

# 시간적 추론이 적용된 위치 기반 서비스에 관한 연구

김제민<sup>o</sup> 박영택

송실대학교 컴퓨터학과

kimjemins@hotmail.com, park@computing.ssu.ac.kr

## The study about location based service that temporal reasoning was applied

JeMin Kim<sup>o</sup> YoungTack Park

Dept. of Computer Science, Soongsil University

### 요 약

차세대 정보통신 기술의 가장 중요한 패러다임으로 '유비쿼터스 컴퓨팅' 이 새롭게 주목받고 있다. 유비쿼터스 환경에서의 서비스 지원 시스템을 개발하기 위한 중요한 문제 중의 하나는 이동 객체(사용자)의 시간과 이벤트의 관계를 파악하고 위치 이동 데이터로부터 시공간 이동 패턴을 탐사하는 것이다. 본 논문에서는 유비쿼터스 환경 내에서 사용자에게 시간과 관련된 서비스를 적절히 제공하기 위해서 다음과 같은 연구를 한다. 첫째, 서비스 관점에서의 시간적 추론(Temporal Reasoning)이다. 각 사용자들은 각자의 취향을 가지고 있으며 이는 시간과 밀접한 관계를 가지고 있다. 시간과 관련된 사용자의 취향이 기록된 각 사용자 프로파일을 기반으로 서비스 지원 시스템은 적절한 서비스를 제공할 수 있다. 둘째, 사용자의 취향을 기록하기 위한 시간적 추론(Temporal Reasoning)이다. 기록된 내용들은 사용자 프로파일(User Profile)을 생성하는데 도움을 준다.

### 1. 서 론

유비쿼터스 컴퓨팅이 지향하는 "Every Computing"은 사용자가 이동하여 장소가 변하거나, 휴대 또는 사용하는 장비가 바뀌거나, 시간이 지나더라도, 지속적으로 "연결되어" 원하는 작업을 수행할 수 있게 해주는 연속적인 컴퓨팅 환경의 구현을 목표로 한다.[1][2] 유비쿼터스 환경을 구현하기위해 여러 가지 서비스 지원 시스템이 개발되고 있으며, 이 중 위치 기반 서비스(Location Based Service)는 개인의 위치를 파악하여 사용자 서비스 등에 있어 개인화되고 동적인 서비스를 제공한다.[3] 위치 기반 서비스를 개발하는데 있어 대두되는 중요한 문제 중의 하나는 현재의 시간과 사용자에게 서비스된 이벤트의 관계를 파악하고 사용자의 위치 이동 데이터로부터 시공간 이동 패턴을 알아내는 것이다. 이를 위해 서비스되는 이벤트간의 시간적 관계와 시공간적 관계가 고려되어야 하며, 시간 논리(interval temporal logic)를 사용하여 시간과 이벤트 및 이벤트에 따라 변하는 사용자의 행위간의 관계를 표현한다.[4]

본 논문은 유비쿼터스 환경 내에서 사용자에게 시간이 고려된 서비스를 적절히 제공하기 위해서 다음과 같은 연구를 한다. 첫째, 서비스 관점에서의 시간적 추론(Temporal Reasoning)이다. 각 사용자들은 각자의 취향을 가지고 있으며 이는 시간과 밀접한 관계를 가지고 있다. 시간과 관련된 사용자의 취향이 기록된 각 사용자 프로파일을 기반으로 서비스 지원 시스템은 적절한 서비스를 제공할 수 있다. 둘째, 사용자의 취향을 기록하기 위한 시간적 추론(Temporal Reasoning)이다. 기록된 내용들은 사용자 프로파일(User Profile)을 생성하는데 도움을 준다.

### 2. 서비스 시나리오

유비쿼터스 컴퓨팅을 논할 때 흔히 집, 자동차, 회사 등의 모든 기기에 내장된 컴퓨터를 네트워크화하여 사용자의 모든 작업 및 행동에 활용하는 것을 볼 수 있다. 또한 어디에서든지 필요한 정보를 얻을 수 있다는 점을 강조하고 개인을 위한 맞춤형 서비스에 초점이 맞춰져 있다는 것을 알 수 있다. 본 장에서는 실제 유비쿼

터스 환경에서 사용자가 받을 수 있는 시간적 추론이 적용된 유비쿼터스 서비스에 대한 예를 간단한 시나리오를 통해 설명한다.

첫 번째 시나리오는 지능형 회의실 안에서 발표자의 스케줄에 맞추어 발표 순서를 자동으로 지정하는 과정을 보여준다. 2004년 7월 15일 정보 과학 빌딩 213호에서 "유비쿼터스 서비스"에 대한 세미나가 시작된다. 이날 세미나 발표자는 Park, Kim, Choi인데, 이들은 세미나 발표 외에 다른 일정이 있고, 이들의 일정은 각각 자신의 상황 정보 (Context-Information)를 관리하는 개인 에이전트 (Personal Agent)에 의해 관리된다. 세미나가 시작되기 전에 세미나 도우미 에이전트(Meeting Assistant Agent)는 발표자들의 개인 에이전트로부터 당일 스케줄을 넘겨받는다. 이때 전체 세미나 시간 (t1~t30)중 Park이 세미나에 참가 가능한 시간은 (t1~t6), Kim은 (t9~t30), Choi는 (t9~t12) 이다. 이러한 개인 스케줄을 넘겨받은 세미나 에이전트는 이들 스케줄을 바탕으로 시간적 추론을 적용하여 발표 순서를 정한다. 즉 세미나 시작 시간에 가장 먼저 도착하여 발표가 끝나자마자 바로 다른 장소로 이동하는 Park이 가장 먼저 발표를 하고, 세미나에 참가하는 시간은 끝지만 끝날 때 까지 세미나에 참석할 수 있는 Kim 보다 중간에 떠나야하는 Choi가 다음 발표를 하도록 자동으로 발표 순서를 작성한다.

두 번째 시나리오는 위치기반 서비스가 적용된 올 네트워크 환경에서 사용자 프로파일을 이용한 지능형 서비스의 흐름을 보여준다. Hong은 매일 일어나자마자 냉장고로 가서 우유를 마신 후 화장실로 가서 샤워를 한다. 샤워를 마친 후 TV 아침 뉴스를 시청하고, 오디오를 켜서 음악 방송을 듣는다. 음악 방송으로 어느 정도 심신을 맑아진 Hong은 컴퓨터를 부팅하여 하루 일과를 시작하고, 중간 중간 탁자 의자에서 음료를 마시면서 휴식을 취한다. Hong의 이러한 반복된 일과는 저장되어 Hong의 취향을 담은 프로파일로 학습된다.

즉 시간별로 이루어지는 여러 객체와 Hong의 인터랙션은 시간적 추론과 이벤트 추론을 통하여 통합 상황 정보(Relation Model)로 만들어져서 사용자 히스토리 파일에 저장되며, 학습 엔진은 Hong의 사용자 히스토리 파일을 바탕으로 Hong에게 서비스할 이벤트 패턴을 학습하여 Hong의 사용자 프로파일 생성한다. 생성된 사용자 프로파일을 바탕으로 서비스 에이전트는 위치기반 서비스와 연계하여 다음과 같은 서비스를 차례로 수행한다.

- [t16]에 거실의 등불을 켜다.
  - [t18]에 화장실 샤워기의 물 온도를 따뜻하게 한다.
  - [t19~t20]에 화장실의 등불을 켜다.
  - [t22~t35]에 TV를 켜고, Hong이 좋아하는 프로그램이 시작되는 채널을 세팅
  - [t38~t50]에 오디오를 켜고, Hong이 좋아하는 프로그램이 시작되는 주파수를 세팅
  - [t51]에 컴퓨터를 미리 부팅
  - [t130]에 컴퓨터를 대기 모드로 세팅
- Hong은 자신에게 맞춰진 서비스를 제공 받음으로써 일반적인 위치기반 서비스보다 진화된 홈 네트워크 환경에서의 생활을 영위할 수 있다.

### 3. 위치 기반 서비스 (Location Based Service)

본 논문은 중앙 집중적인 센서 네트워크 방식으로부터 위치 추적(location tracking)이 이루어진 상태에서 유비쿼터스 사용자들의 동적 상황에 대한 시맨틱 상황을 추론하는 위치 기반 서비스를 기반으로 하고 있다. 기본적인 유비쿼터스 상황은 캠브리지의 BAT 시스템[6]의 상황과 같지만, BAT기반의 시스템에서는 쿼드 트리 기반의 환경 모델과 공간 인덱싱 방식을 취하여 상황 인지를 수행한 반면에 본 논문에서 사용하기 위해 개발된 시스템은 qualitative 방식으로 상황을 모델하고 지식기반 방식을 적용하여 위치 기반 상황 인지를 수행한다.

본 장에서는 유비쿼터스 환경에서의 각 사물에 대한 기하학적 모델을 구축하고 동적 사물이 이동하는 단계마다 사용자들 간의 위치 관계를 qualitative 모델로 구축하며 이를 기반으로 위치 기반 서비스를 하기 위해 개발된 시스템\*\*에 대해 설명한다. 본 시스템은 객체 구성관리, 정적 공간 모델링, 위치 추적, 상황정보 해석, 상황 정보 통합기능으로 구성된다.

먼저 객체 구성 관리는 집이나 사무실 같은 실제 공간에 배치되는 여러 가지 객체 (TV, 냉장고, 오디오, 침대 등등)들을 정의하고, 관리한다. 이를 위해 각 객체의 식별정보와 타입정보(54인치 TV, 32인치 TV 등등) 및 객체의 2차원적인 크기 (X,Y Size)를 정의한다. 여기에 부가적으로 상황정보 해석을 원활하게 하기 위해 각 객체의 영역(크기 및 타입)을 추가로 정의한다.

정적 공간 모델링은 그래픽 사용자 인터페이스(Graphic User Interface : GUI)를 통해서 집이나 사무실 같은 일정한 공간을 홈맵으로 표현하고, 정의된 홈맵에 여러 객체들을 초기 배치하는 기능이다. 이를 위해 하나의 정적인 공간(집, 사무실)을 표현하는 정보(그래픽 파일 또는 캐드 파일)를 먼저 입력 받은 후 공간의 좌표 값을 설정한다. 여기에 홈맵에 배치될 객체(TV, 냉장고, 오디오, 침대 등등)의 식별정보와 타입정보(54인치 TV, 32인치 TV 등등) 및 홈맵의 특징 기준점으로부터의 2차원 위치를 표현하는 X,Y좌표와 크기를 위치 센서와 객체-속성 테이블로부터 입력 받는다. 위치 추적은 그래픽 사용자 인터페이스(Graphic User Interface : GUI)를 통해 동적/정적 객체의 이동을 홈맵 상에서 표현한다. 이 기능은 동적 객체(사람, 청소로봇, 애완동물 등)의 식별정보와 홈의 특징 기준점으로부터의 2차원 위치를 표현하는 X,Y좌표를 위치 센서로부터 입력 받는다.

상황 정보 해석은 동적 객체(사람, 애완동물 등)가 정적 공간(집, 사무실)내에서 움직였을 경우, 공간 내에서의 동적 객체의 이동 상황을 인지하여 Spatial Model로 표현하고, 이동 객체와 일반 객체의 상황을 인지하여 Qualitative Model로 표현하는 기능이다. 이를 위해서 동적 객체의 식별정보와 특징 기준점으로부터의 2차원 위치를 표현하는 X,Y 좌표를 위치 센서로부터 입력 받고, 정적 공간 모델링 기능으로부터 홈맵 정보와 객체의 Equation을 입력 받는다. Qualitative Model의 명령어는 Positive-overlap, Containment, Negative-overlap으로 나누어져서, (객체 ID, 동적 객체의 ID, 시간)으로 구성된다.

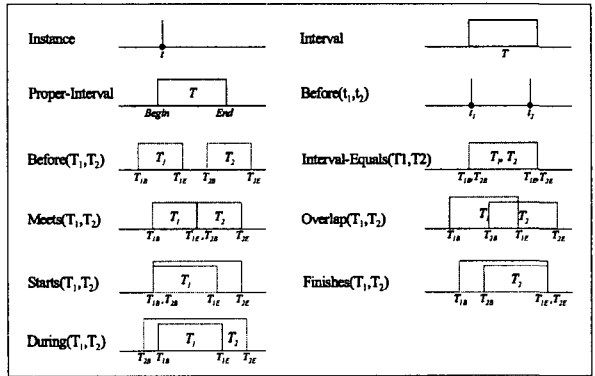
마지막으로 상황 정보 통합은 일정한 시간 동안 발생하는 상황 정보들을 에이전트가 이해할 수 있는 형태로 통합하는 기능이다. 이를 위해 Spatial Model과 Qualitative Model을 실시간으로 입력받는다. 입력 받은 자료를 바탕으로 정리된 상황 정보를 에이전트가 이해할 수

있는 형태인 Relation Model로 표현한다. Relation Model은 (동적 객체의 행위 명령어[동적 객체와 객체의 관계를 표현하는 단어, 객체 ID, 동적 객체ID, 상황 시작 시간, 상황 종료 시간])의 형태로 표현된다.

### 4. 시간적 추론

Allen, J.F 는 행위와 시간간의 관계를 표현함으로써 인공 지능 기법을 이용한 시간적 추론(Temporal Reasoning)이 가능하도록 하였다. 다음[그림 1]은 Allen이 정의한 시간과 시간 간격간의 관계를 보여준다.

instance는 하나의 시간 포인트를 나타내고, interval은 시간 간격을 나타낸다. 시간 간격들의 관계를 표현하기 위해서는 proper-interval이라는 개념을 사용되는데, proper-interval은 시작점(start-point)과 끝점(end-point)이 항상 존재하며, 이 두 점이 같지 않아야 성립된다. 시간 간격간의 관계는 [그림 1]과 같이 크게 7가지로 표현된다.



[그림 1] 시간 간격들 간의 관계

본 논문에서는 위치 기반 서비스로부터 관계명령어를 받아들여 시간적 추론을 실행 하기위해 JESS(Java Expert System Shell)를 사용하였다. JESS는 썬 마이크로 시스템에서 제작한 규칙-기반 전문가 시스템 저작 도구이며, 자바 환경에서 구동된다. JESS는 규칙을 바탕으로 한 추론 엔진을 제공하고, 추론 엔진에서 사용하는 사실(fact)과 규칙(rule)들을 묘사해 준다. 이러한 규칙 기반 프로그램은 수백 혹은 수천 개의 규칙을 가지고 있으며 규칙 베이스 내장된 데이터들을 규칙에 계속적으로 적용한다.

[표 1]은 위치 기반 시스템에서 생성된 관계 명령어를, [표 2]는 관계 명령어로부터 생성된 시간적 사실(Temporal Facts)을 보여준다. 이와 같은 시간적 사실들은 JESS로 정의된 시간적 규칙(Temporal Axiom)에 적용된다. 즉, 관계 명령어들을 추론엔진이 인식할 수 있는 형태로 바꾸는 것이다. 본 논문에서는 시간적 추론을 실행하기 위한 다음과 같은 기본 사실들을 설정하였다. 서비스나 사용자의 행위가 시작된 시간과 끝난 시간을 하나의 시간 객체(Time-Instance)로 표현하고, 서비스나 사용자의 행위가 실행된 시간을 하나의 시간 간격(Time-Interval)으로 표현했으며, 시간 간격과 시간 객체들의 관계를 Time-between으로 표현한다. 또한 begins와 ends를 이용하여 각 시간 간격의 시작점과 끝점을 표현한다.

본 논문을 위해 구축된 JESS로 구현된 규칙 기반 시스템은 작업 메모리에 적재된 초기 사실들을 바탕으로 규칙 베이스 내에 정의된 시간 규칙(Time-Axiom)을 이용하여 시간적으로 연관된 사용자와 객체 및 각 사용자간의 관계를 추론한다. 추론된 사실 역시 작업 메모리에 적재되기 때문에 작업 메모리의 데이터가 방대해지게 된다. 따라서, 이러한 문제점을 보완하여 실제 사용되는 시간적 추론 결과들만을 추출하기 위해 의미 해석 블록을 생성한다.

```
[t1~t6](bed,park,on) , [t1~t6](bed,kim,on)
[t1~t6](bed,park,on) , [t8~t10](tv,park,stay)
[t12~t20](sofa,park,on) , [t22~t25](toilet,park,on)
[t28~t30](toilet,kim,on) , [t34~t50](toilet,choi,on)
[t28~t30](refrigerator, kim, stay) , [t28~t30](light, kim, stay)
[t30~t32](light, choi, stay) , [35~60](radio,park,listen)
[35~75](radio,kim,listen) , [t10~t80](bed, park, on)
[t20~t70](bed, kim, on) , [t53~t80](TV,kim,view)
[t70~t80](TV,choi,view)
...
```

[표 1] Relation Operation

```
stay(on.bed.park.t1.t6)
stay(on.bed.kim.t1.t6)
stay(front.TV.park.t8.t10)
```

```
(instance (time-instance t1))
(instance (time-instance t6))
(interval (time-interval T0))
(time-between (time-interval T0)
  (begin-instance t1) (end-instance t6))
(begins (begin-instance t1)
  (time-interval T0))
(ends (end-instance t6) (time-interval T0))
(time-span (time-interval T0)
  (time-event Event0))
(before (t_bef t6) (t_aft t8))
(instance (time-instance t1))
(instance (time-instance t6))
(interval (time-interval T1))
  (time-between (time-interval T1)
    (begin-instance t1) (end-instance t6))
(begins (begin-instance t1) (time-interval T1))
(ends (end-instance t6) (time-interval T1))
(time-span (time-interval T1)
  (time-event Event1))
```

[표 2] 추론을 위해 생성된 시간적 사실들

이는 작업 메모리에 존재하는 모든 사실중 시간적 관계를 나타내는 사실과 이러한 사실에 연관된 서비스(Event)를 연결하여 블록화시킨 것이다. 이때 각 서비스는 어떤 사용자가 어떠한 객체와 어떠한 관계를 가지고 있는지를 리스트 형태로 저장한다. 이처럼 생성된 의미 해석 블록은 위치 기반 서비스 시스템을 통하여 사용자에게 시간과 연관된 서비스를 제공받은 물론이고 사용자 프로파일을 구축 하기 위해 사용되는 사용자 히스토리 파일을 만드는 중요한 자료로 쓰인다.

```
intervalEquals((T1,Event1), (T2,Event2)), Event1(bed.park,on) Event2(bed.kim,on)
timeBetween((Event1,t1,t2),Event1(bed.park,on)
timeBefore((T1,Event1), (T2,Event2)),Event1(tv.park,stay) Event2(sofa.park,on)
timeBefore((T1,Event1), (T3,Event3)),Event1(toilet.park,on) Event3(toilet.choi,on)
eventInterval((T,Event1), Event1(refrigerator, kim, stay)
intervalMeet((T1,Event1), (T2,Event2)), Event1(light.kim,stay) Event2(light.choi,stay)
intervalStart((T1,Event1), (T2,Event2)), Event1(radio.park,listen)
Event2(radio.kim,listen)
intervalDuring((T1,Event1), (T2,Event2)), Event1(bed.kim,on) Event2(bed.kim,on)
intervalFinish((T1,Event1), (T2,Event2)), Event1(TV.kim,view) Event2(TV.choi.view)
```

[표 5] 의미 해석 블록

5. 실험

본 논문이 제안하는 시간적 추론이 적용된 위치 기반 서비스의 실행 가능성을 평가하기 위해 위치 기반 서비스에 연동되도록 구현된 시간적 추론 엔진(Temporal Reasoning Engine)과 이를 바탕으로

로 한 사용자 히스토리 파일 생성에 대해 실험해 보았다. 본 실험은 3명의 사용자가 유비쿼터스 서비스가 적용된 지능형 홈 환경에서 겪는 20개의 상황에 대해 시뮬레이션을 실행한 후, 생성되는 관계 명령어들을 바탕으로 시간적 추론 엔진이 의미해석 블록과 사용자 히스토리 파일을 생성하는 과정을 보여준다. 관계 명령어를 시간적 규칙이 적용된 추론엔진에 적용한 결과 시간적 추론이 적용된 새로운 상황과 이러한 상황을 표현하는 의미 해석 블록을 얻을 수 있었다. 이와 같은 의미 해석 블록과 기존의 관계 명령어들을 바탕으로 [그림 2]와 같은 사용자 히스토리 파일이 최종적으로 추출되었다.

```
Object_ID || Person_ID || startTime || andTime || Event || Relation || object_status)
bed04a || park || t1 || t6 || on || kim || nil
bed04a || kim || t1 || t6 || on || park || nil
sofa03a || park || t12 || t20 || on || nil || nil
TV05b || park || t8 || t10 || view || nil || on
stool07b || park || t22 || t25 || on || kim,choi || nil
stool07b || kim || t28 || t30 || on || park,choi || nil
stool07b || choi || t34 || t50 || on || park,kim || nil
refrigerator0s1 || kim || t26 || t30 || stay || nil || nil
light0g1 || kim || t28 || t30 || stay || choi || nil || nil
light0g1 || choi || t30 || t32 || stay || kim || on
```

[그림 2] 사용자 히스토리 파일

7. 결론 및 향후 연구

본 논문은 유비쿼터스 환경에서의 서비스 지원 시스템을 개발하기 위한 중요한 문제 중의 하나인 이동 객체(사용자)의 시간과 이벤트의 관계를 파악하고 위치 이동 데이터로부터 시공간 이동 패턴 분석에 대한 연구를 주 목적으로 하고 있다. 이는 사용자에게 시간적 요소가 고려되는 서비스를 제공할 수 있게 한다. 따라서 유비쿼터스 환경 내에서 사용자에게 시간이 고려된 서비스를 적절히 제공하기 위해서 다음과 같은 연구를 진행하였다. 첫째, 서비스 관점에서의 시간적 추론(Temporal Reasoning)이다. 둘째, 사용자의 취향을 기록하기 위한 시간적 추론(Temporal Reasoning)이다. 위 두 연구 사항의 공통점은 사용자 프로파일(User Profile)의 존재이다. 즉, 각 사용자 개개인에 맞는 시간적 요소가 적용된 유비쿼터스 서비스를 제공하기 위해서는 사용자의 취향 및 제공받은 서비스들이 시간 간격(Time Interval)별로 기록되어있는 사용자 프로파일이 필요하며, 이러한 사용자 프로파일을 구축하기 위해서는 시간적 추론이 필요하다. 이를 위해 본 논문에서는 시간적 추론 엔진을 (Temporal Reasoning Engine) 구현하여 유비쿼터스 지원 시스템중의 하나인 위치 기반 서비스 시스템에 적용하였다.

참고문헌

[1] Abowd G, Mynatt E. "Charting past, present, and future research in ubiquitous computing", ACM Transactions on Computer Human Interaction, Vol.7, No.1, pp.29 ~ 58, March 2000  
 [2] 김지인, "유비쿼터스 컴퓨팅: 어떻게 할 것인가?", 정보과학회지, 제 21 권 제 5호, pp.5 ~ 17, 2003.5  
 [3] 이준욱, 백용현, 류근호, "위치 기반 서비스를 위한 이동 객체의 시간 패턴 탐사 기법", 한국정보과학회 논문지D, VOL.29, NO.05, pp.0335 ~ 0346, 2002.10  
 [4] James F Allen, George Ferguson, "Actions and Events in Interval Temporal Logic", In Oliveira Stock (ed.), Spatial and Temporal Reasoning, Kluwer Academic Publishers, pp.205 ~ 245  
 [5] Nissanka B. Priyantha, Anit Chakraborty, Hari Balakrishnan, "The Cricket Location-Support System", 6th ACM International Conference on Mobile Computing and Networking (ACM MOBIKOM), Boston, MA, August 2000  
 [6] Mike Addlesee, Rupert Curwen, Steve Hodges, Joe Newman, Pete Steggles, Andy Ward, Andy Hopper, "Implementing a Sentient Computing System", IEEE Computer, August 2001  
 [7] 우희경, "전북대학교 '차세대 LBS 응용 연구센터' · 부산대학교 '플라스틱정보소재 연구센터', 한국정보과학회지, VOL.22, NO.01, pp.0064 ~ 067, 2004.01