다중 상황인지 워크플로우 지원을 위한 uWDL 확장

안중진, 최종선, 김범수, 최재영, 유재우
숭실대학교 컴퓨터학과
e-mail : {jjahn, jschoi, bskim}@ss.ssu.ac.kr
{choi, cwyoo}@ssu.ac.kr

An uWDL Extension for Supporting Multiple Context-Aware Workflows

Joongjin Ahn, Jongsun Choi, Bumsu Kim, Jaeyoung Choi, Chaewoo Yoo
Dept. of Computing, Soongsil University

요 약

최근 비지니스 프로세스를 위한 워크플로우 기술에 상황 인지 기능을 추가하여 유비쿼터스 컴퓨팅 환경에 적용하려는 연구가 활발히 진행되고 있다. 그러나 기존의 상황인지 기반 워크플로우 언어들은 Sequence Control-Flow 패턴과 같이 하나의 서비스만을 지원하는 단일 워크플로우 수준에 머물러 있었다. 이에 본 논문에서는 Parallel Split Control-Flow 패턴과 같이 동시에 하나 이상의 서비스를 지원하기 위해 uWDL 을 확장한 언어 구조를 제안한다. 제안하는 언어 구조는 다중 워크플로우에서 플로우 간 공통으로 사용하는 변수를 모아놓은 데이터 구조를 기반으로 다중 워크플로우 서비스를 지원한다.

1. 서론

워크플로우(Workflow)는 현대 비즈니스 오퍼레이션의 자동화 및 정보 처리와 밀접하게 관련된 개념으로, 제조 및 생산, 서비스 등 다양한 분야에서 공통적으로 나타나는 업무절차 처리를 위한 태스크를 개념적 레벨에서 기술한 것이다[1]. 또한, 기업체의 업무 처리와 정보 처리를 자동화하고 최적화 시켜 작업처리의 효율을 향상시킬 수 있는 일련의 단계나 행위들을 의미한다[2].

워크플로우 기술은 제조과정에서만 적용되는 것은 아니며, 기업의 전자결재나, 각종 행정업무, 소프트웨어나 그래픽을 개발하는 일 등 여러 분야에 광범위하게 적용되고 있다. 최근에는 분산 컴퓨팅 시스템에 활발히 적용되고 있고, 나아가서 그리드 컴퓨팅이나 유비쿼터스 컴퓨팅에도 적용하려는 시도가 활발히 이루어지고 있다.

워크플로우 기술을 유비쿼터스 환경에서의 서비스 자동화에 이용하려는 대표적인 연구인 uWDL(Ubiquitous Workflow Definition Language) 기술[6]은 사용자의 상황정보를 서비스 실행조건으로 표현 할 수 있다. 그러나, uWDL 은 워크플로우가 가지는 다양한 기능을 충분히 제공하지 못하며, 특히 여러 사용자들의 독립적인 워크플로우들이 유기적으로 관련되는 서비스의 흐름을 하나의 통합된 워크플로우로 표현할 수 있는 기능을 포함하지 않는다. 유비쿼터스 컴퓨팅 환경에서의 서비스는 다수의 사용자를 중심으로 동적인 상황정보가 발생한다. 따라서, 이를 기반으로 상황인지 워크플로우 서비스를 표현하기 위한 워크플로우 언어는 다수의 사용자가 요구하는 서비스 흐름을 하나의 워크플로우 시나리오에 표현할 수 있는 기능을 제공하여야 한다.

본 논문에서는 상황인지 워크플로우 서비스의 실행조건으로써 규칙 기반(rule-based) 컨텍스트 모델을 사용하는 확장된 uWDL 을 제안한다. 확장된 uWDL 은 다수의 사용자를 위해 독립적으로 존재하는 서비스 워크플로우를 필요에 따라 유기적으로 통합하고, 정확하게 분리할 수 있는 기능을 포함한다. 따라서, 상황인지 워크플로우 개발자는 다수 워크플로우의 동적인 결합 및 분리가 빈번하게 발생하는 실제 유비쿼터스 환경에서의 상황인지 워크플로우 관련 응용프로그램 개발에 확장된 uWDL 기능을 활용할 수 있을 것으로 기대한다.

"본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음" (IITA-2009-(C1090-0902-0007))

2. 관련 연구

2.1 유비쿼터스 컴퓨팅 (Ubiquitous Computing)

유비쿼터스는 “언제 어디에든 존재한다” 라는 의미의 라틴어로 1988 년 처음으로 제록스 팰러 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[3]” 이라고 정의하고 이는 컴퓨팅 환경이 현실세계의 사물과 환경속

으로 스며들어 일상생활에 통합되는 것[4]을 의미한다.

2.2 워크플로우

WfMC 에서는 워크플로우를 “전체적인 또는 부분적인 비즈니스 프로세스의 자동화를 의미하며, 이때 문서, 정보, 태스크가 한 사용자에서 다른 사용자로 일련의 업무절차 규칙에 의한 처리를 위해 전달된다[5]”로 정의 하고 있다. 워크플로우는 하나의 큰 작업이 수행 완료될 때까지 일어나는 하위 작업들의 수행 흐름을 표준화된 방법으로 표현한다. 이 때 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 기반의 워크플로우 언어

BPEL4WS[7], WSFL[8], XLANG[9]은 비즈니스 및 분산 컴퓨팅 환경에 적용하기 위한 웹 서비스 기반의 언어들이다. 이 언어들은 서비스의 전이 조건으로 이전 서비스의 결과 값과 서비스 및 시스템의 이벤트 정보를 사용한다. 또한 XPath 기능을 통해 다른 XML 형태의 결과 값을 이용할 수 있다.

그러나 위 언어들을 유비쿼터스 컴퓨팅 환경에서 사용하기에는 아래와 같은 문제점이 있다.

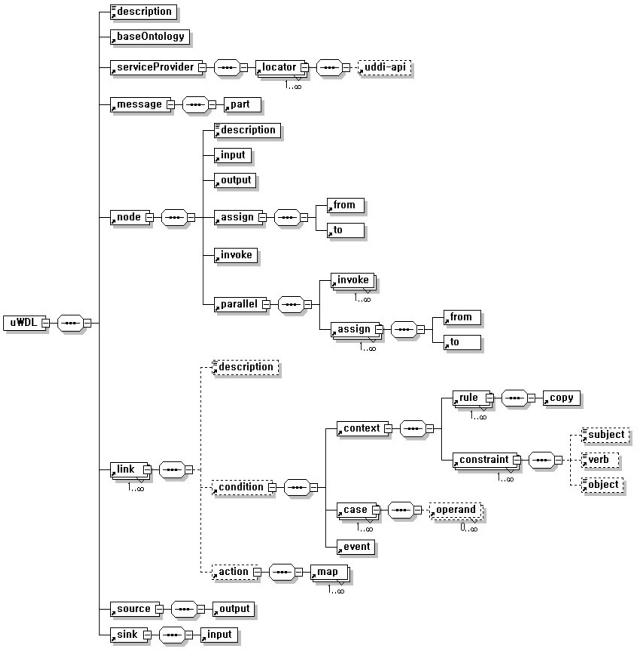
첫째, 유비쿼터스 환경에서 워크플로우가 서비스를 제공하기 위해서는 서비스의 결과 값이나 단순한 이벤트 정보가 아닌 사용자가 처해있는 환경으로부터 발생하는 다양한 상황 정보를 서비스의 전이조건으로 사용해야 한다.

둘째, XPath를 이용하여 컨텍스트 정보를 얻었을 경우 XPath의 개발 목적상 논리 및 조건 연산자만으로는 유비쿼터스 환경에서 발생할 수 있는 고수준의 상황 정보를 표현하기가 어렵다.

2.4 유비쿼터스 환경 상황인지 워크플로우 언어

유비쿼터스 환경에서 발생하는 컨텍스트와 이를 이용하는 서비스간의 연관 관계를 워크플로우 시나리오에 효과적으로 표현하기 위한 상황인지 워크플로우 언어로써 uWDL 은 유용하다. 그림 1 은 uWDL 의 스키마를 나타낸다.

그림 1 의 uWDL 은 유비쿼터스 환경에서 사용자의 특성을 파악하고 그에 맞는 서비스들의 관계를 기술할 수 있는 언어로써, 상황인지 워크플로우 서비스 개발자는 uWDL 을 이용하여 서비스들 간의 연관 관계를 워크플로우로 구성하여 작업의 분업 및 자동화를 이룰 수 있다. 이를 위해 가장 널리 사용되고 있는 웹 서비스(Web Services)의 WSDL 을 이용하여 uWDL 의 서비스를 기술하고, 워크플로우에 의해 구성된 서비스들의 연관 관계는 컨텍스트의 종류에 따라 동적으로 재구성되어 상황에 맞는 서비스를 제공하게 된다[6].

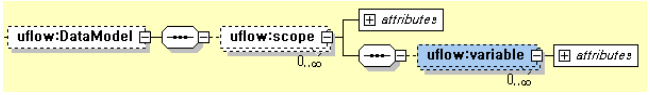


(그림 1) uWDL 의 스키마 구조

그러나, 기존 uWDL 은 단일 사용자를 위한 상황인지 워크플로우 시나리오만을 기술할 수 있는 기능을 제공한다. 따라서, 다수의 사용자를 위한 워크플로우가 발생하는 실제 유비쿼터스 환경에서의 상황인지 워크플로우 응용프로그램 개발에 적용되기에는 부적합하다.

3. 본 론

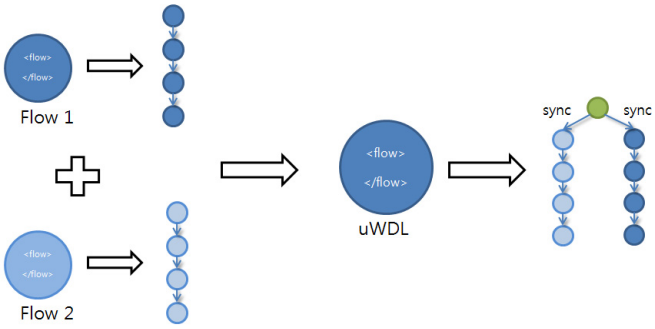
기존 uWDL 에 다수 사람들이 동시에 여러 가지 서비스를 제공받을 수 있는 멀티 플로우(Multi-flow) 기능을 부여하기 위해 <DataModel>, <wait>, <synchronize> 요소를 추가 하였다. 이 중 <DataModel> 요소는 플로우 간에 공통적으로 사용하는 변수를 모아놓은 저장소로 데이터 공유뿐만 아니라 분기된 플로우 혹은 노드의 상태를 값을 저장하여 연관성 있는 플로우, 노드 간에 IPC (Inter-Process Communication)기능을 제공한다. 아래 (그림 2)는 <DataModel> 요소 스키마를 보여준다.



(그림 2) <DataModel> 요소의 스키마 구조

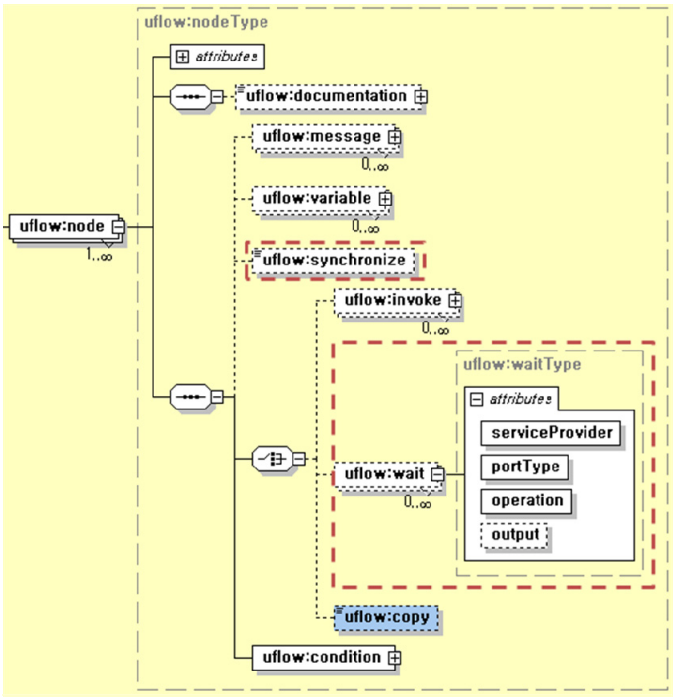
<node> 원소는 워크플로우 시나리오에서 하나의 독립적인 작업을 수행하는 단위로 하나의 노드 내부에서 기존 워크플로우 언어의 여러 노드로 구성되는 흐름 관계를 부분적으로 포함할 수 있다. 아래 (그림

3)은 멀티 플로우의 개념을 보여준다.



(그림 3) 멀티 플로우 개념도

아래 (그림 4)은 <node> 요소 스키마를 보여준다.



(그림 4) <node> 요소의 스키마 구조

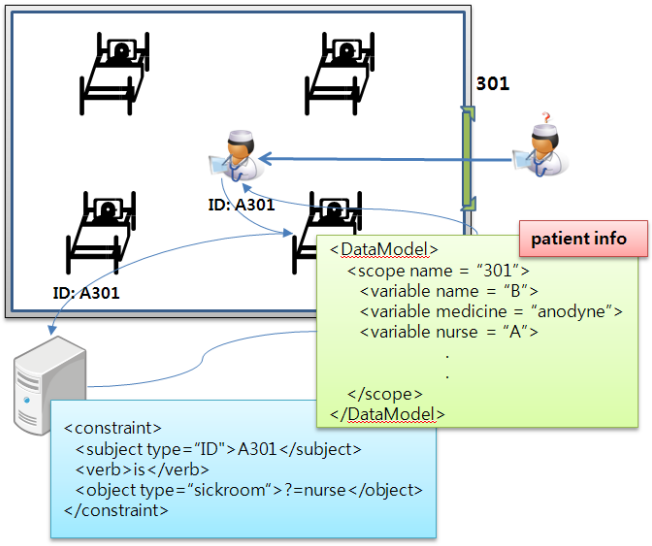
위 (그림 4)의 스키마 구조에서 <synchronize> 요소는 두 개 이상의 노드가 하나의 노드로 통합될 때 사용하며 <wait> 요소는 웹 서비스 호출을 위한 요소이다.

4. 실험 및 평가

4.1 시나리오

유비쿼터스 환경의 많은 서비스들 중에서 범위를 병원 의료 서비스로 한정하여 표현한다. 시나리오의 목적은 "환자들의 건강 상태 및 약물 투여 정보를 제공해주는 서비스를 구축한다."이다. 이에 따라 시나리오를 설계하면 (그림 5)와 같다.

"간호사 A 씨는 출근 후 환자들의 정보를 보여주는 전자 차트를 들고 자신이 간병하는 수술 대기 환자들이 있는 301 호 병실로 이동한다. 간호사 A 씨가 B 환자의 침대 곁에 가면 침대에 부착되어 있는 센서가 전자 차트에 ID 번호를 감지하여 서버에 전송하게 된다. 서버에서는 현재 전송 받은 ID 가 B 환자의 담당 간호사인지 비교하고 담당 간호사면 건강 상태와 약물 종류, 양, 투여 시간 등의 정보를 전자 차트의 리더기에 전송하여 전자 차트에 정보를 보여준다."



(그림 5) 의료 서비스 시나리오

위 (그림 5)에서 간호사 A 씨의 ID 인 A301 인지 비교한 후에 A301 이면 환자 B 씨의 정보를 DataModel 을 이용하여 간호사 A 씨 전자 차트에 보여준다.

5. 결 론

본 논문에서는 다수의 사용자를 위해 독립적으로 존재하는 워크플로우를 하나의 상황인지 워크플로우 시나리오에 자연스럽게 통합하고, 분리할 수 있는 기능을 제공하는 확장된 uWDL 제안하였다. 또한, 간단한 상황인지 워크플로우 시나리오 작성을 통해 제안하는 확장 uWDL의 다중 워크플로우의 통합 에 대해 실험하였다. 따라서, 개발자들은 제안하는 확장 uWDL을 이용하여 실제 유비쿼터스 환경에서 발생 가능한 다수의 워크플로우를 유연하게 통합하는 상황인지 워크플로우 응용프로그램을 작성할 수 있을 것이다. 향후에는 확장된 uWDL을 처리 하는 엔진에 대한 연동 기술에 대해 연구할 것이다.

참고문헌

[1] D. Georgakopoulos and M. Hornick, "An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure", Journal on Distributed and Parallel Database Systems, Vol3, No2, pp. 119-153, 1995.

[2] A. Sheth, "Workflow Automation: Applications,

Technology and Research", Tutorial notes of SIGMOD Conference, May 1995.
[3] M. Wesier, "Some Computer Science Issues in Ubiquitous Computing", Communications of the ACM, July 1993/Vol.36, No7 75-84.
[4] M. Wesier, "The Computer for the 21st Century", Sci. Amer., Sept., 1991
[5] D. Hollingsworth, "The Workflow Reference Model", Technical Report TC00-1003, Workflow Management Coalition, 1994.
[6] Kyuongho Shin, "AWorkflow Language for Context-Aware Services," mue,pp.1227-1232, 2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07), 2007.
[7] Tony Andrews, Francisco Curbera, Yaron Goland, "Business Process Execution Language for Web Services," BEA Systems, Microsoft Corp., IBMCorp., Version 1.1, 2003.
[8] Frank Leymann, "Web Services Flow Language(WSFL 1.0)," IBM Corp., 2001.
[9] Satish Thatte, "XLANG Web Services for Business Process Design," Microsoft Corp., 2001.