SGML 표기법을 이용하는 수식 편집기의 설계 및 구현

김 태 훈[†] · 현 득 창[†] · 이 수 연[††]

요 약

현재, 과학, 기술 문서 등에 많이 사용되는 수식의 표기법이 TeX, EQN등과 같이 특정 시스템에 의존하여 시스템간의 상호교환이 안되고 있다. 따라서 본 논문에서는 워드프로세서나 전자출판등에서 작성되는 수식 정보의 공유 및 시스템간 상호 교환성을 확보하기 위하여, 국제 표준 SGML 표기법으로 수식을 표현할 수 있고, 또한 수식용 문서형 정의(Document Type Definition)에 따라 수식 구조의 오류 검사 기능을 갖는 대화형 수식 편집기의 설계 및 구현을 한다. 사용자 접속으로는 수식 심볼 아이콘에 대한 마우스 입력과 키보드 입력 등의 직접 조작 방식을 구현하였다. 구현 환경으로는 워크스테이션 상에서 UNIX를 운영 체제로 하는 X 윈도우 시스템과 그래픽컨 사용자 접속인 OSF Motif를 사용하였고 메뉴 및 화면 구성을 위하여 OSF UIL(User Interface Language)을 사용하였다.

An Implementation of Mathematics Editor Using SGML Notation

Taehoon Kim[†] · Deukchang Hyun[†] · Sooyoun Lee[††]

ABSTRACT

The notation of mathematics equation which is often used in the scientific and technical documents tend to depend on specific systems such as TeX and EQN. So, This paper describes an editor to generate automatically mathematics equations conforming to SGML notation which is developed for interchange of textual information made in Wordprocessor or DTP system, and also describes the design and implementation of interactive mathematics equation editor which has error detect facility using Document Type Definition for mathematics equation. As an user interface, a direct manipulation method such as selecting icons of mathematics symbols and on-spot keyboard input has been implemented. This mathematics editor has been implemented in UNIX environment which include X-window, OSF Motif as a graphical user interface and OSF UIL(User Interface Language) as a tool of designing the menus and screen layout.

1. 서 론

현재 워드 프로세서 등과 같이 컴퓨터를 이용한 텍스트 처리가 일반화되고 있지만, 사용자간에 문서를 전자 문서의 형태로 교환하는데 있어 생기는 애로점을 해결하기 위한 이 기종간의 시스템에서 상호교환하는 정보교환의 중요성이 날로 증대되고있다.

문서의 중요한 요소 중의 하나인 수식은 처리 단계가 매우 어렵고 다양하여 전용 처리 시스템들이 계속 개발되어져 왔다. 초기 수식 편집 시스템은 수식을 시스템 고유의 명으로 입력하는 방식으로 수식 입력이 매우 까다로웠고 수식을 수정하기도 매우 어려웠

† 준 회 원:광운대학교 컴퓨터공학과
†† 정 회 원:광운대학교 컴퓨터공학과, 광운대학교 신기술연구소 연구원
논문접수:1994년 10월 28일, 심사완료:1996년 3월 20일

다. 그 예로써는 Bell Lab.의 NROFF/TROFF-EQN [1]이나 Knuth의 TeX[2] 등이 있다. 근래 윈도우를 중심으로 하여 수식을 심볼 위주로 입력하는 대화형 입력 방식으로 바뀌고 있으나, 수식의 표기법 등 서로 다른 환경을 가지고 있기 때문에 수식 문서를 다른 수식 처리 시스템으로 전송하거나 편집 시에 많은 불편함이 많다[3,4,5]. 그 예로써 Mac 시스템에서 사용되는 MathType, Frame Technology의 Frame Maker 등이 있다.

1980년대에 이르러 서로 다른 환경을 갖는 시스템들 간에서 문서의 상호 교환을 위하여 국제 표준화 기구(International Standard Organization:이후 ISO라 칭함)에서는 문서 관련 국제 표준을 제정하여 발표하였는데, 구체적으로 SGML(Standard Generalized Markup Language)[6], ODA(Open Document Architecture)[7]등을 들 수 있다.

SGML은 계층 구조를 갖는 정보를 표현하기에 적합한 메타언어(meta language)로써 문서 처리, 텍스트 정보의 저장과 검색, 하이퍼미디어와 같은 SGML 응용 분야에 적용할 수 있다[4,5].

그 중, 문서나 출판물에 SGML을 응용하기 위하여서는 문서나 출판물의 작성에 알맞는 형태로 엘리먼트의 명칭, 속성을 포함하는 엘리먼트의 계층 구조를 문서 형 정의(Document Type Definition:이후 DTD라 칭함)라는 형태로 정의하여야 한다. 따라서 SGML 응용에 적합한 DTD의 연구 및 공유는 이 기종간 문서 정보의 교환이나 하이퍼미디어 정보 교환에 매우 중요하다.

따라서 본 연구에서는 다음과 같은 요구 조건(User requirement)을 만족시키는 수식 편집기를 설계 구현한 것이다.

- SGML 표기법의 수식을 자동 생성.
- 수식의 논리적 구조 표현은 SGML DTD(ISO/IEC IS 9573)[2]을 기초로 한다.
- 사용자의 상호 작용의 제어:수식의 DTD를 기준하여 수식의 발생 구조의 오류 발견이 가능하여야 한다.
- 편리한 사용자 접속:직접 화면 조작[10].

본 논문의 구성은 2장에서는 수식 표기법, 3장에서는 SGML 표기법을 갖는 수식 편집기 시스템의 설계, 4장에서는 수식 편집기의 구현 및 고찰, 5장은 결론으로 하였다.

2. 수식 표기법

2.1 수식 표기법

수식의 표현은 수식 편집기를 중심으로 볼 때 수식 편집기의 입력 데이터 형식(편집 가능한 형식, 또는 포맷화되지 않은 형식이라고도 함)과 편집기에서 포맷 처리가 완료되어 화면 출력에 사용되는 데이터 형식(편집 불가능한 형식 또는 포맷화된 형식)으로 대별할 수 있다[4,5].

포맷화된 형식은 기존 수식 요소들의 배치 정보 및 폰트 정보 등이 포함되며 이러한 정보는 수식 편집기에 따라 각기 상이한 형식으로 표현되는 편집기 내부 표현이므로 서로 다른 수식 응용간에 수식 정보의 교환이 불가능하다.

이러한 편집에 관련된 정보는 수식 편집기의 포맷터가 적절히 이해하고 동작하기 위한 특정한 구문으로 정의된 명령으로 나타나게 되며 이를 소위 마크업이라고 한다.

마크업의 종류는 특정 시스템에 의존하는 방식이나 수식의 요소들의 배치처리를 제어하기 위한 정보가 포함되는 절차적 마크업(Procedural Markup)과 시스템 독립적이며 수식의 논리적 형식 중심으로 표현 가능한 범용 마크업(Generalized Markup)으로 나눌 수 있다[1,7]. <표 1>에서는 여러 가지 수식 표기 마크업 실례들이며 TeX및 EQN의 형식은 절차적 마크업 방식이나 SGML은 범용 마크업의 대표적인 예이다.

TeX나 EQN에서는 수식 요소를 제어하기 위하여 수식이 갖는 의미(Semantics)에 적합한 구조를 나타내기보다는 수식 요소의 배치 처리 중심으로 한 마크업 명령들을 사용하고 있다[2,3]. 예를 들면, Q^2_c의 경우, 의미적으로는 Q_c의 자승이 맞지만, EQN에서는 alpha Q sup 2 sub c로 표현(즉 Superscript 2, Subscript c와 같은 배치 정보로 표현)하고 있으며, 특히 alpha Q sub c sup 2(Q_c^2)와의 구분이 애매함이 발생할 수 있다[9]. 이에 비하여 SGML에서는 수식 각 요소를 엘리먼트명이나 엔티티명들으로 정의하게 함으로써 시스템에 독립적인 수식 표기가 가능하게 되며,

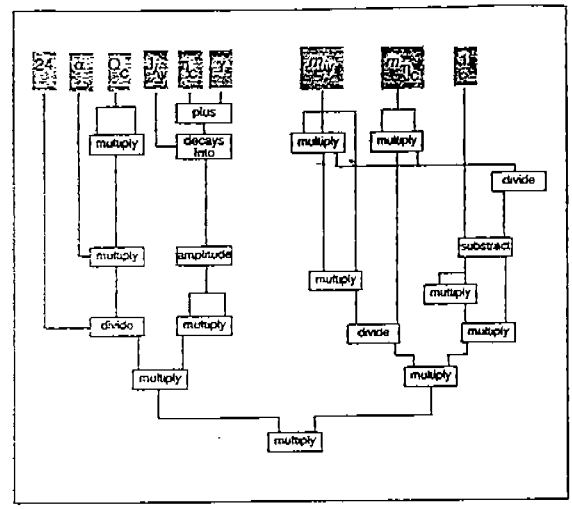
〈표 1〉 다양한 형태의 수식 표기법
〈Table 1〉 A mathematical formula marked up in various formatting

System	Coded Formula
Formatted Formula	$\Gamma(J/\psi \to \eta_c \gamma) = \frac{\alpha Q_c^2}{24} \left\| A(J/\psi \to \eta_c \gamma) \right\|^2 \frac{m_\psi^3}{m_{\eta_c}^2} \left(1 - \frac{m_{\eta_c}^2}{m_\psi^2} \right)^3$
TeX	$$\Gamma(J/\psi \rightarrow \eta_c \gamma) = {{ \alpha Q_c^2 } \over 24} \left\| A (J/\psi \rightarrow \eta_c \gamma) \right\|^2 {{ m_\psi^3 } \over { m_{\eta_c}^2}} \left (1 - {{ m_{ \eta_c }^ 2 } \over { m_\psi ^ 2 }} \right)^3$$
EQN (also SMFF)	Gamma(J/psi rarrow eta sub c gamma) = < alpha Q sup 2 sub c > over 24 left vbar A (J/psi rarrow eta sub c gamma) right vbar sup 2 < m sup 3 sub psi > over < m sup 2 sub < eta sub c > > left (1 - < m sup 2 sub < eta sub c > > over < m sup 2 sub psi > right) sup 3
SGML (ISO 9573)	<mfn> <fname> Γ <of>J/ψ → η _c γ = <frac> α <square> Q _c </square> <over> 24</frac> <square> <fence type=bar> <mfn> <fname> A <of> J/ψ → η _c γ </mfn> </fence> </square> <frac> <power> <degree> 3 <of> m _ψ </power> <over> <square> m _{η _c} </square> </frac> <power> <degree> 3 <of> <fence> 1 - <frac> <square> m _{η _c} </square> <over> <square> m _ψ </square> </frac> </fence> </power> </mfn>
SGML (AAP Maths)	<g>G</g>(<fr sol>J</><g>y</g> &ar; <g>h</g> <inf>c</inf>) = <fr><g>a</g>Q ² <inf>c</inf> </> 24 </fr> <fen>A(<fr>J</><g>y</g></fr> &ar; <g>h</g> <inf>c</inf>) <rp> </fen> ² <fr>m <inf><g>y</g></inf> ³ </> m <inf><g>h</g> <inf>c</inf></inf> ² </fr> <fen par> 1 - <fr> m <inf><g>h</g> <inf>c</inf></inf> ² </> m <inf<g>y</g>/ ² <rp par</fen> ³

엘리먼트에 속성을 부여하여 포맷팅 뿐만 아니라 수식 정보의 검색등에도 이용 가능하다.

〈표 1〉에서의 SGML은 ISO에서 정의한 수식의 DTD와 미국 출판 협회(American Association of Publishing: 이후 AAP라 칭함)가 정의한 DTD에 각기 다른 SGML 표기법을 표시하고 있다. 이 두 표기의 근간이 되는 DTD는 엘리먼트들의 계층적 구조로써 구성이 되어 있어 〈표 1〉에서 나타낸 수식 예는 (그림 1)과 같은 트리 구조로 형성될 수 있음을 알 수 있다[9].

다시 Q^2_c의 예를 보면 ISO의 SGML DTD에 따르는 수식 표기법이 AAP의 DTD보다는 의미적으로 보다 정확한 특성을 가지며 감마나 아이타 같은 표기도 ISO의 경우는 국제적으로 공통 표준인 엔티티명을 채택한 반면 AAP의 경우는 그리스어라는 엘리먼트명, 즉 〈g〉로 표현하는 등이 상이점이 있다. 이러한 상이점을 직접 조작 화면 방식이 아닌 수식 편집기에



(그림 1) 〈표 1〉의 수식의 트리 구조
(Fig. 1) The tree structure of 〈Table 1〉

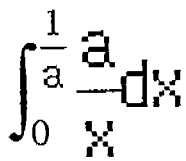
서 작업할 때는 매우 중요한 차이를 갖는다.

이와 같은 이유로써 본 연구에서는 ISO에서 제시한 수식 DTD을 따르는 수식 표기법을 채택하였다[9].

2.2 DTD 및 DI(Document Instance)

수식 정보를 SGML화하기 위해서는 수식의 논리 구조를 정의하고 있는 DTD와 그 DTD에 따라 작성된 문서실례 즉 DI가 필요하다. 따라서 DTD는 해당 문서의 논리 계층 구조를 명시하는 필수적인 부분이며 일반적으로 i)엔티티 선언(Entity Declaration), ii)엘리먼트 선언(Element Declaration), iii)속성 선언(Attribute Declaration)과 같은 3가지 주요 구성요소로 이루어진다[10, 11].

DI는 DTD에서 정의된 규칙에 따라 실제의 문서를 표기하는 것을 의미하며 수식의 작성 및 교환을 위하여서는 수식 DTD 및 수식 표기 부분이 동시에 요구된다. (그림 3)은 (그림 2)와 같은 수식을 표기하기 위한 SGML 표기법을 이용한 DTD 및 DI를 나타낸다.

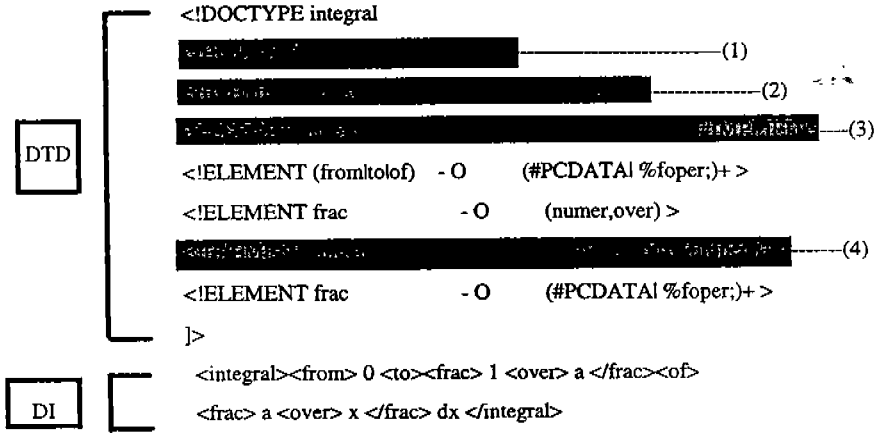


(그림 2) Integral 수식의 사례
(Fig. 2) Instance of Integral formula

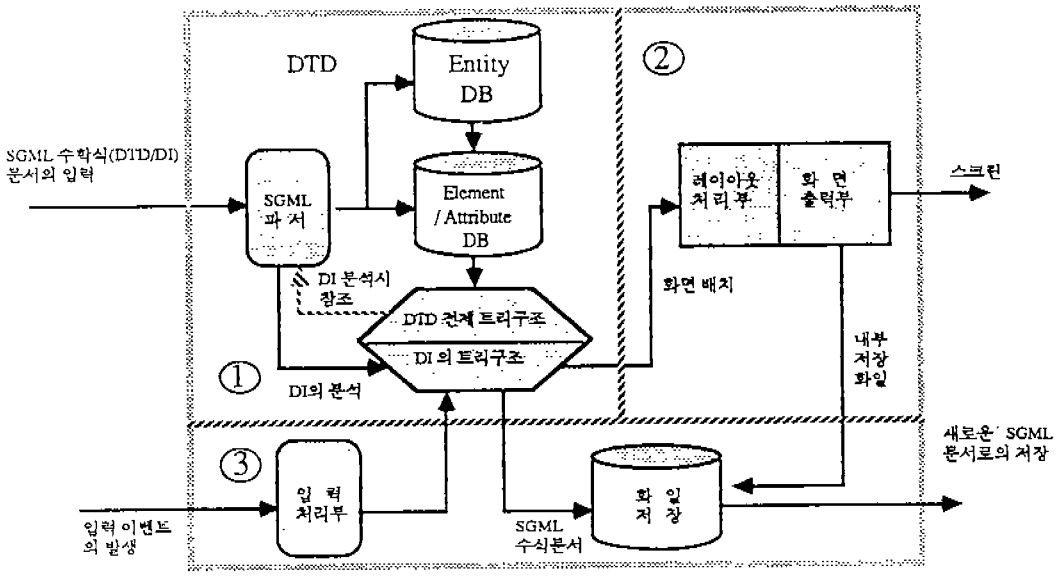
(그림 3)의 (1)에는 엔티티 선언이 명시되어 있고, 이 엔티티는 마치 매크로 선언과 동일한 역할을 한다. (그림 3)의 (2)는 엘리먼트 선언이며, integral 엘리먼트는 from과 to 그리고 of와 같은 하위 엘리먼트를 가지며, 하위 엘리먼트의 출현 순서는 연결자 ","에 따라 순차적 순위임을 나타내고 있다. 또한 (그림 3)의 (3)은 integral 엘리먼트에 대한 속성을 지정하는 속성 선언을 나타내고 있다. 즉 integral 엘리먼트의 속성 id를 선언하고 있으며, 이 id속성의 값은 시스템에서 제공함을 나타내고 있다[3, 6]. 그리고 (그림 3)의 (4) 엘리먼트 선언에서의 (#PCDATA|%foper;)+는 하위 엘리먼트들이 나타나는 순서가 #PCDATA(문자의 연속 발생 가능 표시)와 엔티티 %foper;에 선언된 엘리먼트중 하나가 선택되고("|" 기호:OR), 한번 이상 반복적으로 발생할 수 있음을 발생지시자인 "+"로 표기하고 있다.

3. 수식 편집기 시스템 설계

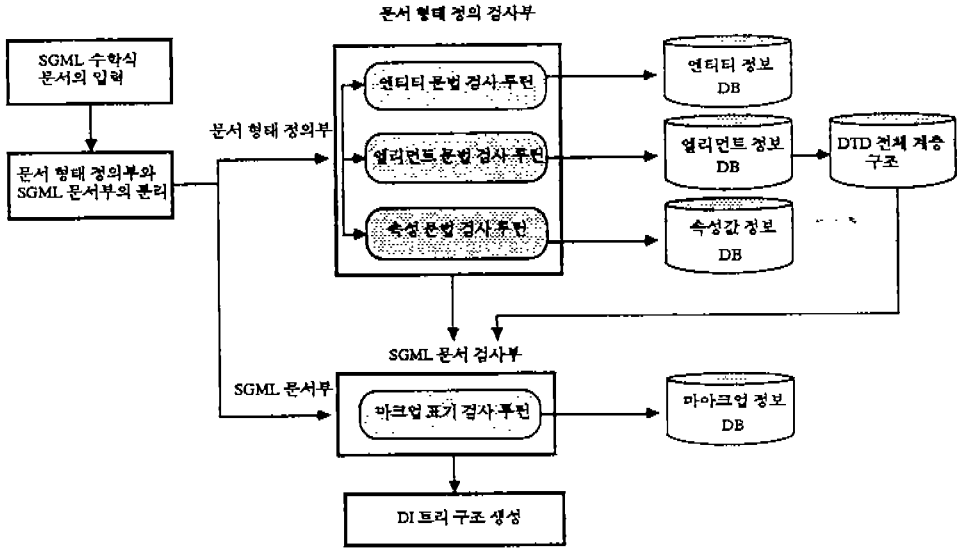
SGML 표기법의 수식을 자동 발생시키는 수식 편집기의 구조는 (그림 4)와 같으며, 세 개의 처리부로 구성되어 있다. (그림 4)의 ①은 내부 데이터 처리부로써, DTD 단독 또는 DTD와 DI를 입력받아 DTD 및 DI의 내부 데이터베이스로 변환하며, (그림 4)의 ②는 DI 트리의 수식 오브젝트에 대한 레이아웃 처



(그림 3) SGML 수식 표기법의 사례에 대한 DTD & DI
(Fig. 3) DTD & DI for an Math formula



(그림 4) 수식 편집기의 구성도
(Fig. 4) An Diagram of Math system



(그림. 5) SGML 파서를 통한 DTD 및 DI의 처리
(Fig. 5) Parsing of DTD & DI

리부이고, (그림 4)의 ③은 화면 직접 조작에 의한 수식의 입력을 받는 사용자 접속 처리부이다.

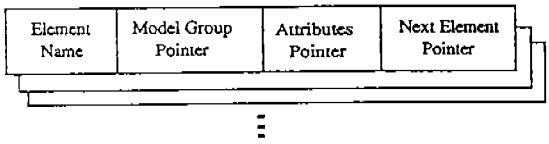
3.1 내부 데이터 처리

내부 데이터 처리는 SGML 문서를 입력받아 SGML 파서를 이용하여 처리하는데, 이 처리부는 SGML문서를 DTD와 DI로 분리하여, DTD 처리부와 DI 처리부에서 각각 처리한다. DTD 처리부에서는 i) SGML 문서의 DTD에 대한 구문 검사 ii)엔티티, 엘리먼트, 속성에 대한 데이터베이스 구성, iii)DTD 전체 계층 트리 구조 생성한다. 또한 DI 처리부에서는 i)SGML 문서의 DI에 대한 구문 검사, ii)DI에 대한 트리 생성을 처리한다. (그림 5)에서 SGML 파서를 이용한 SGML문서 DTD/DI의 처리 과정을 보였다.

3.1.1 DTD 처리

SGML 수식 문서 DTD의 구문 검사는 ISO 8879 (SGML)에 명시된 SGML 문서 표현 구문에 의거하여 SGML 파서 내의 구문 검사 루틴을 거쳐서 이루어진다[11,12]. 그리고 DTD의 각 엔티티 선언 및 엘리먼트선언, 속성선언을 분석함으로써 (그림 5)의 각 데이터베이스를 생성한다. (그림 6)은 한 엘리먼트 선언과 그 속성선언에 대해 생성된 데이터 구조를 나타내고 있다.

(그림 6)과 같이 생성된 엘리먼트 데이터 구조들은 DI 처리 시에 DI에 표현된 각 엘리먼트들(태그)이 맞게 구성되어 있는지를 검증하기 위해, (그림 7)와 같



* Element Name : 엘리먼트의 이름
* Model Group Pointer : 해당 엘리먼트와 자손 엘리먼트들 간의 계층 구조에 관한 정보를 참조할 수 있는 포인터
* Attributes Pointer : 해당 엘리먼트가 갖는 속성값들에 대한 포인터
* Next Element Pointer : 다음 엘리먼트의 주소를 갖기 위한 포인터

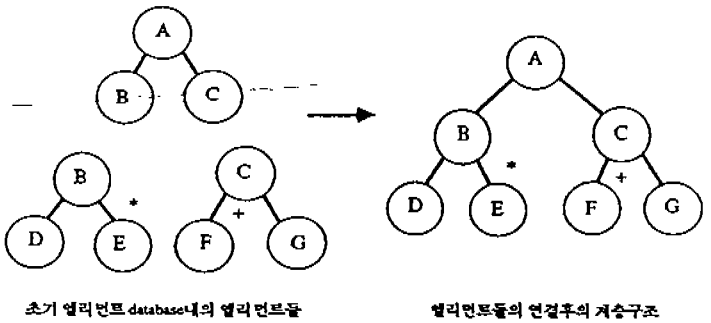
(그림 6) 처리된 엘리먼트 데이터의 구성
(Fig. 6) Data structure of element

이 (그림 6)의 각 엘리먼트 구조의 ModelGroup 내용에 의해 상호 연결하여, 수식 DTD의 전체 계층 트리 구조를 생성한다.

3.1.2 DI 처리

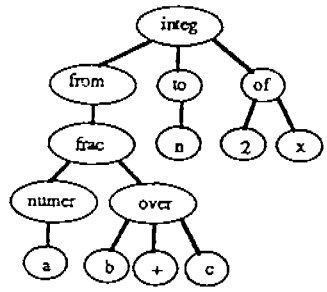
SGML 문서의 DI에 대한 처리는 DI의 구문을 IS 8879의 DI의 구문에 의거한 검사 루틴에서 검증하며, DI 트리 구조는 DTD 처리에서 생성된 DTD 전체 계층 트리 구조와 DI에서 나타나는 엘리먼트들의 계층 구조가 서로 일치하는 경우에만 생성된다. DI에 대한 각 엘리먼트들의 계층 구조는 스택 구조(stack structure)를 사용함으로써 처리된다. (그림 8)에 한 DI를 구성한 엘리먼트들과 그에 따른 트리 구조 형성에 대한 예를 보였다.

그리고 DI 트리를 구성하는 한 노드는 포맷팅 정보



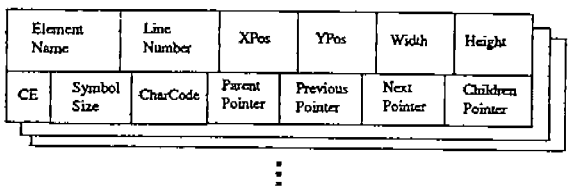
(그림 7) SGML 문서 DTD의 전체 계층 트리 구조 형성
(Fig. 7) Generating tree structure of DTD

```
<integral>
    <from>
        <frac>
            <numer>a
            </numer>
            <over>b+c
            </over>
        </frac>
    </from>
    <to>n</to>
    <of>2x</of>
</integral>
```



(그림 8) DI와 트리 구조의 예
(Fig. 8) Example of DI and Tree structure

를 컴가시킨 SGML 엘리먼트 단위로 구성된다. (그림 9)은 DI 트리의 한 노드에 대한 데이터 구조를 보였다.



(그림 9) 수식 엘리먼트의 데이터 구조
(Fig. 9) Data structure of an Math formula

위의 필드(field)들은 크게 수식 오브젝트를 화면에 출력할 때의 포맷팅 정보와 트리 구조 운영에 필요한 다른 엘리먼트에 대한 포인터로 나눌 수 있다. CE 필드는 해당 엘리먼트가 자손 엘리먼트를 갖는지를 나타내는 플래그(flag)이다. CharCode 필드는 해당 수식 오브젝트의 심벌을 화면에 출력하기 위한 폰트 테이블 내에서의 심볼 코드 값이다.

3.2 레이 아웃 처리

수식 DI 트리의 오브젝트들을 화면에 출력하기 위한 레이아웃 처리 과정은 크게 다음과 같이 2가지로 분류하여 처리된다.

1) DI 트리 구조의 갱신후의 레이아웃 처리(하향접근방식)

2) 특수한 수식 오브젝트의 레이아웃 처리(상향접근방식)

DI 트리 구조의 갱신 후의 레이아웃 처리는 DI 트리의 최상위 노드에서부터 자손 엘리먼트가 없을 때까지 수식 엘리먼트 노드들의 포맷팅 정보들을 부모 엘리먼트의 오브젝트 종류에 따라 재구성함으로써 하향접근방식만으로도 처리가 가능하다.

그러나 특수한 수식 오브젝트의 레이아웃 처리는 자손 엘리먼트의 입력 형태에 따라 자신의 수식 심볼 위치 및 폭, 넓이 등이 상대적으로 변하는 것들에 대한 처리가 필요하다. 이러한 수식 오브젝트에는 평방근(square root), 분수, 행렬 등이 있으며, 이 처리는 상향접근방식처리를 우선으로 해서 최상위 노드까지 전파하여 다시 하향접근방식에 의해 포맷처리된다.

본 수식 편집기의 화면 처리는 레이아웃 처리가 끝난 트리의 최상단 노드에서부터 전체 DI 트리의 노드들을 방문하면서 각각의 노드의 포맷팅 정보에 따라 해당되는 수식 폰트를 화면에 출력함으로써 이루어진다.

3.3 사용자 접속

본 수식 편집 시스템은 복잡한 수식을 작성하는데 대화형(interactive) 인터페이스를 제공하여 사용자가 심볼 위주의 버튼을 선택하여 수식을 수정하도록 지원하였다. 또한 (그림 10)에서와 같이 WYSWYG방식을 채택하여 화면에서 표현된 수식 모양을 똑같이 출력할 수 있고[10], 또한 섬세한 수식 폰트를 지원하기

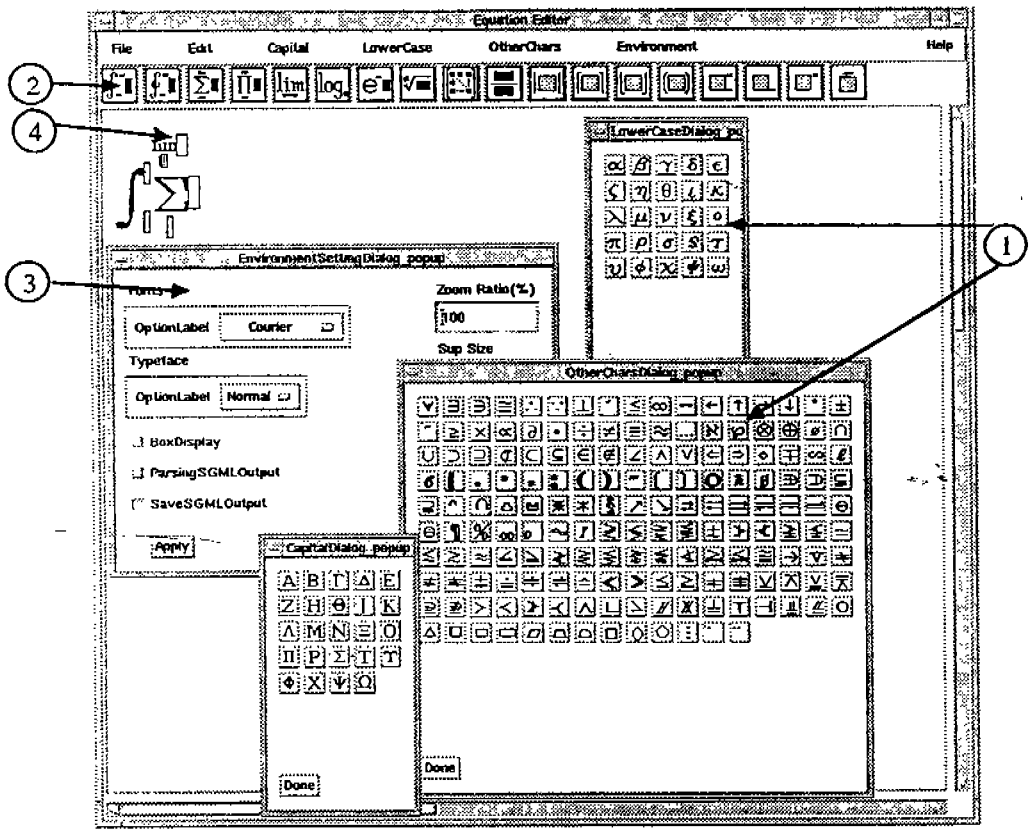
위해 type-1 폰트를 지원했으며, 키보드를 통해 입력이 불가능한 그리스문자,특수 문자들을 지원하기 위해 별도의 키보드자판메뉴를 지원하여 사용자가 쉽게 원하는 문자를 입력할 수 있게 하였다((그림 10)의 ①). 그리고 수식의 오브젝트(integral, fraction, sigma etc)들은 그 기능별로 분류를 하여 버튼화시켜, 사용자의 목적에 따라 선택하여 쓸 수 있는 방식을 채용하고 있다((그림 10)의 ②). ((그림 10)의 ③)은 화면 출력되는 수식에 대한 폰트 체, 폰트 크기 및 화면의 출력 배율 등의 속성을 지정하는 메뉴 판이다.

사용자 접속 처리부를 통한 수식의 입력은 내부 데이터 처리부에서 생성된 전체 계층 트리 구조에 의해 검증이 된 후, DI 트리에 삽입 및 삭제, 변경이 된다. 또한 한 수식 오브젝트가 선택되면 자손 오브젝트의 종류와 배치 특성에 맞게 입력 영역을 사각형 영역으로 표시하고 입력 커서를 표시한다((그림 10)의 ④). 본 편집기에서는 사각형 영역들의 입력 순서에 제한을 두지 않음으로써 수정 및 입력을 용이하게 하였다.

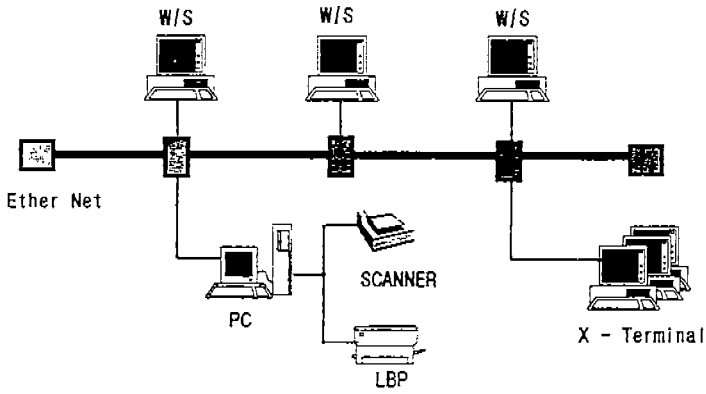
4. 구현 및 고찰

4.1 구 현

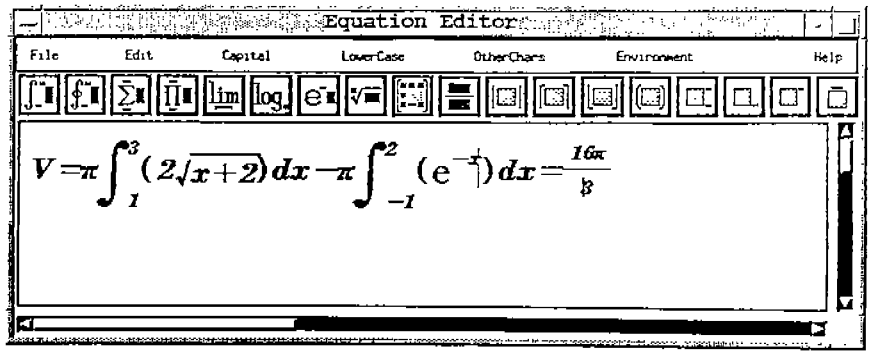
설계 및 구현은 SUN O.S 4.1.1, X 윈도우 시스템, Motif, OSF UIL을 탑재한 SUN SPARC를 호스트로 하고 NCD 16 터미널이 Ethernet으로 연결된 환경하(그림 11)에서 수행하였다. 본 수식 편집기에서는 ISO 9573 수식 트리 표현 구조를 갖는 DTD을 표준으로 따르고 있고, 이 DTD에 따라 원하는 수식을 작성할 수 있다. (그림 12)에서는 이 DTD을 이용하여 작성한 수학식의 예와 수정 후의 예를 나타내었다.



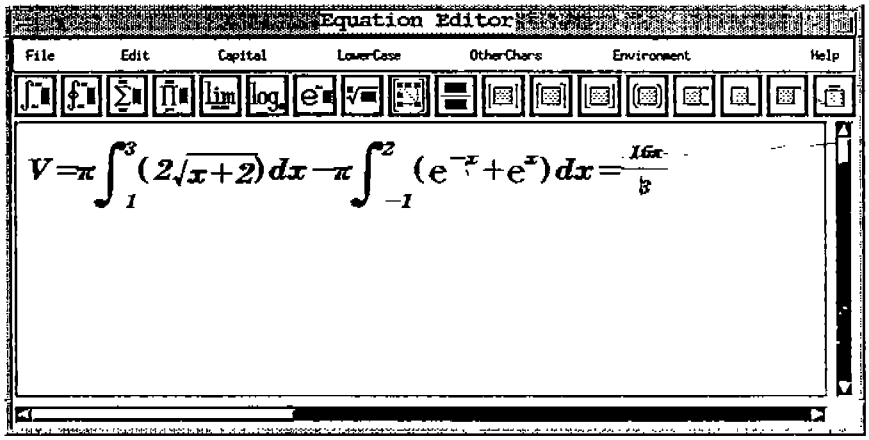
(그림 10) 수식 편집 시스템의 사용자 접속
(Fig. 10) User Interface on Math Author system



(그림 11) 구현 시스템 환경
(Fig. 11) The environment of the system to development

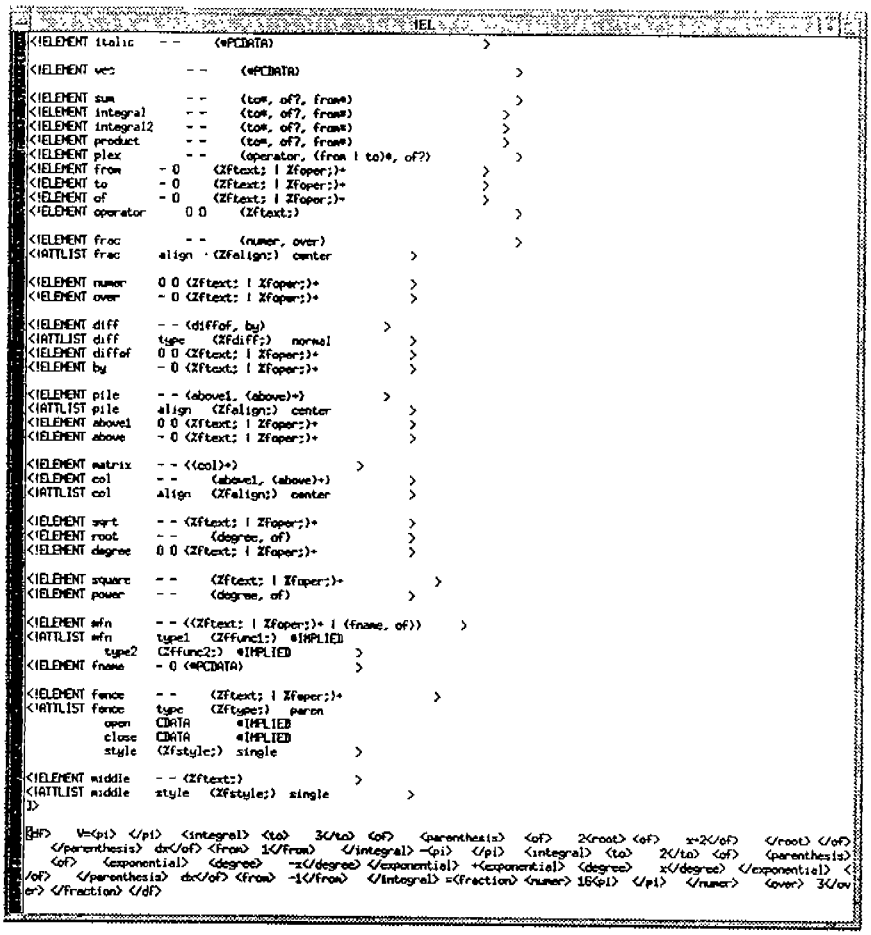


(a)



(b)

(그림 12) 수식 편집의 예(a)와 수정 후의 예(b)
(Fig. 12) Example of Editing(a) and result after editing(b)



(그림 13) (그림 12)에 대한 SGML 문서 저장 결과
(Fig. 13) SGML DI for (Fig. 12)

(그림 13)은 (그림 12)에서 작성된 수식을 저장하였을 때 수식 SGML표기법의 결과를 나타낸다.

본 수식 편집기는 SGML 응용 시스템으로 SGML 파서를 갖고 있는데, 본 파서의 기능은 〈표 2〉에서 나타낸 것과 같이 ISO 8879에서 제시하고 있는 SGML 특수 기능을 처리할 수 없는 파서로써 단지 정식으로 표현된 SGML 구문만을 처리할 수 있는 파서이다. 그 기능에 대한 비교를 〈표 2〉에 나타나 있다.

〈표 2〉 파서 기능 비교
〈Table 2〉 Compare Features of parsers

구분	SGML 기본파서	SGML 확장파서	본 수식시스템의 구현파서
SGML 구문검사	YES	YES	YES
축약기능			
DATATAG	NO	YES	NO
OMITTAG	YES	YES	NO
RANK	NO	YES	NO
SHORTTAG	YES	YES	NO
링크	NO	NO	NO
기타	NO	NO	NO

5. 결 론

본 논문은 수식을 표현하는데 특정 시스템에 의존

적인 형태를 벗어나 수식 데이터의 상호교환을 할 수 있는 표기법을 국제표준인 SGML을 이용하여 수식을 표현할 수 있는 수식 편집기를 구현하였다.

본 SGML 수식 편집기에 내장된 파서는 완전한 SGML 파서가 아니라 프로토타입 수준의 파서이므로 SGML의 여러 가지 기능들을 제공하기 위해서는 완전한 기능을 갖는 파서의 개발이 필요하다. 또한 문서 시스템들의 효과적인 문서 전송을 위해 SGML에 대한 활발한 연구가 필요하다고 생각되며 SGML 관련 응용(SGML Application)에 관한 연구가 계속되어야 할 것이다.

참 고 문 헌

[1] Joseph F. Ossanna, "NROFF/TROFF User's Manual", Bell lab., 1990.

[2] Donald E. Knuth, "TEX and Metafont", EY-BX003-DP-001, 1990.

[3] Technical Report-Techniques for using SGML, ISO/IEC/TR 9573:1988(E), 1988.

[4] Draft International Standard-Electronic manuscript preparation and markup, ISO/DIS 12083, 1989.

[5] Martin Bryan, "An Author's Guide to the Standard Generalized Markup Language", 1988.

[6] International Standard-Standard Generalized Markup Language(SGML), ISO 8879-1986(E), 1986.

[7] International Standard-Office Document Architecture(ODA), ISO 8613, 1988.

[8] 유경택, "대화형 수학식 작성 에디터의 구현", 광운대학교 석사학위 논문, 1990.

[9] Eric Van Herwijnen, "Practical SGML", kluwer academic publishers, 1990.

[10] 공업 진흥청, "텍스트와 사무 시스템-SGML 언어의 표준화 연구", 1992.

[11] 현득창, 임광택, 이수연, "SGML Parser를 이용한 SGML 문서 에디터의 구현에 관한 연구", 한국정보 과학회 논문집 20(4), 1993.

[12] 홍온선, 정희경, 이수연, "SGML에 대한 기본 파서의 구현 1", 한국 통신 학회, 한국 통신 학회 논문집 16(11), 1992.

김 태 훈

1992년 광운대학교 공과대학 전자계산기공학과 졸업(공학사)

1994년 광운대학교 대학원 전자계산기공학과 졸업(공학석사)

1994년~현재 현대전자(주) 근무

관심분야:멀티미디어 처리 시스템, 화상처리

현 득 창

1990년 광운대학교 공과대학 전자계산기공학과 졸업(공학사)

1992년 광운대학교 대학원 전자계산기공학과 졸업(공학석사)

1992년~현재 광운대학교 대학원 컴퓨터공학과 박사과정

1995년~현재 현대미디어시스템 IETM 개발팀

관심분야:멀티/하이퍼미디어 문서처리, SGML, HyTime, HTML

이 수 연

1969년 광운대학교 전자통신공학과 졸업(공학사)

1977년 연세대학교 전자통신공학과 졸업(공학석사)

1983년 일본 교또대학교 정보공학과 졸업(공학박사)

1977년~현재 광운대학교 컴퓨터 공학과 교수

1994년~현재 광운대학교 신기술연구소 연구원

관심분야:하이퍼미디어 문서처리, 디지탈 전자도서관, SGML, HyTime, HTML