

유비쿼터스 환경을 위한 웹 서비스 기반의 워크플로우 언어 설계

한주현^o, 김은희, 최재영
승실대학교 컴퓨터학부

{jhhan, ehkim}@ss.ssu.ac.kr choi@comp.ssu.ac.kr

Web Services-based workflow language design for ubiquitous environment

Joohyun Han^o, Eunhoe Kim, Jaeyoung Choi
School of Computing, Soongsil University

요 약

본 논문에서는 유비쿼터스 환경에서 발생하는 컨텍스트와 이를 이용하는 서비스간의 연관 관계를 온톨로지 기반의 기술 방식을 이용하여 효과적으로 표현할 수 있는 uWDL (Ubiquitous Workflow Definition Language)을 소개한다. uWDL은 유비쿼터스 환경의 사용자 특성을 이용하여 그에 맞는 서비스들의 관계를 워크플로우 형태로 기술할 수 있는 언어로, uWDL을 이용하여 서비스들 간의 연관 관계를 기술함으로써 작업의 통합 및 자동화를 이룰 수 있다. 이를 위해 가장 널리 사용되고 있는 웹 서비스(Web Services)의 WSDL을 바탕으로 uWDL의 서비스를 기술하고, 워크플로우에 의해 구성된 서비스들의 연관 관계는 컨텍스트 정보에 따라 동적으로 재구성되어 상황에 맞는 서비스를 제공하게 된다.

1. 서 론

워크플로우의 개념은 주로 비즈니스 프로세스에서 사용되었다. 비즈니스 프로세스란 일반적으로 기능적인 역할과 관계를 정의하는 조직 구조의 맥락에서 업무 목표나 정책적인 목적을 총괄적으로 실행하는 일련의 연관된 업무 절차이다 [1]. 다양한 비즈니스 프로세스의 효율성 향상에 기여했던 워크플로우 개념이 최근에는 그림 1과 같이 분산 컴퓨팅 시스템에 활발히 적용되고 있고, 나아가서는 그리드 컴퓨팅이나 유비쿼터스 컴퓨팅에도 워크플로우를 적용하려는 시도가 활발히 이루어지고 있다.

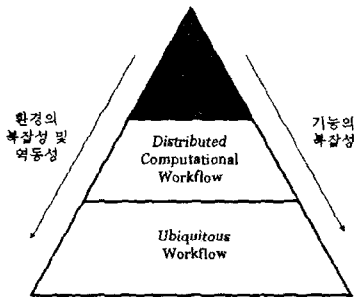


그림 1. 대상 환경 및 기능의 복잡성

그러나 기존의 비즈니스 프로세스에 적용되었던 워크플로우를 유비쿼터스 환경에 적용하기 위해서는 해결해야 할 여러 가지 문제점을 지니고 있다. 첫째로 비즈니스 프로세스는 정적인 특성을 지니는데 반해 분산 및 유비쿼터스 컴퓨팅 환경은 동시에 여러 워크플로우가 여러 대의 노드에 분산 수행되어 워크플로우 간에 자원 경쟁이 발생하기 때문에 보다 복잡하고 동적인 특성을 지닌다. 둘째로 작업이 수행되는 자원의 수와 종류가 사람의 이동에 따라 수시로 변하며 예측 불가능하다는 어려움이 있다. 따라서 이러한 특성을 처리할 수 있어야 하므로

순간순간 변화하는 동적인 환경에 적용할 수 있는 적응력있는 워크플로우가 필요하다.

본 논문에서는 유비쿼터스 환경에서 발생하는 모든 컨텍스트, 프로파일 및 이벤트와 이를 이용하는 서비스간의 연관 관계를 온톨로지 형태의 규칙 기반(rule-based) 기술 방식을 통하여 효과적으로 표현할 수 있는 uWDL (Ubiquitous Workflow Definition Language)을 소개한다. 이는 유비쿼터스 환경에서 사용자의 특성을 파악하고 그에 맞는 서비스들의 관계를 기술할 수 있는 언어로, uWDL을 이용하여 서비스들 간의 연관 관계를 워크플로우로 구성함으로써 작업의 통합 및 자동화를 이룰 수 있다. 이를 위해 가장 널리 사용되고 있는 웹 서비스(Web Services)의 WSDL을 이용하여 uWDL의 서비스를 기술하고, 워크플로우에 의해 구성된 서비스들의 연관 관계는 컨텍스트의 종류에 따라 동적으로 재구성되어 상황에 맞는 서비스를 제공하게 된다.

2. 관련 연구

2.1 유비쿼터스 컴퓨팅(Ubiquitous Computing)

유비쿼터스는 "언제 어디에든 존재한다"라는 의미의 라틴어로 1988년 처음으로 제록스 팰로 알토(Palo Alto) 연구소의 마크 와이저(Mark Weiser)에 의해 소개되었다. 그는 유비쿼터스 컴퓨팅을 "현실 세계의 어디서나 네트워크와 연결된 컴퓨터를 눈에 띄지 않는 형태로 언제 어디서나 사용 가능하며 사용자 상황에 따라 서비스가 변하는 환경 [2]"이라고 정의하고 이는 컴퓨팅 환경이 현실세계의 사물과 환경속으로 스며들어 일상생활에 통합되는 것 [3]을 의미한다.

2.2 워크플로우

WfMC에서는 워크플로우를 "전체적인 또는 부분적인 비즈니스 프로세스의 자동화를 의미하며, 이때 문서, 정보, 태스크가

한 사용자에서 다른 사용자로 일련의 업무절차 규칙에 의한 처리를 위해 전달된다 [1]"로 정의하고 있다. 워크플로우는 하나의 큰 작업이 수행 완료될 때까지 일어나는 하위 작업들의 수행 흐름을 표준화된 방법으로 표현한다. 이때 워크플로우의 하위 작업들 간에는 의존성이나 수행 순서, 동시 수행 가능 여부 등의 다양한 관계가 나타난다. 큰 작업을 이러한 작은 작업의 연관된 흐름으로 표현하면 각 작업에 대해 효율적인 자원 배분이 가능하게 되어 전체 작업의 효율성을 향상시킬 수 있다.

2.3 웹 서비스 (Web Services)

웹 서비스는 W3C(World Wide Web Consortium)를 중심으로 표준화가 이루어지고 있으며, 기존의 분산 컴퓨팅을 위한 환경인 CORBA나 DCOM이 지원하지 못했던 언어 및 플랫폼 독립적인 특성을 지원한다. 현재 웹 서비스는 IBM과 MS에 의해 주도적으로 개발되고 있으며, IBM에서는 "웹 서비스는 표준화된 XML 메시지를 통해서 네트워크로 접근할 수 있는 오퍼레이션들을 기술할 수 있는 인터페이스이다. 웹 서비스는 특정 작업을 수행할 수 있다"라고 정의하고 있으며, MS사에서는 "웹 서비스는 다른 응용프로그램에 서비스와 데이터를 제공하는 프로그램 로직의 일부이다. 응용프로그램은 HTTP, XML, SOAP과 같이 어디서든지 사용할 수 있는 웹 프로토콜과 데이터 포맷을 이용해서 웹 서비스를 접근할 수 있다"로 정의한다.

2.4 웹 서비스 기반의 워크플로우 언어

웹 서비스의 서로 다른 서비스간에 발생하는 데이터의 흐름을 명시하기 위해서는 표준화된 언어가 필요하다. WSFL(Web Services Flow Language)은 IBM에서 독자적으로 개발하고 있는 언어로, WSDL(Web Services Definition Language)을 기반으로 서비스들의 연관관계를 명시할 수 있다. 또한 MS, BEA와 함께 BPEL4WS로의 표준화 작업이 진행되고 있다 [4].

3. 본 론

기존의 워크플로우 언어들은 유비쿼터스 환경에서 발생하는 컨텍스트, 프로파일 및 이벤트에 대한 기술을 포함하고 있지 않다. 그렇기 때문에 기존의 워크플로우 언어를 통해서 유비쿼터스 환경에 필요한 서비스들의 연관 관계를 표현하는데 한계가 있다.

이를 해결하기 위해서 본 논문에서는 기존의 워크플로우 언어에 컨텍스트를 표현할 수 있는 엘리먼트(element)를 추가하므로 보다 효과적으로 흐름 관계를 기술할 수 있는 언어인 uWDL(Ubiquitous Workflow Definition Language)을 소개한다. uWDL은 서비스의 흐름을 기술하는 언어로서 센서에 의해 감지되는 정보들을 온톨로지를 통하여 고수준의 컨텍스트, 프로파일 및 이벤트 등의 정보로 변경하고 이와같은 정보를 기반으로 서비스의 흐름을 결정할 수 있는 기능을 제공한다. 또한 온톨로지 형태의 추론 서비스를 이용하기 위한 규칙 기반(rule-based)의 서술 기능도 포함하고 있다. 그림 2는 uWDL 스키마 구조를 나타내며, 기존의 웹 기반의 워크플로우 언어의 장점을 제공하면서 유비쿼터스 환경에 따라 서비스의 흐름을 결정할 수 있는 컨텍스트, 프로파일 및 이벤트 등의 기능을 추가하였다.

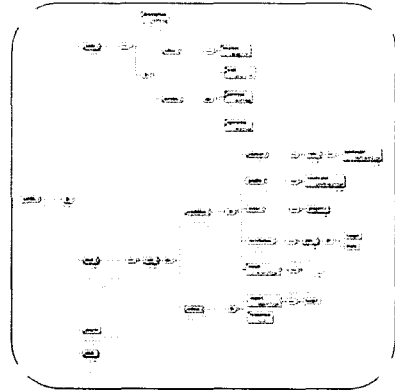


그림 2. uWDL 스키마

그림 3은 uWDL의 전체 모습을 나타낸다. <link> 엘리먼트는 서브 엘리먼트인 <export>의 속성값에 따라 "Control"과 "Data" 링크로 구분되며, 각각은 서비스의 흐름과 데이터의 이동을 명시한다. 또한 <link> 엘리먼트는 서브 엘리먼트인 <condition>을 가지며, 이는 다시 <context>, <profile> 및 <event> 등의 서브 엘리먼트를 통해서 유비쿼터스 환경에서 발생하는 컨텍스트, 프로파일, 이벤트 등을 표현하기 위한 기능을 제공한다.

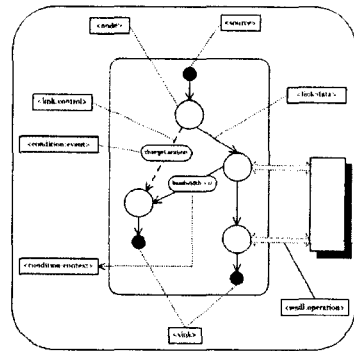


그림 3. uWDL의 전체 개요도

3.1 <node> 엘리먼트

<node> 엘리먼트는 특정 유비쿼터스를 위한 웹 서비스(Web Services)를 가리키기 위해 사용된다. 웹 서비스는 자신의 프로그램 또는 서비스를 명시하기 위해서 WSDL(Web Services Definition Language)을 통해서 자신의 PortType 및 Operation 등의 구체적인 서비스를 명시하게 되고, 이는 <service> 엘리먼트의 <wsdl>, <porttype>, 그리고 <operation> 엘리먼트를 통해서 원하는 서비스의 위치, 서비스 타입 및 오퍼레이션에 대한 정보를 기술하게 된다.

3.2 <link> 엘리먼트

<link> 엘리먼트는 uWDL 언어의 가장 중요한 부분으로 유비쿼터스 환경에서 발생할 수 있는 컨텍스트, 프로파일 및 이벤트 등을 명시하며, 이에 따라 서비스의 흐름이 변화되

록 정의한다. <link> 엘리먼트는 크게 <condition>과 <action> 엘리먼트로 구분된다. <condition> 엘리먼트는 <context>, <profile>, 및 <event> 등을 통하여 해당 노드의 컨텍스트, 프로파일 및 이벤트의 상태를 명시하며 상태가 참(true)일 경우 <action>에서 명시하는 행동을 수행한다. <action> 엘리먼트는 <export>와 <transition>으로 구분되며, <export>의 속성(attribute)을 통해 제어 링크(control link)와 데이터 링크(data link)로 나뉜다. 그리고 <transition>은 현재 노드의 상태(state) 변화를 명시하기 위해서 사용된다.

<condition> 엘리먼트 구성요소 중에 중요한 엘리먼트로는 <context>와 <profile> 엘리먼트를 들 수 있다. <context> 엘리먼트는 유비쿼터스 환경에서 발생하는 저수준 컨텍스트(low-level context)를 규칙 기반(rule-based)의 온톨로지(ontology) 표기법을 이용하여 명시할 수 있도록 <constraint> 엘리먼트를 가지고 있다. <constraint> 엘리먼트는 속성으로 주어(subject), 동사(verb), 목적어(object)를 포함하며, 표기법은 <constraint subject="Temperature" verb=">" object="30">와 같이 온도가 30도 보다 클 경우를 나타낸다.

4. 시나리오

uWDL의 검증을 위해 유비쿼터스 환경의 많은 서비스들 중에서 범위를 회의 준비를 위한 서비스로 한정하여 표현한다. 시나리오의 목적은 "자신의 일정 계획에 따라 회의를 자동으로 준비해 주는 서비스를 구축한다"이다. 이에 따라 시나리오를 설계하면 그림 4, 5와 같다.

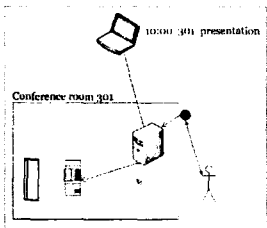


그림 4. 회의 준비 시나리오

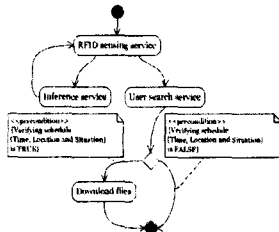


그림 5. UML 모델링

"회사원 A씨는 출근 후 자신의 노트북을 이용하여 일정 계획에 아침 10시에 301호 회의실에서 발표가 있음을 기록한다. 발표 준비를 하던 A씨는 9시 40분경에 회의를 위해 301호 회의실로 이동한다. 301호 회의실 문 위쪽에는 RFID 센서가 설치되어 있어 A씨의 기본 정보(이름, 노트북의 IP 주소 등)를 서버로 전송한다. 서버에서는 전송된 A씨의 IP 주소를 이용하여 일정 계획 서비스로부터 일정 계획 정보를 획득하여 현재 시간(Time), 위치 정보(Location) 및 현재 상황(Situation) 등을 비교하여 참(true)일 경우 A씨의 발표 자료를 다운로드 하여 해당 프로그램으로 실행한다."

그림 5는 시나리오를 UML의 액티비티 다이어그램(Activity Diagram)을 이용하여 모델링한 모습이다. 서비스는 크게 RFID sensing service, User search service, Download files, Inference service로 나뉜다. RFID sensing service는 RFID 센서로부터 전송되는 컨텍스트 정보(이름, IP 주소 등)를 관리한다. User search service는 RFID sensing service

로부터 전달받은 컨텍스트 정보를 이용하여 해당 정보가 uWDL에 의해 명시된 규칙(rule)에 적합한 경우 Download files 서비스를 호출하여 해당 파일을 전송받아 실행시킨다.

표 1은 위의 시나리오를 바탕으로 uWDL 언어를 이용하여 설계한 프로그램 중에 User search service에 해당하는 내용을 나타낸다. <node> 엘리먼트는 웹 서비스로 기술된 서비스를 명시하고 있다. <link> 엘리먼트의 <condition:context> 엘리먼트는 <constraint>를 이용하여 Time, Location, Situation의 조건을 검사한다. <action> 엘리먼트는 <condition>이 참일 경우, Download_files 서비스를 호출한다.

표 1. 시나리오를 적용한 uWDL 인스턴스

```
<?xml version="1.0" encoding="UTF-8"?>
<s:uWDL xmlns:s="http://cs.ssu.ac.kr/uwdl/0.1alpha" version="0.1"
log="0">
<s:node name="User_search_service">
<s:service>
<s:wSDL>http://cs.ssu.ac.kr/axis/services/user_search_service.wsd
</s:wSDL>
<s:operation>verifyingUser</s:operation>
</s:service>
</s:node>
<s:link nodeName="User_search_service">
<s:case type="default">
<s:condition>
<s:context>
<s:rule priority=1>
<s:constraint subject="Time" verb="=" object="13:00"/>
<s:constraint type="AND" subject="Location" verb="is"
object="Room301"/>
<s:constraint type="AND" subject="Situation" verb="is"
object="presentation"/>
</s:rule>
</s:context>
</s:condition>
<s:action>
<s:export type="data">
<s:target>Download_files:ip_address</target>
</s:export>
</s:action>
</s:case>
</s:link>
<s:sink>result</s:sink>
</s:uWDL>
```

5. 결론

uWDL (Ubiquitous Workflow Definition Language)는 BPXL4WS, WSFL, XLANG 등 기존의 워크플로우 언어들의 장점들을 포함하며, 유비쿼터스 환경에서 다루어지는 컨텍스트, 프로파일 및 이벤트 정보를 기술할 수 있는 언어이다. 또한 온톨로지를 기술하기 위한 언어인 DAML+OIL과의 인터페이스를 위하여 규칙 기반(rule-based)의 표현 방식을 사용하고 있다.

6. 참고 문헌

- [1] D. Hollingsworth, "The Workflow Reference Model", Technical Report TC00-1003, Workflow Management Coalition, 1994.
- [2] M. Weiser, "Some Computer Science Issues in Ubiquitous Computing", Communications of the ACM, July 1993/Vol.36, No.7 75-84.
- [3] M. Weiser, "The Computer for the 21st Century" Sci. Amer., Sept., 1991
- [4] IBM, Web Service Flow Language 1.0