(제 10 회 한글 및 한국어 정보처리 학술대회)

한글 환경에서의 다국어정보 환경구축*

정휘웅
부산대학교 인지과학협동과정
부산 금정구 장전동 산 30, 우:609-735
hwjeong@hyowon.cc.pusan.ac.kr

윤애선
부산대학교 불어불문학과
부산 금정구 장전동 산 30, 우:609-735
asyoon@hyowon.cc.pusan.ac.kr

Building Multilingual Information Structure in Korean Environment

Hwi-woong Jeong
Interdisciplinary Research Program of Cognitive Science
Pusan National University

Aesun Yoon
Department of French
Pusan National University

요약

인터넷은 다양한 언어로 구성된 정보를 사용자들에게 제공해 준다. 따라서 인터넷 환경상의 정보 다국어화는 앞으로도 점차 가속화될 것으로 보인다. 그러나 각 국가별 지역 코드는 다국어 정보화를 가로막는 하나의 걸림돌이 되고 있다. 본 논문에서는 실사용자(end-user)와 개발자(developer) 환경에서 발생하는 다국어 지원의 문제점에 대해 알아보고, 이를 부산대학교 언어 정보 연구실에서 연구중인 다국어 지원 방법과 연관하여 특히 웹 환경에서 다국어가 동시에 지원될 수 있는 방안에 관해 연구하였다. 한글 환경에서 다국어가 원활히 지원되기 위해서는 유니코드 도입과 함께 다국어 입력 알고리즘이 개발되어야 하며, 이에 따른 다국어 입력 컨트롤 및 라이브러리 개발이 선행되어야 한다. 또한 웹 환경에서도 KS-C-5601 기반이 아닌 Unicode 기반 웹 환경 구축이 진행되어야 할 것이다.

1 서론

인터넷의 급속한 보급은 사용자들에게 보다 다양한 언어를 접할 수 있는 기회를 만들었다. 이에 따라 기존에 자국의 전산 환경에 따라 진척된 국가별 전산 환경은 점차 Global 화 되어가는 추세에 있다. 실례로 인터넷 환경의 핵심적 도구인 웹 브라우저는 Unicode 와 UTF-8 코드체계와 같은 다국어 지원 영역을 만들어놓고 있으며, Macintosh 의 경우 Unicode 를 시스템의 기반으로 채택하고 있다.

그러나 아직까지 많은 수의 사용자를 확보하고 있는 일반 PC 의 경우 과거 시스템과의 호환성 문제로 인해 글로벌화를 위한 통일화 연구 및 부수 연구에 대한 진척이 많이 이루어지지 않고 있는 실정이며, 이러한 현상은 웹 환경에도 영향을 미치고 있다. 이에 따라 현재 한글로 제작된 대부분의 인터넷 홈페이지는 KS-C-5601 코드로 작성되어 있으며, ISO 규격에 따른 홈페이지 제작이 아닌 경우가 많아 encoding 방식에 따라 웹 페이지의 문자가 다른 코드 해석방식으로 나타나는 경우가 대부분이며, 따라서 한글환경에서 다국어 정보를 구축하고 이용하는데 큰 걸림돌이 된다.

2 장에서는 한글 환경에서 발생하는 다국어 처리의 문제점을 실사용자와 개발자 측면에서 살펴보았으며, 3 장에서는 이러한 다국어 지원 문제점을 해결하기 위한 방법을 제시하였다. 5 장에서는 결론과 앞으로 추가적으로 연구되어야 할 영역에 대해 언급하겠다.

2 한글환경에서 다국어처리

지금까지 한글환경에서 다른 언어의 전산화는 영어와 한글 및 일부 한자어에 국한되어 진행되었다. 이러한 제한은 이 두 언어가 현재 한국어에 가장 많은 영향을 주는 언어라는 점에도 기인하나, 초기 PC 및 통신망 환경이 언어 문자의 다양한 특성을 고려하여 다국어를 동시에 지원할 수 있는 시스템을 제공하지 못했다는 데서 더 큰 원인을 찾을 수 있다. 하지만 인터넷의 보급은 다국어 지원에 대한 요구를 증가 시켰다. 따라서 지금의

* 이 논문은 1997 년도 한국학술진흥재단의 학술연구 조성비(국제협력공동연구과제)에 의하여 연구되었음.

한글 환경은 실사용자 및 개발자 모두에게 많은 문제점을 제기한다.

1.2 실사용자 차원

현재 다국어 지원은 운영체제 차원과 어플리케이션 차원으로 구분된다. 실사용자에게 다국어를 지원하는 문제는 운영체제 차원에서 다각도로 연구되어 왔다. 그러나 여전히 입력 및 어플리케이션 지원 차원에서 많은 문제점을 내포하고 있다.

1.2.1 입력

윈도우 운영체제의 경우 극동권 언어를 제외한 서구 언어의 자판 세트를 설치하여 다국어 입력이 가능하도록 지원한다. Microsoft 워드의 경우에도 이러한 기능을 지원하고 있으며, Internet Explorer 의 경우 극동권 언어 IME(Input Method Editor)를 설치하여 일본어 및 중국어를 입력할 수도 있다. 그러나 서구 언어의 경우 국내에서 사용되는 자판과 그 배열이 상이하다.

또한 자국 언어의 원활한 입력을 위하여 특정 키는 액센트(accent)등 특수기호가 첨가된 문자를, 이외의 문자는 형태적 조합 방식을 선택하여 입력하는 방식을 채택하고 있다. 예를 들어 네덜란드어의 경우 á를 입력하기 위해서는 ' + a를 입력해야 한다. 불어의 경우 é를 입력하기 위해서는 숫자키 2를 눌러야 한다.[1] 이 경우 한글과 서구언어 자판을 숙지하는 사용자의 경우 입력에 많은 문제점이 없으나, 처음 다국어를 입력하는 사용자는 혼란에 빠질 수 있다.

이를 위해 Microsoft word 의 경우 ctrl + 형태적 기호 + 알파벳 문자의 형태로 다국어를 통합적으로 입력하는 알고리즘을 도입하고 있다. 아래아 한글의 경우 다국어 자판 설치를 운영체제에 의존하지 않고 독자적으로 도입하여 지원하고 있으나, 이 역시 앞서 언급한 언어간 자판 상이성의 문제점을 여전히 안고 있다.

더욱이 자판에서 찾기 어려운 문자의 경우 심벌 문자로 처리하기 위해 마우스를 이용하여 입력하는 방식도 있다. 이 경우 일반 문자가 아닌 기호 형식으로 문자코드가 처리되어, 문자처리의 궁극적 목적인 자연언어 처리 자료로 사용될 수 없다.

2.2.1 Windows Application 의 다국어 지원

지금까지 다국어 지원은 인쇄매체를 기준으로 한 워드프로세서 기준으로 발전되어 왔다. 탁상 출판이 보편화되기 시작한 1980 년대 추세는 인쇄상에서 요구되는 다양한 문자 출력에 그 초점이 모여져 있었다. 따라서 다국어 입력 역시 지역자판 및 지역 코드 기준으로 개발되었으며, 어플리케이션간 호환성을 고려하지 않은 문자 코드로, 그 궁극적 목표는 양질의 인쇄 품질을 얻는데 있었다.

그러나 인터넷의 보급과 함께 호환이 어려운 이진(binary) 형태의 정보는 점차 Markup 언어화 되고 있다. 이 과정에서 각 국가별로 지역화된 text 문서가 스크립트화 되는 구조가 채택되었으며, 극동언어권에서는 영어와 자국어를 표현하기 위해 DBCS(Double Byte Code Set)을 사용하였다. 한글의 경우에도 이 DBCS 는 지금까지 많은 어플리케이션의 핵심 영역으로 차지하고 있다. 이는 스크립트 문서가 Unicode 나 UTF-8 포맷으로 쉽게 전환되지 못하는 하나의 문제점이 되고 있다.

공통 코드 사용으로 다국어 입출력이 해결된다 하더라도 각 국가별 코드를 원활하게 입력할 수 없으면 다국어 지원의 효과는 반감될 것이다. 본 연구에서는 각 어플리케이션별 코드 중심 정보 전송과 Local code 가 Unicode 에 어떠한 영향을 미치는가에 대해 실험하였다. 결과는 Visual Basic 과 Visual C++ MFC, 한글 Windows 98 환경을 기준으로 실험하였다. 전송 문자열은 U+00E0 (à)과 한글문자 U+AC00(가)를 전송하였다.[2]

표 1 일반 문자열 전송

사용 함수	출력
VB Print 명령	? 가
TextOutA	? 가
TextOutW	a 가
ExtTextOutA	? 가
ExtTextOutW	a 가

표 2 Uncode 전송

사용 함수	출력
VB Print 명령	a 가
TextOutA	a 가
TextOutW	꼭ㅆㄱ
ExtTextOutA	a 가
ExtTextOutW	꼭ㅆㄱ

표 3 strconv 함수 이용, local code 변환

사용 함수	출력
VB Print 명령	a 가
TextOutA	a 가
TextOutW	꾀 ij
ExtTextOutA	a 가
ExtTextOutW	꾀 ij

이러한 실험 결과를 바탕으로 살펴볼 때, 현재 한글 환경에서 지원되는 다국어 영역은 매우 제한되어 있다. 또한 실제로 지원되는 함수군도 NLS(National Language

Support)API 에 의해 많은 제약이 따르므로, Unicode 자체가 어플리케이션 개발에서 중심적 역할을 하기 어려운 것을 확인할 수 있었다.[3] 실제로 Netscape 에서 공개한 Mozilla 의 경우도 시스템에서 지원하는 Code Page 가 아닌 독자적인 코드 페이지를 기준으로 가상 글꼴을 만들어 다국어를 지원하고 있다.[4]

3.2.1 Web Application 의 다국어 지원

웹 어플리케이션은 비교적 다국어 지원이 원활하게 이루어지고 있다고는 하나, 아직까지 취약한 점이 많다. 이는 개발자 차원의 지원이 부족함과 동시에 개발자들 간에 Unicode 도입에 대한 인식이 제대로 형성되어 있지 않기 때문일 것이다. 실례로 웹 개발 환경에서 다국어 지원 및 지역코드 구분을 위한 `content="text/html; charset=utf-8"`와 같은 형식의 HTML 명령은 국내 대부분 웹 페이지에서 제대로 지원되지 못하고 있다. 이 경우 각 웹 페이지는 사용자가 선택하는 encoding 모드에 따라 그림 1과 같이 영어 문자 기준으로 해석된다.

그림 1 영어코드로 해석된 한글 홈페이지

2.2 인터넷 개발자 차원

일반 어플리케이션 개발 환경은 많은 API 와 시스템 지원으로 어렵기는 하나 조금씩 다국어 지원이 가시화되어 가고 있으나, 인터넷 개발 환경에서는 아직 다국어 지원 문제가 제대로 해결되지 못하고 있다. 이는 Markup Language 와 UTF-8 과 같은 encoding 차원에서 그 문제점을 찾을 수 있다.

1.2.2 Markup Language 지원

웹 환경이 보편화되고 웹 환경 기반 어플리케이션 개발 및 데이터베이스 개발이 급속도로 이루어짐에 따라 많은 개발자들은 텍스트 문서화된 개발 환경에서 어플리케이션을 개발하고 있다. 이러한 Markup 언어가 기존 컴파일 중심 언어와 다른 점은 Markup 언어의 경우 실사용자에게 제시되는 정보와 실행용 코드가 하나의 텍스트 문서에 공존한다는 점이다. 따라서 모든 하나의 페이지는 하나의 코드 페이지로 구성되어야 하며, 이는 다국어 표현에 많은 제약이 된다.

비록 스크립트의 실사용자 노출을 방지하기 위해 scriptlet 과 같은 형식의 코드 지원도 있기는 하나, 궁극적으로는 지역 코드로 해석이 가능한 원시 코드와 실사용자에게 제시되는 문자 정보가 공존한다는 것은 개발 과정에서 문자 정보 및 기초 정보가 손상될 수 있다는 것을 보여준다. 비록 HTML 4.0 에서 Lang 태그를 이용하여 태그 내부에 존재하는 문자의 언어를 명시적으로 알려준다고는 하나,[5] 실질적으로 이를 편집하는 개발자 입장에서는 이러한 다국어가 지원되는 텍스트 편집기가 없을 경우 다국어 지원 페이지를 작성할 수 없다.

가령 HTML 로 작성한 <P Lang="Kor"> 태양 </P> <P Lang="Eng"> sun </P> <P Lang="Grk"> ηλιος </P> 과 같은 문자열을 한글 환경에서 입력하면, 그리스어 ηλιος 는 한글과의 충돌로 인해 제대로 표현되지 못한다.

HTML 에서는 é 를 입력하기 위해 é 와 같은 방식을 사용하여, &H80 이후 문자와 극동권 문자 코드와의 충돌을 부분적으로 해결하고 있다. 하지만 이는 서구 언어 중심적 사고로서 극동권 언어 및 러시아어, 그리스어 등 글로벌 언어 지원에는 미약한 점이 많다. 이는 다국어 입력 기능이 지원되는 일반 운영체제 차원의 함수 및 기능 지원, 그리고 입력된 코드를 표현하기 위한 글꼴의 지원이 미약하기 때문이다.

2.2.2 UTF-8

지역코드는 지금도 많은 개발자들로부터 선호되고 있다. 한글 환경의 경우 KS-C-5601 코드로 제작된 페이지가 그 주류를 이루고 있다. 그러나 브라우저 환경에서는 다국어 지원이 앞으로 추세로 진행되어 나갈 것이며, 이는 개발자들이 적극적으로 도입해야 하는 영역일 것이다. 그러나 현재 개발되어 있는 웹 어플리케이션 중 다국어 지원이 비교적 잘 이루어지고 있는 Front Page 의 경우에도 UTF-8 변환 작업을 하는 경우 text editor 의 언어 지원 미숙으로 인해 모든 문자가 알아볼 수 없는 형태로 바뀌는 문제가 있음을 확인할 수 있다. 그림 2는 "대한민국"이라는 홈 페이지를 제작한 뒤, 이를 UTF-8 코드로 변환하여 스크립트 모드로 본 예이다.

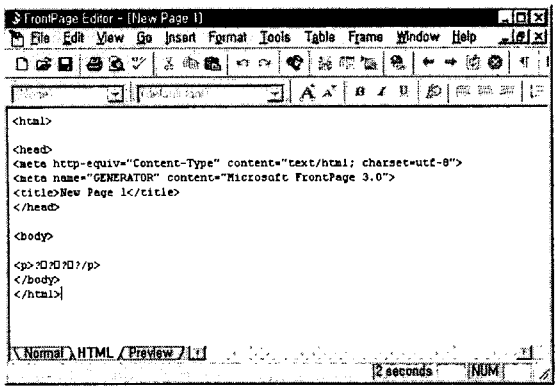

그림 2 UTF-8 로 변환된 한글 홈페이지

3 한글과 다국어 처리 제안

2 장에서 언급한 문제점들을 해결하기 위해 본 논문에서는 한글과 다국어가 원활히 처리되기 위해 다음과 같은 사항들을 제안한다.

1.3 Unicode 도입

Unicode 는 국제 규격 코드 세트로서 KS-5700 코드는 이 Unicode 기준에 맞도록 구성되어 있다. 또한 한글의 초성, 중성, 종성 구성에 의한 고어 입력도 가능하도록 되어 있어 다국어 지원 뿐만 아니라 우리말의 코드화에도 많은 장점을 보이고 있다.[6]

그러나 현재 국내에서 작성중인 대부분의 문서는 Unicode 기준이 아닌 KS-C-5601 기준으로 작성되고 있다. 이는 대부분의 문서 편집기 및 일반 어플리케이션이 Unicode 에 대한 지원이 제대로 이루어지고 있지 않기 때문이다. 그러나 Unicode 는 한글 환경에서 다국어 지원을 위해 하루빨리 보급되어야 할 영역이다. 부산대학교 언어정보 연구실의 경우 전자사전 개발 시스템과 불어 교육 시스템에서 윈도우 운영체제에서 개발자들에게 적극적으로 사용을 유도하고 있는 Unicode 를 도입하여 시스템 개발의 기반으로 이용하고 있다.

2.3 다국어 텍스트 문서 편집기

개발자 환경 및 실사용자 환경에서 Unicode 지원이 더딘 이유는 Unicode 가 지원되는 컨트롤이 제대로 없기 때문이다. 2 장에서 언급하였듯이, 다국어 기반 스크립트 언어를 작성하기 위해서는 다국어 입력 가능 문서 편집기가 요구된다.

만약 다국어 텍스트 문서 편집기가 일반 사용자 및 개발자들에게 제공된다면 이를 이용한 간단한 문서 제작기 및 어플리케이션 개발이 용이하게 될 것이다. 본 연구에서는 그림 3과 같이 베타버전으로 개발된 다국어 텍스트 문서 편집기의 이전 형태인 one-line 다국어 텍스트 입력기를 제시한다.

그림 3 다국어 입력 컨트롤

본 컨트롤은 서유럽어 및 다국어 입력 기능이 포함된 컨트롤이며, 두 줄 이상의 문서는 작성할 수 없도록 구성되어 있다. 본 컨트롤과 같은 형태의 진보된 형태로 보다 빠른 속도로 구동되는 다국어 텍스트 문서 편집기가 개발된다면 웹 및 어플리케이션 환경 모든 곳에서 한글 및 다국어 지원이 용이하게 이루어질 수 있을 것이다. 본 컨트롤에서 정의한 입력 방식은 인간의 심상구조에 맞는 후위표기 방식이며, 특수문자와 조합 문자간 형태적 동일성을 기초하고, 문서내 사용자 사용 빈도 특성을 고려하여 정의하였다.

표 4 유럽권 문자 입력 알고리즘

특수기호명	예	한/영 자판 입력
Acute	á	O + /
Breve	ŏ	O + $
Cedilla	ç	O + |
Circumflex	â	O + ^
Diaeresis, umlaut, trema	ä	O + "
Dot	ż	O + *
Double Acute	ű	O + :
Grave	à	O + ₩
Haček or carom	ǎ	O + #
Macron	ē	O + _
Ogonek	ą	O + `
Ring	å	O + @
Slash	ø	O + %
Tilde	ñ	O + ~

3.3 Hinting 지원 Unicode 글꼴의 개발

한글 글꼴은 영어 글꼴과 KS-C-5601 지정 특수문자, 한글, 한자로 구성된다. 윈도우 환경은 코드에 맞는 문

자를 Unicode와 대치하여 적절하게 조합하여 사용하고 있다. 그러나 웹 환경에서 이러한 글꼴들은 영어의 경우 그림 4와 같이 제대로 모양이 맞지 않는 문제점을 나타낸다.

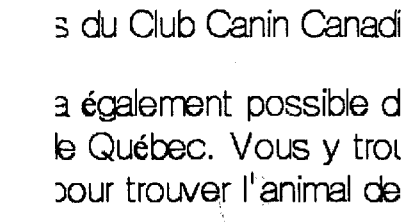

그림 4 영어간 불일치되는 글꼴

이러한 문제점을 해결하기 위해서는 Unicode가 지원되는 다국어 글꼴의 개발이 요구된다. 비록 Cyberbit사에서 13mb에 달하는 Unicode 지원 글꼴을 개발 해 두었으나, 이 글꼴의 경우 그림 5와 같이 hinting이 제대로 되어 있지 않아 화면상에 미려하게 나타나지 않는다. 한글과 일본어, 중국어의 경우 hinting이 어려워 낮은 해상도에서는 bitmap 글꼴을 사용하고 있는 경우도 있으나, 궁극적으로는 hinting조절이 잘 되어 있는 양질의 Truetype 글꼴의 개발이 시급하다.[7]

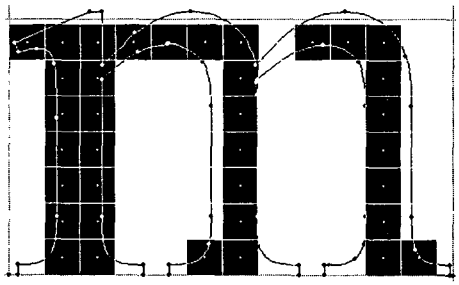

그림 5 hinting이 제대로 되지 않은 경우

부산대학교 언어정보 연구실의 경우 이러한 문제를 해결하기 위해 hinting과 회색톤 표현 기법을 도입하여 기존 서체에 기반한 일반 크기에서도 가독성이 높고 미려한 다국어 지원 트루타입 글꼴을 개발하였다.

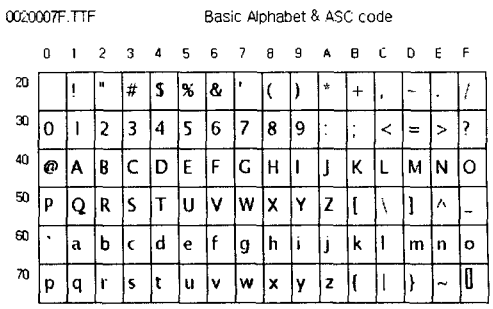

그림 6 hinting이 지원되는 트루타입 글꼴

4.3 입력 알고리즘 개발 및 라이브러리화

앞서 언급한 다국어 텍스트 문서 편집기의 경우 Active X 컨트롤 형태로 어플리케이션이나 웹 환경에 지원되거나, DLL 형식으로 구성되어 개발자들이 이를 이용하여 쉽게 다국어 지원 어플리케이션을 작성할 수 있도록 지원되어야 할 것이다. 그러나 입력 방식의 경우 하나의 언어가 아닌 다국어를 입력하여야 하므로 이에 대한 새로운 알고리즘이 요구될 것이다.또한 이렇게 개발된 라이브러리는 그림 7와 같이 확장성 있게 다른 환경에도 적용될 수 있을 것이다.

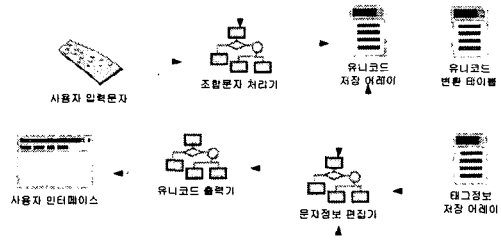

그림 7 Unicode 입력기 및 처리 루틴의 적용

부산대학교 언어정보 연구실에서는 서유럽어의 경우 앞서 언급하였듯이 형태적 특성에 따른 후위 표기 방식을 채택하고, 그리스어와 러시아어의 경우 기존 자판 도입 방식에 의한 다국어 입력 알고리즘을 도입하고 있다.

5.3 웹 환경상의 Unicode 및 UTF-8 도입

웹 환경의 다국어 지원은 Unicode만으로는 이루어지지 않는다. Explorer와 Netscape Communicator의 경우 Unicode를 전송할 경우 그 코드를 제대로 번역하지 못한다. 이러한 문제를 해결하기 위해 UTF-8 코드체계의

도입과 그림 8과 같이 이를 변환하는 시스템을 웹 개발 환경에 적극 도입하여야 할 것이다.

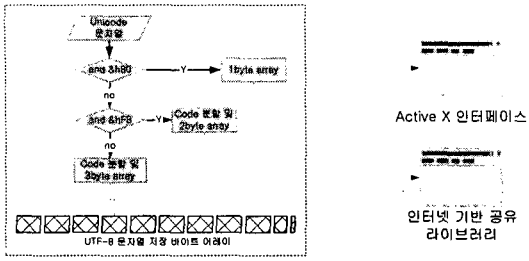

그림 8 UTF-8 변환기와 그 응용

비록 Unicode가 Windows98 버전에서 지원되는 Internet Exploer에서 지원된다고는 하나, 과거 버젼 및 다른 Netscape Navigator와의 호환성을 고려한다면 UTF-8 코드에 대한 지원이 뒤따라야 할 것이다. 더욱이 원시 코드가 많이 포함되는 웹 환경의 경우 UTF-8을 이용하여 코드 자체의 정보 크기를 줄임으로서 인터넷 통신환경상의 네트웍 부담을 줄일 수 있을 것이다. 부산대학교 언어정보 연구실의 경우 전자사전 개발 시스템에 UTF-8을 도입하여 그림 9와 같이 발음기호 표현과 다국어 표현에 본 변환기를 시험적으로 사용하고 있다.

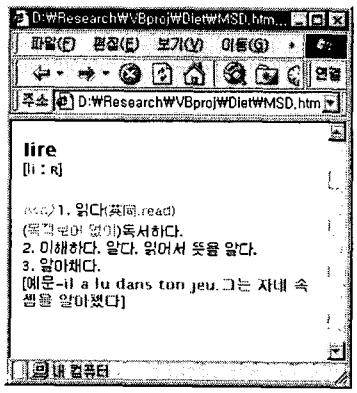

그림 9 UTF-8로 구성된 웹 페이지(발음기호)

4 결론

본 논문은 웹 환경에서 한글과 다국어의 문제점 및 해결 방안을 모색하여 보았다. 첫째, 영어권 중심으로 구성되어 있는 컴퓨터 환경에서 한글 소프트웨어가 발전하기 위해서는 Unicode에 대한 도입이 시급하다. 둘째, 스크립트 언어의 보급과 함께, 스크립트 언어에 대한 다국어 지원과 이를 뒷바침할 수 있는 공통된 편집기 및 어플리케이션 기반이 요구된다. 셋째, 다국어를 한글 환경에서 원활하게 지원하기 위한 입력 차원의 사용자 인터페이스 연구가 요구된다. 넷째, 다국어 지원이 가능한 다양한 글꼴이 개발되어야 할 것이다. 다섯째, 웹 환경에서 원활한 코드 전달을 위해 UTF-8과 같은 공통 코드에 대한 연구가 활발히 있어야 할 것이다. 이와 같이 한국어 및 다국어가 동시처리될 수 있다면, 한편으로는 국내의 정보화 사각지대에 놓인 많은 언어의 정보화 기반 사업을 가속화할 수 있으며, 다른 한편으로는 국내에서 개발된 소프트웨어의 국제화에 이바지할 수 있을 것이다.

참고문헌

[1]Kano, Nadine, *Developing International Software for Windows 95 and Windows NT*, Microsoft Press, 1995

[2] The Unicode Consortium, *The Unicode Standard-version 2.0, Addison Wesley*, 1996

[3] premium.microsoft.com – International Support in Window NT 5.0(/msdn/labrary/conf/pdc97/intl_supnt_5.htm)

[4] 월간 마이크로소프트웨어, 넷스케이프의 풀소스를 분석한다, MOZILLA - 7부 : 핵심 모듈 5 – 모질라로 S/W 세계화를! 국제화와 지역화, 1998/6

[5] www.w3c.org - HTML 4.0 Specification

[6] 김경석, 컴퓨터 속의 한글 이야기, 영진출판사, 서울, pp.203-208, 1995

[7] www.microsoft.com - Basic hinting philosophies and TrueType instructions (/typography/hinting/tutorial.htm)