

# 로봇 서비스 개발 및 실행을 지원하는 개발도구

## A Development Tool for Creating and Executing Robot Services

조용성\*, 최종선\*, 최재영\*\*

Yongseong Cho\*, Jongsun Choi\*, Jaeyoung Choi\*\*

### Abstract

There are many actively researched works on context-aware workflows for developing intelligent robot services. It is possible to describe not only the definition of robot services but also the contextual information of user's surroundings which can be used as the transition conditions for executing those robot services. Therefore it is required to prepare a context-aware workflow document to support the intelligent robot services.

In this paper, we propose a development tool which enables robot service developers to describe context-aware workflow documents and execute them. The tool users can write robot service workflows easily by using intuitive GUI of the tool. In the experiment, we showed processes of writing robot service workflows in scenario format, then automatically documenting and executing them.

### 요약

최근 지능형 로봇 서비스를 개발하기 위한 상황인지 기반의 워크플로우가 연구되고 있다. 이 연구에서는 로봇 서비스의 정의뿐만 아니라 각 로봇 서비스를 실행하기 위한 전이 조건인 사용자 주변의 상황 정보를 기술할 수 있다. 이러한 지능형 로봇 서비스를 지원하기 위해서는 먼저 상황인지 기반의 워크플로우 문서를 작성해야 한다.

이에 본 논문에서는 로봇 서비스 개발자가 손쉽게 상황인지 기반의 워크플로우 문서를 작성하고 실행할 수 있도록 지원하는 개발도구를 제안한다. 제안하는 개발 도구는 직관적인 GUI 형태로 로봇 서비스 워크플로우를 작성할 수 있다. 본 논문의 실험에서는 제안하는 개발 도구를 이용하여 시나리오 형태의 로봇 서비스 워크플로우를 작성한 후, 이를 자동으로 문서화하고 실행하는 과정을 보인다.

*Key words* : Robot Service, Context-aware Workflow, Development Tool, Documents Generation, Robot Service Execution

---

\* School of Computer Science and Engineering,  
Soongsil University

★ Corresponding author  
choi@ssu.ac.kr, +82-2-820-0684

※ This research was supported by the MSIP (Ministry of Science, ICT & Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (NIPA-2013-H0301-13-2006) supervised by the NIPA (National IT Industry Promotion Agency)

Manuscript received. Aug. 20, 2013; revised Sep. 5, 2013; accepted Sep. 5, 2013.

## 1. 서론

최근 로봇 산업은 자동화된 서비스를 요구하는 흐름에 맞춰 많은 변화를 겪고 있다[1][2][3]. 로봇 산업이 본격적으로 부흥을 일으킨 산업화 시대에는 단순한 작업을 반복하던 로봇이 주류였지만, 최근에는 사용자가 필요로 하는 것을 자동으로 처리해주는 로봇이 필요하다. 이러한 로봇을 일반적으로 지능형 로봇 또는 인텔리전트 로봇으로 부르며[4], 현재 이것을 개발하기 위한 많은 방법들이 존재한다.

인텔리전트 로봇을 개발하기 위해 사용되는 주된 방법은 플랫폼 기반의 개발 방법으로, 세계 각지에서 많은 연구가 이루어지고 있다[5][6][7]. 플랫폼 기반의 개발 방법은 일반적인 컴퓨터에서 사용되는 운영체제(OS)를 사용하는 것과 같다. 일반 어플리케이션이 OS가 제공해 주는 API를 사용해 개발하는 것과 같이 플랫폼이 제공하는 API를 사용해 로봇을 제어할 수 있다. 또 다른 개발 방법으로 컴포넌트 기반의 개발 방법이 있다[8][9][10]. 컴포넌트 기반의 개발 방법은 주로 로봇 제작회사에서 사용하며, 고정된 로봇의 디바이스를 제어하는 소프트웨어를 제작하여 결합하는 방법을 사용한다.

이 외에도 Client/Server(CS) 기반의 개발 방법이 있다. CS 기반의 개발 방법은 고성능 서버 컴퓨터가 네트워크를 통해 로봇과 통신하며 제어하는 방법으로 로봇의 제한적인 연산 능력을 고성능의 컴퓨터를 사용해 향상시키기 위한 목적으로 사용된다.

앞에서 설명한 개발 방법은 인텔리전트 로봇을 개발하기 위한 보편적인 방법으로 기존의 소프트웨어를 개발하는 방법과 큰 차이가 없다. 다만 현실세계의 다양한 환경에 적합한 로봇을 만들기 위해서는 고수준의 프로그래밍 기술이 필요하며, 변화하는 환경에 따라가기 위해 주기적으로 새로운 로봇 소프트웨어를 개발해야 할 필요가 있다.

이러한 점을 개선하기 위해 최근 웹 서비스와 상황인지 기반의 워크플로우[11]를 사용한 로봇 서비스 개발 방법이 연구되고 있다. 이 연구는 로봇이 제공하는 기능을 웹 서비스로 배포하고, 배포된 웹 서비스를 사용해 상황인지 기반의 워크플로우 문서로 기술하여 흐름을 따라 로봇의 기능을 호출하는 방법에 대한 것이다. 이 방법은 워크플로우 문서만 변경하여 새로운 환경에 적합한 로봇을 만들 수 있는 장점이 있다. 그러나 기존의 상황인지 워크플로우 개발 도구는 로봇 서비스를 직관적인 GUI로 표현하고 자동으로 문서화하는 것에 중점을 두고, 워크플로우 문서를 실행하는 방법을 지원하지 않았기 때문에 일반 사용자들에게 상황인지 워크플로우 기반의 로봇 서비스를 제공하기 위해서 문서를 생성하고 실행하기 위한 과정을 분리해서 해야만 했다.

이에 본 논문에서는 유비쿼터스 환경에서 사용될 로봇 서비스를 문서로 자동 생성하여 워크플로우 엔진을 통해 실행시킬 수 있는 로봇 서비스 개발 도구를 제안한다. 제안하는 개발 도구는 로봇 서비스 제공을 위한 워크플로우 문서의 생성과 실행이 분리되어있던 기존 개발 도구의 문제점을 해결하기 위해 3가지의 모듈로 구성된다. 첫 번째 모듈은 직

관적인 GUI 기반의 워크플로우 작성 모듈이다. 로봇 서비스 개발자는 드래그 앤 드랍 기능을 사용하여 손쉽게 로봇 서비스 제공을 위한 워크플로우 시나리오를 작성할 수 있다. 두 번째 모듈은 GUI 기반으로 작성된 워크플로우를 CAWL[11] 문서로 자동 생성하는 모듈이다. 사용자에게 상황인지 기반의 로봇 서비스를 제공하기 위해서는 워크플로우 실행 엔진에 전달될 CAWL 문서가 필요하다. 이 모듈은 로봇 서비스 개발자가 작성한 GUI 기반의 워크플로우 시나리오에서 실행에 필요한 CAWL 문서를 자동 생성할 수 있다. 세 번째 모듈은 워크플로우 실행 엔진이다. 이 모듈은 CAWL 문서를 사용해 정의된 워크플로우 흐름에 따라 로봇을 제어할 수 있는 웹 서비스를 호출하여 사용자에게 로봇 서비스를 제공하는 역할을 한다.

본 논문에서는 제안하는 개발도구의 효용성을 검증하기 위해 상황인지 기반 워크플로우의 개발 및 워크플로우 문서의 자동 생성, 생성된 워크플로우 문서와 실행 엔진간의 연동 과정을 보여준다.

## II. 관련 연구

유비쿼터스 본 절에서는 로봇 서비스를 제공하기 위한 기존의 개발 방법들과 여러 개발 도구, 그리고 상황인지 기반의 워크플로우에 대해서 살펴본다.

ROS (Robot Operating System)[6]는 미국에서 개발된 로봇용 오픈소스 운영체제이다. 일반적인 운영체제와 같이 하드웨어의 추상화, 저 수준의 기기 제어, 그리고 API를 제공한다. OPRoS (Open Platform for Robotic Services)[8]는 한국에서 개발된 컴포넌트 기반의 로봇 소프트웨어 아키텍처이다. 기존의 아키텍처들을 아우를 수 있는 기반을 제공하여 재사용성, 이식성, 확장성을 보장한다. RT-미들웨어 (Robotics Technology Middleware)[7]는 일본에서 연구한 플랫폼으로 각각의 디바이스를 제어하는 소프트웨어를 제작하고 이를 결합하여 계층적으로 사용할 수 있는 플랫폼이다. OROCOS (Open Robot Control Software)[9]는 유럽에서 연구하는 플랫폼으로 각 디바이스를 제어하는 소프트웨어를 특정 네트워크 어플리케이션으로 조합하여 사용할 수 있다. MSRS (Microsoft Robotics Studio)[10]는 마이크로소프트사에서 개발한 컴포넌트 기반의 로봇 개발 도구이다. 컴포넌트의 재사용성과 손쉬운 개발을 장점으로 갖고 있다.

Context4BPPEL[12]은 일반적인 워크플로우 문서에

상황인지 정보의 기술을 위해 BPEL[13]을 확장한 것으로 실행 환경을 위한 아키텍처가 제공된다. 다만 아직까지 실행 환경의 구축이 완료되지 않아 워크플로우 엔진에 대한 연구가 더 필요하다.

CAWL (Context-Aware Workflow Language) [11]은 상황인지 기반의 워크플로우를 기술할 수 있는 언어이다. CAWL은 웹 서비스를 바탕으로 워크플로우를 구성하고, 실행의 전이 조건으로 상황인지 정보를 기술할 수 있는 언어이다. 본 논문에서는 로봇 서비스 제공을 위해 제안하는 개발도구를 이용하여 CAWL 문서를 자동 생성한다.

GridFlow[14]는 영국 Warwick 대학에서 개발한 그리드 환경에서의 워크플로우 개발환경이다. GridFlow는 파서 및 시뮬레이터, 스케줄링, 실행 등의 다양한 기능을 제공해 주는 통합 환경이다. 손쉬운 그래픽 인터페이스를 사용하여 손쉽게 접근이 가능한 것이 장점이다. 다만 각 서비스에 관련된 상황인지 정보의 기술이 다소 복잡하다. LTSA-BPEL4WS는 웹 서비스 기반 워크플로우 표준 언어인 BPEL4WS[15]로 작성된 프로세스의 모델을 생성하기 위한 플러그인이다. 워크플로우 흐름에 대한 그래픽을 직관적으로 표현할 수 있다.

### III. 제안하는 로봇 서비스 개발 도구

유비쿼터스 본 절에서는 로봇 서비스를 제공하기 위한 기존의 개발 방법들과 상황인지 기반의 워크플로우, 그리고 개발 도구들에 대해 살펴본다.

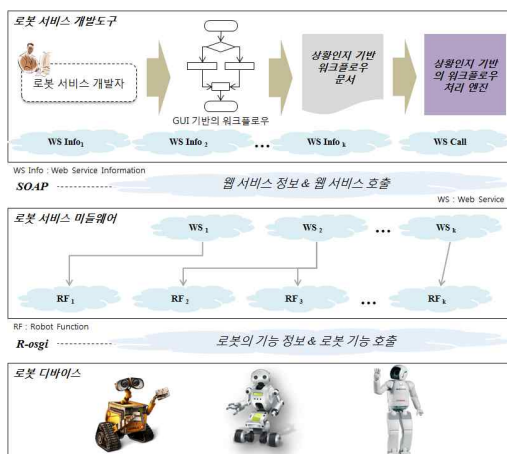


Fig. 1. System architecture of context-aware based workflow

그림 1. 상황인지 기반 워크플로우를 활용한 시스템 개요

### 1. 상황인지 기반의 워크플로우를 활용한 로봇 서비스 개요

본 절에서는 상황인지 기반의 워크플로우를 활용한 시스템의 개요와 본 논문에서 제안하는 개발도구의 역할에 대해 알아본다.

그림 1은 제안하는 개발 도구를 사용하여 로봇 서비스를 제공할 수 있는 상황인지 기반의 워크플로우 시스템의 전체 개요도이다. 시스템의 가장 아래는 현실 세계에서 구동되고 있는 로봇 디바이스를 나타낸다. 각각의 로봇 디바이스는 제공할 수 있는 기능을 R-ODGI 통신 인터페이스를 사용해 외부로 노출시키고 있다. 외부로 노출된 로봇의 기능은 로봇 서비스 미들웨어에 의해서 취합된다. 로봇 서비스 미들웨어는 크게 두 가지의 주된 기능을 구현하고 있다. 첫째는 실제 로봇 디바이스의 기능을 취합하는 역할이고, 둘째는 취합한 로봇 디바이스의 기능을 범용적인 통신 인터페이스인 웹 서비스를 사용해 외부로 노출시키는 것이다. 그림 1의 로봇 서비스 미들웨어 부분에서 보이듯이 외부로 노출된 웹 서비스를 호출하면 실제 로봇의 기능을 사용할 수 있도록 구성되어 있다.

그림 1의 상단부에는 본 논문에서 제안하는 개발도구가 나타난다. 제안하는 개발 도구는 로봇 서비스 미들웨어가 외부로 노출하고 있는 웹 서비스의 정보를 읽어온다. 읽어온 웹 서비스의 리스트는 워크플로우를 구성하는 노드의 정보로 사용된다. 상황인지 기반의 워크플로우를 활용한 로봇 서비스 개발자는 이것들을 사용해 GUI 기반으로 워크플로우를 작성할 수 있다. 로봇 서비스 개발자가 작성한 GUI 기반의 워크플로우는 개발 도구가 제공하는 기능을 사용해 로봇 서비스 실행을 위한 문서로 자동 변환된다. 변환된 문서는 상황인지 기반의 워크플로우 실행엔진에서 사용되며, 문서에 정의된 웹 서비스를 워크플로우의 흐름대로 호출하여 로봇 서비스를 제공한다.

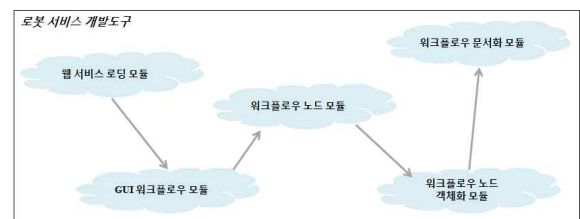


Fig. 2. The proposed structure of the robot service development tools

그림 2. 제안하는 로봇 서비스 개발 도구의 구조

2. 제안하는 개발 도구의 개요 및 구성

본 절에서는 본 논문에서 제안하는 개발 도구의 구성과 주요한 구성 요소에 대해 알아본다.

그림 2는 제안하는 로봇 서비스 개발 도구의 구조도이다. 개발 도구는 크게 가지의 구성 요소를 가지고 있다.

먼저 그림 1에서 확인한 로봇 서비스 미들웨어가 제공하는 웹 서비스의 리스트를 받아오는 모듈이다. 이 모듈은 로봇 서비스 미들웨어에서 웹 서비스의 리스트를 받아와서 GUI 기반의 목록을 생성한다. 웹 서비스의 목록은 로봇 서비스 개발자가 작성하는 워크플로우의 노드가 되며, 각 노드는 웹 서비스의 정보를 보관하게 된다. 또 다른 구성 요소는 GUI 기반의 워크플로우 작성 모듈이다. 이 모듈은 사용자가 직관적으로 서비스 워크플로우를 작성할 수 있도록 도와준다. 그리고 작성된 워크플로우를 객체화하여 메모리에 저장할 수 있도록 한다. 세 번째 구성요소인 문서 생성 모듈은 작성된 워크플로우에 부합하는 상황인지 기반의 워크플로우 문서를 생성하는 역할을 한다. GUI 기반의 워크플로우 작성 모듈에서 생성된 객체들을 활용한 객체 모델링 방법을 사용한다. 네 번째 구성요소는 상황인지 기반의 워크플로우 실행엔진이다. 실행엔진은 문서 생성 모듈에서 생성한 상황인지 기반의 워크플로우 문서를 사용하여 사용자에게 로봇 서비스를 제공할 수 있는 역할을 한다.

3. 상황인지 워크플로우 문서의 자동 생성 방법

3.2 절에서 언급했듯이 상황인지 기반의 워크플로우를 활용해 로봇 서비스를 제공하기 위해서는 상황인지 기반의 워크플로우 문서가 필요하다. 본 절에서는 GUI로 표현된 워크플로우로부터 문서를 자동 생성하는 객체 모델링 방법에 대해 살펴본다.

그림 3은 상황인지 기반의 워크플로우 문서를 자동 생성하는 객체 모델링 방법에 대한 개요도이다. 본 절에서 알아볼 객체 모델링 방법을 이해하기 위해 먼저 상황인지 기반 상황인지 기반의 워크플로우 문서를 살펴본다. 상황인지 기반의 워크플로우 문서는 XML을 기반으로 한다. XML을 기반으로 하기 때문에 문서 내부는 정형화된 태그들로 구성되어 있다. 각각의 태그들은 다양한 속성 값을 포함할 수 있으며, 태그들 간에는 종속성을 가지고 있다.

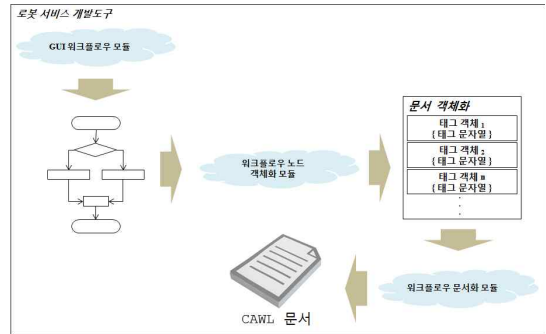


Fig. 3. Automatic generation method of context-aware workflow documents

그림 3. 상황인지 기반 워크플로우 문서의 자동생성 방법

객체 모델링 방법은 기존에 구조화된 문서를 처리하기 위해 사용되고 있던 DOM (Document Object Model)과 흡사하다. 이 방법은 문서 내부에 정형화된 태그들을 하나의 객체로 표현하는 방법이다. 하나의 태그는 객체지향에서 사용되는 클래스로서 표현되며, 각각의 태그가 사용하는 속성들은 클래스 내부의 멤버로 선언된다. 태그들 간에 형성되는 종속 관계는 클래스 간의 상속을 사용해 표현한다.

이 방법은 여러 장점을 가지고 있다. 먼저 워크플로우의 잦은 변경에 유리하다. 로봇 서비스 개발자는 워크플로우를 작성하고 수시로 변경할 수 있다. 객체 모델링 방법은 메모리 내부에 값을 유지하고 있기 때문에 빠른 수정이 가능하다. 또한 데이터의 저장과 불러오기가 용이하다. 그 이유는 객체 모델링 방법은 클래스간의 상속을 사용하기 때문에 다형성을 응용한 최상위 부모 클래스의 타입으로 저장하고 불러오기가 가능하기 때문이다. 마지막으로 문서 구조의 변경에 취약하지 않다. 문서의 태그가 변경되거나 삭제가 되더라도 클래스의 구조만 변경함으로써 대처가 가능하기 때문이다.

본 논문에서 제안하는 개발 도구는 이와 같이 다양한 장점들 가지는 객체 모델링 방법을 사용해 로봇 서비스 개발자가 작성한 GUI 기반의 워크플로우를 상황인지 기반의 워크플로우 문서로 자동 생성한다.

4. 상황인지 워크플로우 문서를 사용한 실행 과정

3.3에서 상황인지 기반의 워크플로우 문서를 자동 생성하는 방법에 대해 알아보았다. 본 절에서는 3.3절에서 생성된 문서를 사용해 로봇 서비스를 실행하는 상황인지 기반의 워크플로우 실행엔진에 대해 살펴본다.

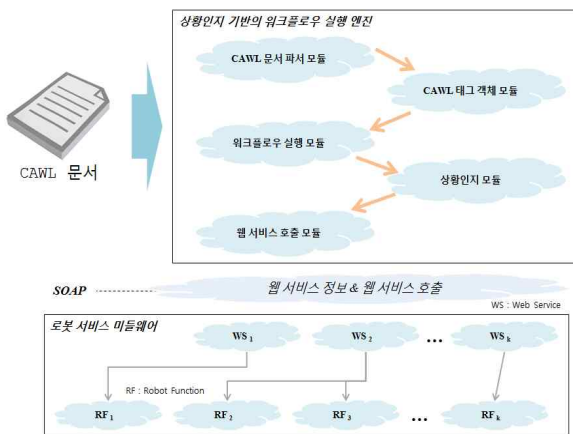


Fig. 4. Context-aware based workflow document and execution engine

그림 4. 상황인지 기반의 워크플로우 문서와 실행 엔진

그림 4는 제안하는 개발 도구에서 상황인지 기반의 워크플로우 문서를 실행 엔진에 전달하여 로봇 서비스를 제공하는 개요도이다.

상황인지 기반의 워크플로우 실행 엔진은 3.3 절에서 생성된 상황인지 기반의 워크플로우 문서를 실행의 매개체로 사용한다. 상황인지 기반의 워크플로우 실행 엔진은 내부의 파서를 사용해 워크플로우 문서에서 실행을 위한 정보를 추출한다. 워크플로우 실행엔진은 추출된 정보를 바탕으로 사용자 주변의 상황 정보를 인지하여 워크플로우 흐름에 따라 웹 서비스를 호출한다. 웹 서비스는 3.1 절에서 살펴본 로봇 서비스 미들웨어를 통해 로봇의 기능을 호출하게 된다. 호출된 로봇 기능은 실제 로봇에 전달되어 최종 사용자에게 로봇 서비스를 제공하게 된다.

그림 4에서 나타낸 상황인지 기반의 워크플로우 실행 엔진은 독립적인 모듈로 구성되어 있다. 제안하는 개발도구에 종속적이지 않으며, 별도의 실행 파일로도 구동이 가능하다. 이런 특징으로 인해 상황인지 기반의 워크플로우 실행 엔진은 워크플로우 문서만 있다면 어디에서도 실행이 가능하다.

### IV 실험

본 절에서는 제안하는 개발도구를 사용해 로봇 서비스를 개발하는 과정에 대해 살펴본다. 본 절의 세부 절에서는 로봇 서비스가 제공될 시나리오를 구성하고 이를 기반으로 상황인지 기반의 워크플로우를 작성하는 방법, 문서화하여 실행하는 전반적인 과정에 대해 살펴본다.

### 2. 로봇 서비스 제공을 위한 시나리오

본 절에서는 실험에서 사용되는 가상의 시나리오를 구성한다. 구성된 시나리오는 본 논문에서 제안하는 개발 도구를 사용해 워크플로우로 표현된다.

그림 5는 실험에서 사용할 로봇 서비스의 시나리오를 나타내며, 로봇 서비스의 구동을 확인하기 위한 목적으로 간단한 시나리오를 사용한다.

로봇 서비스를 제공할 시나리오의 내용을 살펴보면 다음과 같다. 회의실에 사람이 들어온다. 회의실에 사람이 들어온 것을 인지하면 회의실의 불을 켜고 의자를 제공한다.

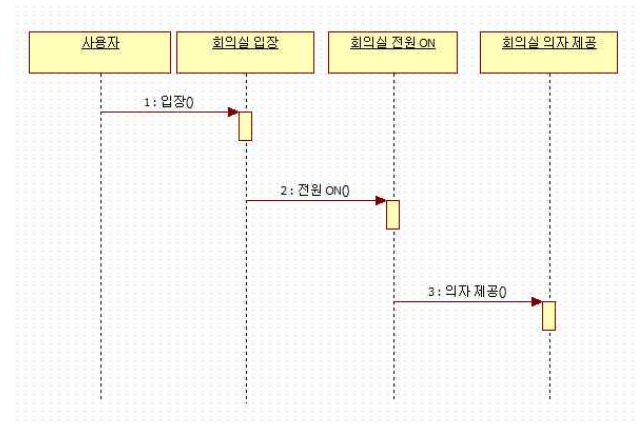


Fig. 5. Robot service scenario

그림 5. 로봇 서비스 시나리오

### 2. GUI 기반의 워크플로우 작성

본 절에서는 4.1 절에서 만든 시나리오를 기반으로 개발 도구에서 워크플로우를 작성하는 방법을 설명한다.

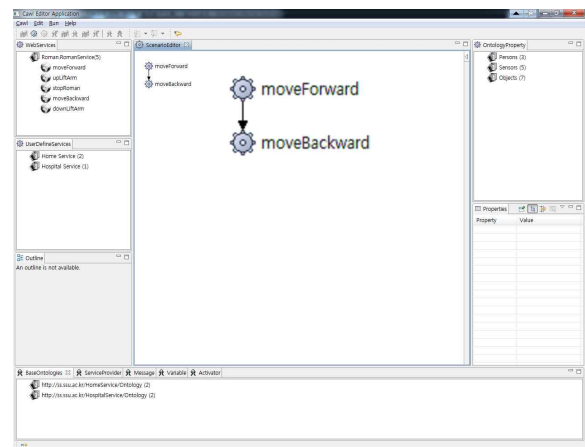


Fig. 6. GUI-based workflow

그림 6. GUI 기반의 워크플로우

그림 6은 4.1절에서 만든 시나리오를 바탕으로 개발 도구에서 워크플로우를 작성한 모습이다. 먼저 좌측의 메뉴는 로봇의 기능을 호출할 수 있는 웹 서비스의 목록을 나타낸다. 이는 3.1 절에서 확인한 로봇 서비스 미들웨어에서 제공받은 목록이다. 로봇 서비스 개발자는 이를 사용해 워크플로우를 작성할 수 있다. 그림 6의 가운데 부분은 로봇 서비스 개발자가 작성한 워크플로우가 나타난다. 시나리오의 내용과 같이 회의실에 사람이 있는지를 확인하고, 확인된 이후 회의실의 불을 켜주는 기능과 의자를 제공하는 기능이 워크플로우에 정의되어 있는 것을 확인할 수 있다.

**2. 상황인지 기반의 워크플로우 문서 생성**

4.2 절에서 작성된 워크플로우는 상황인지 기반의 워크플로우 문서로 자동 생성된다. 본 절에서는 4.2 절에서 작성한 워크플로우에서 문서가 자동 생성되는 과정과 결과물을 살펴본다.

```

<CAWL name = "MTRoomService" version = "1.0.0.0" targetNamespace = "RobotServiceTest">
  <baseOntologies>
    <ontology namespace = "http://ss.ssu.ac.kr/Ontology" location = "http://ss.ssu.ac.kr/Ontology/CAWL" />
  </baseOntologies>
  <serviceProvider name = "ssCom" type = "Company">
    <locator type="static"/>
  </serviceProvider>
  <message name = "msg4MTRoomService">
    <part name = "personName" type = "Person" />
  </message>
  <message name = "msg4TimeInfo">
    <part name = "crtTime" type = "militaryTime" />
  </message>
  <variable name = "var4MTRoomService" type = "msg4MTRoomService">
    <initialize part = "personName"><from expression = "John" /></initialize>
  </variable>
  <activator name = "MTRoomServiceActivator">
    <variable name = "var4TimeInfo" type = "msg4TimeInfo">
      <initialize part = "crtTime"><from expression = "1340" /></initialize>
    </variable>
    <condition expression = "C1 and C2">
      <context>
        <rule contribute = "positive">
          <constraint name = "C1">

```

Fig. 7. Generated Please put the title of figure here  
그림 7. 상황인지 기반의 워크플로우 문서 생성

그림 7은 4.2 절에서 작성한 워크플로우가 상황인지 기반의 워크플로우 문서로 변환된 모습을 보여준다. 기본적으로 개발도구는 로봇 서비스 개발자가 워크플로우를 작성하면 자동으로 문서를 생성하게 된다. 생성된 문서는 파일 형태로 저장되며 그림 7에서 나타나듯이 개발 도구를 통해서 확인할 수도 있다.

**2. 상황인지 기반의 워크플로우 실행 엔진을 통한 로봇 서비스 제공**

상황인지 기반의 워크플로우 실행 엔진은 4.3절에서 생성된 상황인지 기반의 워크플로우 문서를 사용해 로봇 서비스를 실행할 수 있다. 본 절에서

는 실행엔진을 사용해 로봇 서비스가 실행되는 과정과 결과를 확인한다.

그림 8은 상황인지 기반의 워크플로우 실행엔진이 로봇 서비스 미들웨어의 웹 서비스를 호출한 모습이다. 4.2절에 있는 워크플로우의 실행 흐름대로 호출되는 것을 확인할 수 있다.

그림 9는 그림 8에서 확인한 워크플로우 실행 엔진의 웹 서비스 호출을 로봇 서비스 미들웨어에서 처리한 결과를 보여준다. 그림 9의 상단에서는 그림 8에서 확인한 워크플로우 실행 엔진의 처리 순서대로 웹 서비스가 호출되는 것을 확인할 수 있다. 그림 9의 하단에는 이에 해당하는 로봇 기능이 호출되어 실제 로봇이 동작한 것을 확인할 수 있다.

```

Somar Client [OSGi Framework] C:\Program Files\Java\jre7\bin\javaw.exe (2013-8-16 下午4:02:41)
[SOMAR Client] SOMAR Client has runned.
osgi>
"Framework is launched."

id      State      Bundle
0       ACTIVE    org.eclipse.osgi_3.8.0.v20120529-1548
5       ACTIVE    org.eclipse.osgi.services_3.3.100.v20120522-1822
19      ACTIVE    SOMAR_Client_1.0.0.qualifier
23      ACTIVE    RomanService_1.0.0.qualifier
25      ACTIVE    org.eclipse.equinox.console_1.0.0.v20120522-1841
26      ACTIVE    ch.ethz.sks.r.osgi.remote_1.0.0.RC4
27      ACTIVE    org.eclipse.osgi.util_3.2.300.v20120522-1822
28      ACTIVE    TemperatureService_1.0.0.qualifier
29      ACTIVE    org.apache.felix.gogo.runtime_0.8.0.v201108120515
30      ACTIVE    org.objectweb.asm_3.3.1.v201105211655
31      ACTIVE    RobotService_1.0.0.qualifier
32      ACTIVE    PickupService_1.0.0.qualifier
33      ACTIVE    SensorService_1.0.0.qualifier
34      ACTIVE    javax.servlet_3.0.0.v201112011816
35      ACTIVE    javax.xml_1.3.4.v201005080400
36      ACTIVE    org.apache.felix.gogo.shell_0.8.0.v201110170705

osgi> [Roman Service] moveToChair() is called.
[Roman Service] moveToSwitch() is called.
[Roman Service] turnOnLight() is called.
[Roman Service] turnOffLight() is called.
[Roman Service] provideChair() is called.
[Roman Service] holdTheChair() is called.
[Roman Service] dropTheChair() is called.

```

Fig. 8. The results from the processing of workflow execution engine

그림 8. 워크플로우 실행 엔진의 처리 결과

```

Somar Client [OSGi Framework] C:\Program Files\Java\jre7\bin\javaw.exe (2013-8-17 上午8:48:15)
[SOMAR Client] SOMAR Client has runned.
osgi> [Roman Service] moveToDoor() is called.
[Roman Service] moveToSwitch() is called.
[Roman Service] turnOnLight() is called.
[Roman Service] moveToChair() is called.
[Roman Service] provideChair() is called.

```

Fig. 9. The results of robot middleware services

그림 9. 로봇 서비스 미들웨어의 실행 결과

**V 결론**

본 논문에서는 로봇 서비스 개발자가 손쉽게 로봇 서비스의 개발 및 실행할 수 있도록 지원하는 개발 도구를 제안하였다. 제안하는 개발 도구는 상황인지 기반의 워크플로우를 활용한 로봇 서비스 시스템에 사용되었다. 제안하는 개발 도구는 로봇 서비스 개발자로 하여금 상황인지 기반의 워크플로우를 사용하는 로봇 서비스를 손쉽게 개발할 수 있

도록 GUI 기반의 작업을 지원한다. 그리고 GUI로 작성된 워크플로우를 활용하여 로봇 서비스 개발자가 수작업으로 작성하기 어려운 상황인지 기반의 워크플로우 문서를 자동으로 생성할 수 있다.

개발 도구를 통해 생성된 상황인지 기반의 워크플로우 문서는 워크플로우 실행 엔진을 통해 로봇 서비스를 실행할 수 있다. 워크플로우 실행 엔진은 개발 도구를 통해 생성된 워크플로우 문서를 사용하여 로봇 서비스 실행에 필요한 정보를 파악한다. 그리고 현재 사용자 주변의 상황을 인지하여 사용자 주변의 상황이 로봇 서비스를 실행해야 하는 조건이 되면 워크플로우 문서에 정의된 웹 서비스를 호출하여 실제 로봇을 제어할 수 있다.

## References

- [1] Yao Wang, Stephan Wegner, "Error resilience video coding techniques, Real-time video communication over unreliable networks," IEEE Signal Processing Magazine, Vol.17, pp.61-82, 2000
- [2] Alessandro Massaro, Fabrizio Spano, Paolo Cazzato, Carola La Tegola, Roberto Cingolani, and Athanassia Athanassiou, "Robot Tactile Sensing: Gold Nanocomposites As Highly Sensitive Real-Time Optical Pressure Sensors", Robotics & Automation Magazine, IEEE (Volume:20 , Issue: 2 )
- [3] Heiko Koch, Alexander König, Alexandra Weigl-Seitz, Karl Kleinmann, and Jozef Suchy, "Multisensor Contour Following With Vision, Force, and Acceleration Sensors for an Industrial Robot", IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, VOL. 62, NO. 2, FEBRUARY 2013
- [4] Guoqin Gao, Haiyan Zhou, Xuemei Niu, "An Intelligent Variable Spraying Decision-Making System Based on Fuzzy Neural Network for Greenhouse Mobile Robot", Intelligent Computing for Sustainable Energy and Environment Communications in Computer and Information Science Volume 355, 2013, pp 257-265
- [5] Julian Mason, Zachary Dodds, William D. Smart, "ROS at every level: using the robot operating system in CS 0, 1, 2, and beyond (abstract only)", Proceeding SIGCSE '13 Proceeding of the 44th ACM technical symposium on Computer science education Pages 757-757
- [6] André Dias, Jose Almeida, Alfredo Martins, Eduardo Silva, "Real-Time Visual Ground-Truth System for Indoor Robotic Applications", Pattern Recognition and Image Analysis Lecture Notes in Computer Science Volume 7887, 2013, pp 304-313
- [7] Sivabalan Arumugam, Ritesh Kumar Kalle, Anand R. Prasad, "Wireless Robotics: Opportunities and Challenges", Wireless Personal Communications June 2013, Volume 70, Issue 3, pp 1033-1058
- [8] Park, F.C., "Robotics in Korea [Regional]", Robotics & Automation Magazine, IEEE (Volume:20 , Issue: 1 )
- [9] Shuichi Nishio, Koji Kamei, Norihiro Hagita, "Ubiquitous Network Robot Platform for Realizing Integrated Robotic Applications", Intelligent Autonomous Systems 12 Advances in Intelligent Systems and Computing Volume 193, 2013, pp 477-484
- [10] Tanya Lee Ann Crenshaw, "Using Robots and Contract Learning to Teach Cyber-Physical Systems to Undergraduates", Education, IEEE Transactions on (Volume:56 , Issue: 1 )
- [11] Jongsun Choi, Yongyun Cho, Jaeyoung Choi, "The Design of a Context-Aware Workflow Language for Supporting Multiple Workflows", Journal of Korea Society for Internet Information Vol. 10, No.6, pp. 145-157, 2009
- [12] Angelica Nieto Lee, Jose Luis Martinez Lastra, "Enhancement of industrial monitoring systems by utilizing context awareness", 2013 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), San Diego
- [13] Yu Nan, Guangsheng He, Zhuo Zhao, Cheng Liu, "A QoS-Based Self Adaptive Control Strategy and Implementation for BPEL Process", Proceedings of the 2012 International Conference on Communication, Electronics and Automation Engineering Advances in Intelligent Systems and Computing Volume 181, 2013, pp 101-107
- [14] Pawel Czarnul, "Modeling, run-time optimization and execution of distributed workflow applications in the JEE-based BeesyCluster environment", The Journal of Supercomputing January 2013, Volume 63, Issue 1, pp 46-71

- [15] Adrian Rutle, Hao Wang, Wendy MacCaul, "A Formal Diagrammatic Approach to Compensable Workflow Modelling", Foundations of Health Information Engineering and Systems Lecture Notes in Computer Science Volume 7789, 2013, pp 194-212

---

**BIOGRAPHY**

---

**Yongseong Cho** (Member)



2013 : M.S. in School of Computer Science and Eng., Soongsil University, Seoul, Korea  
2013~Present : Ph.D. candidate, School of Computer Science and Eng. Soongsil University.

**Jongsun Choi** (Member)



2000: the B.S. in School of Computer Science and Eng. Soongsil University, Seoul, Korea  
2002: M.S. in School of Computer Science and Eng. Soongsil University  
2010: Ph.D. in School of Computer Science and Eng. Soongsil University

2013~present: Assistant Professor, School of Computer Science and Eng, Soongsil University

**Jaeyoung Choi** (Member)



1984: B.S. in Dept. of Control and Instrumentational Eng., Seoul National University  
1986: M.S. in Dept. of Electrical Eng., University of Southern California  
1991: Ph.D. in School of Electrical Engineering, Cornell University

1995~present: Professor, School of Computer Science and Engineering, Soongsil University.