

수학교과 수준별 교수-학습 자료에서 MathML과 XSLT의 활용

홍 은 표 (창원상남중학교)

이 수 현 (창원대학교)

1. 서론

수학교육에 있어서 정보통신기술(ICT)의 활용은 학생들의 흥미를 유발시키고, 문제 해결에 따른 학생들의 반응을 분석·판단하여 개인에게 적절한 피드백을 제공할 수 있게 한다. 문자 정보, 그림, 그래프, 애니메이션, 시뮬레이션 등과 같은 여러 가지 형태의 정보를 통합적으로 제공함으로써 수학적 개념과 원리를 보다 시각적으로 명확하게 전달할 수 있고, 지식의 통합과 분석이 이루어질 수 있도록 다양하고 방대한 수학 자료를 제공함으로써 종합적인 사고력을 신장시킬 수 있다(김강문, 2000).

학교에서의 수업에 적용할 수 있는 ICT-활용 교육 유형은 정보탐색하기, 정보분석하기, 정보안내하기, 웹토론하기, 협력연구하기, 전문가와 교류하기, 웹펜팔하기, 정보만들기 등 여러 가지 유형(한국교육학술정보원, 2001)이 있으나 수학교과에서는 컴퓨터나 웹 상에서 편리하게 수식을 사용할 수 없기 때문에 다양한 형태의 ICT-활용 수업 유형을 적용할 수가 없다. 실제로 웹토론하기, 전문가와 교류하기와 같은 수업 유형은 학생들에게 있어서 자신이 가진 어려운 문제를 해결하는 데 매우 유용하게 활용될 수 있는 수업 유형이지만 수식 표현의 제약 때문에 수학교과에서는 잘 활용되지 못하고 있다. 또한 웹을 이용한 학습자료의 경우에도 아직까지는 수식을 이미지 파일로 표현한 경우가 많아서 수식의 재사용 및 계산이 불가능하고, 학생들이 문제에 대한 수식 형태의 답을 제시했을 때 그 답이 정답인지 오답인지를 판단하여 적절한 피드백을 제공하는 학습 자료를 만들기

가 어렵다.

최근 교육인적자원부에서는 시도 교육과학연구원과 공동으로 수준별 교수-학습 자료를 개발하였다(교육인적자원부, 2001a). 이 자료는 단원의 개관, 준비학습, 기본과정, 형성평가, 보충자료, 심화자료, 특별보충자료, 참고자료로 구성되어 있고 HTML 문서로 만들어져서 구성요소들이 각각의 하이퍼링크로 메뉴와 연결되어 있다. 이와 같은 교수-학습 자료에서는 학생들이 각 구성요소들 간의 단계성과 연관성을 무시하고 임의로 특정 구성요소를 선택할 수 있다. 또한 교사들이 학습안내와 학습정리 등을 위하여 전체 수준을 모두 학생들에게 제시하고자 할 경우 각각의 구성요소들을 다시 통합해야 한다는 단점이 있다.

본 논문에서는 교수-학습 자료의 여러 구성요소를 기본 구성요소와 선택적 구성요소로 구분하고, 선택적 구성요소들을 학생들의 수준에 따라 선택적으로 제공할 수 있는 수준별 교수-학습 자료의 개발 방법을 제안하고, 수준별 수업에서 활용하는 방안을 제시하였다. 이를 위하여 본 논문에서는 XML(eXtensible Markup Language)을 기반으로 하여 MathML(Mathematical Markup Language)과 XSLT(XSL Translation)를 이용하였다. MathML(W3C, 2001; R. Miner & J. Schaeffer, 2001)은 수식 표현을 위한 XML의 응용분야로서 인터넷에서 수학적인 내용이나 표기를 표현하고, 처리하고, 공유하는데 필요한 해결책을 제공하는 언어이며, XSLT(W3C, 2002a)는 XML 문서에 대하여 형식이나 스타일을 지정하거나 다른 형태로의 변환 등에 사용되는 표준화된 기술이다.

수학교과 교수-학습 자료를 MathML을 이용하여 작성함으로써 웹에서의 수식 표현이 자연스럽게 되어 효과적인 학습이 가능하도록 하였으며, XSLT를 이용하여 수준별 교수-학습 자료를 구성할 수 있도록 하였다. 즉, 모든 수준의 학습 자료를 MathML을 이용하여 하나의

* 2002년 11월 투고, 2003년 11월 심사 완료.

* ZDM분류 : U50

* MSC2000분류 : 97U50

* 주제어 : 수준별 교수-학습 자료, MathML, XML, ICT-활용 수업.

문서로 작성하고 각 수준에 맞는 문서를 원문서에서 XSLT를 이용하여 추출한다. 본 논문에서 제안한 방법의 장점은 모든 수준의 학습 자료를 하나의 문서로서 작성하므로 교수-학습 자료 개발자가 더욱 쉽게 문서를 작성할 수 있도록 한다는데 있다. 이로 인해서 학습자료의 개발과 유지관리에 효율성을 기할 수 있다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 관련연구로서 MathML에 대한 소개와 MathML을 이용하여 학습자료를 개발한 사례를 살펴보고, 수준별 교수-학습 시스템에 관한 기존의 연구들을 살펴본다. 3장에서는 본 논문에서 제안하는 방법인 수준별 교수-학습 자료의 개발 방법을 기술하고 4장에서 이를 활용할 수 있는 방안 에 관하여 서술한다. 끝으로 5장에서 결론을 맺는다.

2. 관련연구

2.1 MathML

HTML과 이미지 파일을 이용한 기존 웹에서의 수식은 표현과 인코딩(encoding) 면에서 문제점이 있다. 예를 들어, 텍스트의 크기가 커지거나 작아지면 수식을 표현하는 이미지와 텍스트간의 균형이 깨어져서 읽기가 불편해진다. 인코딩의 측면에서는 수식 표현을 검색하고자 할 때 사용할 수 있는 유용한 도구가 없고, 특정 표현을 잘라 붙이는 것과 같은 과정이 매우 복잡하고 어렵게 된다.

MathML은 위와 같은 웹에서의 수학적 표현의 한계를 극복하기 위해 만들어진 XML의 응용이다. MathML은 인터넷에서 수학적인 내용이나 표기를 다루는 방법에 대한 저수준의 형식으로 웹에서 수식을 표현하고, 처리하고, 공유하는데 필요한 해결책을 제공하는 마크업 언어(markup language)이다.

W3C에서는 2001년 MathML2.0 권고안(W3C, 2001)을 발표하였고, MathML을 지원하기 위한 Amaya 브라우저(W3C, 2002b)를 비롯한 다양한 개발도구들을 소개하고 있다.

2.1.1 표현과 내용

수식을 전달하기 위해서 사용할 수 있는 방법은 수식의 모양을 보이는 대로 설명하는 방법과 그것의 의미를 전달하는 두 가지 방법이 있을 수 있다. 예를 들어, $f(x)$ 의 경우에, 첫 번째 방법은 「f 프라임 괄호 열고 x 괄

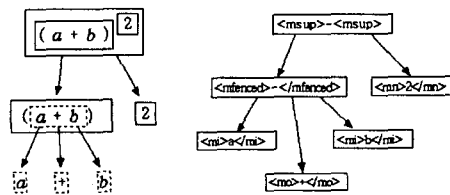
호 닫고」와 같이 전달하는 것이고 두 번째 방법은 「함수 $f(x)$ 의 미분」과 같이 전달하는 방법이다.

MathML에서는 위의 두 가지 방법을 모두 사용하고 있다. 첫 번째 방법은 수식의 표현에 중점을 둔 표현 마크업(presentation markup)이고, 두 번째 방법은 수식이 가지고 있는 의미를 전달하는 내용 마크업(content markup)이다. 내용 마크업은 수식의 의미를 표현하기 때문에 다른 응용프로그램에 그 의미를 전달할 수 있다. 사용자는 상황이나 목적에 맞게 두 가지 방법 중 하나를 선택하여 사용하거나 두 가지 방법을 적절히 섞어서 사용할 수도 있다.

2.1.2 수식 트리와 표현 마크업

일반적으로 수식은 여러 가지의 수와 부호가 모여 하나의 수식을 구성한다. 수식을 위한 수학적 기호와 숫자들은 상당히 많지만 그것들을 배열하는 방법은 위첨자, 아래첨자, 열, 분수, 행렬과 같이 몇 가지로 유형을 정할 수 있다. 이러한 요소들은 중첩되어 나타나기도 하고 문맥에 따라 변수의 개수도 달라질 수 있지만 복잡하고 중첩된 표현도 단순한 몇 개의 스키마 계층 구조로 표현될 수 있다.

예를 들어, $(a+b)^2$ 과 같이 위첨자를 사용하는 수식은 $(a+b)$ 와 2로 나누어질 수 있다. 그리고 $(a+b)$ 는 괄호와 $a+b$ 로 나누어질 수 있다. 또한 $a+b$ 도 a , $+$, b 와 같이 나누어질 수 있다. 이러한 작업은 더 이상 나누어지지 않을 때까지 계속되며, 이 과정을 <그림 1>의 (a)에서 보는 바와 같이 트리의 형태로 나타낼 수 있다. MathML의 표현 마크업에서는 이와 같이 만들어진 수식 트리를 이용하여 인코딩 한다. <그림 1>의 (b)는 수식의 구조로부터 만들어진 MathML 수식 트리를 보여준다.



(a) (b)
<그림 1> MathML 수식 트리

표현 마크업에는 30개의 요소와 약 50개의 속성이 있다. $(a+b)^2$ 을 위한 MathML 표현 마크업은 다음과 같다.

```
<msup>
  <mfenced>
    <mi>a</mi>
    <mo>+</mo>
    <mi>b</mi>
  </mfenced>
  <mn>2</mn>
</msup>
```

MathML에서는 $(a+b)^2$ 의 최상위 구조인 위첨자 형식을 <msup> 태그를 이용하여 표현한다. 그리고 첫 번째 하위표현인 $(a+b)$ 에는 <mfenced> 태그를 사용하고 두 번째 하위표현인 2에는 <mn> 태그를 사용한다. <mfence>도 하위표현을 가지고 있는데 여기에서는 식별자(identifier)로서의 <mi>와 연산자로서의 <mo>가 사용되었다.

2.1.3 내용 마크업

MathML 내용 마크업은 전위표기(prefix notation)를 사용한다. 이는 수식 「5+8」을 「add 5 and 8」과 같이 표현하는 방법을 말한다. 「5+8」처럼 중위표기(infix notation)를 사용할 경우에는 $x-y/2$ 와 같은 식의 해석에 있어 $(x-y)/2$ 와 $x-(y/2)$ 를 구별하기 위해 반드시 괄호를 사용해야 하지만 전위표기를 사용하는 MathML에서는 괄호를 사용하지 않아도 된다.

내용 마크업에서 사용되는 대부분의 내용 요소들은 연산이나 수학적 데이터 타입을 표시한다. 아래의 예에서 보는 자와 같이 <apply> 요소는 이와 같은 연산들과 변수들을 그룹화하는 역할을 한다. <apply> 요소의 첫 하위요소는 연산이 되어야 하고 나머지 하위요소들은 그 연산에 대한 변수들이 되어야 한다. MathML에서 연산과 함수들은 대부분 <cos/>, <intersect/>과 같이 빈 요소들로 표시된다.

내용 마크업에서의 가장 보편적인 토큰 요소는 <ci>와 <cn>이다. 이들은 각각 식별자와 수를 표시한다. 새로운 기호, 연산, 함수의 정의를 위하여 <csymbol> 토큰이 사용될 수도 있다. 내용 마크업에는 약 12개 정도의 속성을 가진 100여 개의 요소들이 있다. $(a+b)^2$ 을 위한 MathML 내용 마크업은 다음과 같다

```
<apply>
  <power/>
  <apply>
    <plus/>
    <ci>a</ci>
    <ci>b</ci>
  </apply>
  <cn>2</cn>
</apply>
```

2.2 MathML을 이용한 사례

MathML은 웹에서의 수식 사용에 대한 제약을 극복할 수 있는 가능성을 열어 주고 있으므로 이를 수학교과 ICT-활용 교육에 적용하는 것은 다양한 교수-학습 전략을 사용할 수 있음을 의미한다. 그러나 아직까지는 MathML 관련 도구들이 보편화되어 있지 않기 때문에 ICT-활용 교육에서 MathML을 활용한 경우는 드물다.

본 절에서는 MathML을 편리하게 사용할 수 있도록 개발된 수학적 생성기와 MathML을 이용하여 구현된 문제풀이 저작도구에 대해 살펴보기로 한다.

2.2.1 MathML을 위한 수학적 생성기

특정 수식에 대한 MathML 코드를 일일이 손으로 표현하는 것은 상당히 복잡한 작업을 요구하므로 보다 쉽게 수식을 MathML 코드로 변환할 수 있는 MathML 생성 도구들이 필요하게 되었다. 이를 위한 도구들 중 대표적인 것이 MathType(Design Science, 2002a)으로 수식 템플릿을 이용하여 수식을 입력하면 그것을 MathML이나 TeX 형식으로 변환시켜 준다. <그림 3>은 MathType에서의 MathML 코드 생성 화면이다.

료가 제공된다.

이은주(2002)의 개별학습을 위한 웹기반 수준별 교수-학습 시스템도 학습 주제 선택, 기본학습, 형성평가, 수준별 학습에 관련된 내용 등을 모두 데이터베이스로 구축하였다. 이 시스템은 형성평가 결과에 따른 수준별 학습을 위하여 별도의 학습자료를 만드는 대신 데이터베이스에 저장되어 있는 문제를 난이도별로 추출한 수준별 학습지를 자동 생성하여 제공하도록 구현되어 있다.

기존의 웹기반 수준별 교수-학습 시스템은 각 학생들의 학습 능력을 파악하고 그 능력에 맞는 적절한 학습자료를 제공하며, 이를 통한 학습진행 정도와 결과를 교사가 효과적으로 관리할 수 있도록 하는 것에 중점을 두고 있다. 그리고 수준별로 각각의 학습자료를 만들거나 데이터베이스에 저장된 문제를 난이도별로 추출하는 방법을 사용하여 수준별 학습자료를 제공하고 있다. 앞에서 살펴본 시스템들처럼 각 수준별 학습자료를 독립적으로 작성하게 되면 두 개 이상의 수준에 공통적으로 포함되어야 하는 학습내용의 경우 수준마다 반복적으로 작성해야 하는 불편함이 있고, 모든 수준에 공통적으로 들어있는 특정 학습 내용을 수정하고자 할 경우 모든 학습 자료를 일일이 수정하여야 한다는 어려움이 있다. 그리고 문제의 난이도를 이용하여 수준별 학습지를 만들게 되면 문제를 제외한 다른 학습내용을 수준별 학습지에 포함시키지 못한다는 단점을 가지게 된다.

그러므로 이러한 시스템들이 수학교과에서 보다 효과적으로 활용되기 위해서는 본 논문에서 제시하는 바와 같이 수학적 정보를 포함할 수 있으면서, 수준별로 편리하게 활용될 수 있는 학습자료의 개발을 위한 노력이 필요하다.

3. MathML을 이용한 수준별 교수-학습 자료

최근 교육인적자원부에서 시도 교육과학연구원과 공동으로 개발한 제7차 교육과정의 교수-학습 자료(교육인적자원부, 2001a)를 바탕으로 XML과 MathML 그리고

XSLT를 이용하여 수준별 교수-학습 자료를 만들었고, 이를 수준별 수업에서 활용하는 방안을 제시하였다.

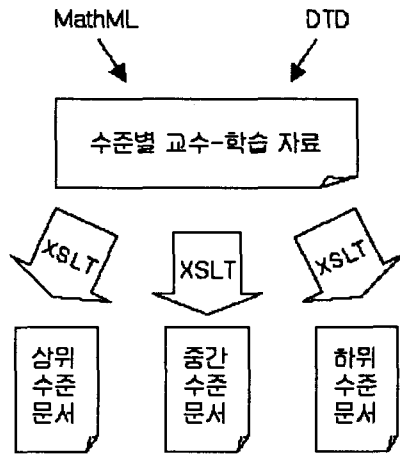
3.1 개발 방향

3.1.1 개발 개요

기존의 웹페이지에서는 수식을 주로 그림을 이용하여 표현하고 있는데, 이런 경우 수식의 크기를 바꾸거나 내용을 수정하는 것이 어렵다. 예를 들어, 글자 크기 10포인트의 텍스트와 연결되어 있는 수식을 위한 그림은 글자 크기의 변화에 따라 그림의 크기를 줄이거나 늘이면 현저하게 가독성이 떨어진다. 따라서 글자 크기를 수정하거나 수식 자체를 수정해야 할 경우 개발자는 그림 파일을 수정하거나 재작성해야 한다. 또한 그림으로 표현된 수식은 수식의 내재적 의미에 대한 정보를 전혀 포함하고 있지 못하므로 다른 수학 응용프로그램에서 활용할 수가 없다.

MathML을 이용하여 수식을 표현할 경우 위와 같은 문제점을 해결할 수 있다. MathML은 마크업 형태로 수식을 표현하기 때문에 문서의 글자 크기를 수정하면서 같은 소스 코드 내에서 수식의 크기를 조절하거나 수식 자체의 내용을 수정할 수 있도록 해 준다. 그리고 MathML을 지원하는 Mathematica와 같은 수학 응용프로그램에서 교수-학습 자료에 사용되었던 수식 정보를 재사용할 수 있다는 장점을 가진다.

수준별 교수-학습 자료를 개발할 때에는 각 수준을 통합하여 개발하는 것이 학습내용과 문제 수준을 정확히 판단하고 비교할 수 있기 때문에 각 수준별로 분리해서 개발하는 것보다 유리하다. 그러나 학생들과 수업을 할 때에는 각 수준에 맞는 자료를 가지고 수업을 해야 한다. XML을 기반으로 만든 교수-학습 자료는 이와 같은 작업을 가능하게 한다. 즉, 모든 수준이 통합되어 있는 기본 문서에서 수준별로 필요한 부분만을 추출하여 수준별 교수-학습 자료를 만들 수 있다. 그리고 난이도별로 문제를 추출할 수도 있고 원하는 학습주제만 추출하여 활용할 수도 있다. 본 논문에서 제안하는 수준별 교수-학습 자료의 개발의 개요를 <그림 3>에 나타내었다.



<그림 3> 제안한 개발방법의 개요

3.1.2 수준별 교수-학습 자료의 구성요소

본 논문에서는 교육인적자원부에서 개발한 수준별 교수-학습 자료의 구성요소들 중에서 단원의 개관, 준비학습, 기본과정, 참고자료는 모든 수준의 학생들에게 꼭 필요한 기본 구성요소들이고 형성평가, 보충자료, 심화자료, 특별보충자료는 학생들의 수준에 따라 선택적으로 제공되어야 하는 선택적인 구성요소라는 관점에서 수준별 교수-학습 자료를 만들었다.

수준별 수업을 운영하는 방법은 각 학교의 교실 여건, 교사 정원, 과목별 수업시수 등에 따라 달라질 수 있다. 즉, 각 학교의 실정에 따라 학생들의 수준을 상·중·하 세 단계로 나눌 수도 있고, 상·하 두 단계로 나눌 수도 있다. 그리고 수준별로 반을 재편성하여 이동수업을 할 수도 있고, 학급 내에서 수준별 분단을 편성하여 수업을 할 수도 있다.

수준별 수업을 운영함에 있어서 학생들의 수준을 상, 중, 하 세 단계로 나눌 경우, 기본 구성요소와 심화자료는 상위 학생들에게, 기본 구성요소와 형성평가는 중간 학생들에게, 기본 구성요소와 보충자료는 하위 학생들에게 제시하여 수준별로 수업을 진행할 수 있고, 상·하

두 개의 수준으로 나눌 경우에는 상위 학생에게는 기본 구성요소, 형성평가, 심화자료를 제공하고 하위학생들에게는 기본 구성요소, 형성평가, 보충자료를 제공하면 된다. 그리고 특별보충자료는 진급을 위한 필수적인 학습 내용을 포함하는 것으로 특별보충과정에 참여하는 학생들을 위한 자료로 제공될 수 있다.

<표 1> 수준별 교수-학습 자료의 구성요소

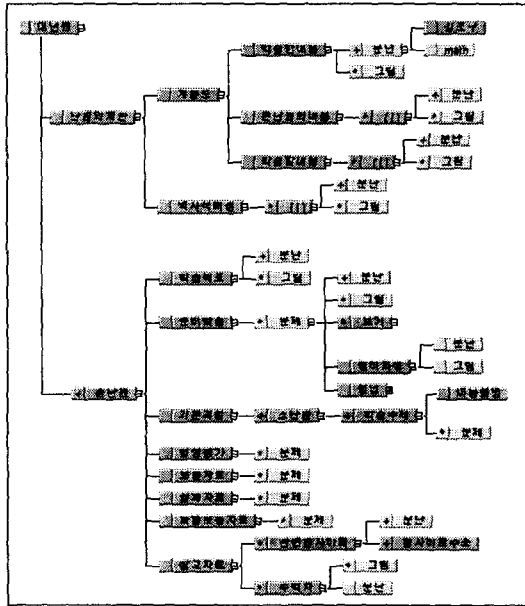
구성요소	3단계				2단계		
	상	중	하	특별보충	상	하	특별보충
단원의개관	○	○	○	○	○	○	○
준비학습	○	○	○	○	○	○	○
기본과정	○	○	○	○	○	○	○
보충자료	x	x	○	x	x	○	x
형성평가	x	○	x	x	○	○	x
심화자료	○	x	x	x	○	x	x
특별보충자료	x	x	x	○	x	x	○
참고자료	○	○	○	○	○	○	○

<표 1>은 수준별 교수-학습 자료에서의 구성요소 포함 여부를 보여준다. 이는 교육인적자원부의 중학교 교육과정 편성·운영 자료 중 수준별 교육과정 편성·운영 자료(교육인적자원부, 2001)를 참고하였다.

3.2 수준별 교수-학습 자료의 개발

3.2.1 문서의 구조 정의

본 논문에서는 제안하는 교수-학습 자료를 위한 문서의 구조는 <그림 4>와 같다. 그림에서 사각형 기호 (■)는 하나의 XML 태그에 해당되며, 기호 앞부분의 +, * 기호는 각각 해당 태그가 1번 이상 또는 0번 이상 반복되어 나타날 수 있음을 표시한다. 또한 사각형들에 대하여 좌우로 선으로 연결한 것은 XML 태그의 포함관계를 나타낸다. 예를 들어, '대단원'은 '단원의 개관'과 '중단원'을 포함하고 있으며, '학습목표'는 '문단'과 '그림'으로 구성된다.



<그림 4> 수준별 교수-학습 자료의 문서 구조

DTD(Document Type Definition)는 유효한 XML 문서를 저작하기 위한 규칙들의 집합으로 문서의 구조를 정의하고 요소와 속성을 선언하기 위하여 사용된다.

앞에서 살펴본 것처럼 교수-학습 자료가 개발될 때에는 필요한 구성요소를 모두 포함하고 있어야 한다는 것을 전제로 문서의 구조를 정의하는 DTD를 개발하였다. 이는 만들어진 문서의 유효성을 검증하는 것뿐만 아니라 수준별 교수-학습 자료를 만들기 위한 응용프로그램 개발의 기본 구조가 될 수 있다. 그리고 정의된 DTD를 준수하여 만들어진 모든 XML 문서는 문서의 교환이나 재사용이 가능하므로 교수-학습 자료의 활용가치를 높여 줄 수 있다.

<그림 4>에서의 문서 구조를 DTD로 나타내면 다음과 같다. MathML을 위한 DTD는 W3C의 mathml.dtd(W3C, 2002c)를 이용하였다.

```
<!ENTITY % math SYSTEM
    "http://www.w3.org/Math/DTD/mathml1/mathml.dtd">
%math;
<!ELEMENT 대단원 (단원의개관, 중단원+)>
<!ELEMENT 단원의개관 (개통도, 역사적배경)>
```

```
<!ELEMENT 계통도 (학습한내용, 본단원의내용, 학습할내용)*>
<!ELEMENT 학습한내용 (문단+ | 그림)*>
<!ELEMENT 문단 (#PCDATA | 강조구 | math)*>
<!ELEMENT 강조구 (#PCDATA)>
<!ELEMENT 그림 (#PCDATA)>
<!ELEMENT 본단원의내용 (문단+ | 그림)*>
<!ELEMENT 학습할내용 (문단+ | 그림)*>
<!ELEMENT 역사적배경 (문단+ | 그림)*>
<!ELEMENT 중단원 (학습목표, 준비학습, 기본과정, 형성평가,
    보충자료, 심화자료, 특별보충자료, 참고자료)>
<!ELEMENT 학습목표 (문단+ | 그림)*>
<!ELEMENT 준비학습 (문제)*>
<!ELEMENT 문제 (문단+ | 그림* | 보기* | 풀이과정 | 정답)*>
<!ELEMENT 보기 (#PCDATA | 목록)*>
<!ELEMENT 목록 (#PCDATA | math)*>
<!ELEMENT 풀이과정 (#PCDATA | 문단 | 그림)*>
<!ELEMENT 정답 (#PCDATA | math)*>
<!ELEMENT 기본과정 (소단원)>
<!ELEMENT 소단원 (학습주제)>
<!ELEMENT 학습주제 (내용설명, 문제)*>
<!ELEMENT 내용설명 (문단+ | 그림* | 요약+)*>
<!ELEMENT 요약 (#PCDATA | 문단)*>
<!ELEMENT 형성평가 (문제)*>
<!ELEMENT 보충자료 (문제)*>
<!ELEMENT 심화자료 (문제)*>
<!ELEMENT 특별보충자료 (문제)*>
<!ELEMENT 참고자료 (관련웹사이트* | 수학자)*>
<!ELEMENT 관련웹사이트 (문단+ | 웹사이트주소)*>
<!ELEMENT 웹사이트주소 (#PCDATA)>
<!ELEMENT 수학자 (그림+ | 문단+)*>
<!ATTLIST 대단원
    대단원명 CDATA #REQUIRED>
<!ATTLIST 문단
    정렬 (왼쪽 | 가운데 | 오른쪽) #IMPLIED>
<!ATTLIST 중단원
    중단원명 CDATA #REQUIRED>
<!ATTLIST 문제
    종류 CDATA #REQUIRED
    수준 (1 | 2 | 3) #REQUIRED
    번호 CDATA #REQUIRED>
<!ATTLIST 소단원
    소단원명 CDATA #REQUIRED>
<!ATTLIST 학습주제
    학습주제명 CDATA #REQUIRED>
<!ATTLIST math
    xmlns CDATA #REQUIRED>
```

3.2.2 XML 문서 작성

위의 같이 작성된 DTD를 기준으로 수준별 교수-학

습 자료를 위한 XML 문서를 작성할 수 있다. XML 문서의 작성에 있어서 GUI 기반의 저작 도구를 사용하면 누구나 쉽게 교수-학습을 위한 XML 문서를 작성할 수 있다.

교수-학습 자료의 기본 틀은 위에서 제시한 DTD를 기준으로 작성되어야 하고 수식 표현을 위한 MathML은 W3C의 mathml.dtd를 기준으로 작성되어야 한다. MathML을 지원하는 수식편집기로는 EzMath, Amaya, MathType 등이 있으나 본 논문에서는 MathType (Design Science, 2002a)을 이용하여 XML문서에 수식을 위한 MathML 표현 마크업 코드를 삽입하는 방법을 사용하였다.

XML 문서는 대단원을 기준으로 만들어지며 대단원은 단원의 개관과 여러 개의 중단원으로 구성된다. 중단원은 학습목표, 준비학습, 그리고 여러 개의 학습주제를 가진 소단원의 반복으로 이루어지는 기본과정, 수준별 문제의 반복으로 이루어지는 형성평가, 보충자료, 심화자료, 특별보충자료로 구성된다. 참고자료도 중단원에 포함된다.

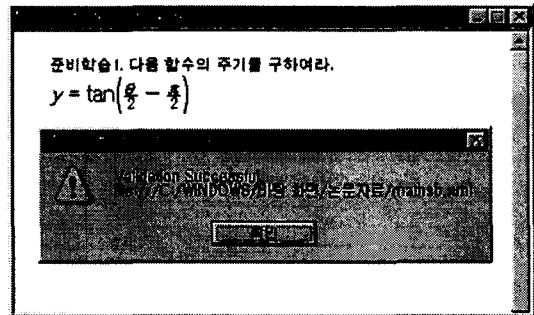
수식을 표현하기 위해 사용되는 MathML 코드는 `$` 태그로 시작되고 `$` 태그로 끝이 나와 한다. MathML 코드는 <문단>, <목록>, <정답> 요소 내에서 사용될 수 있고, 반복되어 사용될 수 있다.

수준별 교수-학습을 위한 XML 문서의 한 예를 다음에 예시하였다. 문서의 전체적인 구조와 수식 표현을 중심으로 보여주기 위하여 문서의 일부분을 점(...)으로 표시하였다. 굵은 글씨 부분은 수식 표현에 관련된 부분을 나타낸다.

```
<?xml version="1.0" encoding="euc-kr" standalone="no"?>
<!DOCTYPE 대단원 SYSTEM "mathsb.dtd">
<?xml-stylesheet type="text/xsl" href="mathsb.xsl"?>
<대단원 대단원명="IV. 삼각함수">
  <단원의개관>... </단원의개관>
  <중단원 중단원명="1. 삼각형의 응용">
    <학습목표>... </학습목표>
    <준비학습>... </준비학습>
    <기본과정>
      <소단원 소단원명="§1. 사인법칙과 코사인법칙">
        <학습주제 학습주제명="(1) 사인법칙">... </학습주제>
      </소단원>
    </기본과정>
```

```
<형성평가>
  <문제 종류="형성평가" 수준="1" 번호="1">
    <문단>아래의  $\triangle ABC$ 에서 <math
      xmlns="http://www.w3.org/1998/Math/MathML">
      <mrow>
        <mo>&ang;</mo><mi>B</mi>
      </mrow>
    </math>의 크기와 a의 길이를 구하여라
  </문단>
  <그림>hun01.gif</그림>
  <정답 이름="hun1">
    <math
      xmlns="http://www.w3.org/1998/Math/MathML">
      <mrow>
        <mi>x</mi><mo>+</mo><mn>2</mn>
      </mrow>
    </math>
  </정답>
</문제>
</형성평가>
<보충자료>...</보충자료> <심화자료>...</심화자료>
<특별보충자료>...</특별보충자료>
<참고자료>...</참고자료>
</중단원>
</대단원>
```

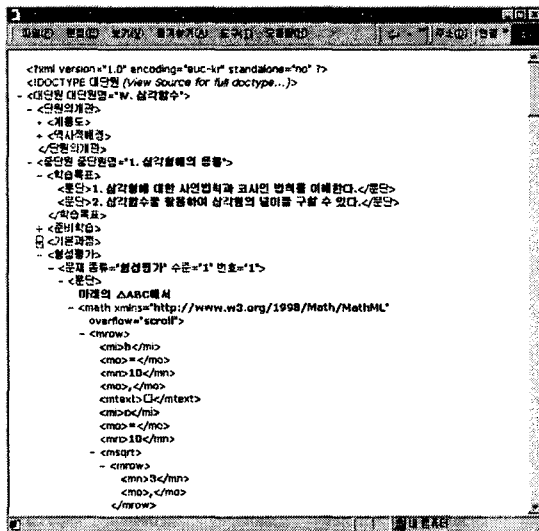
작성된 문서가 XML 형식에 맞으면서 DTD에서 정의된 문서의 구조에 맞게 작성되었는지를 확인하기 위해서는 유효성 검증 파서(parser)를 사용할 수 있다. 파서는 수식 입력 부분이나 기본 문서가 유효하게 작성되었는지 아닌지를 검증하여 그 결과를 알려준다. 유효성 검증 파서로는 MSXML, XML Spy 등 많은 도구가 개발되어 있다. <그림 5>는 MSXML3.0 파서의 유효성 검증 결과를 보여 준다.



<그림 5> XML 문서의 파서 검증 결과

3.2.3 XML 문서의 출력

작성된 XML 문서는 스타일 시트(style sheet)에 의해 브라우저나 응용프로그램에서 출력되는데, 다양한 스타일 시트를 적용함으로써 여러 가지 형태로 활용될 수 있다. Microsoft의 인터넷 익스플로러(IE)에는 스타일 시트 처리기가 내장되어 있어 작성된 문서를 <그림 6>과 같이 화면에 출력할 수 있다. IE에서의 기본 출력은 문서의 각 요소와 그것이 포함하고 있는 데이터를 계층적인 형태로 화면에 출력해 주는 것으로 문서의 구조를 파악하기에는 편리하나 교수-학습 자료로 활용하기에는 문서의 가독성이 떨어진다.



<그림 6> IE에서의 XML 문서 출력

XML 문서를 표시하거나 특정 형식을 지정해 주기 위하여 XSLT가 사용될 수 있다. XML 문서는 XSLT에 의해 HTML 형식으로 출력되어 브라우저 화면에 나타나게 된다. 이를 위하여 XSLT 파일에는 각 구성요소의 출력 형식을 지정하는 템플릿과 이들 템플릿의 적용 여부 그리고 문서의 외형을 지정해 주기 위한 HTML 코드들이 포함되어 있다. 본 논문에서는 XML로 표현된 교수-학습 자료를 수학 교재의 형태로 출력하기 위해서 XSLT를 이용하여 스타일 시트를 작성하였다.

학생들의 수준에 따라 선택적으로 제공되어야 하는 구성요소들은 서로 다른 템플릿으로 만들어져 있어야 쉽

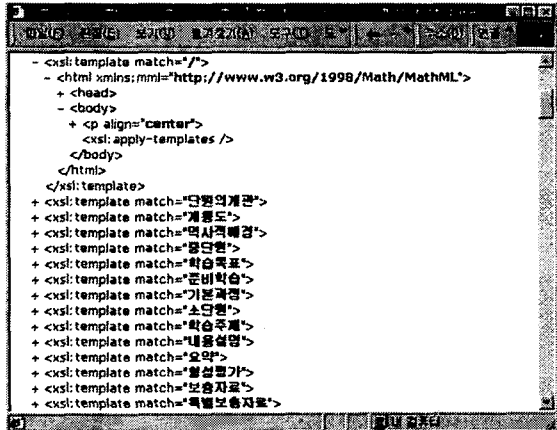
게 수준별 교수-학습 자료를 추출할 수 있다. 수식의 경우에는 IE에서 MathML을 표시할 수 있는 디스플레이 엔진인 MathPlayer를 이용하여 화면에 출력하게 된다. 이 방법은 수식을 출력하기 위한 복잡한 작업을 하지 않아도 된다는 장점이 있고, 본 논문의 수준별 교수-학습 자료와 같이 다양한 문서 형식 안에 수식을 위한 MathML 코드를 삽입하여 사용할 수 있게 해 준다. 다음은 수식을 화면에 출력하기 위한 XSLT 파일의 소스 코드이다.

```

.....
<head>
<object id="mmlFactory"
classid="clsid:32F66A20-7614-11D4-BD11-00104BD3F987">
</object>
<?IMPORT namespace="mml" implementation="#mmlFactory"?>
<title>
<xsl:value-of select="대단원/@대단원명"/>
</title>
</head>
.....
<xsl:template match="math">
<xsl:apply-templates select="*" />
<xsl:template match="/">
<xsl:apply-templates/>
</xsl:template>
<xsl:template match="text()">
<xsl:value-of select="."/>
</xsl:template>
<xsl:template match="*[@*]">
<xsl:copy>
<xsl:apply-templates select="*|text()|@*" />
</xsl:copy>
</xsl:template>
<xsl:template match="mml:*">
<xsl:eval no-entities="t" doc:id="eval">'&lt;mml:' +
this.nodeName.substring(this.nodeName.indexOf('"')+1)
</xsl:eval>
<xsl:for-each select="@*">
<xsl:eval no-entities="t">' ' + this.nodeName</xsl:eval>
="<xsl:value-of select="."/>"
</xsl:for-each>
<xsl:eval no-entities="t">'&gt;'</xsl:eval>
<xsl:apply-templates select="*|text()"/>
<xsl:eval no-entities="t">'&lt;/mml:' +
this.nodeName.substring(this.nodeName.indexOf('"')+1)
+'&gt;'
</xsl:eval>
</xsl:template>
</xsl:apply-templates>
</xsl:template>

```

기본 문서의 출력을 위해서는 각 구성요소들을 표현하기 위한 템플릿이 모두 적용되어야 한다. 이를 위한 XSLT 파일은 <그림 7>과 같다. XSLT 파일의 구조를 간결하게 보이기 위하여 웹브라우저에서 표시하였다.



<그림 7> 기본 문서를 위한 XSLT

4. 수준별 교수-학습 자료의 활용

XML 문서의 장점 중의 하나는 사용 목적에 따라 서로 다른 스타일 시트를 적용하여 다양하게 자료를 활용할 수 있다는 것이다. 본 논문에서는 이와 같은 XML 문서의 장점을 활용하여 수준별로 필요한 구성요소를 추출하여 수준별 교재로 활용하는 방법과 형성평가의 결과에 따라 서로 다른 수준의 학습 내용을 제공하는 수준별 개별학습에서의 활용 방법에 대해 설명한다.

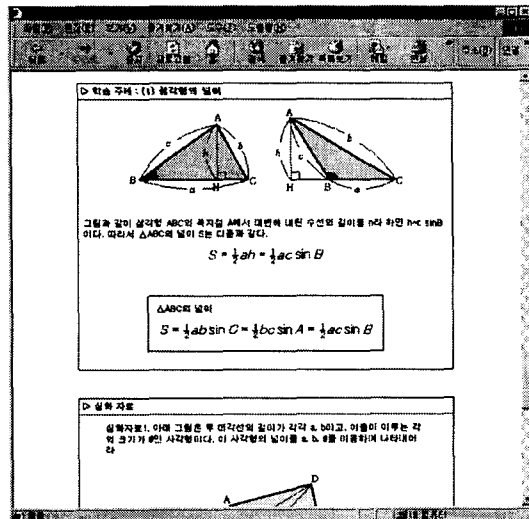
4.1 수준별 교재로서의 활용

상위 수준에 있는 학생들을 위한 수준별 교수-학습 자료를 출력하기 위해서는 앞에서 언급한 교수-학습 자료의 구성요소들 중에서 단원의 개관, 준비학습, 기본과정, 심화자료, 참고자료 요소들만 추출하여 출력하면 된다. 이를 위해서는 <그림 7>의 기본 문서를 위한 XSLT 파일을 수정하여 상위 수준 문서에 필요한 템플릿만 적용되도록 <표 2>와 같은 XSLT 파일을 적용하면 된다. 그리고 XML 형식의 기본 문서가 수정된 XSLT 파일을 참조하도록 하면 <그림 8>과 같이 기본과정 다음에 심화자료가 출력되는 상위 수준의 학생들을 위한 교수-학습 자료를 화면에 출력할 수 있다.

<표 2> 상위 수준 구성요소 추출을 위한 XSLT

```

<xsl:template match="/">
  . . . . .
  <body>
    <p align="center">
      대단원 : <xsl:value-of select="대단원/@대단원명"/>
    </p>
    <xsl:apply-templates select="대단원/단원의개관"/>
    <xsl:apply-templates select="대단원/중단원/학습목표"/>
    <xsl:apply-templates select="대단원/중단원/준비학습"/>
    <xsl:apply-templates select="대단원/중단원/기본과정"/>
    <xsl:apply-templates select="대단원/중단원/심화자료"/>
    <xsl:apply-templates select="대단원/중단원/참고자료"/>
  </body>
  . . . . .
</xsl:template>
  
```



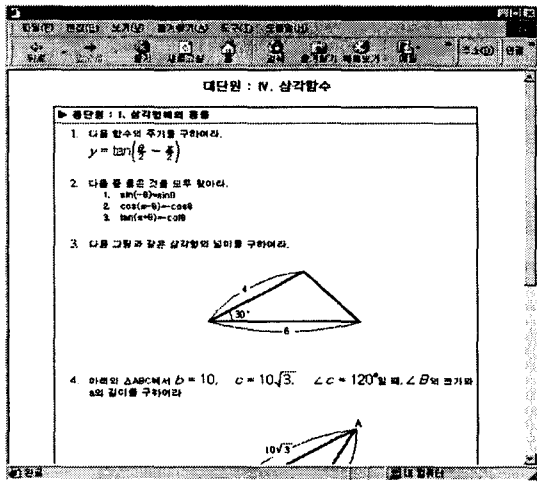
<그림 8> 상위 학생을 위한 문서 출력 화면

기본 문서에 들어 있는 문제들을 수준별로 추출하여 수준별 문제지로 활용할 수도 있다. 이를 위하여 기본 문서의 <문제> 요소는 종류, 수준, 번호와 같은 속성을 가진다. XSLT에서는 이 요소의 「수준」 속성을 이용하여 원하는 수준의 문제만을 추출하여 출력할 수 있다. <표 3>은 이를 위한 XSLT 코드이고, <그림 9>는 출력 화면이다.

<표 3> 난이도별 문제 추출을 위한 XSLT

```

<body>
<p align="center">
대단원 : <xsl:value-of select="대단원/@대단원명"/>
</p>
<ol>
<xsl:for-each select="//문제[@수준 = '1']">
<li/><xsl:apply-templates/><br/>
</xsl:for-each>
</ol>
</body>
    
```



<그림 9> 난이도별 문제 출력 화면

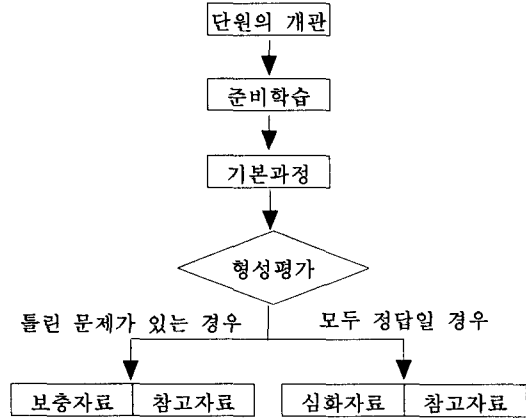
4.2 웹기반 개별학습에서의 활용

웹을 이용한 개별학습은 수준별 교수-학습을 위한 수업 형태 중의 하나이다. 이는 각 학생의 수준에 따라 선택적인 학습 자료를 제공해 주는 형태로 진행되거나 학생들의 반응에 따라 적절한 수준의 피드백을 제공해 주는 형태로 진행될 수 있다.

앞에서 만들어진 수준별 교수-학습 자료는 이와 같은 수준별 개별학습을 위한 자료로 활용될 수 있다. 기본과정을 마친 학생들에게 형성평가를 스스로 풀도록 하고 그 결과에 따라 보충자료와 심화자료를 제공하는 형태로 교수-학습 자료를 활용할 수 있다.

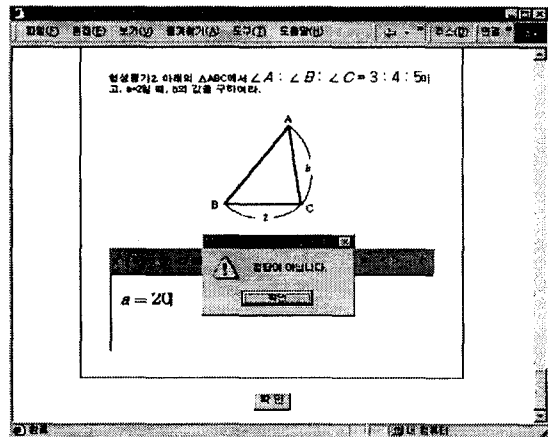
본 논문에서는 형성평가 문제에 대한 학생들의 응답이 모두 정답일 경우에는 상위 수준의 학습내용인 심화자료와 참고자료를 제공하고, 그렇지 않은 경우에는 하

위 수준의 학습내용인 보충자료와 참고자료를 제공하도록 구성하였다. <그림 10>은 이와 같은 학습의 진행 과정을 보여 준다.



<그림 10> 학습의 진행과정

형성평가 문제에 대한 답을 입력할 때는 MathML을 지원하는 수식입력 애플릿인 WebEQ 자바 툴킷(Design Science, 2002b)을 이용하여 학생들 스스로 답을 입력하도록 하였다. 이 방법은 정답에 수식을 포함시킬 수 있으므로 선택형 문제보다 평가 결과의 정확성을 높일 수 있고, 학생들에게 수식을 작성하는 방법에 대한 연습을 할 수 있도록 해 준다. <그림 11>은 형성평가 문제에 대한 답을 입력하고 그것의 정답 여부를 확인하는 과정을 보여 준다.



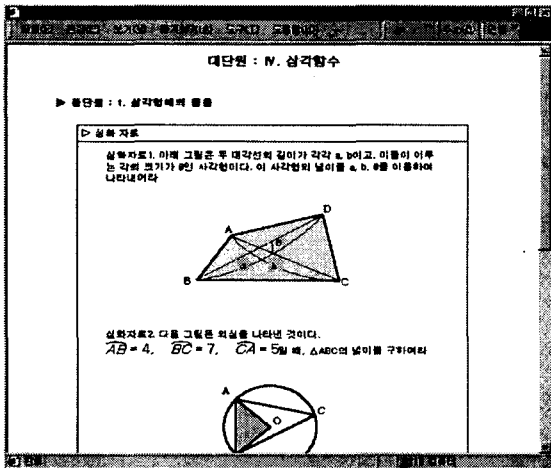
<그림 11> 형성평가 문제에 대한 응답 결과

형성평가 문제의 정답은 기본 문서의 <정답> 요소 안에 MathML 코드 형태로 입력되어 있고, 수식입력 애플릿은 학생들이 입력한 답을 MathML 코드로 변환하여 준다. 자바스크립트와 XSLT를 이용하면 정답과 학생들이 입력한 답을 서로 비교하여 정답 여부를 판별할 수 있으며 형성평가 결과에 따라 학습 내용이 서로 다르게 출력되도록 구성할 수 있다. <그림 12>는 이를 위한 자바스크립트가 포함된 XSLT 코드를 보여 준다.



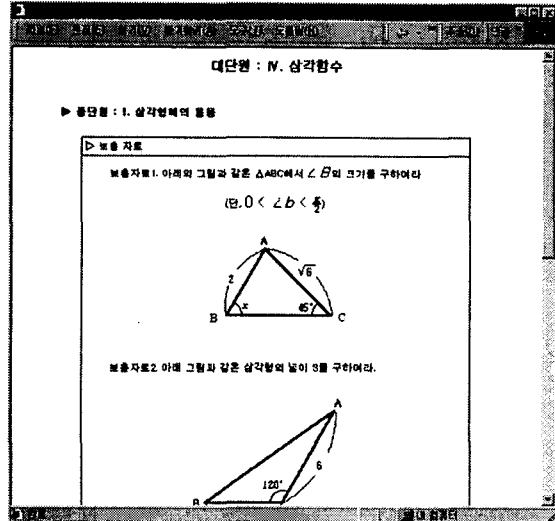
<그림 12> 형성평가 결과 처리를 위한 XSLT

<그림 13>은 형성평가 문제가 모두 정답일 경우에 제공되는 상위 수준의 학생들을 위한 수준별 교수-학습 자료이다.



<그림 13> 상위 수준의 형성평가 이후 학습내용

<그림 14>는 형성평가 문제 중 틀린 문제가 있을 경우에 제공되는 하위 수준의 학생들을 위한 수준별 교수-학습 자료이다.



<그림 14> 하위 수준의 형성평가 이후 학습내용

5. 결 론

ICT-활용 교육에 있어서 수학교과외의 경우 수식 사용이 자유롭지 못하다는 이유로 인해 다양한 교수-학습 전략을 사용하지 못하고 있다. 이러한 수식 표현의 문제를 해결하고 보다 다양하게 수식 정보를 활용할 수 있도록 하기 위한 노력의 대표적인 것이 MathML이다. MathML은 XML의 응용 분야이고 수식의 내재적 의미를 포함할 수 있도록 설계되었기 때문에 수식을 자유롭게 표현할 수 있다는 장점을 가질 뿐만 아니라 수준별 교수-학습 자료와 같이 다양한 형태로 활용될 수 있는 ICT-활용 교육을 위한 자료를 개발할 수 있게 한다.

본 논문에서는 XML과 MathML을 이용하여 수준별 교수-학습 자료를 작성하는 방법과 이 자료를 수준별 교재와 개별학습 자료로 활용하는 방안을 제시하였다. 이와 같은 수준별 교수-학습 자료는 수식의 수정을 쉽게 하고 다른 수학 응용프로그램에서 수식을 재사용할 수 있게 한다. 그리고 학생들의 수준에 맞게 제공되어야 하는 선택적인 학습 내용들을 추출하여 수업에 활용할

수 있게 하고, 교수-학습 자료에 포함되어 있는 문제들을 수준별로 분리하여 수업에 활용할 수 있게 한다.

특히, 여러 수준의 학습 자료를 하나의 문서로 작성할 수 있도록 함으로써 문서의 수정이나 유지관리를 쉽게 할 수 있다. 또한 수준별 자료의 작성에 있어서 기존의 방식에서는 구성요소를 선택함으로써 각 수준의 문서를 작성하였지만, 본 논문에서 제안하는 방법에서는 문단이나 문제 단위로 수준을 부여할 수 있으므로 더욱 세밀한 수준별 교수-학습 자료의 작성이 가능하다.

현재 수준별 교수-학습 자료의 제작에 있어 MathML과 XML을 직접 입력하여야 하지만, 앞으로 GUI 기반의 저작도구의 개발된다면 더욱 손쉽게 문서를 작성할 수 있을 것이다.

참 고 문 헌

- 교육인적자원부 (2001a). 제7차 교육과정 공동개발 교수-학습자료, (http://www.ienet.re.kr/2001_teacher/main.htm).
- 교육인적자원부 (2001b). 중학교 교육과정 편성·운영 자료(III) 수준별 교육과정 편성·운영의 실제, 서울: 교육인적자원부.
- 김강문 (2000). 정보화시대의 대안적 수학교육 - 인터넷 활용을 중심으로, 울산대학교 석사논문.
- 김명신 (2001). 수준별 교육을 위한 웹기반 교수-학습 시스템의 설계 및 구현, 원광대학교 석사논문.
- 신수경 (2001). MathML을 이용한 중학교 수학문제 유형 별 풀이 저작도구 설계 및 구현, 성신여자대학교 석사논문.
- 이은주 (2002). 개별학습을 위한 웹기반 수준별 교수·학습 시스템, 서울교육대학교 석사논문.
- 한국교육학술정보원 (2001). 정보통신기술 활용 지도자료, 서울:교육인적자원부.
- 황경아 (2000). MathML을 위한 수학적 생성기 구현에 관한 연구, 대전산업대학교 석사논문.
- 황재연 (2001). 수학 교과목의 수준별 교육과정 운영을 위한 웹코스웨어의 설계 및 구현, 한국교원대학교 석사논문.
- Design Science (2002a). "MathType," (<http://www.dessci.com/features/basic.stm>).
- Design Science (2002b). "WebEQ,"(<http://www.dessci.com/webmath/webeq/features.stm>).
- R. Miner & J. Schaeffer (2001). "A Gentle Introduction to MathML," (<http://www.dessci.com/support/tutorials/mathml/default.stm>).
- W3C (2001). "Mathematical Markup Language (MathML) Version 2.0," (<http://www.w3.org/TR/MathML2/>).
- W3C (2002a). "XSL Transformations (XSLT) Version 2.0," (<http://www.w3.org/TR/xslt20/>).
- W3C (2002b). "Amaya Overview," (<http://www.w3.org/Amaya/Amaya.html>).
- W3C (2002c). "A Parsing MathML," (<http://www.w3.org/TR/MathML2/appendixa.html>).

An Application of MathML and XSLT for Differentiated Instruction-Learning Materials in Mathematics Course

Hong, Eun-Pyo

Changwon-Sangnam Middle School

E-mail: suuaa@hitel.net

Lee, Su-Hyun

Department of Computer Engineering, Changwon National University

E-mail: suhyun@sarim.changwon.ac.kr

In this paper, we propose an authoring methodology of differentiated instruction-learning materials in mathematics course using MathML and XSLT, and give utilizing suggestions of these materials as differentiated textbook and individual learning. Incorporating MathML into instruction-learning materials, there are many advantages such that view of the materials in web browser is improved since mathematical expressions are compatible with text, and an expression can be modified in more convenient way since the expression is represented in text format not an image. In authoring of differentiated materials, the author provides one document for various levels of students. Then, each level of materials is generated from the document via XSLT transformation. This enables author-driven progress control rather than student-driven which does in previous approaches, since students don't need to select components of the material.

-
- * ZDM Classification : U50
 - * 2000 Mathematics Subject Classification : 97U50
 - * Key Word : Differentiated learning, MathML, XML, Lecture using ICT.