

상황인지 다중-워크플로우 처리기

정재훈^o 김범수 최종선 최재영

송실대학교 컴퓨터학과

jheong@ss.ssu.ac.kr bskim@ss.ssu.ac.kr jschoi@ss.ssu.ac.kr choi@ssu.ac.kr

A Handler for Context-aware Multiple Workflows

Jae-Hoon Jeong^o Bum-Soo Kim Jong-Sun Choi Jae-Young Cho

School of Computing Soong-Sil University

요 약

분산 및 그리드 환경에서의 워크플로우 기술은 현재까지 지속적으로 발전하고 있으며, 최근에는 이러한 기술을 유비쿼터스 컴퓨팅 환경에 적용하려는 시도가 이루어지고 있다. 이러한 환경에서의 서비스 제공자는 다양한 상황 정보들을 인식하여 사용자에게 최적의 서비스를 제공 할 수 있어야 한다. 그러나 기존 워크플로우 기반의 상황인지 미들웨어들은 단일 워크플로우 서비스를 제공하는 수준에 머물러 있다. 이에 기존 시스템들은 다양하고 복합적인 서비스를 제공하는데 한계가 있다. 이를 위해 본 논문에서는 확장된 CAWL을 바탕으로 다중 워크플로우 서비스를 처리하고 제공할 수 있는 상황인지 다중 워크플로우 처리기를 제안한다. 제안하는 처리기는 상황인지를 기반으로 다수의 사용자에게 복합적인 워크플로우 서비스를 동시에 제공하며, 동시에 진행되는 병렬 서비스의 흐름을 처리할 수 있다.

1. 서 론

워크플로우 기술은 IBM, Microsoft, Oracle 등과 같은 세계적인 컴퓨터 회사들의 지원을 받으며 기업의 통합 솔루션을 제공하기 위해 발전하여 왔다. 지속적으로 연구되어온 워크플로우 기술은 최근 유비쿼터스 컴퓨팅 환경에 적용하려는 시도가 이루어지고 있다.[1][2]

유비쿼터스 컴퓨팅 환경에서 사용자는 동적으로 변하는 환경변화에 적합한 서비스를 제공받아야 한다. 그리고 이러한 일련의 서비스들은 상황인지를 기반으로 서비스들의 수행 절차를 기술한 워크플로우로 표현할 수 있다.[1] 상황인지 기반의 워크플로우는 사용자의 상황정보에 대해 컨텍스트 모델을 워크플로우의 전이 조건으로 사용한다.

현재까지 연구되었던 상황인지 기반의 워크플로우 시스템은 서비스 대상에게 단일 워크플로우 서비스만을 제공했다. 이러한 시스템은 동적으로 변하는 유비쿼터스 환경에서 사용자의 요구사항에 맞는 서비스를 제공하기에 부족하다. 따라서 동적인 사용자 요구사항에 맞는 효과적인 서비스 제공 방안을 연구할 필요가 있다.

이를 위해 본 논문에서는 상황인지 기반의 다중 워크플로우 처리기를 제안한다. 제안하는 처리기는 서비스 환경의 환경 변화와 동적인 사용자 요구사항에 적합한 서비스를 제공하며 서비스의 흐름을 병렬적으로 처리할 수 있을 뿐만 아니라, 기존의

워크플로우 서비스의 재 사용을 통하여 시스템의 성능을 향상시킬 수 있다.

2. 관련연구

워크플로우는 작업들 간의 연관된 흐름을 표현한다. 워크플로우 처리는 언어에 명시된 전이조건 또는 제약조건을 워크플로우 작업들의 흐름을 결정하는데 사용한다[3, 6] 기존 웹서비스 기반의 워크플로우 언어인 WSFL[3], WS-BPEL[4], XLANG[5]들은 서비스 전이 조건은 이전 서비스의 결과값이나 서비스 또는 시스템의 이벤트 정보를 사용한다. 그러나 이러한 언어들은 유비쿼터스 환경에 적용하기에 부족하다. 유비쿼터스 환경에서 적합한 서비스를 제공하기 위해서는 컨텍스트 정보를 서비스의 전이 조건으로 사용해야한다.

말레이시아 멀티미디어 대학에서는 서로 다른 네트워크 디바이스를 통합 관리하기 위해 CWME[7]을 제안하였다. CWME은 CE(Context Engine)을 통해 컨텍스트 정보를 모아 표현이 쉽고 명확한 RDF형식으로 컨텍스트 정보를 생성하여, 서비스 수행환경에 활용한다. 하지만 디바이스 장치 관리를 위한 목적의 워크 플로우 시스템은 인간이 서비스 대상으로 포함될 수 있는 유비쿼터스 환경에서 사용 되기에 부족하며, 컨텍스트 활용도 또한 높지 않다. 그렇기 때문에 동적인 서비스 요구 사항을 반영하는 서비스 제공이 어렵다는 단점이 있다.

“본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구 결과로 수행되었음”
(NIPA-2009-(C1090-0902-0007))

WS-BPEL4People[8]은 비즈니스 프로세스에서 사람을 서비스 환경에 존재하는 하나의 대상으로 정의하고 서비스를 제공할 수 있는 방안을 제공한다. 대상이 되는 사람은 서비스 제공 환경에서 하나의 파트로 기술 될 수 있으며 WS-BPEL4People에서는 이러한 파트를 시나리오에 기술 할 수 있는 규칙을 WS-BPEL2.0의 확장을 통해서 제공하고 있다. WS-BPEL4People은 유비쿼터스 환경에서 효율적인 서비스 기술로서 활용될 수 있다. 하지만 WS-BPEL2.0의 기술의 단순 확장이라는 점에서 유비쿼터스 환경의 동적인 요구 사항을 반영한 워크플로우 서비스를 제공하기에는 우리가 있다.

uFlow[1]은 상황인지 기반의 워크플로우 시스템으로 구조적 컨텍스트 모델을 설계하였고, 이 구조적 컨텍스트 모델을 워크플로우의 서비스 전이 조건으로 사용하는 웹서비스 기반의 워크플로우 언어인 uWDL을 기반으로 한다. uFlow 프레임워크는 구조적 컨텍스트 모델을 바탕으로 유비쿼터스 환경에서 동적인 워크플로우 서비스를 제공 할 수 있다. 하지만 워크플로우 서비스를 순차적으로만 기술 할 수 있기 때문에 환경에서 제공 될 수 있는 서비스의 다양성에 한계를 보인다.

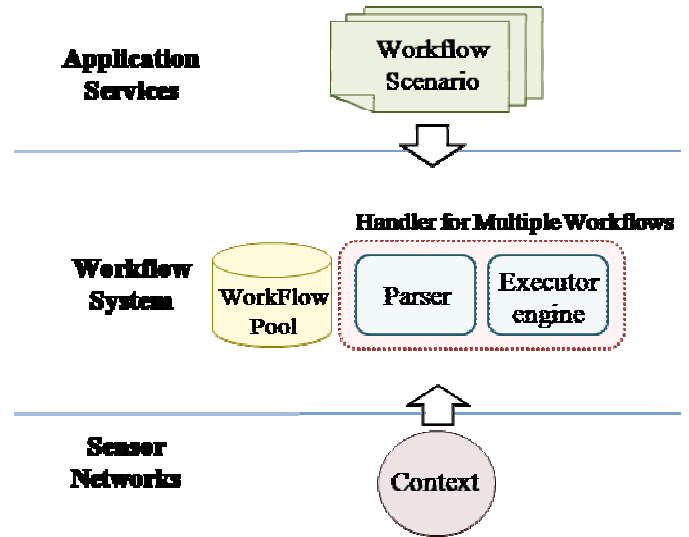
이에 본 논문에서는 다중 워크플로우를 기술할 수 있는 CAWL[10]을 기반으로 유비쿼터스 환경에서 발생하는 컨텍스트 정보를 효과적으로 반영하여 동적인 사용자 요구 사항을 반영할 수 있는 상황인지 기반의 다중 워크플로우 처리기를 제안한다.

3. 제안하는 다중-워크플로우 처리기

본 논문에서는 CAWL을 기반으로 하는 다중 워크플로우 처리기를 소개한다. 이에 본 장에서는 제안하는 처리기의 요구사항을 소개하고, 본 처리기의 설계 및 구성요소에 대해 설명한다. 본 처리기는 [그림 1]같이 사용자에게 워크플로우 서비스 제공을 위해 각각의 업무 수행 환경의 결합으로 구성된다.

유비쿼터스 컴퓨팅 환경에 속한 다수의 사용자에게 워크플로우 서비스를 제공하기 위한 다중 워크플로우 환경은 [그림 1]과 같이 3 가지 주요 환경으로 구성된다. 첫째로 사용자에게 제공할 서비스를 상황인지 기반의 언어로 기술하여 서비스 시나리오를 개발하는 워크플로우 시나리오 개발환경, 둘째로 시나리오를 기반으로 다수의 사용자에게 동시에 서비스를 제공할 수 있도록 기술된 서비스의 수행조건을 검사하며 서비스 명령을 처리하여 워크플로우 서비스를 제공하는 다중 워크플로우 처리 엔진,

마지막으로 워크플로우 서비스의 전이 조건으로 사용되는 컨텍스트 정보를 취득하기 위한 네트워크 환경으로 나뉜다.



[그림 1] 다중 워크플로우 환경 구성도

제안하는 처리기는 CAWL로 작성된 시나리오에서 다중 워크플로우 서비스를 제공하기 위해 워크플로우 파서와 실행 엔진으로 구성된다. 그리고 다중 워크플로우를 처리하기 위한 실행 엔진의 핵심 기능은 스케줄러가 담당한다.

3.1 파서

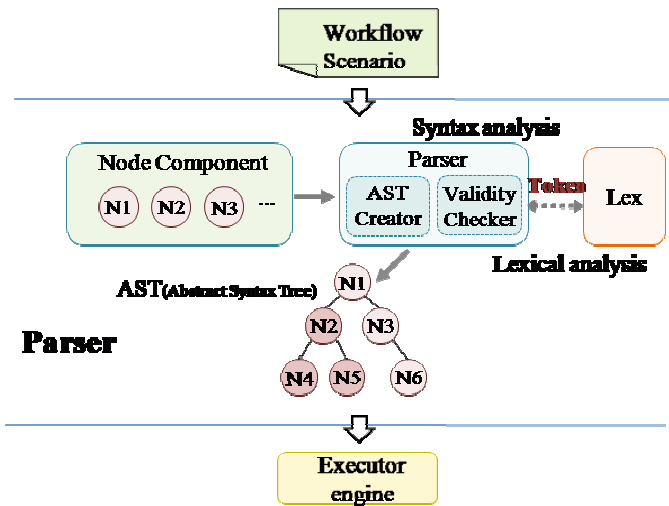
워크플로우 파서(Workflow Parser)는 워크플로우 시나리오의 유효성을 검증하고 플로우 단위(사용자에게 서비스를 제공하기 위한 최소 단위)로 나누어 워크플로우 풀(Workflow Pool)에 저장한다.

워크플로우 파서가 처리하는 일련의 과정은 [그림 2]와 같다. XML 기반 워크플로우 언어로 작성된 시나리오 문서를 Lex를 통해서 토큰(Token)단위로 분해하고 토큰이 언어의 규칙과 일치하는지 확인하여, 문서의 유효성 검사를 수행한다. 이러한 유효성 검사를 통해 파서는 문서에 유효한 토큰이라고 판단되면 토큰에 해당하는 노드를 생성하고 문서 규칙에 맞게 트리를 확장한다.

AST(Abstract Syntax Tree)는 CAWL로 기술된 XML구조의 시나리오를 처리기에서 처리 가능한 하나의 트리 형태로 구성한 것이다. 시나리오 문서가 트리 형태로 구성되면 워크플로우

서비스를 제공하기 위한 시스템에서 처리하는 각 단계를 기존의 시나리오를 참조하거나, 다른 일련의 처리 절차가 없이도 파서를 통해 구성된 트리 탐색만을 통해 워크플로우 서비스를 순차적으로 제공하는 방안을 제시한다. 또한, 서비스를 위해 문서에 기술된 변수들을 변수 테이블에 따로 저장하여 관리한다.

CAWL로 작성된 워크플로우 시나리오에서 <activator>는 최초 서비스 워크플로우를 수행하기 위한 컨텍스트 조건을 기술한다. 시나리오에 기술된 각 워크플로우의 노드에는 서비스를 수행시키는 <invoke>와 노드의 전이 조건에 해당하는 컨텍스트 정보를 기술한다.



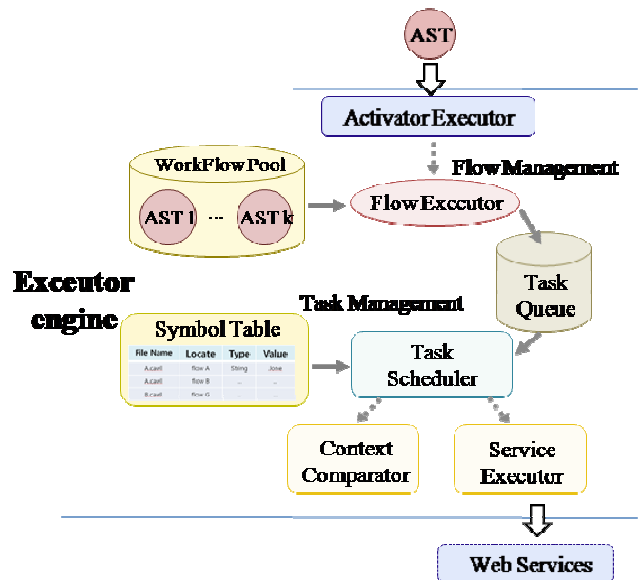
[그림 2] 워크플로우 파서

문서에 대한 AST가 생성된 후에 사용자에게 제공할 수 있는 서비스 플로우 단위로 워크플로우 풀에 저장한다. 워크플로우 풀은 중복된 서비스 플로우에 대해 다시 파싱과정을 거치지 않고 이미 트리형태로 저장되어 있는 플로우를 재 사용하여 서비스를 제공함으로써 파서의 처리 부하를 줄일 수 있다는 장점이 있다.

워크플로우 시나리오 문서에 기술되고 사용되는 변수는 범위(Scope)를 갖는다. 변수의 사용되는 범위에 따라 노드 변수, 플로우 변수, 전역 변수로 구분되어 사용되기 때문에 변수 테이블에서는 변수의 범위와 범위들의 상하 관계 또한 저장 가능해야 한다. 저장된 변수를 사용할 때, 저장된 정보를 탐색하여 변수가 올바른 범위 내에서 사용될 수 있도록 변수 테이블을 구성하여 플로우 서비스를 제공할 때나 플로우 서비스 재 사용시에 안정적인 방안을 제시하고 있다.

3.2 워크플로우 처리 엔진

워크플로우 엔진은 기술된 컨텍스트 정보와 센서를 통해 들어오는 정보를 비교한 후 일치하는 조건의 컨텍스트 정보가 센싱되어 들어온 경우 엔진은 기술된 워크플로우를 실행시킨다.



[그림 3] 워크플로우 엔진

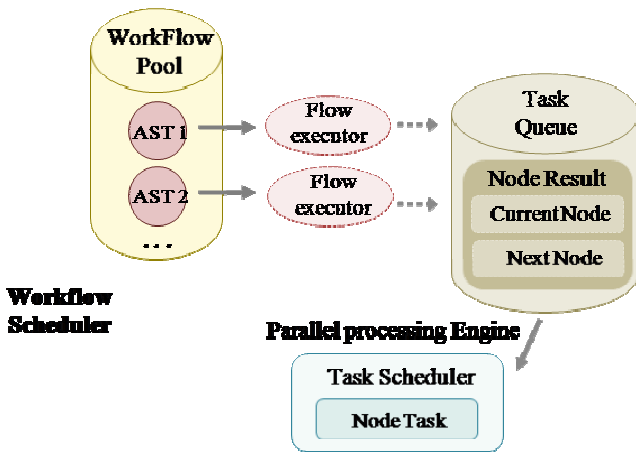
[그림 3]은 워크플로우 엔진의 처리 과정을 나타낸 것이다. 시나리오 파싱을 통해 얻어진 AST의 <activator>에 기술된 컨텍스트 정보와 유비쿼터스 환경에서 서비스를 위해 센싱되어 들어오는 컨텍스트 정보를 컨텍스트 비교기에서 비교하여 <activate>에 기술된 최초 워크플로우 서비스를 수행한다.

수행되는 플로우들은 노드 단위로 서비스를 제공하며, 이러한 단위 서비스들은 서비스 제공을 위해 발생하는 일련의 작업(Task)단위로 작업 큐(Work Queue)에 저장된다. 작업 스케줄러(Task Scheduler)는 이러한 단위 작업들을 순차적으로 처리하여 서비스를 제공한다.

동시에 많은 플로우를 수행하기 위해 멀티스레드(Mult-thread) 환경으로 여러 사용자에게 서비스를 제공한다. 스레드 환경으로 여러 사용자에게 서비스를 제공하면 각 사용자 서비스 수행 환경이 개별적으로 동작하기 때문에 서비스의 동시성을 만족 시킬 수 있다.

3.3 워크플로우 스케줄러

복합적으로 다수의 사용자에게 동시에 여러 워크플로우 서비스를 제공하기 위해서 스케줄링(Scheduling) 기법이 필요하다. 스케줄링의 목적은 워크플로우 서비스를 수행할 때, 누락되는 서비스나 지연되는 서비스가 발생하지 않도록 하여 서비스를 제공받는 사용자로 하여금 동등한 조건의 서비스를 제공하기 위함이다. 이러한 스케줄링을 제공하기 위한 워크플로우 스케줄러는 [그림 4]와 같다.



[그림 4] 워크플로우 스케줄러

워크플로우 스케줄러는 서비스를 위해 수행되어야 하는 플로우들의 수행 흐름 제어와 서비스를 위해 업무의 처리를 관리하고 처리한다.

워크플로우 풀에서 가지고 온 플로우들은 플로우 실행기(Flow Executor)를 통해서 각각 제어된다. 플로우 실행기는 하나의 워크플로우를 제어하기 위해 스레드 형태로 워크플로우 마다 제공되며, 제어하는 워크플로우의 <link>에 기술된 수행 흐름에 맞게 제어한다. 플로우 실행기를 통해서 진행되는 노드의 처리 업무는 업무 큐(Task Queue)에 저장되어 해당 업무를 처리한다.

제공되는 워크플로우 서비스는 다중-워크플로우 구조이기 때문에 다수의 플로우 실행기를 통해서 워크플로우 서비스가 진행된다. 실행되는 플로우 실행기들에서 발생하는 단위 업무들은 업무 큐에 수행 할 순서대로 저장되게 된다. 업무 스케줄러는 동시에 수행되고 있는 여러 워크플로우 업무들을 하나씩 처리한다.

스케줄러에 의해서 처리되는 업무는 하나의 업무만을 처리하지만 여러 워크플로우들의 업무를 특정 워크플로우의 편중없이 처리 순서에 맞게 업무를 처리하기 때문에 다수의 워크플로우가 실시간으로 수행되는 효과를 볼 수 있다.

4. 실험 및 평가

본 논문에서 제안하는 다중 워크플로우 처리기의 설계 목적은 복합 워크플로우 모델을 지원하는 CAWL을 기반으로 여러 흐름의 워크플로우를 통합하는 복합 워크플로우 서비스를 제공하는 것이다. 이를 위해 본 실험에서는 CAWL을 이용하여 시나리오를 작성하였으며, 이를 이용하여 사용자 일정에 따라 적합한 서비스를 제공하는지 시뮬레이션 실험으로 확인하였다. 제안하는 시스템의 구현 환경은 윈도우즈 비스타를 탑재한 펜터임 Core2Duo 3.0Ghz를 이용하였으며, 개발 도구와 플랫폼으로써 JDK 1.6 과 이클립스를 사용하였다.

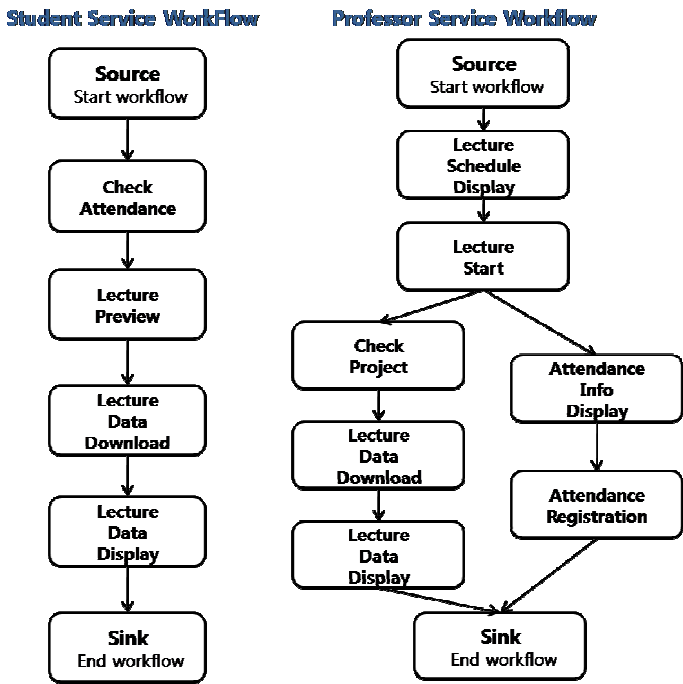
[표 1] CAWL 강의 준비 시나리오

학생 A는 강의실에 도착하여 자신의 지정석에 앉게 되면 자신의 출석을 체크하는 출석 서비스를 제공받는다. 그 후 학생 A는 자리에 있는 모니터를 통해 오늘의 전체적인 강의 내용을 볼 수 있도록 모니터링 서비스를 제공받는다. 수업 시작 시간이 되면 수업의 강의 자료를 다운받아 모니터를 통해 강의 자료를 보며 수업을 진행한다.
교수는 강의실에 들어오면 자신이 기록했던 오늘의 강의 스케줄을 개인 화면을 통해 볼 수 있는 서비스를 제공받고, 강의가 시작하게 되면 강의를 위한 강의 자료를 준비하는 서비스와 학생들의 출결 상태를 확인하고 등록하는 서비스를 동시에 제공받고, 그 후 강의를 진행한다.

[표 1]은 강의 준비를 위한 시나리오를 나타낸다. 강의실에서 제공받는 서비스를 학생을 위한 워크플로우 서비스와 교수를 위한 워크플로우 서비스를 동시에 제공한다.

[그림 5]은 시나리오를 바탕으로 하는 워크플로우 흐름을 보여준다. Student Service Workflow는 강의실에 들어오는 학생들 각각을 위해 서비스를 제공하기 위한 워크플로우이며, Professor Service Workflow는 강의를 진행하기 위해 강의실에 들어오는 교수를 위한 서비스를 제공하기 위한 워크플로우이다. 두 서비스 워크플로우는 서로 독립적으로 동시에 수행되어 사용자 각각에게 서비스를 제공한다. Professor Servie Workflow는 병렬적인 흐름의 워크플로우이며 서비스를 제공받는 사용자에게는 동적인 흐름을 갖는 워크플로우 서비스가 제공된다.

“본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구 결과로 수행되었음”
(NIPA-2009-(C1090-0902-0007))



[그림 5] 강의 준비 시나리오 흐름도

실험 내용은 시나리오를 CAWL을 기반으로 작성하고, 이를 처리하는 과정을 보이는 것이다. CAWL 문서는 [표 1]에 시나리오를 바탕으로 내용은 [표 2]와 같다.

[표 2] CAWL 시나리오

```
<uWDL name="" version="2.0.1"
targetNamespace=""
<activator name="JohnScheduleActivator">
  <!-- 중략 -->
  <activate flow="ProfessorService_Flow" />
  <activate flow="StudentService_Flow" />
  <!-- 중략 -->
<node name="LectureStart">
  <!-- 중략 -->
  <invoke ... />
  <parallel>
    <transition linkId="CheckProject" />
    <transition
linkId="AttendanceInfoDisplay" />
  </parallel>
  <!-- 중략 -->
```

```
<synchronize>
  <nSync name="AttendanceRegistration">
  <nSync name="DisplayLectureData">
</synchronize>
  <!-- 중략 -->
</uWDL>
```

[표 2]는 실험의 주요 부분을 위한 내용만을 나타내었다. 복합 워크플로우 서비스를 제공하기 위한 시나리오 문서에는 다수의 플로우를 기술할 수 있어야 한다. 이를 위해 본 실험에서는 상황의 복잡한 정도를 최소 수준으로 가정하여 2 개의 워크 플로우로 표현하였다.



[그림 6] 시나리오 실행 결과

[그림 6]의 실험 결과를 보면 학생과 교수를 위한 두 개의 시나리오가 동시에 수행됨을 알 수 있다. 두 개의 시나리오는

병렬적으로 동시에 수행된다.

교수 시나리오에서 <parallel>태그에 의해서 순차적인 워크플로우가 두 개의 병렬적인 흐름을 갖는 워크플로우로 분기하여 출석 체크를 위한 서비스와 수업 진행을 위한 서비스로 분기하여 병렬적인 흐름으로 서비스가 수행되고 있다. 그리고 교수 시나리오 플로우의 <sink>노드에서 기술된 <synchronize>태그에 의해 두 개의 병렬적인 흐름을 갖는 플로우는 하나의 흐름으로 병합된 뒤에 플로우 서비스를 끝낸다.

5. 결론

본 논문에서는 유비쿼터스 환경에서 다수의 사용자에게 복합적인 워크플로우 서비스를 제공할 수 있는 상황인지 기반의 워크플로우 시스템을 제안한다.

서비스 기반 워크플로우 기술은 많은 분야에서 연구되고 있으며 최근에는 유비쿼터스 환경에서 동적인 요구 사항을 반영하여 서비스를 기술하기 위한 방안으로 사용되고 있다. 워크플로우 기술은 서비스의 위치를 추적하거나 수행되어야 할 서비스를 관리하는데 용의하며, 기술된 워크플로우를 통일한 서비스를 제공받는 사용자에게 서비스를 제공해 줄 수 있기 때문에 재사용성이나, 서비스 자원의 관리 측면에서도 높은 효율을 보일 수 있다.

본 시스템은 기존 워크플로우의 재사용성을 증대시켜 서비스 개발 비용을 감소시킬 수 있다. 또한 유비쿼터스 환경에서 서비스를 제공할 수 있는 제공 매체인 URC로봇에 상황인지 기반의 워크플로우 시스템을 적용하여, 인간을 위한 로봇 워크플로우 서비스 분야로 연구를 확장시킬 수 있다. 이러한 서비스를 다양한 관점에서의 검증은 위해서는 다양한 도메인을 기반으로 하는 시나리오 개발이 필요하며, 향후 실질적인 경험을 위해 상황정보를 실제로 적용하는 연구를 수행해야 한다.

6. 참고문헌

[1] Joo Han, Eun Kim, Young Cho, Jaey Choi, "A Ubiquitous Workflow Service Framework", LNCS 3983, pp. 30-39, 2006.

[2] Jun Li, Yingyi Bu, Shaxun Chen, Xianping Tao, Jian Lu,

"FollowMe: On Research of Pluggable Infrastructure for Context-Awareness", 20th International Conference on Advanced Information Networking and Applications(AINA'06), Volume 1, pp. 199-204, 2006.

- [3] By Prof. Dr. Frank Leymann, "Web Services Flow Language (WSFL 1.0)", Distinguished Engineer Member IBM Academy of Technology, IBM Software Group, May 2001.
- [4] Tony, Andrews, Francisco, Curbera, et al, "Business Process Execution Language for Web Services", BEA System. Microsoft Corp. IBM Corp., Version 1.1, 2003.
- [5] Satish, Thatte, "XLANG: Web Services for Business Process Design", Microsoft Corp., 2001.
- [6] D.Hollingsworth, "The Workflow Reference Model", Technical Report TC00-1003, Workflow Management Coalition, 1994.
- [7] Y. C. Ngeow, A. K. Mustapha, E. Goh, H. K. Low, "Context-aware Workflow Management Engine for Networked Devices", International Journal of Multimedia and Ubiquitous Engineering Vol.2, No.3, October, 2007.
- [8] Luc Clement, Dieter konig, Vinkesh Mehta, Ralf Mueller, Ravi Rangaswamy, Michael Rowley, Ivana Trickovic, "WS-BPEL Extension for People(BPEL4People) Specification Version 1.1", OASIS BPEL4People TC, 2009.
- [10] CAWL