

XML 기반 문제 은행 Framework

김연정, 조동섭
이화여자 대학교 컴퓨터학과

The Item Pool Framework based on XML

Kim Yen-Jung, Cho Dong-Sub
Dept. of Computer Science and Engineering,
Ewha Womans University

요 약

기존에 문제 은행 솔루션은 이미지 형태 또는 데이터가 응용 제품과 독립적이지 못하는 한계를 가지고 있다. 기존에 생산되었던 콘텐츠를 활용하면서도 응용제품에 독립적인 문제은행 시스템을 만들기 위해서 XML을 이용한 데이터베이스의 구축이 필요하다. 이를 위하여 데이터를 XML로 전환하고 이렇게 전환된 XML 데이터를 XSL을 통해 표현할 수 있어야 한다. 본 논문은 기존의 문제은행 방식의 문제를 해결하기 위해 XML을 기반으로 하는 문제은행을 구축하는 방법을 알아보도록 하겠다.

1. 서론

국내의 온라인 시장은 인터넷이 발전함에 따라 수요가 매우 커지고 있으며 이런 수요에 맞추기 위해 많은 데이터가 필요로 되어진다. 이런 온라인 서비스를 제공하기 위해서 필요로 하는 많은 데이터를 데이터베이스로 효율적으로 구축하는 것이 저장 검색을 통한 재사용에 중요한 점이 되고 있다.

현재 우리가 사용하는 데이터 대부분이 기존의 오프라인에서 사용되었던 데이터들이다. 기존에는 이 데이터들을 스캔을 통한 이미지나 단순한 데이터로 문제은행으로 구축했다. 이러한 문제은행 솔루션들은 HTML로 제작하거나 RDB의 테이블 형태로 데이터를 구축함으로써 기존의 출판물을 재활용하지 못해 온라인과 오프라인이 유기적으로 연동되지 못하는 문제점을 안고 있다.

이 문제점을 해결하기 위해서는 이미 가지고 있는 오프라인의 데이터를 재사용하여 온라인과 오프라인을 유기적으로 통합하는 XML기반의 문제은행솔루션이 필요하다.

기존의 문제은행 솔루션은 이미지 형태 또는 데이터가 특정 응용 제품에 종속되는 상태로 문제은행을 구축함으로써 정보의 재활용 시 정보의 손실이 많았으며, 기존 출판물을 재활용하지 못하는 문제를 가지고 있다. 콘텐츠를 관리하고 재활용하려는 입장에서 보면 특정 제품에 종속되는 데이터는 정보의 혼란도가 높아 국제적으로 통용되는 XML에 기반한 콘텐츠보다 재활용성이 매우 낮다고 할 수 있다. XML에 기반한 문제은행 솔루션은 데이터를 조직화/정규화/지식화 하는 효과를 가지게 되며, 기존 출판물을 통해 자동으로 생성됨으로써 데이터의 재활용성과 함께 확장성을 제공한다.

2장에서는 문제은행 시스템이란 무엇이며 기존 문제은행 시스템의 한계를 알아보고 3장에서는 현재 문제은행 시스템의 대안인 XML 시스템에 대해 알아보겠다. 4장에서는 XML을 기반으로 하는 문제은행 시스템의 개발 단계를 알아보고 5장에서는 결론과 향후 연구 과제를 알아보겠다.

2. 문제은행 시스템

문제 은행 시스템이란 사용자의 수준과 난이도에 맞는 문제를 온라인에서 제공하고 사용자가 틀린 문제

본 연구는 2002년도 두뇌한국 21산업에 의하여 지원되었음.

에 대한 평가와 점수를 실시간으로 서비스하며 해당 영역에 대한 자료를 제공하는 시스템을 말한다. 콘텐츠를 기존에 오프라인에서 제공되었던 것을 기반으로 한다.

온라인 문제 은행 솔루션이 이전에도 존재하고 있었으나 현재의 문제 은행 솔루션은 HTML로 제작되거나 RDB 테이블 형태로 데이터를 구축함으로써 기존의 출판물을 재활용하지 못하는 단점을 가지고 있다. 이러한 점은 온라인과 오프라인이 유기적으로 연동하지 못하는 문제점을 발생시킨다.

기존의 온라인 문제 은행 솔루션은 시험지를 스캔하여 이미지 형태로 제공하거나 HTML로 시험지를 작성해 해당 문제를 제공하고 있다. 이러한 시스템은 정적인 시스템으로 여러 가지 기능적 한계를 가지고 있다. 이러한 기능적 한계로는 우선 사용자 수준에 맞는 문제 서비스와 문제 조합을 통한 다양한 문제지 생성이 어렵다는 점을 들 수 있다. 또한 기존 출판물의 재활용이 어려우며 HTML의 한계로 인한 정보 손실 또한 많다. 문제 작성과정에서도 출판물을 HTML로 코딩해야 하기 때문에 모든 콘텐츠를 변환하는데 상당한 시간이 필요하다. 또한 문제가 발생할 때마다 다시 HTML로 시험지를 작성해 서비스 할 수밖에 없다. 관리 측면에서도 파일 시스템으로 관리함에 따라 데이터 양이 늘어날수록 관리와 문제에 대한 검색 어려워진다.

HTML로 제작했을 경우의 한계점은 HTML은 표현할 수 있는 태그가 한정되어 있다는 점과 HTML은 오프라인상의 정보를 표현하는데 손실이 많다는 것이다. HTML은 사용자 요구에 따른 동적인 문서 생성이 어렵다. HTML은 다른 데이터로의 변환이나 데이터베이스화가 또한 어렵다. 이러한 한계를 극복하기 위해 자바 스크립트나 DHTML 등을 좀더 동적인 사이트를 구축하려 하지만 HTML의 근본적인 한계를 남아있다.

기존의 오프라인에 있는 콘텐츠를 그대로 사용하여 사용자 수준에 맞게 임의로 문제를 조합해 실시간으로 서비스하는 시스템을 제한 하고자 한다. 대부분은 오프라인에서는 매킨토시용 워드프로세서인 퀵익스프레스를 사용해 디자인하고 해당 콘텐츠를 작성한다. 이렇게 기존의 퀵 파일을 그대로 활용해 웹으로 서비스할 수 있는 시스템을 만드는 것이다.

이런 문제점들을 해결하기 위해서는 문제가 가지고 있는 고유한 속성(난이도, 영역)등 데이터로 유지하고 해당 콘텐츠를 데이터베이스화할 수 있어야 한다.

또한 웹으로 서비스하기 위한 스타일링을 자유롭게 적용해야 하며 사용자 검색 조건에 따라 자유롭게 영상, 오디오, 비디오와 같은 많은 저장 공간을 필요로 하는 디지털 데이터의 사용이 일반화되고 있다. 또한 디지털 데이터의 복사는 아날로그 데이터의 복사와 달리 여러 번의 복사를 수행하여도 데이터의 품질이 손상되지 않으며, 인터넷과 같은 컴퓨터 네트워크를 통한 디지털 데이터의 복사는 전파의 속도가 빠른 뿐만 아니라 그 범위도 매우 넓고, 누구든지 동일한 품질의 디지털 데이터를 이용할 수 있다는 장점을 가지게 된다. 그러나, 이러한 복사의 용이성은 저작권 보호의 관점에서 볼 때 문제가 발생한다. 즉, 많은 시간과 노력을 투자하여 제작한 영상, 오디오, 비디오 디지털 데이터가 저작자의 허가 또는 적절한 대가의 지불 없이 불법적으로 복사되어짐에 따라 저작권이 보호받지 못하는 경우가 발생하게 되고, 결국 저작자는 자신의 저작물을 공개하기를 꺼리는 바람직하지 못한 결과를 초래하게 된다.

문제를 조합해야 한다. 이러한 요구 조건을 만족시키기 위해서는 데이터 타입을 XML로 하는 것이 가장 적합하다.

3. XML 시스템

XML 시스템은 그림1와 같은 구성을 가지고 있다. XML 시스템은 다음과 같은 4가지 부분으로 구성되었다. 첫 번째는 XML 도큐먼트를 오쏘링하고 파싱하는 부분이다. XML 콘텐츠 매니지먼트 그리고 XML 프리젠테이션 과 XML 검색이 있다.

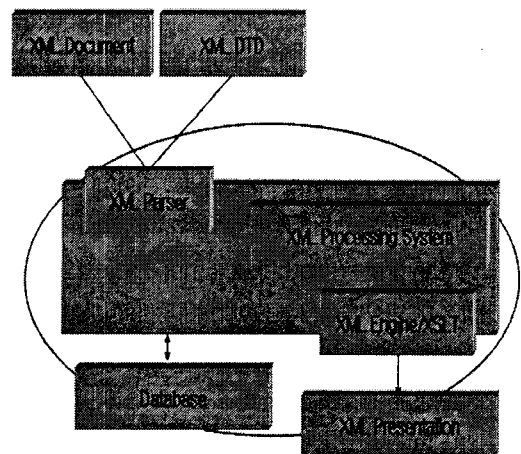


그림 1 XML 시스템

도큐먼트 오소링/파싱은 문서를 생성하고 해당 문서의 밸리데이션을 체크하는 부분이다. 문서 생성은 여러 가지 방법이 있는데 XML 전용 에디터나 TO-XML 변환 프로그램을 통해 생성할 수 있고 데이터 지향 XML의 경우 특정 문서 구조에 맞춰 애플리케이션 프로그램으로 생성할 수 있다. 문서의 파싱은 XML 파서를 통해 이뤄지며 해당문서와 DTD(Document Type Definition)를 비교해 밸리데이션을 체크한다. 밸리데이트(Validate)한 XML 문서는 콘텐츠 관리기를 거쳐 데이터베이스에 저장된다.

프리젠테이션 단계에서는 데이터베이스에서 가져온 XML을 XSLT를 통해 웨이나 CD-ROM, 종이 등으로 내놓는다. XML 검색은 XML에 문서에 대한 엘리먼트, 애트리뷰트 검색 등의 구조화 검색 시스템을 말한다.

4. XML 문제은행 시스템

그림 2는 시스템 개발 단계를 보여 주는 그림이다. 이러한 과정을 통해 시스템 개발이 이루어진다. 한 단계씩 자세히 알아보도록 하겠다.

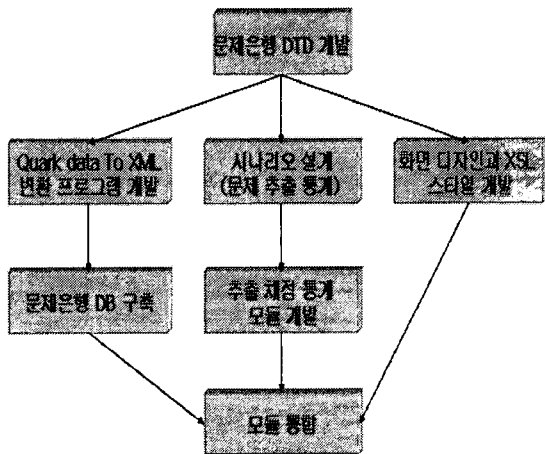


그림 2 시스템 개발 단계

문제은행 시스템 구성은 서비스 대상을 온라인을 통해 게임에 참가하는 사용자를 대상으로 하며 출판사에서 작성한 퀴크 데이터 파일을 XML로 변환해 데이터베이스로 구축한다. 기존의 퀴크 익스프레스로 작성된 파일을 Xtension 프로그램을 이용해 XML 문서로 변환한다. Xtension 프로그램은 미리 개발된 DTD에 맞게 퀴크의 해당 객체들에 태그를 붙여 xml 파일로 변환한다. 생성된 XML 파일은 CMS(Component

Management System)의 벌크 임포트(Bulk Import) 유틸리티를 이용해 데이터베이스에 저장하게 된다. 저장된 데이터는 추출 재검, 통계 등의 로직을 이용해 CMS와 서로 통신하게 한다.

XML 문서를 저장 관리하기 위한 응용 소프트웨어가 CMS이다. CMS는 이런 저장과 데이터 핸들링에 관한 일련의 과정을 처리할 수 있는 다양한 기능을 제공하며 XML 문서는 루트 엘리먼트를 최상위로 하는 트리 구조를 갖는다.

XML의 구조적 특성을 이용한 검색과 버전 관리를 위해서는 문서를 트리 구조로 저장 관리하는 것이 가장 효과적이다. 따라서 대부분의 CMS는 스토리지로 객체지향 데이터베이스를 사용하고 있다. CMS를 데이터베이스에 매우 의존적이기 때문에 어떤 스토리지를 사용하느냐에 따라 구현 방식이 매우 달라지게 된다. CMS의 일반적인 기능은 XML 파서를 내장하고 외부 엔티티를 저장 관리하는 것이다. 그리고 개발자를 위한 API를 제공한다. XML 문서의 색인과 검색을 위한 구조 검색 시스템 연동 가능하다. 사용자 권한에 의한 문서 생성 삭제 열람할 수 있다. 문서 Import와 Export 인터페이스를 가지고 있으며 문서 체크인 아웃 인터페이스(문서 버전 관리)를 가지고 있다.

문제은행 시스템 개발에 있어서 가장 우선적으로 문제은행 DTD 개발이 선행되어야 한다. DTD개발이 완료되면 퀴크 데이터를 XML로 변환하는 프로그램과 서비스 시나리오 설계화면 디자인과 XSL 스타일개발이 이뤄진다.

앞에서 말한 바 같이 XML의 DTD는 매우 중요하다. 문제 은행 DTD는 각 과목별 문제집을 분석해 공통으로 적용 가능한 엘리먼트 구조를 추출했다. 문제를 검색하기 위한 메타데이터는 속성 값으로 처리했는데 문제를 검색하는 조건은 여러 가지 있을 수 있다. 실제 스타일에서 처리하기 위해 필요한 정보가 있는데 이 정보들은 속성으로 DTD 작성시 고려되어야 한다.

개발된 DTD를 기본으로 퀴크 데이터-to-XML 변환 프로그램을 만들게 된다. 퀴크로 문서를 XML로 변환하기 위해서는 퀴크 데이터에 대한 사전 작업이 필요하다. 이 작업은 각각의 퀴크파일을 일정 포맷에 맞게 정리해 주는 작업이다. 퀴크 파일을 정리한 뒤 퀴크의 문서 객체에 문항 DTD의 해당 태그를 붙여 .xml 파일로 변환하는 프로그램을 개발한다.

퀴크로 작성된 파일을 XML 로 변환한 후 해당 문서를 데이터베이스에 저장한다. 저장 과정은 CMS 벌크 임포트 유틸리티를 이용해 이뤄진다. CMS는 먼저

문서를 내장된 XML 파서를 이용해 파싱하고 밸리데이트한 문서에 대해서만 임포트를 실시한다. 밸리데이트 검사가 끝나면 문서의 선언부에서 해당 문서와 외부 엔티티를 읽어 블롭(BLOB) 타입의 객체로 데이터베이스에 저장한다. 그 다음 문서 본문을 DOM트리 형태로 분해한 후 각각의 엘리먼트와 속성을 하나의 객체로 데이터베이스에 저장하게 된다. 이는 XML이 갖는 가장 큰 특징이라 할 수 있는 구조화 문서의 특성을 살리기 위해서다.

시나리오는 문제를 추출해 해당 사용자가 문제를 푼 다음 채점한 후 채점된 결과를 확인하는 일련의 과정으로 설계한다. 적용 설계는 UML을 이용해 사용자 케이스(Use-Case)를 정의하고 이에 맞는 시퀀스 다이어그램을 적용한다. 가장 큰 시나리오 흐름은 문제를 추출하고, 채점하고, 통계 처리를 위한 설계이다. DTD 개발을 완료하고 해당문서에 대한 샘플 문서를 완성하면 이를 바탕으로 스타일을 적용한다. 문제는행에 사용되는 XML 도큐먼트는 문제가 갖는 특수한 성질을 갖는다. 주관식, 객관식으로 나뉘질 수 있으며 서로의 연관성을 찾는 연결형 문제도 있을 수 있다. 이러한 일련의 문제 갖는 속성을 분별해 해당 스타일을 적용하는 것이 문제은행 시스템 스타일링에서 중요하다. 작성된 XSL 파일은 시험지의 유형에 따라 XSLT엔진을 이용해 HTML로 바뀌어진다.

5. 결론

지금까지 사용자의 수준과 난이도에 맞는 문제를 온라인에서 제공하고 사용자가 틀린 문제에 대한 평가와 점수를 실시간으로 서비스하며 해당 영역에 대한 자료를 제공하는 문제은행 시스템이 무엇인지 알아보고 기존 문제은행 솔루션의 문제점이 무엇인지 알아보았다. 그리고 그것의 대안으로 각 문제가 가지고 있는 고유한 속성을 데이터로 유지하고 해당 콘텐츠를 데이터베이스화할 수 있으며 웹서비스를 하기 위한 스타일링을 자유롭게 적용해야 하며 사용자 검색 조건에 따라 자유롭게 문제를 조합할 수 있는 XML을 기반으로 하는 문제은행 시스템을 살펴보았다.

이 시스템이 더욱 효과적으로 사용되기 위해서는 시나리오 설계의 완성도가 더 필요하다. 단순히 난이도에 따라 문제를 제출하고 채점하는 방법을 떠나 사용자의 점수가 서로 유기적으로 영향을 미치는 점수 관리하는 알고리즘의 개발이 필요하다.

[참고문헌]

- [1] 김연정 “능동형 퀴즈 생성을 위한 서버의 설계 및 구현”, 한국 게임 학회 학술대회 논문집 pp.174-176, 2002년 7월
- [2] 양광모 “XML 기반 문제 은행 시스템 구축 사례”, 마이크로 소프트웨어 pp.280-284, 2001년 3월호
- [3] 김채미, 김삼석, 최학열 “XML DTD와 스키마로 객체지향 모델링하기”, 마이크로 소프트웨어 pp.344-284, 2001년 1월호
- [4] 하명희, 박남숙, “XML을 이용한 문제은행 시스템 설계 및 구현”, 한국OA학회지 (제6권 제2호), 2001년, 06월 발행 Vol.8, No.2 pp.33 - 42
- [5] 김경수, 최문영, 주경수 “객체지향 데이터베이스 기반의 XML 응용을 위한 통합 설계 방법론” 한국 OA학회, OA학회논문지 (제7권 제1호), 2002년, 03월 발행 Vol.7, No.1 pp. 54 - 61