

# 상황 인식 기반의 개인 맞춤형 추천 시스템\*

## (A Customized Recommendation System based on Context Awareness)

윤혜진 · 신현정 · 창병모

숙명여자대학교 정보과학부 컴퓨터과학전공

(panggru@sookmyung.ac.kr, twinkyy83@sookmyung.ac.kr, chang@sookmyung.ac.kr)

### 요약

본 연구는 사용자의 주변 상황을 기반으로 변화하는 상황에 자동적으로 적응하는 개인 맞춤형 추천 서비스를 개발하여 사용자에게 보다 편리하고 실용적인 서비스를 제공함을 목적으로 한다. 본 시스템은 변화하는 상황에 자동적으로 적응하기 위해 상황 적응 시스템을 이용한다. 프로그래머는 정책 기술 언어를 이용해 상황 적응 정책 파일을 작성하고, 본 연구에서 적용한 상황 적응 시스템은 작성된 정책 파일의 내용에 따라 변화된 상황에 맞도록 유비쿼터스 프로그램을 자동적으로 적응시키고, 개인 맞춤형 추천 웹 서비스는 변화된 상황을 기반으로 사람의 선호도와 검색 히스토리가 반영된 개인 맞춤형 음식점 추천 서비스를 제공한다.

## 1. 서론

상황 인식 컴퓨팅이란 사용자의 입력을 최소화하여 언제 어디서든 서비스를 받을 수 있도록 환경을 제공하는 것이다. 상황 인식 컴퓨팅에 대한 요구가 높아지면서 활발한 연구를 통해 다양한 연구 결과가 나왔다. 이러한 연구 결과들을 사용한 실제 서비스가능한 애플리케이션 개발은 미미한 상황이다.

본 연구의 목적은 사용자의 주변 상황을 기반으로 변화하는 상황에 자동적으로 적응하는 보다 편리하고 실용적인 개인 맞춤형 추천 서비스를 제공하고자 한다. 또한 상황 적응 시스템을 이용하여 개발함으로써 실제 서비스에도 적용 가능함을 보여준다. 이를

위해 정책 파일을 이용해 간단히 상황 적응이 가능한 상황 적응 시스템과 사용자의 개인 선호도와 과거 검색 히스토리 등의 개인 정보를 이용하여 상황 인식 기반의 맞춤형 추천 정보를 제공하는 음식점 추천 서비스를 개발하여 보았다.

본 논문의 연구 내용은 정책 기술 언어 (Policy Description Language)[1]를 이용하여 변화하는 상황에 자동으로 적응하는 규칙을 정책 파일에 작성한다. 본 연구에 적용한 상황 적응 시스템[5]은 정책 파일에 따라 변화하는 상황에 자동으로 유비쿼터스 프로그램을 적응시켜 변화된 상황을 기반으로 사용자의 선호도와 검색 히스토리가 반영된 개인 맞춤형 추천 서비스를 제공할 수 있도록 돕는다.

이 논문은 관련 연구, 시스템 설계 및 구현, 결론 및 향후 연구로 구성된다.

\* 본 연구는 교육과학기술부/한국과학재단 우수연구센터 육성사업의 지원으로 수행되었음. (R11-2008-007-02002-0)

## 2. 관련 연구

상황 인식 컴퓨팅에서 수행되었던 대표적인 연구들을 살펴보면 다음과 같다.

Gaia 위에서 개발한 상황인식 시스템[3]은 상황에 따라 나타나야 하는 반응을 1차 논리 구조(first-order logic)에 기반하여 작성하고 처리한다. 상황 인식 자원 관리 미들웨어 CARMEN[4]은 메타데이터의 내용을 기반으로 하여 무선 인터넷 환경에서 변화된 환경을 재구성한다. 본 연구에서 이용한 상황 적응 시스템은 모든 유비쿼터스 환경에 대해 적용 가능하며, 상황 적응 정책을 정의하면 해당 규칙에 따라 간단히 프로그램을 적용시킬 수 있다.

### 2.1. JCAF

JCAF[2]는 상황 인식 프로그램 개발을 위한 하부구조와 API를 제공하는 자바 기반의 상황 인식 프로그래밍 프레임워크이다. 이 프레임워크는 상황 인식을 효과적으로 지원할 수 있다. JCAF는 현실의 상황을 가상으로 표현하기 위한 Entity, Relationship, ContextItem 등의 라이브러리를 제공하며, 이렇게 표현된 상황 정보를 수집하고 유지 및 관리하는 ContextServie와 상황 변화를 감지하는 EntityListener 등의 라이브러리를 제공한다.

### 2.2. 정책 기술 언어

정책 기술 언어[1]는 변화하는 상황에 어떻게 적용시킬지 적응하기 위하여 사용된다. 먼저 정책 기술 언어를 사용하여 상황을 표현하는 방법에 대해 살펴보고, 다음으로 상황 적응 정책을 어떻게 정의하는지 설명한다.

유비쿼터스 환경에서 엔티티는 물리적·논리적인 공간이나, 움직이거나 움직이지

않는 객체를 나타낸다. 예를 들어, “floor”, “sickroom”은 공간 엔티티이고, “PDA”는 움직이는 엔티티, “bed”는 움직이지 않는 엔티티이다. 공간 엔티티나 움직이지 않는 엔티티의 경우는 그 위치가 고정되어 변하지 않는 반면, 움직이는 엔티티의 경우에는 동적으로 그 위치나 상황이 변하게 된다. 실세계의 엔티티들은 프로그램에서 Entity 클래스의 인스턴스로서 [그림 1]과 같은 문법 구조로 표현된다.

$$\begin{aligned}
 p \in \text{Entity-Expression} & := id_1 : id_2 \mid \$id \mid \$id_n \mid * \\
 & \quad \mid p_1/p_2 \mid \dots/p \\
 r \in \text{Relation-Expression} & := id_1(p_1, id_2, p_2) \\
 & \quad \mid \sim r \mid r_1 r_2 \\
 n \in \text{Number}
 \end{aligned}$$

[그림 1] 상황 표현을 위한 문법 구조

Entity-Expression은 한 개의 엔티티나 엔티티의 집합을 표현한다.  $id_1 : id_2$ 는 Pda:BobPda와 같이  $id_1$ 이라는 이름을 갖는 클래스의 인스턴스  $id_2$ 를 표현한다.  $\$id$ 는  $id$ 라는 이름을 갖는 클래스의 일반적인 인스턴스 중의 하나를 나타내는 변수를 의미한다.  $\$Room$ 은 Room이라는 클래스의 어떤 변수를 담은 변수가 된다.

$p_1/p_2$ 는 엔티티들간의 종속 관계를 표현한다. 예를 들어, 아래 표현은

Building : UbiSoft/Floor : fl1/\$Room

은 ubihosp라는 건물의 1층에 있는 방을 의미하게 되고, 아래의 식은

Building : UbiSoft/.../\$Pda

UbiSoft라는 건물에 있는 모든 Pda들의 인스턴스를 의미한다.

다음의 식은 UbiSoft 건물의 fl1에 있는 모든 엔티티들을 의미한다.

Building : UbiSoft/Floor : f11/\*

또한 ‘~’은 NOT을 의미하며, ‘^’은 AND 연산자로서 여러 개의 관계에 대해 동시에 참, 거짓을 판단하도록 유도한다.

상황 적응 정책은 변화하는 상황에 어떻게 적응해야 할 것인가에 대하여 정의한 것이다. 이를 위한 문법 구조는 다음과 같다.

```
d ∈ Adaptation-Rule ::= r => a | d1 d2
a ∈ Action ::= p1.id(p2) | id1(p1,id2,p2) | a1;a2
```

[그림 2] 상황 적응 정책의 문법 구조

Adaptation-Rule은 상황에 따라 어떻게 반응해야 할지를 기술하기 위한 식이다. 이 식은 조건(r)이 참이 된다면 행동(a)을 실행하라는 것을 의미한다. 적응 규칙이 실행되기 위한 규칙은 Relation-Expression을 이용하여 표현되고, 이 조건이 만족되었을 때 자동으로 실행되는 행동은 매소드 호출

```
$Pda.showInfo($Person)
```

이나

```
id1 ( p1 , id2 , p2 )
```

새로운 관계의 생성으로 표현된다. 예를 들어, ‘UbiSoft라는 빌딩의 사무실에 사람이 들어오면 관리자의 PC에 그 사람의 정보를 보여준다.’라는 상황 적응 정책은 아래와 같이 작성될 수 있다. 이것은 UbiSoft라는 빌딩의 사무실에 어떤 사람이 들어왔다는 상황 변화가 일어났을 때 자동으로 실행되어 관리자의 PC에 그 사람의 정보를 보여주게 된다.

```
Location($Person, IsIn, Building:UbiSoft/.../$Room) ^
Ownership($Admin, Owns, $PC)
=> $PC.showInfo($Person)
```

[그림 3] 상황 적응 규칙의 예

### 2.3 상황 적응 시스템

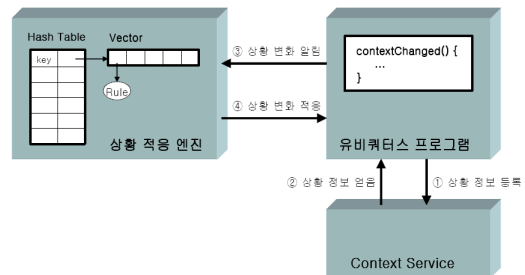
상황 적응 시스템[5]은 상황 적응 정책이 기술되어 있는 정책 파일을 읽고 분석하기 위한 정책 파일 리더기와 그 분석 내용에 따라 유비쿼터스 프로그램을 상황에 적응시키기 위한 상황 적응 엔진으로 구성되며 시뮬레이션을 제공한다.

정책 파일 리더기는 상황 적응 정책을 클래스 형태로 분석하고 저장한다. 상황 적응 엔진은 정책 파일 리더기가 분석해 준 상황 적응 정책의 내용을 기반으로 유비쿼터스 프로그램을 변화하는 상황에 적응시킨다. [그림 4]는 상황 적응 엔진이 실행되는 시스템의 구조를 보여준다.

상황 적응 엔진은 프로그래머가 작성한 정책 파일을 분석하고, 그 정보를 이용하여 해시 테이블을 구성한다. 그 후, 응용 프로그램에서의 상황 변화가 인지되면 해시 테이블에서 변화와 관련된 규칙을 검색하여 조건이 참의 값을 갖는지 검사한다. 해당 조건의 일치여부에 따라 해당하는 행동을 자동적으로 수행하게 된다.

### 3. 시스템의 설계 및 구현

본 논문에서 개발한 상황 인식 기반의 개인 맞춤형 추천 서비스는 사용자의 주변 상황을 인식하여 변화되는 상황에 따라 본 시



[그림 4] 유비쿼터스 프로그램과의 연동 모습

스탬이 적응하여 사용자에게 맞는 정보를 제공해준다.

### 3.1. 시스템의 개요

본 시스템은 다음과 같은 주요 특징을 갖는다.

#### 가. 상황 인식을 기반한 웹 서비스

본 시스템은 GPS를 기반으로 현재 위치와 같은 상황 정보를 이용하여 사용자에게 맞는 정보를 제공한다.

#### 나. 편리한 사용자 인터페이스

본 시스템은 사용자가 PDA와 GPS 장치를 가지고 있으면 어디서든 서비스를 이용할 수 있다.

#### 다. 상황 적응 개인 맞춤형 추천 서비스

본 시스템은 사용자의 주변 상황에 따라 각 개인에게 맞는 내용을 구성하여 추천해준다.

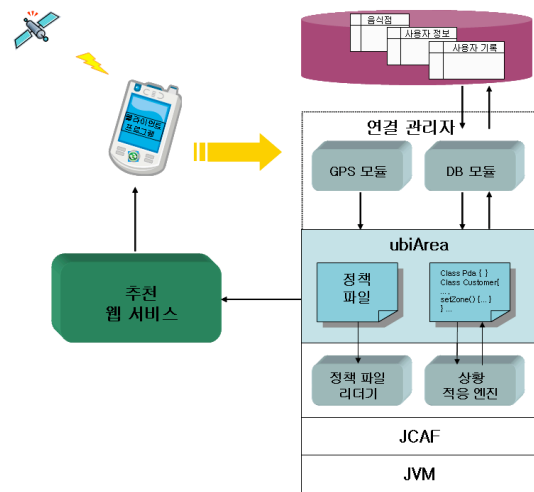
#### 라. 정책 파일을 이용한 편리한 개발 환경

본 논문에서는 상황 적응 시스템이 제공해주는 환경을 이용하여 개발자가 정책 파일만 작성하면 자동으로 본 시스템을 적응시켜준다.

### 3.2. 시스템의 구성도

본 시스템은 크게 네 부분으로 나뉜다. 첫 번째 부분은 상황 적응 시스템과 유비쿼터스 프로그램인 ubiArea 이다. 상황 적응 시스템은 현실의 상황이 변화할 때마다 ubiArea를 정책 파일에 미리 작성된 적응 규칙에 따라 자동적으로 적절히 반응하도록 만들어 줌으로써 사용자의 변화하는 상황 즉 현재 위치에 따라 적응시킨다.

두 번째 부분은 상황 적응 시스템에 의해



[그림 5] 개인 맞춤형 음식점 추천 서비스의 구성도

적용된 유비쿼터스 프로그램인 ubiArea로부터 받은 정보를 이용하여 사용자의 선호도와 검색 히스토리에 따라 웹 페이지에 보여주는 우선순위를 정하여 사용자에게 더 필요한 정보가 먼저 보일 수 있도록 하는 음식점 추천 시스템이다.

세 번째 부분은 연결 관리자(Connection Manager)이다. 연결 관리자는 사용자의 현재 위치를 받아오는 GPS모듈과 음식점의 정보를 가져올 수 있도록 도와주는 DB모듈로 구성되어 있다.

마지막 네 번째 부분은 사용자의 PDA에 설치되어 현재 위치 정보를 연결 관리자에게 전달해주는 클라이언트 프로그램이다.

### 3.3. 음식점 추천 서비스의 정책 파일

[그림 6]은 본 시스템을 위한 정의이다. ubiar라는 유비쿼터스 공간이 있고, 그 공간 안에 Zone이 존재한다는 의미이다. 두 번째 정의는 개체인 Customer가 Zone 안에 있다는 의미이고, 세 번째 정의는 Customer가 PDA를 가지고 있다는 상황 표현이다. 또한 Customer는 위치를 가질 수 있다는

상황을 의미한다.

```
Areas:ubiar[Zone]
Location(Customer, IsIn, Zone)
Ownership(Customer, Owns, Pda)
Ownership(Customer, Have, Position)
Status(Restaurant, IsOccurriedEmpty, Seats)
```

[그림 6] 음식점 추천 서비스의 정의

[그림 7]은 본 시스템에 대한 정책 파일이다. 첫 번째 규칙은 사용자가 PDA를 가지고 있고, 사용자가 현재 위치를 가지고 있으면 사용자는 Zone 안에 존재한다 라는 정책이다. 이 정책은 사용자가 가지고 있는 PDA에서 현재 위치 정보를 갖게 되면 사용자의 현재 위치 정보를 중심으로 Zone 안에 있다는 의미하게 된다. 두 번째 정책은 학생이 PDA를 가지고 있고, 그 학생이 Zone안에 존재한다면, 그 PDA에 Zone의 정보를 보여주라는 정책이다. 이 정책은 사용자의 현재 위치 정보를 중심으로 일정 영역인 Zone안에 있는 음식점의 목록들을 보여야를 의미한다.

```
Ownership($Customer, Owns, $Pda) ^
Ownership($Custermer, Have, $Position)
=> $Customer.setZone($Position)

Location($Customer, IsIn, Areas:ubiar/$Zone) ^
Status(Restaurants, IsOccurriedEmpty, Seats)
=> $Pda.getList($Pda)
```

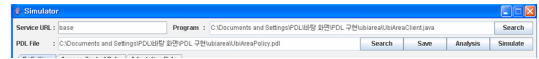
[그림 7] 음식점 추천 서비스를 위한 상황 적응 정책

#### 4. 실험

음식점 추천 서비스는 상황 적응 시스템에서 제공하는 시뮬레이터를 이용해 서비스

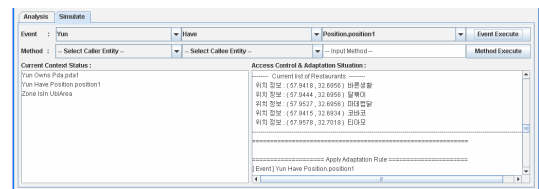
를 실행시킨 후, 웹 브라우저를 이용해 서비스 이용 가능하다.

[그림 8]은 유비쿼터스 프로그램인 ubiArea와 정책 파일을 등록한 화면을 보여준다.



[그림 8] 상황 적응을 위한 환경 설정

시스템이 실행되면 사용자의 변화되는 현재 위치에 따라 현재 상황 정보를 [그림 9]의 시뮬레이터의 Current Context Status 화면에서 확인할 수 있다.



[그림 9] 음식점 추천 서비스에 대한 현재 상황 및 실행된 상황 적응 정책

본 논문에서는 실험 대상 지역을 숙명여대 근처로 지정하여 사용자의 현재 위치와 서로 다른 선호도에 따른 추천 정보를 확인해보았다.

[그림 10]은 사용자의 현재 위치정보에 따라 사용자의 선호도가 반영되어 변화되는 추천 목록들을 확인할 수 있다.



[그림 10] 사용자의 현재위치에 따른 추천

## 5. 결 론

본 시스템은 사용자의 현재 변화되는 상황에 적응하여 그 사용자에게 맞는 실용적이고 편리한 개인 맞춤형 추천 서비스를 제공한다.

실제 응용 서비스에서도 상황 적응 시스템을 기반으로 정책 기술 언어를 이용해 서비스에 대해 정의하고, 변화되는 상황에 적응해야 하는 규칙을 작성하여 손쉽게 상황 적응 정책만을 이용해 사용자 혹은 PDA 같은 정보 통신 기기의 주변 상황 변화에 따라 사용자가 입력하지 않아도 상황 적응 시스템이 자동으로 이 서비스를 적응시켜 사용자가 원하는 서비스를 제공받을 수 있도록 도와준다. 이러한 결과는 실제 애플리케이션에도 상황 적응 시스템이 응용 가능성을 보여준다.

## 참고 문헌

- [1] Joonseon Ahn, Kyung-Koo Doh and Byeong-Mo Chang, A Policy Description Language for Context-based Access Control and Adaptation in Ubiquitous Environment, TRUST '06, Seoul, Korea, Aug. 2006
- [2] J. E. Bardram, The Java Context Awareness Framework-A Service Infrastructure and Programming Framework for Context-Aware Applications, Third International Conference, Pervasive2005, Munich, Germany, May, 2005
- [3] A. Ranganathan, Roy H. Campbell, An infrastructure for context-awareness based on first order logic, Springer-Verlag London Limited 2003, November 2003
- [4] Paolo Bellavista, Rebecca Montanari, Context-Aware Middleware for Resource Mngement in the Wireless Internet, reference IEEECS, Aug.2003
- [5] Minkyong Oh, Jiyeon Lee, Byeong-Mo Chang, Joonseon Ahn, And Kyung-Goo Doh, A Programming Environment for Ubiquitous Computing Environment, SIGPLAN notices Volume42(4), pp.14-22, April 2007