

□ 특집 □

# World Wide Web(WWW)

김 종 근<sup>†</sup> 임 인택<sup>††</sup>

◆ 목 차 ◆

- |           |                |
|-----------|----------------|
| 1 서론      | 5 홈 페이지의 설계    |
| 2 현재 및 특징 | 6 WWW의 과제 및 장래 |
| 3 역사      | 7 결론           |
| 4. 관련기술   |                |

## 1. 서론

데이터 통신이 컴퓨터 사용 환경에서 중요한 기능이 되면서 의학분야, 공학분야, 예술분야, 사회/과학분야, 정치/경제분야 등의 여러 분야에서 데이터 통신망을 이용하여 다양한 정보를 수집하고 정보를 공유하게 되었다. 이러한 통신망은 특정한 하드웨어 및 소프트웨어 기술을 이용하여 구축된다. 그러나 하나의 특정한 망 구성 기술이 모든 사용자들의 요구를 만족시킬 수는 없기 때문에 서로 다른 하드웨어 구성을 가지는 컴퓨터들로 이루어진 망, 또는 서로 다른 운영체제를 사용하는 망들을 서로 연결시켜 그것을 하나의 통일된 망처럼 사용하게 하는 기술을 망의 연동(Internetworking)이라고 하며, 네트워크의 네트워크라고 할 수 있는 인터넷(Internet)이 그

좋은 예라 할 수 있다.

인터넷은 '정보의 바다'라고 불리울 정도로 많은 유용한 정보들이 있으며, 이러한 정보들을 이용하기 위한 여러가지기술이 개발되고 다양한 서비스가 제공된다. 그 중에는 1992년에 개발되어진 World Wide Web이라는 GUI(Graphic User Interface) 형태의 정보 서비스가 구현됨에 따라 인터넷에서 하이퍼텍스트 형태의 정보검색 뿐만 아니라 사운드, 그래픽, 동화상 등의 멀티미디어 정보들도 제공하게 되었다.

본고는 World Wide Web(WWW)에 대해서 그 특징, 역사, 기술, 홈 페이지의 설계, 그리고 연구 과제 및 장래에 대해 해설한다.

WWW의 해설은 이미 많이 나와 있다. 또한 그에 대한 정보도 상당히 많이 있으며, 관련되는 문헌도 많이 소개되어 있다. 따라서 WWW의 일반적인 구조에 대한 해설은 최소한으로 줄이고 있기 때문에

<sup>†</sup> 정회원 . 영남대학교 전산공학과 부교수  
<sup>††</sup> 준회원 : 영남대학교 멀티미디어 정보통신학과 석사과정

상세한 정보를 얻기 위해서는 참고문헌을 참조해 주기 바란다. 또한 이 문서중의 현재는 1996년 3월 1일을 의미하고 있다.

## 2. 현재 및 특징

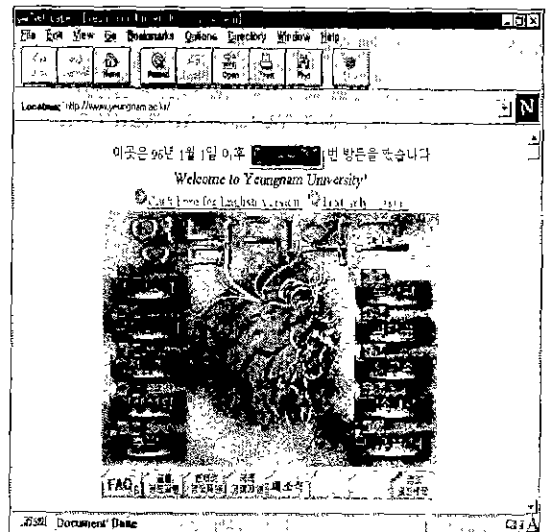
최근 인터넷(Internet)에서 그 중심적인 역할을 하고 있는 WWW의 인기는 날이 갈수록 높아지고 있다. 컴퓨터관련 잡지는 물론이고, 신문이나 TV, 하물며 여행안내 잡지에 까지 인터넷 및 WWW에 대한 특집을 싣고 있다. 물론 이러한 열기가 단순히 유행만은 아니며, 학술관련 연구자들만의 네트워크였던 인터넷과 WWW가 착실하게 일반에게도 보급되고 있다. 또한 인터넷의 접속 서비스를 제공하는 NSP(Network Service Provider) 수도 착실하게 증가하고 있다. WWW의 홈 페이지를 가지는 것도 이전에는 컴퓨터 관련 대학의 학과나 기업이었으나, 일반 대학의 학과 및 기업 그리고 개인으로까지 그 범위가 확대되었다. WWW는 이와 같이 성장을 계속하고 있고, 동시에 관련되는 환경에도 큰 영향을 주고 있다.

그러면 WWW라는 것은 무엇인가. 간단히 설명하면, WWW는 1989년에 CERN(Organization European Research Nuclear)의 Tim Berners-Lee 등에 의해 제안된 하이퍼텍스트 개념에 기반을 가지는 분산형 정보시스템이다. 주요한 특징은 다음과 같다.

- 인터넷에 산재해 있는 정보를 하이퍼텍스트개념으로 정리한다.
- 하이퍼텍스트 정보를 취급하기 위한 수단을 제공한다.
- 다양한 자원을 지정하기 위해 통일된 접근 수단을 제공한다.

다음에는 WWW의 표면적인 특징, 즉 WWW에

대해 사용자측에서 본 특징을 WWW를 조작하는 실제 환경에 맞추어 소개한다. WWW의 표면적인 주요한 특징은 netsurf의 감각과, 멀티미디어 취급과, 개방성에 있다고 할 수 있다. 그림 1은 영남대학교의 홈페이지(URL은 <http://www.yeungnam.ac.kr>)에 WWW클라이언트인 Netscape로 접속한 상태를 보인다.



[그림 1] 영남대학교의 홈페이지를 접속한 상태

이 중에서 그림 속의 팻말의 글자위나 그림 밑에 있는 문자열을 마우스로 클릭하면, 그 문자열에 관계있는 정보가 표시된다. 예를 들면, 학교소개를 클릭하면, 영남대학교 소개의 페이지가 표시된다. 이들 문자열이 표시되어 있는 부분이 다른 정보로 '링크'가 되어 있다. 링크를 따라간 정보에 또 다른 정보를 연결시켜 주는 링크가 있다면, 그 링크를 선택하여 정보로부터 정보로 계속 찾아 갈 수 있다. 이와같이 인터넷이라는 '정보의 바다'를 서핑(surfing)하고 있는 것과 같은 감각을 'netsurf'라고 부르는 것이다. 링크된 정보는 반드시 WWW 서버상의 HTML 문서[8, 11]가 아니라도 좋다. 음성, 정지화, 동화, 그래

프, MIDI데이터 등 문서 이외의 데이터와 Gopher, FTP, WAIS 등 네트워크 상의 다른 서비스를 지정할 수도 있다. 사용자측에 각종 데이터에 대응하는 뷰어(Viewer)를 준비하는 것으로, 이들 모든 자원을 WWW라는 통일된 세계에서 취급할 수 있는 것이다. 마우스를 한번 클릭하여 음악이 시작되기도 하고, 영화가 시작되기도 하는 그야말로 ‘멀티미디어’ 세계인 것이다. 또한 WWW서버로의 접속은 그 플랫폼을 가리지 않는다. Unix Workstation, PC, Macintosh 등 어떤 컴퓨터용으로든 WWW의 클라이언트 프로그램이 개발되어 있다. 이러한 WWW 클라이언트 프로그램에는 일리노이 대학의 NCSA(National Center for Supercomputing Applications)에서 개발한 Mosaic 및 Netscape Communications사의 Netscape가 많이 사용되고 있다. 또한 문자 베이스의 WWW 클라이언트(Lynx 등)도 많기 때문에 상황에 따라서 편리하게 사용할 수 있다.

이상 netsurf 감각과 멀티미디어 취급, 개방성 등의 WWW의 표면적인 특징에 더하여 다음에 설명하는 WWW의 강력한 기능에 의해, 종래에 특정 컴퓨터내에 갇혀있던 다양한 정보를 간단하게 인터넷에 제공할 수 있게 되었다. 이러한 요소들이 상호작용하여 WWW의 급속한 발전으로 연결된 것이다.

### 3. 역 사

WWW의 역사는 다음과 같이 3개의 기간으로 분류된다. 먼저 1989년에 CERN의 Tim Berners-Lee가 WWW프로젝트를 제안한 후, NCSA가 X윈도우용 Mosaic 1.0을 발표할 때까지인 1993년 6월까지를 여명기라 부른다. 1993년 6월부터 NCSA의 개발중심 멤버들이 빠져나와 지금의 Netscape Communications사가 설립되는 1994년 4월까지를

팽창기라 부른다. 그리고 1994년 4월 이후를 전국시대라고 부른다.[12] 이상과 같은 발전 분류에 따라 WWW 발전 역사를 살펴본다.

WWW의 여명기에는 CERN이 WWW 발전의 중심적인 존재였다. 물론 WWW 이전에도 하이퍼텍스트를 사용하는 시스템은 있었지만, 대부분 연구 수준에 머물러 있었다. WWW가 발전한 이유는 전 세계가 마침 인터넷으로 연결되기 시작하는 좋은 시기였던 것과 함께, CERN에서의 WWW 프로젝트가 다음과 같이 구체적이고도 명확한 목적을 가지고 있었기 때문이다.

CERN과 같이 가속기를 사용하여 고에너지물리를 연구하는 분야에서는, 전 세계에서 연구자들이 모여 실험하고, 실험이 끝나면 연구자들은 전세계로 흩어져 버렸다. 이러한 환경에서 전세계에 분산되어 있는 연구자들 간의 공동연구를 원활하게 추진하는 기능으로 WWW 프로젝트가 추진되었던 것이다.

WWW는 착실하게 성장을 계속하였다. 그러나 여명기에는 문자중심의 WWW 클라이언트가 중심이었다. telnet으로 'info.cern.ch'에 접속하여 CERN에서 사용하고 있던 문자중심의 WWW 클라이언트를 사용하는 것이 일반적이었다.

1992년 가을에는 NCSA에서 WWW를 주목하고, Mosaic개발을 시작하였다. NCSA는 미국내에 5개 있는 슈퍼컴퓨터센터의 하나로, 원격지로부터 슈퍼컴퓨터를 사용하도록 서비스를 제공하는 기관으로 CERN과는 매우 비슷한 입장이었다. WWW는 그러한 목적에 잘 맞았기 때문에 Mosaic을 개발하게 된 것이다. 그래서 1993년 6월 Mosaic 최초의 정식 버전인 X윈도우용 mosaic 1.0이 발표되고 팽창기에 들어간다.

Mosaic은 그래픽 인터페이스와 광범위한 UNIX 워크스테이션의 플랫폼을 지원했기 때문에, 단숨에 인터넷의 사용자들에게 확산되었다. 또한 1993년 가

올에는 Windows판, Macintosh판을 발표하였다. 이 때에는 인터넷의 주요 사용자 뿐만 아니라, 일반인들에게도 주목을 받기 시작하였다. 이때부터 WWW의 세계는 급속하게 팽창하기 시작하였다.

Mosaic이 나오기 부서는 WWW의 발전 중심이 NCSA로 옮겨졌다. 물론 CERN도 중심 역할을 하였지만, NCSA는 계속하여 새로운 버전의 Mosaic을 발표하여, 이 당시의 WWW 발전의 원동력이 되었다.

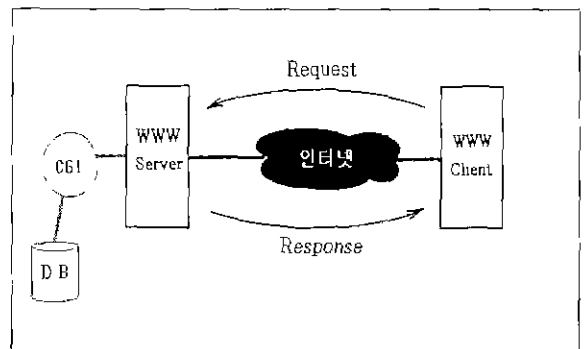
그러나, Netscape Communications사에 개발 중심팀버를 빼앗긴 1994년 봄 이후, NCSA는 그 구심력을 조금씩 잃어가게 되었다. Mosaic의 지원을 계속하고는 있지만, 새로운 소프트웨어가 나오는 페이스가 급격하게 떨어졌다. 역시 이러한 기술을 리드하기 위해서는 새로운 기술을 재빨리 제품화하는 개발력이 필요하였기 때문이다. NCSA가 구심력을 잃어버린 이후, WWW에 관해서는 여러가지 움직임이 있어, 전국시대와 같은 상황에 와있다.

현재로는 CERN과 MIT가 1994년 10월에 설립한 W3C(World Wide Web Consortium)[2,3]와 IETF(Internet Engineering Task Force)[1]가 지금부터의 WWW 발전 방향을 움직일 것으로 보고있다. 참고로 W3C의 디렉터는 CERN에서 최초로 WWW를 제안한 Tim Berners-Lee이고, 이 조직은 X 컨소시움을 모델로 하고있다. IETF는 인터넷의 표준화에 관한 중심적인 역할을 하고 있다. HTTP[9,10], HTML, URI[4,5] 등의 표준화 작업을 진행하고 있다. 이외에 기업으로는 Netscape Communications사 등이 WWW에 깊은 관계를 가지고 있다.

## 4. 관련 기술

### 4.1 WWW 기능

WWW의 기능은 그 중심이 URL, HTTP, HTML, CGI 등에 의해 형성된다. WWW는 이러한 강력한 기능의 통합에 의해 인터넷상에 있는 정보를 공유할 수 있도록 한다. 본 절에서는 WWW의 기능 전체를 살펴본 후, 개별 요소 기술에 대해 설명한다. CGI에 대해서는 WWW의 동적인 측면을 강조한다.



[그림 2] WWW 시스템의 구성

URL은 인터넷상의 다양한 자원을 통일된 방법으로 표현한다. HTTP는 WWW의 클라이언트와 서버가 정보를 주고 받기 위한 프로토콜이다. HTML은 하이퍼텍스트를 표현하기 위한 언어이다. WWW시스템의 전체 구성은 그림 2와 같이 된다. WWW 클라이언트는 URL에 의해 서버로 접속하고, HTTP를 사용하여, 지정된 정보인 HTML문서를 획득한 후, 화면에 표시한다. 그 화면상에서 사용자가 링크를 선택하면, 그 사용자의 액션은 URL이라는 형태로 WWW 클라이언트에 전달되고, 계속 지정된 서버와의 대화를 유발시킨다. 사용자에 의해 선택된 URL이 지시하는, 정보가 다른 서비스의 텍스트계의 데이터인 경우에는 그들 서버가 사용하는 통신 프로토콜(FTP서버라면 FTP)를 사용하여 정보를 획득한다. 정보는 그 서비스 고유의 형식을 가지고 있지만, HTML로 변환되어 화면에 표시된다.

URL이 지시하는 정보가 화상 및 음성 등의 이진

데이터인 경우에는, 그 데이터를 출력할 수 있는 외부 뷰어에 넘겨주어 기동시킨다. 이러한 기술에 의해 사용자가 링크를 선택하면 화상 및 음악이 시작되는 것이다.

일부의 서비스는 게이트웨이에 의해 제공된다. 실제의 정보는 데이터베이스 등의 다른 서버에 격납되어 있지만, 문의와 응답을 게이트웨이에서 중계하는 기술이다. 이 기능을 CGI (Common Gateway Interface)라 부른다. 클라이언트측에서는 HTTP로 액세스할 수 있기 때문에, 사용자에게는 WWW내에 통합된 서비스처럼 보인다.

1) URL(Uniform Resource Locator)

URL은 인터넷에 있는 수많은 자원을 통일된 표현법으로 기술하는 방법이다. URL은 자원을 지정하기 위해 호스트명, 파일명, 액세스시에 사용하는 프로토콜, 액세스시의 명령어 등의 요소를 포함하는 표현형식으로 다음과 같다.

```
scheme://host.domain[:port]/path[?key]
```

scheme부에는 http, gopher, ftp, wais, nntp, telnet, file 등 다양한 프로토콜을 지정할 수가 있다. 구체적인 URL 기술 예를 다음에 보인다.

- WWW서버 www.ncsa.uiuc.edu에 있는 홈페이지 index.html를 가리키는 URL:  
http://www.ncsa.uiuc.edu/index.html
- FTP서버 ftp.yeungnam.ac.kr에 있는 파일 /pub/ls-IR.Z를 가리키는 URL:  
ftp://ftp.yeungnam.ac.kr/pub/ls-IR.Z

2) HTTP(HyperText Transfer Protocol)

WWW에는 빠르고 효율적인 하이퍼텍스트 정보의 전송을 위한 클라이언트-서버 프로토콜을 HTTP로 규정하고 있다. 하이퍼텍스트는 이 프로토콜에

따라 수행되는데, 기본적으로는 클라이언트가 서버에 요구(request)를 송신하고, 그것에 대해 서버가 클라이언트에 대해 응답(response)을 돌려주는 형식으로 되어있다. 이러한 요구 메시지(request message)는 몇 개의 요구 헤더 필드(request header field)를 포함하는 요구 헤더(request header)로 구성된다. 각 요구 헤더 필드는 CRLF(carriage-return line-feed) 문자에 의해 종료되는 한 텍스트 라인이다. 요구(request)는 다음과 같은 형식을 가진다.

```
GET /index.html HTTP/1.0
Accept: text/plain
Accept: application/x-html
Accept: application/html
.
.
Accept: text/x-setext
Accept: */*
User-Agent: NCSA Mosaic for X .....
[a blank line, containing only CRLF]
```

요구의 첫라인은 메소드 필드(method field)로써 다음과 같은 형식을 갖는다.

```
HTTP_method identifier HTTP_version
```

위의 예에서 처음의 GET이 메소드라 불리는 것으로 대상이 되는 정보에 대해 수행할 처리를 나타낸다. 다른 메소드로는 HEAD, PUT, POST, DELETE 등이 있다. 두 번째의 'index.html'은 클라이언트가 서버에게 요구하는 정보를 나타내는 URL이다. 마지막의 'HTTP/ 1.0'은 클라이언트에서 사용하는 HTTP의 버전을 나타낸다.

Accept 필드는 서버에게 클라이언트에서 처리할 수 있는 데이터 형식의 리스트를 MIME (Multipurpose Internet Mail Extentions)[15,16] 형식으로 넘겨준다.

그 외에도 요구 필드(request field)에는 여러가지가 있는데, 위의 예에서는 그 중 하나로 User-Agent 헤더가 있다. 이것은 서버에게 요구를 보내는 클라이언트에 대한 정보를 제공한다. 또 공백 라인(blank line)은 헤더의 끝과 클라이언트에서 서버로 보내는 데이터의 시작을 가르킨다.

서버가 클라이언트에게 돌려주는 응답의 예는 다음과 같다.

```
HTTP/1.0 200 OK
Data: Friday, 12-Sep-94 10:12:23 GMT
Server: NCSA/1.3
MIME-Version: 1.0
Content-Type: text/html
Last-modified: Friday, 12-Sep-94
10:10 :06 GMT
Content-Length: 1808
```

```
<HTML>
... 중략 ...
</HTML>
```

즉 서버가 요구를 받았을 때, 서버는 지정된 객체(파일 또는 프로그램)에 지정된 메소드를 적용시키고, 그 결과를 다시 클라이언트에게로 넘겨준다. 응답 헤더(response header)의 처음 행이 상태라인(status line)이라고 불리는 것으로 요구를 해석한 결과를 클라이언트에게 알리고 있다. 이러한 상태라인은 일반적으로 다음의 형식을 갖는다.

<i>HTTP_version status_code explanation</i>
---

HTTP\_version은 서버에 사용되고 있는 프로토콜의 버전이며, 상태 코드(status\_code)는 200~599의 숫자로써 연결의 상태를 나타낸다. 그리고 설명(explanation) 필드는 그 상태에 대한 더 상세한 정보를 나타내는 문자열이다. 위의 예에서 보면 서버에서 사용되고 있는 HTTP 버전은 1.0이며, 상태 코드는 200으로써 주어진 요구가 성공적으로 끝났다는 것을 말한다. 참고로 상태 코드(status\_code)가 200~299는 작업의 성공적 수행, 300~399는 리다이렉션(re-direction) 즉, 요구(request)에서 주어진 URL의 객체가 다른 곳으로 옮겨졌음을 뜻하며 그 객체의 새로운 URL을 클라이언트에게 보내는 것, 400~599는 여러 메시지를 뜻한다.

나머지 응답 헤더들은 서버에 대한 정보와 보내는 응답에 대한 정보를 포함한다. 그리고 그 뒤에는 하나의 행을 띄우고 실제의 데이터(이 경우에는 HTML문서)가 전송된다.

### 3) HTML(HyperText Markup Language)

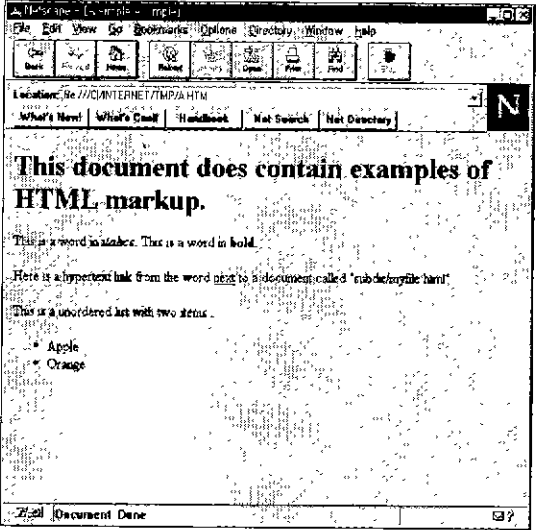
WWW에서 사용하고 있는 하이퍼텍스트를 기술하기 위한 언어가 HTML이다. HTML은 ISO 8879로 정의된 SGML(Standard Generalized Markup Language)의 DTD(Document Type Definition)에 의해 정의된 언어로 문장의 요소를 표현하기 위해 SGML기법에 따라 태그를 붙이고 있다. 즉 <element-name> . . . </element-name>과 같은 표현 형식이다.

다음은 HTML문서의 간단한 예를 보인다.

```

<html>
<head>
<title>A simple example </title>
</head>

<body>
<H1> This document does contain
examples of HTML markup.</H1>
This is a word in <I>italics</I>. This is
a word in <B>bold</B>.<p>
Here is a hypertext link from the word
<A HREF="subdir/myfile.html">next</
A> to a document called "subdir/myfi
le.html".<p>
This is a unordered list with two
items : <br>
<UL>
<LI>Apple
<LI>Orange
</UL>
</body></html>
    
```



[그림 3] Netscape에서 실행한 HTML 문서

4.2 CGI : WWW의 동적인 측면

이 장에서는 WWW의 동적인 측면을 나타내는 CGI(Common Gateway Interface)를 구체적인 예를 통하여 소개한다. 클라이언트에서 HTTP로 WWW 서버에게,

```
GET /cgi-bin/command/path HTTP/1.0
```

이 예를 클라이언트에서 표시해 보면, 태그의 정보에 의해 구성되는 문서를 볼 수가 있다. 각 클라이언트에 의해 조금씩 달라질 수 있으나, Netscape에서는 그림 3과 같이 보여진다.

태그에 사용되는 알파벳은 대소문자 구별이 없다. 여기서는 가장 중요한 태그는 <A>, 앵커(Anchor)이다. 이 앵커 태그는 그 태그로 표시되는 문자열(혹은 화상)을 앵커로 하는 것인데, 'HREF'속성에 URL를 지정하여, 해당 앵커를 '링크'로 사용한다. 이 앵커 태그에 의해 비로소 HTML 문서는 하이퍼 텍스트 기능을 가진다.

으로 요구를 넘겨주면, WWW서버는 환경변수 'PATH\_INFO'에 '/path'가 설정되고, 'command'가 실행된다. 이 명령을 실행하여 표준출력으로 나온 결과를 액세스 해온 WWW클라이언트에게 데이터로서 넘겨 준다. 이 명령과 WWW서버 간의 처리과정을 결정하는 것이 바로 CGI이다.

이 CGI에 의해 WWW서버에 액세스하는 WWW 클라이언트에 대해 'path'에 대응하는 실제 파일과는 관계없이, 'path'를 WWW 측에서 해석하고 그 해석한 결과로 동적으로 HTML문서를 생성하여 클라이언트측에 넘겨 줄 수가 있다. 이러한 일련의 동

작을 '동적인 측면'이라 부를 수 있다. WWW에서는 이러한 일을 용이하게 할 수가 있다. 이것이 WWW를 재미있게 하고, 가능성을 크게 하는 중요한 이유이다.

### 1) Image Map과 URL Redirection

그림 1의 내부 그림은 영남대학교의 홈페이지에서 부 페이지를 선택하는 것이다(URL은 <http://www.yeungnam.ac.kr/>). 이 그림이 Image Map으로 클릭하는 위치에 따라 표시되는 페이지가 달라진다. 예를 들어 사용자가 '정보자원연결' 부분을 클릭하면 WWW 클라이언트는 정보자원 연결을 위한 페이지를 표시한다. 이것의 수행되는 모양은 다음과 같다. 클라이언트는 Image Map의 URL뒤에 '?'와 마우스로 클릭된 위치의 좌표를 부가하여, 다음과 같은 요구를 송신한다.

```
GET /cgi-bin/imagemap/ ++?307,57 HTTP/1.0
```

서버에서는 CGI명령어인 `imagemap`이 가동되고, 그 좌표를 해석한 후, 페이지가 있는 위치 및 화일명을 포함하는 헤드를 가진 메시지를 클라이언트에게 보낸다. 클라이언트는 이것을 보고 새로운 페이지를 가져오기 위한 요구를 낸다.

### 2) POST 메소드

WWW서버에서 단순히 정보를 브라우즈하는 수동적인 수단으로 부터, 보다 능동적인 수단이 되게 하기 위해서는 클라이언트 측에서도 서버에게 정보를 보내는 것이 필요하다.

WWW의 초기에는 클라이언트에서 서버로 정보를 보내기 위해서는, GET 메소드를 사용하여 URL 속에 그 정보를 인코딩하여 보냈다. 이 방법은 검색을 위한 키워드 정도를 보내는 데는 충분했지만,

FORM이 만들어져 문서를 보낼 수 있게 됨으로서 URL 자체가 너무나 커져 상당히 불편하였다. 이러한 배경으로 POST 메소드가 사용되게 되었다. 이 POST 메소드는 통상 CGI 명령과 혼합하여 사용된다.

다음은 FORM과 POST 메소드를 사용한 WWW의 예를 보인다.

```
<html><head>
<title>Example of a HTML FORM</title>
</head>

<body>
<FORM ACTION="http://www.yeungnam.ac.kr/cgi-bin/uncgi/polling.cgi"
METHOD=POST>
<p> Search string:
<INPUT TYPE="text" NAME="srch" VALUE="dogfish">
<p> Search Type:
<SELECT NAME="srch_type">
<OPTION> Insensitive Substring
<OPTION SELECTED> Exact Match
<OPTION> Sensitine Substring
<OPTION> Regular Expression
</SELECT>
<p> Search databases in: <em>(multiple items can be selected)</em><br>
<INPUT TYPE="checkbox" NAME="srvr" VALUE="Canada" CHECKED> Canada
<INPUT TYPE="checkbox" NAME="srvr" VALUE="Russia"> Russia
<INPUT TYPE="checkbox" NAME="srvr" VALUE="Sweden" CHECKED> Sweden
<INPUT TYPE="checkbox" NAME="srvr" VALUE="U.S.A."> U.S.A. <p>
<INPUT TYPE="submit">
<INPUT TYPE="reset">
</FORM>
</body></html>
```



위의 예에서 이 FORM문에는 변수명으로 srch, srch\_type, srvr이 사용되었고, 각각에 srch=dogfish, srch\_type=Exact Match, srvr=Canada, srvr=Sweden의 값들이 할당되었다. 이 때 HTML FORM이 이 데이터들을 POST 메소드를 사용해서 HTTP 서버에게 보내는 요구는 다음과 같다.

```

POST /cgi-bin/uncgi/polling.cgi HTTP/1.0
Accept: text/plain
Accept: application/x-html
Accept: application/html
Accept: text/x-html
Accept: text/html
Accept: audio/*
.
.
Accept: text/x-setext
Accept: */*
User-Agent: NCSA Mosaic for the X Window System/2.4 libwww/2.12 modified
Content-type: application/x-www-form-urlencoded
Content-length: 58

srch=dogfish&srch_type=Exact+Match&srvr=Canada&srvr=Sweden
    
```

위의 예에서와 같이 GET 메소드에 비해 POST 메소드의 사용에는 다음과 같은 특징이 있다.

첫째, POST 메소드는 전송하는 데이터를 URL이 아닌 메시지 본체(message body)에 실어 보낸다. 둘째, 요구 헤더(request header)에 두 개의 새로운 요구 헤더 필드, Content-type와 Content-length,가 있다. Content-type은 서버에게 보내는 데이터의 MIME-type을 지정하고, Content-length는 그 데이터의 바이트 수를 가르킨다. 그 외에도 HTTP method에는 HEAD, LINK, PUT 등이 있다.

## 5. 홈 페이지 설계

최근에 국내에는 홈 페이지의 제작을 도와주는 수많은 서비스 회사가 생겨나고 있다. 또한 개인의 홈 페이지를 개설할 수 있도록 해 주는 서비스도 성업중이다. 이러한 붐과 함께 국내에는 수많은 개인 및 조직의 홈 페이지가 개설되어 있다. 따라서 단순히 홈 페이지를 가지는 것 만으로는 다른 홈 페이지와 차별화가 어렵다. 특히 기업의 홈 페이지 구축에는 상당한 노력이 필요하며, 그저그런 홈 페이지라면 없는 것보다 못할지도 모른다. 개인의 홈 페이지라 할지라도 액세스하는 사람이 없다면 재미가 없을 것이다. 다음에는 개인이나 조직이 홈 페이지를 가질 때 명심해야 할 사항을 소개한다. 여기서는 홈 페이지를 액세스하는 사람을 사용자라 부른다.

### 1) 영문 홈 페이지를 가진다.

전세계에 정보를 제공한다는 기분으로 가능하면 영문 홈 페이지를 가지기 바란다. 그러나 영어만의 홈 페이지를 가지는 것도 국내 사용자에게는 자존심 상하는 일이다.

### 2) 전 페이지의 통일성

제일 처음의 페이지가 상당히 멋지더라도 다른 페이지에는 온통 텍스트 뿐인 경우는 뭔가 빈약하게 보여 재미가 없다. 전 페이지에 꼭 화려한 그림 등을 넣을 필요는 없지만 모든 페이지에 대한 통일된 디자인이 사용자에게 주는 인상은 상당히 달라진다.

### 3) 사용자에게 흥미있는 내용

사용자는 실증을 쉽게 낸다. 재미가 없거나 도움이 되지 않는다면 즉시 액세스를 멈추어 버린다. 이렇게 되면 그 홈 페이지는 사용자가 찾아오지 않는 쓸쓸한 곳이 되고 만다. 사용자의 관심을 끌 수 있는

요점은 다음과 같다. 특히 첫 두 사항은 한번 찾아온 사용자를 다시 찾아오게 하는 중요한 사항이다.

- 항상 새로운 내용을 추가시킨다.
- 도움이 되는 살아있는 정보를 올린다.
- 유익한 내용 즉, 즐거움이 있어야 한다.

#### 4) 정보 수집 및 관리체계의 일원화

커다란 조직에서 본격적인 홈 페이지를 가지고 계속 좋은 정보를 서비스하기 위해서는 조직의 곳곳에서 항상 발생하는 새로운 정보를 제공해 주어야 하고, 이러한 정보를 일관되게 관리하고 WWW상에 정보를 올리는 관리체계가 필요하다.

#### 5) 최소의 데이터로 최대의 정보 제공

사용자가 정보전송을 기다리다 지치는 일이 없도록 액세스시간이 최소화되도록 설계되어야 한다. 그러기 위해서는 WWW설계시 다음 사항들을 주의한다.

- 불필요한 그림 등은 최소한 억제하거나 사이즈를 적게 한다.
- 그러나 텍스트만으로 구성된 재미없는 페이지도 많지 않도록 주의한다.
- 필요한 정보를 쉽게, 적은 횟수의 페이지를 거쳐 찾을 수 있도록 시나리오를 작성한다.
- 특히 영문 홈페이지를 작성시에는 데이터 전송 지연이 크다는 가정하에 데이터량을 더욱 줄인다.

#### 6) 화상 데이터

화상을 사용하여 홈 페이지를 만들 수 있는 것이 WWW에 의한 정보제공의 커다란 이점이지만, 적절한 크기의 화면을 사용하도록 주의해야 한다. 커다란 화면은 전송에 시간이 걸리고 네트워크에도 부하가 걸린다. 또한 256색 정도 밖에 표시할 수 없는 컴퓨터가 많기 때문에 사용하는 색수는 가능한한 억제

하는 것이 좋으며, 또한 이것은 파일 용량도 줄어들게 된다.

문자 베이스의 WWW클라이언트로 액세스하는 사용자를 위해 문자 베이스의 페이지도 준비하는 것이 좋다. 특히 Image Map을 사용하는 메뉴에는 문자 베이스의 메뉴도 제공되는 것이 좋다.

#### 7) 검색

홈 페이지나 링크에서 키워드를 키인하여 그 서버에 있는 모든 정보(타이틀이나 HTML문서의 전문 등)를 검색할 수 있다면 사용자에게는 아주 편리할 것이다.

## 6. WWW의 과제 및 장래

이 장에서는 WWW의 과제와 장래에 관하여, 표준화의 동향과 함께 살펴본다.

#### 1) 보안문제

기업의 인터넷 활용이 급증하면서 특히 보안문제는 초미의 관심사로 대두되고 있다. 기업의 전산자원은 물론 중요한 정보 및 기밀 자료들의 유출과 해킹으로 인한 피해를 막기 위해서는 치밀한 보안 대책이 필요하다. 주로 이런 해킹 유형과 동향을 살펴보면, 이더넷 수준의 해킹, 프로토콜을 이용한 해킹, 유닉스 운영체제의 결함을 이용한 해킹 등으로 나누어 볼 수 있다.

한 예로써, WWW를 전자거래 등의 상업에 이용하기 위해서는 보안문제를 해결한 HTTP를 실현하는 것이 본질적으로 필요하다. 이 분야에는 EIT사가 제안하는 SHHTTP(Secure HTTP)와 Netscape Communications사가 제안하는 SSL(Secure Socket Layer)가 유망한 것 같다. 이들 중 어느 하나 혹은

양자를 조합한 것이 W3C등에서 채용되면, 그것이 바로 표준이 될 것으로 생각된다. 또 미국을 중심으로 개발되고 있는 암호화 기술이 전 세계 인터넷 사용자를 위한 공통 방식으로 사용되기 위해서는 미국이나 영국의 암호화 기술의 수출 장벽 타파와 암호화 기술이 평준화되어야 하는 문제 해결이 우선이다. 이러한 데이터 암호화 기술은 2000년대를 이끌어 갈 중요기술로 평가받고 있으며, 민감한 데이터, 특히 재정 처리와 의료 기록은 암호화가 필수적이다.

인터넷 보안대책에는 크게 두가지 측면이 있다. 하나는 기술적 측면, 즉 한 기관 네트워크와 인터넷 사이의 액세스를 제어하는 '파이어 월(Firewall)'을 어떻게 구축하는가에 대한 것이다. 그러나 다른 하나, 무시할 수 없는 것이 '인간적인 측면'인데 여기에는 각 개인이 보안에 대한 기본적인 마인드를 갖는 것이 필요하다.

## 2) URI

URL에 관해서는 IETF의 URI 워킹그룹에서 검토되어, 현재는 RFC 1738로 정리되어 있다. 그러나 URL에서는 문서의 위치(어느 기계의 어느 'path'에 있는가) 및 어느 프로토콜로 가져올 것인가에 의존한다. 즉, URL로 지정하면 완전히 같은 문서가 가까이 존재하더라도 반드시 지정된 장소에서 지정된 문서만 액세스할 뿐 아니라, 놓여있는 장소가 변하면 액세스가 실패하는 등의 문제가 있다.

그래서 문서의 위치 및 프로토콜에 의존하지 않는 URI(Universal Resource Identifiers)를 실현하기 위해 IETF의 URI 워킹그룹에서 URN(Uniform Resource Names) 등의 검토가 되고 있다. 구체적으로는 DNS(Domain Name System)와 닮은 실현이 될 것 같다. WWW클라이언트에서 URN으로 문서를 액세스하면 어떤 서버에서 URN를 URL

(Uniform Resource Locators)로 구체적인 해결을 하고, 그 URL로 해당 문서를 취득한다. 여기서 URL이 IP주소, URN이 도메인 이름에 해당하는 것을 알 수 있다. 이 URN과 URL를 연결시켜 주는 소스 인코딩 정보의 기술방법은 URC(Uniform Resource Characteristics)로서 검토되고 있다.

## 3) HTML 3.0

HTML 초기 버전은 처리시간이 걸리는 고기능 요소는 별로 없었다. 그러나 WWW가 넓게 보급되고 여러가지 기능에 대한 요구도 점점 많아지고, 컴퓨터 자체도 처리속도가 향상됨으로서 고기능을 제공하기 시작하는 경향이 있다.

현재 일반적으로 사용되는 HTML인 HTML 2.0이 인터넷 Draft로서 공개되어 있다. IETF에서는 HTML에 관한 논의가 계속되고 있기 때문에 그림, 표, 수식 등을 지원하는 HTML 3.0의 인터넷 Draft도 공개되어 있다.

## 4) 자바/핫자바

인터넷과 분산처리, 하이퍼 텍스트 기술을 통합한 새로운 개념이 속속 등장하고 있는 가운데, 현재 가장 주목을 받고 있는 기술로는 썬에서 개발한 자바와 핫자바로 각각 프로그래밍 언어와 그 프로그램이 수행되는 운영 환경이다. 자바언어는 바이너리 코드 포맷 형태로 프로그램을 만들어 냈으므로 핫자바가 설치된 시스템이면 하드웨어나 운영체계에 구분없이 이기종 네트워크환경에서 모두 수행될 수 있다. 또한 객체지향, 다이내믹 로딩(dynamic loading), 바인딩, 멀티스레딩을 지원한다. 이와 더불어 네트워크 환경에서 분산된 형태로 운영되기 때문에 강력한 보안 기능을 지원하며, 이식성과 신뢰성도 뛰어나다. 또한 텍스트, 그래픽, 사운드, 비디오 등을 삽입할 수 있어 시각적이고 정적인 하이퍼텍스트 페이지를

각종 애니메이션, 대화형의 특별한 애플리케이션이 추가되는 동적인 페이지로 구현할 수 있다. 현재는 Netscape 2.0이상에서도 자바언어를 처리할 수 있는 기능을 제공하고 있다.

핫자바는 자바로 만들어진 일종의 브라우저로써, 기존의 WWW 브라우저가 미리 정의해 놓은 몇몇 프로토콜(http, ftp등)의 데이터만 인식하는데 비해, 프로토콜의 제한이 없다. 핫자바는 다양한 코드를 필요에 따라 다운로드해 자신이 인식할 수 있는 프로토콜로 변형, 인식한다. 이는 자바로 씌여진 어떠한 프로그램도 핫자바를 통해 여러 운영환경에서 수행할 수 있고, 프로그램 업데이트도 자동으로 할 수 있다는 의미이다. 이렇듯 핫자바는 솔라리스 2.x와 윈도우 95에서 작동되는 베타 버전이 나와 있다. 자세한 정보는 <http://java.sun.com>에서 구할 수 있다.

### 5) VRML

HTML의 한계를 극복하기 위해 등장한 언어중의 하나인 VRML(Virtual Reality Modeling Language)은 3차원 가상공간을 표시하기 위한 언어 사양이다. 정확히 말하자면 실리콘그래픽사가 개발한 3차원 그래픽 라이브러리인 OpenGL상에서 동작하고, Open Inventor(오브젝트지향 3차원 그래픽 개발툴)의 파일형식인 iv파일의 서브셋(Subset)사양이다. 원래 Open Inventor의 iv파일은 텍스트 형식과 바이너리 형식의 두종류가 있지만, VRML에서는 실리콘그래픽사로부터 무상으로 사양이 공개되고 있는 ASCII 텍스트 형식만을 지원하고 있다. 텍스트 형식으로 3차원 공간의 데이터를 기술한 wrl파일을 작성한다.

인터넷상의 WWW 서버상에 설치된 VRML 형식의 3차원 공간 기술파일(\*.wrl)이 전송돼, VRML 이 용자측의 컴퓨터에 탑재돼 있는 전용 VRML 브라우저상에서 텍스트 형식의 데이터를 3차원 데이터로

변환(렌더링)함으로써, 3차원 가상공간이 디스플레이에 표시된다. 이에 따라 표시된 3차원 가상공간을 마우스 조작만으로 이동할 수 있다. 또한, 3차원 공간안의 물체에 링크시킬 수 있고, 그 물체를 마우스 클릭함으로써, 다른 3차원 공간을 읽어들이거나, 2차원의 데이터(HTML파일)를 읽어 들일 수 있다. 이러한 VRML은 3차원 가상공간을 표시하는데 비해, 비교적 데이터 전송량이 적게 소요되는 이점이 있다. 1993년 말부터 태동하기 시작한 VRML 연구는 현재 VRML 2.0이 만들어진 상태이며, 이 버전에는 인벤터 파일 포맷이 추가되고, 네트워킹 관련 기능이 보강돼 더욱 편리한 인터페이스를 제공하고 있다. VRML로 만들어진 웹 사이트를 보기 위해서는 일단 넷스케이프와 같은 VRML 브라우저가 필요한데, 그 중 가장 널리 알려진 것은 실리콘 그래픽사에서 만든 WebSpace Navigator인데 이는 다음 사이트에서 베타 버전을 구할 수 있다.

<http://www.sgi.com/Products/WebFORCE/WebSpace>  
<ftp://ftp.sd.tgs.com/pub/templete/WebSpace>

### 6) MBone

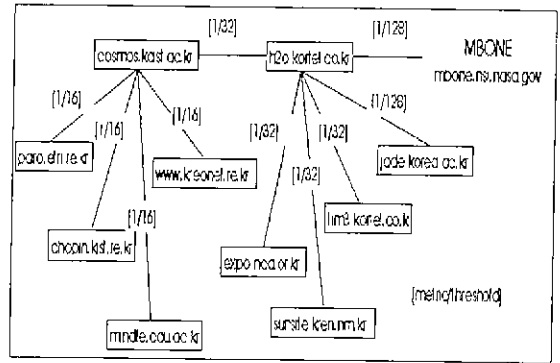
MBone(Multicast backBone)[13,14]은 인터넷 상에서 화상회의와 같이 여러 참가자가 있고, 이들간에 오디오나 비디오 같은 멀티미디어 데이터를 전송하는 애플리케이션을 가동하기 위해 만들어진 '가상 네트워크'이다. 여기서 가상 네트워크라는 것은 기존의 인터넷에 연결돼 있는 호스트들을 소프트웨어적으로 처리해 마치 또다른 네트워크가 인터넷 위에 있는 것처럼 보이도록 했기 때문이다. IP 멀티캐스트가 MBone에서 가장 기본이 되는 프로토콜이며, 이를 실현하기 위해 '터널링'이라는 방법을 사용한다. 이 때 터널(tunnel)이란, 두 멀티캐스트 라우터(multicast router, 하나의 네트워크 주소마다 멀티캐스트 라우팅을 제공하는 호스트)간의 '가상의 연결'

을 말하며, 이 터널들이 바로 MBone을 구성한다. 그리고 화상회의 등의 응용프로그램에서 메시지를 전송하기 위해서는 전송 시간이나 전송되었다는 보장과 같은 것들이 필요한데, 이 때문에 개발된 새로운 프로토콜이 RTP(Real-Time Transfer Protocol)이다. RTP에는 이런 정보 외에도 현재 전송되는 오디오 데이터나 비디오 데이터의 코딩 방법이 무엇인지, 또한 해당 응용프로그램을 사용하는 사람의 이름과 전자우편 주소 등 회의 참가자에 대한 정보가 함께 포함된다. RTP에 대한 보다 자세한 정보는 'http://www.focus.gmd.de/minos/employees/hgs/rtp/faq.html'에서 볼 수 있다. 참고로, MBone에 관련된 프로토콜 개발이나 운영 등을 담당하는 조직은 1986년에 구성된 IETF(Internet Engineering Task Force)이다.

현재 MBone 상에서 문서를 공유하고, 이를 동기식으로 웹 공동 작업이 가능하도록 하는 연구는 W3C(WWW Consortium)의 'Collaboration Development Area'를 중심으로 이뤄지고 있다. 이 연구는 현재 두가지 접근 방식으로 진행되고 있다. 기존의 웹 브라우저에 멀티캐스트 기능을 확장하도록 하는 방식(대표적으로 Shared Mosaic)과 기존의 웹 브라우저를 그대로 이용한 채 MBone과 웹 사이에 중간 매개 프로그램(대표적으로 Webcast)을 이용하는 방식이 그것이다.

여기서 웹 입장에서 보면 MBone은 멀티캐스팅이 지원되는 강력한 허부망이며, MBone 입장에서 보면 웹은 WB(White Board) 대신 사용할 수 있는 새로운 응용이다. 보통 웹에서 차세대 3대 기술로 꼽는 것은 자바, VRML, API인데, 이 중에서 API가 MBone 활용에 크게 기여했다. 그러나 굳이 API를 논하지 않더라도 웹은 인터넷상에서 대부분의 응용 서비스를 포함시켰으며, 조만간 MBone과 웹의 결합도 확실한 사실로 나타날 것이다. 그림 4는 현재

국내의 기관별 주요 라우터와 MBone 토폴리지를 나타낸다.



〔그림 4〕 국내의 기관별 주요 라우터와 MBone 토폴리지

### 7) 기반 구조

WWW의 장래는 상기와 같은 표준화와 더불어 네트워크의 기반 구조에도 크게 영향을 받을 것으로 생각된다. WWW를 포함한 인터넷이 건전하게 성장해 가기 위해서는, 네트워크의 전송 용량이 충분해야 한다. 그러나 한국의 현재 상황은 회선의 전송 용량이 충분하다고 할 수 없다. 이러한 전송 용량의 부족은 특히 데이터량이 많은 WWW의 확산에 큰 걸림돌이 된다. 그러나 이러한 문제점은 이미 충분히 인지하고 있으므로 점차 좋아질 것으로 생각된다.

## 7. 결론

World Wide Web(WWW)에 관련된 특징, 역사, 기술, 홈 페이지의 설계, 과제 및 장래 등에 대해 살펴 보았다. 특히 기술적인 측면으로는 동적인 면에 초점을 맞추어 해설하였다. 또한 홈페이지의 설계시에 고려되어야 할 몇가지 사항을 제안하고, WWW의 과제 및 장래에 대하여 표준화 동향을 중심으로

해설하였다.

WWW는 컴퓨터 관련 업계는 물론이고 컴퓨터와 그다지 관계가 없는 분야에서도 주목을 집중하고 있고 또 그 변화도 상당히 빠르다. 시시각각으로 새로운 기술이 개발되고 있는 이 분야에서는 이 글이 발표될 즈음에는 이미 상당히 진부한 이야기가 될지도 모른다. 그러나 WWW을 이해하는데 조금이라도 도움이 된다면 하는 것이 하나의 바램이다.

참 고 문 헌

1. Internet Engineering Task Force,  
http://www.ietf.cnri.reston.va.us/
2. World Wide Web Consortium(W3C),  
http://www.w3.org/hypertext/WWW/Consortium/Prospectus/Overview.html
3. World Wide Web Consortium(W3C),  
http://www.w3.org/pub/WWW/Addressing/Addressing.html
4. RFC 1630, "Universal Resource Identifiers(URI)" (1994).
5. RFC 1738, "Uniform Resource Locators(URL)" (1994).
6. RFC 1737, "Functional Requiements for Uniform Resource Names" (1994)
7. Internet Draft : Hypertext Markup Language Specification Version 2.0 <draft-ietf-html-spec-01.txt> (1995).
8. Internet Draft : Hypertext Markup Language Specification Version 3.0 <draft-ietf-html-spec3-00.txt> (1995).
9. Internet Draft : Hypertext Transfer Protocol Specification <draft-ietf-http-v11-spec-01.txt> (1996).

10. Internet Draft : Hypertext Transfer Protocol Specification <draft-ietf-http-v10-spec-05.txt> (1996).
11. IAN. S. GRAHAM, THE HTML SOURCEBOOK, John Wiley & Sons, Inc., 1995.
12. 益岡 龍介, 木庭袋 圭祐 "World Wide Web(WWW)" (日本情報處理學會紙 Vol. 36, No.12, pp.1155-1165, December (1995).
13. The Mbone Information Web,  
http://www.best.com/~prince/techinfo/mbone.html
14. Mbone-Korea Web, http://paro.etri.re.kr/
15. RFC 1521, "MIME(Multipurpose Internet Mail Extentions) Part One" (1993).
15. RFC 1522, "MIME(Multipurpose Internet Mail Extentions) Part Two" (1993).



김 증 군

1981년 · 영남대학교 전자공학과 (공학사)  
 1987년 : 영남대학교 전자공학과 계산기전공 (공학 석사)  
 1991년 : (일본) 전기통신대학 정보공학과 (공학 박사)  
 1984년 ~ 1991년 경북전문대학 전자계산과 전임강사  
 1991년 ~ 현재 : 영남대학교 진신공학과 부교수  
 관심분야 : 분산처리 시스템, 컴퓨터 망, 운영체제, 멀티미디어 정보 처리, 성능평가



임 인 택

1996년 : 영남대학교 건축공학과(공학사)  
 1996년~현재 영남대학교 멀티미디어 정보통신학과 석사과정  
 관심분야 : 컴퓨터망, 멀티미디어 정보처리