

XPDL 기반의 워크플로우 동적 저장소 메커니즘

*정재우⁰ 심성수 안형진 박민재 문기동 **오종대 *김광훈
*경기대학교 전자계산학과 워크플로우 연구실 **청운대학교 컴퓨터과학과
kdk2el@kyonggi.ac.kr

A XPDL based Workflow Dynamic Repository Mechanism

*Jae-Woo Jung⁰ Sung-Su Sim Hyung-Jin Ahn Min-Jae Park
Ki-Dong Moon **Jong-Tae Oh *Kwang-Hoon Kim
*Dept. of Computer Science, Kyonggi University, **Chungwoon University

요 약

워크플로우 관리 시스템은 모델링에서 비즈니스 프로세스 정의 데이터를 엔진으로 전송하기 위한 방법으로 정적 저장소와 동적 저장소 방식을 이용한다. 정적 저장소를 통한 데이터 전송은 로컬한 데이터 교환에 장점이 있지만 유연성과 확장성의 문제점을 갖는다. 본 논문에서는 유연성과 확장성 문제 해결 방법으로 동적 저장소 방식을 이용하며 알려지지 않는 외부 데이터의 신뢰성을 체크하기 위하여 WfMC 표준에서 제정한 XPDL을 사용함으로써 문법적, 의미적 타당성을 검증한다. 즉 XPDL 기반의 동적 저장소 전송 메커니즘을 설계하고 신뢰성 있는 데이터 전송을 위하여 문법적, 의미적 타당성 검증에 필요한 컴포넌트를 제시하고자 한다.

1. 서 론

워크플로우 관리 시스템은 비즈니스 업무를 컴퓨터에 의해 자동으로 실행하고 관리하는 소프트웨어 시스템이다. 워크플로우 표준 기관인 WfMC는 각 기업의 워크플로우 관리 시스템 제품들이 여러 워크플로우 구현을 위한 표준 명세서를 정의하고 있으며 이 표준 명세서는 워크플로우 엔진과 다른 구성들과의 인터페이스를 제공한다. 인터페이스 1에서는 각 기업의 빌드 타임 툴들이 서로 다른 엔진들간에 프로세스 정의 데이터를 import/export 하는데 있어서 워크플로우 프로세스 정의 모델에 대한 표준 메타 모델을 제시하였고 프로세스 정의 저장소는 표준 메타 모델을 통하여 정의된 속성들을 관리하고 전송하는데 사용한다.

워크플로우 프로세스 정의 메타-모델이 정적 저장소를 통하여 프로세스 정의 데이터를 엔진에 전송하는데 있어서 로컬 지역간의 신뢰성 있는 데이터 전송이 가능하지만 데이터의 유연성과 확장성에 문제가 대두되었다. 동적 저장소 방식은 이 유연성과 확장성 문제를 해결하지만 외부 데이터에 대해 신뢰성 있는 데이터를 전송하기 위하여 문법적, 의미적 타당성을 검증해야 한다. 이 검증을 위하여 WfMC에서 제정한 XML방식 기반의 XPDL을 사용하며 동적 저장소의 신뢰성 있는 데이터 전송을 위한 메커니즘을 논의한다.

따라서 본 논문에서는 XPDL 기반의 동적 저장소를 통한 데이터의 문법적인 검증과 적합성을 논의하고 워크플로우 런타임시 워크플로우 엔진에서 요구할 수 있는 운영 컴포넌트를 설계한다.

이를 위하여 2장에서는 WfMC에서 명세한 워크플로우 인터페이스¹ 대한 언급과 정적 저장소와 동적 저장소의

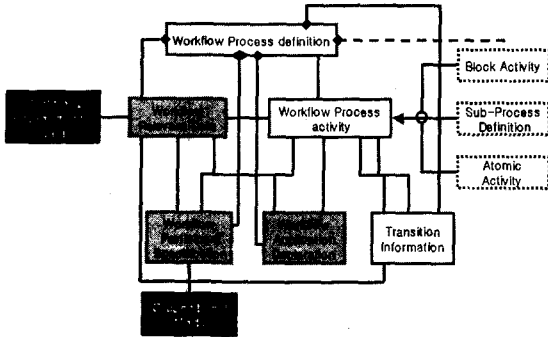
데이터 전달방식을 소개하고 정적 저장소의 문제점을 제시한다. 3장에서는 XPDL 기반의 동적 저장소 방식을 이용한 데이터 전송을 소개하고 신뢰성 있는 데이터 전송을 위하여 문법적, 의미적 타당성 검증에 필요한 메커니즘을 설계한다. 이 메커니즘을 기반으로 한 런타임시 필요한 운영들의 컴포넌트를 설계하며 4장에서는 결론을 맺는다.

2. 관련 연구

2.1 워크플로우 프로세스 정의

워크플로우 프로세스 정의는 워크플로우 관리 시스템에 의하여 모델링, 실행을 다룰 수 있게 자동화를 지원하는 어떤 형식을 가지고 있는 비즈니스 프로세스의 한 표현이다. 프로세스 정의의 구성은 액티비티들의 연결조각과, 그들의 관계, 프로세스의 시작과 끝, 그리고 참여자들, IT 어플리케이션들과 데이터의 결합과 같은 개개의 액티비티의 정보 등으로 이루어진다.

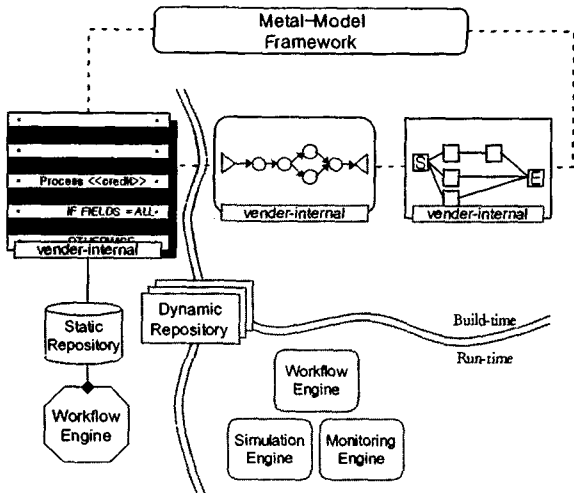
프로세스 정의는 이 정의들을 접근하고 묘사하는 일반적인 방법들을 제공하기 위해 워크플로우 프로세스 정의 메타-데이터 모델을 수립하고 이 메타-데이터 모델은 프로세스 정의 내에서 사용되는 속성들과 그 속성들의 관계 및 특성들을 분류한다. 또한 이 메타-데이터 모델은 관련된 프로세스 모델 내에서 프로세스 정의들을 그룹화 하고 서로 다른 프로세스 정의 또는 모델들에 대한 공통 정의 데이터를 사용하도록 다양한 조약들과 서로 다른 프로세스 정의들 또는 모델들에 대하여 공통 정의 데이터 사용을 정의한다. 이 모델에 기반을 둔 기업의 특정 툴들은 공통 교환 포맷을 통해 모델들을 전송할 수 있다. 다음 그림 1은 메타-모델 속성들을 보여준다.



[그림 1] 워크플로우 프로세스 정의의 메타 모델

2.2 프로세스 저장소

프로세스 정의 import/export 인터페이스는 워크플로우 관리 시스템에 연관된 어떤 형태의 워크플로우 정의 저장소로부터 운영되는 것을 전제로 한다. 즉 이 import/export 인터페이스는 기업 내부 로컬의 정적 저장소 뿐만 아니라 외부의 동적 저장소를 사용하여 프로세스 정의 데이터를 엔진으로 보낼 수 있다. 다음 그림 2는 프로세스 정의들이 정적 저장소 및 동적 저장소를 통한 프로세스 정의 데이터 전송을 보여주고 있다.



[그림 2] 프로세스 정의의 데이터 저장 방식

그림 2의 내용을 살펴보면 각 기업들은 프로세스 모델링 툴을 이용하여 프로세스 정의의 메타-모델에 기반을 둔 프로세스 정의 모델을 각 기업의 요구에 맞게 그려낸다. 프로세스 정의 데이터에 정적 저장소는 각 기업의 구조에 맞게 사용되고 있으며 로컬 워크플로우 엔진에 프로세스 정의 모델을 전송한다. 그러나 이 정적 저장소는 외부 엔진에 전송할 때 데이터의 유연성과 확장성의 문제점을 가지고 있다. 이 문제점을 해결하기 위해 동적 저장소를 통한 데이터 전송을 하며 동적 저장소는 각 기업에서 정의된 데이터들을 외부 워크플로우 엔진으로 전송하는데 사

용된다. 이 동적 저장소 방식을 통한 전송은 알려지지 않은 외부 데이터에 대한 신뢰성 있는 데이터 전송을 위해 각 기업의 프로세스 정의 모델들에 대한 문법적, 의미적 타당성 검증이 필요하다.

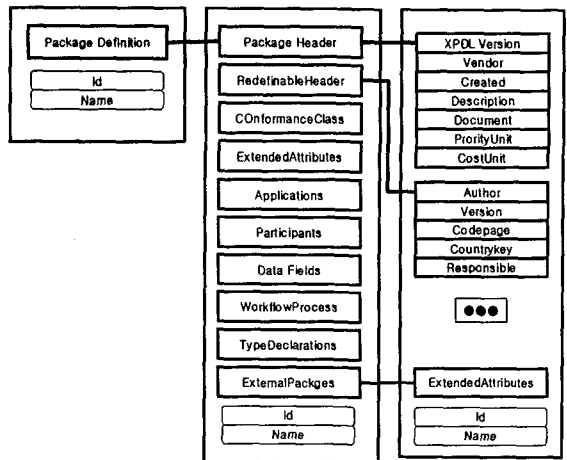
3. 동적 저장소 방식

본 논문에서는 WfMC에서 동적 저장소 방식에서 제공된 XPDL방식을 소개하고 동적 저장소 방식에서 갖는 신뢰성 문제점을 XPDL을 이용한 해결 방안을 제시한다. 이 XPDL 방식은 XML 기반인 데이터 전송 방식이며 XPDL을 통한 데이터 전송의 주요 요소들 중의 하나는 다양한 툴들에 의해 사용된 정보들을 처리할 수 있는 확장성이다. 워크플로우 프로세스 정의를 표시한 재현된 속성들에 기반을 둬으로써 XPDL은 서로 다른 접근들을 지원한다. 또 다른 XPDL의 중요한 요소는 공통 표현을 위해 사용되는 기업 특정 속성들을 지원하는 일반적인 구조로써 메타-모델을 묘사하고, 프로세서 정의 내에서 포함된 객체들과 속성들을 정의함으로써 XPDL 문법은 직접적으로 이런 객체들과 속성들에 관련된다. 이 접근은 다음과 같은 기업에 의해 제공될 두 운영들을 필요로 한다.

- XPDL부터 워크플로우 정의 import
- 내부 워크플로우 정의 표현을XPDL로 export

3-1. XPDL

XPDL 명세를 통한 동적 저장소 방식 데이터 전송에 있어서, 어떤 모델링 도구에 의해 정의된 프로세스 정의가 그 명세에 적합할 지라도 외부의 알려지지 않는 모든 정의들을 호환되는 것을 장담할 수는 없다. 즉 프로세스 정의 import/export 인터페이스가 XPDL을 통한 데이터 전송을 할 경우 XPDL이 문법적으로 타당해야 하고 의미적으로 적합했을 경우 타당한 XPDL 문서를 전송한다고 가정할 수 있다.



[그림 3] XPDL 구조

3-2. 문법적, 의미적 검증을 통한 XPDL

그림 3은 XPDL에 정의된 주요 요소들을 트리 형태로

보여준다. 이 주어진 요소는 XML-Schema를 이용하여 문서의 구조를 정의함으로써 XML 파서를 통하여XML를 검증하고 처리한다. 즉 데이터 형식이나 여러 속성값, 반복 횟수 등과 같은 검증 기능에 해당하는 규칙들을 내장 시킴으로써 문법적인 타당성을 검증한다.

```

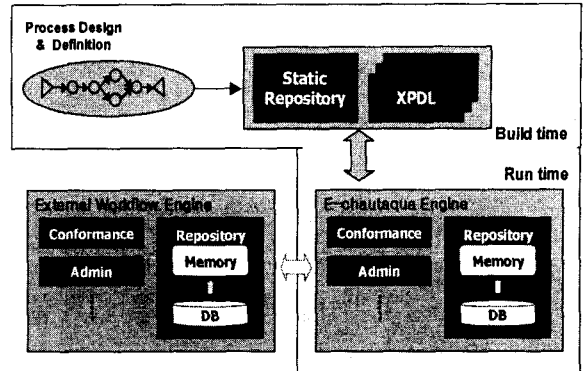
<xsd:element name="Activity">
<xsd:complexType>
  <xsd:sequence>
    <xsd:element ref="xpdl:Description" minOccurs="0"/>
    <xsd:element ref="xpdl:Limit" minOccurs="0"/>
    <xsd:choice>
      <xsd:element ref="xpdl:Route"/>
      <xsd:element ref="xpdl:Implementation"/>
      <xsd:element ref="xpdl:BlockActivity"/>
    </xsd:choice>
    ....
  </xsd:element>
  
```

[그림 4] XPDL을 위한 전체 스키마 구조 표현

그림 4와 같이 XPDL 요소들의 계층적 구조와 순서는 XML-Schema를 통한 문법적인 타당성을 검증할 수 있고 타당한 데이터 타입과 범위에 대한 요소들의 내용을 체크를 한다. 예를 들어 month 요소가 1-12의 integer로 정했을 경우 12 이상의 경우가 나오면 invalid로 체크 한다.

또한 동적 저장소를 통하여 엔진으로 데이터 전송 시 외부데이터 모델전송에 대한 의미적인 검증도 필요하다. 위에서 언급하였듯이 알려지지 않는 모든 외부 모델의 정의 데이터는 의미적 적합성을 판단하기는 쉽지 않다. 이런 의미적 검증에 효율적인 방법으로는 메타 모델의 요소 관계를 분석하여 적합성을 찾아 내는 것이다. Application, Participants, Activity, Transition, Datatypes 등의 요소들은 프로세스 메타 모델을 통하여 정의 되고 런타임시 엔진에서 처리될 속성들을 가진다. XPDL 문서에 정의된 이 요소들이 엔진으로 전송될 때 프로세스 정의 내에서 정의된 요소들이 실제적으로 문서에 정의된 요소인지를 파악함으로써 의미적 검증에 적합성을 찾아 낸다. 예를 들어 XPDL를 통하여 들어온 데이터의 정의가 엔진에서 이 응용을 실행하기 전에 그 모델에 응용이 정의 되어있는지 확인을 해야 한다. 문법적으로 검증된 XPDL문서에 정의된 요소들이 프로세스 모델에 정의되지 않는 요소들이라면 의미적으로 타당하지 않는 것이다. 이런 요소들의 관계는 정의되지 않는 속성을 찾기 위한 운영들의 컴포넌트를 형성함으로써 적합성을 검증한다.

다음 그림 5는 프로세스 정의 데이터가 XPDL를 통하여 문법적, 의미적인 타당성 검증을 위하여 컨포먼스 컴포넌트를 워크플로우 엔진에 정의함으로써 의미적 검증을 위한 워크플로우 시스템 프레임 워크를 보인다. 이 컨포먼스 컴포넌트에서 타당성이 증명되면 엔진 저장소나 메모리에 적재하여 워크플로우 관리 시스템은 신뢰성 있는 데이터를 이용 가능하게 된다.



[그림 5] Conformance 컴포넌트를 포함한 XPDL 기반의 동적 저장소 구조

4. 결론 및 향후 연구방안

본 논문에서는 워크플로우 관리시스템이 모델링에서 엔진으로 워크플로우 프로세스 정의 데이터를 전송하는데 있어서 정적 저장소가 갖는 데이터 유연성과 확장성에 대한 문제점 해결로 동적 저장소를 소개하였다. 동적저장소를 통한 신뢰성 있는 전송을 위하여 문법적인 타당성과 적합성을 문제를 제시하였고 해결하는데 XPDL를 논의하였다. 이 XPDL에 XML-스키마를 통한 검증과 의미적 적합성을 위한 운영들을 나열하였고 이 운영을 관리하는 컴포넌트를 설계하였다. 향후 이 컴포넌트 기반으로 하는 XPDL 데이터 전송을 구현할 예정이다.

Acknowledgement

본 연구는 한국과학재단 목적기초연구(R05-2-2-000-01431-0)지원으로 수행되었음

참고문헌

- [1] Workflow Management Coalition Specification Document, The Workflow Reference Model., Document Number TC00-1003,19-Jan-95
- [2] Workflow Management Coalition Specification, "Workflow Process Definition Interface - XML Process Definition Language", WfMC, October 25,2002
- [3] HangYi Kim, SeungIL Lee, Dongsoo Han, "Design and Implementation of the API for Retrieving ProcessTemplate in Workflow System"
- [4] Mr. Juergen Boldt, Workflow Management Facility Specification,V1.2, OMG Document Formal/00-05-02-Contract
- [5] Sung-Soo Shim, Kwang-Hoon Kim "Workcase based Very Large Scale Workflow System Architecture", 한국데이터베이스학회, October, 2002