

# 메타데이터의 분할 및 갱신을 위한 TeM 코덱 설계

양승준<sup>1</sup>, 장현성<sup>2</sup>, 김영태<sup>3</sup>, 강경옥<sup>4</sup>, Nguyen Ngoc Thanh<sup>5</sup>  
한국전자통신연구원 방송미디어연구부<sup>1,2,3,4</sup>  
대전광역시 유성구 가정동 161  
한국정보통신대학원대학교 공학부<sup>5</sup>

## TeM Codec Design for Metadata Fragmentation and Update

Seung-Jun Yang<sup>1</sup>, Hyun Sung Chang<sup>2</sup>, Young-tae Kim<sup>3</sup>, Kyeongok Kang<sup>4</sup>, Nguyen Ngoc Thanh<sup>5</sup>  
Electronics and Telecommunications Research Institute (ETRI)  
161 Gajeong-Dong, Yuseong-Gu, Daejeon  
Information and Communications University (ICU)<sup>5</sup>  
E-mail<sup>1</sup>: [sjyang@etri.re.kr](mailto:sjyang@etri.re.kr)

### 요약

MPEG-7은 멀티미디어 데이터를 XML로 기술한 새로운 표준이다. MPEG-7은 구조 및 의미 정보, 미디어 형식 정보 등과 같은 유용한 기술 및 기술구조들을 제공하며, MPEG-7 시스템에서는 메타데이터의 텍스트 인코딩 포맷인 TeM은 기술의 동적 혹은 점진적인 전송을 지원한다. XML 문서의 본래의 기술을 추가적인 XML 헤더가 덧붙여져 여러 개로 나누어질 수 있으며, 이러한 결과로 나온 AU들은 각기 전송을 하여 터미널에서 수신된 AU를 이용한 갱신이 가능하다. 본 논문에서는 이러한 MPEG-7 시스템의 갱신 메커니즘을 이용한 TeM 코덱을 설계 및 구현하였다. 구현된 TeM 코덱의 인코더는 AU 생성에 있어서 시각화된 편집 도구를 제공하며, 디코더는 XSLT 기반의 갱신 기법을 기반으로 하여 구현하였다.

### 1. 서론

향후 디지털 방송 환경에서 데이터 서비스 제공자들은 시청자의 다양한 요구사항을 만족시키기 위하여 여러 종류의 메타데이터를 제작하고 이를 제공하게 될 것이다. 이러한 메타데이터는 생성 과정에서나 생성된 후에 부분적으로 변화될 필요가 있을 수 있다. 서버단에 위치한 메타데이터의 경우에는 별다른 문제없이 수정이 용이한 반면에 이미 전송이 이루어진 메타데이터의 경우에는 단말측에 대한 메타데이터 갱신을 위한 기법이 필요할 것이다. 단지 일부분의 갱신을 위해 전체 메타데이터를 재전송한다는 것은 채널의 활용면에서 매우 비효율적인 일이다. 따라서 이에 효과적인 메타데이터 갱신 기법이 필수적으로 요구된다.

MPEG-7은 멀티미디어 데이터를 XML(eXtensible Markup Language)로 기술한 새로운 표준이다. MPEG-7은

구조 및 의미 정보, 미디어 형식 정보 등과 같은 유용한 기술 톨들을 제공하며, MPEG-7 메타데이터의 텍스트 인코딩 포맷인 TeM(TeM: Textual format for multimedia description streams)은 기술(description)의 동적 혹은 점진적인 전송을 지원한다. XML 문서의 본래의 기술은 추가적인 XML 헤더가 덧붙여져 여러 개로 나누어질 수 있으며, 이러한 결과로 나온 AU(Access Unit)들은 각기 전송이 가능하도록 규정되어 있다[1].

본 논문에서는 이러한 MPEG-7 시스템의 갱신 메커니즘을 이용하여 TeM 메타데이터를 위한 인코더와 디코더를 구현하였다. 구현된 인코더는 AU 생성을 위한 시각적 편집 도구를 지원하며, 구현된 디코더는 텍스트 레벨에서 멀티미디어 콘텐츠 기술의 수신된 부분들을 구성하기 위한 방법으로 XSLT (eXtensible Stylesheet Language Transformation) 기반의 갱신 메커니즘을 이용한다.

본 논문은 다음과 같이 구성된다. 다음의 제 2장에서는 MPEG-7 시스템과 TeM에 대한 간략한 소개를 기술하고, 제 3장에서는 구현된 인코더와 디코더에 대한 구조와 구현 방법에 관해 기술한다. 제 4장에서는 제안된 방법으로 구현된 코덱에 대한 실험과 그 결과를 기술하고 마지막으로 제 5장에서는 제안된 시스템에 대한 결론을 제시한다.

### 2. MPEG-7 시스템과 TeM 코덱

MPEG-7 시스템은 멀티미디어 콘텐츠에 대한 기술을 전송하기 위한 도구를 제공한다. 전송을 위한 기술 스트림(description streams)은 텍스트 인코딩 포맷인 TeM 또는 이진 부호화 포맷인 BiM(Binary format for multimedia description streams)으로 부호화 되어 전송되어질 수 있다. MPEG-7 시스템에서는 콘텐츠 기술 메타데이터의 전송을 위해 전체 기술을 다수 개의 프래그먼트

(fragment)로 나누고, 각 프래그먼트에 규정된 XML 헤더를 덧붙여 그 결과로 AU 를 만들 수 있으며, 이렇게 생성된 AU 는 각기 전송이 가능하다. 즉, 콘텐츠 기술 메타데이터 갱신을 위해 전체 콘텐츠 기술 메타데이터를 부호화하여 재전송하는 대신에 갱신을 위한 특정 부분의 AU 만을 전송함으로써 동적으로 갱신이 가능하도록 하는 메커니즘을 제공하는 것이다[1][2][4].

구현된 TeM 소프트웨어 코덱은 인코더와 디코더로 구성된다. 이들 사이의 입출력 인터페이스를 위해서는 위와 같은 공통의 포맷이 필요하다. 메타데이터가 위치한 인코더 측면에서는 콘텐츠 기술 메타데이터를 패키징하는 작업을 수행하게 된다. 먼저 트리 구조 형태로 표현되는 기술을 다수개의 FUU(Fragment Update Unit)로 분할하게 된다. 이러한 FUU 들은 AU 을 구성하는 기본 단위가 되며, 이러한 연속적인 AU 들이 기술 스트림으로 전송되어진다. 각 AU 는 이러한 FUU 를 처리하기 위하여 명령, 위치, 콘텐츠에 대한 정보를 포함하게 된다.

터미널(디코더) 측면에서는 전송 계층으로부터 기술 스트림을 수신하게 된다. 기술 스트림을 해독하기 위해, 각 AU 를 분해하면 콘텐츠 기술을 조작하기 위한 명령, 위치, 콘텐츠에 대한 정보를 얻게 된다. 이러한 콘텐츠 기술에 대한 정보 구성이 아래의 그림 1 에 도시 되어져 있다.

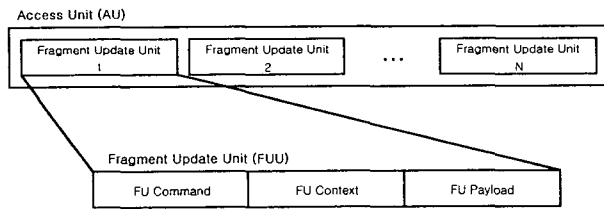


그림 1. MPEG-7 시스템의 AU 구조

그림 1 에서 각 AU 는 다수개의 FUU 로 구성되어진다. 각 FUU 는 FU Command(Fragment Update Command) , FU Context(Fragment Update Context), FU Payload(Fragment Update Payload)로 구성되어진다. FU Command 는 FUU 가 수행되어질 명령을 의미하며, add, delete, replace, reset 가 사용될 수 있다. FU Context 는 FU Command 가 실행되어질 위치를 지시하는 위치정보를 포함한다. FU Payload 는 FU Context 가 지시하는 위치에 FU Command 가 실행되어 나타나게 될 메타데이터가 포함된다[1][4].

### 3. TeM 코덱의 설계와 구현

#### 2.1 TeM 인코더

TeM 인코더는 텍스트 포맷의 메타데이터를 AU 로 분할하여 부호화하고 갱신을 위한 AU 를 전송하기 위하여 시각적인 AU 편집 도구를 제공하도록 구현되었다. 구현된 인코더는 기 생성된 메타데이터를 전송하기 위해 입력된 메타데이터를 분할하고 부호화한 후, 전송

이 가능하도록 구성되어 있으며 메타데이터를 기술 스트림으로 부호함에 있어 단말측에서 사용될 초기 복호화 정보(DI: DecoderInit) 와 reset 명령을 별도로 생성하도록 구성되어 있다.

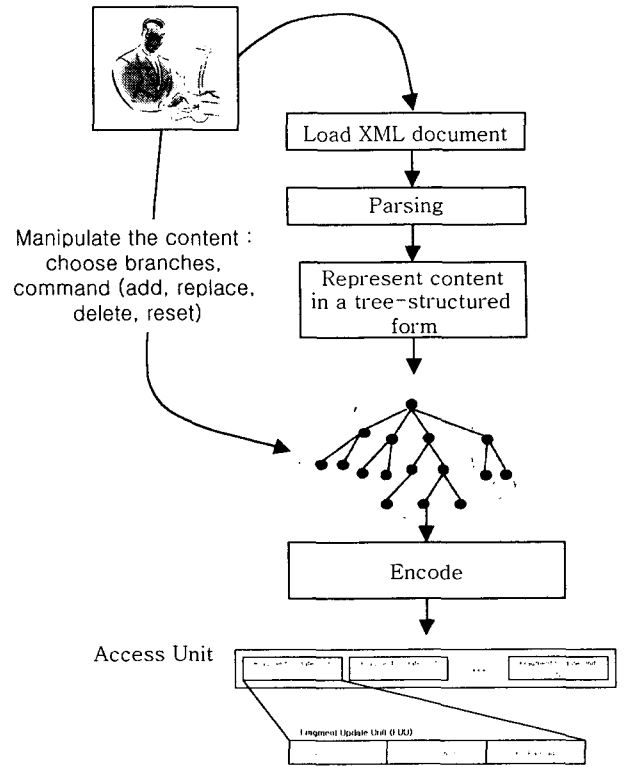


그림 2. AU 생성을 위한 인코더 구조도

그림 2 에서 인코더는 전송 또는 갱신이 필요한 AU 를 생성하기 위해 기 생성되어 있는 XML 메타데이터와 그와 관계된 스키마를 입력 받게 된다. 입력된 메타데이터는 주어진 스키마에 의해 파싱(parsing) 과정을 거치게 되고, 그 다음에 트리 구조 형태로 별도의 창에 디스플레이 되도록 하였다. 사용자는 주어진 디스플레이 창에서 전체 또는 특정 영역의 메타데이터를 조작하기 위해 서브 메타데이터에 대한 트리 가지를 선택할 수 있으며 이에 대한 명령어(add, delete, replace, reset)를 지정할 수 있다. 선택된 각각의 서브 트리는 각기 다른 FUU 로 순차적으로 저장되게 된다. 이와 같은 과정을 반복하여 원하는 특정 부분에 대한 선택이 끝나면 저장된 FUU 를 하나의 AU 로 구성하게 된다. 구성된 AU 는 사용자가 원하는 임의의 영역에 저장되거나 전송망을 통하여 단말측으로 전송될 수 있다.

AU 생성을 위한 시각적 편집 창이 아래의 그림 3 에 도시 되어 있다. 그림 3 에서 인코더는 입력된 메타데이터를 파싱하고 트리 구조 형태로 디스플레이하여 사용자에게 대화형 편집 도구를 제공한다. 콘텐츠 기술 메타데이터의 특정 엘리먼트를 선택한 경우, 선택된 위치로부터 하위 구조를 포함하게 되며, 이것이 하나의 FUU 로 규정된다.



```

<xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
  <xsl:include href="init.xsl"/>
  <!--Note that the variable named initial_description_tree
is defined in "init.xsl" which is generated from
DecoderInitialisation information. See Fig. 5(a). -->
  <xsl:template match="/">
    <xsl:copy-of select="$initial_description_tree"/>
  </xsl:template>
</xsl:stylesheet>

```

(e) Reset \* \* \* XSL \* \* \*

\* \* \* 5. \* \* \* \* \* \* \* \* \* XSL \* \* \*

이러한 XSLT 기반의 메타데이터 갱신 구조는 기존에 존재하는 XSLT 모듈을 이용하여 간단히 구현할 수 있으며, 추가적인 별도 모듈이 필요하지 않는다[2].

#### 4. 실험 및 결과

TeM 코덱은 MicroSoft사의 Windows™ O/S 플랫폼을 기반으로 구현되었다. 인코더는 XML 문서를 파싱과 조작을 위해 Xerces C++ 1.5.1. 라이브러리를 사용하였으며 사용자 환경을 위한 GUI를 구성하기 위해서 윈도우 환경을 지원하는 도구들을 사용하였다. 이러한 GUI는 그림 6에 나타내었다.

그림 6(a)에서 인코더는 입력된 문서를 파싱하여 트리 구조 형태로 디스플레이 한다. 사용자가 이러한 디스플레이창에서 문서를 FUU로 분할하고 갱신을 위한 조작을 하는 방법이 그림 6(b)에 나타나 있다. 그림 6(c), (d)는 인코딩된 AU 결과와 전송을 위한 AU의 선택 과정을 보여준다.

그림 7에서는 기 전송된 기술이 새로운 갱신 AU를 수신하여 갱신된 결과를 비교 화면으로 제시하였다.

#### 5. 결론

본 논문에서는 텍스트 레벨에서의 메타데이터 전송을 위한 인코더/디코더 시스템을 구현하였다. 구현된 시스템은 메타데이터 부호화 과정 및 갱신을 위한 과정에서 저작에 대한 편의성과 구현의 용이성을 제공하는 것을 중심으로 구현되었다.

인코더에서 제공하는 시각적인 AU 편집도구는 AU 생성을 위한 사용자 편의성과 저작 효율을 증대하는 효과가 있을 것으로 기대되며 메타데이터 저작을 위한 편집 툴과 같은 다양한 응용에서 독립적인 모듈로 포함되어 사용되어질 것으로 기대된다.

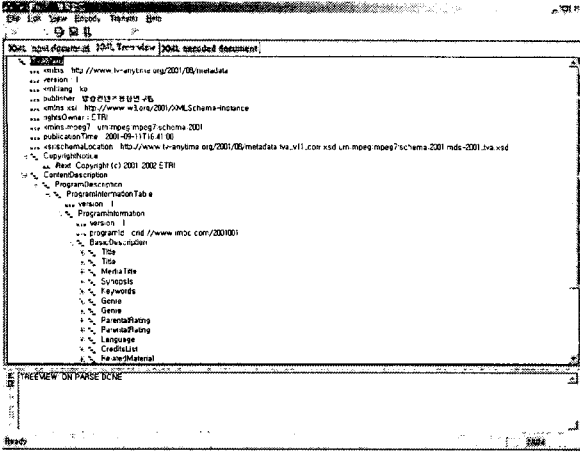
디코더에서 사용된 XSLT 기반의 갱신 메커니즘 기법은 단말측에 XSLT 모듈이 존재할 경우, 기존에 존재하는 모듈을 이용하여 용이하게 구현할 수 있으며, 별도의 추가적인 모듈이 필요하지 않는다는 장점이 있다.

#### 참고 문헌

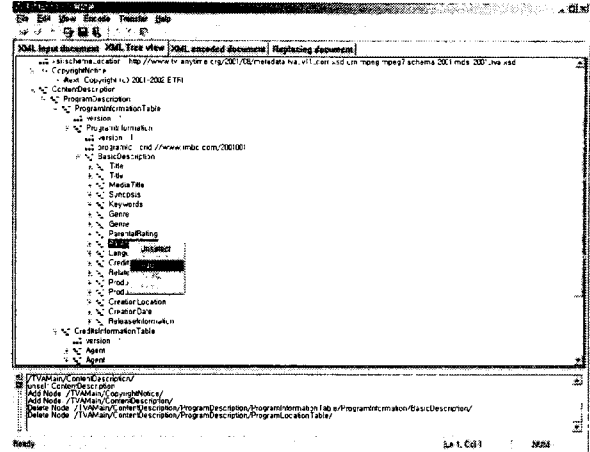
- [1] "Text of ISO/IEC 15938-1/FDIS-Part 1 : Systems", ISO/IEC JTC1/SC29/WG11, N4285, 2001.
- [2] 장현성, 양승준, 강경욱, "XSLT를 이용한 TeM 메타데이터 갱신구조의 설계 및 구현", 제 14 회 영상처리 및 이해에 관한 워크샵, Jan. 2002, pp. 455-460.
- [3] XSL Transformations(XSLT) version 1.0, W3C Recommendation, Nov. 1999.(available at the site of <http://www.w3.org/TR/xslt>)
- [4] Cedric Thienot, Claude Seyrat, "A streamable XML binary encoding for TV Anytime metadata", MD WG, AN282, Jun 2001.

#### 감사의 글

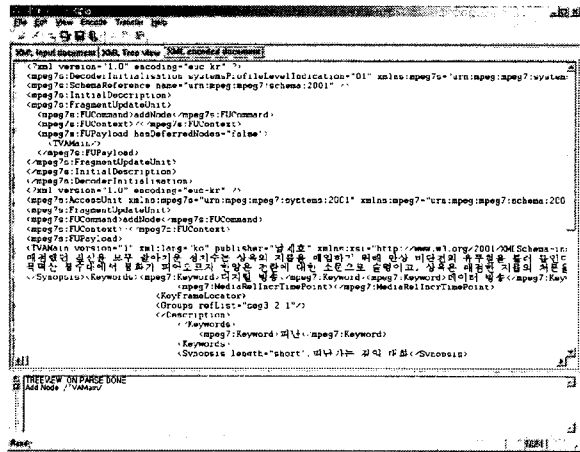
본 논문은 정보통신부의 "MPEG-7 기반 메타데이터 방송 기술 개발 사업" 과제의 일환으로 수행된 연구 결과이며, 도움을 주신 방송콘텐츠응용팀원들께 감사드립니다.



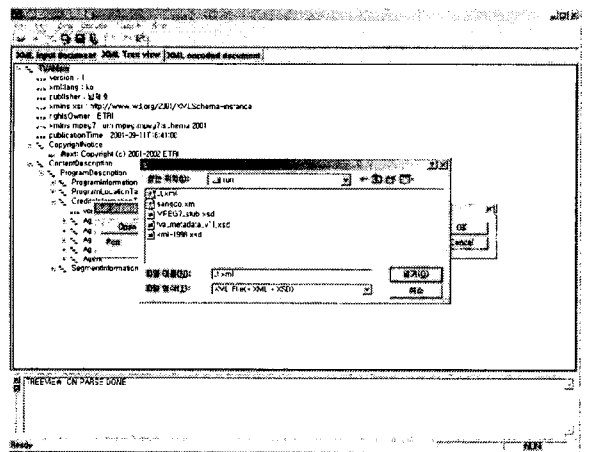
(a)



(b) AU

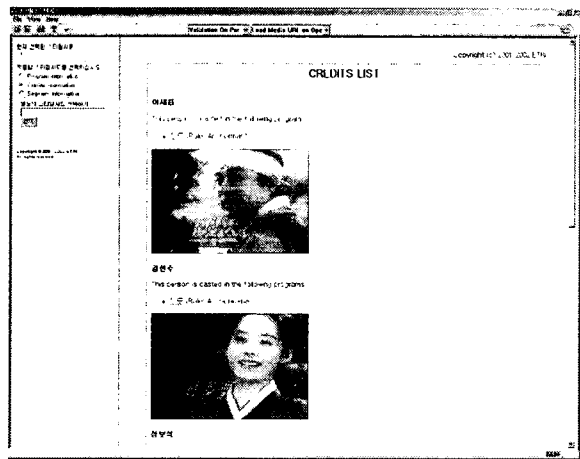


(c) AU

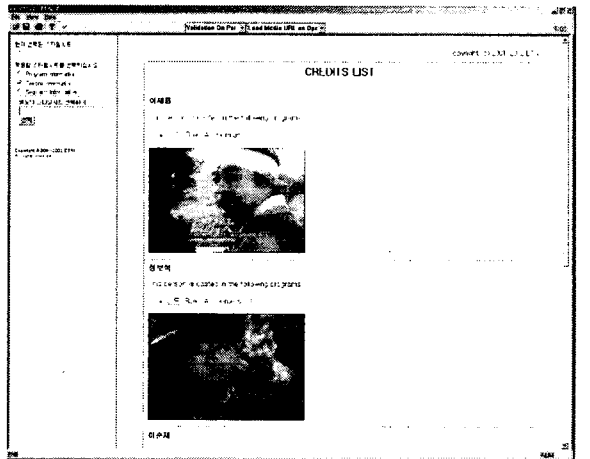


(d) AU

6. GUI



(a)



(b)

7. GUI