <http://ieeexplore.ieee.org/iel5/9593/30313/01392789.pdf>
- [15] Öztürk, P., and Aamodt, A., "Towards a model of context for case-based diagnostic problem solving", *the interdisciplinary conference on modeling and using context*, (1997), 198~208.
- [16] Preuveneers, D., Bergh, J. V., Wagelaar, D., Georges, A., Rigole, P., Clerckx, T., Berbers, Y., Coninx, K., Jonckers, V. and Bosschere, K. D., "Towards an exensible context ontology for Ambient Intelligence", *Second European Symposium (EUSAI 2004)*, Eindhoven, (2004), 8~11.
- [17] Philippe D., Patrick G. And Alex V., "Integrating the Physical World with the Web to Enable Context-Enhanced Services", <http://hpl.hp.com/techreports/2003/HPL-2003-192.pdf>
- [18] Riva, O., "A conceptual model for Structuring Context-Aware Applications", *4th Berkeley-Helsinki Ph.D. Student Workshop on Telecommunication Software Architectures*. University of Berkeley, (2004), 21~24.
- [19] Samulowitz, M., Michahelles, F., and Linnhoff-Popien, C., "Capeus: An architecture for context-aware selection and execution of services", *New developments in distributed applications and interoperable systems*, Kluwer Academic Publishers, (2001), 23~39.
- [20] Schmidt, A., Aidoo, K. A., Takaluoma, A., Tuomela, U., Laerhoven, K. V. and Velde, W. V., "Advanced interaction in context", *1st International Symposium on Handheld and Ubiquitous Computing (HUC'99)*, Springer-Verlag, (1999), 89~101.
- [21] Schmidt, A., Beigl, M. and Gellersen, H. W., "There is more to Context than Location", *Computers & Graphics Journal*, Vol.23 (1998), 893~901.
- [22] Strang, T., Linnhoff-Popien, C., Frank, K., Stefani, J. B., Demeure, I. and Hagimont, D., "CoOL: A Context Ontology Language to enable Contextual Interoperability", *IFIP International Federation for Information Processing (DAIS 2003)*, (2003), 23~247.
- [23] Wang, X. H., Zhang, D., Gu, T. and Pung, H., "Ontology based context modeling and reasoning using OWL", *Context Modeling and Reasoning at 2nd IEEE International Conference on Pervasive Computing and Communication (PerCom '04)*, (2004), 18~22.
- [24] MIDAS, <http://web.mit.edu/zacka/www/midas.html>



### 부록 A. 새롭게 제안한 일반적 상황모형의 각 요소 정의

Context Entities	Context Definition
User(identity)	= *user or person* = heartRate + bodyHeat + bloodPressure + homePhoneNo. + job + emailAddress + age + preference + name + medicalHistory + nearbyPeople + gender + height + weight + userID + homeAddress + officeAddress + cellPhoneNo
Environment	= *사용자 주변환경 요소* = humidity + temperature + sound + light
PrivacyAccess	= *시스템이 접근할 수 있는 사용자의 프라이버시* = privacyInformation + privacyLevel
Agenda	= *사용자의 일정* = agendaTitle
Time	= *시간* = duration + futureTime + currentTime + pastTime
Service	= *사용자에게 제공되는 서비스* = serviceID + serviceName + serviceContent
Application	= *서비스를 제공하기 위한 응용 프로그램 (소프트웨어) * = applicationID + applicationName + applicationContent
CompEntity	= *일정한 구역 내에 존재하는 유비쿼터스 센서 네트워크 인프라* = Network + Resources + Sensors Network= networkID + bandwidth + delayTime + connectivity Resources= resourceID + batteryRate + memoryCapacity + CPU Sensors= sensorID + sensorName + sensorUse
Device	= *사용자가 가지고 있는 장치* = deviceID + deviceProfile + nearbyDeviceID
Location	= *위치* = Zone + Coordinate Coordinate = longitude + latitude
Zone	Zone = rule + role + zoneName + zoneID
Intention	= *사용자의 의도* = intentionTitle
Task	= *사용자가 현재 하는 일 또는 미래에 할 일* = taskTitle + taskTime
Activity	= *사용자의 행위* = dynamicState + staticState + direction + speed + motion + behavior

부록 B. 상황개체 집합표

	Comp Entity	User	Task	Activity	Environment	Location	Intention	Device	Application	Time	Service	Agenda	Privacy Access	Role	Rule
[Ozturk, P. et al. '97]		○			○	○									
[Schmidt, A. et al. '99]	○	○		○	○	○		○							
[Gray, P. D. et al. '01]		○				○		○		○					
[Samulowitz, M. et al. '01]	○	○	○					○			○				
[Held, A. et al. '02]	○	○				○		○							
[Strang, T. et al. '03]		○			○			○							
[Henricksen, K. et al. '03]		○		○		○		○							
[Hofer, T. et al. '03]	○	○				○				○					
[Gu, T. et al. '04]	○	○		○		○									
[Riva, O. et al. '04]	○	○		○	○	○		○				○			
[Dogac, A. et al. '04]	○	○		○	○	○		○		○		○			
[Khedr, M. et al. '04]	○	○		○		○		○	○					○	
[Wang, X. H. et al. '04]	○	○		○		○									
[Preuveneers, D. et al. '04]	○	○			○				○						
[Kaenampompan, M. et al. '04]	○	○	○	○	○	○	○	○	○	○		○		○	○
[Kalyan, A. et al. '05]		○		○	○	○	○	○		○	○	○			
MIDAS		○			○	○		○	○	○					

## Abstract

# A Context Model Comparison Methodology for Developing Generic Context Model used in Ubiquitous Multi-Services

Park Tae Hwan\* · Kwon Ohbyung\*

Acquiring context data in a timely and correct way is now regarded as one of the crucial characteristics of the proactive service which runs on ubiquitous computing environment. Moreover, context model should be well designed to provide a solid context-aware system. Since the ubiquitous computing systems aim to provide context-aware services everywhere with any available devices, legacy services which uses context models assuming single or limited domain should be extended enough to be useful even for multi-domain multi-services. This leads us to a motivation to build a generic context model with an appropriate type of model. Hence, the purpose of this paper is to propose a generic context model by assessing a variety of model types with a sort of evaluation measures.

**Key words** : Ubiquitous Environment, Context Model, Context Aware System, Ontology

---

\* School of International Management, Kyunghee University