

# 모바일 퍼스널 어시스턴트의 지능 행위 구현을 위한 스마트 스크립트 시스템

김 인 철<sup>†</sup> · 오 휘 경<sup>\*\*</sup>

## 요 약

본 논문에서는 동적 모바일 컴퓨팅 환경에 적합한 계획 실행 모델을 제시하고, 이 모델에 기초해 개발된 스마트 스크립트 시스템을 소개한다. 이 스마트 스크립트 시스템은 모바일 퍼스널 어시스턴트의 작업 지식을 기술하는 스마트 스크립트 언어와 스크립트들을 작업 목표와 환경 변화에 따라 동적으로 실행하는 실행 엔진 등을 포함한다. 또한, 본 논문에서는 스마트 스크립트 시스템의 유용성과 성능을 평가하기 위해 응용 서비스인 Smart Reservation를 구현하고, 실험을 전개한 결과를 소개한다.

키워드 : 모바일 퍼스널 어시스턴트, 계획 실행 모델, 스마트 스크립트 언어, 실행 엔진

## A Smart Script System for Implementing Intelligent Behaviors of Mobile Personal Assistants

In-Cheol Kim<sup>†</sup> · Hui-Kyoung Oh<sup>\*\*</sup>

### ABSTRACT

In this paper, we present the plan execution model for dynamic mobile computing environments, and then introduce the smart script system developed on these base models. The smart script system includes the smart script language, in which the task knowledge of a mobile personal assistant is represented, and the script execution engine, by which the scripts are dynamically executed in response to the given task goal and the environmental changes. In order to evaluate the utility and the performance of our system, we implement an application service called Smart Reservation and conduct some experiments.

Keywords : Mobile Personal Assistant, Plan Execution Model, Smart Script Language, Execution Engine

### 1. 서 론

최근 스마트폰, 태블릿 PC 등의 다양한 스마트 이동 단말기들이 급속히 보급되면서, 스마트 이동 단말기 사용자들 위해 다양한 편의 서비스를 제공하는 모바일 퍼스널 어시스턴트(Mobile Personal Assistant)에 대한 연구도 활발히 진행되고 있다[1, 2]. 이동 단말기 사용자가 수행하고 싶은 작업들을 위임받아 자동 혹은 반자동으로 이들을 대신 수행할 수 있는 모바일 퍼스널 어시스턴트는, 자신이 달성해야 할 작업 목표뿐만 아니라 끊임없이 변화하는 실제 환경 및 정보 환경에 맞추어 자신이 보유하고 있는 작업 수행 지식, 즉 작업 계획을 효과적으로 실행할 수 있어야 한다. 이러한

모바일 퍼스널 어시스턴트의 능동적, 지능적 행위를 효과적으로 구현하기 위해서는, 효율성이 높은 반응형 계획 실행 시스템(Reactive Plan Execution System)의 개발이 요구된다[3, 4]. 본 논문에서는 동적 모바일 컴퓨팅 환경에 적합한 작업 계획 모델과 실행 모델을 제시하고, 이에 기초해 개발된 스마트 스크립트 시스템(Smart Script System)을 소개한다. 이 스마트 스크립트 시스템은 모바일 퍼스널 어시스턴트의 작업 지식을 기술하는 스마트 스크립트 언어(Smart Script Language)와 스크립트들을 작업 목표와 환경 변화에 따라 동적으로 실행하는 실행 엔진(Execution Engine) 등을 포함한다. 끝으로, 본 논문에서는 스마트 스크립트 시스템의 유용성과 성능을 평가하기 위해 응용 서비스인 Smart Reservation를 구현하고, 실험을 전개한 결과를 소개한다.

※ 본 연구는 경기도의 경기도지역협력연구센터사업의 일환으로 수행하였음.

<sup>†</sup> 종신회원: 경기대학교 컴퓨터과학과 교수

<sup>\*\*</sup> 준회원: 경기대학교 컴퓨터과학과 석사과정

논문접수: 2011년 3월 20일

심사완료: 2011년 3월 28일

### 2. 계획 및 실행 모델

스마트 스크립트 시스템은 다음과 같은 계획 모델(Plan

Model)과 계획 실행 모델(Plan Execution Model)을 기초로 개발되었다.

**계획 모델 p = (ID, Name, Type, Body, Comment),**

- 이때 ID는 식별번호(identification number),
- Name은 계획 이름(plan name),
- Type은 사건-주도 계획(event-driven plan),
- 목표-주도 계획(goal-driven plan), 혹은
- 명령-주도 계획(command-driven plan)중 하나
- Body는 <a1, ..., an>와 같은 일련의 동작들,
- Comment는 계획에 관한 설명문이다.

계획 모델 p는 서로 다른 3 가지 유형의 계획들을 표현할 수 있다. 먼저, 사건-주도 계획  $p_e$ 은 특정 사건이 발생하면 자동으로 실행되는 계획을 의미하며, 목표-주도 계획  $p_g$ 은 특정 목표(goal)를 달성하기 위해 실행될 수 있는 계획을 의미하며, 명령-주도 계획  $p_c$ 은 언제든지 사용자의 필요에 따라 선택하여 즉시 실행시킬 수 있는 계획을 의미한다.

모바일 어시스턴트를 위한 계획 실행 모델(Plan Execution Model) PEM을 다음과 같이 정의한다.

**계획 실행 모델 PEM = <P, S, W, U, I, N, T>**,

- P : 계획(plan)들의 집합,  $P = P_e \cup P_g \cup P_c$ , 여기서
  - $P_e$ 는 사건-주도 계획들(event-driven plans),
  - $P_g$ 는 목표-주도 계획들(goal-driven plans),
  - $P_c$ 는 명령-주도 계획들(command-driven plans).
- S : 외부 환경(environment)에서 발생하는 사건(event)들을 감지하는 감지기(sensor),  $S : E \rightarrow 2^F$ ,  
이때 E는 사건들(events), F는 사실들(facts)의 집합.
- W : 환경의 현재 상태를 나타내는 사실들의 집합,  
즉 월드 모델(world model),  $F \subseteq W$
- U : 계획 실행과 연관된 사용자의 제어 명령들(control commands),  $u \in \{execute\_plan, stop\_plan, activate\_plan, deactivate\_plan, \dots\}$
- I : 실행할 계획을 결정하는 해석기(interpreter),  
 $I(W, u, P) = p$ , 이때, 계획  $p \in P$ 는 월드모델  $W = PRECOND(p)$ 이거나 사용자 명령  $u \in U$ 와 연관된 것
- N : 실행이 결정된 계획의 실제 수행과정을 관리하는 의도 구조(intention structure),  
 $N(p) = SUCCESS(DO(a1))?$   $N(p') : STOP(p)$ ,  
이때  $BODY\_SEQ(p) = \langle a1, a2, \dots, an \rangle$ ,  
 $BODY\_SEQ(p') = \langle a2, a3, \dots, an \rangle$ , 동작  $ak \in A$
- T : 계획을 구성하는 각 동작(action)을 실제 수행하여 환경을 변화시키는 실행기(effector),  $T : A \rightarrow E$

**3. 스마트 스크립트 언어**

앞서 소개한 계획 및 실행모델에 기초하여 다음과 같이 스마트 스크립트 언어를 설계하였다. 스마트 스크립트 언어에서는 다양한 내장변수와 사용자-정의 변수를 이용할 수 있다. 스마트 스크립트 언어에서는 환경의 상태를 사실(fact)들의 집합으로 표현한다.

사실(Fact) ::= "(" 변수(Variable) 값(Value) ")"

즉, 하나의 사실은 (%SMSRA "02-123-3456")와 같이 하나의 변수와 그것의 현재 값으로 표현한다. 스크립트 언어에서 사용 가능한 동작들은 크게 실행 순서를 결정하는 제어동작들과 일반 실행동작들로 구성된다. 제어동작들로는 목표 동작, 스크립트 동작, IF 동작, ENDIF 동작, WHILE 동작, ENDWHILE 동작, WAITUNTIL 동작, WAIT 동작, SUCCESS 동작, FAIL 동작, SET 동작 등이 있다. 특히 이중에서 목표 동작은 새로운 부속 목표(subgoal)를 추가하는 동작, 스크립트 동작은 새로운 스크립트의 실행을 요구하는 동작, SET 동작은 사용자-정의 변수에 값을 배정하는 동작 등을 각각 의미한다. 스마트 스크립트 언어에는 발화(SPEAK), 전화 걸기(CALL), 문자메시지 발송(SENDSMS), 음악파일 연주(PLAYMUSIC) 등 다양한 실행동작들이 정의되어 있다.

스마트 스크립트 언어에서는 명령-주도, 사건-주도, 목표-주도 등 3 가지 종류의 스크립트를 정의할 수 있다. 명령 주도 스크립트는 사용자가 직접 실행할 수 있는 스크립트를 의미한다.

명령-주도 스크립트(Command-driven Script)  
 ::= "(" "script" 스크립트이름(Script-Name)  
     [":context" 조건리스트(Condition-List)  
     ":body" 동작리스트(Action-List)  
     [":comment" 설명문(Description)] ")"

사건 주도 스크립트는 외부 환경에서 발생하는 특정 사건에 대응하여 자동으로 실행되는 스크립트를 의미한다.

사건-주도 스크립트(Event-driven Script)  
 ::= "(" "script" 스크립트이름(Script-Name)  
     ":precondition" 조건리스트(Condition-List)  
     [":context" 조건리스트(Condition-List)  
     ":body" 동작리스트(Action-List)  
     [":comment" 설명문(Description)] ")"

목표 주도 스크립트는 사용자나 상위 계층의 스크립트가 목표만 제시하면, 해당 목표 달성을 위해 자동으로 선택되어 실행할 수 있는 스크립트를 의미한다.

목표-주도 스크립트(Goal-driven Script)  
 ::= "(" "script" 스크립트이름(Script-Name)  
     ":goal 목표(Goal)  
     [":precondition" 조건리스트(Condition-List)  
     [":context" 조건리스트(Condition-List)  
     ":body" 동작리스트(Action-List)  
     [":comment" 설명문(Description)] ")"

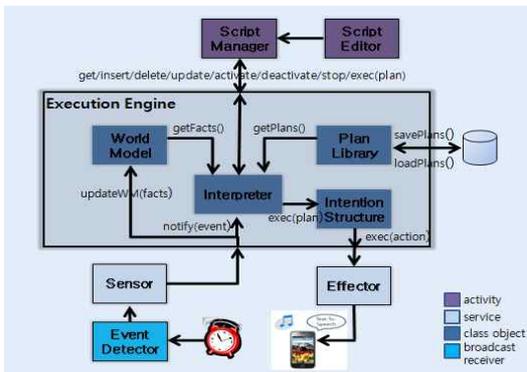
스마트 스크립트 프로그램은 아래와 같이 목표 선언부, 사실 선언부, 그리고 스크립트 정의부 등 크게 세 부분으로 구성된다.

스크립트 프로그램(Script Program)  
 ::= [ 목표선언부(Goal-Declaration) ]  
     [ 사실선언부(Fact-Declaration) ]  
     스크립트정의부(Script-Definition)  
 목표선언부(Goal-Declaration) ::=

"/ ("goals" 목표동작리스트(Goal-Action-List) ") "  
 사실선언부(Fact-Declaration) ::= "  
 "/ ("facts" 사실리스트(Fact-List) ") "  
 스크립트정의부(Script-Definition) ::= "  
 "/ ("scripts" 스크립트리스트(Script-List) ") "

#### 4. 스마트 스크립트 시스템

스마트 스크립트 시스템은 (그림 1)과 같이 크게 스크립트 자동 실행을 총괄하는 실행 엔진(Execution Engine), 환경으로부터 다양한 사건과 상태 정보를 받아들이는 센서(Sensor), 각 기본 작업/동작의 실행을 담당하는 실행기(Effector), 그리고 사용자가 스크립트를 편집하고 수정할 수 있는 스크립트 편집기(Script Editor), 사용자가 스크립트의 실행을 실시간으로 감시하고 제어할 수 있는 스크립트 관리기(Script Manager) 등으로 구성된다. 센서는 문자메시지 도착, 전화 벨 울림, WiFi 검출, 예약 시간 통보 등 스마트폰 환경에서 발생 가능한 다양한 사건들을 감지하고 이들과 연관된 새로운 사실(Fact)들로 실행 엔진 내부의 월드 모델을 갱신한다.

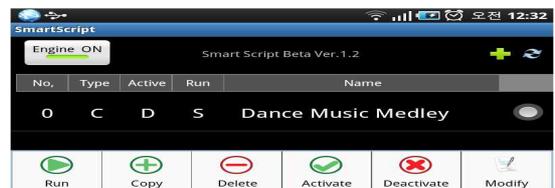


(그림 1) 스마트 스크립트 시스템의 구조

스크립트 실행을 총괄하는 실행 엔진은 내부에 각기 다른 역할을 담당하는 월드 모델(World Model), 계획 라이브러리(Plan Library), 해석기(Interpreter), 의도 구조(Intention Structure) 등의 구성요소들을 포함하고 있다. 월드 모델은 센서를 통해 감지된 환경의 상태나 사용자 입력 데이터, 혹은 각 기본 작업/동작의 실행 결과 데이터 등을 나타내는 사실들을 저장, 갱신함으로써, 언제나 스크립트가 실행되고 있는 현재 시스템의 상황을 표현한다. 반면에 계획 라이브러리에는 현재까지 사용자가 정의한 스크립트들이 실행 대기 상태로 저장되어 있으며, 필요할 때 DB에 스크립트들을 저장하거나 DB에 저장된 스크립트들을 가져오기도 한다. 해석기는 월드 모델의 사실들을 기초로 계획 라이브러리내의 각 사건-주도 스크립트의 전-조건이 만족되는지 검사하거나, 또는 사용자가 제시한 작업 목표(Task Goal)에 각 목표-주도 스크립트가 매치되는지 검사하여, 다음 실행 할 하나의 스크립트를 선택하는 역할을 수행한다. 의도 구조는 해석기에 의해 선택된 스크립트를 전달받아 해당 스크립트 몸체(Body)에 기술된 기본 작업/동작들을 차례대로 실행함

으로써 각 스크립트가 의도하는 작업을 수행하는 역할을 담당한다. 실행 엔진은 항상 백그라운드에서 수행 상태에 있으면서 스크립트와 연관된 사건 발생과 사용자 입력에 따라 빠른 스크립트 실행을 보장하기 위해 하나의 Android 서버로 구현되었다.

실행 엔진이 스마트 스크립트 시스템의 논리적 의사결정(decision-making) 과정을 구현한 것이라고 보면, 별도의 Android 서비스로 구현된 실행기(Effector)는 웹브라우저, 캘린더, Email 프로그램 등 다른 기존의 전용 프로그램들을 이용하거나 Android API로 구현한 자체 프로그램을 이용하여 실행 엔진에서 요청하는 각각의 기본 작업/동작을 실제로 실행해주는 역할을 담당한다.



(그림 2) 스크립트 관리기의 실행 화면



(그림 3) 스크립트 편집기의 실행 화면

스크립트 관리기(Script Manager)는 스마트 스크립트 시스템의 주된 사용자 인터페이스를 제공한다. 사용자는 스크립트 관리자를 통해 명령-주도 스크립트중 하나를 직접 실행시킬 수 있으며, 실행 중인 스크립트를 중지시킬 수 있고, 사건-주도 스크립트들을 활성화하거나 비활성화할 수 있는 등 스크립트들에 대한 실시간 제어가 가능하다. 또한, 스크립트 편집기의 도움을 받아 새로운 스크립트를 생성, 복사, 변경, 삭제할 수 있다. 스크립트 편집기(Script Editor)는 스마트폰 화면의 크기와 키보드의 제한성을 고려하여, 대부분의 편집 과정에 그래픽 아이콘과 메뉴 선택 방식을 적용함으로써 사용자의 키보드 입력을 최소화하도록 구현하였다. (그림 2)는 실행 중인 스크립트 관리기 화면을, (그림 3)은 스크립트 편집기 화면을 각각 나타낸다.

#### 5. 응용 서비스

본 연구에서 개발된 스마트 스크립트 시스템의 성능과 유용성을 알아보기 위해, 스마트폰 사용자를 위해 주변 식당을 검색하고 예약해주는 모바일 퍼스널 어시스턴트의 한 응용서비스인 Smart Reservation을 개발하였다. Smart Reservation은 (그림 4)와 같이 주변 식당 검색을 위해, GPS를 이용하여 사용자의 현재 위치정보를 받아들 수 있고, 또 사용자로부터 음성질의를 입력받을 수 있다. Smart Reservation은 사용자의 음성질의를 SPARQL 형식의 시맨

텍 질의로 변환한 다음, 네트워크를 통해 원격의 식당 정보 웹서비스에 검색 요청을 보낸다. 그리고 웹서비스로부터 받은 검색 결과를 토대로 Google Maps Mashup 기능을 이용하여 식당 위치를 지도에 표시해줄 수 있을 뿐 아니라, 직접 온라인 식당 예약을 해줄 수 있다.



(그림 4) Smart Reservation의 구성도



(그림 5) Smart Reservation의 실행 화면

Smart Reservation은 스마트 스크립트 시스템을 토대로 개발되었으며, 따라서 Smart Reservation의 주된 작업 지식은 스마트 스크립트들로 표현되어 스마트 스크립트 실행 엔진에 의해 수행되었다. Smart Reservation의 효과적인 구현을 위해 Open API 및 웹서비스 호출을 이용한 별도의 상태 변수들과 기본 동작들이 추가 구현되었고, 사용자의 편의성을 고려하여 음성 중심의 상호작용을 지원하는 새로운 사용자 인터페이스가 개발되었다. (그림 5)는 개발된 Smart Reservation의 실행 화면 중 일부이다. 현재 Smart Reservation의 구현은 서울 인사동 일대의 식당 정보만을 이용하는 한계점을 가지고 있음에도 불구하고, 몇 가지 시연을 통해 다양한 모바일 퍼스널 어시스턴트를 매우 효과적으로 개발할 수 있는 지원도구로서, 스마트 스크립트 시스템의 유용성을 보여주기에 충분하였다. 네트워크의 상태에 영향을 받기는 하지만, 현재의 서비스 응답 속도도 사용자들이 충분히 만족할 만한 수준을 보여주었다.

### 6. 실험

스마트 스크립트 시스템의 성능을 평가하기 위해 <표 1>과 같이 서로 다른 유형의 스크립트들을 작성하여, 실행 시간을 측정하여 보았다. 실험에 사용된 스크립트들은 각각 명령-주도, 사건-주도, 목표-주도 스크립트들이며, 이들은 총 4개, 4개, 7개의 실행 동작들을 포함하였다.

세 가지 유형의 스크립트에 대한 실험결과, 명령-주도 스크립트가 약 7.3초의 실행시간이 소요되어 가장 빨랐고, 사건-주

<표 1> 스크립트 실행 실험 결과

| 스크립트     | 유형    | 동작 수 | 실행시간(초) |
|----------|-------|------|---------|
| script-1 | 명령-주도 | 4    | 7.3     |
| script-2 | 사건-주도 | 4    | 13.1    |
| script-3 | 목표-주도 | 7    | 27.4    |

도 및 목표-주도 스크립트들은 각각 13.1초, 27.4초의 실행시간이 소요되었다. 실행시간에 영향을 주는 다른 많은 요인들도 있을 수 있으나, 사건-주도 및 목표-주도 스크립트들은 작업 목표와 환경변화에 적합한 스크립트 선정을 위한 추론 계산 노력이 더 필요하기 때문에 좀더 많은 실행시간을 소모한 것으로 판단된다. 2 번의 사용자 음성입력과 5 번의 원격 웹서비스 호출 및 Open API호출 동작을 포함한 목표-주도 스크립트의 실행이 약 30 초 미만에 완료된 것은 스마트 스크립트 시스템의 효율성과 성능을 보여주기에 충분하였다.

### 7. 결론

본 논문에서는 모바일 퍼스널 어시스턴트의 지능 행위를 구현하기 위한 스마트 스크립트 시스템의 설계와 구현에 대해 소개하였다. 스마트 스크립트 시스템은 모바일 컴퓨팅 환경 변화에 매우 빠르게 반응할 수 있으며, 실행 중인 스크립트들에 대한 실시간 감시와 제어 기능을 사용자에게 제공함으로써 모바일 퍼스널 어시스턴트에게 요구되는 사용자와의 높은 상호작용성을 얻을 수 있다.

### 참고 문헌

- [1] T. Gruber, "Siri: A Virtual Personal Assistant", Keynote Presentation at the Semantic Technology Conference (SemTech-09), 2009.
- [2] D. Morley and K. Myers, "The SPARK Agent Framework", Proc. of the AAMAS-04, pp.712-719, July, 2004.
- [3] F. Ingrand and F. Py, Proc. of the 4th Workshop on Planning and Plan Execution for Real-World Systems, ICAPS-09, 2009.
- [4] N. M. Gregory, et al, "IDEA: Planning at the Core of Autonomous Agents," Proc. of AAI-01, 2001.



#### 김인철

e-mail : kic@kyonggi.ac.kr  
 1985년 서울대학교 수학과(학사)  
 1987년 서울대학교 전산학과(이학석사)  
 1995년 서울대학교 전산학과(이학박사)  
 1996년~현 재 경기대학교 컴퓨터과학 교수  
 관심분야: 자동계획, 기계학습, 지능로봇



#### 오휘경

e-mail : ohkv770@kyonggi.ac.kr  
 2011년 경기대학교 컴퓨터과학과(학사)  
 2011년~현 재 경기대학교 컴퓨터과학과 석사과정  
 관심분야: 자동계획, 기계학습, 모바일지능