

UCI Ad-hoc 2006-UCI-02/2006. 12

유비쿼터스 환경에서 ID 체계 연구

한국정보사회진흥원

본 이슈페이퍼는
유비쿼터스 환경에서 ID 체계 연구를 통한
향후 국제 표준 URN과의 연계방안 모색하기
위해 작성된 발간물입니다.

본 자료의 내용은 필자 개인의 의견으로
본원의 공식적인 견해가 아니며, 논문 내용과
관련된 문의나 건의사항은
한국정보사회진흥원 u-서비스지원단으로
연락주시기 바랍니다.

TEL : (02) 2131-0449

FAX : (02) 2131-0309

유비쿼터스 환경에서 ID 체계 연구



이창열(동의대학교), 이재진(DPC), 진승헌(ETRI)

□ 차 례 □

1. 개론	1
2. 유비쿼터스 세계 분석	2
2.1 개념	2
2.2 전통적 세계와 비교	3
2.3 유비쿼터스 정보의 계층 구조	3
2.4 사례 분석	4
2.4.1 일상 생활 1	4
2.4.2 일상 생활 2	5
2.4.3 도시 관리 세계	6
2.4.4 SCM 1	6
2.4.5 SCM 2	7
2.5 기존 모델	8
2.5.1 PML	8
2.5.2 EPCIS	9

3. 유비쿼터스 정보 체계	10
3.1 유비쿼터스 서비스	10
3.1.1 구조	10
3.1.2 유비쿼터스 정보 센터	11
3.2 공통 정보 체계	12
3.3 유비쿼터스 정보 기술 언어	13
3.3.1 상위 구조	13
3.3.2 상세 설명	14
3.3.3 XML 표현	23
3.4 응용 서비스	24
3.4.1 정의	24
3.4.2 서비스	24
3.4.3 사례	25
4. UCI에 적용	29
4.1 UCI와 관계 정의	29
4.2 UCI의 확장 가능성	30
5. 결론	31
참고문헌	32
부록. UCML 소스	33

□ 표 차례 □

<표 2-1> 전통적 시스템과 유비쿼터스 시스템의 패러다임3

<표 4-1> UCID와 UCI 비교29

□ 그림 차례 □

(그림 2-1) 유비쿼터스 세계의 시스템 구조	2
(그림 2-2) 유비쿼터스 세계에서 정보의 계층 구조	4
(그림 2-3) PML 스키마 구조	8
(그림 2-4) EPCIS Events 구조	9
(그림 3-1) U-City 서비스 구조	10
(그림 3-2) UISC 정보 관리 체계	11
(그림 3-3) U-City Markup Lanaguage 구조	13
(그림 3-4) ECEF 기반 위치 표현	17
(그림 3-5) UCML의 구조	23

1. 개요

유비쿼터스란 용어는 1988년 Xerox PARC(Palo Alto Research Center)의 Mark Weiser에 의해 IT 분야에 처음 사용되었다. 이러한 유비쿼터스 기술이 한정된 영역에서 쓰이고 있다가, 21세기 들어오면서 IT 인프라가 보편화되고 IT 기술이 일상에 쉽게 적용되면서 실제적인 유비쿼터스 환경으로 진입이 시작되었고 우리나라에서 2004년 Ubiquitous IT Korea Forum을 만들고 미래 성장 동력으로써 u-Korea 추진 전략을 수립하였다. 유비쿼터스 기술을 기반으로 하는 미래의 도래는 u-IT839에서 언급한 8대 서비스와, 3대 인프라, 그리고 9대 성장 엔진을 갖추면 실현될 수 있는 단순한 문제인가? 그것을 살펴보기 위하여 유비쿼터스라는 개념을 다시 한번 살펴보기로 한다.

Mark Weiser에 의하면 유비쿼터스란 다음과 같이 정의할 수 있다 :

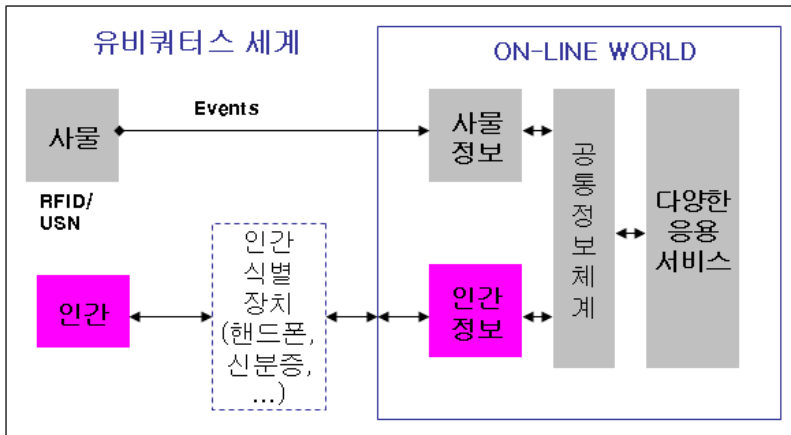
- 언제 : 언제나 네트워크에 연계되고
- 어디서나 : 어디서나 네트워크에 연계되고
- 누구나 : 누구나 네트워크에 접근할 수 있고
- 어떤 장치로든 : 다양한 장치를 이용하여 접근
- 어떤 서비스를 : 원하는 어떤 서비스를 받을 수 있는 체계를 갖추고 있다.

위와 같은 환경이 갖추어지지 위하여 U-Korea 전략에서 기반이 되는 u-IT839 기술은 필수적이다. 이러한 기반 기술에 의하여 갖추어진 유비쿼터스 환경이란 "What you act(are acted) is what your system" 즉 일상 생활이 시스템적으로 기술될 수 있는 체계가 필요한 것이다. 행동이 시스템적으로 표현되려면 행동을 지원하는 시스템이 기술되어야 하고, 이는 모든 사물에 센서가 부착되고, 의미적 정보가 연계되어서 가능한 것이며, 이러한 세계는 인간과 사물이 공존하는 유비쿼터스 세계인 것이다.

2. 유비쿼터스 세계 분석

2.1. 개념

유비쿼터스 세계는 현실 세계의 사물과 인간이 온라인에서 식별과 사건화된 정보를 가지고 접근하며, 다양한 응용 세계와 연계가 가능한 Seamless 공통 정보 체계(사물정보-ID/Events, 인간정보-ID/Profiles, Event Description Lanaguage)에 기반을 둔다. (그림 2-1)처럼 사물과 인간 정보가 공통정보체계를 통하여 다양한 응용에 표현되는 것이다.



(그림 2-1) 유비쿼터스 세계의 시스템 구조

즉 유비쿼터스 세계는 인간의 수단(핸드폰, 신분증, Digital ID 등)과 사물의 수단(RFID, Sensor, DNS, ...)을 기반으로 표현된 정보가 다양한 응용 서비스에서 활용되는 체계인 것이다.

2.2. 전통적 세계와 비교

유비쿼터스 구조가 기존 체계에는 없었던 것인가? 그렇지 않은 것이다. 구조는 있을 수 있지만 단지 인간과 사물에 대한 표현 수단이 효과적이지 못하여서 이를 기반으로 표현되는 응용 서비스 또한 효과적이지 못했기 때문에 특별한 의미를 부여하지 못했을 것이다. 현 세계는 인간과 사물에 대한 표현(예를 들어 인간은 전화번호, 휴대폰번호, 버스카드 번호, 계좌번호, 주민등록번호 등, 사물은 RFID와 같은 수단을 활용한 정보 표현)이 효과적이어서 이를 기반으로 하는 응용이 유용하기 때문에 이러한 응용이 발달된 세계가 유비쿼터스 세계일 것이다.

전통적 세계와 유비쿼터스 세계의 차이는 <표 2-1>과 같은 시각적 차이를 가지고 있다.

<표 2-1> 전통적 시스템과 유비쿼터스 시스템의 패러다임 비교

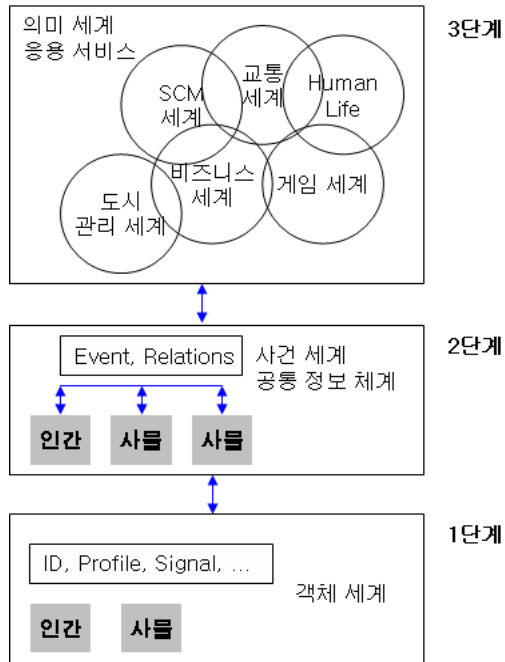
항목	전통적 세계	유비쿼터스 세계
정보 대상	사물 정보	인간 행동, 사물 정보
정보 범위	사물에 대한 사건	인간 생활 및 사물의 사건
사물 정보	인간에 의한 수동적	사물 자체에 의한 능동적
상호 연계	응용이 개별적	응용이 상호 연계

2.3. 유비쿼터스 정보의 계층 구조

유비쿼터스 세계의 정보는 (그림 2-2)와 같이 3단계 계층으로 표현할 수 있다 :

- 객체 세계 : 센서가 부착된 인간과 사물로 구성된 세계. Primitives
- 사건 세계 : 인간 수단(핸드폰, 신분증, ...)과 사물의 수단(RFID, Sensor, DNS, ...)이 시간과 장소에 의해 표현된 세계. Events
- 의미 세계 : 사건 세계의 정보를 이용하는 세계. “The view to the event world” .

객체 세계 정보는 → 시간과 공간이 추가된 사건 세계로 → 그리고 적용된 분야가 존재하는 의미 세계로 표현된다. 이들 3단계 계층에 대한 이해를 쉽게 하기 위하여 시나리오를 구성하여 해석해 보기로 한다.



(그림 2-2) 유비쿼터스 세계에서 정보의 계층 구조

2.4. 사례 분석

2.4.1. 일상생활 1(Top-Down 분석)

- 의미 세계 : “홍길동은 송파동 피자헛에 패밀리타입 피자를 10:00 시에 주문하였다”
- 사건 세계 : “주민등록번호 XX(홍길동)는 기관ID YY.송파(송파동 피

자켓)에 UNSPSC 분류로 50349683(피자)이며 ItemID가 ZZ(패밀리 타입 피자)를 1개에 대한 OrderID KK(주문)를 10:10에 발생시켰다” (전체 Business ID MM).

- 객체 세계 :
 - * 인간 → 주민등록번호
 - * 기관 → 기관 ID : GLN, ...
 - * 피자 → EPC, GTIN, UNSPSC, ...
 - * 주문 → OrderID
 - * 시간 → ISO
 - * 기타 : BizStep, Transaction ID

2.4.2. 일상생활 2(Bottom-Up 분석)

- 객체 세계 :
 - * 택시ID → RFID : 12356
 - * 택시기관 → 154(GLN)
 - * 위치 → GPS, 정류장 번호
 - * 인간 → 010-294-593 (mobile phone number)
 - * 시간 → ISO
- 사건 세계 : 2006-08-30:10:10:10+GMT1에 홍길동(010-294-593)이 택시 12356에서 신고됨.
- 의미 세계 : 홍길동이 종로 3가에서 택시를 2006-08-30:10:10:10에 탔다.

2.4.3. 도시 관리 세계(Bottom-Up 분석)

- 객체 세계 :
 - * 다리 → 시설 ID
 - * 교각 →
 - : 진동 센서 → (시간, 진동)+
 - : 환경 센서 → (시간, 오염도)+
 - * Sink Node
 - : 진동 경고 수위
 - : 오염 경고 수위
- 사건 세계 : 시설 ID가 567(성수대교)의 ID 459의 진동센서(3번째 교각에 설치)에서 진동 5cm가 30초마다 생성되고 있다.
- 의미 세계 : 성수대교가 붕괴 위험에 있다.

2.4.4. SCM 1(Bottom-Up 분석)

- 객체 세계 :
 - * 기관 : 123(GLN)
 - * 창고(DC) : 123.5(GLN)
 - * 날짜 : 2006-08-30:10:11:14+GMT1
 - * 창고 Gate 4 : 1345(GIAI : RFID Reader ID)
 - * 제품 : EPC, ISO15459, ...
 - * Transaction : Order ID, ASN

- 사건 세계 : 오전 10:01분 DC 9번 문에 입고되는 물품 리스트가 관찰되었다
- 의미 세계 : 주문서대로 통보 물품이 창고 9번 문에 도착하였다.

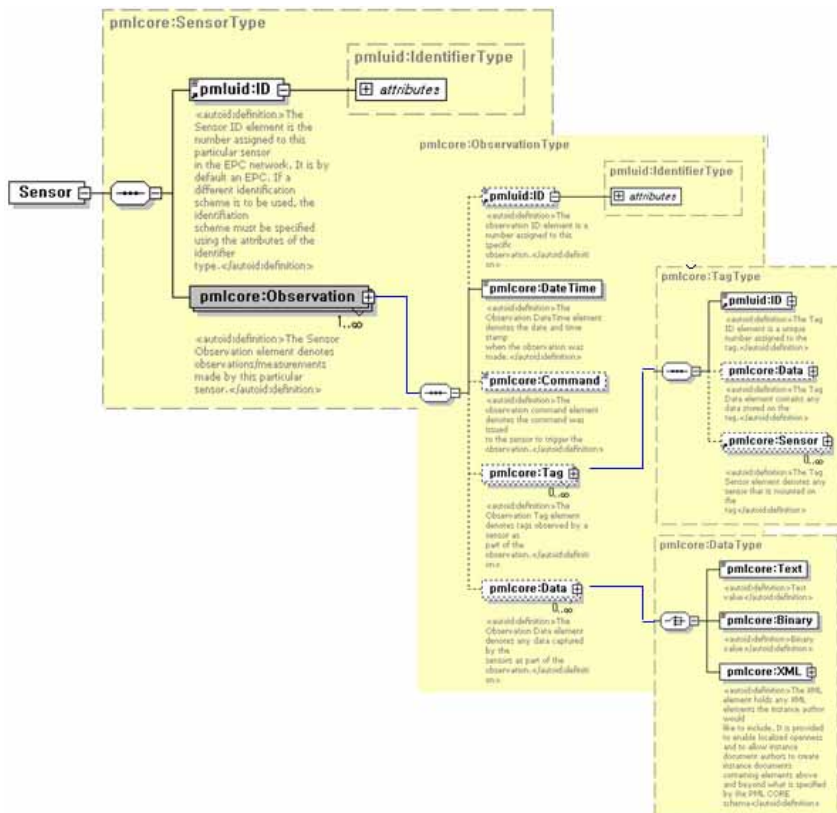
2.4.5. SCM 2(Top-Down 분석)

- 의미 세계 : 회사 보유 코카콜라 재고 정보를 확인하시오
- 사건 세계 : 2006년 8월 30일 10시 XX회사(GLN 1234)의 1번에서 5번 창고(sub-GLN 1 ~ 5)에서 코카콜라(EPC : 24.45.68.XX) 태그 확인
- 객체 세계 :
 - * 기관 : GLN(123)
 - * 창고(DC) : 123.1, 123.2, 123.3, 123.4, 123.5
 - * 날짜 : 2006-08-30:10:11:14+GMT1
 - * 코카콜라 : EPC 23.45.68.XXX
 - * Transaction : Transaction ID
 - * Business Step : Stocking

2.5. 기존 모델

2.5.1. PML

PML(Physical Markup Language)의 목적은 리더(센서)로 수집된 데이터를 표준 구조로 표현하는 것으로 현재 더 이상 사용되지 않지만, 간단한 객체 세계와 사건 세계를 표현하는데 적합한 표현 구조를 가지고 있다. PML의 목적은 리더(센서)로 수집된 데이터를 표준 구조로 표현하는 것이다.

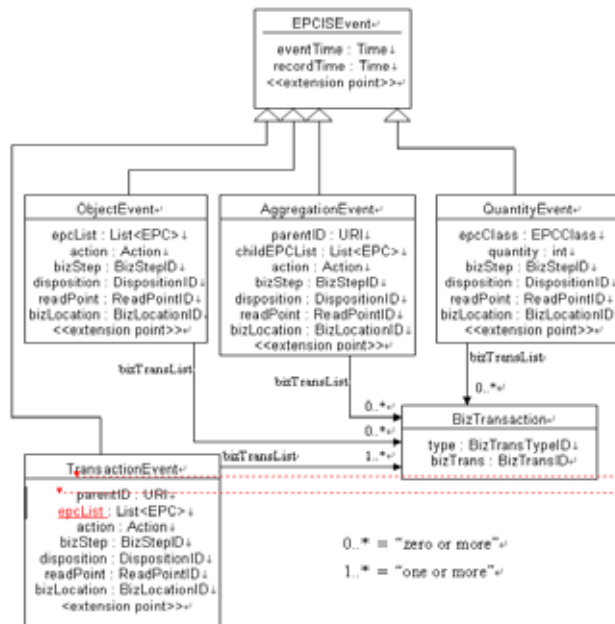


(그림 2-3) PML 스키마 구조

즉 센서를 통하여 읽은 정보를 표현하는 것으로 어떤 센서가(센서 ID로 표현) 다량의 읽은 정보를 표현한다.

2.5.2. EPCIS

EPCIS(EPC Information Service)는 사건 세계와 의미 세계를 표현하는 언어이다. 표현된 사건을 SCM(Supply Chain Management)이라는 응용 분야에서 해석하기 때문에 응용 분야에 적합한 구조를 가진 사건 표현 언어로 다양한 용도에 일반적으로 사용하기에는 문제점을 가지고 있다. (그림 2-4)는 EPCIS를 사용하여 표현하는 정보 구조를 나타낸 것으로 응용 분야에 적합한 사건 표현 언어로 구성되었다. 즉 모든 사건은 4개 차원(What, When, Where, Why)로 표현되며, 사건은 객체 형태에 따라(ObjectEvent AggregationEvent, 그리고 QuantityEvent)로 표현하며, 비즈니스 관련 TransactionEvent로 구성된다.



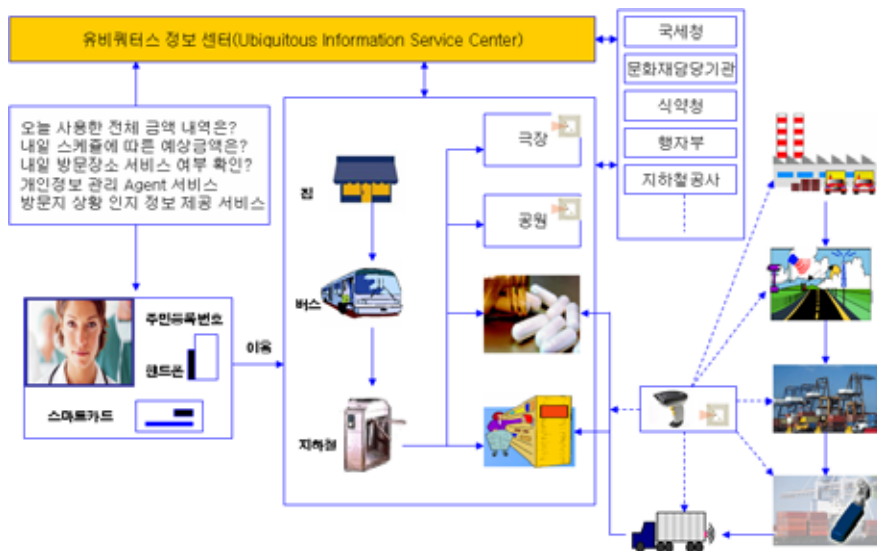
(그림 2-4) EPCIS Events 구조

3. 유비쿼터스 정보 체계

3.1. 유비쿼터스 서비스

3.1.1. 구조

EPCIS처럼 특정 분야에 의존적이지 않으면서 일반성 있게 사용할 수 있는 사건 표현 언어를 개발하기 위하여 우선 의미 세계의 일반적 구조를 살펴볼 필요가 있다. (그림 3-1)은 일반적 응용을 표현한 것으로, SCM 부문, 인간의 Life Cycle, 도시 관제 부분 등을 포함하는 U-City의 서비스 구조를 표현한 것이다.



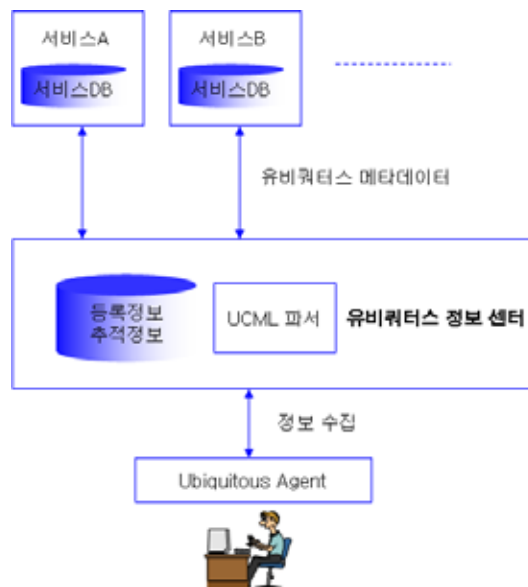
(그림 3-1) U-City 서비스 구조

(그림 3-1)의 가장 큰 특징은 다양한 분리된 응용이 하나의 서비스로

연계되기 위하여 유비쿼터스 정보 센터(Ubiquitous Information Service Center; UISC)가 필요하다는 것이다. UISC의 역할에 대하여 좀더 자세히 살펴보기로 한다.

3.1.2. 유비쿼터스 정보 센터

유비쿼터스 정보 센터에서 서비스하는 구조는 (그림 3-2)처럼 다양한 서비스로부터 사건 정보를 수집하고, 필요시 서비스하는 구조로 되어 있다. 다양한 응용 서비스는 서로 다른 체계를 가지고 있기 때문에 연계하고, 서비스하는 것에 문제가 있을 수 있다. 그러므로 다양한 응용 서비스가 공통적으로 표현될 수 있는 **핵심 공통 정보 체계**가 존재하여야 한다.



(그림 3-2) UISC 정보 관리 체계

3.2. 공통 정보 체계

공통 정보 체계는 UCID(Ubiquitous City Identifier)를 기반으로 이루어지며 이는 의미 세계에 존재하는 다양한 서비스를 유비쿼터스 정보 체계와 연계하는 수단으로 존재한다. 다양한 서비스는 유비쿼터스 정보 체계에 해당 서비스를 추적할 수 있는 U-City 메타데이터를 제공하여 공통정보체계는 해당 서비스를 추적할 수 있는 정보를 간직하게 된다. 본 정보를 기반으로 유비쿼터스 사회에서 발생하는 모든 사건에 대하여 필요시 필요한 서비스를 할 수 있어야 하며 다음과 같은 메타데이터를 포함하여야 한다 :

- service Namespace
- service ID
- service Location
- service Date

이와 같은 체계를 포함하는 U-City 표현 언어를 살펴보기로 한다.

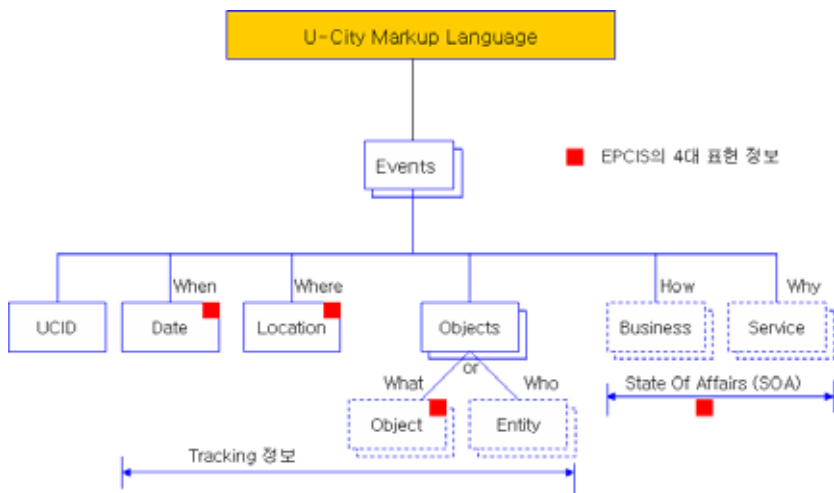
3.3. 유비쿼터스 정보 기술 언어

3.3.1. 상위 구조

유비쿼터스 세계의 Event 구조는 다음과 같은 원칙을 가지고 설계하였다 :

- 6하 원칙(5W 1H) 구조 표현
- EPCIS 정보 구조 표현
- 추적 정보 표현
- U-City의 사건 표현(Entities 추가)

그리고 모든 정보의 최소 단위는 (그림 3-3)에 기술된 것 처럼 객체에 대한 동일한 개념의 시간/장소가 포함되는 정보가 U-City 표현 언어의 최소단위인 것이다.



(그림 3-3) U-City Markup Lanaguage 구조

3.3.2. 상세 설명

3.3.2.1. UCID

UCID(U-City ID)는 공통정보체계를 표현하는 ID로 다양한 서비스에 사용될 수 있어야 한다. 그러므로 UCID는 다음과 같은 구조를 가진다.

○ Namespace 등록

- 기관 확인
- local namespace 할당
- resolution 선택
 - * URN 변환 또는 WSDL 사용
 - * resolution 방법 선택 사항은 추후 좀 더 자세히 알아보기로 하며 본 자료에서는 추가 사항은 생략한다.
- sub-Namespace 사용 가능

○ UCID 사용

- urn:ucid:namespace(.subnamespace):LocalID

○ resolution 선택

- 웹 서비스
- URN resolution

예를 들어 다음은 UCID의 사례이다 : urn:ucid:subway.line7:123.456.

본 사례는 지하철 7호선에서 발생한 사건에 대한 ID를 표현하고 있다.

3.3.2.2. Date

날짜 표현은 YYYY-MM-DDThh:mm:ssTZD 형태로 표시된다. 예를 들어

- 일반 표현 : 2005-09-20T10:20Z
- 서머 타임 : 2005-09-20T10:20+01:00
- 서머 타임 : 2005-09-20T10:20-01:00

위와 같이 표현되며, 생략되는 정보로

- 2005-10-29
- 2005-10
- 2005-10-29T10
- 2005-10-29T10:20
- 2005-10-29T10:20:32
- 2005-10-20T10:23:33Z

가 전부 가능하다.

또한 인간이 표현하는 symbol로 “제2차대전” 형태를 표시할 수도 있다.

○ 사건적 표현

```
<Date>  
  <dateType>symbol</dateType>  
  // symbol, start, end, record, generated //  
  <value>RFID시범사업기간</value>  
</Time>
```

○ 범위 표현

```
<Date>
  <dateType>from</dateType>
  <value>2006:10:11</value>
</Date>
<Date>
  <dateType>to</dateType>
  <value>2006:11:11</value>
</Date>
```

3.3.2.3. Location

Location은 “한국정보사회진흥원” 과 같은 이름, 또는 사업자 번호, GLN, GPS와 같은 표현 등으로 표현할 수 있다. 그러므로 Location 표현은 다양한 표현을 수용하기 위하여 다음과 같이 기술되어야 한다.

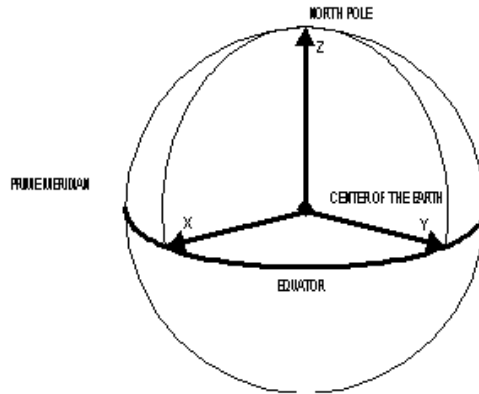
○ 위치 표현

```
<Location>
  <locType>symbol</locType>
  <value>한국정보사회진흥원</value>
</Location>
```

○ sub-GLN 사용

```
<Location>
  <locType>GLN</locType>
  <value>10.정보관.901</value>
</Location>
```

○ 공간 표현(Global Position)



(그림 3-4) ECEF 기반 위치 표현

GPS에서 사용하는 표현은 지구 중심을 기준으로 좌표를 표시하는 카테시안 시스템 (Earth-Centered, Earth Fixed Cartesian Coordinate(ECEF))으로 (그림 3-4)와 같이 x , y 그리고 z 에 의해 표현된다. 단위는 미터(m)이다. x 는 본초 자오선을 통과한 축으로 지구 중심에서 거리를, y 는 xz 판에 직각인 적도를 통과하는 축으로 지구 중심에서 거리를, 그리고 z 는 북극을 통과하는 것으로 지구 중심에서 거리를 나타낸다. 이 값은 쉽게 사용하는 고도, 위도, 경도를 사용하는 Universal Transverse Mercator(UTM), Military Grid Reference System(MGRS), World Geographic Reference System(GEOREP) 그리고 Universal Polar Stereographic(UPS)와 National Grid System을 포함하는 지역 체계로 쉽게 변환될 수 있다.

○ 이동 위치 표현

```
<Location>
  <locType>ECEF</locType>
  // symbol, GLN, ECEF, 사업자등록번호, ....., 주소, ... //
  <direction>from</direction>
  // from, to, north, east, west, south, fixed, ... //
  <value type = "x">-742507.1</value>
  <value type = "y">-5462738.5</value>
  <value type = "z">3196706.5</value>
</Location>
```

3.3.2.4. Objects

객체(Objects)는 사람(Entity)과 사물(Object)로 구성된 것으로 2가지 형태 중에 1개 이상은 반드시 표현되어야 한다. 표사람을 제외한 대상으로 직접 표현과 ID 표현이 가능하다. 사건 세계에서 객체와 사람은 반드시 1번 이상 표현되어야 한다. 객체는 상황에 따라 데이터(예를 들어 센서) 정보를 포함할 수 있다.

가. Object

○ 최대 표현

```
<Objects>
<Object>
  <parentID> // 선택   EPCIS의 aggregation model
  <objectType>EPC</objectType>
  <value>urn:epc:10.124.459</value>
```



```

</parentID>
<childID> // 선택
  <objectType>EPC</objectType>
  <value>urn:epc:10.124.460</value>
  <data>49504</data> // 선택
</childID>
<childID> // 선택
  <objectType>EPC</objectType>
  <value>urn:epc:10.124.461</value>
</childID>
<Content> // 선택 EPCIS의 objective model
  <objectType>EPC</objectType>
  <value>urn:epc:10.124.461</value>
</Content>
<Content> // 선택 - 반복 EPCIS의 quantity model
  <objectType>symbol</objectType>
  <value>라면 1박스</value>
</Content>
</Object>
</Objects>

```

나. Entity

Entity는 사람이거나 기관이 될 수 있다. 응용에 따라서 다양한 형태의 표현이 가능하다. 즉 직접적 정보로 주민등록번호, 신분증 번호 등을 가질 수 있고 그리고 보조 정보로 주소, 지위, 연락처 등이 표현될 수 있다.

응용에 따라 다양하게 표현할 수 있기 때문에 상세 표현은 생략한다.

○ 직접표현

```
<Objects>
  <Entity>
    <entityType>symbol</entityType>
    // symbol, 주민등록번호, DigitalIdentity, //
    <value >홍길동</value>
    <others>연락처:234-4964</others>
  </Entity>
</Objects>
```

○ 사람

```
<Objects>
  <Entity>
    <entityType>주민등록번호</entityType>
    <value>561209-1277718</value>
  </Entity>
</Objects>
```

○ Digital ID 사용

```
<Objects>
  <Entity>
    <entityType>DigitalIdentity</entityType>
    <value>idp:ms:1234</value>
  </Entity>
</Objects>
```

3.3.2.5. Business

상태는 현 사건이 발생한 상황에 대한 내부적 비즈니스 상태, 프로세스 단계 등에 대하여 기술한다. 예를 들어 창고인 경우 “Stocking”, "Keeping", "Loading"과 같은 상태로 정의될 수 있다. 본 상태의 일반적 표현은 생략한다. 왜냐하면 다양한 응용에 다양한 상태가 존재하기 때문이다.

○ 직접표현

```
<Business>  
  <businessType>symbol</businessType>  
  <value>집에 가는 중</value>  
</Business>
```

○ WMS

```
<Business>  
  <businessType>wms</businessType>  
  <value>stocking</value>  
  <others>step 4</others>  
</Business>
```

○ Process

```
<Business>  
  <businessType>process</businessType>  
  <value>step 2</value>  
</Business>
```

3.3.2.6. Service

Service는 현 사건이 발생한 상황에 대한 외부적 비즈니스 상태에 대하여 기술한다.

Service와 Business 사이의 명확한 구분은 응용에 따라 다를 수 있지만 기본적으로 Business는 기관 내부 단계, Service는 기관 외부 단계나 서비스를 적으면 된다.

○ 주문

```
<Service>
  <serviceType>invoiceID</serviceType>
  <value>123489</value>
  <description>피자 주문</description>
</Service>
```

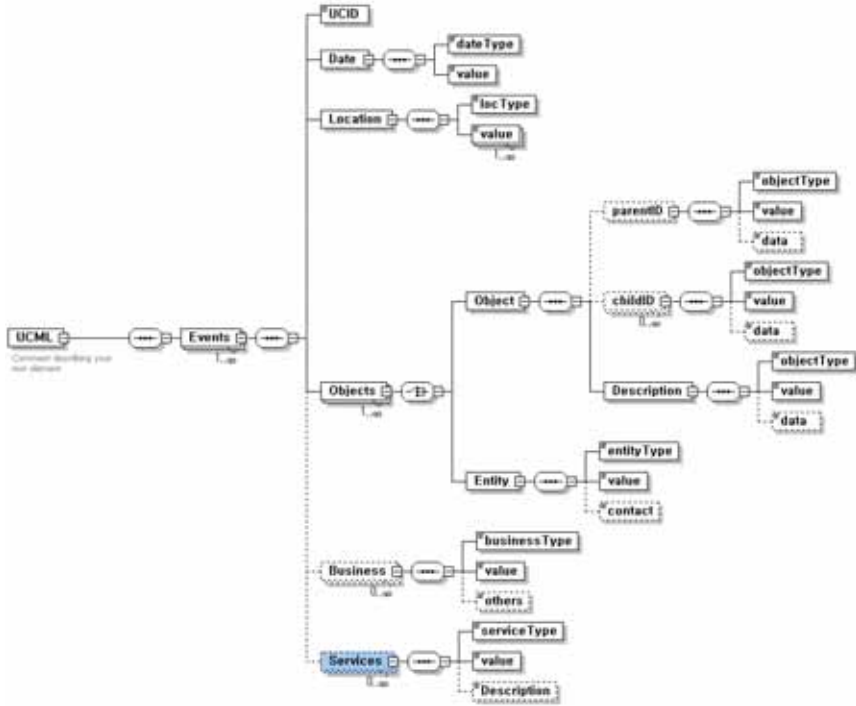
○ Process

```
<Service>
  <serviceType>transactionID</serviceType>
  <value>1ff9</value>
</Service>
```

○ 기호

```
<Service>
  <serviceType>symbol</serviceType>
  <value>부장 명령</value>
  <description>출장 사항</description>
</Service>
```

3.3.3. XML 표현



(그림 3-5) UCML의 구조

3.4. 응용 서비스

3.4.1. 정의

(그림 3-1)에서 이미 다양한 응용 서비스에 대한 구조를 기술한 바 있다. 다양한 응용 서비스가 UCML에 의해서 표현이 된다면 비록 응용이 달라도 공통정보체계를 가짐으로써 유비쿼터스 세계에 대한 다양한 응용이 가능한 것이다. 예를 들어 (그림 3-1)에서 언급한 다음 서비스가 있다 :

- 오늘 사용한 전체 금액 내역은?
- 내일 스케줄에 따른 예상 금액은?
- 내일 방문장소 서비스 여부 확인?
- 개인정보관리 Agent 서비스
- 방문지 상황인지 정보 제공 서비스

이들 서비스는 비록 다른 응용에서 사용되는 것 일지라도 공통정보체계를 가짐으로써 서비스가 가능한 것이다.

3.4.2. 서비스

내부적으로 볼 때, 다양한 응용이 연계되려면 기술적 표준 모델 뿐만 아니라 정보를 관리하는 물리적 기구도 필요한 것이다. 이러한 서비스를 위하여 유비쿼터스정보센터는 다양한 서비스(극장 관리, 공원관리, 약국 판매, 지하철 관리, 쇼핑몰 관리 등)이 서로 연계되어 서비스될 수 있게 응용 사이에 동일한 ID 체계에 기반한 Loosely Coupled 모델에 기반한 서비스를 제공하여야 한다.

3.4.3. 사례

3.4.3.1. 서비스 사례

○ 의미 세계 표현

- 표현 : 오늘 사용한 전체 금액은?
- 설명 : 내가 오늘 탄 지하철 비용, 신용카드로 구입한 금액, 핸드폰 사용 정보

○ 객체 세계

- 본인 확인 : Smart Card 번호, 신용카드 번호, 핸드폰 번호

○ 사건 세계 표현

- 시공간 표현 : 시간에 대하여 다양한 응용 공간에 질의하는 형식으로 표현
- 오늘 : 2006-10-10
- 표현

```
<UCML>
<!-- Events 1 -->
<Events>
  <UCID>urn:ucid:subwaytotal:123.456</UCID>
  <Date>
    <dateType>recorded</dateType>
    <value>2006-10-10</value>
  </Date>
  <Location>
    <locType>symbol</locType>
    <value>삼성역</value>
```

```

</Location>
<Objects>
  <Object>
    <content>
      <objectType>smart_card</objectType>
      <value>124355</value>
    </content>
  </Object>
</Objects>
<Service>
  <serviceType>Price</serviceType>
  <value>800</value>
  <description>지하철 요금</description>
</Service>
</Events>
<!-- Events 2 -->
<Events>
  <UCID>urn:ucid:emart:123.6888</UCID>
  <Date>
    <dateType>recorded</dateType>
    <value>2006-10-10</value>
  </Date>
  <Location>
    <locType>symbol</locType>
    <value>이마트 서울 지점</value>
  </Location>
</Objects>

```



```

    <Object>
      <content>
        <objectType>visa</objectType>
        <value>55455</value>
      </content>
    </Object>
  </Objects>
  <Service>
    <serviceType>Price</serviceType>
    <value>12,000</value>
    <description>구매 요금</description>
  </Service>
</Events>

<!-- Events 3 -->
  <Events>
    <UCID>urn:ucid:skt:439053154.435</UCID>
    <Date>
      <dateType>recorded</dateType>
      <value>2006-10-10</value>
    </Date>
    <Location>
      <locType>symbol</locType>
      <value>한국</value>
    </Location>
  </Objects>
  <Object>

```

```
<content>
  <objectType>SKT</objectType>
  <value>010-2094-3945</value>
</content>
</Object>
</Objects>
<Service>
  <serviceType>Price</serviceType>
  <value>1,000</value>
  <description>통화 요금</description>
</Service>
</Events>
</UCML>
```

○ 유비쿼터스 정보 센터

공통정보체계를 기반으로 구축되는 Events 자료를 수집하고 있으며, 필요시 이들 정보를 서비스한다.

4. UCI에 적용

4.1. UCI와 관계 정의

UCI(Universal Content Identifier)는 URN 기반 식별자이다. UCI가 기존에 디지털 콘텐츠에 적용되었지만, 국가적 ID로 활용되면서 유비쿼터스 사회에 적합한 형태로 개념이 확장될 수 있으며, 그러한 가능성으로써 유비쿼터스 사회의 ID를 살펴보았다. 본 연구에서 제시한 UCID와 UCI의 개념적 차이는 다음 표와 같다.

<표 4-1> UCID와 UCI 비교

항목	UCID	현재 UCI
ID 구조	URN	URN
Resolution	Web Service/ DDDS 선택	DDDS 방식
연계 정도	Loosely Coupled	Tightly Coupled
정보 표준	UCML	식별메타데이터
ID 부여 체계	ID 부여 모듈 배포	ID 부여 모듈
정보 교류	협조적 차원 / 필요시 제공안해도 됨	의무적 차원
적용 분야	Everything	디지털 콘텐츠
특정분야 기존 ID	사용 가능	사용 가능
국제적 유일성	Namespace 사용/ 등록해야 함	Namespace 사용/ 등록 했음
관리 기관	아직 개념 연구임	지정됨

위 표와 같이 현재의 UCI는 UCID와 동일한 ID이지만, 바로 확대하기에는 문제가 있다. 대표적인 것으로 변환(Resolution) 방식이 다를 수 있으며, Web Service를 사용하는 기관은 정보 제공의 의무도 없는 필요시 선택적으로 줄 수 있는 느슨한 연계 구조이며, 각 기관의 독립성이 한층 강조된 모델이다.

4.2. UCI의 확장 가능성

UCI를 유비쿼터스 사회의 ID 체계로 발전하기 위하여 개념 모델링과 정보 서비스 체계 등에 전면 개편이 이루어져야 한다.

다양한 형태의 기존 ID 체계를 준수하면서, 다양한 응용에 사용될 수 있는 연계 개념을 가진 ID 체계로써, 그리고 <그림 3-1>에서 처럼 유비쿼터스 사회의 중심으로 ID가 존재하기 위하여 기본적으로 기술적 개발보다는 다양한 부서(건교부, 행자부, 산자부 등) 사이 유비쿼터스 ID를 적용하기 위한 사회적 합의가 앞서서 필요하다.

5. 결론

유비쿼터스 환경에서 모든 정보는 교류되고 접근되어야 한다. 이러한 수단을 제공하기 위하여는 다양한 응용에서 사용할 수 있는 공통의 정보 체계를 갖추어야 하고, 공통의 정보 체계를 기반으로 표현되는 정보를 수집하고, 관리하는 시스템 체계 또한 갖추어야 한다.

본 연구에서는 공통 정보 체계의 구조와 이 구조가 다양한 유비쿼터스 응용에서 어떻게 사용될 수 있는지 살펴보았다. 특히 공통 정보 체계로 U-City Identifier(UCID)와 U-City Markup Language(UCML)에 대하여 살펴보았으며, 이들 정보가 어떻게 운용되고, 서비스 되는지 살펴보았다.

또한 기존에 개발된 UCI가 유비쿼터스 사회에서 어떻게 변화하여야 하는지 그 가능성을 살펴보았다.

UCID와 UCML이 본 연구의 완성은 아니며, 이제 처음 제시하는 초안이라고 생각하고 있으며, 지속적 연구가 필요할 것이다.

[참고문헌]

- [1] M. Weiser, "The Computer for the 21st Century", Scientific America, pp 94-104, Sept., 1991; reprinted in IEEE Pervasive Computing, pp. 19-25, Jan.-Mar. 2002.
- [2] 유코리아포럼 : www.ukoreaforum.or.kr
- [3] 정보통신부, "u-Korea 구현을 위한 USN 구축 마스터 플랜(안)“, 2005년 11월
- [4] Teruyasu MURAKAMI, "Japan's National IT Strategy and the Ubiquitous Network", NRI Paper, Nomura Research Institute, Nov., 2005
- [5] Tadashi TSUJI, "Next-Generation Ubiquitous Network Strategy", NRP Paper, Nomura Research Institute, May 2006

부록 : UCML 소스

```
<?xml version="1.0" encoding="euc-kr"?>
<!-- edited with XMLSpy v2006 rel. 3 sp1 (http://www.altova.com) by
ChangYeol Lee, Dongeui University -->
<!-- All rights reserved by Chang Yeol Lee -->
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
<xsd:element name="UCML">
  <xsd:annotation>
    <xsd:documentation>Comment describing your root element</xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Events" maxOccurs="unbounded">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="UCID"/>
            <xsd:element name="Date">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="dateType"/>
                  <xsd:element name="value"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="Location">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="locType"/>
                  <xsd:element name="value" maxOccurs="unbounded"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="Objects" maxOccurs="unbounded">
              <xsd:complexType>
```

```

<xsd:choice>
  <xsd:element name="Object">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="parentID" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="objectType"/>
              <xsd:element name="value"/>
              <xsd:element name="data" minOccurs="0"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="childID" minOccurs="0" maxOccurs="unbounded">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="objectType"/>
              <xsd:element name="value"/>
              <xsd:element name="data" minOccurs="0"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
      <xsd:element name="Description">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="objectType"/>
            <xsd:element name="value"/>
            <xsd:element name="data" minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Entity">
  <xsd:complexType>
    <xsd:sequence>

```



```

        <xsd:element name="entityType"/>
        <xsd:element name="value"/>
        <xsd:element name="contact" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="Business" minOccurs="0" maxOccurs="unbounded">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="businessType"/>
            <xsd:element name="value"/>
            <xsd:element name="others" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="Services" minOccurs="0" maxOccurs="unbounded">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="serviceType"/>
            <xsd:element name="value"/>
            <xsd:element name="Description" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>

```