

프로세스 중심 웹 콘텐츠 관리 도구의 설계

이형원, 김세웅

강릉대학교 컴퓨터공학과

e-mail : lhw@kangnung.ac.kr, sewoong98@cs.kangnung.ac.kr

Design of a Process-centered Web Content Management Tool

Hyung-Won Lee, Se-Woong Kim

*Dept. of Computer Science and Engineering, Kangnung National University

요 약

과거 소프트웨어 개발 업체들이 소프트웨어 위기를 겪었던 것과 유사하게 최근 인터넷 업체들은 웹 위기에 빠져들고 있으며 이러한 웹 위기에 대한 효과적인 해결 방안으로 최근 제시되고 있는 것이 웹 콘텐츠 관리이다. 그러나 기존의 웹 콘텐츠 관리 시스템은 콘텐츠의 생산과 배포에 초점을 두고 있는 반면 웹 콘텐츠 관리 프로세스의 지원 기능은 상대적으로 미약하다. 본 논문에서는 소프트웨어 개발 프로세스를 효율적으로 정의하고 실행할 수 있는 프로세스 중심 소프트웨어 공학 환경인 PRAISE에 웹 콘텐츠 관리 기능을 통합한 프로세스 중심 웹 콘텐츠 관리 도구의 설계 결과를 보인다.

1. 서론

과거 소프트웨어 개발 업체들이 소프트웨어 위기를 겪은 것처럼 최근 인터넷 업체들은 유사한 형태의 위기인 웹 위기에 빠져들고 있다. 특히, 웹 개발 팀들은 기존의 이슈들과 더불어 더욱 동적인 요인들에 직면하고 있다. 이러한 웹 위기의 원인들을 가장 효과적으로 해결할 수 있는 방법으로 최근 제시되고 있는 것이 웹 콘텐츠 관리(WCM)이다[1,2].

웹 콘텐츠 관리 정책이 가져다주는 혜택을 성공적으로 누리기 위해서는 웹 콘텐츠 관리 기법의 자동화와 웹 콘텐츠 관리 프로세스의 자동화가 필수적이거나 기존 웹 콘텐츠 관리 시스템(WCMS)들은 콘텐츠 생성과 배포에 초점을 두고 있는 반면 프로세스 지원 기능은 매우 미약하여 다음과 같은 문제를 해결하지 못하고 있다. 첫째, 웹 개발 및 유지보수에 적합한 명확하게 정의된 웹 콘텐츠 관리 프로세스가 없으며 이는 조직 내의 웹 콘텐츠 관리 실행의 비효율성을 가져오게 된다. 둘째, 웹 콘텐츠 관리 프로세스를 쉽게 정의하고 수정하고 개선할 수 없으며 콘텐츠 관리 기능의 자동화 수준을 높이는 데 한계가 있다. 셋째, 콘텐츠 관리는 전체 웹 개발/유지보수 프로세스와 강하게 연결되어야 하며 콘텐츠 관리는 전체 프로세스 내에서 이해되고 실행되어야 한다. 그러나 전체 웹 개발 프로세스의 자동화와 콘텐츠 관리 프로세스의 자

동화를 연계시킬 효율적인 방법이 없다.

이러한 문제점들을 극복하는 방법은 소프트웨어 개발 프로세스를 효율적으로 정의하고 실행할 수 있는 PSEE, 즉 프로세스 중심 소프트웨어 공학 환경에 콘텐츠 관리 기능을 강하게 연계함으로써 웹 개발의 효율성과 생산성을 극대화하는 것이다. 본 논문에서는 PSEE PRAISE[3]에 웹 콘텐츠 관리 기능을 통합한 프로세스 중심 웹 콘텐츠 관리 도구의 설계 결과를 보인다.

본 논문의 2장에서는 기존 WCMS에 대해 분석하고 3장에서는 본 시스템 개발의 기반이 되는 메타 모델에 관하여 기술한다. 4장에서는 시스템 구조를 설명하고 5장에서는 간단한 실행 예를 보인다. 마지막으로 6장에서 결론 및 향후 연구과제에 대해 논한다.

2. 기존 WCMS

현재 상용화된 많은 WCMS 제품[4,5,6,7]들의 기능의 초점은 템플릿, 콘텐츠의 생산, 관리, 재사용, 다른 형태로의 변환, 배포기능 등에 초점을 두고 있다. 기존 WCMS는 콘텐츠를 생성, 조작하는 기술은 매우 뛰어나지만 콘텐츠 생성에서 폐기까지 적용되는 프로세스는 상당히 제한적이다. 예를 들어 Microsoft Content Management Server[4]의 경우 콘텐츠가 생산 및 두 단계의 승인을 거쳐 출판되는 고정된 프로세스를 지원하고 Documentum 4i[5]의 경우 자료 수집 혹은 기존 콘텐츠, 생산, 검토/승인, 완성/변환, 출판의 프로세스를 거친다. 또한 생산의 단계에서 사용

본 연구는 정보통신부 정보통신연구진흥원의 IT기초기술연구지원 사업(과제번호:05-기초-001)으로 수행한 연구 결과임.

자가 정의 할 수 있는 프로세스를 지원하며 좀더 폭 넓고 유연한 프로세스 모델을 지원하지만 콘텐츠 생성에서 폐기까지 모든 작업 중 생성의 경우만 사용자가 프로세스를 정의 할 수 있다. 따라서 사용자는 조직에서 필요로 하는 워크플로우를 포함하고 있는 제품을 선택하거나 필요한 프로세스를 수동으로 적용하여 사용한다. 이상과 같이 대부분의 기존 도구들은 미리 고정된 워크플로를 지원하기 때문에 개발 조직이나 개발 대상 웹 사이트에 따른 다양한 웹 콘텐츠 관리 프로세스의 구축에는 적합하지 않다.

3. 메타 모델

본 시스템 개발에 근거가 되는 메타 모델은 작업자들의 권한, 유형, 속성의 관리를 위한 작업자 메타 모델, 웹 개발동안 생성되는 콘텐츠 관련 정보들의 관리를 위한 웹 콘텐츠 메타 모델, 프로세스 정의와 실행을 위한 웹 콘텐츠 관리 프로세스 메타 모델로 구성된다.

3.1 작업자 메타 모델

작업자 메타 모델은 웹 개발에 참여하는 작업자의 관리를 위한 것으로 권한 모델, 작업자 타입 모델, 작업자 모델로 구성된다. 권한 모델에서는 네 가지의 유형이 있으며 각 유형 별로 세분화된 권한 레벨이 정의된다. 권한 유형에는 1) 작업자의 추가, 삭제, 유형 정의 권한, 2) 특정 프로세스 혹은 모든 프로세스 실행 권한, 3) 자신이 실행시킨 프로세스의 중단 혹은 모든 프로세스의 중단 권한, 4) 할당받은 콘텐츠의 확인 혹은 모든 콘텐츠의 확인 권한이 있다. 작업자 타입 모델은 권한 모델에 정의된 행위에 대한 권한 설정 정보를 작업자의 유형별로 정의하기 위한 모델이다. 작업자 모델은 각 작업자들의 기본적인 정보를 관리하기 위한 속성들로 작업자의 ID, 패스워드, 이름, 작업이 할당될 경우 작업지시를 받을 메일, 작업자의 유형과 작업자의 권한을 설정하는 속성으로 구성된다.

3.2 웹 콘텐츠 메타 모델

웹 콘텐츠 메타 모델은 웹 콘텐츠 관련 정보를 관리하기 위한 콘텐츠 모델과 웹 콘텐츠 데이터베이스에 접근하기 위한 정보를 저장하는 목적시스템 모델로 구성된다. 첫째, 콘텐츠 모델을 살펴보면 각 웹 콘텐츠는 일련의 프로세스를 거쳐 완성되며 각 프로세스 스텝에서 이루어지는 행위와 결과를 별도로 관리할 필요가 있다. 콘텐츠 모델은 콘텐츠 관련 정보 및 해당 콘텐츠의 동적인 변경을 가하는 행위와 관련된 정보를 정의한다. 콘텐츠 모델은 콘텐츠의 ID와 버전, 콘텐츠와 관계된 프로세스와 관계된 스텝의 정보, 실제 콘텐츠가 저장되는 테이블 정보, 스텝을 수행하는 작업자 정보, 프로세스 시작, 종료 시간 데이터 등으로 구성된다. 둘째, 목적시스템 모델은 웹 콘텐츠에 대한 실제 정보에 접근하는데 사용되는 속성들의 집합이며 해당 DB에 접근하기 위한 권한 정보와 관리 대상이 DB 데이터가 아닐 경우 파일의 경로 등이 저장되며 이를 통해 웹 콘텐츠 관리 도구가 웹 콘텐츠의 구조와 내용에 접근하게 된다.

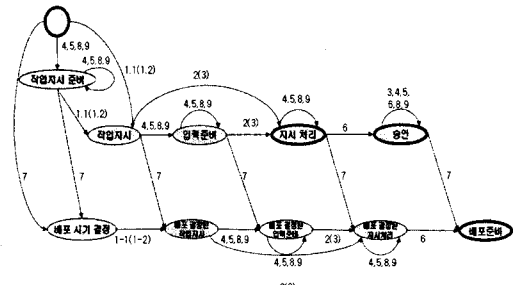
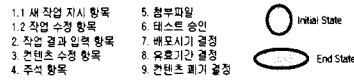
3.3 웹 콘텐츠 관리 프로세스 메타 모델

웹 콘텐츠 프로세스 관리 모델은 프로세스의 기본

요소가 되는 웹 콘텐츠 관리 프로세스 스텝의 유형 및 속성 항목들의 집합과 프로세스 실행과 관련된 데이터베이스 스키마로 구성된다.

3.3.1. 프로세스 스텝과 프로세스 항목

프로세스 항목은 작업자가 웹 콘텐츠를 다루며 설정하거나 지시하는 일들의 최소 단위이다. 작업자는 프로세스에 정의된 순서에 따라 한 단계에서 하나 혹은 그 이상의 항목들을 처리하게 되며 프로세스를 진행하며 콘텐츠는 처리된 항목에 따라 그 상태가 변화하게 된다. 입력된 항목에 따른 콘텐츠 상태 전이를 이용하여 주요 웹 콘텐츠 관리 프로세스 스텝들을 식별하고 그들과 동적인 속성들 간의 상호연관성을 추출하기 위해 속성에 따른 웹 콘텐츠의 상태 변화를 (그림 1)과 같은 STD(state transition diagram)으로 정의하고 분석하였다. 이 STD는 사용자가 정의한 프로세스의 오류를 검증하고 진행 중인 스텝의 현재 상태를 알 수 있는 프레임워크로도 사용된다.



(그림 1) 콘텐츠의 STD

(그림 1)의 작업지시 항목, 작업결과 입력 항목, 작업 수정 항목, 테스트 항목은 각 항목간의 제약 관계가 명확하며 배포 시기 결정 항목만이 어느 상태에서나 입력이 가능하다. 따라서 제약 관계가 명확한 세 가지 항목을 기준으로 스텝을 구분하고 다른 항목을 스텝에 추가하여 다음과 같은 다섯 가지 미리 정의된 스텝을 정의하였다.

- 1) 새 작업 할당 스텝: 콘텐츠의 생성을 지시하는 스텝으로 작업 수행에 필요한 사항들을 준비하고 결정하는 역할을 담당한다.
- 2) 수정 작업 할당 스텝: 콘텐츠의 수정을 지시하는 스텝으로 작업 수행에 필요한 사항들을 준비하고 결정하는 역할을 담당한다.
- 3) 콘텐츠 편집 스텝: 실제로 콘텐츠를 생성하거나 수정하는 스텝으로 해당 콘텐츠 관련 전문가에 의해 수행된다. 콘텐츠 편집 스텝은 실행 순서에 있어 새 작업 할당 스텝이나 수정 작업 할당 스텝 이후에 정의되어야 한다.
- 4) 검토 스텝: 생성되거나 수정된 콘텐츠를 검토하는 스텝으로 실행 순서에 있어 콘텐츠 편집 스텝 이후에 정의되어야 한다.
- 5) 테스트 스텝: 생성되거나 수정된 콘텐츠를 테스트하는 스텝으로 경우에 따라서는 콘텐츠를 수정할 수

도 있으며 실행 순서에 있어 콘텐츠 편집 스텝 이후에 정의되어야 한다.

이러한 미리 정의된 스텝들과 과 프로세스 항목들을 바탕으로 프로세스 모델 작성자는 프로세스를 정의할 때마다 스텝의 많은 정보를 선택하는데 드는 시간과 노력을 줄일 수 있으며 미리 정의된 스텝으로 부족한 부분은 사용자 정의 스텝을 통하여 추가함으로써 프로세스 정의에 유연성과 편의성을 극대화 하였다.

3.3.2 프로세스 정보저장소

프로세스가 진행되는 동안 웹 콘텐츠 관리 프로세스 메타 모델에 따른 프로세스 정보의 관리가 이루어지며, 프로세스 관련 정보들은 프로세스 정보저장소에 위치하게 된다.

프로세스 테이블

프로세스 테이블은 프로세스 모델 저장 시 해당 프로세스 모델에 대한 메타 정보를 저장하기 위한 공간이며, 프로세스의 ID, 이름, 소유자, 작성자의 프로세스 정보와 프로세스 시작 작업자, 실행횟수, 실패횟수, 최소 실행시간, 최대 실행시간, 평균 실행시간 등 프로세스에 관련된 메타정보를 저장한다.

스텝 테이블

스텝 테이블은 프로세스 모델 편집기에서 프로세스 모델이 저장될 때 프로세스 스텝들에 대한 메타 모델을 저장하기 위한 공간이다. 저장되는 메타데이터는 프로세스와 스텝의 ID, 스텝을 실행할 작업자의 타입과 작업자의 ID, 스텝의 이름과 유형 등이다.

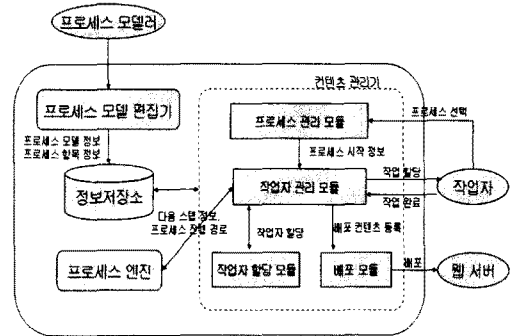
스텝 인스턴스 테이블

사용자가 프로세스를 정의하고 저장하면 프로세스 테이블과 스텝 테이블에 레코드들이 생성된다. 이후 콘텐츠 관리기는 각각의 스텝에서 처리된 결과를 저장할 필요가 생긴다. 즉, 스텝 실행과 독립된 스텝 자체에 대한 정보는 스텝 테이블에 저장하게 되고 프로세스 실행 시의 스텝 인스턴스에 관한 정보는 별도의 테이블을 동적으로 생성하여 저장하며 ID, 프로세스 ID와 스텝이 할당된 시간, 마감 요청시간, 종료된 시간과 작업내용, 첨부파일, 배포일정, 유효기간, 콘텐츠 저장여부를 포함한다.

4. 시스템 구조

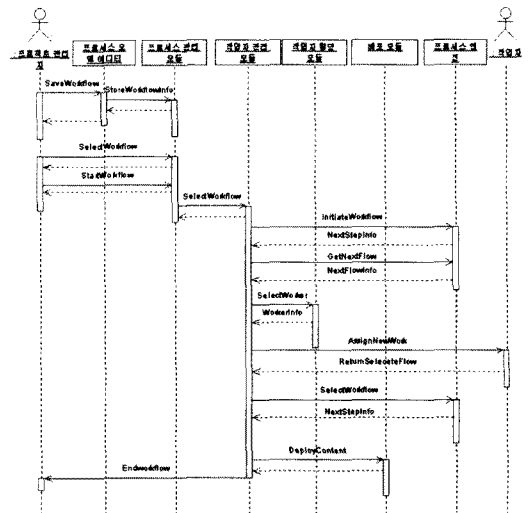
본 프로세스 중심 웹 콘텐츠 관리 도구의 구조는 (그림 2)와 같다. 프로세스 모델 작성자가 프로세스 모델 편집기를 이용하여 원하는 웹 콘텐츠 관리 프로세스를 프로세스 모델링 언어 RAISE[8]의 문법에 따라 작성한 후 저장하면 프로세스 모델 정보와 프로세스 항목 정보가 정보저장소에 저장된다. 정보저장소는 웹 콘텐츠 관리 프로세스 전반에 걸쳐 발생하는 정보들을 저장하기 위한 장소이다. 웹 콘텐츠 관련 정보를 위한 콘텐츠 정보저장소와 프로세스 관련 정보를 위한 프로세스 정보저장소로 논리적으로 분리되어 있으며 프로세스 실행 과정에서 콘텐츠 관리기에 필요한 정보를 제공하고 콘텐츠 관리기로부터의 입력 정보를 저장한다. 프로세스 엔진은 규칙 기반 전문가 도구 중 하나인 Jess[9]로 구현된 규칙의 집합으로 Jess 추론 엔진을 바탕으로 실행되며 프로세스 모델에 정의된 내용에 따라 다음 단계에 수행할 스텝들을 결정하게 되며 그 결과를 콘텐츠 관리기의 작업자 관

리 모듈에 전달한다. 콘텐츠 관리기는 웹 콘텐츠의 관리를 위한 핵심 구성 요소로 프로세스 관리 모듈, 작업자 할당 모듈, 작업자 관리 모듈, 배포 모듈로 구성되어 있다. 프로세스 관리 모듈은 실행될 프로세스의 ID를 작업자 관리 모듈에 전달하고 콘텐츠 관리에 필요한 웹 서버의 테이블과 프로세스 진행 중 필요한 프로세스 스텝의 실행 결과를 가지고 있는 테이블들을 생성하게 된다. 작업자 할당 모듈은 각 프로세스 스텝을 직접 수행할 작업자를 선정하는 역할을 담당한다. 작업자 관리 모듈은 첫째, 진행 중인 모든 프로세스들을 관리하며 둘째, 프로세스 엔진, 작업자 할당 모듈, 배포 모듈과 통신하며 셋째, 작업자에게 프로세스 진행의 다음 스텝과 작업자가 해야 할 일을 전달하며 넷째, 스텝이 종료되고 난 후 그 정보를 프로세스 엔진에게 전달하여 프로세스의 다음 스텝 정보를 얻어온다. 배포 모듈은 웹 콘텐츠의 배포를 담당하는데 프로세스가 종료되면 배포 모듈을 통하여 콘텐츠가 콘텐츠 결집 영역에서 웹 서버로 배포된다.



(그림 2) 시스템 구조

구성 요소 간 시퀀스 다이어그램이 (그림 3)에 있으며 다음은 간략한 설명이다.

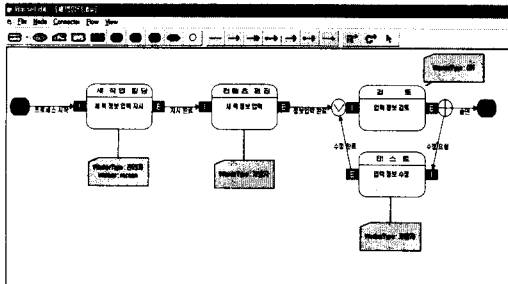


(그림 3) 시스템 구성 요소 간 시퀀스 다이어그램

- 1) 프로세스 모델 편집기를 통하여 프로세스 모델을 정의하고 이를 프로세스 관리 모듈에게 전달한다.
- 2) 프로젝트 관리자는 프로세스 관리 모듈을 통해 프로세스를 시작시킨다.
- 3) 프로세스 관리 모듈은 작업자 관리 모듈에게 프로세스가 실행됨을 알린다.
- 4) 작업자 관리 모듈은 프로세스 엔진에게 프로세스 모델을 실행시키고 다음 스텝 정보를 얻어온다.
- 5) 작업자 관리 모듈은 프로세스 엔진에게 다음 진행 경로 목록을 얻어온다.
- 6) 작업자 관리 모듈은 실행시킬 스텝의 작업자 정보를 작업자 할당 모듈에서 얻어온다.
- 7) 작업자 관리 모듈은 작업자에게 작업을 할당하고 작업 결과와 다음 프로세스 진행 방향을 얻어온다.
- 8) 작업자 관리 모듈은 프로세스 엔진에게 다음 프로세스 진행 방향을 알려주고 프로세스 엔진으로부터 다음 스텝 정보를 얻어온다.
- 9) 프로세스가 종료되었으면 배포 모듈로 배포될 콘텐츠 정보를 입력한다.

5. 실행 예

(그림 4)는 새 도서 정보를 입력하는 간단한 웹 콘텐츠 관리 프로세스를 PRAiSE 프로세스 모델 편집기에서 생성한 결과이다.

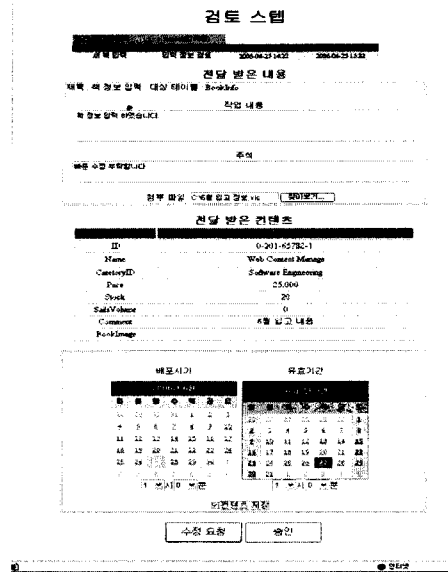


(그림 4) "새책입력" 프로세스

(그림 5)는 "새책입력" 프로세스의 "입력 정보 검토" 스텝에 대한 웹 페이지가 자동적으로 생성되어 선정된 작업자에게 전달된 후 작업을 수행한 결과가 입력된 모습이다. 만약 책 제목의 오류가 발견되어 작업자가 "수정 요청" 버튼을 눌렀다고 가정하면 "입력 정보 검토" 스텝은 종료되고 그 결과는 작업자 관리 모듈에게 전달된다. 작업자 관리 모듈은 프로세스 엔진에게 스텝 수행 결과를 전달하고 다음에 실행할 스텝이 테스트 타입의 "입력 정보 수정" 스텝이라는 정보를 얻어온다.

6. 결론

본 논문에서는 웹 개발 과정에서 웹 콘텐츠 관리가 자연스럽게 이루어질 수 있는 환경을 제공하기 위하여 PSEE PRAiSE를 기반으로 하여 여기에 웹 콘텐츠 관리 기능이 통합된 프로세스 중심 웹 콘텐츠 관리 도구에 대한 설계 결과를 기술하였다. 이러한 접근 방법은 기존 웹 콘텐츠 관리 도구에 비해 다음과 같은 장점을 갖는다. 첫째, 그래픽 프로세스 언어



(그림 5) 작업자 화면

를 이용하여 웹 콘텐츠 관리 프로세스를 정의하며 둘째, 유연한 프로세스 커스터마이징과 변경이 가능하고 셋째, 프로세스 정의, 프로세스 실행, 콘텐츠 관리 기능의 분리하였으며 넷째, 기존 메일링 프로그램을 이용하여 개발자와의 통신이 이루어진다.

현재 프로토타입을 개발 중에 있으며 향후 연구 과제로 이를 상용화하기 위한 노력이 필요하며 성능 개선을 위한 다양한 시도가 이루어져야 할 것이다.

참고문헌

- [1] Susan Dart, Configuration Management, The Missing Link in Web Engineering, Artech House, 2000.
- [2] Russel Nakano, Web Content Management, Addison-Wesley, 2002.
- [3] 이형원, 이승진, "PRAiSE: 규칙 기반 프로세스 중심 소프트웨어 공학 환경," 정보과학회논문지:컴퓨팅의 실제, Vol.11, No.3, pp.246-256, June 2005.
- [4] Bill English, Olga Londer, Shawn Shell, Todd Bleeker, Stephen Cawood, Microsoft Content Management Server 2002, Addison Wesley, 2004.
- [5] 다크멘텀 콘텐츠 관리 시스템, <http://pentasys.com/kr/>, Oct. 2004
- [6] Vignette Process Workflow Modeler, <http://www.vignette.com/>
- [7] Interwoven TeamSite Content Management Server, <http://www.interwoven.com/>
- [8] 이형원, "RAiSE: 다양한 의미론과 사용의 용이성을 제공하는 그래픽 프로세스 모델링 언어," 정보처리학회논문지D, Vol.12-D, No.7, Dec. 2005.
- [9] E.J. Friedman-Hill, "Jess 7.0 Manual," <http://jessrules.com/jess/docs/70/>, Sep. 2006.