

저작권 관리를 위한 MPEG-21 프레임워크 기반의

REL 문서 처리 시스템 설계 및 구현

고광산* · 최일선* · 정재길** · 정희경*

*배재대학교 컴퓨터공학과 · **배재대학교 전자공학과

Design and implementation of the REL documents processing system based on MPEG-21 Framework for rights managements

Gwang-San Ko* · Il-Sun Choi* · Jae-Gil Jung** · Yeon-Sik Park*** · Hoe-Kyung Jung*

*Dept. of Computer Engineering, Paichai Univ. · **Dept. of Electronic Engineering, Paichai Univ. ·

E-mail : *e-ksanko@hanmail.net, {*choijk · **jgjeong · *hkjung}@mail.pcu.ac.kr

요 약

최근 디지털 콘텐츠에 대한 무분별한 저작권 기술개발로 인해 콘텐츠 제공자와 소비자들 사이에 호환성, 콘텐츠 보호, 콘텐츠 유통 등의 문제가 야기되고 있다. 따라서 MPEG(Moving Picture Expert Group)-21 프레임워크는 저작권 관리를 위해 XML(eXtensible Markup Language)에 기반 한 저작권 표현 언어인 REL(Right Expression Language)을 제안하였다.

이에 본 논문에서는 MPEG-21 프레임워크의 이해 없이도 쉽게 콘텐츠에 대해 효율적으로 저작권 생성 및 편집이 가능한 REL 문서편집 시스템과 REL 문서의 소비 방법을 제시하는 클라이언트 시스템을 설계 및 구현 하였다.

ABSTRACT

Problem of compatibility, contents protection, contents circulation etc. was caused between contents offerer and consumers by indiscreet rights technical development about digital contents. Therefore, MPEG-21 framework proposed the REL(Right Expression Language) based on XML(eXtensible Markup Language) by rights expression language for rights management.

We designed and implemented of the REL document editing system that rights creation and edit are available efficiently about contents easily without comprehension of MPEG-21 framework and client system that present consumption method of REL document.

키워드

MPEG-21, REL, XML, 저작권, 디지털 콘텐츠

1. 서 론

최근 인터넷의 고속화와 관련 기술의 급속한 발달로 인하여 콘텐츠는 생성 단계부터 디지털로 만들어지고 있다. 그러나 콘텐츠들은 미비한 저작권 관리와 및 불완전한 저작권 보호로 인해 그 가치를 인정받지 못하고 있으며, 그에 따른 업계의 손실 역시 인터넷 사용의 증가 추세에 비례하여 빠르게 늘어나고 있는 실정이다. 또한 소비 구조, 요소 기술 등이 표준화 되지 않았기에 콘텐츠 제공자들은 그에 따른 비즈니스 모델을 각기 다르게 구현하게 되었고, 이에 따라 멀티미디어 콘텐츠들의 전송 방식, 소비 구조, 요소 기술 등이

상이해서 콘텐츠 제공자와 소비자 모두에게 피해를 주고 있는 실정이다. 이에 MPEG 위원회는 MPEG-21 프레임워크를 제안하여 콘텐츠의 생성, 유통, 이용, 권리 표현 및 기존의 서비스들을 포함 할 수 있는 환경을 조성하기 위한 연구를 진행하고 있다[1,2,3]. 여기서는 저작권 관리를 위해 XML의 기술 표현에 의한 XrML(eXtensible rights Markup Language)의 확장된 저작권 표현 언어인 REL(Rights Expression Language)이 제안되었다. REL의 사용으로 합리적이고 상호 호환적인 콘텐츠 보호, 유통 및 관리를 위한 표준화된 저작권 정보 표현이 가능하게 되었다. 그러나 REL 문서를 저작하기 위해서는 REL의 구조 및 복잡한

REL 스키마(Schema) 문서에 대한 전문적인 지식이 요구된다.

이에 본 논문에서는 저작권 생성 및 편집이 가능한 REL 문서 편집 시스템과 저작 시스템을 기반으로 만들어진 REL 문서를 소비하는 연구도 병행하여 저작권 문서의 소비 방법을 제시하는 REL 처리를 위한 클라이언트 시스템을 설계 및 구현하였다.

II. REL

REL은 XML 스키마 기술을 사용하여 XML 네임스페이스와 호환성을 지니기 때문에, 디지털 콘텐츠와 웹 서비스 제공자들에 대한 비즈니스 모델의 범위를 한층 넓히고, 디지털 권리 언어의 기능으로서 여러 가지 웹 서비스에 접근하기 위해 필요한 권리와 조건을 개발자가 확장할 수 있게 하는 가능성을 확장 시켰다고 볼 수 있다. 즉, 수학적 표현에 따른 명확성, 비즈니스 모델, 라이프 사이클 관리, 사용내역 추적, 패턴 매칭 등의 표현, 확장 아키텍처, 보안 성능이 향상되었다[4].

REL 구조는 세 부분으로 나뉘게 되는데, REL 중심(Core)(REL-R)에서는 엘리먼트 License를 비롯한 핵심 요소 및 REL 전체의 개념 정의 등을 포함하며, REL 표준 확장(REL-SX) 부분은 REL 중심에서 반드시 필요하지는 않지만 REL 소비에서 일반적이고도 광범위하게 사용할 수 있는 개념들을 정의하고 있다. REL 멀티미디어 확장(REL-MX) 부분에서는 멀티미디어 데이터들의 사용, 삭제 및 수정 등 미디어 데이터들의 행동 등을 확장한 정보가 정의 되어 있다. 각각의 구조를 통해서 각각 하나의 스키마가 존재하며 필요한 경우 REL 중심에 확장하여 사용할 수 있다.

III. REL 문서 생성, 편집 및 클라이언트 시스템 설계

본 시스템은 저작자가 쉽게 REL 문서를 저작할 수 있는 폼 기반의 문서 생성 모듈, 생성된 문서를 표현을 해 주는 문서 표현 모듈, 동시에 저작된 문서를 트리 기반으로 편집 가능한 문서 편집 모듈, 문서 검증을 실시하는 문서 검증 모듈 등 네 부분으로 구성된다. 그림 1은 REL 문서 편집 시스템의 전체 구성도이다. 문서 생성 모듈은 REL 문서 구조를 정의한 REL-R, REL-SX, REL-MX 세 가지 스키마 문서를 구조화시킨 REL 스키마 구조정보 파일과 실질적 REL 문서를 생성하는 문서 생성 모듈로 이루어지며 폼 인터페이스를 통해 사용자로부터 생성 요구를 받아들여 REL 문서를 DOM(Document Object Model) 객체 트리 구조로 구성하여 저작권 정보를 DOM 객체 트리에 삽입하게 되고 모든 생성요구를 마치고 되면 REL 문서를 생성한다[5].

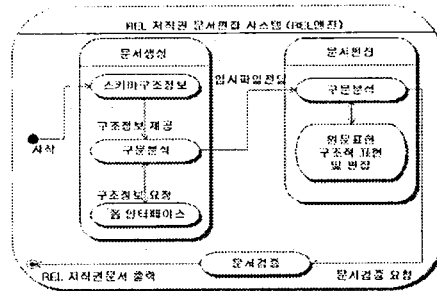


그림 1. REL 문서편집 시스템 전체 구성도

문서 생성 모듈로부터 생성된 REL 문서는 문서 편집 모듈로 전달된다. 전달된 문서는 DOM 객체로 재구성하고 이를 먼저 문서 표현 모듈로 전달해 사용자가 문서 구조를 확인 할 수 있게 된다. 또한 계층적 구조 편집기를 통해 계층적 표현 모듈에 편집 인터페이스를 제공하여 사용자가 요구하는 문서구조로 문서 편집을 실행하게 된다. 모든 편집을 마친 DOM 구조는 문서 검증을 통해 REL 문서로 저장되도록 설계되었다.

REL 문서편집 시스템에서는 노드들의 기본 정보 및 프로퍼티 정보를 통합하여 관리하기 위해 그림 2와 같이 NODEDATA 타입의 데이터 구조를 정의하였다.

```
TFC_EXT_API typedef struct tagNODEDATA
{
    bool        bError;
    CString    szNodeName; // 노드 이름
    CString    szText; // 노드 텍스트
    DOMNodeType pNodeType; // 노드 종류
    IXMLDOMNode* pNode; // 기존 편집 웹)
    UINT       nInsertType; // 삽입 위치 참조 웹)
    BOOL       bExistAtt; // 노드 속성 존재 여부
} NODEDATA;
```

그림 2. 노드 정보 관리를 위한 데이터 구조

이와 같은 데이터 구조를 갖는 NODEDATA로 클래스를 인터페이스화 시켜 타 시스템에서도 DLL 임포트만으로 필요한 함수를 직접 사용 가능하도록 설계하였으며, 이를 이용해 트리 형태의 연결 구조나 선형 연결 구조, 또는 두 형태의 혼합된 연결 구조 등의 논리 구조를 이용하는 다른 시스템으로의 확장이 용이 하도록 설계하였다. 이와 같은 구조체 정보로 생성된 REL 문서는 편집 또는 분석 시스템에서 참조하며 각각 필요한 정보의 검색과 수정을 보다 용이하다.

3.1 REL 문서 생성 모듈

REL 문서 생성을 위해 REL에 대한 스키마를 분석하여 구조화하고 이 구조를 기반으로 폼 인터페이스를 통해 사용자가 입력한 부분에 대한 REL 문서의 저작이 이루어진다. REL 문서를 생성하기 위해 REL에서 사용하는 스키마인 REL-R,

REL-SX, REL-MX 에 대한 정보의 구조 및 정의 부를 분석하고, 중요 요소와 속성을 추출하고 효율적인 REL 문서 생성 할 수 있도록 설계하였다. REL 문서는 스키마 구조로 되어 있어 다양한 표현이 가능하며 다중의 스키마 문서를 포함 할 수 있어 사용자가 원하는 다른 스키마들을 포함하여 사용할 수 있다.

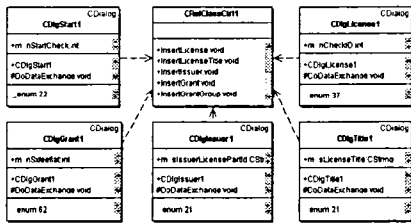


그림 3. 문서 생성 모듈 클래스 구조

문서 생성 모듈은 그림 3과 같이 RelClassCtrl 이라는 클래스를 중심에 두고 다중 다이얼로그를 통해 사용자 요구를 받아들인다. 각각의 다이얼로그들은 해당하는 요소를 REL 구조부에 요청하게 되고 REL 구조부에서는 요구된 정보를 생성하여 RelClassCtrl에 전달하여 DOM에 저장하게 된다.

```

void CRelClassCtrl::Make()
{
    Crel_mxDoc Doc;
    CDlgStart dlgStart;
    if(dlgStart.DoModal() != IDOK) return;
    r.CLicenseGroup root; //기 본 트
    r.CLicense license; //부 록 개 기
    if(dlgStart.m_nStartCheck==1) //0.License LicenseGroup 선택 다 이
    {
        Doc.SetRootElementName(_T
        ("um.mpeg.mpeg21:2003.01REL(LBIS)"),_T("licenseGroup"));
        int i=0; j=0;
        j=dlgStart.m_nLicenseCount;
        for(i=0; i<j; i++)
        {
            r.CLicense license;
            InsertLicense(license, root, true); //license 생
        }
    }
    else if(dlgStart.m_nStartCheck==2)
    {
        Doc.SetRootElementName(_T
        ("um.mpeg.mpeg21:2003.01REL(LBIS)"),_T("rLicense"));
        InsertLicense(license, root, false);
    }
}
    
```

그림 4. REL 문서 생성 알고리즘

그림 4는 REL 문서 생성 알고리즘으로 CRelClassCtrl 클래스의 Make() 모듈은 각각의 다이얼로그들을 통한 관리하는 모듈로 사용자 요구에 맞추어 다이얼로그를 생성하고 생성된 다이얼로그에 입력된 정보를 수집하여 REL 문서를 생성한다.

3.2 REL 문서 편집 모듈

문서 생성 모듈로부터 생성된 REL 문서를 사용자가 직접 보면서 편집할 수 있게 해 주는 부분으로 구조적 편집 모듈과 원문 표현 모듈로 이루어진다. 원문 편집 모듈은 사용자가 XML 문서를 작성 편집할 수 있는 기본적인 환경을 제공한다. 문서 편집 모듈에서 전체 문서의 계층 구조를 볼 수 없는 단점을 극복하기 위해 계층구조를 볼 수 있는 구조창을 두었다. 문서 중에서 속성이 있

는 요소는 저작자의 혼란을 피하기 위해 트리 요소의 아이콘 색상으로 구별 하도록 하고 요소가 가지는 문자열과 태그는 화면에 출력하고 트리구조 내용과 속성 내용은 컨텍스트 메뉴와 대화상자를 이용해서 수정할 수 있도록 하였다. 구조적 편집 모듈을 통해서 전달되는 정보는 DOM 객체에 전달되어 해당하는 일을 처리한다.

```

CMainFrame::OnCreate(LPCREATESTRUCT lpCreateStruct)
{
    //OUTPUT 메시지 생성
    if(!m_wndOutput.Create(this, ID_VIEW_OUTPUT,
    _T("Output"), CSize(150, 100), CBS_BOTTOM, TRUE))
    {
        TRACE0("Failed m_wndOutput\n");
        return FALSE; // fail to create
    }
    // tree view 생성
    LPCTSTR lpszWindowName = _T("Tree View");
    if(!m_pTreeView.Create(this, ID_TREE_VIEW,
    lpszWindowName, CSize(200, 75), CBS_LEFT, TRUE))
    {
        TRACE0("Failed m_wndTreeView\n");
        return FALSE; // fail to create
    }
}
    
```

그림 5. REL 문서 표현 및 편집부 생성 알고리즘

그림 5는 문서 표현 및 편집 생성 알고리즘으로 어플리케이션 내부에 생성되는 구조적 표현부인 TreeView 영역과 메시지 출력부인 OutPut 메시지 출력부의 생성을 보여준다. 원문 출력부는 기본적인 뷰 영역의 EditView를 상속받아 생성한다.

3.3 클라이언트 시스템

REL 클라이언트는 MPEG-21 프레임워크 내에 존재하는 DIP(Digital Item Player) 역할을 한다. DIP는 일반적인 디지털 아이템들의 소비 물로 볼 수 있으나 MPEG-21 내에서 사용되는 모든 미디어에 대해 처리 할 수 있는 특징을 가진다. REL 클라이언트 모듈은 MPEG-21에서 제시하는 DIP 모델 중 REL 엔진을 포함하고 동영상에 해당하는 아이템을 처리할 수 있는 최소화된 DIP이다. 그림 6은 REL 클라이언트 모듈 구성도를 보인다.

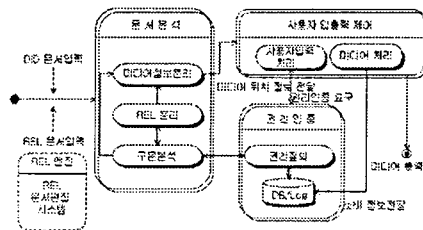


그림 6. REL 클라이언트 모듈 구성도

REL 파싱은 문서를 읽어 들여 문서 구조상의 문제를 확인 한 뒤 내부 문서를 검색한다. Grant

에 해당하는 모든 엘리먼트를 검색 해 놓고 Grant 내부의 정보를 추출하여 데이터베이스에 저장하거나 인증 정보를 비교한다.

IV. 구현

본 논문의 REL 문서편집 시스템과 클라이언트는 IBM PC 호환 컴퓨터에서, Windows 2000 운영체제 환경에서 Visual C++ 6.0(Microsoft Developer Network Library .Net 2003)을 사용하여 구현하였고, REL의 데이터 처리 및 관리를 위해 사용된 파서는 MS MSXML 4.0을 사용하였다.

폼 기반의 문서 생성 모듈은 DLL(Dynamic Linking Language)로 구현하여 타 시스템에서도 같은 REL 문서의 생성이 가능하게 하였다. REL 문서편집 시스템을 실행하면 Wizard 형식의 문서 생성 모듈이 동작하게 되어 폼 기반의 생성 도구가 호출되게 된다. 저작자는 이벤트로 진행되는 폼 인터페이스에 권리 정보 등의 내용을 입력하게 되면 문서가 저작 되게 된다. 특히 LicenseGroup 또는 License를 중심으로 title, inventory, grant, issuer 등의 핵심 엘리먼트를 중심으로 저작되어 권한 생성자가 빠르고 쉽게 REL 문서를 생성 할 수 있도록 하였다.

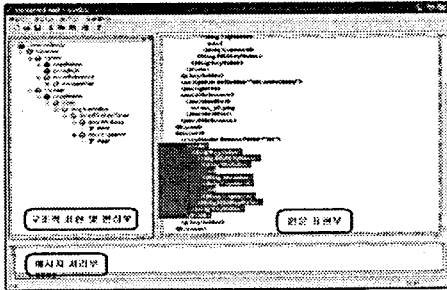


그림 7. 문서 표현 및 편집 모듈 사용자 인터페이스

그림 7은 REL 문서 표현 및 편집 모듈에 해당하는 사용자 인터페이스로 문서표현 및 편집 모듈은 구조적 표현 및 편집부, 원문 표현부, 메시지 처리부로 나뉜다.

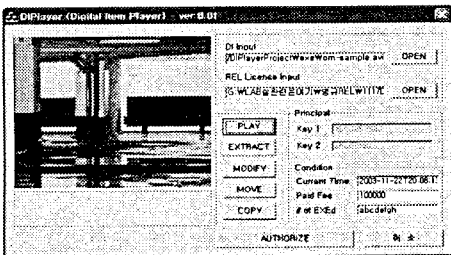


그림 8. DIP 사용자 인터페이스

그림 8은 클라이언트 시스템 인터페이스로 사용하고자 하는 REL 문서와 사용하고자 하는 콘텐츠를 선택하고 AUTHORIZE 버튼을 눌러 미디어에 해당하는 인증을 확인할 수 있다.

V. 결론

본 논문은 MPEG-21 프레임 워크에서 사용되는 REL 문서를 DOM 객체로 생성 및 구문 변환 처리하여 새로운 REL 문서를 생성하고 문서를 편집할 수 있는 REL 문서 저작 시스템과 생성된 REL 문서를 분석하여 권리 정보에 따라 디지털 아이템을 소비 할 수 있는 클라이언트 시스템을 설계 및 구현하였다.

본 시스템의 장점으로는 국제 표준에서 제안하는 REL 처리에 대한 구성을 기반으로 구현함으로써 변화에 능동적으로 대처 가능하고, MPEG-21 프레임워크와 REL의 구조 및 문법에 대한 지식이 부족한 사용자에게 무결성이 유지되는 REL 문서로써 저작권 표현 문서를 쉽게 생성할 수 있고, 폼 기반의 REL 문서 편집 시스템을 통해 저작권 정보들을 WYSIWYG 편집이 가능하게 하고, 구조적인 편집까지 지원하기 때문에 보다 정확하고 유효한 문서를 유지할 수 있다. REL 문서를 통해 저작권 정보를 분석하고 변경되는 사용자의 권한 및 기타 정보를 지속적으로 관리함으로써 DIP 운용에 대한 방향을 제시하고 있다는 장점이 있다.

본 연구 결과는 멀티미디어 콘텐츠의 저작권 보호를 위한 REL 문서의 생성, 편집, 분석에 관한 효과적인 운용 방안에 대해 제시하고 있으며 MPEG 21 시스템 통합이 완료되면 방송통신 및 디지털 방송 등에 적용되어 신 혁신적인 디지털 미디어 서비스가 제공될 것으로 사료된다.

향후 연구되어야 할 과제로서, REL의 온톨로지 정보를 가지는 RDD 시스템과의 연동서비스를 통해 REL에서 알아낼 수 없는 REL 엘리먼트 정보에 대한 처리를 가능토록 해야 하며 REL 문서에 따른 디지털 아이템의 소비에 대한 모델링 작업에 대한 설계가 보완 되어야 할 것이다.

참고문헌

- [1] 김해광, "MPEG 21 멀티미디어 프레임워크", TTA저널, 제82호, 2002.
- [2] "MPEG 21" <http://www.chiariglione.org/mpeg/index.htm>
- [3] MPEG Requirements Group, "MPEG 21 Overview," Editors: J. Bormans and K. Hill, ISO/IEC JTC1/SC29/WG11 N4511
- [4] "Digital content forum" <http://www.dcforum.or.kr>
- [5] 정희경, "알기쉽게 해설한 XML", 이한출판사, 2005