인터넷을 이용한 하이퍼텍스트 기반 서지정보검색 시스템에 관한 연구

A Study on Hypertext-based Bibliographic Information Retrieval System Using Internet

박지연(Ji-Yeon Park)* · 문성빈(Sung-Been Moon)**

□ 목 차 □

1. 서 론
1.1 연구의 목적 및 의의
1.2 연구의 방법
2. 하이퍼텍스트와 정보검색
2.1 하이퍼텍스트에서의 정보검색방법
2.2 서지데이타의 하이퍼텍스트화
3. WWW와 하이퍼텍스트
3.1 WWW의 개요
3.2 CGI(Common Gateway Interface)
3.3 WWW와 데이타베이스의 연결
4. 시스템 설계 및 구현
4.1 시스템 구성
4.1.1 WWW서버의 설치
4.1.2 데이타베이스 구축
4.1.3 CGI 프로그램
4.2 연세대학교 대학원 학위논문 검색모듈
4.2.1 브라우징 기능
4.2.2 키워드검색 기능
4.2.3 관련레코드로의 이동
4.2.4 브라우저의 다양한 기능 조합
5. 시스템 평가 및 결론

초 록

본 연구는 하이퍼텍스트에 기반을 둔 서지정보검색 시스템을 구현하여 하이퍼텍스트 시스템이 구조화된 서지데이터의 브라우징 및 검색에 유용함을 보여주었고, 키워드검색기법을 적용하여 이용자의 방향상실문제를 보완하였다. 또한 WWW를 이용하여 시스템을 구현함으로써, 본 시스템의 장점 및 잠재력을 제시하였다.

ABSTRACT

In this study, we constructed a hypertext-based bibliographic information retrieval system, which is very useful tool to browse and retrieve structured data. We minimized the problem of the user's disorientation with the keyword retrieval technique. We also presented the potential advantages of this system which could be obtained by implementing it on WWW.

* 연세대학교 대학원 문헌정보학과
** 연세대학교 문헌정보학과 조교수
■ 논문접수일 : 1995년 10월 31일

1. 서 론

1.1 연구의 목적 및 의의

정보검색시스템에 있어서, 정보접근의 중요한 측면은 시스템의 구조와 관련이 있다. 현재 온라인 검색시스템에서 가장 많이 채택하고 있는 불(Boolean)검색시스템은 도치색인을 이용한 선형모델(linear model)로서, 색인어를 통해 원하는 정보에 접근한다. 이러한 방법은 탐색문작성과 컴퓨터처리가 용이하다는 장점이 있지만 이용자가 만족할 만한 검색결과를 얻지 못했거나, 초기 정보요구와 유사한 관련정보를 계속 브라우징(browsing)하고자 할 때 레코드간의 이동이 쉽지 않다.

이에 반해 하이퍼텍스트 시스템(hypertext system)은 정보의 기본단위인 노드(node)와 노드들을 연결해 주는 링크(link)들의 네트웍 구조로, 이용자는 탐색어를 사용하지 않고 링크를 따라 자유롭게 이동하면서 원하는 정보에 접근한다. 이러한 구조는 인간의 사고과정과 유사한 연관적(associative)이며 비선형적(non-linear)인 접근 방법을 제공하여 관련 정보의 브라우징을 용이하게 한다.

그러나 하이퍼텍스트 시스템 역시 한계점을 갖고 있는데, 정보공간이 커질수록 브라우징만으로 원하는 정보에 접근한다는 것이 매우 어렵고 때로는 이용자의 방향상실문제(disorientation)를 야기시킨다. 따라서 이와 같은 하이퍼텍스트 시스템의 문제점을 해결하기 위해 그래픽지도(graphical map)와 같은 브라우징 도구를 제공하거나 키워드검색과 같은 기존의 정보검색기법들을 적용하여, 좀 더 효과적으로 정보에 접근하기 위한 여러가지 연구가 이루어지고 있다. 그러나 이러한 연구는 대부분 구조화되어 있지 않은 텍스트나 그래픽 데이터를 대상으로 하고 있으며, 구조화되어 있는 서지정보검색에 있어서의 응용이나 그 효율성에 대해서는 널리 연구되지 않은 실정이다.

최근에는 하이퍼텍스트 시스템에 멀티미디어 기술을 응용하여 문자정보뿐만 아니라 그래픽, 영상, 음성, 동화상등의 정보도 함께 구조화하고 있는데, 이를 하이퍼미디어 시스템(hypermedia system)이라고 하며 이에 대한 연구가 매우 활발히 진행되고 있다.

이러한 추세는 세계최대의 정보네트웍인 인터넷에서도 찾아볼 수 있는데, 인터넷 정보탐색 도구의 하나인 WWW(World Wide Web)는 하이퍼텍스트에 기초하여 인터넷의 다양한 자료를 검색해 볼 수 있도록 한 클라이언트/서버(client/server)모형의 분산 하이퍼미디어 시스템이다. WWW는 Mosaic, Netscape등의 다양한 브라우저가 개발됨으로써 기존의 문자를 사용하는 방식 대신 그래픽 사용자 환경에서 정보를 보다 쉽고 편리하게 검색할 수 있다. 특히, 다양한 데이터 형식과 프로토콜을 지원하여 인터넷상의 다른 도구들 즉, FTP, Gopher, WAIS, Usenet 등과의 인터페이스도 제공하며, CGI(Common Gateway Interface)를 통해 외부 프로그램을 실행시킬 수 있도록 한다.

이에 본 연구는 WWW를 이용하여 서지데이터베이스를 하이퍼텍스트화 하고, 본 시스템을 WWW 서버상에서 구현하여 이용자가 GUI환경의 브라우저를 통해 쉽고 편리하게 데이터를 검색할 수 있도록 하는데 그 목적을 두고 있다.

또한 하이퍼텍스트의 브라우징과 방향상실문제를 보완하기 위해 키워드 검색기법을 적용하고자 한다. 이를 통해서 이용자는 방대한 정보공간에서 자신의 탐색대상 범위를 축소함으로써 보다 효과적인 브라우징이 가능할 것이다.

1.2 연구의 방법

본 연구는 문헌조사를 통해 하이퍼텍스트 시스템과 WWW의 제반 특성 및 유용성을 살펴보고, WWW 서버상에서 하이퍼텍스트에 기반한 서지정보검색시스템을 구현하였다.

먼저 WWW 서버를 설치하기 위해, SunOS 4.1.1에서 운용되는 SPARC Workstation을 사용하였고, HTTPD(HyperText Transfer Protocol Daemon)는 NCSA(National Center for Supercomputing Application)의 1.4.3 version을 사용하였다.

서지데이터베이스는 mSQL(miniSQL)이란 데이터베이스 시스템 이용하여 구축하였고, 서지데이터는 1993년 이후 발표된 연세대학교 본 대학원 석.박사학위논문의 서지사항을 대상으로 하였는데, 이는 저자가 직접 키워드를 선정하므로 링크선정시 이를 그대로 사용할 수 있어서 용이하기 때문이다.

구축된 데이터베이스와 WWW를 연결하기 위한 인터페이스는 WWW서버에서 제공되는 CGI를 이용하여 X-Window하에서 구현하였고, CGI프로그램을 위해서는 C언어를 사용하였다. 그리고 검색된 모든 결과는 HTML형식으로 제공되도록 하였다. 단, 본 시스템은 이용자가 인터넷상에서 GUI환경의 WWW 브라우저를 사용한다는 것을 전제로 한다.

2. 하이퍼텍스트와 정보검색

2.1 하이퍼텍스트에서의 정보검색방법

하이퍼텍스트의 정보공간은 노드와 링크로 구성된 복잡한 그물형태의 네트웍 구조로, 이용자는 링크를 따라 노드간을 이동해가면서 정보를 검색하게 된다. 이러한 행위를 네비게이션(navigation)이라 하는데, Nielson(1990)은 "네비게이션은 하이퍼텍스트에서의 매우 적극적이고 전략적인 이용자의 역할을 잘 나타내주는 은유적인 용어"라고 지적하였다. 즉, 선형구조를 가진 기존의 정보소장매체에서는 이용자는 저자가 이미 정해 놓은 순서에 따라 순차적으로 정보를 검색하게 되는 반면, 하이퍼텍스트에서는 이용자가 자신의 의도에 따라 선택한 링크를 통해 움직이게 되므로, 정보접근 경로를 이용자가 직접 결정할 수 있다는 것이다. Newby(1992)도 "이제 적합성(relevance)에 기반한 기존 정보검색 시스템에서 네비게이션을 기본 개념으로 하는 시스템으로 전환되어야 한다"고 주장하면서, 네비게이션을 이용자와 시스템간의 동적인 상호작용을 가능하게 하는 중요한 개념으로 보았다.

네비게이션과 유사한 개념의 용어로 브라우징이 함께 사용되고 있는데, 네비게이션이 하이퍼텍스트의 구조적인 측면을 강조한 용어라면, 브라우징은 정보검색의 방법적인 측면을 나타낸다고 할 수 있다. 즉, 이용자는 링크를 따라 자유롭게 이동하는 네비게이션을 통해 관련 정보를 브라우징하면서 원하는 정보에 접근하게 된다. 브라우징은 이용자가 정확하게 자신의 정보요구를 질의어로 공식화할 수는 없지

만 절실한 정보요구를 가지고 있을 때 용이한 정보검색방법(김영귀 1990)으로, 이용자는 브라우징을 통해 유용한 정보를 선별하고 적합한 정보를 찾는데 필요한 관련 개념을 획득할 수 있다. 이는 부시가 제시한 인간의 연관적인 기억구조를 잘 반영하는 정보검색방법으로, 따라서 하이퍼텍스트에서의 정보검색은 하이퍼텍스트의 구조적인 장점을 이용한 브라우징을 통해서 이루어진다고 할 수 있다.

그러나 하이퍼텍스트의 정보공간이 방대해질수록 브라우징만으로 적합한 정보를 찾는다는 것은 매우 어렵다. Conklin(1987)은 하이퍼텍스트의 단점의 하나로 방향상실(disorientation, lost in space)문제를 지적하였는데, 이는 이용자가 복잡하게 연결된 노드들의 전체 구조를 알지 못한채 링크를 따라 계속 검색하다 보면, 전체 데이터베이스에서 현재의 위치를 알 수 없게 되어 계속 진행해나가야 할 방향을 잃어버리게 되는 경우를 의미한다. 대부분의 시스템에서는 이를 해결하기 위해 그래프형태의 지도를 이용하여 현재의 위치를 파악할 수 있도록 하거나, 이전 노드로 되돌아 갈 수 있는 기능, 지금까지 참조했던 노드들에 대한 히스토리 정보를 소장하는 등 여러 보조도구를 제공하고 있다. 그러나 브라우징만으로 원하는 노드에 접근한다는 것은 이용자에게 여전히 많은 인지적 부담과 시간상의 부담감을 안겨주고 있다.

이러한 한계를 극복하기 위해 기존의 정보검색기법들과의 결합을 시도하는 많은 연구들이 진행되고 있는데, 가장 보편적으로 사용되고 있는 기법이 색인을 이용한 키워드검색 방법이다. 하이퍼텍스트에서 키워드검색 기법을 적용하는 방식은 크게 두가지로 나눌 수 있다(고영곤 등 1995). 하나는 내용기반(content-based) 검색이고 다른 하나는 구조기반(structure-based) 검색이다. 내용기반 검색은 모든 노드를 독립적인 엔티티로 간주하여 개별적으로 질의어에 합당한 노드를 찾는 방식이고, 구조기반 검색은 정보망의 구조를 고려하여 주어진 패턴과 가장 유사한 부정보망(sub-network)을 찾는 방식이다. 예를 들어, "hyper*"와 같이 hyper란 단어가 들어 있는 모든 노드를 찾는 경우에는 내용기반 검색이 되고, "hypertext라는 단어를 포함한 노드에 연결되어 있는 모든 노드를 찾아라"와 같은 질의의 경우에는 구조기반 검색이 된다. 예에서 보는 바와 같이 구조기반 검색기법을 사용하기 위해서는 하이퍼텍스트의 구조를 기술할 수 있는 질의언어가 설계되어야 하고 이를 해석하고 적절한 부정보망을 찾아낼 수 있는 검색엔진이 개발되어야 하므로, 현재 대부분의 시스템은 내용기반 검색기법을 적용하고 있다.

이렇게 하이퍼텍스트에 적절한 정보검색 기법을 적용함으로써, 이용자는 부적합한 공간을 배제시켜 자신의 탐색대상범위를 축소시킬 수 있으며, 제한된 영역에서 브라우징함으로써 정보접근 과정을 단축시킬 수 있다.

2.2 서지데이터의 하이퍼텍스트화

하이퍼텍스트에 기초한 정보시스템은 최근 몇년간 구조화되어 있지 않은 텍스트, 그래픽 정보의 접근도구로서 큰 주목을 받아왔지만, 구조화되어 있는 서지정보검색에 있어서 하이퍼텍스트의 응용이나 그 효율성에 대해서는 널

리 연구되어 오지 않았다.

Dimitroff와 Wolfram(1993)은 Hyper-Lynx란 시스템을 통해서 구조화된 서지데이터가 하이퍼텍스트 환경에 적합하다는 것을 보여주고 있는데, 가장 큰 이유는 서지레코드의 몇몇 구성요소들이 데이터베이스내의 관련 레코드들간의 링크를 형성하는데 기초로 사용될 수 있기 때문이다. 예를 들면, 저자나 디스크립터는 링크로 선정하기에 적합한 요소들인데, 디스크립터는 레코드의 개략적인 내용을 함축하고 있기 때문에 같은 디스크립터를 갖는 레코드들을 연결하면 이용자에게 관련 레코드들에 대한 중요한 정보를 제공할 수 있다. 같은 저자를 갖는 레코드들을 연결하는 것도 유용한 정보가 될 수 있는데, 이는 저자들은 종종 한 주제분야에서 한 저작이상을 생산하고, 또 대개 이용자의 질의어와 관련있는 다른 연구들을 하기 때문이다. 공저자도 이와 같은 이유에서 유용한 링크가 될 수 있다. 제목이나 초록으로부터 키워드를 뽑아 이를 링크로 사용할 수도 있으나, 이런 경우에는 수작업색인 혹은 자동색인과 같은 별도의 작업이 요구된다.

검색과정은, 이용자가 검색대상이 될만한 용어들을 담고 있는 색인을 통해 데이터베이스에 접근하여 용어를 선택하면, 이 용어가 출현한 첫 레코드에서부터 검색이 시작된다. 검색용어가 출현한 첫 레코드와 마지막 레코드가 원형리스트 형태로 연결되므로, 이용자는 링크를 따라 원하는 검색용어가 출현한 모든 레코드를 브라우징 할 수 있다. 브라우징 도중 어느때라도 저자나 공저자, 혹은 현 레코드의 다른 디스크립터와 연관된 다른 레코드로 이동할 수 있으며, 다른 검색어를 선정하기 위해 색인으로 돌아갈 수도 있다.

이렇게 하이퍼텍스트에 기초하여 서지데이터를 구성할 경우 많은 서지적 요소들이 링크로 사용될 수 있기 때문에 다차원(multi-dimensional)브라우징이 가능하다.

3. 인터넷과 하이퍼텍스트

3.1 WWW의 개요

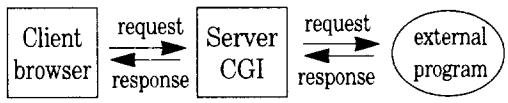
WWW는 1989년 Tim Berners-Lee에 의해 CERN내의 각종 연구자료들과 여기저기 산재해 있는 정보들을 효율적으로 사용하기 위한 목적에서 시작된 프로젝트로, 트리구조의 메뉴방식을 따르고 있는 Gopher의 불편함을 해결하기 위해 하이퍼텍스트의 링크개념을 도입하였다. 이를 기반으로 하여 텍스트외에 그래픽, 동화상, 음성정보에 이르기까지 각종 형식의 데이터를 손쉽게 볼 수 있도록 하고 있으며, 인터넷상에서 여러가지 프로토콜형식으로 존재하는 문헌들을 통일된 하나의 인터페이스를 통해 접근가능하다. 현재 WWW를 통해 이용가능한 서비스는 Gohper, WAIS에서 제공하는 모든 기능, Usenet, Telnet을 통한 모든 서비스, FTP, Archie, Veronica, Whois, Finger 등 인터넷 서비스의 거의 대부분이라 할 수 있다.

이러한 WWW는 1993년 NCSA에서 Mosaic이라는 브라우저를 개발하면서 GUI환경에서 보다 편리하게 정보를 이용할 수 있게 되었으며, 1994년에는 Mosaic의 기능을 좀 더 향상시킨 Netscape가 등장하면서 WWW의 이용은 폭발적으로 증가하게 되었다.

WWW는 FTP, Gopher, WAIS등과 마찬가지로 클라이언트/서버 모형에 기초하고 있는데, WWW에서 클라이언트는 그래픽이나 텍스트 모드의 브라우저를 사용하는 측이 되고, 서버는 HTTPD실행 프로그램이 설치된 컴퓨터가 된다. WWW의 동작과정을 살펴보면, 이용자가 하이퍼텍스트 문서에서 한 링크를 선택하면 클라이언트는 URL로부터 연결될 문서의 호스트주소를 읽어서 URL(Uniform Resource Locator)에 지정되어 있는 프로토콜을 실행하여 지정된 서버에 연결한다. HTTP 프로토콜인 경우, 지정된 호스트의 WWW서버에 연결된다(유사라 1995). 클라이언트는 URL에 연결된 호스트의 특정 문서를 지정하게 되는데, 서버는 해당 문서에 대한 MIME(Multipurpose Internet Mail Extensions)형식과 함께 데이터를 보내고 클라이언트는 MIME 형식에 따라 이용자의 단말기에 문서를 출력한다. MIME은 정보의 손실없이 메세지내용의 표현이나 변환형식을 지정하기 위한 형식으로, type/subtype형태로 표현된다. WWW는 하이퍼텍스트 문서를 작성하기 위해 HTML이라는 표준 마크업언어를 사용하는데, HTML형식의 텍스트의 경우는 text/html과 같다.

3.2 CGI(Common Gateway Interface)

일반적으로 WWW는 클라이언트가 URL을 통해 화일을 요청하면 서버가 이를 찾아내어 클라이언트에게 보내주는 방식으로 동작하게 된다. 그러나 이와는 달리 클라이언트가 요청한 화일을 서버가 실행시켜서, 그 실행결과를 보내주는 경우가 있는데, 이러한 기능을 CGI(Common Gateway Interface)라 한다. 다시말해서 CGI는 서버를 통해서 외부 프로그램이나 게이트웨이를 실행시키기 위한 인터페이스 프로그램이라 할 수 있다. CGI를 이용한 WWW의 동작과정을 모형화하면 <그림 1>과 같다.



<그림 1> CGI를 이용한 WWW의 동작과정

클라이언트가 CGI실행에 필요한 URL을 요청하게 되면 서버는 이에 해당하는 프로그램을 실행시키는데, CGI는 프로그램과 인수를 전달할 때 POST와 GET 두가지 방법을 사용할 수 있다. POST를 사용할 경우 표준입력방식으로 값을 넘겨주고, GET을 사용할 경우에는 환경변수(environment variable)로 값을 넘겨준다.

실행시킨 결과를 클라이언트로 전달할 때는 HTML문서뿐만 아니라 일반적인 텍스트, 이미지, 오디오 화일과 같은 다양한 형식을 전달할 수 있으며, 다른 문서에 대한 참조정보를 돌려줄 수도 있다. 이러한 정보를 클라이언트에게 알려주기 위해서 실행결과 화일 앞부분에 헤더(header)를 사용하는데, 완전한 MIME형태의 헤더형식은 Content-type : type/subtype 이며, 다른 문서에 대한 참조정보인 경우에는 Location : /directory/filename과 같이 명시한다.

클라이언트가 CGI변수를 통해 서버에게 주는 정보들은 HTML문서에서 제공되는 버튼을 클릭하거나 직접 텍스트를 입력함으로써 전달된다. 이러한 버튼이나 텍스트 입력박스들을

Forms라고 하며 HTML 태그 〈form〉…〈/form〉의 형식으로 정의된다.

3.3 WWW와 데이터베이스의 연결

데이터베이스의 여러가지 다양한 장점을 갖고 있는 데이터베이스시스템과 WWW를 연결하려는 연구들이 활발히 진행되고 있다. 인터넷상의 수많은 정보원들은 대부분 체계적으로 정리되어 있지 못하며, 현재 사용되고 있는 WWW검색엔진들도 논문이나 기술보고서 등의 학술자료를 검색하는데 있어서는 상용데이터베이스 시스템에 비해 정확도 면에서 많이 뒤떨어 진다. 따라서 이러한 자료들을 별도의 데이터베이스에 체계적으로 구축하여 이를 WWW에 연결시키는 경우 이러한 단점을 보완할 수 있다. 데이터베이스와 WWW를 통합시키는 경우, 데이터베이스의 데이터는 WWW의 브라우저를 사용하여 보게 되고, 데이터의 저장시스템으로는 DBMS를 이용한다. 이렇게 함으로써 데이터베이스의 장점을 그대로 유지하면서 이용자편의를 위해 구성된 하이퍼텍스트 형태의 인터페이스를 제공할 수 있다.

WWW와 데이터베이스를 연결하는 방법은 세가지로 나누어 볼 수 있다(이윤홍 1995). 첫째, 앞절에서 언급한 CGI를 이용하는 방법으로, 현재 가장 많이 이용되고 있는 방법이다. 이 경우에는 현재의 시스템을 변형시키지 않고 그대로 통합이 가능한 잇점이 있으나, 검색시간과 성능면에서 비효율적이라는 단점이 있다. CGI를 이용한 가장 대표적인 예를 들면, WAIS Gateway와 SQL 데이터베이스를 연결한 Gateway를 들 수 있다. 먼저 WAIS Gateway는 인터넷상에 퍼져있는 수많은 정보들을 자동색인하여 키워드검색할 수 있도록 한 WAIS를 연결하는 게이트웨이로, 손쉽게 인터페이스가 가능하다. 반면, 한글키워드 검색이 불가능하고 디스크용량면에서 데이터화일 크기 정도의 색인화일이 생성되므로 비효율적이다. WAIS Gateway에는 NCSA HTTP서버에 포함되어 있는 BasicWAIS, BasicWAIS를 다소 수정한 KidoWAIS, SFgate, WAISgate 등이 있다. SQL 데이터베이스를 위한 게이트웨이도 Form인터페이스를 사용하는데, 여기에는 Sybase를 WWW에 통합시킨 Web/Genera, GSQL, Oracle과 Plexus HTTP서버를 통합시킨 OraPlex등이 있다

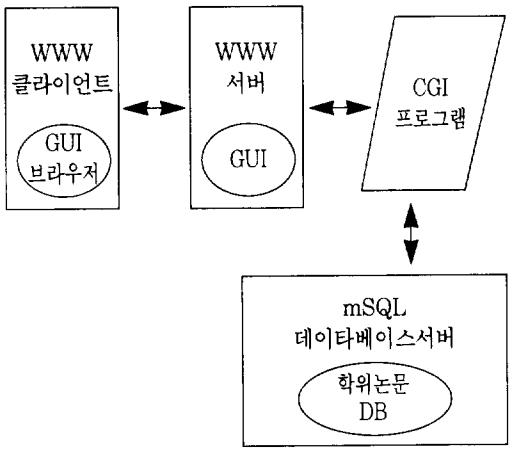
두번째 방법으로는 WWW서버를 수정, 확장시키는 방법이 있는데, CGI Gateway가 아닌 데이터베이스 Gateway개념을 이용한다. 이 방법은 다소 복잡하지만 URL형태를 그대로 사용하면서 데이터베이스와 인터페이스할 수 있는 경로명을 추가하면 되며, 성능면에서 CGI를 사용하는 것 보다 효과적이다. 이러한 예로써 OMEGA/Web이 있는데, 이는 상용 WWW브라우저를 사용하면서 HTTP서버를 확장시킨 것으로, 전문검색기능과 객체지향 멀티미디어 데이터베이스 관리기능이 있다.

마지막으로 WWW클라이언트를 수정, 확장하는 방법이 있다. 이 경우에는 WWW서버는 필요없게 되며 클라이언트와 데이터베이스 사이에 새로운 프로토콜이 필요하게 된다. 성능면에서는 가장 효과적일 것으로 기대되나, 아직까지 개발된 상용시스템은 없다.

4. 시스템 설계 및 구현

4.1 시스템 구성

본 연구에서 구현한 시스템은 크게 WWW 클라이언트, WWW 서버, CGI 프로그램, mSQL 데이터베이스, 이렇게 4가지 구성요소로 나누어 볼 수 있다. 이를 모형화하면 <그림 2>와 같다.



<그림 2> 시스템 구성

WWW 클라이언트는 정보를 검색하는 이용자로, 그래픽 환경의 WWW브라우저를 사용하는 것을 전제로 하여 시스템을 구성하였다. WWW브라우저로는 PC나 UNIX환경의 Mosaic, Netscape 등을 사용할 수 있으나 UNIX용 브라우저는 한글을 지원하지 못하므로 영문으로만 볼 수 있다. 본 시스템의 테스트용 브라우저로는 PC의 경우는 Netscape 1.2N 버젼을, UNIX에서는 Netscape 1.1 N 버전을 사용하였다.

WWW서버는 이용자의 요구에 따라 화일을 불러들이거나, CGI프로그램을 실행시켜 CGI 프로그램이 검색한 결과를 이용자에게 전달하는 기능을 한다. CGI프로그램은 mSQL 데이터베이스와 연결하여 이용자의 요구에 적합한 데이터를 검색하고, 이를 이용자가 볼 수 있는 HTML문서 형식으로 변환하여 서버에게 넘겨주는 역할을 한다. 여기서 CGI프로그램과 mSQL데이터베이스는 WWW와 같은 클라이언트/서버 모형으로 구현되어야 하는 것이 원칙이나, 본 테스트용 시스템의 경우에는 CGI 프로그램과 데이터베이스 서버가 같은 사이트에 있게 되므로 독립형(standalone mode)으로 동작하게 된다.

4.1.1 WWW 서버의 설치

WWW 서버 프로그램은 Unix환경의 HTTPD를 사용하는데, 이는 설치될 호스트의 시스템 종류에 따라 다르다. 본 시스템의 호스트는 SunOS 4.1.1의 Sun SPARC기종으로 이에 알맞는 httpd_1.4.2_sunos4.1.3을 설치하였다. 프로그램은 NCSA의 무명씨(anonymous) FTP서버에 접속하여 이미 컴파일 되어 있는 소스를 다운로드 받아 설치하였다(URL = ftp://ftp.ncsa.uiuc.edu/Web/httpd/Unix/ncsa_httpd/httpd_1.4/httpd_1.4.2_sunos4.1.3.tar.Z.)

4.1.1.1 서버의 환경 설정

시스템에 알맞는 서버의 환경을 설정해 주기 위해서 몇몇 구성(configuration) 화일을 수정해야 한다. 본 시스템에 따라 수정한 내용을 살펴보면 다음과 같다.

a) access.conf 화일

〈Directory /usr/local/etc/httpd/cgi-bin〉

Options Indexes FollowSymLinks ExecCGI

〈/Directory〉

CGI의 기능을 수행하기 위한 디렉토리를 서버가 설치되는 실제 디렉토리로 지정해 주고, CGI프로그램을 실행가능하게 하기 위해서 Options에 ExecCGI를 추가하였다.

〈Directory /usr/local/etc/httpd/htdocs〉

서버에서 제공하는 각종 문서들이 저장되고 관리될 서버의 홈 디렉토리를 htdocs로 지정하였다.

b) httpd.conf 화일

• ServerType standalone

서버의 형태는 독립형으로 지정하였는데, 독립형은 동시에 여러명의 클라이언트가 서비스를 요구해 올 때 demon 프로세스가 자신이 갖고 있는 정보를 복사하여 각각의 요구를 처리하므로 효율성이 높다.

• Port 80

본 시스템은 루트(root)의 권한으로 httpd 프로그램을 실행시키기 때문에 이미 예약되어 있는 80번 포트를 사용하였다.

• ServerRoot /usr/local/etc/httpd

서버가 설치되는 루트 디렉토리를 지정하였다.

• ServerName lis.yonsei.ac.kr

서버의 이름을 지정하는 것으로 본 시스템의 서버명은 lis.yonsei.ac.kr(IP주소는 165.132.86.53)이다.

c) srm.conf 화일

• DocumentRoot /usr/local/etc/httpd/htdocs

앞서 access.conf 화일에서처럼 서버가 제공할 각종 문서가 있게될 디렉토리를 지정하였다.

• UserDir .www

서버가 설치되어 있는 시스템에 계정을 갖고 있는 특정 사용자가 개인의 WWW문서들을 제공하는 개인 홈디렉토리를 만들고자 할 때 사용하는 디렉토리 이름을 .www로 지정하였다. 가령 jypark이라는 계정을 갖고 있는 시스템 이용자가 .www 디렉토리 아래에 개인 WWW 문서들을 두고 이를 URL로 지정하고자 할 때는 http://lis.yonsei.ac.kr/~jypark/화일명과 같이 하면 된다.

• DirectoryIndex index.html

특정화일을 지정하지 않고 디렉토리 이름만 지정했을때 자동적으로 제공해 줄 화일의 이름을 index.html로 지정하였다.

이렇게 환경을 설정하고 루트의 권한으로 httpd demon 프로그램을 실행시키면 서버가 설치되며, 시스템이 부팅될 때 자동적으로 프로그램이 수행되도록 하기 위해서 실행명령을 /etc/rc.local 화일에 추가시켰다.

4.1.1.2 홈페이지 작성

http://lis.yonsei.ac.kr을 URL로 보낼 경우 가장 먼저 지정되는 화일로 앞절에서 지정한 index.html 화일을 홈페이지로 설정하였다. 홈페이지에는 학과소개를 포함한 여러 정보원들이 있으며 이 중 한 모듈을 연세대학교 대학원 학위논문검색으로 하였다. 홈페이지는 HTML로 작성하였으며 이를 PC용 Netscape

로 불러들인 화면은 〈그림 3〉과 같다.

<그림 3> 본 학과의 홈페이지

4.1.2 데이터베이스 구축

본 시스템의 서지데이터베이스는 mSQL이라는 데이터베이스 시스템을 사용하여 구축하였다. mSQL은 SQL(Structured Query Language)문법의 대부분을 지원하며 C나 C++과 같은 프로그래밍언어와의 API(Application Programming Interface)도 가능한 관계형 데이터베이스시스템으로, David J. Hughes가 제작한 공개프로그램이다. mSQL은 네트웍환경을 지원하며 SunOS 4.1.1상에서 운용가능하므로 본 시스템과 호환성을 갖는다. mSQL도 무명씨 FTP를 이용하여 다운로드 받아 설치하였다.(URL = ftp://Bond.edu.au/pub/Minerva/msql//msql-1.0.6.tar.gz)

대상 서지데이터는 1993년 이후 발표된 연세대학교 본 대학원 학위논문의 서지사항으로, 연세대학교 중앙도서관 온라인목록시스템인 YOURS를 이용하여, 학문적 연관성을 갖는 문헌정보학과와 전산과학과의 89개 학위논문을 검색하여 구성하였다. 학과별로 보면 문헌정보학과가 35건, 전산과학과가 54건이며, 학위별로는 석사논문이 78건, 박사논문이 11건이다. 데이터의 입력은 UNIX의 한글 vi에디터를 사용하였다. 데이터베이스의 레코드구조는 <표 1>과 같다.

위와 같은 구조를 가진 레코드는 CGI를 통해 검색되어 WWW상에서 <그림 4>와 같은 HTML형식의 구조로 변환된다.

본 시스템은 관련 레코드를 연결시키는 링크로 '지도교수'와 '핵심되는 말'을 사용하였다. 학위논문이라는 데이터는 그 특성상, 대부분 한 저자가 하나의 논문만을 생산하기 때문에 저자를 링크로 사용하기는 어렵다. 반면 지도교수는 하나이상의 논문을 지도하고 있으며 연구주제 또한 연관성이 있는 분야이므로 관련레코드를 연결해주는 링크요소로 사용될 수 있다. 핵심되는 말은 논문의 저자가 직접 부여한

<표 1> 데이타베이스의 레코드 구조

테이블:THESIS (기본키:번호)

번 호	제 목	저자	지도교수	학 위	학 과	학위취득연도	핵심되는 말
숫자형	문자형	문자형	문자형	문자형	문자형	숫자형	문자형
1	대학도서관 온라인열람 목록의…	사공복회	정영미	박사	문헌정보학과	1994	온라인열람 목록, 온라인 목록, OPAC…
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

```
〈HTML〉
〈HEAD〉
〈TITLE〉전체 서지사항〈/TITLE〉
〈/HEAD〉
〈BODY〉
〈UL〉
〈LI〉번호:1
〈LI〉제목:대학도서관 온라인 열람목록의……
〈LI〉저자:사공복희
〈LI〉지도교수:〈A HREF="thesis.cgi?advisor+정영미"〉정영미〈/A〉
〈LI〉학위:박사
〈LI〉학위취득연도 : 1994
〈LI〉학과:문헌정보학과
〈LI〉핵심되는말  〈A HREF="thesis.cgi?keyword+온라인열람목록〉온라인열람목록〈/A〉,
                〈A HREF="thesis.cgi?keyword+온라인목록〉온라인목록〈/A〉,
                〈A HREF="thesis.cgi?keyword+OPAC〉OPAC〈/A〉, ……
〈/UL〉
〈/BODY〉
〈/HTML〉
```

〈그림 4〉 HTML형식의 레코드

것으로, 논문의 내용을 개략적으로 나타내주는 키워드로서의 기능을 하므로 중요한 링크정보가 된다. 사전 작업결과 89건의 논문에 수록된 키워드는 모두 412개이며, 저자가 부여한 키워드를 원칙적으로 그대로 사용하되, 2편이상의 논문에 출현한 키워드 중 띄어쓰기가 상이한 키워드만 수정하여 입력하였다. 모두 10건으로 그 내용은 다음과 같다(표 2).

〈표 2〉 수정된 키워드

번 호	수정 전	수정 후
1	B-ISDN	BISDN
2	객체 지향 기법	객체지향기법
3	기능 실체	기능실체
4	대학 도서관	대학도서관
5	순환 신경망	순환신경망
6	온라인 열람목록	온라인열람목록
7	정보 사서	정보사서
8	정보 요구	정보요구
9	정보 조사제공	정보조사제공
10	지리정보 시스템	지리정보시스템

4.1.3 CGI 프로그램

이용자가 WWW상에서 mSQL데이터베이스를 검색하고자 할 때 WWW서버와 데이터베이스를 연결해주는 인터페이스 프로그램이 필요한데, 이것이 바로 CGI프로그램이다. CGI 프로그램은 이용자의 요구에 따라 mSQL데이터베이스와 연결하여 질의에 적합한 데이터를 검색하고, 이를 HTML문서 형식으로 변환시켜 WWW서버에게 전달하는 역할을 한다.

본 시스템에서 CGI프로그램은 C언어를 사용하였으며, CGI프로그램에서의 데이터 조작

은 SQL문을 사용하였다. 데이터베이스 검색시 이용자는 직접 데이터베이스 서버에 접근하는 것이 아니라 CGI프로그램을 통해 접근하게 되므로, 일반 프로그램언어에 SQL문을 포함시켜 처리하는 형태인 내장형 SQL(ESQL : Embedded SQL)방식을 사용하였다.

데이터검색에 사용되는 SQL문은 기본적으로 select문을 사용하며 이용자의 요구에 따라 여러가지 검색조건이 추가된다. 예를 들어 브라우징 기능 중에서 제목별로 브라우징을 원하는 경우에는 다음과 같은 검색문이 사용된다.

```
select * from thesis order by title
```

위 검색문의 결과, thesis란 테이블에 있는 모든 데이터가 제목의 올림차순(ascending order)으로 정렬되어 검색된다. 복합필드검색의 경우에는 좀 더 복잡한 검색문이 필요하다.

```
select * from thesis where title like "%
        정보%" and
advisor = "정영미" order by title
```

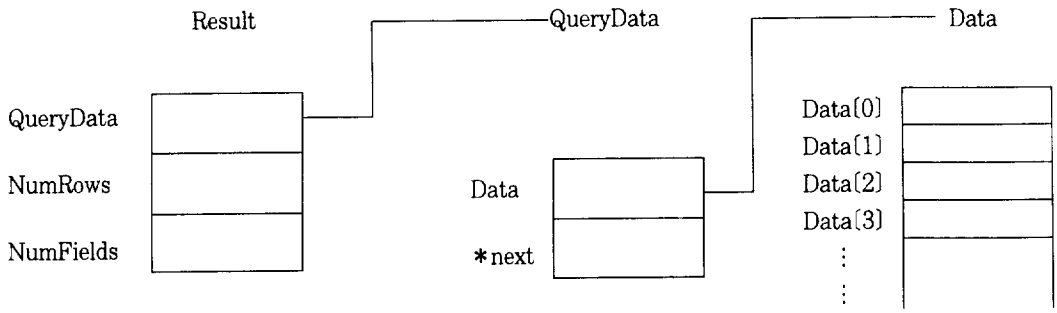
위 검색문은 'thesis'란 테이블에서 '정영미' 교수가 지도한 논문 중 제목에 '정보'란 단어가 들어 있는 모든 레코드를 제목의 올림차순으로 정렬하여 출력한다. 이러한 SQL검색문은 CGI프로그램에서 출력함수를 통해 mSQL 데이터베이스로 전달된다.

SQL질의에 따라 검색된 결과는 mSQL데이터베이스의 레코드구조가 아닌 CGI프로그램이 처리할 수 있는 적절한 구조로 변형되어야 하는데, 이 구조는 다음과 같다(그림 5).

검색결과는 Result라는 구조체(structure) 변수로 전달되는데, 이 변수에는 실제 데이터(QueryData)와 함께 검색결과와 일치하는 레코드의 수(NumRows), 질의에서 요구한 필드의 수(NumFields)가 저장된다. 실제 데이터를 담게 될 QueryData변수 역시 구조체변수로, 검색결과로 전달된 레코드의 수가 하나 이상일 경우를 위해 다음 레코드를 가리키는 포인터변수인 *next를 포함한다. 이러한 구조에 따라 mSQL데이터베이스에서의 레코드는 CGI프로그램에서 다음과 같이 표현된다.

Result -> QueryData -> Data[0]:번호(no)

Result -> QueryData -> Data[1]:제목



<그림 5> CGI프로그램에서의 데이터구조

(title)

Result -> QueryData -> Data[2]:저자 (author)

Result -> QueryData -> Data[3]:지도교수 (advisor)

이렇게 처리된 데이터를 이용자에게 전달할 때는, 화일의 앞부분에 "Content-type: text/html"을 포함시켜 먼저 화일의 형식을 WWW서버에게 알려주고, 실제 데이터와 HTML 태그를 함께 출력하도록 함으로써 이용자가 검색된 문헌을 HTML문서 형식으로 WWW브라우저를 통해 볼 수 있도록 한다.

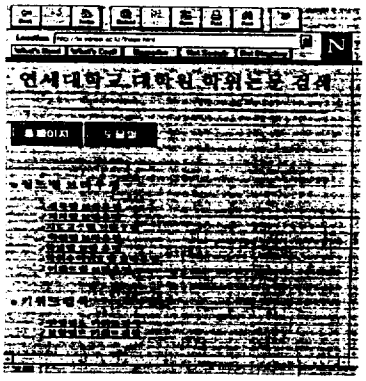
본 시스템의 검색기능 중 키워드 검색의 경우는 이용자가 직접 키보드를 통해 입력한 데이터를 처리해야 하는데, 이를 위해서는 HTML에서 지원하는 Form태그를 사용하였다. 그리고 이용자가 입력한 데이터를 CGI프로그램에 전달하는 방법으로는 POST를 사용하여 표준입력방식으로 데이터가 전달되도록 하였다. 다음은 단일필드 키워드검색에서 사용한 Form의 예이다(그림 6).

4.2 연세대학교 대학원 학위논문 검색모듈

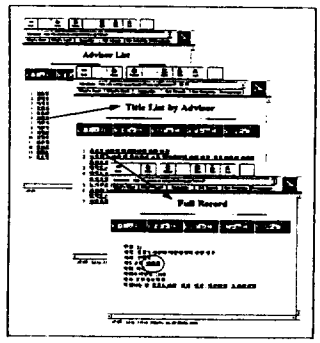
본 시스템의 데이터 검색기능은 크게 브라우징기능과 키워드 검색기능으로 나누어 볼 수 있는데, 두 기능은 분리된 것이 아니라 상호보완적 의미를 가지며 서로 혼합되어 사용된다. 검색 초기화면<그림 7>과 각 기능에 대한 자세한 설명은 다음과 같다.

```
<DL>
<DT><IMG SRC="/images/ball.gif">검색필드 선택<BR>
<DD>
        <INPUT TYPE=RADIO NAME=field CHECKED VALUE="title">제목<BR>
        <INPUT TYPE=RADIO NAME=field VALUE="author">저자<BR>
        <INPUT TYPE=RADIO NAME=field VALUE="advisor">지도교수<BR>
        <INPUT TYPE=RADIO NAME=field VALUE="degree">학위<BR>
        <INPUT TYPE=RADIO NAME=field VALUE="dept">학과<BR>
        <INPUT TYPE=RADIO NAME=field VALUE="year">학위수여년도<BR>
        <INPUT TYPE=RADIO NAME=field VALUE="keyword">핵심되는말<BR>
<DT><IMG SRC="/images/ball.gif">불(Boolean)연산자 선택<BR>
        <DD>
        <INPUT NAME=keyword1 SIZE=30><BR>
        <INPUT TYPE=RADIO NAME=operation CHECKED VALUE="and">AND<BR>
        <INPUT TYPE=RADIO NAME=operation VALUE="or">OR<BR>
        <INPUT TYPE=RADIO NAME=operation VALUE="none">NONE<BR>
        <INPUT NAME=keyword2 SIZE=30><BR>
</DL><BR><BR>
<CENTER><INPUT TYPE=SUBMIT VALUE="검색시작"><BR>
<INPUT TYPE=RESET VALUE="다시입력"></CENTER>
</FORM>
```

<그림 6> CGI에서 사용된 Form 형식의 예



〈그림 7〉 학위논문 검색 초기화면



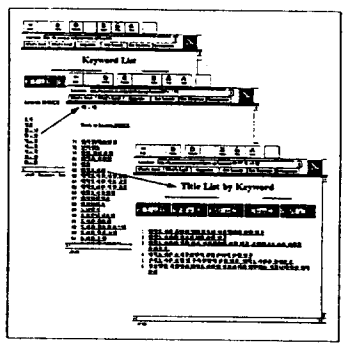
〈그림 8〉 지도교수별 브라우징 검색

4.2.1 브라우징 기능

하이퍼텍스트시스템의 가장 대표적인 기능은 브라우징이라 할 수 있는데, 본 시스템은 서지레코드의 각 필드별로 브라우징이 가능하다. 이러한 기능은 정보요구가 명확치 못하거나 접근방법에 익숙치 못한 이용자에게 다양한 접근점을 제시함으로써 정보접근을 용이하게 하고, 브라우징을 진행해 나감에 따라 적합한 정보를 찾는데 필요한 관련 개념을 획득하게 됨으로써 유용한 정보를 선별하는데 도움을 줄 수 있다. 또한 필드별 출력을 통해 정보공간을 축소시키는 효과도 얻을 수 있다.

각 리스트는 이용자가 선택한 필드별로 가나다순으로 정렬되어 출력되며, 출력된 리스트는 레코드별로 전체 서지사항과 곧바로 연결된다. 전체 서지사항을 확인한 후 다른 필드에 대한 계속적인 브라우징을 원할 경우 어느 화면에서라도 링크를 통해 곧바로 검색초기화면과 연결된다. 또한 브라우징 도중 키워드검색을 원하는 경우에도 해당 화면에서 바로 검색이 가능하다. 이러한 링크는 아이콘형태로 표현하여 이용자가 사용하기 편리하도록 하였다. 지도교수별 브라우징의 예를 들면 〈그림 8〉과 같다.

키워드별로 브라우징할 때는 HTML의 문서내(Intra-document)링크기능을 이용하였다. 현재 본 시스템에서 구축한 89건의 레코드로부터 총 412개의 키워드가 추출되었는데, 412개의 키워드를 한꺼번에 브라우징하기란 매우 어렵다. 특히, 데이터의 양이 점점 많아질 경우 이는 더욱 어려워지므로 키워드리스트의 화면을 색인형태로 구성하였다. 문서내 링크는 다른 화일로 연결되어 이동하는 것이 아니라 하나의 문서에서 지정해 놓은 해당 단어를 바로 찾아가도록 한다. 이러한 기능은 한 문서의 길이가 매우 길때 유용하다. 키워드별 브라우징의 화면은 〈그림 9〉와 같다.



〈그림 9〉 키워드별 브라우징 검색

4.2.2 키워드검색 기능

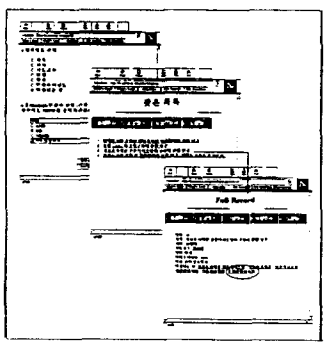
데이터가 계속적으로 추가되어 정보공간이 방대해지면 브라우징 기능만으로 적합한 정보를 찾는다는 것은 이용자의 정보접근에 대한 부담을 증가시키며, 때때로 이용자의 방향상실 문제를 야기시키기도 한다. 본 시스템에서는 이러한 문제를 보완하기 위하여 키워드검색 기능을 추가하였다.

먼저 이용자가 자신이 원하는 정보에 대한 검색어와 조건들을 입력하면 이를 충족시키는 레코드들의 제목들이 우선적으로 출력된다. 이렇게 부적합한 정보공간을 일차적으로 배제시킴으로써 탐색대상범위가 축소되며, 이용자는 탐색시간 및 경로를 줄일 수 있다. 이용자는 축소된 공간을 대상으로 계속 브라우징을 통해 원하는 정보에 접근해 나간다. 본 시스템에서 키워드검색 기능은 단일필드검색과 복합필드검색의 두가지로 나누어 진다.

4.2.2.1 단일필드검색

이 기능에서 이용자는 자신이 원하는 필드를 하나만 선정하여 검색어를 입력하게 되는데, 선정한 필드내에서는 불연산자를 사용하여 두개의 검색어를 조합할 수 있다. 선정가능한 필드는 레코드의 모든 필드이며 불연산자중 AND와 OR만 사용할 수 있다. 단일필드검색의 예를 들면 다음과 같다(그림 10 참고).

질문 핵심되는말에 'OPAC'이나 '온라인열람목록'을 포함하고 있는 논문을 찾아라.

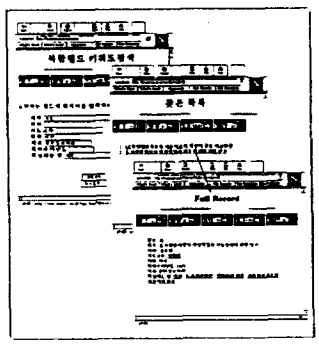


<그림 10> 단일필드 키워드 검색

4.2.2.2 복합필드검색

이 기능은 여러 필드를 동시에 검색조건으로 선정할 수 있는 기능으로, 보다 복잡하고 다양한 정보요구를 검색식을 작성하지 않고서도 쉽게 표현할 수 있다. 원하는 필드의 버튼을 클릭하고 해당 필드에 대한 검색어를 입력하는데, 선택한 각 필드는 모두 AND로 연결되며 키워드를 입력하지 않은 필드는 처리하지 않는다. 필드당 하나의 검색어만 사용할 수 있으며 불연산자를 사용할 수 없다. 다음과 같은 질문에 대한 검색의 예를 들면 <그림 11>과 같다.

질문 문헌정보학과 석사학위논문 중 제목과 핵심되는 말에 '색인'이 들어간 논문을 찾아라.



<그림 11> 복합필드 키워드검색

4.2.3 관련 레코드로의 이동

인간의 사고는 연상적이므로 이용자의 초기 정보요구는 브라우징 과정 중 관련있는 새로운 정보요구로 발전될 수 있다. 이러한 점을 반영한 것이 바로 하이퍼텍스트 시스템으로, 본 시스템은 학위논문이라는 데이터의 특성을 고려하여 지도교수와 키워드를 링크요소로 선정하여 관련레코드로 직접 이동할 수 있도록 구성하였다.

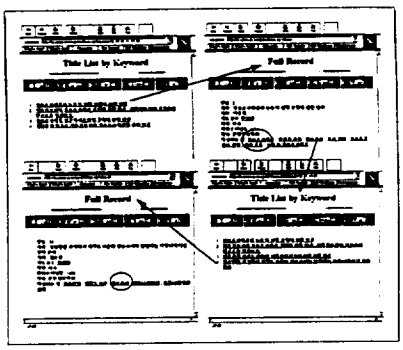
이용자는 브라우징이나 키워드검색을 통해 검색한 레코드의 브라우징 도중 관련있는 다른 레코드를 검색하고자 할 때, 초기검색화면으로 돌아가거나 새로운 키워드를 입력할 필요없이 제공된 링크를 통해 직접 이동한다. 예를 들어 〈그림 12〉와 같이 '정보조사제공' 이란 용어를 키워드로 갖는 레코드들을 브라우징하다가 이와 관련있는 '정보사서' 를 키워드로 갖는 레코드를 검색하고자 할 때, 해당 키워드를 클릭하면 '정보사서' 를 키워드로 갖는 레코드의 제목들이 출력되며 원하는 경우 전체 서지사항으로 연결된다. '정보사서' 를 선택했을 때, 전체 서지사항으로 곧바로 연결시키지 않고 해당 제목들을 우선 출력하는 것은 이용자가 계속 브라우징해 나갈 공간을 알려줌으로써 이용자에게 야기될 수 있는 방향상실문제를 보완하기 위해서이다.

4.2.4 브라우저의 다양한 기능 조합

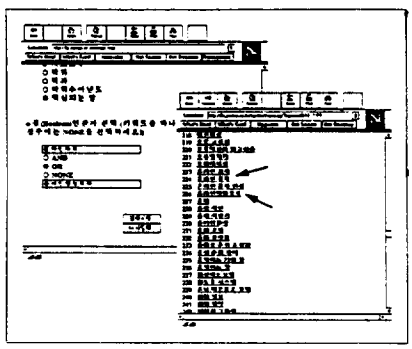
본 시스템은 이용자가 인터넷상에서 Netscape나 Mosaic과 같은 그래픽환경의 WWW 브라우저를 사용하는 것을 전제로 하여 구성하였기 때문에, 시스템 사용시 브라우저의 다양한 기능을 사용할 수 있어서 매우 유용하다. 본 시스템에서 테스트용 브라우저로 사용한 UNIX용 Netscape 1.1N 버전으로 살펴본 대표적인 기능들을 살펴보면 다음과 같다.

첫째, 여러개의 윈도우를 띄워 동시작업하는 것이 가능하다. 이러한 기능은 키워드 검색시 유용하게 사용될 수 있다. 본 시스템은 아직 시소러스가 구축되어 있지 않고 키워드 통제 기능이 없으므로, 키워드 검색시 다른 화면에서 키워드 리스트를 참조하면 이용자가 적절한 키워드를 선정, 입력하는데 도움을 받을 수 있다.

둘째, 검색결과를 다양한 형태로 저장할 수 있



〈그림 12〉 키워드를 이용한 관련 레코드로의 이동



〈그림 13〉 키워드 검색시 참조가능한 키워드별 브라우징 화면

다. 일반 텍스트 형태나 HTML형식으로도 가능하며 포스트스크립 형태로도 저장할 수 있다. 검색결과를 전자우편을 통해 바로 전송할 수도 있는데, 이 경우 검색결과화일과 함께 다른 로컬화일도 함께 원하는 상대방에게 보낼 수 있다.

셋째, 현재의 문서상에서 원하는 단어를 찾아볼 수 있다. 단어를 입력하면 해당단어가 하이라이팅되며, 이러한 과정을 계속 반복할 수 있다. 길이가 매우 긴 문서의 경우 한 윈도우에서 전체 문서를 한번에 볼 수 없으므로 이러한 찾기 기능을 이용하면 손쉽게 원하는 위치로 이동할 수 있다.

넷째, 검색결과를 편집할 수 있다. 검색결과의 일부분을 복사하여 다른 검색결과와 조합할 수 있으며, 일반 화일도 함께 편집할 수도 있다.

다섯째, Back버튼과 Forward버튼을 제공하므로 바로 이전, 이후의 화일로 이동할 수 있으며, 이용자가 참조해 왔던 화일들에 대한 히스토리정보를 소장하고 있어서 원하는 이전화일로 곧바로 이동가능하다.

여섯째, 검색결과를 출력할 수 있다. 일반 문서편집기와 같은 다양한 옵션들을 제공하므로 출력이 매우 용이하며, 화일로 출력할 경우 저장기능과 같은 효과를 가져올 수 있다.

이외에 인터넷상에서 사용할 수 있는 여러가지 기능을 모두 복합적으로 사용할 수 있으므로, 더욱 융통성있는 검색이 가능하다.

5. 시스템 평가 및 응용

하이퍼텍스트 시스템은 링크조직을 통해 인간의 사고과정과 유사한 연관적이며 비선형적인 정보접근 방법을 제공하며, 기존의 전통적인 정보검색시스템에 비해 브라우징 기능이 우수하고 관련정보로의 연결이 용이하다는 평가를 받고 있다. 이러한 하이퍼텍스트 시스템은 계속적으로 발전하여 멀티미디어 데이터를 수용하기에 이르렀고, 인터넷의 정보검색에도 하이퍼텍스트의 개념을 적용한 WWW(World Wide Web)란 정보검색도구가 등장하게 되었으며, WWW의 사용은 가히 폭발적이라 할 만큼 그 이용이 증가하고 있는 추세이다. 그러나 하이퍼텍스트 시스템 역시 문제점을 안고 있는데, 정보공간이 방대해지면 브라우징만으로 원하는 정보에 접근하기 어렵고 때때로 이용자의 방향상실문제를 불러 일으키기도 한다.

이에 본 연구는 서지데이터를 대상으로 하여 하이퍼텍스트 시스템이 구조화된 서지데이터의 브라우징 및 검색에 유용함을 보여주었고, 이용자의 방향상실문제를 보완하기 위해 키워드 검색기법을 적용하여 보다 효과적인 검색이 가능하도록 하였다. 또한 본 시스템을 WWW를 이용하여 구현함으로써 얻을 수 있는 장점과 가능성을 제시하였다.

본 연구에서 구축한 하이퍼텍스트시스템의 장·단점을 살펴보고, 이에 따른 앞으로의 응용방향을 제시해보면 다음과 같다.

첫째, 인터넷의 정보탐색도구의 하나인 WWW를 이용하여 시스템을 구현함으로써 클라이언트/서버 모형의 네트웍 환경이 지원된다. 따라서 인터넷에 접속가능하고 WWW브라우저를 사용할 수 있는 이용자들은 누구나 본 시스템을 이용할 수 있다. 또한 WWW는 인터넷의 대부분의 서비스들과 인터페이스가 가능하므로 이를 이용하면 더욱 융통성있는 시스템

을 구축할 수 있다. 예를 들면, WAIS를 연결할 경우 WAIS의 자동색인 기능과 키워드검색기능을 이용할 수 있으므로 손쉽게 검색시스템을 구축할 수 있다. 또한 FTP를 활용해 볼 수 있는데, 본 시스템은 데이터베이스시스템을 사용하였고 서지사항만을 대상으로 하였기 때문에 문헌의 전문(full-text)을 데이터베이스에 저장하기가 어렵다. 이러한 경우 대상문헌의 전문은 일반 텍스트나 포스트스크립형태로 별도의 화일시스템에 저장하여, 이용자가 원할 경우 FTP서버와 연결하여 다운로드받을 수 있도록 서비스하는 것도 방법이 될 수 있다.

둘째, 본 시스템은 mSQL이라는 데이터베이스시스템에 서지데이터베이스를 구축하고 이를 WWW와 연결하였다. 따라서 데이터 검색 · 갱신 등의 기본적인 데이터처리작업외에, 동시성제어, 데이터복구, 보안, 무결성, 데이터독립 등과 같은 데이터베이스의 장점을 그대로 이용할 수 있으므로, 시스템 구현시 데이터관리를 위한 별도의 응용작업이 필요없다. mSQL이외에도 Informix나 Oracle등과 같은 SQL인터페이스를 가진 서버용 데이터베이스시스템은 CGI를 통해 적절한 인터페이스를 제공하면 기존의 데이터를 그대로 이용할 수 있다. 이를 응용하여 도서관 OPAC과 WWW의 연결을 고려해 볼 수도 있다. 현재 대학도서관이나 연구도서관에서 인터넷을 통해 도서관에 접근할 수 있도록 하려는 움직임이 활발한데, 대부분이 Telnet을 통한 원격접속만 가능하다. 이를 HTTP를 통해 WWW상에서 제공한다면 이용자편의의 인터페이스를 제공할 수 있을 것이다.

셋째, 본 시스템은 WWW 브라우저를 데이터브라우저로 이용하고, 아이콘을 이용해 기능버튼을 구성하여, 그래픽환경의 이용자 인터페이스를 제공한다. 따라서 시스템을 처음 접하는 이용자도 쉽게 사용할 수 있고, 기존에 WWW를 이용해 본 이용자라면 동일한 인터페이스환경을 제공하므로 더욱 용이하다. 시스템 구현시에도 윈도우 화면구성을 위한 별도의 프로그래밍작업이 필요없으므로 시스템 개발에 드는 시간과 비용도 절감할 수 있다.

넷째, 본 시스템은 WWW상에서 구현되었기 때문에 WWW의 가장 큰 장점으로 인정받고 있는 멀티미디어 데이터 처리가 가능하다. 현재는 데이터베이스시스템을 이용하여 텍스트만을 대상으로 구성하였지만 추후 멀티미디어 데이터와의 연결도 가능하다. 다시 말하면, 텍스트형태의 데이터는 그대로 데이터베이스시스템에 저장하여 관리하고, 이미지나 소리, 동화상과 같은 데이터는 WWW가 처리할 수 있는 형태, 예를 들면 JPEG, MPEG와 같은 표준형태로 압축하여 별도의 화일시스템에 저장한다. 그리고 인터페이스 프로그램을 통해 관련있는 데이터끼리 서로 연결해 준다.

다섯째, 본 시스템은 서지데이터를 하이퍼텍스트화하여 하이퍼텍스트시스템이 구조화된 서지데이터의 브라우징 및 검색에 유용함을 보여주었다. 본 시스템에서는 지도교수와 키워드를 링크로 사용하였는데, 지도교수 링크를 통해 한 지도교수의 관련 연구들을 살펴볼 수 있고, 키워드 링크를 통해서는 관련 레코드뿐만 아니라, 관련있는 새로운 주제로 직접 이동하는 것도 가능하다. 본 시스템에서 선정한 링크이외에 색인작업을 통해 제목에 나타난 주제어를 링크시킬 수도 있으며, 일반 연구논문의 경우에는 저자나 학술잡지명도 링크요소가 될 수

있다. 특히, 학술잡지의 각 권(volum)이나 호(no)의 논문을 차례로 링크시키면 목차를 브라우징하는 것과 같은 효과도 가져올 수 있다.

여섯째, 본 시스템은 서지레코드의 각 필드별로 브라우징기능을 제공한다. 이러한 기능은 정보요구가 명확치 못하거나 접근 방법에 익숙치 못한 이용자에게 다양한 접근점을 제시함으로써 정보접근을 용이하게 한다. 또한 필드별로 데이터를 모아줌으로써 탐색해 나갈 정보공간을 축소시켜주는 효과도 얻을 수 있다.

일곱째, 본 시스템은 하이퍼텍스트시스템에서 발생할 수 있는 가장 큰 문제점인 방향상실을 보완하기 위해 키워드 입력을 통한 검색기능을 제공한다. 이러한 기능은 이용자의 정보요구를 표현한 검색어와 조건들을 만족시키는 레코드들을 대상으로 탐색할 수 있도록 해줌으로써, 일차적으로 부적합한 공간을 배제시켜 보다 효과적인 검색이 가능하도록 한다. 또한 이용자는 탐색에 소요되는 시간 및 경로를 줄일 수 있다.

그러나, 본 시스템은 키워드 검색을 위해 별도의 색인작업을 하지 않고 mSQL 데이터베이스에서 제공하는 필드별 색인기능을 이용했으므로, 부정확하고 불충분한 입력에 대한 적절한 통제기능을 갖고 있지 못하다. 키워드별 브라우징을 통해 부분적으로나마 보완이 가능하지만 좀 더 효과적인 검색을 위해서는 별도의 색인작업이나 시소러스구축이 필요하며, 이를 동일한 인터페이스로 이용자에게 제공하기 위해서는 색인브라우저나 시소러스브라우저와 같은 도구를 개발해야 한다. 이러한 브라우저는 용어간의 관계를 그래픽을 이용하여 보여주며, 브라우저를 통해 검색식을 작성할 수 있도록 하고, 또한 이미지와 텍스트를 이용해 검색어를 설명해 주는 등의 여러가지 기능이 포함되어야 한다.

여덟째, 본 시스템은 복합필드검색이 가능하며 불연산자를 사용할 수도 있다. 하이퍼텍스트에 키워드검색 기법을 적용한 대부분의 시스템이 전문색인을 이용한 초보적인 검색기능을 제공하는데 반해, 본 시스템은 하나이상의 필드를 선택하여 각 필드별로 원하는 키워드를 입력할 수 있으며, 하나의 필드내에서는 불연산자를 사용함으로써 보다 복잡하고 다양한 정보요구를 표현할 수 있다.

정보검색시스템에 관한 연구는 점점 더 이용자와 시스템간의 상호작용을 개선하려는 방향으로 진행되어 가고 있으며, 정보의 소유에서 정보로의 접근개념으로 변화하고 있는 현 시대는 네트웍의 위력을 실감하게 하고 있다. 본 연구를 통해 구현된 시스템은 이러한 변화를 수용할 수 있는 융통성있는 시스템이며, 현재의 도서관이 '디지털 도서관(digital library)' 혹은 '가상도서관(virtual library)' 형태로 나아가는데 있어서 하나의 방법 및 가능성을 제시해 줄 것으로 기대한다.

참 고 문 헌

고영곤. 1993. 탐색과 브라우징을 지원하는 하이퍼미디어 시스템의 설계 및 구현. 연세대학교 대학원 전산과학과 석사학위논문.

고영곤 등. 1995. "하이퍼미디어 시스템과 정보검색" 정보과학회지 13(1):65-77.

김명철 등. 1994. "하이퍼미디어를 이용한 문화예술 정보검색" 한국정보관리학회 전국논문대회논문집(제1회):11-14.

김영귀. 1990. "탐구적 검색기법으로서의 브라우징" 도서관학논집 17:111-141.

김용운. 1995. "WWW의 소개와 이용 및 설치" 코리아네트 '95 특강자료집: 583-615.

문성빈. 1995. "인터넷을 이용한 전자도서관 구현" 코리아네트 '95 특강자료집: 459-473.

유사라. 1995. "W3 가상도서관 활용을 위한 HTML 문서작성과 이미지/사운드 처리:45-66.

윤정기. 1994. 추론기능을 이용한 하이퍼텍스트 기반 검색시스템의 설계에 관한 연구. 중앙 대학교 대학원 박사학위논문.

이윤홍. 1995. "DBMS와 WWW" 코리아네트 '95 발표자료집:391-400.

정영미. 1989. "하이퍼텍스트의 개념과 응용에 관한 고찰" 정보관리학회지 6(2):3-20.

Aboud M. et al. 1993. "Querying a Hypertext Information Retrieval System by the Use of Classification" Information Processing and Management. 29(3):387-396.

Alexandra D. and Dietmar W. 1993. "Design Issues in a Hypertext-based Information System for Bibliographic Retrieval" Proceedings of the 56th ASIS Annual Meeting. 30 (October):191-198.

Andonoff E. et al. 1992. "A Hypertext Approach for Manipulation Object-oriented Database" Information Processing & Management. 28(5):567-580.

Bruse R. Schatz and Joseph B. Hardin. 1994. "NCSA Mosaic and the World Wide We: Global Hypermedia Protocols for the Internet" Science. 265:895-901.

Bruce W. Croft and Howard R. Turtle. 1993. "Retrieval Strategies for Hypertext" Information Processing & Management. 29(3):313-324.

Byte Count by Service. http://www.cc.gatech.edu/gvu/stats/NSF/Extrap.GIF.

Conklin J. 1987. "Hypertext : An Introduction and Survey" IEEE Computer. Sep.:17-41.

David E. et al. 1994. "On the Creation of Hypertext Links in Full-text Documents : Measurement of Inter-linker Consistency" Journal of Documentation. 50(2):67-98.

Dunlop M. D. and C. J. van Rijsbergen. 1993. "Hypermedia and Free Text Retrieval" Information Processing & Management. 29(3):287-298.

Emily B. and Joseph D. 1991. Hypertext/hypermedia handbook : McGraw-Hill.

Eva Bertha. 1993. "Inter- and Intrabibli-

ographical Relationships : A Concept for a Hypercatalog" Opportunity 2000: Understanding and Serving Users in an Electronic Library : 212-223.

Foss C. L.. 1989. "Tools for Reading and Browsing Hypertext" Information Processing & Management. 25(4):407-418.

Frisse M. E. 1988. "Searching for Information in a Hypertext Medical Handbook" Communications of the ACM. 31(7):880-886.

Gregory B. Newby. 1992. "An Investigation of the Role of Navigation for Information Retrieval" Proceedings of the 55th ASIS Annual Meeting. 22:20-25.

Hans C. et al. "Concept-based Retrieval of Hypermedia Information : from Term Indexing to Semantic Hyperindexing" Information Processing & Management. 29(3):373-386.

Kinnell S. K. 1988. "Information Retrieval in the Humanities using Hypertext" Online. 12(2):34-35.

Lucarella D. and Zanzi A. 1993. "Information Retrieval from Hypertext : an Approach using Plausible Inference", Information Processing & Management. 29(3):299-312.

Marchionini G. and Schneiderman B. 1988. "Finding Facts vs. Browsing Knowledge in Hypertext Systems" Computer. 21(1):70-80.

Maristella Agosti. 1993. "Hypertext and Information Retrieval" Information Processing Management. 29(3): 283-285.

Mark F. Frisse and Steve B. Cousins. 1992. "Models for Hypertext" Journal of the Americal Society for Information Science. 43(24):183-191.

MIME. http://www.oac.uci.edu/indiv/MIME/MIME.html

Nielson J. 1990. "The Art of Navigating in Hypertext" Communication of the ACM. 33(3):296-310.

Nigel Woodhead. 1991. Hypertext and Hypermedia: Theory and Applications. Wilmslow: Sigma Press.

Richard P. 1993. "A Hypertext-based Thesaurus and a Subject Browsing Aid for Bibliographic Databases" Information Processing & Management. 29(3):345-357.

Roy R. et al. 1993. "Retrieval Hierarchies in Hypertext" Information Processing & Management. 29(3): 359-371.

The Common Gateway Interface. http:hoohoo.ncsa.uiuc.edu/cgi/primer.html.

Welsch E. K. 1992. "Hypertext, Hypermedia and the Humanities" Library Trends. 40(4):614-646.