과학기술정보를 위한 특수문자 표기법 표준화에 관한 연구*

A Study on the Encoding Scheme Standard of Special Characters in Science & Technology Information

이 수 상**
Soo-Sang Lee

차 례

1. 서 론
2. 특수문자 표기사례
3. 특수문자 표준화의 현안분석
4. 특수문자 표준화 요건과 과제
5. 결 론
• 참고문헌

초 록

과학기술정보 서비스를 위한 메타데이터 작업에서 특수문자의 표기법은 항상 중요한 문제였다. 그러므로 본 연구에서는 특수문자의 표기법의 표준화를 위한 방안을 모색하는 작업을 하였다. 이를 위하여, KISTI의 YesKisti에 나타난 특수문자의 다양한 표기사례를 조사하고, 특수문자 표기문제를 표준화의 관점에서 관련된 주요한 현안과 표준동향을 검토하였다. 그런 다음 국내 환경에서, 과학기술정보 특수문자 표기의 표준화를 위한 요건과 과제를 도출하였다.

키 워 드

과학기술정보, 메타데이터, 특수문자, 표기법, ISO 8879 표준, LaTeX 형식

* 본 연구는 과학기술정보표준화위원회의 "특수문자 표기법" 표준안 개발 사업의 일환으로 작성된 것임
** 부산대학교 문헌정보학과 조교수
(Assistant Professor, Dept. of Library, Archive & Information Studies, Pusan Univ., sslee@pusan.ac.kr)
• 논문접수일자 : 2005년 8월 20일
• 게재확정일자 : 2005년 9월 12일

ABSTRACT

The encoding schemes of special characters in science & technology information services are always serious matters. Therefore, this paper draw up a general plan for encoding scheme of special characters related in some way. For this work, I have made an investigation into encoding cases of KISTI'S YesKisti service. And I have reviewed the important issues and standardization trends. Finally, this study proposes some requisites and problems for encoding scheme standard of special characters in science & technology information services.

KEYWORDS

Science & Technology Information, Metadata, Special Characters, Encoding Schemes, ISO 8879 Standard, LaTeX Format

1. 서 론

1.1 문제의 제기

각종 정보 시스템의 서비스나 정보유통에서, 특수문자의 표기법 문제는 오래전부터 주요한 논의대상이었다. 주로 메타데이터 영역이나 원문의 디지털 표현에서 문제가 되었다. 특히 과학기술 분야의 정보 시스템에서 표제, 저자, 초록, 요약 등의 메타데이터 정보를 표현하기 위하여 다양한 형태의 특수문자의 사용이 요구된다. 알파벳, 숫자, 기호 등과 같은 단일의 문자 형태나 화학식, 수식 등과 같은 복수의 문자들을 조합하는 표현식을 사용하여야 하기 때문이다. 물론, 다국어나 다언어의 문자사용도 가능해야 한다.

일반적으로 과학기술정보는 문헌정보와 사실정보로 구성된다고 한다(백두권 2005). 여기서 문헌정보는 기술논문, 기술세미나, 연구보고서, 특허, 의장, 저작권 등에 대한 정보를 말하며, 사실정보는 수치데이터, 도표, 소프트웨어, DB, 인력정보, 기자재정보, 규격정보 등을 의미한다. 또한 과학기술정보는 메타데이터 정보와 원문정보로 구분하여 여러 가지 필요한 접근을 한다. 원문정보는 아날로그 형태 또는 디지털 형태로 존재하며, 디지털 형태의 원문정보는 표현하는 포맷에 대한 고려가 필요하다.

여기서 관심을 갖는 사항은 온라인 검색 시스템에서 과학기술정보 서비스를 위한 메타데이터에서 발생하는 문제에 관한 것이다. 메타데이터를 입력하고 원하는 메타데이터 정보를 화면으로 출력하는 경우(주로 웹 기반의 화면으로 출력하는 경우), 원본의 정보를 그대로 유지한 채 입력하고 출력하는 문제이다. 흔히들

이를 표기법(encoding scheme) 문제라 한다.

입력표기(input encoding)든, 출력표기(display encoding)든 그것이 국내 컴퓨팅 환경에서는 영어나 한글로 된 언어문자와 아라비아 숫자 문자와 같은 일반적인 문자라면 문제가 되지 않는다. 그러나 그것이 그리스 문자, 특수기호, 수식이나 화학식과 같은 특수한 형태의 문자나 표현식이라면 고려해야 할 사항이 매우 많다. 입력의 형태나 방식도 다양하며, 입력된 그대로 출력표기가 된다는 보장도 없기 때문이다.

이 연구는 과학기술정보에 나타난 특수문자의 표기법 문제에 관한 것이다. 표기문제에서 보면, 두 가지 차원의 문제로 나누어볼 수 있다. 첫째, 메타데이터에서 특수문자를 입력하기 위한 표기문제, 둘째, 입력된 특수문자를 DB의 테이블로 저장하고, 프린터로 출력하며, 그리고 웹 화면에서 출력하는 등의 출력표기 문제가 바로 그것이다. 그리고 특수문자의 표기법 문제는 데이터의 호환성 문제와도 연관이 있다. 즉, 응용 프로그램 간 또는 시스템 간에 표기된 데이터를 교환(반입과 반출)할 경우, 표기된 특수문자가 원래의 모습을 유지할 수 있도록 해주어야 하기 때문이다.

1.2 연구내용 및 방법

이 연구의 목표는 과학기술정보에 대한 메타데이터에서 특수문자의 표기법에 관한 요건을 정리하고, 표준화 과제를 확인하며, 해결방향을 모색하는데 있다. 이를 위하여 국내 대표적인 과학기술정보 시스템인 KISTI의 YesKisti(KISTI 2005)를 대상으로 특수문자의 다양한 표기사례를 조사하고, 특성을 분석하는 작업을 간단히 시도하게 된다. 과학기술정보 영역에서 나타난 특수문자의 표기사례와 유형을 조사하기 위해서이다. 또한 특수문자 표기문제를 표준화의 관점에서 관련된 국내외 동향들을 조사분석하고, 주요한 현안들을 검토하는 작업도 한다. 그런 다음, 국내의 환경에서 요구되는, 과학기술정보 특수문자 표기의 표준화 요건과 핵심과제를 도출한다.

주요 연구방법은 사례조사와 문헌조사이다. 사례조사는 YesKisti를 대상으로 표기법 문제의 다양한 유형을 조사하며, 문헌조사는 특수문자 표기와 연관되는 국내외 표준화 사례와 동향, 고려사항들을 관련문헌을 중심으로 분석한다. 다만, 본 연구는 논의의 대상을 과학기술정보 영역에서 접근하고, 다국어와 다양한 언어의 표기문제는 제외하고, 특수문자 문제에만 한정한다.

2. 특수문자 표기사례

2.1 조사방법과 표기유형

국내 과학기술정보의 특수문자 표기사례를 살펴보기 위하여, 가장 대표적인 서비스라 할 수 있는 KISTI의 YesKisti를 대상으로 선정하였다. 사례조사 작업은 2005년 7월 중순부터 8

월 중순까지 약 1개월간 진행하였다. 조사단계는 3단계로 나누었다. 1단계에서는 다양한 유형의 특수문자가 포함될만한 주제영역을 선정하는 작업을 하였다. 2단계에서는 각 주제별 국내 및 국외 학술지를 선정하는 작업을 하였다. 그리고 3단계에서는 선정된 학술지를 대상으로 YesKisti를 통하여 학술지를 검색하고, 각 학술지의 권/호별로 수록논문 하나하나를 서지정보와 원문을 검토해 가면서 특수문자의 표기유형을 파악하였다.

조사한 주제는 수학, 물리학, 기계공학, 금속공학, 환경공학, 전기 · 전자공학, 고분자화학, 세라믹 · 시멘트 등이었다. 국내학술지는 하나의 주제 분야별로 여러 개의 학술지를 조사하였다. 해외학술지는 수학, 물리학, 화학 등의 주제가 포함되고, 국가별로 구분하여 Applied Thermal Engineering, European Polymer Journal, Journal of Number Theory, Surface Science, SYNLETT, Thin Solid Films, ZAAC의 7가지만을 선정하여 조

〈표 1〉 국내 과학기술정보의 주요한 특수문자 표기사례 유형

구 분	표기지침 및 방법	표기사례
아래첨자 (예: a_2)	$기본문자_첨자$	a_2
	$기본문자_{첨자}$	$a_{1,2}$
	기본문자~첨자	a~2
	기본문자/sub 첨자/	a/sub 2/
	기본문자〈sub〉첨자〈/sub〉	a〈sub〉2〈/sub〉
	일반자판의 숫자로 표기	CH4, η3
위첨자 (예: a^2)	$기본문자^첨자$	a^2
	$기본문자^{첨자}$	$a^{1,2}$
	기본문자/sup 첨자/	a/sup 2/
분수	표준문자 세트에서 선택	1/3
	일반자판으로 표기	1/3
기타 특수한 기호	비슷한 기호 사용	β를 b 또는 B로 표기
	무조건 생략	ρ-adic를 -adic로 표기
	LaTeX 형식 사용	δ는 \delta로 표기
	문자체명 사용	“(따옴표)는 „로 표기
	문자 코드 값 사용	–(하이픈)은 –로 표기 ε은 ε로 표기
문자식	$문자식$ 사용	$(SiC)_p$
	〈f〉문자식〈/f〉	〈f〉ε〈/f〉

사하였다. 이러한 과정을 거쳐서 특수문자 표기유형을 검토하고 파악하는데 도움이 되는 약 150건의 논문을 선정해 냈다.

또한 다양하게 나타나는 표기사례와 실제 원문에 나타나는 표기를 대조해보기 위하여 국내학술지의 경우는 YesKisti의 원문보기를 이용하였으며, 원문보기를 제공하지 않을 시에는 해당 학회 홈페이지를 이용하였다. 해외학술지의 경우는 주로 ScienceDirect를 이용하였으며 해당 학회 홈페이지도 이용하여 원문과 대조작업을 하였다. 그리고는 조사된 사례별로 오타나 유형파악이 어려운 사례는 분석대상에서 삭제하였다.

사례조사에서 파악된 특수문자의 주요한 표기유형은 〈표 1〉과 같다. 특정한 표기지침이나 방법의 판별이 곤란한 것은 유형분석에서 제외하였다.

2.2 표기사례의 주요 특성

사례조사에서 나타난 가장 큰 특징은 특수문자 표기에 다양한 유형이 존재한다는 점이다. 이를 설명하면 다음과 같다.

- 아래첨자의 경우는 두 가지 유형으로 구분된다. ① '_', '~', sub 등 다양한 부호나 태그를 사용하는 경우, ② 특정한 지침이 없이 숫자를 그대로 사용하는 경우이다.
- 위첨자의 경우도 아래첨자와 마찬가지로 다양한 표기형식이 존재한다.
- 분수표기는 모양이 같은 특정한 문자체가 있는 경우 그것을 사용하거나, '/'와 같은 부호를 사용하기도 한다.
- 특수문한 기호는 유사한 발음의 영어문자를 사용하거나, 아예 표기를 하지 않는 경우도 있다.

유형을 종합해 보면, LaTeX 형식, HTML 문자체 참조형식, 특수한 태그 사용 등의 표기방법을 사용하고 있다. 이처럼 다양한 유형이 발생하게 된 배경을 KISTI의 담당 연구원로부터 확인[1]한 바, YesKisti에 수록된 데이터에서 주요 표기방법은 다음의 세 가지 범주라고 한다. ① LaTeX 형식의 표기방법으로서, $로 시작해서 $로 끝나는 경우이다. ② '〈sub〉, 〈/sub〉, 〈sup〉, 〈/sup〉, /sub /, /sup /' 등과 같이 태그를 사용하는 경우로서, HTML 문서에서 보여질 수 있도록 하기 위하여 표기한 방식이다. ③ 기타 방식으로서, 입력자가 자기 나름대로 기술한 경우이다.

이전에는 태그를 사용하는 방식을 사용하다가, 2000년도부터 한국전산원의 지식정보자원관리사업을 시작할 때부터 LaTeX 형식의 표기방식을 사용하였다고 한다. 그 배경으로는, 국가연구보고서 DB를 구축함에 있어 요약문을 입력할 때 다양한 복잡한 수식이 많이 표시되어 있어서, 이전의 태그 방식의 표기법으로는 한계가 있다고 판단하여 LaTeX 형식으로 표

1) 연구원 면담은 2005년 7월 26일에 이루어졌다.

기하게 되었다는 것이다[2].

3. 특수문자 표준화의 현안분석

3.1 특수문자의 정의

과학기술정보 영역에서 특수문자는 메타데이터 정보의 표현에 사용되는 특수한 형태로 표기되는 문자를 말한다. 특수문자라는 말을 특수하지 않은 문자(일반문자)의 상대적인 개념으로 볼 경우, 일반문자와 특수문자의 유형과 특성을 구분하여야 한다. 그러나 일반문자와 특수문자의 경계를 명확하게 구분할 수 없다. '특수'의 범주를 정하기가 곤란하며, 그것이 문자의 사용환경이나 문자문화와도 연관이 있기 때문이다. 언어문자의 경우를 보더라도, 특수한 언어의 범주를 명확하게 규정하기 힘들다. 라틴어, 그리스어, 키릴 문자 등과 같은 언어형 특수문자(알파벳), 문장에 사용하는 부호, 도량형기호, 수학연산자 등과 같은 부호나 기호형태의 특수문자가 다양하기 때문이다. 사실 그러한 부호나 기호가 언어문자에서 파생된 경우가 많다.

그러나 컴퓨팅 환경에서 특수문자라 하면, 2벌식이나 3벌식과 같은 일반 키보드(자판)로 입력할 수 없어 별도의 입력방식이 요구되는 그런 문자로 이해하여 왔다고 보아도 된다. 그러한 특수한 방법을 이용하여 입력표기를 한다는 차원에서 편의상 사용했던 용어이다. 물론, 여기에는 단일의 문자뿐만 아니라 수식, 화학식 등과 같은 표현식도 포함하여, 보다 광의로 해석하는 경우가 된다.

일반적으로 과학기술정보에 사용되는 문자는 언어문자뿐만 아니라, 특수한 형태의 기호, 숫자, 언어문자 등의 문자체(character entity)와 화학식, 수식 등과 같이 각종 문자체를 조합하여 구성하는 문자식(character expression)의 형식을 사용한다. 그러므로 본 연구에서의 특수문자는 특수문자체와 문자식으로 구분하여 정의하기로 한다[3].

특수문자체는 특수한 형태의 문자체라는 의미이지만, 넓게 보면 언어문자, 숫자문자, 각종 조판기호나 수학기호 등과 같은 일반적인 형태의 문자체도 포함가능하다. 특정 언어의 알파벳이나 많이 사용하는 아라비아 숫자와 같은 언어표현을 위한 문자와는 구별을 하고자 하는 의도를 나타내는 것으로, 언어표현 문자로 표시가 되지 않는 과학기술용의 정보표현을 위한 숫자, 기호, 글꼴 등 특수한 형태의 문자체인

2) 다만, 이러한 방법이 KISTI 내의 모든 부서에 전달이 되지 않아서 각 부서별로 독자적인 표기방법을 지금까지 사용하여 왔으며, 초기에는 LaTeX 형식을 웹 상에서 보여줄 수 있는 방법이 없었으나, 최근에 MathML이 등장하면서부터 웹 상에서 LaTeX 형식으로 입력된 내용을 원래의 특수문자 형식으로 100% 변환하여 보여주는 것이 가능하게 되었다고 한다. 그러나 아직은 LaTeX 형식의 표기법이 KISTI의 공식적인 표준으로 정립이 되지 않은 상태라고 한다.

3) 흔히들 특수문자는 일반 키보드로 입력하지 못하는 특수문자 코드로 인식하지만, 과학기술 영역에서는 수식이나 화학식과 같은 표현식도 특수문자의 주요한 고려사항이었다. 그러므로 특수문자 코드만을 특수문자라 할 수 없었다. 따라서 본 논문에서는 특수문자 코드로서 문자는 HTML, XML, SGML에서 사용하는 개념인 문자엔티티(character entity) 즉, 문자객체를 줄여서 문자체라 하고, 표현식을 문자식이라 한 것이다.

것이다. 즉, 일반 키보드로 간편하게 입력이 안 되어 가상 키보드를 사용하여야 하는 문자군을 의미한다. 문자식은 문자체의 다양한 방식의 조합으로 구성되는 화학식, 분자식, 수식 등을 말한다. 문자체는 단일의 문자를 말하며, 문자식은 복수의 문자를 조합한 것이다.

그러므로 여기서는 특수문자체와 문자식의 표준화 문제, 표기 문제, 호환성 문제 등으로 구분하여 표기법 표준화와 관련된 현안을 살펴보고자 한다.

3.2 특수문자체의 표준화 문제

우선 특수문자체(special character entities)의 의미를 정리하면, ① 일반 키보드에 있는 자판으로 쉽게 입력할 수 없는 문자체, ② 국내 컴퓨팅 환경에서 본다면, 아스키(ASCII) 문자나 한글의 자모문자, 그리고 일반 키보드로 입력이 가능한 기호를 제외한 문자군[4]을 말한다. 특수문자체의 범주와 유형은 특정한 표준에 따라 달리 정해지기도 한다. 또한 입력기(editor)에 따라 사용하는 유형과 범주가 다를 수 있다.

특수문자체의 표준화는 현재 어떻게 이루어지고 있는가? 두 가지로 나누어 생각할 수 있다. 첫째, 일반적인 표준문자 세트의 일부분으로 정의되는 경우이다. ASCII 표준, KSC 5601 표준 등과 같이 특정한 언어를 위한 것이거나, 다국어 또는 다문자를 위한 언어용 표준 문자 세트에서 특수문자체로 정의된 경우에 해당된다. ASCII 표준, KSC 5601 표준은 우리나라의 컴퓨팅 환경에서 가장 많이 사용하는 문자표준인데, 여기서 정의하는 특수문자가 과학기술정보를 표현하는 데 필요한 다양한 문자체를 포괄하는냐 하는 판단이 중요하다. 전체 128자의 문자체를 정의하고 있는 ASCII 표준으로는 포괄성이 너무 낮다. KSC 5601 표준도 약 600개 정도의 특수문자를 정의하고 있지만, 이 중에서 과학기술정보의 표현에 필요한 특수문자는 그다지 많지 않은 편이다.

둘째, 특수문자체를 많이 포함하고 있는 표준으로서 ISO 8859 표준과 ISO 8879 표준 등이 있다. ISO 8859 표준은 과학기술용 특수문자보다는 유럽의 다양한 언어문자들을 많이 포함하고 있는 언어문자체 중심의 표준이다. 반면, ISO 8879 표준은 약 1000개 정도의 문자 세트를 정의하고 있으며, 조판기호, 수학 및 과학용 기호, 라틴 계열의 알파벳 문자, 그리고 러시아와 그리스 문자 등과 같은 라틴 계열 이외의 언어문자들을 망라하고 있어, 다양한 범주와 유형의 특수문자를 포함하고 있는 대표적인 특수문자체의 표준이라고 할 수 있다[5].

4) 이 경우, 일반 키보드로 입력이 가능한 문자체 중 특수문자체로 정의된 것이 있을 수 있다. 만일 이러한 문자를 입력할 때, 어떤 입력값을 우선시 할 것인가 하는 판단의 문제가 생긴다. 그러나 대체적으로 동일한 속성이며, 코드 값이 같으므로, 일반 키보드를 통해 쉽게 입력되도록 한다.

5) 언어문자체나 특수문자체 모두를 포함하는 가장 포괄적인 문자 세트를 정의하고 있는 유니코드(Unicode)도 고려대상이긴 하지만, 현재는 과학기술용 정보표기를 위한 표준보다 참조용으로 더 많이 활용되고 있다.

문제는 해당 표준들이 과학기술분야의 여러 가지 형태의 정보를 표현할 수 있을만한 다양한 범주와 유형의 문자체를 정의하고 있는가 하는 점이다. 그러한 표준이라면, 표준 코드로서 선정하여도 무방하다. 주요한 요건은 과학기술정보의 표현에 적합한 문자체 즉, 망라성과 포괄성이 높은 그런 표준문자 세트이어야 한다. 그러한 것이라면, 새롭게 제정하거나 기존의 것을 그대로 활용하여도 무방하다.

3.2.1 ASCII 표준

ASCII(American Standard Code for Information Interchange) 표준은 1963년 미국표준협회(ASA)에 의해 결정된 미국 표준의 문자 코드로 128개의 문자를 제공하는 7비트 코드이다. ASCII는 본래 전신(teletype) 터미널용으로 설계되었기 때문에 처음 32개의 코드는 프린트되지 않는 특별한 제어 문자로 사용된다. ASCII는 영어 문자를 표기하는 데 필요한 문자와 소수의 특수문자로 구성되어 있어 세계 여러 나라에서 사용하는 모든 숫자, 국가언어, 기호 등을 충분히 표현할 수 없다. 그럼에도 불구하고 오늘날 사용되는 많은 문자 세트가 ASCII를 시작점으로 하여 확장된 문자의 집합을 만들고 있다.

3.2.2 KSC 5601 표준

초기 컴퓨터 보급 이후, 한글 코드 표준의 부재로 인한 데이터 손상이 빈번하게 발생하자 정부가 제정한 한글 코드 표준이 바로 KSC 5601-87이다. 당시 ASCII 코드로는 표현할 수 없었던 문자들에 대해 ISO에서는 ISO 2022를 표준으로 제정하여 2바이트 이상의 문자 코드를 사용할 때 지켜야 할 부호확장법을 정의했는데, KSC 5601 표준이 바로 ISO 2022 부호확장법에 따른 코드이다. ISO 2022를 따르고 있어 국외 네트워크나 소프트웨어 사용에 유리하다고 판단하여 채택되었지만, 2350자로 한글을 제한한 완성형 코드이기 때문에 표기할 수 있는 문자에 대한 제약이 크고, 특히 문서 입력에 있어서 제대로 입력되지 못하는 글자가 속출하는 등의 문제가 발생하였다. 이러한 문제를 해결하고자 1992년에는 완성/조합의 이원적인 코드 체계인 KSC 5601-92를 표준으로 지정하였다.

3.2.3 ISO 8859 표준

128자의 ASCII 문자들은 영어로 된 정보를 교환하는 데는 충분하지만, å(스위스와 그 외 노르딕 언어들)과 같은 로마자를 사용하는 대부분의 유럽 언어들은 ASCII 표준으로는 표현할 수 없는 추가적인 기호들이 필요하게 되었고, 그 결과 8비트 문자 세트를 구현하게 되었다. 국제 표준 기구(ISO)는 여러 가지 8비트 문자 세트를 16가지 범주로 구분하여 다양한 유형의 유럽 언어문자들을 정의하였고, 각 문자 세트는 ISO에서 발표한 문자 코드로 나타낸다.

- ISO/IEC 8859-1:1998(또는 Latin-1)

  가장 광범위하게 사용되는 ISO 8859 표준으로 서유럽 언어의 대부분의 문자와 기호를 포함한다.

- ISO 8859-2:1999(또는 Latin-2)
  중앙과 동유럽 언어의 문자를 포함한다.
- ISO 8859-3:1999(또는 Latin-3)
  터키, 몰타, 에스페란토 문자를 포함한다.
- ISO 8859-4:1998(또는 Latin-4)
  에스토니아, 라트비아, 리투아니아, 그린란드, 라플란드 문자를 포함한다.
- ISO 8859-5:1999 - 키릴 문자를 포함한다.
- ISO 8859-6:1999 - 아랍 문자를 포함한다.
- ISO 8859-7:2003
  그리스 문자를 포함한다.
- ISO 8859-8:1999
  히브리 문자를 포함한다.
- ISO 8859-9:1999(또는 Latin-5)
  ISO 8859-1과 거의 같다.
- ISO 8859-10:1998(또는 Latin-6)
  Latin-4 계열 문자를 재배열한 것이다.
- ISO 8859-11:2001 - 타이 문자를 포함한다.
- ISO 8859-13:1998(또는 Latin-7)
  발트 해 연안 문자를 포함한다.
- ISO 8859-14:1998(또는 Latin-8)
  켈트 문자를 포함한다.
- ISO 8859-15:1999(또는 Latin-9)
  거의 사용되지 않는 기호를 제외한 8859-1의 나머지 문자와 기호를 재구성한 것이다.
- ISO 8859-16:2001(또는 Latin-10)
  남동 유럽 언어를 포함한다.

3.2.4 ISO 8879 표준

ISO 8879:1986 - Standard Generalized Markup Language(SGML)은 다양한 형태의 전자문서들을 서로 다른 시스템들 사이에 정보 손상 없이 효율적으로 전송, 저장, 자동처리를 위한 국제표준화기구(ISO) 문서처리표준의 하나이다. IBM에서 수행된 범용 마크업 언어(GML, Generalized Markup Language) 프로젝트의 성공을 통해 미국립표준국(ANSI, American National Standards Institute)은 정보 처리를 위한 방법으로 GML에 기반하여 텍스트를 설명하는 언어의 표준을 개발하였는데, 이것이 바로 표준 범용 마크업 언어(SGML, Standard Generalized Markup Language)이다.

미출판인협회는 SGML을 출판용으로 활용했고, 국제표준화기구(ISO)는 SGML의 표준(ISO 8879:1986)을 승인하여 문서의 구조를 정의할 수 있는 메타 언어로서의 국제표준으로 공개했다. 그러나 SGML은 문법상의 많은 예외와 난해한 매개변수를 사용한다는 단점이 있어 널리 사용되지 못했다. 그러나 문서의 표현 특성보다는 구조에 중점을 두고 만들어졌고, SGML 컴파일러가 문서형태정의(DTD)를 통해 어떤 문서라도 해석할 수 있기 때문에 이식성이 좋다는 장점이 있어, 이를 근거로 HTML과 XML이 개발되었다.

SGML 문자체(character entity)로서 ISO 8879 표준은 크게 출판용 기호(General and publishing symbols), 수학 및 과학 기호(Mathematics and sciences), 라틴 문자(Latin based alphabets)와 라틴 계열 이외의

문자(Non-Latin based alphabets)로 범주화된 약 1000개의 기호와 문자를 포함한다. 문자체는 문자 번호나 약자를 사용하여 기호나 문자를 표기한 것으로, 일반적인 문법은 앰퍼샌드(&) 다음에 약자로 된 이름 또는 파운드(#) 기호와 숫자가 따라오고, 세미콜론(;)으로 끝난다. 예를 들어, 직각은 ∠ 또는 ∟ 로 표기할 수 있다. ISO 8879 표준의 문자체 범주는 다음과 같다.

- 일반 및 출판기호(General and publishing symbols)
  - 출판기호(Publishing)
  - 수와 특수 그래픽 기호(Numeric and special graphic)
  - 괘선과 선그리기 기호(Box and line drawing)
- 수학 및 과학영역 문자(Mathematics and sciences)
  - 일반기호(General technical)
  - 그리스 기호(Greek symbols)
  - 추가 그리스 기호(Alternative greek symbols)
  - 수학기호 확장 : 일반(Added Math symbols: Ordinary)
  - 수학기호 확장 : 이항 연산자(Added Math symbols: Binary operators)
  - 수학기호 확장 : 관계(Added Math symbols: Relations)
  - 수학기호 확장 : 부정의 관계(Added Math symbols: Negated relations)
  - 수학기호 확장 : 화살표 관계(Added Math symbols: Arrow relations)
  - 수학기호 확장 : 구분자(Added Math symbols: Delimiters)
- 라틴 계열의 알파벳(Latin based alphabets)
  - 라틴어 확장-1(Added Latin 1)
  - 라틴어 확장-2(Added Latin 2)
  - 발음부호(Diacritical marks)
  - 라틴 계열 이외의 알파벳(Non-Latin based alphabets)
  - 러시아어 키릴 문자(Russian cyrillic)
  - 러시아어 이외의 키릴 문자(Non-Russian cyrillic)
  - 그리스 문자 추가-1(Greek letters)
  - 그리스 문자 추가-2(Monotoniko greek)

3.2.5 HTML 4.0 문자체 참조

HTML(HyperText Markup Language)은 웹에서 정보를 표현하는 웹 문서를 작성하는데 사용되는 언어이다. HTML은 2.0 버전, 3.2 버전에 이어 4.0이 1997년 12월에 개발되었다. 현재 많이 사용하는 HTML 4.01은 4.0판의 오류를 수정해서 1998년 4월에 만든 개정판이다. HTML 3.2와 비교한 HTML 4.01의 특징으로는 먼저 국제화(Internationalization)를 들 수 있다. 보다 강력해진 마크업(Markup) 기능을 갖게 되어 다양한 언어를 구현하고 전송하는데 용이하게 된 점이다. HTML 4.01은 3.2 버

전에 비해 인라인 프레임과 강화된 테이블 기능을 지원하고, 일반적인 미디어 오브젝트와 응용 프로그램을 복합적으로 삽입할 수 있는 방법을 제공한다. 또한 문서의 레이아웃은 기본적으로 스타일 시트를 통해서 이루어지게 되므로 웹 문서의 관리가 손쉬워졌고, DHTML을 가능케 한다. HTML 4.01에서는 스크립팅(Scripting) 기능을 통해 다이내믹한 페이지를 만들 수 있으며, 네트워크 응용 프로그램을 만들기 위한 수단으로도 사용할 수 있다.

HTML 4.0에서의 문자체 참조는 문자 세트에서 한 문자를 참조하는 SGML 구조를 따른다. HTML 4.0에서는 다음과 같은 문자체 참조를 지원한다.

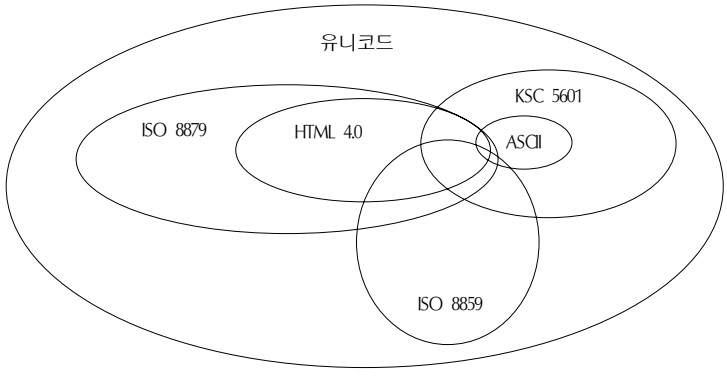
- ISO8859-1(Latin-1) : 라틴-1 계열 문자체에서 이미 보편화되어 사용되고 있는 ( ), ©(©)와 ®(®) 등을 포함하여 확장한 것이다.
- 기호, 수학기호, 그리스 문자 : 이 문자체 참조에는 그리스 글자, 괄호와 수학기호(계산, 결과, 합계 등)를 포함한 다양한 특수문자체를 지원해 준다. 이 문자체 세트는 현대 그리스어의 모든 글자를 포함하고 있다. 그러나 문서를 작성하는데 필요한 그리스 구두점, 액센트와 결합하여 강조하는 글자와 간격없는 액센트(tonos, dialytika) 등은 포함하고 있지 않다.
- 조판기호 및 국제적 통용문자 : 조판작성용(markup) 문자(", &, >, < 등)와 라틴확장, 일반 구두점 등의 기호들을 표현하고 있다.

HTML에서의 문자체 참조 방식은 HTML 2.0버전에서부터 사용하여 왔다. ISO 8879 표준문자체 중에서는 확장 라틴-1 계열 문자들을 포함하고 있다. 그래서 위에서 HTML 4.0에서 ISO8859-1에서 보편화되어 사용되는 기호들을 포함해서 확장했다고 표현한 것 같다. 수학식 기호는 HTML 4.0에서 추가된 것이고, 구두점 및 조판기호는 이전 버전에서도 지원하고 있다.

3.2.6 유니코드 표준

ISO 8859 표준의 등장으로 다양한 문자를 여러 사람이 사용할 수 있게 되었으나, 모든 사람이 서로 다른 문자 세트를 사용함으로써 호환성 문제에 직면하게 되었다. 즉, 사용자는 두 가지 서로 다른 문자 세트가 두 개의 다른 문자에 대해 같은 번호를 사용하거나, 같은 문자에 대해 다른 번호를 사용함으로써 데이터 손상의 위험을 겪게 된다.

이에 Apple, IBM, Microsoft 등이 컨소시엄을 통해 전 세계 문자 코드를 표현할 수 있는 문자 세트를 제공하였고, ISO/IEC JTC1에서 1995년 9월 국제표준으로 제정하였다. 공식명칭은 ISO/IEC 10646-1(Universal Multiple-Octet Coded Character Set)이다. 유니코드(Unicode)는 데이터의 원활한 교환을 위해 문자 1개에 부여되는 값을 16비트로 통일하였다. ISO/IEC 10646-1의 문자판에는 전 세계에서 사용하고 있는 26개 언어의 문자와 특수기호에 대해 일일이 코드 값을 부여하고 있는데, 2^{16} = 65,536개의 문자를 수용할 수 있다.



〈그림 1〉 주요 표준문자 세트의 관계

이상과 같이 설명된 표준문자 세트에서 문자체의 포함관계를 도식화하면 대략 〈그림 1〉과 같다.

3.3 문자식의 표준화 문제

문자식은 과학기술 영역의 정보를 표현하는데 사용되는 수식, 화학식과 같은 표현식으로, 표준화 차원에서 보면 우선 문자식의 유형을 파악하여야 그 가능성을 검토할 수 있을 것이다. 기본적으로 문자식은 표준문자 세트의 레퍼토리(repertories)에 포함되지 않는다. 또한 두 가지 이상의 문자체로 구성되며, 조합하는 방식이 다양하다는 특성을 가지고 있다.

정리하면, 문자체의 다양한 조합방식에 의해 구성되므로, 문자식 자체에 대해서는 표준화된 형상을 규정할 수 없다. 즉, 문자식 모양을 완성형으로 표준화 하는 것은 불가능하다. 단지 조합하는 방식을 표준적인 지침으로 정할 수는 있다. 문자식은 일반문자(언어, 숫자 등)와 특수문자 모두를 사용한다. 그러므로 입력표기에서 조합하는 방식을, 출력표기에서는 그것을 이용자가 가독할 수 있는 형태로 재구성하는 방식에 대한 지침이나 방법이 필요하다.

조합방식의 유형을 구분하면, 〈표 2〉의 사례

〈표 2〉 문자식의 조합방식

구 분	사 례
단순조합 방식	Na, Ca, $y=ax+2$, f(X) 등
복합조합 방식	$2^{3,}$ cm^2, $x_2^{y,}((x^2)^3)^{4,}$, H_2O, $\frac{1}{3}$, $\sum_{n=0}^{a}$, $y=\frac{x+4}{x^2+x+1}$, $y=\sqrt[n+1]{x}$, $y=\sqrt{x^2+\sqrt{x+12}}$ 등

처럼 문자체를 단순하게 나열한 단순형 조합과, 문자체의 조합에 첨자나 분수, 근호, 미적분, 분자식, 수학행렬 등이 사용되는 복합형 조합 방식으로 구분할 수 있다.

전자의 경우는 일반 키보드로도 조합이 가능하다. 입력을 위한 조합방식이 단순하므로, 출력표기를 위한 특별한 처리를 하지 않아도 되는 장점이 있다. 후자의 경우는 문자식의 표기방법을 공유하기 위해서는 표준적인 표기지침이나 규칙을 제정하여 준수하는 것이 바람직하다.

문자식에서 주의해야할 사항은 단위(예: ℃, cm^2)나 분수(예: $\frac{1}{4}$, $\frac{1}{2}$, $\frac{1}{3}$, $\frac{7}{8}$ 등)와 같이 조합으로 구성될 수도 있고, 표준으로 정한 특수문자 세트(예를 들어, ISO 8879 표준)에 포함된 문자체인 경우, 어떻게 할 것인가 하는 문제이다. 만일 그것이 문자식에 포함된 것이라면, 표준에 포함된 문자체보다 문자식 표기 지침을 준수하여 표기하는 것이 유리할 수 있다[6].

과학기술정보 영역에서 특수문자의 표기법 문제는 문자식에 관한 것이라 하여도 과언은 아니다. 특수문자체는 적절한 표준을 제정하여 사용하면 되는 문제이고, 문자식은 문자식의 표기(입력표기 및 출력표기)에 관한 지침이나 관련된 처리기술을 요구하고 있다.

3.4 입력표기 문제

입력표기(input encoding)는 사람이나 도구에 의해 특수문자체나 문자식을 컴퓨터의 입력화면(입력기)에서 표현하는 경우이다. 특수문자체나 문자식에 따라 입력표기의 방식은 다르다. 특수문자의 입력표기 문제는 단순히 입력만의 문제가 아니다. 어떻게 입력하느냐에 따라 출력되는 방식이 다르고, 입력된 데이터의 호환문제도 연관하여 생각해야 한다.

3.4.1 문자식의 입력표기

문자식의 입력은 문자체처럼 자판이나 입력메뉴를 사용할 만큼 단순하지 않다. 문자체의 조합이 다양하고 복잡하기 때문이다. 단순형 조합은 자판이나 입력 메뉴를 이용하여 해당되는 문자체를 입력하여 구성하면 된다. 그러나 복합형 조합은 표준화된 조합방식을 위하여 규칙이나 지침으로 정하거나, 특수한 방법을 사용할 수밖에 없다.

복합형 문자식의 입력방식은 다음 3가지로 구분이 가능하다. ① 규칙이나 지침을 제정하여 사용하는 경우이다. 예를 들어, KISTI에서 사용하고 있는 LaTeX 형식의 입력표기 지침을 말한다. ② XML로 표기하는 방식이 있다. 예를 들어, 각종 수학이나 화학, 물리 등에서 많이 사용하는 수식이나 화학식을 표기하기에 용이한 MathML을 사용하는 경우이다. ③ 문자식을 이미지 형태로 구성하여 표기하는 방식이다. 훈글과 같은 워드프로세서에서 사용하는 방식으로서, 메타데이터의 입력에서는 권장하

6) 이 문제는 몇 가지 고려해야 할 사항이 있으므로, 다음 기회에 별도로 다루고자 한다.

기 어려운 방식이다. 그러므로 여기서는 규칙이나 지침을 따르는 사례와 XML 방식에 대하여 간단히 살펴본다.

1) LaTeX 형식의 특수문자체 입력표기 지침[7]

- 특수문자체나 수식은 KSC5601 코드 표를 참고로 하여 입력하되, KSC5601 코드 표로 표현할 수 없는 특수문자나 수식은 LaTeX을 이용하여 처리해 준다.
- LaTeX 형식으로 입력을 시작할 경우에는 〈TEX〉와 〈/TEX〉라는 태그 명령을 사용하여, 시작과 끝을 표시해 준다.
- 명령어 모드는 '\' 표시 바로 다음에 명령어를 쓴다. 명령어의 유형은 분수, 근호 등 다양하며, 그 다음에 '[ ]'을 사용하여 선택항목들을 나열하고, '{ }' 안에 명령이 수행되는 대상을 쓰면 된다.
- 첨자표시의 경우 위첨자는 '^', 아래첨자는 '_' 를 사용한다.
- 명령어의 적용대상인 블록을 지정할 경우는 '{'와 '}'를 사용하여 시작과 끝을 표시한다.
- 수식 모드는 수식이 문장 내에 있을 경우 '$수식$' 로, 별도의 문장으로 있을 경우 '$$수식$$' 로 통일하여 사용한다.
- 문자식의 입력표기 사례는 다음 〈표 3〉과 같다.

2) XML방식

- XML을 사용하는 문자식의 표기방식으로는 MathML을 가장 많이 이용한다.
- MathML은 웹 상에서 수학식 표현의 효율적인 표기 및 저장, 공유를 가능하게 하기 위한 XML(eXtensible Markup Language) 기반의 수학식 표기 언어로 W3C(World Wide Web Consortium)에서 정의한 것이다. 텍스트 형태로 표시된다.
- MathML을 사용할 경우, 웹 브라우저(Internet Explorer)에서는 해당 문자식을 출력표기할 수 있도록 하는 플러그인(plug-in) 프로그램인 mathplayer를 필요로 한다[8].
- MathML의 사례는 〈그림 2〉와 같다.

〈표 3〉 문자식 입력표기 사례

구 분	출력 사례	입력 사례
위, 아래 첨자	x_2, x^2	〈TEX〉x_2〈/TEX〉, 〈TEX〉x^2〈/TEX〉
분수의 표시	$y=\frac{x+4}{x^2+x+1}$	〈TEX〉$y=\frac{x+4}{x^2+x+1}$〈/TEX〉
근호 표시	$y=\sqrt[n+1]{x}$	〈TEX〉$y=\sqrt[n+1]{x}$〈/TEX〉
연산자 표시	$3+4$, $3\div 4$	〈TEX〉$3+4$〈/TEX〉, 〈TEX〉$3\div4$〈/TEX〉
온도 표시	H_2O is $36^{\circ}C$	〈TEX〉$H_2O\;is\;36^{\circ}C$〈/TEX〉

7) 이 지침의 상세한 내용은 KISTI의 "LaTeX을 이용한 특수문자 및 수식 입력지침서"를 참조하면 된다.
8) 플러그인 프로그램 다운로드 사이트 〈http://www.dessci.com/en/products/mathplayer/download.htm〉.

```
〈math〉
 〈mstyle displaystyle="true" mathcolor="#0000ff" fontfamily="serif"〉
  〈mi〉N〈/mi〉
  〈msup〉
   〈mi〉a〈/mi〉
   〈mo〉+〈/mo〉
  〈/msup〉
  〈mo〉+〈/mo〉
 〈/mstyle〉
〈/math〉
```

```
〈math〉
 〈mstyle displaystyle="true" mathcolor="#0000ff" fontfamily="serif"〉
  〈mi〉C〈/mi〉
  〈msup〉
   〈mi〉a〈/mi〉
   〈mo〉+〈/mo〉
  〈/msup〉
  〈mo〉2〈/mn〉
 〈/mstyle〉
〈/math〉
```

〈그림 2〉 $Na^{+}/Ca^{+}2$의 두 가지 수식에 대한 MathML 표기사례

3.5 출력표기 문제

출력표기(display encoding)는 컴퓨터에 입력(저장)된 특수문자를 사람이나 기계가 인식할 수 있도록 출력하는 경우이다. 여기서는 입력표기된 메타데이터를 웹 상에서 표현하는 출력표기 문제에 한하여 논의하기로 한다. 웹 출력표기 문제를 논하기 위해서는 다음 세 가지 사항이 전제되어야 한다. 첫째, 특수문자의 웹 출력은 HTML 4.0 규격에서 정의된 문자를 기반으로 표기된다. 그러므로 HTML 4.0에서 수용가능한 문자 세트의 경우는 문제없이 표기가 가능하다고 할 수 있다. 둘째, HTML 4.0 기반의 웹 출력은 출력코자 하는 웹 문서의 기본 문자 세트는 설정해야 한다. 즉, 한글문자(KSC 5601)인 경우는 'euc-kr', 유니코드 문자인 경우는 'UTF-8'과 같이 설정되는 방식에 따라 특수문자의 출력표기는 달라질 수 있다. 마

지막으로, 출력표기 문제는 입력표기 방식에 영향을 받을 수밖에 없다. 어떻게 입력하였는가에 따라 출력표기 되는 방식이나 형태가 다르기 때문이다.

그러한 전제를 배경으로, 특수문자체의 출력표기는 어떻게 될까? 예를 들어, ISO 8879 표준에 기반한 특수문자체인 경우, 많은 문자가 HTML 4.0 환경에서 표기가 가능하다. 그것이 문자형상(glyph)으로 입력표기 되었든, 문자 코드 값이나 문자체명으로 입력표기 되었든 상관이 없다. 그러나 그것이 HTML 4.0 문제세트에 없는 유니코드 값으로 입력된 경우라면, 웹 상에서는 그냥 유니코드 값으로 표기될 수밖에 없다. 만일 웹의 기본문자 세트 설정이 유니코드로 되어 있다면, 해당되는 문자체의 출력표기는 가능할 것이다.

또한 문자 세트가 LaTeX 형식을 준수한 것이라면, 특별한 처리과정을 거쳐야만 출력표기가 될 것이다. 즉, LaTeX 형식의 텍스트를 XML(예: MathML)로 변환하고, 그것을 웹 상에서 표기되도록 하는 도구(예: mathplayer 프로그램)가 요구된다. 이 경우는 LaTeX 형식의 문자식 입력표기 지침의 처리과정과 동일하다. LaTeX 형식의 텍스트 문자식을 MathML로 변환시켜 주는 대표적인 프로그램으로 'ASCIIMathML.js'가 있으며[9], 〈그림 3〉과 같이 HTML 문서에 이 스크립터를 부르기만 하면 간단히 변환된다.

입력표기 형식이 XML인 경우도 마찬가지의 처리과정을 따른다. 〈그림 4〉에서처럼, 플러그인 프로그램을 설치한 경우는 텍스트로 출력표기 되고, 설치하지 않은 경우는 우측과 같

```
〈html xmlns:mml="http://www.w3.org/1998/math/mathML"〉
〈head〉
〈object id="mathplayer" classid="clsid:32F66A20-7614-11D4-BD11-00104BD3F987"〉
〈/object〉
〈?import namespace="mml" implementation="#mathplayer"?
〈script type="text/javascript"
src="ASCIIMathML.js"〉〈/script〉
〈title〉your title...〈/title〉
〈/head〉
〈body onload="translate( )"〉
```

〈그림 3〉 ASCIIMathML.js 스크립터를 호출하는 HTML 문서

9) 이 스크립터(script)는 GNU 공개 S/W로서 무료로 다운로드하여 사용이 가능하다. 다운로드 페이지는 〈http://www1.chapman.edu/~jipsen/mathml/asciimath.html〉이다.

〈그림 4〉 LaTeX 형식의 입력표기에 대한 출력표기사례

이 출력표기 된다. 출력표기 되는 텍스트 형태는 LaTeX 형식의 입력표기 지침에 따라 표기된 것을 처리 프로그램(mathplayer)을 통해 만들어진 것이다. 그러나 우측은 LaTeX 형식의 입력표기 지침 그대로 출력된 경우이다.

3.6 호환성 문제

입력표기와 출력표기는 특정한 시스템 내에서 해당되는 문제라면, 시스템과 시스템 간의 데이터호환에서도 특수문자 문제는 고려된다. 즉, 입력표기된 특수문자 데이터는 다른 시스템과 전송 또는 교환될 경우, 정보의 변경이나 손실이 없어야 한다는 것이다. 이 문제는 '특수문자의 호환성' 문제라 할 수 있다.

특수문자의 호환성 문제는 특수문자체와 문자식 모두에 해당되는 것으로서, 특수문자체는 데이터의 표기에 사용하는 표준문자 세트를 가능한 자동환경에서 인지할 수 있는 기법이 요구된다. 경우가 다를 수 있지만, 문자식은 해당 문자식을 표기하는데 적용된 표준지침을 어떤 형식으로든 명시하여 가독에 불편이 없도록 해야 한다.

특수문자의 호환에 있어 가장 중요한 사항은 국제적으로 통용되는 표준문자 세트나 표준지침을 사용하여야 한다는 점이다. 다양한 데이터베이스를 통합하여 서비스할 경우, 각각에서 정의된 특수문자의 호환을 위한 매핑 작업과 같은 장치를 마련하여야 한다.

4. 특수문자 표준화 요건과 과제

4.1 표준화 사례

과학기술정보 서비스에서 특수문자의 표준화 현황을 국내외 대표적인 사례를 통해 살펴

보면 다음과 같다.

4.1.1 NCBI의 PubMed 서비스

NCBI의 PubMed 서비스는 1997년부터 NLM(National Library of Medline)이 인터넷을 기반으로 하는 Web상에서 Medline을 무료로 검색할 수 있도록 개발한 서비스이다. 생화학 및 의학 분야의 자료들을 중심으로 전 세계의 주요 논문들을 검색할 수 있게 해주는 검색 시스템으로서 이외에도 독성학, 영양학, 약물학, 수의학, 정신의학, 의료공학, 병리학, 스포츠 의학 등에 대해서도 다루고 있다.

PubMed는 자료 구축시 웹 상에서 특수 문자 처리를 위해 ISO 8879 표준을 기반으로 한 SGML/XML 문자체를 사용한다. PubMed 서비스의 SGML/XML 문자체는 ISO 8859 표준과 동일하나, 서비스의 특성을 고려해서 자체적으로 문자체의 배열 순서와 범주를 정의하고 있다.

4.1.2 KAIST 과학도서관의 NDSL 서비스

NDSL은 국내 학술연구자들에게 해외 학술저널과 프로시딩의 원스톱 원문 서비스를 제공하는 국내 최대의 해외 학술정보 포털이다. NDSL에서 전 세계의 저널과 프로시딩을 통합적으로 검색하고 전자원문을 다운로드할 수 있으며, 전자저널로 이용할 수 없는 경우 국내외 협력기관을 통한 원문복사 서비스를 제공하고 있다.

NDSL의 통합DB(e-GateDB)는 자체 입력은 거의 없고 외부 메타데이터 공급업체나 출판사로부터 제공받아 구축하고 있다. 그러므로 특수문자 세트는 ISO 8879와 ISO 9573-13 표준[10]을 기본으로 채택하고 있으며, 수식은 ISO 문자 세트의 문자체와 HTML 태그를 조합하여 표현하고 있다. 그리고 첨자표현은 〈SUP〉와 〈/SUP〉, 〈SUB〉와 〈/SUB〉등의 태그를 해당 문자의 처음과 끝에 사용하는 방식을 채택하고 있다.

ISO 8879와 ISO 9573-13 표준 이외의 문자체로 입수되는 메타데이터는 매핑 테이블을 활용하여 NDSL에서 적용하는 ISO 문자 세트로 변환하여 사용한다. 이처럼 NDSL은 다양한 외부 DB를 통합해야 하므로, 특수문자 표기의 호환성 문제를 중요시 여기고 있다. 따라서 매핑이 안될 경우에 대비하여 다음과 같은 대응책을 사용하고 있다. ① ISO 문자 세트에서 표현하지 못하는 특수문자체가 있는 경우, NDSL에서 임의 생성했다는 의미의 "n."을 앞세워 각 출판사에서 정의한 명칭을 그대로 차용한다. ② 일부 출판사에서 보내오는 LaTeX 형식의 특수문자 표기는 그대로 ISO 문자 세트로 변환하지 않고 TeX 명령어인 〈TEX〉, 〈LATEX〉 태그를 사용하여 LaTeX 문자를 그

10) ISO/IEC TR 9573-13:1991 표준은 수학과 과학 분야의 저작물에서 많이 사용되는 특수한 그래픽 문자들을 위한 문자체 세트를 정의한 것이다.

대로 표기한다.

아울러, NDSL은 웹을 통해 서비스 되므로 ISO 8879 표준과 ISO 9573-13 표준 문자 세트를 적용하는 경우 웹 브라우저에서 별도의 처리 없이 서비스가 가능하다. 이 경우, 특정 문자체가 웹에서 표기가 안되는 문자가 발생하더라도, 특별한 처리를 하지 않는다는 말이다.

4.1.3 KISTI의 YesKisti 서비스

YesKisti는 국내·외 학술지, 학술회의자료, 국내학위논문, 특허자료, 연구보고서, 과학기술인력정보와 KISTI가 생산, 수집한 분석 및 동향정보 등에 대한 검색 및 원문제공 서비스이다. 직접 생산하는 자료나 국내자료의 경우는 자료의 목록과 검색서비스를 위해 특수문자를 국제표준인 ISO 8879 표준의 문자체에 기반을 두고 LaTeX 형식의 특수문자 입력 방식에 의한 직접입력을 실시하고 있다. 해외에서 들어오는 자료에 대해서는 해외 벤더들이 제공하는 목록 및 초록을 그대로 사용해서 서비스를 제공하고 있다.

4.2 표준화 요건

특수문자 표기법 표준화와 관련되는 여러 가지 현안분석에서 검토된 내용을 토대로 표준화 요건을 특수문자체와 문자식으로 나누고, 각각은 표준화 문제, 입력표기 문제, 출력표기 문제, 그리고 호환성 문제로 구분하여 정리하도록 한다.

4.2.1 특수문자체

표준화 문제에 관한 요건은 다음과 같다.

- 과학기술정보 메타데이터 표현에 적합성(망라성과 포괄성)이 높은 특수문자체의 범주와 유형을 가진 것이어야 한다.
- 정의된 범주화 유형은 표준특수문자 세트의 형식으로 결정되어야 한다.
- 표준특수문자 세트는 컴퓨팅 환경에서 사용하는 일반문자 세트 표준인 ASCII, KSC5601, HTML 4.0 문자체, 유니코드 등과의 연관을 고려하여야 한다.
- 입력표기와 출력표기가 용이하고, 외부와의 데이터 호환성이 높은 표준이 개발되어야 한다.

입력표기 문제에 관한 요건은 다음과 같다.

- 표준특수문자 세트는 컴퓨팅 환경에서 사용하는 일반문자 세트의 문자체와 구별하여 표기(태그를 사용)하도록 하여야 한다.
- 입력방식은 문자형상에 의한 입력, 문자 코드 값에 의한 입력, 문자체명에 의한 입력, 명령어 방식의 입력 등 다양한 방식 중에서 선택하도록 한다.
- 특수문자체의 문자형상 입력표기를 위해서는 가상 키보드가 제공되어야 한다.
- 문자 코드 값에 의한 입력과 문자체명에 의한 입력은 HTML 4.0 문자체 참조방식에 준하여 10진수나 16진수 표기가 허용되도록 한다.
- 명령어 방식의 입력은 LaTeX 형식과 같은 범용지침을 표준으로 사용하여야 한다.

출력표기 문제에 관한 요건은 다음과 같다.

- 사용하는 특수문자체는 HTML 4.0 문자체로 표기가 가능하여야 한다.
- HTML 4.0 문자체로 표기가 안되는 경우, 입력표기된 형태 그대로 표기될 수 있도록 하여야 한다.
- 유니코드 값으로 입력표기한 경우, 유니코드 환경설정으로 표기되어야 한다.
- LaTeX 형식과 같이 명령어 방식으로 입력표기한 경우, 그것이 웹 상에서 자동적으로 출력표기가 되도록 필요한 처리를 해주어야 한다.

호환성 문제에 관한 요건은 다음과 같다.

- 특수문자체의 표준은 국제적으로 통용되는 것이어야 한다.
- 특수문자체의 입력표기는 호환성이 높은 방식이 우선되어야 한다.

4.2.2 문자식

표준화 문제에 관한 요건은 다음과 같다.

- 문자식 자체의 형상(이미지)에 대해서는 표준화할 수 없다.
- 문자식을 구성하는 조합방식을 표현하는 표준지침이나 규칙을 제정하여 준수하도록 하여야 한다.
- 문자식 조합을 위한 표준지침은 다양한 유형의 문자식을 간편하게 조합할 수 있도록 하여야 한다.
- 표준지침은 문자식이 어떠한 컴퓨팅 환경에서도 용이하게 표기(입력 및 출력)되는 방법이 제시되어야 한다.

입력표기 문제에 관한 요건은 다음과 같다.

- 문자식 입력표기에 대한 표준화된 지침을 제공하고, 준수하여야 한다.
- 표준지침은 웹 상에서 원활하게 표기할 수 있는 방법이 함께 제시되어야 한다.

출력표기 문제에 관한 요건은 다음과 같다.

- 문자식에 대한 형상이 웹 상에서 출력표기가 되도록 해야 한다.
- LaTeX 형식으로 입력표기된 문자식인 경우, 그것의 웹 출력표기에 필요한 처리가 자동적으로 수행되도록 하여야 한다(예: LaTeX 형식을 MathML로 변환하고, MathML을 웹 상에서 실행하는 처리).

호환성 문제에 관한 요건은 다음과 같다.

- 문자식 조합을 위한 표준지침은 국제적으로 통용되는 지침이어야 한다.
- 문자식의 웹 상에서 출력표기는 국제적으로 통용될 수 있는 방법이어야 한다.

4.3 표준화 과제

표준화 요건의 핵심을 정리하면, 표준화 과제는 다음 세 가지로 정리된다. 첫째, 표준 특수문자 세트의 제정, 둘째, 문자식의 입력표기 표준지침 제정, 그리고 셋째, 웹 출력표기의 자동처리 방안 제시이다. 전자의 두 가지 과제는 표준화 대상이며, 세 번째 과제는 표준화의 원활한 적용을 위한 환경조성 작업이라 할 수 있다. 그러므로 여기서는 두 가지 표준화 과제에

대한 접근전략을 간략하게 정리하도록 한다.

- 표준 특수문자 세트의 제정 : 특수문자에 대한 표준문자 세트의 개발
  - 관련 표준화 현황 분석
  - 적합성이 높은 문자 세트 개발
  - 일반 문자 세트와 연관성 검토
  - 입출력 표기의 용이성 확보
  - 국제적 차원의 특수문자 호환성 확보
- 문자식 표준 입력표기 지침 제정 : 수식 및 화학식과 같은 문자식에 대한 표준화된 입력표기 지침개발
  - 관련 표기지침 사례조사 및 분석
  - 사용 편이성이 높은 지침개발
  - 웹 출력표기와 연관성 검토
  - 국제적 차원의 문자식 호환성 확보

5. 결 론

지금까지 특수문자의 표기법 문제를 표준화의 입장에서 현안들을 정리하고, 국내외 관련 표준화 사례들을 조사하며, 요건과 과제들을 점검해 보았다. 표준을 개발한다는 것은 쉬운 문제가 아니다. 현재까지 검토한 바로는, 특수문자의 표준화는 특수문자체와 문자식으로 나누어서 접근하여야 한다는 점이다. 또한 국내외 주요 과학기술정보 서비스 사례에서 검토하였듯이, 특수문자체는 ISO 8879 표준을, 그리고 문자식은 LaTeX 형식의 표기법을 기본으로 채택하는 것이 가장 적합하다고 판단된다. 즉, 국내의 과학기술정보의 특수문자 표기법은 특수문자체 영역에서는 ISO 8879 표준, 문자식 영역에서는 LaTeX 형식의 표기법을 특정 기관의 실무 수준의 표준이 아니라 국가적 수준의 표준으로 해야 한다는 것이다.

ISO 8879의 특수문자체 표준은 국내의 경우 KAIST의 NDSL과 KISTI의 YesKisti의 현행 업무에서 적용하고 있으며, PubMed와 같은 대규모 외국 시스템에서도 적용하고 있기 때문이다[11].

이 표준 이외에는 가장 적합한 문자체 세트를 개발하거나, 아예 유니코드 문자체에서 검토할 수밖에 없다. ISO 8879 표준을 채택하였을 경우, PubMed 시스템처럼 국내 과학기술정보 표현 환경을 분석하여 문자체의 배열순서를 변경하거나 추가적인 문자체를 개발하여 확장할 수 있으며, NDSL처럼 다른 표준을 추가할 수도 있다. 그러나 이 문제는 현재로서는 그다지 시급하지 않다.

이보다는 ISO 8879 표준에서 정의된 특수문자체들이 웹 상에서 HTML 4.0으로 출력표기가 잘 안되는 문제이다. 이는 브라우저나 컴퓨터 운영체제와 같은 문제이거나, 웹 출력표기를 처리하는 프로그램의 문제이거나, 개인이 브라우저에서 문자표기법을 어떻게 설정(euc-

11) KAIST NDSL에서 특수문자 세트는 ISO 8879-1986과 ISO 9573-13:1991의 2가지 표준을 기본으로 삼고 있다. 그러나 NDSL, YesKisti, PubMed 모두 적용하고 있는 표준문자 세트는 ISO 8879 표준이므로 우선적으로 이것을 기본으로 제안하는 것이다.

kr, UTF-8 등)하였는가 등에 따라 영향을 받는 것 같아 보인다. 또한 적절한 문자 폰트가 없어 출력표기가 안되는 경우도 있을 수 있다. W3C에서는 이 경우에 대하여 해당 문자체의 16진수 형식으로 표기하고, 차후 해결되도록 하는 방식을 권장하고 있다.

문자식 표준은 입력표기법과 출력표기법을 구분하여야 하며, 입력표기법은 KISTI에서 사용하고 있는 LaTeX 형식의 표기법이 가장 무난하다고 생각한다. 우선 LaTeX 형식이 국내외에서 가장 많이 알려졌고, 과학기술자들이 가장 많이 사용하고 있는 방식이며, 또한 최근에 자동으로 웹 출력표기하는 방법이 개발되었기 때문이다. 즉, 입력표기와 출력표기뿐만 아니라 표기법 호환성에도 뛰어나다. 그러나 LaTeX 형식의 입력표기법이 웬만한 문자식에 대한 표기가 가능한 포괄성에 있어 가장 큰 장점이 있지만, LaTeX 형식의 문자식을 구성하는 방식이 어렵다는 점과, 경우에 따라서는 LaTeX이라는 편집프로그램을 설치하여야 하는 문제도 발생한다. 이 문제는 LaTeX 형식의 입력표기가 가능한 전용의 문자식 입력기(editor)를 개발하는 방향으로 해결해 나가면 될 것이다. 물론, 이 입력기에는 ISO 8879 표준의 특수문자체를 편리하게 입력할 수 있는 기능도 포함하면 특수문자 입력기로서 활용도가 매우 높아질 것이다.

제안된 과학기술정보를 위한 특수문자 표기법 표준은 특수문자체와 문자식의 입력과 출력 표기 문제, 그리고 호환에 관한 문제를 중심으로 검토되었다. 비록 표기법의 관점이라고 하지만, 입출력 및 호환 문제 이외에도 색인과 검색 문제, 이용자 친화적인 에디터 개발 등 관련된 연구들이 후속과제로 연결되어야 할 것이다.

참고문헌

백두권. 2005. 과학기술정보 표준화와 NTIS 구축. 『국가과학기술종합정보 시스템 구축을 위한 과학기술정보 표준화 사업 중간보고 및 공청회 발표자료』2005년 7월 19일. [서울: 리츠칼튼 호텔].

에릭 레이. 2001. 『XML 시작하기』. 장은영 역. 서울: 한빛미디어.

이상로. 1997. 코드와 한글. [인용 2005. 8. 2]. 〈http://trade.chonbuk.ac.kr/~leesl/code/〉.

정주원. 1995. 한글 코드에 대하여. [인용 2005. 8. 2]. 〈http://www.w3c.or.kr/i18n/hangul-i18n/ko-code.html〉.

Alan Wood. 2004. HTML4.0 Characters Entity References web page. [cited 2005. 8. 18]. 〈http://www.alan-wood.net/demos/ent4_frame.html〉.

ISO. 2005. ISO Homepage. [cited 2005. 8. 18]. 〈http://www.iso.org/iso/en/ISOOnline.frontpage〉.

KISTI. 2005. YesKisti 홈페이지. [인용 2005. 8. 23]. 〈http://www.yeskisti.net/〉.

KSA. 2005. KSSN 국가표준정보센터 홈페이지. [인용 2005. 8. 10]. 〈http://www.kssn.net/default.asp〉.

KTUCT. 한글 TeX 사용자 그룹(KTUG) 홈페이지. [인용 2005. 8. 10]. 〈http://www.ktug.or.kr/〉.

NDSL. 2002. NDSL 홈페이지. [인용 2005. 8. 23]. 〈http://ndsl.or.kr/eng/new-index.html〉.

PINT Inc. 2005. HTML & XHTML The complete Reference web page. [cited 2005. 8. 18]. 〈http://www.htmlref.com/reference/appc/standard.htm〉.

Powell, Thomas A. 2001. 『HTML 완벽 가이드』. 김병화 역. 서울: 사이텍미디어.

NCBI. 2005. PubMed Homepage. [cited 2005. 8. 18]. 〈http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed〉

Tiffany B. Brown. 2002. Tiffany B. Brown Homepage. [cited 2005. 8. 18]. 〈http://www.tiffanybbrown.com/htmlentities.php#greek〉.

TRIO. HTML 4.01 한국어 번역 웹 페이지. [인용 2005. 8. 18]. 〈http://trio.co.kr〉.

Unicode, Inc. 2005. Unicode Homepage. [cited 2005. 8. 18]. 〈http://www.unicode.org/charts/〉.

Vidar Broken Gundersen, and Rune Mathisen. 2001. ISO character entities and their LATEX equivalents. [cited 2005. 8. 18]. 〈http://www.bitjungle.com/~isoent/isoent-ref.pdf〉.

W3C. 1999. World Wide Web Consortium web page. [cited 2005. 8. 18]. 〈http://www.w3.org/TR/REC-html40/〉.