

<제 목 차 례>

1. 인터넷 살펴보기	1
1-1 인터넷 개요	1
(1) 인터넷 소개	1
(2) 인터넷 주소	4
(3) 도메인 이름 해결 방법	6
1-2 인터넷 관련 기구와 표준화	8
(1) 인터넷 관련 기구	8
(2) 인터넷 관련 표준화	10
2. HTML	12
2-1 인터넷의 웹 서비스	12
(1) 웹 서비스 개괄	12
(2) 웹 브라우저(web browser)	13
(3) 웹 표준 HTML	15
2-2 HTML의 기초	18
(1) HTML 문서의 구조	18
2-3 CSS 익히기	20
(1) CSS 개요	20
(2) CSS 시작	20
(3) CSS의 장점	21
(4) CSS 구문과 선택자	22
3. 컴퓨터시스템	24
3-1 컴퓨터시스템의 구조	24
(1) 컴퓨터 정의	24
(2) 노이만 컴퓨터 구조(Von Neumann computer architecture)	25
3-2 하드웨어와 소프트웨어	27
(1) 소프트웨어(software)	27
(2) 응용 소프트웨어 (application software)	29
(3) 하드웨어(hardware)	29

<표 차례>

<표 1-1> 주요 국가별 IPv4 주소 보유 순위	2
------------------------------------	---

<그림 차례>

[그림 1-1] 도메인 이름을 가지는 호스트 개수(1993-2017)	2
[그림 1-2] ARPANet의 탄생(1969)	3
[그림 1-3] UCLA에 보관중인 ARPANET을 통해 전송된 최초의 메시지 기록	3
[그림 1-4] TCP/IP에서 사용하는 주소	5
[그림 1-5] 초기 이름 해결 방법	6
[그림 1-6] hosts 이름 해결 방법	6
[그림 1-7] DNS 서버를 이용한 이름해결 방법	7
[그림 1-8] 인터넷 관련 기구	8
[그림 1-9] 인터넷 관련 표준화 과정	10
[그림 2-1] 팀 버너스 리	12
[그림 2-2] HTML 표준의 설계 흐름	16
[그림 2-3] HTML <u>문서</u> 구성 예시	18
[그림 2-4] HTML <u>문서</u> 구성	18
[그림 2-5] CSS의 구문(syntax)	22
[그림 3-1] 컴퓨터의 정의	24
[그림 3-2] ALU 기호 표시	24
[그림 3-3] 노이만(John von Neumann)	25
[그림 3-4] "First Draft of a Report on the EDVAC"	25
[그림 3-5] 노이만 구조와 프로그램 내장 개념	26
[그림 3-6] 소프트웨어 종류	27
[그림 3-7] 프로그래밍 언어와 언어 번역기	28

1. 인터넷 살펴보기

1-1 인터넷 개요

(1) 인터넷 소개

① 인터넷 정의

인터넷(Internet)이란 무엇일까. RFC 1462("What is the Internet?", 1993)는, "인터넷은 TCP/IP를 기반으로 많은 사람이 자원을 공유할 수 있도록 한 컴퓨터 네트워크의 한 종류이다"라고 말하고 있다. 일반 네트워크가 지리적으로 떨어져 있는 몇 대 또는 몇 십대의 컴퓨터들을 연결시킨 것에 비해, 인터넷은 전 세계 거의 모든 국가의 수십억대의 호스트 컴퓨터를 연결하여 이루어진 거대한 네트워크의 네트워크(network of network)이며 세계에서 가장 큰 네트워크라 할 수 있다.

송신 장치와 수신 장치 간의 원활한 정보교환을 위해서는 연결된 장치들의 제반 동작을 규정하는 프로토콜(protocol)이라고 하는 통신규약이 필요한데, 인터넷에서는 TCP/IP(Transmission Control Protocol/Internet Protocol) 통신 규약을 사용하고 있다.

㉠ 초창기 인터넷의 주요 가입 기관은 주로 대학이나 정부 연구기관 또는 기업체 연구소이기 때문에, 인터넷은 그곳에 저장되어 있는 고급 정보를 아주 빠른 시간에 공유할 수 있는 세계 최대의 학술 연구 컴퓨터 통신망 역할을 하였는데, 이 무렵 인터넷은 '정보의 바다', '학술 정보의 보물 창고' 또는 '정보의 낙원(Information paradise)' 등으로 불리기도 하였다.

㉡ 미국 엘 고어(El Gore) 부통령^[1]은 인터넷을 정보의 초고속도로(Information Superhighway)라고도 불렀다. 물론, 인터넷을 이용해 얻을 수 있는 정보가 학술관련 정보에 국한된 것은 아니다. 뉴스 네트워크지의 표현대로 인터넷은 '사랑, 과학, 건강, 주식 문제 등 무엇이든 알려주는 꿈의 통신'이라 볼 수 있다. 이렇듯이 인터넷으로 연결된 세계는 새로운 전자지구촌(global electronic village)을 이루며, 이러한 가상의 공동체(virtual community)에 한 번 빠지면 쉽게 헤어날 수 없다는 의미에서 타임지는 인터넷을 정보화 시대의 가장 저렴하고 안전한, 산뜻한 마약(smart drug)이라 표현했다.

② 인터넷 현황

㉠ KrNIC^[2] 통계자료에 따르면, 2017년 3월 현재 인터넷 상에서 호스트를 구별하기 위해 할당된 IP주소의 개수는(IPv4 기준) 모두 36억 4,643만여 개이며, 특수용도의 IP주소 개수는 약 5억 8,851만여 개이다. 국가별 IP주소 보유 순위를 살펴보면 16억여 개를 넘게 보유하고 있는 미국이 1위이며 우리나라는 1억 1,243만 여개로 중국, 일본, 영국, 독일에 이어 세계 6위를 차지하고 있다. 이러한 통계를 미루어보면 전 세계 거의 모든 호스트 간의 접속과 정보교환에는 인터넷이 가장 핵심 역할을 하고 있다고 할 수 있다.

[1] 엘 고어(Albert Arnold Gore Jr.) 미국의 45대 부통령(1993~2001년). 1992년 빌 클린턴의 러닝메이트(running mate)로 출마, 부통령에 당선

[2] 한국인터넷정보센터(Korea Network Information Center; <http://www.krnic.net>) 세계 각국의 인터넷주소자원을 관리하는 NIC 조직의 하나로 1999년 6월에 설립. 일본은 JPNIC, 중국은 CNNIC

㉠ [그림 1-1]은 인터넷에 연결되어 있으면서 도메인 이름(Domain Name)을 가지는 호스트 개수는 보여주고 있는데, 2000년대에 들어오면서 급격한 증가세를 보였으며 2017년 1월 현재 이 호스트의 개수는 2000년 1월에 비해 약 14.7배 증가한 10억 6,266만여 개로 조사되었다.

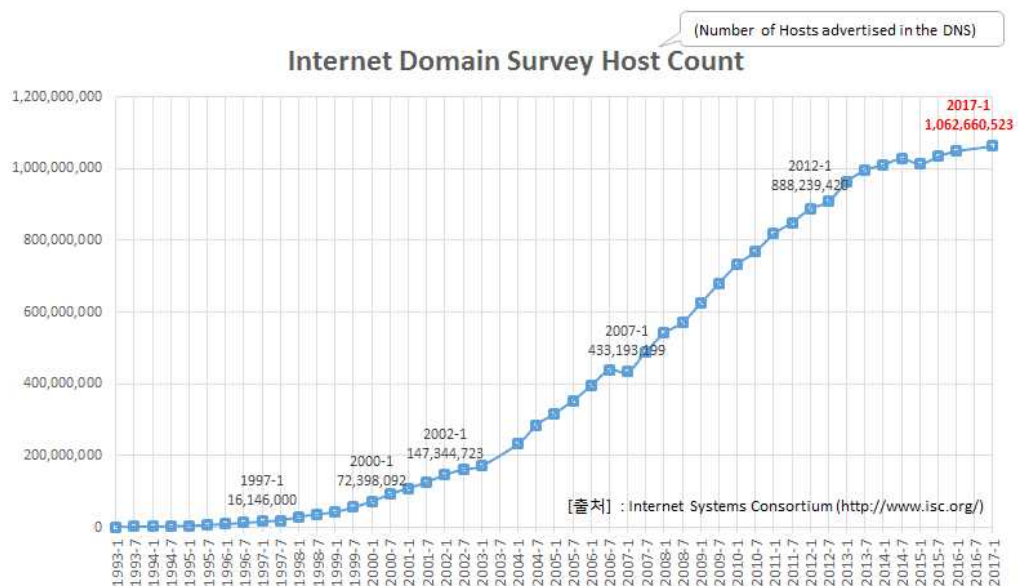
③ 인터넷의 출현

㉠ 인터넷(Internet)의 시작이 언제이냐에 대해서는 여러 이견이 있지만, 떨어져 있는 컴퓨터를 연결하기 위해서 각 컴퓨터가 속해 있는 네트워크를 서로 연결한다는 의미에서 보면, 미국 로스앤젤레스 소재 캘리포니아 대학(UCLA)과 스탠퍼드 대학(SRI) 컴퓨터가 연결된 1969년 10월 29일을 인터넷 시작으로 언급하는 경우가 일반적이다. 오늘날 인류의 과학 산물 가운데 가장

<표 1-1> 주요 국가별 IPv4 주소 보유 순위

순위	국가	IPv4 주소 수(개)
1위	미국	1,612,908,032
2위	중국	338,164,224
3위	일본	203,293,696
4위	영국	122,054,680
5위	독일	119,439,232
6위	대한민국	112,427,008
7위	브라질	83,350,272
8위	프랑스	80,388,144
9위	캐나다	70,353,664
10위	이탈리아	53,931,328
11위	오스트레일리아	48,473,344
기타국가		801,654,384
특수용도		588,514,560
*미할당		60,014,728
합계		4,294,967,296

(2017.03.03 현재)



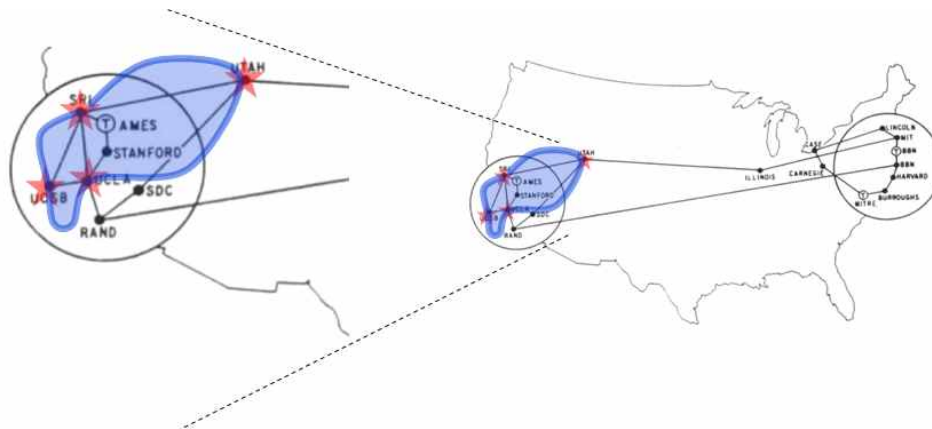
[그림 1-1] 도메인 이름을 가지는 호스트 개수(1993-2017)

큰 영향을 미치고 있는 것 가운데 하나로 손꼽히는 인터넷은 이제 약 50년에 가까운 역사를 가지고 있는 셈이다.

㉠ 인터넷 탄생의 발단은 1957년 10월 4일 구 소련(USSR)이 미국을 앞질러 세계 **최초로** 인공위성 스푸트니크(Sputnik) 1호를 쏘아 올리면서 시작되었다고 볼 수 있다. 구 **소련**

의 인공위성 발사에 충격을 받은 미국은 1958년 2월 군사 분야에 활용할 수 있는 과학 기술 연구를 지원하는 ARPA(Advanced Research Projects Agency)라는 **고등연구계획국을** 국방성(US. Department of Defense)에 설립하고 1960년대에 많은 컴퓨터 과학자를 지원하기 시작했다. 이 과정에서 새 연구자가 참여할 때마다 새로운 컴퓨터를 공급하고 서로 정보를 공유할 필요를 느끼면서 전국에 흩어져 있는 컴퓨터를 서로 연결해 연구의 효율을 높이려는 계획을 세우게 된다.

- ㉠ 1969년에 탄생한 **ARPANET(미국 국방성 연구망)**은 당시 패킷교환망 연구의 자금원 역할을 하였던 ARPA가 UCLA, SRI를 비롯하여 UCSB 대학(University of California, Santa Barbara: 산타바바라 소재 캘리포니아 대학), UTAH 대학(University of Utah, Computer Science Department)의 과학자들을 지원하여 각 대학의 컴퓨터를 서로 연결하도록 개발한 네트워크이다. 이렇게 보면 인터넷은 냉전의 산물이라고도 할 수 있다.
- ㉡ 인터넷 기술의 발명가라고 불리는 UCLA의 레너드 클라인록(Leonard Kleinrock) 교수는 **네트워크를** 구성하는 기술적 기초를 마련하였는데, 인터페이스-메시지-프로세서(IMP;



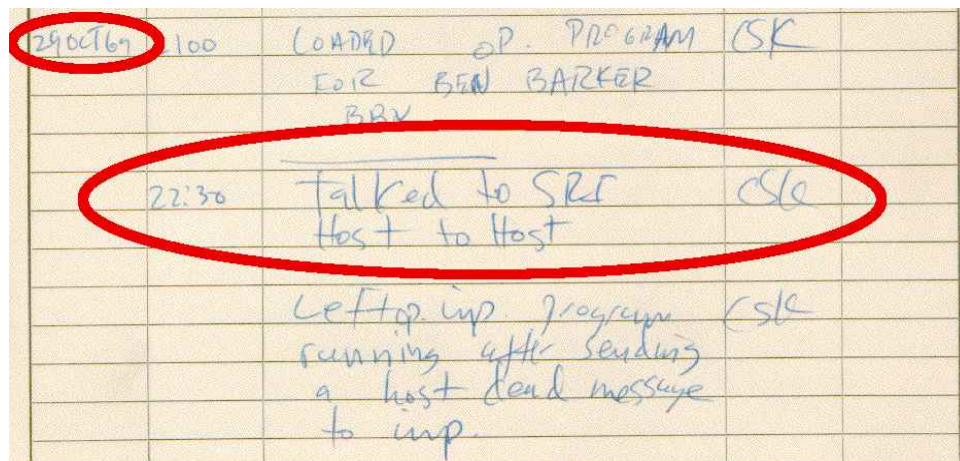
[그림 1-2] ARPANet의 탄생(1969)

Interface Message Processor)를 통해 UCLA 컴퓨터를 SRI 컴퓨터에 연결하고 “LOG”라는 단어를 **최초로** 전송하는데 성공했다(1969년 10월 29일 22시 30분).

- ㉢ 1969년 말 UCLA와 SRI, UCSB, 유타(Utah) 대학 등 4개 대학의 호스트 컴퓨터가 서로 연결되었는데 그 당시 호스트 컴퓨터 사이의 데이터 전송속도는 50Kbps로 극히 초보적인 수준이었다. 이후 ARPANET은 팽창하기 시작해 1972년에는 호스트 컴퓨터가 23개로, 1976년에는 111개로 늘어났고, 1974년 스탠퍼드 대학의 빈트 서프(Vint Cerf, Vinton Gray Cerf) 박사의 논문에서 인터넷^[3]이라는 용어가 처음 등장했다. 1973년 ARPANET은 영국과 노르웨이와 연결하면서 국제 통신망으로 변신을 하고 1980년대 들어서는 ARPANET의 팽창 속도는 더욱 빨라진다.

[3] Internet는 the global Internet, TCP/IP Internet, ARPA/NSF Internet라고도 불린다.

일반적으로 TCP/IP를 사용하는 사설 인터넷(private internet)은 “internet”라 하며, 오늘날 많이 사용하고 있는 의미의 the global Internet은 “Internet”라 한다.



[그림 1-3] UCLA에 보관중인 ARPANET을 통해 전송된 최초의 메시지 기록
(출처 : https://www.lk.cs.ucla.edu/internet_first_words.html)

- ㉠ ARPA가 그들 연구망에 연결된 컴퓨터들을 TCP/IP^[4] 프로토콜^[5]로 전환하는 1980년경부터 인터넷이 널리 확산되었으며, ARPANET은 인터넷 기간망(backbone)이 되면서 TCP/IP 초기 기술을 정착하는 역할을 하였다. 1983년 1월, 미국 국방장관은 인터넷에 접속되는 컴퓨터는 TCP/IP를 사용하여야 한다고 발표하였고, DCA(Defense Communication Agency)는 사용자가 많이 증가한 ARPANET을 연구 목적의 ARPANET과 군사 통신 목적의 MILNET으로 분리하였다.
- ㉡ ARPA의 지원을 받은 BBN(Bolt Beranek and Newman, Inc.)은 당시 많이 사용되고 있던 BSD UNIX(Berkeley Software Distribution UNIX)용 TCP/IP를 개발하였으며, Berkeley 대학 역시 ARPA의 도움을 받아 TCP/IP를 포함한 BSD UNIX를 보급하였다. 특히 BSD UNIX가 제공하는 소켓(socket) 개념으로 프로그래머들은 적은 노력으로 TCP/IP 프로토콜을 이용할 수 있게 되었으며, 과학자들은 TCP/IP와 관련된 실험을 많이 할 수 있게 되었다.
- ㉢ TCP/IP 기술과 과학자들 사이의 인터넷 성공은 다른 그룹들이 TCP/IP를 채택하도록 이끌었다. 1985년부터 미국과학재단(NSF^[6])은 TCP/IP를 기반으로 하여 6개 슈퍼컴퓨터 센터 주변의 통신망을 설립하기 시작하여 1986년에 NSFNET이라는 광대역 기간망(backbone)을 구축하였다. 결국 NSFNET은 모든 슈퍼컴퓨터 센터를 중심으로 많은 지역망을 포함하게 되었고, ARPANET에 연결하게 되면서 오늘날의 인터넷의 기반이 되었다.

(2) 인터넷 주소

① TCP/IP에서 사용하는 주소

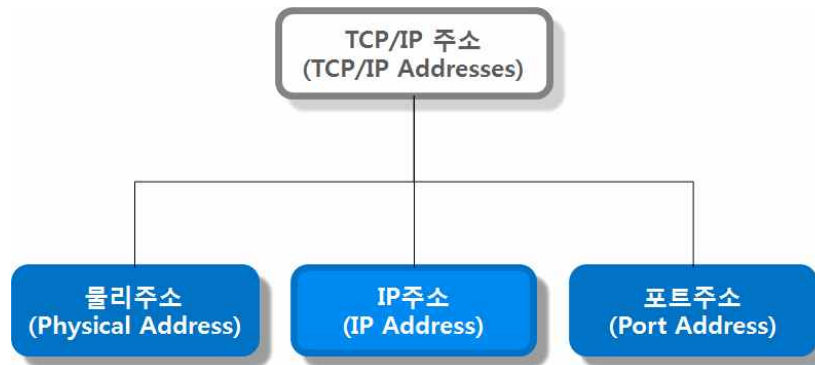
인터넷은 통신규약을 사용하여 인터넷워킹, 즉 네트워크와 네트워크가 서로 연결된 거대한 네트워크라 할 수 있다. 현 인터넷에서 사용하는 통신규약은 TCP/IP이고 인터넷에서의 정보

[4] Transmission Control Protocol/Internet Protocol

[5] 공식적으로는 TCP/IP 인터넷 프로토콜 군(TCP/IP Internet Protocol Suite)이라 부른다.

[6] National Science Foundation

교환의 주체는 네트워크에 접속된 호스트들이다. 인터넷에 연결된 어떤 호스트가 동일한 또는 다른 네트워크에 연결된 호스트에 정보를 전달하기 위해서는 목적지 호스트를 찾아가는 방법이 필요하다. 통신기술 면에서는 다양한 방법이 있을 수 있겠지만 각 호스트에 주소(address)를 부여하고 그 주소를 참조하여 목적지 호스트를 찾아가는 방법을 사용하고 있다. 이를 위해 인터넷에서 사용되고 있는 주소의 종류를 살펴보면 물리주소, IP 주소, 포트(port)



[그림 1-4] TCP/IP에서 사용하는 주소

주소 등이 있다.

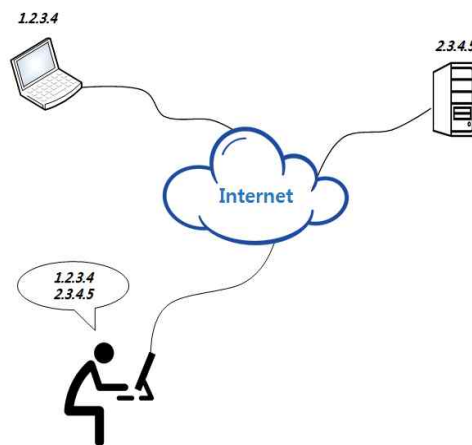
- ㉠ 물리주소는 목적지 호스트가 속한 네트워크에서 OSI 7계층 모델의 물리계층 및 데이터링크 계층을 통해 프레임 수준의 정보를 전달할 때 사용된다.
- ㉡ IP 주소는 인터넷에 있는 호스트들이 서로 통신하는 경우 호스트를 구분하기 위해 사용하는 주소이다. 인터넷 상의 하나의 호스트가 인터넷을 구성하는 다른 네트워크에 접속되어 있는 목적지 호스트를 찾아가는 경우 이를 경로배정(routing)이라 하고 이러한 경로배정을 수행하는 장치를 라우터(router)라 한다. IP주소는 이 라우터가 정보를 전달할 목적지 호스트로 경로배정을 할 때 사용되는 주소이다. 말하자면 인터넷에 있는 특정 호스트로의 경로배정을 위해 IP 계층이 사용하는 주소가 IP 주소이다.
- ㉢ 포트 주소(port address)는 OSI 7계층 모델의 전송계층(transport layer) 프로토콜에서 사용하는 주소로, 목적지 호스트에 전송되어 온 정보를 최종적으로 호스트 내의 어떤 서비스 응용프로그램에 전달하는 지를 지정하는 주소이다. TCP/IP 프로토콜 모델에서는 전송 계층인 TCP와 UDP가 이 주소를 관리하는데 보통 포트번호(port number)로 부르고 있다.

(3) 도메인 이름 해결 방법

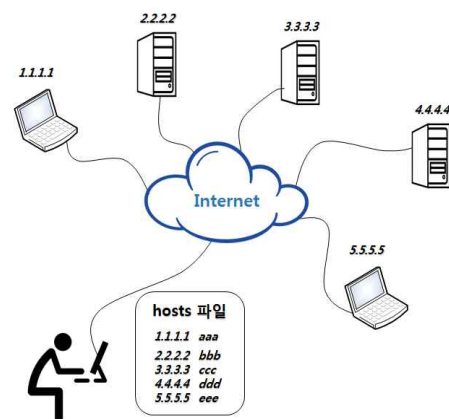
프로토콜로 TCP/IP를 사용하고 있는 인터넷에서는 네트워크에 있는 호스트를 지정하기 위해서는 궁극적으로는 IP 주소를 사용하여야 한다. 숫자로 된 IP 주소가 불편하여 사용자가 도메인 이름을 사용하게 되더라도, TCP/IP에 적용되기 전에 IP 주소로 변환되지 않으면 안 된다. 이렇게 도메인 이름을 IP 주소로 변환하는 것을 이름해결(names resolution)이라 한다.

① 초기 이름해결 방법

네트워크에 접속된 호스트가 몇 대 되지 않은 초창기에는 사용자가 해당 호스트의 IP 주소를 직접 기억하여 사용하였다. 이러한 방법은 호스트 수가 늘어나면서 모두 기억하기가 어려워졌고 네트워크를 사용하는 사용자 모두가 기억해야 하는 등의 문제가 있었다. **([그림 1-5] 참조, 상호참조 기능으로 그림 번호 지정)**



[그림 1-5] 초기 이름 해결 방법

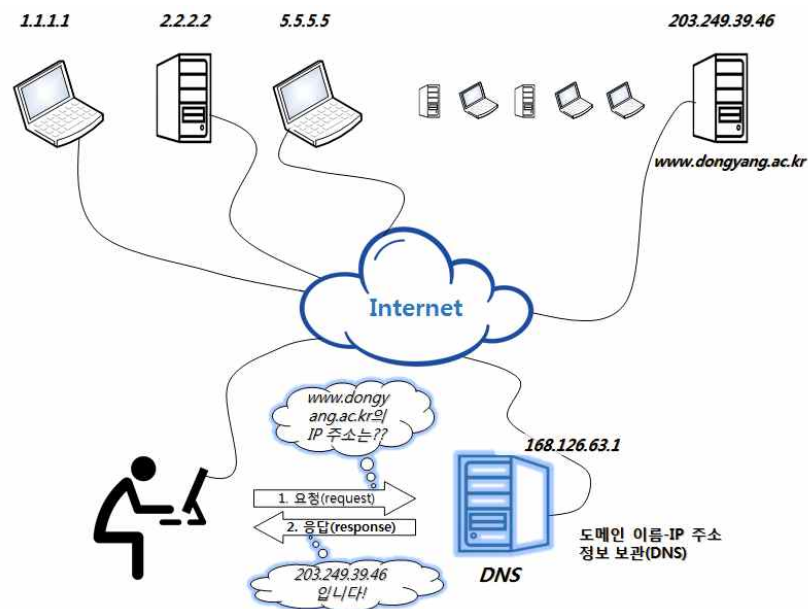


[그림 1-6] hosts 이름 해결 방법

② hosts 사용 이름 해결 방법

인터넷에 접속되는 호스트가 수십~수백 대 규모로 늘어나면서 사용자의 기억력에 의한 이름 해결은 어렵게 되었다. 대신에 사용자 호스트에 호스트 이름에 대응되는 IP 주소를 입력하는 'hosts'파일을 사용하였다. 'hosts'파일은 각 호스트에 대해 IP 주소와 IP 주소에 대응되는 호스트 이름을 한 줄씩 입력한 파일이다. 보통 IP 주소와 호스트 이름 사이에는 탭(tab)을 주로 사용하지만 다른 화이트 스페이스(white space) 문자를 사용할 수가 있으며 '#'를 줄 앞에 붙여 주석(comment)을 표시할 수도 있다.

③ DNS 서버를 이용하여 네트워크 접속



[그림 1-7] DNS 서버를 이용한 이름해결 방법

기하급수적으로 늘어나는 인터넷 상의 모든 호스트에 대해 이름해결을 하는 방법으로 하나의 파일에 도메인 이름과 IP 주소 정보를 기록하는 것은 무리가 있다. 파일 크기가 커지는 것도 문제이고 각 호스트가 동일 내용의 파일정보를 유지한다는 것도 큰 문제이다. 이름해결을 위한 정보를 가지고 있는 'hosts' 파일을 인터넷 상에서 공유하는 방법도 생각해볼 수 있으나 파일의 크기가 커서 이름해결을 원하는 모든 호스트가 번번이 다운로드하기에는 무리가 많다. 인터넷 규모가 커진 현시점에는 이름 해결을 위해 전용 서버를 사용하고 있다. 이 서버가 DNS 서버이다.

DNS 서버는 사용자가 요청(request)하는 특정 도메인 이름에 대해 IP 주소를 응답(response) 반환하는 역할을 하는데, 전화번호를 안내하는 서비스인 '114 서비스'가 사람 이름이나 상호에 대한 전화번호 요청에 해당 전화번호로 응답하는 것과 비슷한 역할이다. 다만 DNS 서버는 전화번호 안내 서비스와 달리, 저장하여야 할 정보가 인터넷 상의 컴퓨터 증가로 엄청나게 크며 이 정보가 수시로 갱신되고 있어서, 하나의 컴퓨터에 집중하여 정보를 저장하지 않고 여러 대의 컴퓨터(DNS 서버)에 분산하여 저장하고 있다.

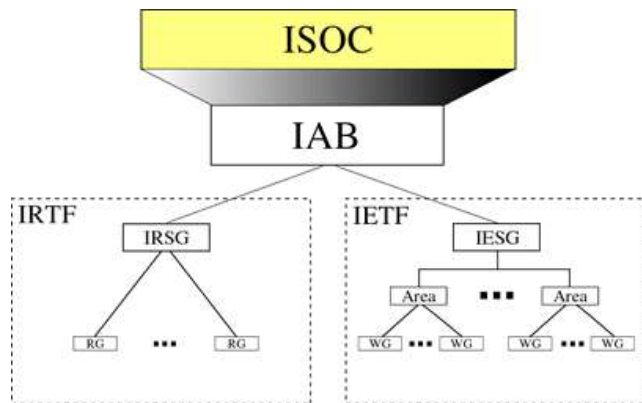
DNS 서버는 인터넷에서 **수시로** 생성, 변경되는 모든 호스트의 도메인 이름이나 IP 주소 정보를 거의 실시간으로 반영하고 있기 때문에 사용자는 더 이상 주소해결에 관하여 신경 쓸 필요가 없어지게 되었다. DNS 환경에서는 도메인 이름만 알고 있으면 언제 어디에서든 해당 호스트의 IP 주소를 획득할 수 있어서 인터넷을 통한 호스트 접속이 가능하다.

1-2 인터넷 관련 기구와 표준화

(1) 인터넷 관련 기구

① ISOC(Internet Society; 인터넷 사회) <http://www.internetsociety.org/>

- ⑦ ISOC는 미국 버지니아 주 레스톤(Reston, Virginia)에 본부를 둔 비영리 전문 회원제 국제기구로 전 세계 100여개 국가의 140개 이상의 조직과 80,000여명의 개인 회원으로 구성되어 있다. IETF나 IAB와 같은 인터넷 관련 여러 위원회의 인터넷 정책이나 방향에 대해 조언하여 인터넷 관련 개발(development)과 발전(evolution) 그리고 인터넷 사용(use of the Internet)을 홍보, 촉진하는 역할을 한다.



[그림 1-8] 인터넷 관련 기구

② IAB (Internet Architecture Board; 인터넷 구조 위원회) <http://www.iab.org/>

- ⑦ IAB는 SOC 자문기구로 인터넷에 사용되는 기술 방향이나 프로토콜의 표준화에 대한 결정을 내리는 기구이다. IAB 회원으로 많은 연구기관과 TCP/IP를 생산하는 기업체가 참여하고 있는데, 인터넷에서 생기는 문제나 요구 사항을 처리하며 TCP/IP의 기술적 방향과 표준화를 추진한다.
- ⑧ IAB의 전신은 1979년 ARPA의 인터넷 프로그램 관리자이었던 빈트 서프(Vint Cerf)가 실험결과를 토론하고 정보를 공유하기 위해 비공식적으로 결성한 ICCB(Internet Configuration Control Board)인데, ICCB가 1984년 인터넷 특정분야를 집중 연구하는 체제로 변경되면서 IAB(Internet Activities Board)가 만들어지고, 1986년 동일한 약자인 IAB(Internet Architecture Board)로 변경되었다.
- ⑨ IAB는 인터넷의 급속한 성장으로 인터넷의 정치적, 상업적 현실을 반영하기 위해 1986년에 인터넷 표준 개발 작업을 수행하는 IETF(Internet Engineering Task Force; 인터넷 기술 전문위원회)와 인터넷 관련 장기 연구를 수행하는 IRTF(Internet Research Task Force; 인터넷 연구 전문위원회) 등의 전문위원회를 포함하는 조직을 구성하였다. IETF와 IRTF에서 제안된 것은 최종적으로 IAB가 승인한다.

③ IETF(Internet Engineering Task Force; 인터넷 기술 전문위원회) <http://www.ietf.org/>

- ⑦ IAB 산하 조직으로 인터넷관련 프로토콜 연구와 개발을 추진하며, 중단기적 기술 문제 중심으로 활동하고 있다. IETF는 여러 개의 영역(area)으로 나누어 활동하는데, 각 영역 관리자(AD; area director)는 IESG(Internet Engineering Steering Group; 인터넷 기술 조정 위원회)에 참여하여 IETF 영역별 연구 활동을 관리하고 인터넷 표준 과정에 대해 책임을 가진다.

④ IRTF(Internet Research Task Force; 인터넷 연구 전문위원회) <http://www.irtf.org/>

- ㉠ IRTF는 TCP/IP 프로토콜과 관련된 장기 기술 문제를 중심으로 연구 활동하는데 IETF에 비해 작은 조직이다.

⑤ IANA(Internet Assigned Numbers Authority) <http://www.iana.org/>

- ㉠ IANA는 USC(University of Southern California) 정보 과학 연구소를 중심으로 활동하는 기관으로, IP 주소와 여러 가지 인터넷 프로토콜의 유일 변수(unique parameter)에 대해 관리와 책임을 가지고 있다.

- ㉡ IANA의 지역별 등록 센터(Regional Registry)는 다음과 같다.

- Asia/Pacific Region : APNIC(Asia-Pacific Network Information Center)
- Americas and Southern Africa : ARIN(American Registry for Internet Numbers)
- Europe and Surrounding Areas : RIPE NCC(Réseaux IP Européens Network Coordination Centre)

⑥ ICANN(Internet Corporation for Assigned Names and Numbers) <http://www.icann.org/>

- ㉠ 인터넷이 세계의 네트워크로 변화하면서 1998년 11월에 미국 주도의 인터넷 주소 관리 체계를 전 세계 인터넷 관계자가 참여하는 주소 관리 체계로 개편하였는데, 이와 관련한 기구가 ICANN이다.

- ㉡ ICANN은 도메인 이름, IP 주소, 인터넷에서의 분쟁, 전자상거래 등과 같은 정책을 수행하는데, 현재 IANA나 다른 그룹에서 이미 수행하고 있는 여러 과제에 대해서 이관을 진행하고 있다.

- ㉢ ICANN의 주요 기능은 인터넷 도메인 이름(Internet domain name), IP 주소 숫자(IP address number), 프로토콜 변수(protocol parameter)와 포트 번호(port number), 인터넷 루트 서버(Internet root server) 시스템의 안정적 운영 등이다.

⑦ InterNIC(Internet Network Information Center) <http://www.internic.net>

- ㉠ InterNIC은 인터넷 **백본망**을 구성하고 있는 NSF(미국 국립과학재단)가 TCP/IP와 인터넷 관련 정보를 관리 배포하고 행정적 관리를 지원하기 위하여 1993년 AT&T, NSI(Network Solutions, Inc.), General Atomic과 협약하여 만든 기구이다.

- ㉡ AT&T는 InterNIC Directory 및 Database 서비스를 제공하였는데 1998년 3월 31일 NSF와의 협력 계약이 만료되면서 서비스를 중단하였다. NSI는 InterNIC과의 계약으로 .com .net .org .edu에 한해서 도메인 이름 등록업무를 독점적으로 수행해 왔으나 1998년 11월부터 일반최상위도메인(gTLD) 등록인가 절차 및 운영을 새로운 비영리 통합기구 ICANN으로 이전하였다. General Atomics는 인터넷 정보서비스를 제공하였는데 계약 기준에 부합하지 못하여 1994년 12월 계약에서 탈락했다.

- ㉢ InterNIC의 인터넷 IP번호 할당 및 ASN 할당, 역 DNS 영역 (in-addr.arpa) 관리 작업은 1997년 12월에 ARIN(American Registry for Internet Numbers: 미국 인터넷 번호 등록)으로 이전되었다.

⑧ KrNIC(한국인터넷정보센터) : <http://www.krnic.net/>

- ㉠ KrNIC은 한국 내 인터넷 정보 센터로 다음과 같은 기능을 가지고 있다.

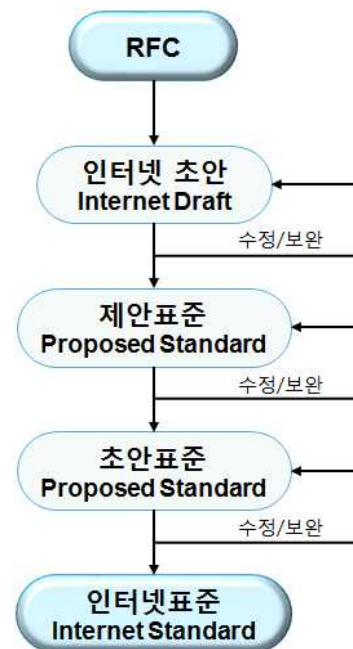
- 도메인 이름, IP (인터넷 프로토콜) 주소 등 인터넷 주소에 관한 정책 개발
- 도메인 이름 등록업무 및 IP주소, AS번호 등의 할당업무

- 국가 루트 네임서버와 인터넷주소 인물정보 데이터베이스의 관리 및 운영
- 인터넷 프로토콜, 차세대 인터넷 주소, 다국어 도메인이름 시스템 등 인터넷 관련 연구개발
- 국제 인터넷 주소관련기구에 대한 참여 및 협력활동
- 국내.외 **인터넷** 활성화를 위한 체계적인 통계산출
- 전문 전화 상담원을 통한 도메인등록 교육
- 인터넷의 효율적 운영과 **이용** 활성화를 위한 지원활동

(2) 인터넷 관련 표준화

① 인터넷 관련 표준화 과정

- ㉠ 인터넷 표준은 인터넷에 적용되는 기술 또는 방법에 관한 규격을 의미하며 IETF에서 만들고 공지한다.
- ㉡ 새로운 인터넷 표준화 과정의 시작은 RFC이다. 제안된 RFC에 대해서는 IETF 워킹그룹 회의와 네트워크를 통해 누구나 자신의 견해를 표명하는 방식으로 논의한다. 적절한 수정과 논의를 거쳐서 인터넷 초안(Internet **Draft**)이 작성되는데 이 초안은 RFC에 대해 제시된 모든 주석을 통합한 형태라 할 수 있다. IETF는 이 과정 후 제안된 RFC를 제안표준(Proposed Standard)으로 적용할지를 결정하게 된다.
- ㉢ 다음 단계인 제안표준 단계에서는 최소 6개월 이상 두 군데 이상에서 이 표준을 실제로 구현, 테스트하여 테스트 과정에서 발생된 문제점을 보완하게 된다. 테스트 및 보완이 완료된 후에 초안표준(Draft **Standard**)이 작성되는데 이 단계의 표준은 4개월 이상 실제 구현과 테스트가 수행된다.
- ㉣ 마지막 단계는 인터넷표준(Internet Standard) 채택 단계인데, IETF는 이 표준을 요구하는 모든 사이트에서 구현하고 테스트하여 RFC 6410이 규정하는 요구조건이 충족되는 경우 인터넷표준으로 채택한다.
- ㉤ 이와 같이 인터넷 관련 표준화 과정은 RFC와 인터넷 초안, 제안표준, 표준초안, 인터넷표준 등의 단계로 처리되어 왔는데, 2011년 10월 이후에는 초안표준 단계를 마지막 단계인 인터넷표준 단계에 흡수하여 처리하고 있다.
- ㉥ 참고로 TCP/IP 프로토콜 가운데에는 표준화 상태가 제안표준 또는 초안표준 상태인 것도 다수 있다.



[그림 1-9] 인터넷 관련 표준화 과정

② RFC(Request for Comments)

[7] **RFC**의 작성자는 IAB 회원이며 **RFC**가 작성되고 있는 동안에는 학문적 연구 논문으로 참고 될 수 없다.

- ㉠ 인터넷 상의 작업문서나 프로토콜 제안, 표준 TCP/IP 프로토콜은 RFC라는 일련의 기술 보고서 형태로 제시된다.
- ㉡ RFC는 새로운 TCP/IP 기능에 대한 문제 제기부터 표준화까지의 흐름을 기록, 공개하는 문서로 '이 내용에 대해 의견을 주십시오'의 의미를 가지고 있다. RFC는 새로운 특정기능의 제안이 일반적이지만 기존 표준을 수정하는 경우도 있다.
 - RFC 예시
IP(791), UDP(768), TCP(793), Telnet(854, 855), FTP(959), SMTP(821), ARP(826), RARP(903), ICMP(792), RIP(1053), DMS(1034, 1035), SNMP(1157), MIB2(1213), NTPV2(1119), PPP(1661), POP3(1725), MIME(1521), NNTP(977), BOOTP(1532), DHCP(1531), SIP(1710)

2. HTML

2-1 인터넷의 웹 서비스

(1) 웹 서비스 개괄

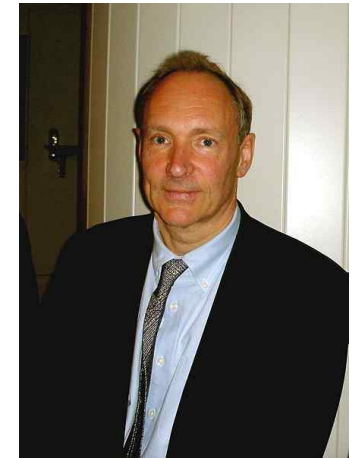
웹 서비스가 인터넷 서비스의 대표로 자리매김하는데 있어서는 HTML이라고 하는 문서 연결방식과 다양한 미디어를 표현하는 웹 브라우저의 발전, 그리고 인터넷에 분포되어 있는 광범위한 자료를 찾아주는 검색엔진의 등장이 큰 역할을 하였다.

웹 문서를 저장, 제공하는 웹 서버와 웹 문서를 요청, 열람하는 웹 클라이언트 간에는 HTML 기반의 문서에 기반하고 있다. 이러한 HTML의 시작은 1989년 3월 스위스 소재 유럽원자핵연구기구(CERN)의 팀 버너스 리^[8](Tim Berners Lee)가 문서 하이퍼텍스트(Hypertext) 방식의 HTML 문서 형식을 제안하면서부터이다.

인터넷 초기단계에는 전자우편, FTP, TELNET을 비롯한 대부분의 인터넷 서비스는 문자기반으로 제공되어서 사용 명령어에의 접근이 어려웠으며 콘텐츠 또는 표현에 제약이 많았다. 웹 서비스의 등장으로 문자기반 서비스뿐 아니라 이미지나 음악을 포함하는 멀티미디어 기반의 서비스도 함께 제공되면서 명령어나 콘텐츠 표현이 다양하고 편리해졌다. 웹 서비스의 중심에 서 있는 HTML은 다양한 콘텐츠의 표현이나 인터넷을 통한 정보 접근에 있어서 가장 중요한 핵심 수단되어 있다.

오늘날 인터넷 서비스 그 자체가 웹 서비스로 인지될 정도로 웹 서비스는 인터넷 서비스 가운데 가장 많이 사용되고 있다. 인터넷 서비스는 여타 서비스와 마찬가지로 웹 서버와 웹 클라이언트 간의 정보 교환으로 구현되고 있다. 웹 서버와 웹 클라이언트 간의 정보로 서비스가 이루어진다. 웹 브라우저(web browser)라고 부르는 웹 클라이언트 프로그램은 인터넷을 통해 HTML 문서와 이미지 등을 웹 서버로부터 받아 사용자에게 보여주는 프로그램이다.

최초의 웹 브라우저는 1990년 팀 버너스리가 만든 'World Wide Web^[9]'라 불리는 브라우저이며, 그래픽 기반의 최초 웹 브라우저는 1993년 미국



[그림 2-1] 팀 버너스 리

[8] 팀 버너스 리(Tim Berners-Lee) : 영국 태생 엔지니어(1955). 스위스 소재 CERN(유럽원자핵연구기구)에 근무하면서 입자물리학 관련 연구결과물 사용 및 공유를 위해 하이퍼텍스트 방식을 제안하고 이를 입증하기 위한 시스템 ENQUIRE를 개발(1980)하고 하이퍼텍스트를 인터넷에서 구현(1989). 최초의 웹 브라우저 'World Wide Web'과 웹 서버 'CERN HTTPd'를 개발(1990). MIT에 W3C를 창립(1994). 영국 왕립학회 회원(2001). 위대한 영국인 100인에 선정(2002). 튜링 상(Turing Award)(2016)

†CERN(European Organization for Nuclear Research) : (프랑스어) *Conseil Européen pour la Recherche Nucléaire*

일리노이 대학의 NCSA^[10] 연구기관에서 만든 모자이크(Mosaic)이다. 웹 사용을 폭발적으로 확산되게 하고 대중화하도록 한 것은, 1994년 모자이크 개발자인 마크 앤드리슨(Marc Andreessen)이 NCSA를 떠나 넷스케이프(Netscape)사에서 개발한 ‘넷스케이프 네비게이터(Netscape Navigator)’이다. 넷스케이프 네비게이터는 웹 브라우저 시장을 순식간에 90% 가까이 장악하면서 웹 대중화에 크게 기여하였는데, 모자이크와 달리 한번 받은 데이터를 로컬 컴퓨터의 캐시에 저장하는 방법을 사용하여 느린 인터넷에 적절하게 보완하는 장점을 가지고 있었다.

(2) 웹 브라우저(web browser)

① 웹 브라우저 종류

㉠ 월드와이드웹(World Wide Web)

세계 최초의 웹 브라우저는 1991년 2월 26일에 팀 버너스리가 개발한 ‘월드와이드웹(World Wide Web)’이다. 이 웹 브라우저는 위지윅(WYSIWYG^[11]) HTML 편집 기능도 구현하고 있는데, 인터넷을 의미하는 ‘월드 와이드 웹’과의 혼동을 피하기 위해 ‘넥서스(Nexus)’ 이름이 변경되었다.

㉡ 모자이크(Mosaic)

최초의 그래픽 지원 웹 브라우저는 모자이크(Mosaic)로 1993년 미국 일리노이대 NCSA 연구소의 마크 앤드리슨(Marc Andreessen)과 에릭 비나(Eric Bina)가 공동으로 개발하였다. 모자이크 개발 책임자였던 마크 앤드리슨이 1994년 연구소를 나와 넷스케이프사를 설립하고 ‘넷스케이프 네비게이터(Netscape Navigator)’ 웹 브라우저를 개발하였는데, 그 후 모자이크는 스파이글래스(Spyglass)회사에 인수된다. 이 회사를 마이크로소프트가 1995년에 저작권료를 지급하고 기존 모자이크 웹 브라우저의 소스 코드를 가지게 되는데, 마이크로소프트는 이를 활용하여 인터넷 익스플로러 웹 브라우저를 개발하게 된다.



㉢ 넷스케이프 네비게이터(Netscape Navigator)

1994년 넷스케이프사가 개발한 넷스케이프 네비게이터는 당시 큰 인기를 받고 있던 모자이크를 대신하여 웹 브라우저 시장의 약 90%를 점유하였다. 웹 서비스의 대중화에도 큰 역할을 하였던 넷스케이프 네비게이터는 1995년 마이크로소프트 인터넷 익스플로러(Internet Explore)가 시장에 나오면서 시장을 잃게 된다. 마이크로소프트는 자사의 웹 브라우저 IE를 윈도우 운영체제에 포함시켜 무료로 배포하는 전략으로 시장을 장악하였는데, 이의 영향으로 시장점유율이 급격히 떨어지게 되었고 결국 1999년 AOL



[9] 'WorldWideWeb' : 나중에 'Nexus'로 이름을 바꿈

[10] NCSA : National Center for Supercomputing Applications

[11] WYSIWYG/WYSIWYG : "what you see is what you get"의 줄임말로, 화면에 보이는 대로 출력되는 응용프로그램이라는 것을 나타낼 경우 사용한다.

에 인수되면서 시장에서 사라졌다. 넷스케이프사 출신의 일부 개발자들은 별도로 모질라 커뮤니티를 만들고, 2002년 파이어폭스 웹 브라우저를 출시했다.

㉔ 인터넷 익스플로러(Internet Explore, IE)

인터넷 익스플로러는 마이크로소프트사가 1995년에 개발한 웹 브라우저이다. 마이크로소프트가 윈도우 운영 체제에 IE를 기본으로 포함하면서, IE는 웹 브라우저 시장을 주도하게 되는데 1999년에 시장 점유율이 최고가 되고 2002년경에는 95%까지 장악하였다. 그러나 웹 표준 준수 및 성능 등의 요구가 대두되면서 IE는 모질라 파이어폭스, 구글 크롬 등에 점차 시장을 내어 주게 된다. 현재는 전 세계 기준으로 보면 크롬, 파이어폭스 다음으로 3위에 머무르고 있다.



㉕ 오페라(Opera)

오페라는 1994년 노르웨이의 텔레너(Telenor) 통신회사가 처음 개발한 후 1995년부터 오페라 소프트웨어(Opera Software ASA)가 관리하는 웹 브라우저이다. 2016년에는 중국 투자자 그룹 'Golden Brick Capital Private Equity Fund I Limited Partnership'에 인수되었다. 오페라는 기본 기능에 충실한 웹 브라우저로 처리 속도가 빠르고 보안이 우수하며, 휴대전화나 PDA, 노트북 컴퓨터의 경우 배터리 절전에 성능이 효과적인 것으로 평가를 받고 있다.



㉖ 모질라(Mozilla)

모질라 웹 브라우저는 모질라 애플리케이션 스위트(Mozilla application suite)라고도 불리는데 넷스케이프사에서 나와 설립한 모질라 재단에서 2002년에 개발한 웹 브라우저이다. '모질라'라는 이름은 '모자이크 킬러(Mosaic killer)'를 의미하는데, 당시 가장 인기 있던 웹 브라우저인 모자이크를 능가한다는 의미와 허구의 괴물 '고질라'에서 연유한다고 한다. 모질라 웹 브라우저는 이메일 클라이언트, 뉴스 클라이언트, HTML 편집기, IRC 클라이언트를 포함하고 있다.

㉗ 사파리(Safari)

사파리(Safari)는 애플이 2003년에 개발한 웹 브라우저이다. 사파리는 아이튠즈와 유사한 북마크 관리 체계를 가지고 있고, 애플의 쿼타임 멀티미디어 기술과 통합되어 있으며, 탭 브라우징 인터페이스를 사용한다. 구글 검색 상자는 사파리 인터페이스의 기본 요소이며, 웹 주소 자동완성과 웹 페이지 텍스트 영역의 맞춤법 검사를 지원한다.



㉘ 모질라 파이어폭스(Mozilla Firefox)

모질라 파이어폭스(Mozilla Firefox)는 모질라 재단과 모질라 코퍼레이션이 2004년 개발한 웹 브라우저이다. 이 웹 브라우저는 2017년 현재 전 세

계 웹 브라우저 시장 점유율이 구글 크롬 다음으로 높으며 데스크톱 컴퓨터의 경우 약 14% 수준이다.

㉠ 구글 크롬(Google Chrome)



구글 크롬(Google Chrome)은 구글이 2008년 개발한 웹 브라우저이다. 구글 크롬은 간단하고 효율적인 사용자 인터페이스를 **제공하며** 안정성과 속도, **그리고** 보안성 면에서 장점을 가지고 있다. 크롬이란 뜻은 원래 그래픽 사용자 인터페이스에서 창틀을 가리키는데, 여기서는 브라우저 틀 영역을 가리키며, 이 영역을 최소화 시키자는 목표로 크롬이라고 이름을 지었다고 한다. 이 웹 브라우저는 2017년 현재 전 세계 웹 브라우저 시장 점유율이 가장 높으며 데스크톱 컴퓨터의 경우 약 63%이다.



㉡ 웹 브라우저 시장 점유율

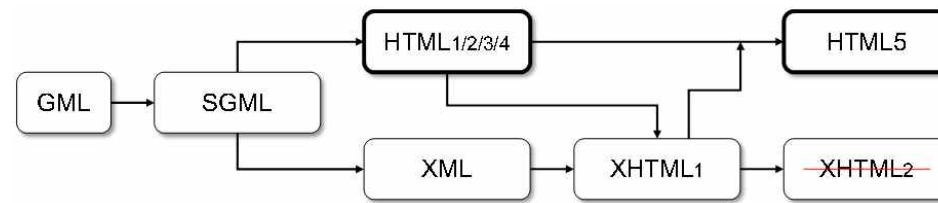
㉠ 전 세계 기준으로 2017년 현재 많이 사용되고 있는 웹 브라우저는 크롬, 파이어폭스, IE, 사파리, 마이크로소프트 에지^[12](Edge, 2016), 오페라(Opera, 1994) 순으로 크롬이 60%를 넘는 월등한 시장점유율을 보이고 있다. 우리나라 경우는 크롬이 54%를 넘으면서 가장 많이 사용되고 있으며 그 다음이 IE, 사파리, 에지, 파이어폭스, 오페라 순이다.

(3) 웹 표준 HTML

HTML은 마크업(markup) 언어의 하나로 1991년 CERN에 근무하던 팀 버너스 리(Tim Berners-Lee)가 하이퍼텍스트에 대한 아이디어를 제안하였다.

마크업 언어는 IBM이 1960년대에 개발한 GML(Generalized Markup Language)과 국제표준화기구(ISO)가 1986년에 문서용 마크업 언어를 정의하기 위해 표준화한 SGML(Standard Generalized Markup Language)에서 시작되었다. SGML은 ISO가 정부나 항공우주, 인쇄와 출판 등의 산업분야에서 기계판독형(machine-readable) 문서를 공유할 목적으로 국제표준으로 설계되었는데, HTML이나 XML, XHTML 등의 마크업 언어의 근원으로 사용되었다.

[12] 마이크로소프트(Microsoft)사가 개발한 웹 브라우저로 2015년에 에지(Edge) 초판, 2016년에 정식 제품을 발표하였다. IE 11 및 IE Mobile을 대체하는 웹 브라우저로 Windows 10, Windows 10 Mobile 및 Xbox One 콘솔의 기본 웹 브라우저이다. ActiveX 및 브라우저 도우미 개체 등은 지원하지 않으며 Adobe Flash Player, PDF Reader, 어도비 플래시 플레이어, asm.js 등은 기본으로 지원하고 있다.



[그림 2-2] HTML 표준의 설계 흐름

- ㉠ 1993년 6월 IETF는 버너스 리(Berners-Lee)와 댄 코놀리(Dan Connolly)가 제안한 “Hypertext Markup Language (HTML)”을 인터넷 초안(Internet-Draft)으로 채택하였다. SGML 기반 문서 형식으로 제안된 최초의 HTML은 20개 요소(element)를 포함하고 있었다. 이 무렵 데이브 라게트(Dave Raggett)가 표(table)와 양식(form) 기능을 추가하는 또 다른 인터넷 초안 “HTML + (Hypertext Markup Format)”을 제안하였다. IETF는 HTML 및 HTML+ 인터넷 초안이 마무리되면서 1994년에 두 초안을 함께 검토하는 HTML 작업그룹(Working Group)을 만들고 인터넷 표준으로는 최초로 HTML 2.0을 1995년 11월에 발표하였다. HTML 2.0은 IETF의 RFC 1866으로 관리되면서 지속적인 확장이 이루어졌다.
- ㉡ 한편 1996년에 HTML 표준에 대한 유지관리가 IETF에서 W3C(World Wide Web Consortium)로 넘겨진 후 IETF의 HTML 작업그룹이 해산되고, IETF에서 논의되고 있던 HTML+ 바탕의 HTML 3.0 표준화는 마무리되지 못하였다. 제안된 HTML 3.0은 HTML 2.0보다 더 구조적이고 기본 표현방법이 향상된 기

능을 가지고 있었지만 완성되지 못하였다.

- ㉢ W3C는 1997년 1월에 HTML 표준이 이관된 후 처음으로 HTML 3.2를 W3C 권고(recommendation)로 발표하고 1997년 12월에는 HTML 4.0을 발표하였다. HTML 4.0은 HTML 4.0의 사용되지 않고 있는 일부 요소에 대한 처리, 프레임셋 요소의 처리에 대한 내용이 추가되었다.
- ㉣ W3C는 1999년 12월 HTML 4.01 권고를 마지막으로 더 이상의 추가 표준화 작업은 추진하지 않았다. 다만, 2001년 5월 HTML 4.01의 오류에 대한 정정에 대해 한차례 발표하였다. 즉 HTML 관련한 표준은 HTML 4.01이 발표된 1999년 이후부터 HTML5가 정식 발표된 2014년까지 약 15년 동안은 HTML 4.01이 표준으로 사용되었다.
- ㉤ W3C는 1998년에 HTML 표준은 더 이상 업데이트 하지 않을 것을 선언하면서 차세대 웹 표준으로 XML 기반의 XHTML을 추진하였다. W3C가 이처럼 HTML 표준에 대한 정책을 변경하게 된 것은, 그동안 웹 문서 작성에 사용되어 왔던



HTML이 표준을 엄격하게 적용하지 않아 발생될 수 있는 미래의 문제점 때문이었다. 그동안 HTML 문서는 문법적 오류가 포함되어 있더라도 일단은 웹 브라우저에 보이도록 하고 있어서 표준을 지키지 않은 웹 문서가 지속적으로 생산되고 있는 문제점이 있었다. 프로그램에서의 **처리의** 용이성이나 **문서의** 확장성 측면에서는 표준을 지키는 것이 훨씬 유리하기 때문이었다. 그리고 동일

한 웹 문서가 웹 브라우저의 종류에 따라 조금씩 다르게 표시되는 문제도 있었는데 웹 개발자는 이에 따른 개발의 어려움을 상존하였었다. 즉 W3C는 HTML 문법에 대한 관용은 누적되어가는 웹 문서가 장치 정보로서의 가치를 떨어뜨리게 되어 그 미래가 우려스럽다는 판단을 하였다.

㉞ (이 영역은 3개의 단으로 표현하고 있으며 쪽의 아랫부분이 동일 선상에 있다)

XHTML은 HTML과 달리 엄격한 규칙을 요구하고 있는데 다음의 주요 내용을 포함하고 있다.

㉞ • 대소문자 구분하고 태그는 반드시 소문자를 사용

• 잘못된 태그 사용에 대해서는 명백한 오류로 처리

• 속성값은 따옴표로 표시

• 빈 태그는 </>로 종료

2002년에는 W3C가 HTML과의 호환을 포기하는 XHTML 2를 제시하게 되는데, 이는 웹 브라우저 개발자나 웹 개발자 모두에게서 외면 받으면서 표준으로의 확산

에 실패하게 되고 2009년에는 XHTML을 포기, 폐기하게 되었다. XHTML은 웹 서비스가 **등장하면서** 생성된 방대한 과거 자료를 제대로 볼 수 없는 문제가 있기 때문에 결정적인 약점을 가지고 있었다.

㉞ HTML의 업그레이드가 정체되고 있던 즈음에, HTML과 관련 기술에 관심 있던 모질라 재단, 오페라 소프트웨어, 애플 소속된 전문가들이 2004년에 WHATWG(Web Hypertext Application Technology Working **Group**)을 구성하여 **HTML과** 호환성을 유지하면서 HTML 기능을 업그레이드하는 연구를 하

게 되었다.

㉞ HTML 연구 주도권을 WHATWG가 잡게 되면서, 2007년 W3C는 새로운 HTML 표준을 위해 WHATWG와 협력하기로 하였다. 새로운 버전의 HTML 표준은 WHATWG의 HTML5를 개발의 시작점으로 하고 이름도 HTML의 5번째 주요 버전을 의미하는 HTML5로 부르기로 하였다.

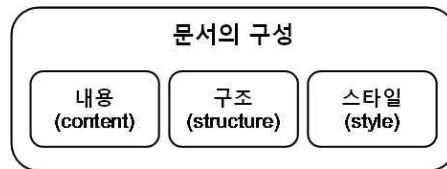
㉞ 2010년 W3C는 WHATWG와 함께 HTML5 **초안을** 작성, 공개, 검토하였으며 2014년 **10월에** 완성된 형태의 HTML5를 발표한 후 2016년 11월에 HTML5.1 권고안을 발표하였다.

2-2 HTML의 기초

(1) HTML 문서의 구조

① 마크업(Markup) 언어의 개념

- ㉠ 마크업은 문서 내용 이외에 문서 구조를 정의하거나 표현하기 위해 정의한 부가적인 정보를 의미한다. 일반적으로 문서는 내용(content), 구조(structure), 스타일(style) 등으로 구성되는데 문서의 마크업은 각 어플리케이션에서 정의하는 고유의 방식으로 이루어진다. 동일한 내용의 문서를 작성하면서 어플리케이션에 따라 문서의 구조나 문서의 스타일 지정을 위한 마크업이 다르다는 것을 의미한다.



[그림 2-4] HTML 문서 구성

- ㉡ 한정된 공간에서 독립적으로 문서를 작성하는 경우에는 큰 문제가 없을 수 있지만 인터넷 환경에서 문서를 작성하고 공유하는 경우에는 적용하는 마크업에 대한 표준화가 불가피하다. 이러한 결과로 SGML, XML, HTML, XHTML 등과 같은 표준화된 마크업 언어가 개발되게 되었다.
- ㉢ SGML(Standard Generalized Markup Language)은 문서 내용이나 구조를 정의하는 메타언어로 시스템 종류에 상관없이 다양한 문서 데이터를 상호 교환하도록 하는 표준 방법을 제공한다. SGML은 처리방법에 대한 표준은 아니며 단지 논리적인 정보만을 제공하기 때문에 포맷 정보는 제공하지 않는다. 미국 국방성의 CALS 프로젝트, 미국 출판협회, 유럽공동체 출판국, 옥스퍼드 대학 출판부, 미국 화학회, 수학회 등에서 SGML이 사용되고 있다. 그러나 SGML은 많은 장점이 있는 반면에 구성과 문법이 복잡해서 문서를 작성하기 힘들다는 문제가 있다. SGML의 이러한 복잡함을 줄이고자 SGML의 일부 태그를 정의해서 웹 문서에 적합하도록 만든 마크업 언어가 HTML이다.



[그림 2-3] HTML 문서 구성 예시

② HTML 문서의 구성

- ㉠ HTML은 웹에서 사용되는 **문서를** 작성하기 위한 컴퓨터 언어의 하나로 웹 서비스를 제안한 초기부터 사용되어 왔다. 웹 문서는 문서의 내용(content)과 구조(structure), 문서의 모양과 디자인(design), 문서의 동작(function)과 상호작용(interactive) 등 3가지 **요소로** 구성되어 있다. HTML은 이 가운데 문서의 내용과 모양 등 2가지 요소를 처리하고 문서에서의 동작과 관련해서는 주로 자바스크립트로 처리되어 왔다.
- ㉡ 최근 HTML5가 등장하면서부터는 각 요소에 대한 적용 기술이 일부 변경되어, 문서의 내용과 구조에 대해서는 HTML이, 문서의 모양이나 디자인은 CSS이 처리하도록 하고 있다. 따라서 웹 **문서를** 작성하기 위해서는 기본적으로 HTML 사용에 대해 정확하게 이해하고 학습하여야 한다.

③ HTML 태그(tag)

- ㉠ HTML은 텍스트 파일(text file)^[13] 형식으로 문서 내용을 나타내고 있다. HTML을 **사용하여** 생성한 웹 문서는 확장자로 '.html' 또는 '.htm' 을 사용한다.
- ㉡ 웹 문서는 웹 **문서를** 작성하기 위해서는 먼저 문서에 표시하고자 하는 정보를 텍스트 편집기를 이용하여 입력한다. 그 다음에 문서 구조나 문서 표현을 위해 순수 텍스트 문서에 마크업(markup) 기호를 부가하는데, 이 기호를 태그(tag) 또는 요소(element)라 한다. HTML 태그는 각괄호(angle brackets, < >)를 사용하여 웹 문서 정보와 구별하고 있으며, 문서 구조 및 문서 표현의 처리 범위를 표시하기 위해 시작 태그와 종료 태그 등 쌍(pair)으로 사용하고 있다. 시작 태그는 < >로 <종료 태그는 </ >로 표시한다.
- ㉢ 예를 들어 문서 내의 글자'텍스트 파일(text file)'에 대해 밑줄을 표시하고 '텍스트 파일' 범위에 대해서는 굵은 글자(bold) 형식으로 표시하고 할 경우, 밑줄 범위에 대해서는 <u>와 </u>을 표시하고 굵은 글자 범위에 대해서는 와 로 표시하여 범위의 시작과 끝을 표시한다.

[13] 파일 종류를 구분하는 방법 중 하나로 텍스트 파일(text file)과 이진 파일(binary file)이 있다. 텍스트 파일은 ASCII 문자의 도형문자(graphic character)나 Unicode 문자 등 문자열 **데이터만**으로 구성된 파일을 의미하고, 이진 파일은 컴퓨터에 저장과 처리 목적으로 이진 형식으로 인코딩된 데이터를 포함하는 파일을 의미한다. 이진 파일의 예로 문서파일에 문자열 데이터 뿐 아니라 통신 또는 문서형식을 위한 제어문자(control character)를 포함하는 파일이 있다.

2-3 CSS 익히기

(1) CSS 개요

CSS는 ‘Cascading Style Sheet’의 약자로 웹 문서의 모양이나 디자인을 설정하기 위한 목적으로 사용된다. CSS는 HTML 태그로 만들 문서의 내용을 화면이나 종이 또는 다른 미디어장치에 어떻게 표시되는지를 설정하는 역할을 한다. CSS는 동일한 내용의 웹 문서라도 여러 개 레이아웃으로 조정할 수도 있다.

CSS는 HTML5가 공식적으로 표준화되기 전부터 웹 문서의 모양이나 디자인에 사용되어 왔지만, 웹 **문서** 작성에 필요한 역할을 3가지로 구분하고 CSS에게 웹 문서의 모양이나 디자인 역할을 본격적으로 요구하게 된 것은 HTML5가 등장하면서부터이다. 웹 **문서** 작성에 있어서 필요한 역할을 웹 문서 **내용** 작성, 문서 내용에 모양 및 디자인 적용, 문서의 동작 구현 등 3가지로 구분하고 각각 HTML 태그, CSS, 자바스크립트 (JavaScript)가 처리하도록 하게 되었다. CSS 관련 규격은 W3C(World Wide Web Consortium; www.w3c.org)에서 표준화를 주도하고 있다.

(2) CSS 시작

CSS는 애초에는 웹 문서에서 웹 문서의 표현 부분을 분리하기 위해 설계되었다. 문서의 레이아웃이나 색상, 폰트 등과 같은 문서 표현부분을 별도의 스타일시트 파일(CSS 파일; 확장자 .css)로 관리함으로써, 웹 문서에의 **접근** 편의성을 개선하고 여러 개의 웹 문서가 동일한 표현 형태를 공유할 수 있도록 하였다. CSS 파일을 공유하는 많은 웹 문서에서 문서의 구조나 디자인을 변경하고자 할 경우에는 웹 문서의 HTML 태그는 그대로 두고 공유하는 CSS 파일의 내용만 간단히 수정하기만 하면 된다.

CSS를 처음 제안한 것은 1994년 10월에 팀 버너스 리(Tim Berners-Lee)와 함께 CERN에 근무하였던 노르웨이인 하콘 비움 리(Håkon Wium Lie)이다. W3C에서 표준으로 처음 권고한 것은 1996년 12월 발표된 CSS1이다. 그 후 W3C는 지속적으로 새로운 기능과 다양한 도구를 추가하였는데 1998년 5월에 CSS2, 2005년 12월에 CSS3을 차례로 표준으로 발표하였다. CSS3에서는 스타일 기능이 모듈화 되어 전체 모듈을 지원하거나 일부 모듈만을 지원할 수 있도록 하고 있다. CSS4는 아직 통합된 하나의 CSS4 규격은 있지 않은 상태이지만 CSS3 모듈인 이미지 값(image value)이나 배경(background), 경계선(border), 선택자(selector) 기능에 기반 한 CSS4 모듈을 준비하고 있다.

① CSS3 기능별 모듈

- ㉠ 선택자 Selectors
- ㉡ 박스 모델 Box Model

- ㉔ 배경 및 경계선 Backgrounds and Borders
- ㉕ 텍스트 효과 Text Effects
- ㉖ 2차원/3차원 변환 2D/3D Transformations
- ㉗ 애니메이션 Animations
- ㉘ 다중 칼럼 레이아웃 Multiple Column Layout
- ㉙ 사용자 인터페이스 User Interface

(3) CSS의 장점

웹 **문서를** 작성하면서 문서 표현 관련한 정보로 CSS를 사용하면 다음과 같은 장점이 있다.

- ① 웹 문서 편집이 다양하고 용이하다. 글꼴, 자간, 행간, 색상, 배경, 테두리, 레이아웃, 마우스 커서(cursor) 형태 등을 문서편집기(Word Processor) 수준으로 처리할 수 있다.
- ② 여러 개의 웹 문서를 통일감 **있게** 작성할 수 있다. 문서의 스타일 정보를 CSS 파일로 저장하고 적용할 수 있어서 웹 문서에 일관된 형식을 적용 할 수 있다.
- ③ 많은 웹 문서를 일괄적으로 쉽게 관리할 수 있다. 스타일 정보를 담고 있는 CSS 파일을 공유하는 웹 문서의 경우, 문서 내용과 상관없이 문서의 모양을 한꺼번에 변경하기 위해서는 CSS 파일의 수정만으로도 가능하다.

아래 3개의 웹 문서는 내용은 모두 같으나 스타일을 지정하는 CSS 내용만을 서로 다르게 적용하여 만든 문서이다.

(4) CSS 구문과 선택자

① CSS 구문(syntax)

스타일시트의 기본 형식은 **선택자(selector)**, **속성(property)**, 속성값(value) 등 **3부분으로** 구성된다. 속성과 속성값을 선언(declaration)이라고 부르는데 스타일시트는 선택자와 선언 블록(declaration **block**)으로 구성되었다고도 할 수 있다.

Selector {Property:Value; ...; Property:Value;}

(선택자) (속성) (속성 값) (속성) (속성 값)

(예) **body {font-size:15pt; color:blue}**

선택자
(selector) 속성
(property) 값
(value)

[그림 2-5] CSS의 구문(syntax)

선택자(selector)는 웹 문서 내에서 스타일을 적용하려는 대상을 지정한다. 기본적으로 CSS 표준은 선택자로 스타일 적용의 대상으로 HTML 태그를 사용한다. 예를 들어 <p> 태그에 스타일 적용을 원하는 경우에는 선택자가 'p'이며, <body> 태그에 스타일을 적용하고자 하는 경우에는 선택자가 'body'이다. 선택자

선택자에서 지정한 태그에 스타일을 지정하기 **위해서 속성과** 속성값을 사용한다. CSS 속성(property)은 스타일을 적용하고자 **하는** 속성을 지정한다. 선택자 태그 내에서 사용하는 문서의 폰트 크기를 설정할 **때는** 속성 'font-size'를 사용하고, 색상을 지정할 **때는** 속성 'color'를 사용한다.

속성값(value)은 **설정된** 속성의 값을 **표시한다**. 속성 'font-size'에 **대한** 속성값의 예로는 '15pt'이다. 'color'의 속성값으로는 'blue'와 같은 색상이름이다.

선택자, 속성, 속성값을 표기하는 방법은 선택자를 제일 앞에 쓰고 중괄호 '{', '}' 내에 **적용할 속성과** 속성값을 입력한다.

속성과 속성값 사이에는 콜론 ':'을 사용하여 구분하며 여러 **개의** 속성을 한꺼번에 적용하는 **경우에는** 속성값 다음에 세미콜론 ';'을 사용한다.

② CSS 선택자(selector)

CSS의 선택자로 기본적으로 HTML 태그가 사용된다. CSS에서 선택자로 HTML 태그를 지정하면 웹 문서에서 표시되는 해당 태그의 내용은 모두 CSS가 설정한 스타일을 가지게 된다.

여러 개의 태그가 동일한 스타일을 가지는 경우나 동일 태그에 대해 여러 종류의 스타일을 지정하는 경우, 모든 태그에 대해 하나의 스타일을 적용하는 경우 등 CSS 선택자를 표시하는 여러 방법은 다음과 같다.

㉠ 태그 선택자(tag selector)

HTML 문서 내 있는 특정 타입의 태그 모두에게 스타일을 적용하는 선택자를 태그 선택자(tag selector) 또는 타입 선택자(type selector)라 한다. 선택자는 특정 태그에 대해 태그 이름으로 스타일을 적용한다. HTML 문서 내에서 선택자로 지정된 태그 모두는 설정된 스타일의 적용을 받는다.

㉡ 전체 선택자(universal selector)

전체 선택자는 HTML 문서 내 모든 태그에 설정된 스타일을 적용한다. 전체 선택자는 선택자로 별표 '*'를 사용한다.

전체 선택자는 문서의 여백이나 마진(margin), 패딩(padding), 글꼴 크기 등과 같은 스타일을 문서의 기본 스타일로 설정할 때 사용하는 것으로 '문서 설정 초기화'의 의미가 있다.

㉢ 그룹 선택자

그룹 선택자는 여러 태그에 대해 공통으로 동일한 스타일을 적용한다. 그룹 선택자는 각 태그를 콤마 ','로 구분하여 사용한다.

그룹 선택자로 지정된 모든 태그는 HTML 문서 내의 해당 태그 모두는 동일한 설정된 스타일을 적용을 받는다.

㉣ 자식 선택자(child selector)

㉤ 후손 선택자(descendant selector)

㉥ 클래스 선택자(class selector)

㉦ 아이디 선택자(id selector)

③ CSS 적용 방법

㉠ 웹 문서에는 여러 종류의 스타일이 스타일시트로 묶여 문서에 포함되어 있다. 웹 브라우저는 웹 문서를 읽을 때 스타일시트의 정보를 파악하여 HTML 문서를 화면에 표시하게 된다.

㉡ 웹 문서에 스타일시트를 포함하는 방법은 크게 3가지가 있다. 가장 많이 사용되는 방법은 외부 스타일시트 방식이고 그 다음이 내부 스타일시트 방식, 인라인 스타일 순서이다.

- 외부 스타일시트(external style sheet)
- 내부 스타일시트(internal style sheet)
- 인라인 스타일(inline style)

3. 컴퓨터시스템

3-1 컴퓨터시스템의 구조

(1) 컴퓨터 정의

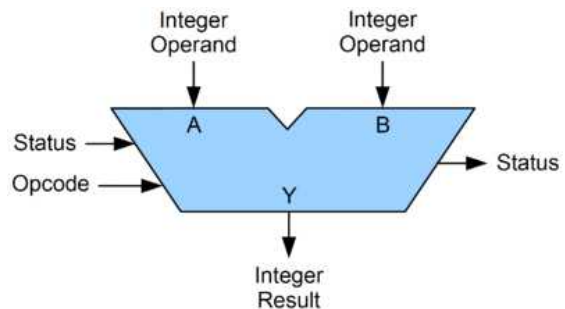
컴퓨터란 무엇인가. 이러한 물음에 때로는 당혹감을 느끼기도 한다. 인간의 활동과 조직의 관리에 있어서 필수 불가결한 존재로 위치하고 있는 컴퓨터를 새삼스럽게 정의한다는 것은, 많은 사람들의 머리에 반드시 숨어 있는 진리나 사랑을 말로 새로이 정의하기 곤란한 것과 유사하다. 우리는 컴퓨터가 무엇인가에 대해서 ‘표현하기는 어렵지만 분명하게 그 무엇을 느낄 수는 있는 것’, 바로 그러한 공감은 가지고 있다.



[그림 3-1] 컴퓨터의 정의

초기에는 컴퓨터를 EDPS(Electronic Data Processing System)라 하였으며, 전기 및 전자(electronic) 원리를 통해 여러 가지 자료(data)를 사용자가 원하는 대로 처리(processing)하여 주는 기계 또는 기술의 조직(system)으로 총칭하였다. 그러한 근본적인 정의에 대해서는 요즈음에 있어서도 크게 다르지 않다. 즉, 컴퓨터를 ‘원시(raw) 자

료(data)를 처리(processing)하여 새로운 정보(information)를 만드는 기계’ 라고 정의할 수 있다. 조금 기술적인 내용을 포함하여 컴퓨터를 표현하면, 필요한 프로그램(program)이나 자료(data)를 입력받아(input), 이를 기계 내부에 저장한 다음(stored-program), 입력된 프로그램을 해석, 제어하고(control), 입력된 자료를 수학적, 논리적인 연산으로 처리하여(ALU^{[14]) 새로운 정보를 생성해 내는(output) 전자식 기계를 의미한다고 할 수 있다. 간단히 컴퓨터를 정의하면 사전적 의미에서 보듯 ‘computer’ = ‘compute’ + ‘er’, 즉 “전자적으로 계산을 수행하는 장치” 라 할 수 있다.}



[그림 3-2] ALU 기호 표시

[14] ALU(Arithmetic Logic Unit) 산술논리 연산장치

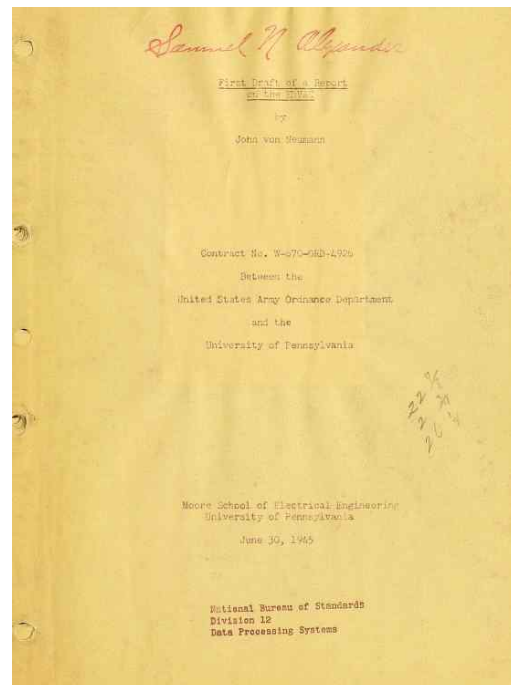
(2) 노이만 컴퓨터 구조(Von Neumann computer architecture)

오늘날 컴퓨터구조의 대부분은 노이만 구조이며 프로그램 내장방식(stored-program computer)의 특징을 가지고 있다. 프로그램 내장방식이라 함은 '수행하고자 하는 프로그램과 데이터는 처리속도가 빠른 전자식 기억장치에 저장된 후 처리'되는 것을 의미한다. 초기에는 프로그램의 저장이 데이터 저장과는 다른 기억장치를 사용하기도 하였으나 현재는 동일한 기억장치의 주소공간을 사용하고 있다.

(아래 2개의 그림을 입력)



[그림 3-3] 노이만(John von Neumann)



[그림 3-4] "First Draft of a Report on the EDVAC"

① 노이만 구조 컴퓨터의 구성 요소

노이만 구조에서 제시하는 전자식 컴퓨터는 CPU 장치(central processing unit)와 기억장치(memory unit), 입력장치(input device), 출력장치(output device), 그리고 이들 여러 장치들을 상호 연결하고 동작하도록 하는 제어장치(control unit) **등으로** 구성된다.

CPU 장치는 산술연산과 논리연산을 처리하는 산술논리장치(ALU; arithmetic logic unit)를 포함하고 있으며, 산술연산 및 논리연산 처리에 필요한 데이터의 임시저장 기능을 가지는 명령레지스터(instruction register) 및 프로그램 카운터(program counter) 등과 같은 레지스터(processor register), 그리고 CPU 장치 내부를 제어하기 위한 CPU 제어장치(CPU control unit) 등을 포함하고 있다.

② 프로그램 내장 컴퓨터 (stored-program computer)

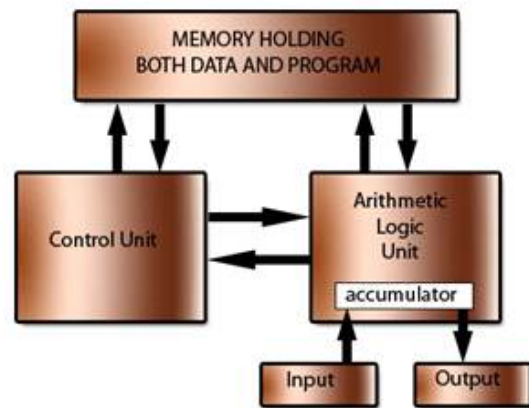
프로그램 내장 컴퓨터라 함은 컴퓨터가 명령어 **처리의** 효율성을 위해 수행하고자 하는 프

로그래(program)과 데이터(data)를 기억장치에 저장한 후 처리하는 것을 의미하는데 노이만 구조 컴퓨터와 하버드 구조 컴퓨터가 있다. 노이만 구조 컴퓨터는 동일한 기억장치에 이들을 저장하며 하버드 구조 컴퓨터(Harvard architecture computer)는 서로 다른 기억장치에 저장하는데, 일반적으로 프로그램 내장방식 컴퓨터라 함은 노이만 구조 컴퓨터를 의미한다.

③ 노이만 구조 컴퓨터의 동작 요약

앞에서 설명한 바와 같이 노이만 구조의 컴퓨터 동작의 핵심은 프로그램 내장방식이다. 프로그램 내장방식은 축적 프로그램 방식이라고도 하는데, 자료 처리에 필요한 모든 프로그램(program)과 자료(data)는 먼저 주기억장치에 반드시 저장한 다음 CPU가 처리하여야 한다는 것을 의미한다. 즉, CPU는 수행할 프로그램이나 자료를 입력장치로부터 직접 가져오지 않고, 접근 속도(access time)가 비교적 빠른 주기억장치로부터 가져와야 한다는 것이다.

CPU를 구성하고 있는 ALU, 레지스터, 제어회로는 수행할 명령어를 주기억장치로부터 읽어 들이고(fetch), 읽은 명령어를 해석한(decode) 다음 수행하며(execute), 수행 결과를 다시 주기억장치에 써넣는(output) 방식으로 동작하고 있는데, 이가 바로 노이만식 컴퓨터 구조의 핵심인 프로그램 내장 방식이다.



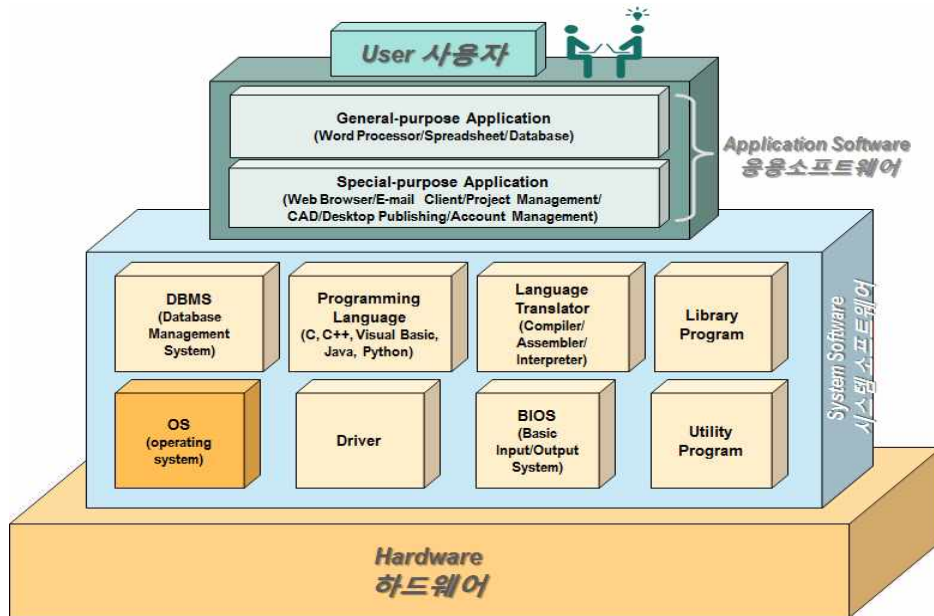
[그림 3-5] 노이만 구조와 프로그램 내장 개념

자연스럽게도 위의 프로그램 내장방식이 효과적으로 동작되기 위해서는, 주기억장치에의 접근이 빨라야만 하고, 이를 위해서는 주소(address)에 의한 기억장치 접근이 좋겠다는 제의도 함께 논의되었다. 그리고 이 방식은, 주기억장치로부터 명령어를 읽고 수행하면서, 명령어는 반드시 순차적으로 처리되어야 한다는 것을 지적하고 있다. 즉, 하나의 명령어가 CPU내에서 수행, 처리되고 있는 중에는 다른 명령어를 읽고, 해석하고, 수행하는 동작을 하지 말고, 반드시 미리 읽은 명령어의 수행, 처리가 완료된 다음 명령어를 처리하여야 한다는 것이다.

최근 들어 노이만 구조 컴퓨터는 컴퓨터 속도 향상을 위해 **제의한** 특성이 오히려 속도 향상에 걸림이 되고 있다는 지적이 있다. 프로그램이나 자료를 주기억장치에 반드시 저장하여야 한다는 노이만 **구조의** 특성으로 인해 CPU의 처리속도가 많이 향상되었음에도 **불구하고** 컴퓨터의 성능은 사용하는 주기억장치의 접근 속도에 크게 의존하는 결과를 초래하였고, 명령어의 순차적 **처리** 특성으로 말미암아 명령어 병렬 실행이 곤란하게 되었다. 따라서 우리는 이를 개선한 형태의 컴퓨터 등장 이미 시작되고 있다는 점에 유의하여야 할 것이다.

3-2 하드웨어와 소프트웨어

컴퓨터를 구조적으로 살펴보면 기본적으로 하드웨어(hardware)와 **소프트웨어(software)**로 구성되어 있음을 알 수 있다. 하드웨어는 기계 장치 그 자체를 말하며, 소프트웨어는 기계 장치를 움직이게 하는 프로그램이나 이에 따르는 기술을 말한다.



[그림 3-6] 소프트웨어 종류

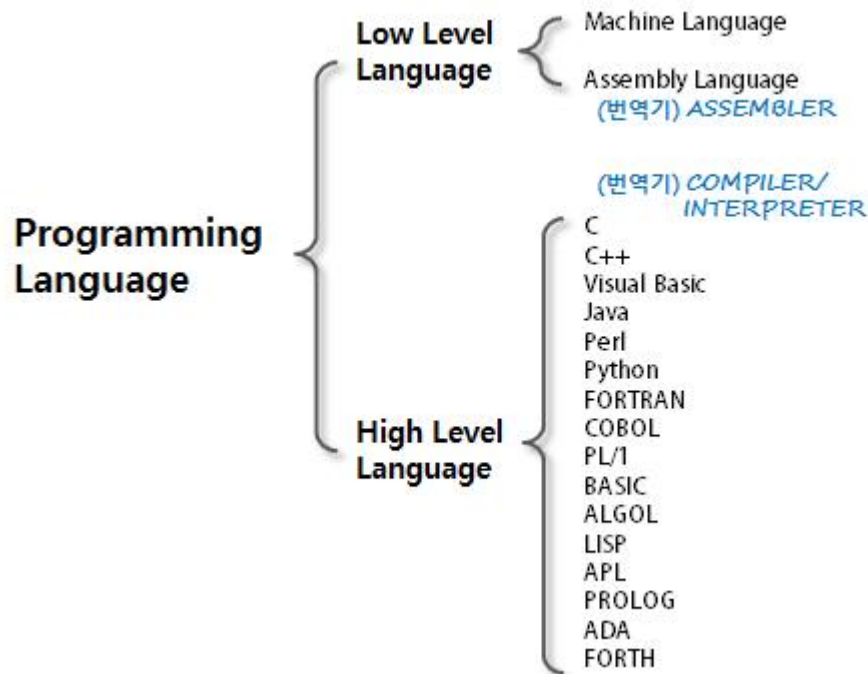
즉, 소프트웨어는 인간과 기계를 연결시켜 주는 하나의 장치로 생각할 수 있다. 쉽게 얘기하면, 하드웨어는 악기의 수리, 조립, 제작의 의미이고, 소프트웨어는 활용(programming) 분야로서 악기의 연주, 악보 제작 등에 비유할 수 있다. 예를 들면 모니터, 하드 디스크, 프린터 등은 하드웨어이고, 윈도우, UNIX, iOS, 안드로이드 등의 운영체제나 한글, 엑셀, 파워포인트, 크롬, 사파리, 게임 프로그램 등은 소프트웨어이다. 이러한 컴퓨터는 복잡하고 다양한 현대 사회에서 필수적인 기구로서, 주로 정보(information)를 표현하거나 처리하는데 사용되고 있다. 현재 컴퓨터는 전자 기술의 발전에 힘입어, 가격이 저렴해지고 크기가 소형화됨으로써 컴퓨터의 이용에 **대한** 경제성이 높아지고 있으며, 앞으로 보급 및 응용 분야가 계속 증가할 것이다.

(1) 소프트웨어(software)

소프트웨어는 컴퓨터 사용자가 하드웨어라는 자원(resource)을 효율적인 사용할 수 있도록 하는 도구이다. 사용자가 복잡한 하드웨어의 동작 원리를 이해하여야만 이를 사용할 수 있다면, 컴퓨터를 사용할 수 있는 사람이란 극소수에 지나지 않을 것이다.

소프트웨어에는, 하드웨어의 직접적인 제어와 총괄적인 지원을 주 기능으로 하는 시스템 소프트웨어(system software)가 있으며, 사용자와 직접적인 대화를 주 기능으로 하는 응용 소프트웨어(application software)가 있다. 아래 그림은 사용자와 하드웨어를 연결하는 **소프트웨어의** 구성을 나타내고 있다.

- ① 시스템 소프트웨어 (system software)
- ② OS(operating system ; 운영체제) : 컴퓨터 하드웨어를 사용자가 손쉽게 효과적으로 사용할 수 있도록 지원하는 거대한 프로그램의 집단
Windows, Mac OS X, Unix, Linux, iOS, Android
- ③ 드라이버(driver) : 주변장치가 가지는 여러 기능을 구현하도록 **기계어로** 구성된 소프트웨어로 운영체제는 드라이버를 경유하여 해당 하드웨어 장치의 동작을 수행
- ④ BIOS(Basic Input & Output System) : PC에 내장되어 부팅과 드라이버-주변장치 사이의 인터페이스 역할을 처리하는 일련의 소프트웨어 루틴(routine) 집합체
- ⑤ DBMS(database management system) : 데이터베이스에 데이터를 저장하고 호출하고 **데이터베이스의** 무결성을 관리하는 시스템으로 데이터 입력, 갱신, 검색, 보고 등의 기능을 제공



[그림 3-7] 프로그래밍 언어와 언어 번역기

- ⑥ 프로그래밍 언어(programming language) : 모든 시스템 소프트웨어나 응용 소프트웨어는 프로그래밍 언어를 이용하여 개발되며 수행을 위해 기계어로 변환
C, C++, Visual Basic, Java, Perl, Python, PASCAL, FORTRAN(FORMula TRANslator), COBOL(COMmon Business Oriented Language), PL/1(Program Language one), BASIC(Beginner's All-purpose Symbolic Instruction Code), ALGOL(ALGOrithmic Language), LISP(LISt Processor), APL(A Program Language), PROLOG(PROgramming in LOGic)
- ⑦ 언어 번역기 (language translator) : 개발된 소스 프로그램은 컴파일러(compiler), 어셈블러(assembler), 인터프리터(interpreter) 등의 언어 번역기를 이용하여 기계어로 변환한 후

CPU에서 수행

- ⑧ 유틸리티 프로그램(utility program) : 컴퓨터와 주변장치, 컴퓨터와 사용자간의 자료 교환 등을 용이하게 하는 프로그램
루틴 프로그램(routine program) : sort, list, print, cat
데이터 관리용 : create files, merge files
- ⑨ 라이브러리 프로그램(library program) : 응용 프로그램을 위해 미리 작성해 놓은 표준화된 프로그램
- ⑩ 자기 진단 (self-diagnostic) : 컴퓨터 고장을 진단, 유지 보수하는 목적의 프로그램

(2) 응용 소프트웨어 (application software)

- ① 웹브라우저(Web Browser)
- ② 전자우편(e-Mail) 클라이언트
- ③ 워드프로세서(Word Processor)
- ④ 데이터베이스(Database)
- ⑤ 스프레드시트(Spreadsheet)
- ⑥ 프레젠테이션 그래픽(Presentation Graphics)

(3) 하드웨어(hardware)

노이만 컴퓨터의 기본 구성은 입력장치, 출력장치, 기억장치와 중앙처리장치, 제어장치 등으로 이루어져 있는데 각 구성 요소의 종류를 요약하면 다음과 같다.

- ① 입력장치(input device) : keyboard(KBD), mouse, digitizer, card reader, joy stick
 - ㉠ 입력 매체 : OMR^[15], OCR^[16], MICR^[17], punch card, paper tape
- ② 기억장치 (memory unit)
 - 주기억장치 (main memory
 - 보조기억장치 (auxiliary memory unit) :
- ③ 중앙처리장치 (CPU ; central processing unit)
 - ㉠ ALU + control unit + register^[18]
- ④ 출력장치 (output device) : printer(hard copy), display terminal (soft copy)

[15] OMR : Optical Mark Reader/Recognition

[16] OCR : Optical Character Reader/Recognition

[17] MICR : Magnetic Ink Character Reader/Recognition

[18] register : 임시 기억 장치