

# uFlow: 홈 서비스로봇을 위한 태스크 모델링 아키텍처

\*김성훈, 이승익, 정승욱, 장철수, 김종배  
한국전자통신연구원 로봇S/W아키텍처연구팀  
e-mail : {saint, the\_silee, swjung, jangcs, jjkim}@etri.re.kr

## uFlow: Robot Task Modeling Architecture for Home Service Robot

\*Sung-hoon Kim, Seung-ik Lee, Seung-wook Jung, Chul-soo Jang,  
Joong-Bae Kim  
Robot Software Architecture Research Team  
ETRI

### Abstract

This paper proposes a task modeling architecture - *uFlow* - for home service robots that facilitates the developmental process and let developers implement their services with ease and efficiency.

### I. 서론

최근 확장 발전되고 있는 지능 로봇의 S/W 아키텍처는 Reactive[1]와 Deliberative 방법으로 분류할 수 있다. 최근에는 이 두 방법을 혼합한 형태인 하이브리드 구조를 지향함으로써 두 방법의 단점을 상호 보완할 수 있도록 연구되고 있다. 하이브리드 아키텍처의 가장 큰 특징은 외부 환경 변화에 즉각적으로 반응 할 수 있을 뿐만 아니라 목표 지향적으로 로봇의 행동을 계획할 수 있는 점이다. 그러나 이러한 하이브리드 아키텍처는 다음과 같은 두 가지 관점에서 홈서비스 로봇에는 적합하지 않은 단점이 있다[2].

첫 번째는 아키텍처의 태스크 모델이 자율 주행에 초점을 맞추고 있어서 정보 서비스 태스크를 포함하는 홈서비스 로봇에 사용하기에 부족한 점이 있다. 두 번째는 대부분의 하이브리드 아키텍처에서 Deliberative 모듈들이 전통적인 인공지능 분야의 심볼 기반 태스크 계획 방법을 사용하고 있다는 점이다. 이러한 심볼 기반의 태스크 계획 방법은 로봇의 주변 환경을 정확하게

모델링하고 인지할 수 있는 방법이 선행되어야 하지만, 현재 기술만으로 로봇의 환경을 정확히 모델링할 수 없는 문제점이 있다.

본 논문에서는 홈서비스 로봇의 다양한 요구 사항들을 만족할 수 있도록 태스크 모델을 제안하고자 한다. 본 논문의 태스크 모델은 홈서비스 로봇이 정보 서비스를 포함하는 다양한 형태의 서비스를 사용자에게 제공할 수 있을 뿐만 아니라, 제한적인 환경 모델링 기법을 사용함으로써 태스크 계획 단계에서의 환경 모델에 대한 불확실성을 부분적으로 극복할 수 있는 장점을 가지고 있다.

### II. 본론

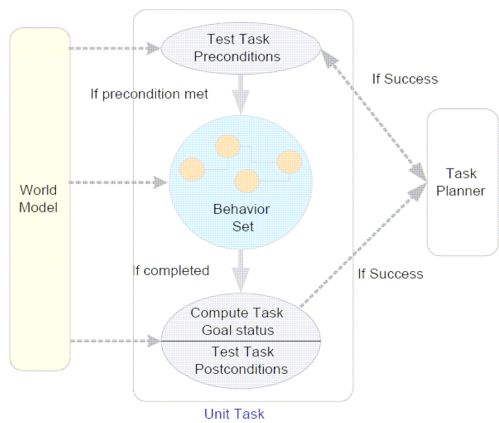
#### 2.1 태스크 요구사항 분석

홈서비스 로봇을 기능적인 측면과 개발의 용이성 측면에서 분석한 태스크 모델의 요구사항은 [표1]과 같다. [표1]의 요구사항을 만족하기 위해서 본 논문에서는 <그림1>과 같이 uFlow로 명명된 모델을 제안한다. uFlow 모델은 크게 월드 모델, 단위 태스크 모델, 태스크 계획자등 세 가지 컴포넌트로 구성된다. 월드 모델은 로봇의 내/외부 상태를 저장하는 메모리 공간이며, 메모리에 저장된 로봇의 상태들은 일정 조건에 따라 태스크 수행여부를 결정한다. 단위 태스크 모델은 태스크 실행 조건을 검사하는 부분과 수행할 행위들의 집합, 그리고 태스크 수행 후에 수행 결과를 검사하는 부분으로 구성되어 있다.

[표1] 홈 서비스 로봇 태스크 요구사항

Requirement	Details	Required in	Description
Synchronization	Starting	EX 1	Two or more tasks start simultaneously. For example, news notification and playing music start at the same time
	Ending	EX 1	Two or more tasks finish simultaneously. For example, playing music stops when news notification finishes
	Sequential	EX 1	Two or more tasks run in designated sequential order. After awaking dad, for example, the robot then read news to him
Repetition	Timed	EX 5	A task runs repeatedly for a given time
	Counted	EX 3	A task runs repeatedly for a given number of counts
State management	-	ALL	Task's execution results should be notified
Triggering	Event	EX 2, 4, 5	Task's execution is triggered by events (external or internal)
	Time	EX 1	Triggered by time
	Command	EX 3	Triggered by user command
Scheduling	Priority	EX 2	Triggered tasks are dynamically scheduled according to their priorities. For example, the robot immediately goes to the charging station if its battery level goes below the critical level, even if the robot is monitoring the house at that time
Composition	-	EX 1	Several tasks are combined into one big task. For example, we can combine all the tasks from EX 1 to EX 5 to compose, say, a <i>daily activity task</i> .
Exception handling	-	EX 3, 4	When an exception occurs, it should be handled properly

심볼 기반 태스크 계획을 사용하는 기존의 하이브리드 구조와 달리 uFlow에서는 상기 세 가지 구성요소에 따라 로봇이 수행할 태스크를 미리 작성하게 된다. 또한 uFlow에서는 월드 모델의 정보를 태스크의 시작과 종료 시에 조건 검사에만 부분적으로 활용한다. 따라서 환경 정보에 따라 동적으로 태스크를 계획하는 하이브리드 구조보다 환경 변화에 대한 강건성을 확보할 수 있다. <그림1>에서 태스크 계획자는 병렬 또는 순차 실행되는 각 태스크들의 실행 순서를 조정하는 조정자의 역할을 수행한다.

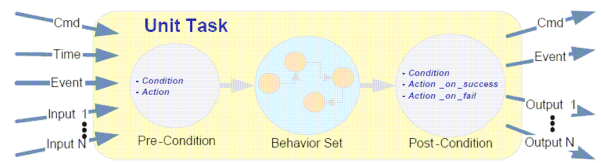


<그림 1> uFlow 모델의 상호작용

## 2.2 uFlow 태스크 모델

<그림1>의 태스크 모델을 자세히 표현하면 <그림 2>와 같다. uFlow 태스크 모델은 입출력을 위한 포트를 기반으로 하고 있다. 입력 포트는 태스크의 실행을 제어하는 제어(Control) 포트와 태스크 내부로 데이터를 전달하는 정보(Information) 포트 구성된다. 출력 포트는 입력 포트와 마찬가지로 태스크가 완료되었을 때나 실행 도중 외부에 생성된 데이터를 다른 태스크

에 전달하기 위한 정보 포트와 다른 태스크를 실행하기 위한 제어 포트 구성된다.



<그림 2> 포트 기반 uFlow 태스크 모델

uFlow 태스크들의 실행 순서와 상호작용 관계는 uFlow 모델의 태스크 플래너에 의해서 결정된다. 상기 모델의 행위 집합은 개발자에 의해서 미리 주어지는 것을 가정하며, uFlow 태스크 모델에서는 주어진 행위를 조합하여 상호 실행 순서관계만을 결정한다.

## III. 실험 결과 및 결론

본 논문에서는 제안하는 태스크 모델은 ETRI의 연구용 로봇 플랫폼인 WeverR2와 테스트 베드에서 일상의 태스크를 통해 모델의 적절성을 실험하였다.

본 논문에서 제안하는 태스크 모델은 월드 모델을 사용한 심볼 기반의 태스크를 계획하지 않는다. 결과적으로 제안하는 시스템은 구조가 간단하며, 태스크 계획 단계에서의 월드 모델 불확실성을 부분적으로 극복할 수 있는 장점을 갖는다.

### 참고문헌

- [1] J. K. Rosenblatt and D. W. Payton, "A fine-grained alternative to the subsumption architecture for mobile robot control," in Proc. *IEEE/INNS International Joint Conference on Neural Networks*, vol. 2, Nagoya, Japan, 1989, pp. 317 - 23.
- [2] Y.-G. Ha, J.-C. Sohn, Y.-J. Cho, and H. Y., "Towards ubiquitous robotic companion: Design and implementation of ubiquitous robotic service framework," *ETRI Journal*, vol. 27, no. 6, pp. 666 - 76, Dec. 2005.