

프로세서와 소프트웨어

全吉男*, 林榮煥**

韓國電子技術研究所 시스템部 部長*, 選任研究院**

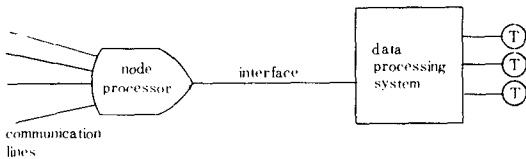
I. 서론

컴퓨터 네트워크란 여러 가지의 컴퓨터와 communication link 및 사용자를 위하여 연결된 terminal의 연결을 말하는 것으로 최근 두드러지게 발전하고 있다. 이러한 상황에 여기서는 node processor와 data processing system 그리고 그것의 연결에 관해서 설명하고자 한다.

우선, 세계적인 표준화 방향과 그에 대한 예를 살펴보고 node processor에 대한 고찰을 했다. 여러 가지 면에서 그것의 기능과 구조 개발 방법을 현재 나와 있는 제품을 중심으로 소개했다.

그리고 data processing system 및 node processor와의 연결 방법에 대해서 논의하고 network system의 응용 분야를 살펴 보았다.

II. 표준화(Standardization)



네트워크 아키텍처란 어느 한 지역의 사용자가 다른 지역의 사용자와 교신하기 위하여 필요한 기능을 연결하는 방법을 정의한 것이다.

현재 전체적으로 표준화 된 것은 없지만 다음과 같은 기관에서 표준화 작업이 진행되고 있다.

Node processor는 data processing system의 front-end-processor의 역할을 하는 것으로 communication 처리를 위한 기능을 수행한다.

실제 이것은 현존하는 network system에 여러 가지 방법으로 개발되어 있으나 전형적으로 layer 1에서 layer 4 까지 제공하고 있다.

Data processing system은 미니컴퓨터나 work-frame computer와 같은 system으로 사용자에게 네트워크를 이용할 수 있는 기능을 제공한다. 즉 higher level의 protocol을 제공함으로써 사용자의 요구에 맞도록 개발될 수 있다.

그러나 통신망을 이용하여 사용자의 요구를 만족시켜 주기 위해서는 여러 가지 해결해야 할 문제점이 많다.

1. ISO

International standards organization은 open systems interconnection의 reference model이라는 seven-level architecture를 제안했다. 그림 1에서 보는 바와 같이 layered architecture로 되어 있는 것으로 자세한 설명은 참고문헌 [1]과 [4]에 나타나 있다.

2. IEEE

IEEE의 computer standard committee에서 local computer network의 표준화 작업의 일환으로 ISO-OSI reference model의 layer 1 과 layer 2를 표준

화 한 IEEE 802를 제안하였다. 이것은 local 네트워크인 Ethernet의 영향을 받은 것으로 참고문헌[16]에 표준 사양이 설명되어 있다.

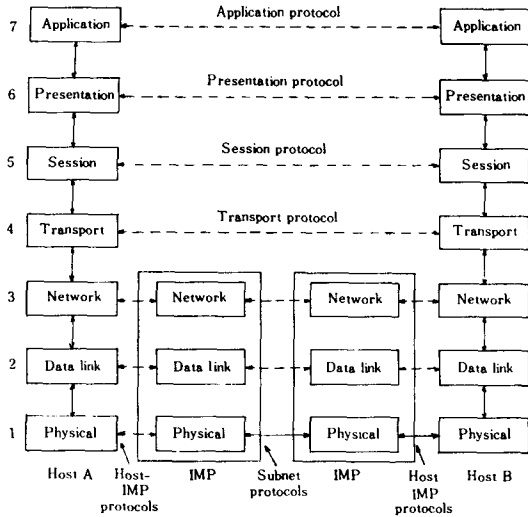


Fig. 1. ISO reference model

3. DOD

DOD의 DARPA (defense advanced research project agency)에서 high level protocol을 표준화하기 위한 작업으로 DoD standard인 TCP (transmission control protocol)과 IP (internet protocol)을 제시했다.

참고문헌 [7]에 설명된 바와 같이 TCP는 packet-switched computer communication network에 있어서 host간의 신뢰성이 높은 protocol을 제공하기 위한 것이고 IP는 네트워크를 서로 연결하기 위하여 개발된 것으로 참고문헌 [6]에 잘 설명되어 있다.

4. CCITT

CCITT (international consultative committee for telegraphy and telephony)의 주요 업무는 전화나 전보 및 데이터 통신 분야의 연결에 대한 기술적인 자문을 하는 기관으로 asynchronous terminal을 연결하는데 사용되는 connector의 핀 배열 및 의미를 규

정지는 V.24나 computer와 computer network을 연결하기 위한 X.25 규정등이 표준화 되어 사용되고 있다. X.75 model은 virtual circuit model에 근거를 둔 internetwork protocol을 규정하는 것이다. 이외의 X. series에 관한 규정은 참고문헌 [1]과 [4]에 잘 나타나 있다.

III. Node Processor

Node processor는 communication system에 있어서 communication line과 data processing machine이나 컴퓨터 네트워크를 연결하기 위한 것이다. 이 장에서 node processor의 기능 및 구조, 그에 필요한 software를 현존하는 node processor 제품 중심으로 살펴 보았다.

1. 기능

Node processor는 기본적으로 OSI reference model의 layer 1과 layer 2의 기능을 제공하고 그 이상으로 대개 layer 3나 layer 4의 기능을 제공하며, 드물게 layer 5나 6의 electronic mail기능을 제공한다.

Physical layer로서 analog signal을 이용한 RS-232-C와 RS-449, digital interface를 위한 X.21이 있다. 이것에 대한 상세한 설명은 참고문헌 [1], [2], [3], [4]를 참조하면 된다.

Data link layer의 예로서 HDLC (high-level data link control)와 SDLC (synchronous data link control), ADCCP (advanced data communication control procedure), LAP (link access procedure), LAPB (link access procedure B) 등과 같은 것이 있다.

HDLC는 physical layer에서 제공하지 않는 transmission error 처리와 transmission speed의 차이를 해결해주는 기능을 가지고 있다.

Network layer에는 CCITT에서 제정한 X.25 protocol을 주로 제공하고 있다. X.25 protocol은 physical layer가 X.21이고 data link layer는 HDLC의 변형인 LAP와 LAPB로 구성되어 있으며 network layer는 virtual circuit용으로 되어 있다. X.25는 미니컴퓨터나 마이크로컴퓨터 또는 chip등 여러 가지 방법으로 implement되어 널리 사용되고 있다.

Transport layer의 주요 기능은 사용자간의 연결을 설정하거나 떼어 놓으며, flow control이나 buffering 그리고 multiplexing 등을 처리하는 것으로 host com-

puter의 프로그램으로도 implement 될 수 있다.

Node processor가 제공할 수 있는 application으로 node processor에 연결된 터미날에 보내는 electronic mail기능이 있을 수 있다.

2. 아키텍처

Node processor의 구조상 특징은 fault-tolerant architecture로 신뢰도가 매우 높은 것이다. 즉 어느 한 부분의 고장에도 불구하고 주어진 기능을 수행하기 위하여 다른 부분이 대신 할 수 있도록 여분을 두는 방식을 택하고 있다. 일반적으로 node processor는 CPU와 memory, I/O interface unit으로 구성되어 여러 가지 communication protocol을 제공하는 일종의 computer이다. 그 예로서 microprocessor를 이용한 telenet과 SESA, minicomputer를 이용한 BBN 및 tym-net 등이 있다.^{15), 16), 17)}

또한 네트워크의 이용이 증가함에 따른 사용자의 요구를 충족시켜 주기 위하여 확장성이 높게 설계되어 있다. 즉 BBN CPU는 사용자가 마이크로프로그램을 할 수 있기 때문에 확장하기가 용이하다.

이 프로세서는 X.25나 Async, Bisync 그리고 HDLC/SDLC 등과 같은 여러 가지 communication protocol을 제공하고 여러 가지 터미날을 연결 할 수 있는 concentrator 역할도 담당한다.

3. Classification

Node processor는 여러 가지 방법으로 개발 될 수 있는 것으로 크게 minicomputer-based processor와 microcomputer-based processor로 구분 할 수 있다.

1) Minicomputer-based Processors

DoD의 ARPA(advanced research projects agency)에 의해서 개발된 Arpanet는 IMP(interface message processor)라는 processor를 사용하고 있다.

원래 Arpanet의 IMP는 Honeywell DDP-516 미니컴퓨터를 이용하여 만들어졌으나 나중에 16K memory 용량을 가진 Honeywell DDP-316을 이용하였고 최근에는 lockheed SUE 미니컴퓨터를 사용하여 여 Pluribus라는 멀티프로세서 IMP를 이용하고 있다.^{18), 19), 20)}

BBN 컴퓨터에서 제공하는 C/30 packet switch processor는 communication environment에 적당한 instruction set를 수행할 수 있고 마이크로프로그램이

가능한 20-bit 컴퓨터 이다.¹⁶⁾

컴퓨터 society를 중심으로 개발하려는 CSNet의 host는 X.25 protocol을 이용하여 public network에 연결하려고 한다. X.25는 PDP11 version 7 UNIX나 VAX UNIX를 이용하여 node processor에서 개발하려고 한다.

TYMNET Inc.는 mini-engine communication processor를 제공한다. 이것은 X.25 뿐만 아니라 여러 가지 interface를 할 수 있는 기능을 제공하는 것으로 상세한 설명은 참고문헌 [19]에 나타나 있다.

2) Microcomputer-based Processors

GTE Telenet의 host/terminal interface processor인 TP 4000은 X.25 protocol을 제공하는 것으로 240개의 communication line을 가지고 있다. 이 processor는 2개의 CPU와 30개의 line processing unit (LPU)로 구성되어 있다. 각 LPU는 각기 microprocessor와 memory, line interface가 있어 communication line을 이용 할 수 있을 뿐만 아니라 X.25 link access procedure나 PAD (packet assembly/disassembly) 소프트웨어를 수행한다.¹⁷⁾

앞으로 마이크로프로세서를 이용하여 node processor를 implement하는 경향이 활발하게 진행 될 것이다. 그 예로서 associated computer consultants는 Z 80 processor를 이용하여 X.25를 implement 하였다.

4. 소프트웨어

Node processor에 필요한 software는 종류에 따라 다양하지만 크게 system software와 application software로 구분 할 수 있다.

1) System Software

Node processor는 주로 network layer나 transport layer의 기능을 software로 implement 한다.

Network layer에서 implement하여야 할 기능은 다양하다.

소프트웨어 디자인을 간단히 하기 위해서 네트워크는 체계적으로 구성되어야 한다.

이 layer에서 transport layer에 제공하는 model로는 datagram model 및 virtual circuit model이 있다.

Virtual circuit model은 transport layer에 완전한 통로를 설정하므로 모든 packet이 보낸 순서대로 전달되지만 데이터그램 모델에서는 순서에 관계없이 전달된다. 이 두 가지 모델의 장단점 및 비교 설명은 참고

문헌 [1], [3] 및 [4]에 잘 나타나 있다.

또 다른 중요한 것은 routing 기능으로 들어 오는 패킷을 어느 outputline을 통하여 보낼 것인가 하는 것을 결정하는 기능이다. 이 routing algorithm에 관한 연구가 활발히 진행되어 참고문헌 [1], [4]에서 보는 바와 같이 여러 가지 알고리즘이 개발되었지만 실제로 응용되는 것은 기본적인 것 뿐이다.

그외 중요한 기능으로는 congestion 및 flow control 하는 방법이다.

그러므로 대개의 노드 프로세서는 X.25 protocol을 implement 하고 또한 다른 기능을 추가 시켰다.

2) 어플리케이션

일반적으로 어플리케이션 소프트웨어는 사용자가 원하는 대로 개발 할 수 있다. 그리고 특별한 경우로 노드 프로세서가 electronic mail이나 file transfer protocol, virtual terminal protocol 등을 제공 할 수 도 있다.

노드 프로세서가 더욱 powerful 해짐에 따라 host computer의 overhead를 줄이거나 maintenance를 위하여 커뮤니케이션에 관한 host computer의 역할을 대신 하려고 하고 있다.

즉 여러 가지 커뮤니케이션 프로토콜이나 어플리케이션 프로그램을 노드 프로세서에서 수행할 수 있도록 개발하고 있다.

5. Node Processor Companies

그 동안 살펴 본 노드 프로세서의 중요한 회사를 살펴보면 다음과 같다.

1) BBN Computer Corp.

Bolt Beranek and Newman (BBN)은 Arpanet을 개발하여 유지해 오고 있고 Telenet을 시작했다.

그 경험을 토대로 이 회사는 네트워크에 사용되는 C/30와 같은 medium-speed communications processor나 pluribus와 같은 high-throughput communications multiprocessor 뿐만 아니라 C/50와 같은 concentrator를 제공하고 있다.

2) GTE Telenet Communications Corp.

이 회사는 Arpanet의 설계자에 의하여 설립된 후 그 기술을 도입하여 텔레컴퓨터 통신 프로세서인 TP를 개발하였다. 이 프로세서는 public이나 private 네트워크에 여러 개의 마이크로프로세서 네트워킹 능력을 제공할 수 있다.

3) Tymnet

패킷 네트워크 분야에 설립된 Tymnet Inc는 Tymnet mini-engine이라는 communication processor를 제공하고 있다. Mini-engine 1과 mini-engine 2는 asynchronous terminal interface를 위한 것이고 mini-engine 3은 Tymnet ISIS 네트워킹 소프트웨어를 수행시킬 수 있는 것이다.

IV. 데이터 프로세싱 시스템

데이터 프로세싱 시스템은 노드 프로세서에 연결된 host 컴퓨터로서 고속도로 처리 할 수 있는 능력을 이용하여 사용자의 요구에 맞는 여러 가지 기능을 제공하는 한편 여러 대의 터미날을 연결할 수 있다.

1. 역할

여기서 데이터 프로세싱 시스템이란 일반적인 컴퓨터 시스템을 지칭하는 것으로 데이터 프로세싱이나 데이터 베이스 처리, 복잡한 연산 등에 이용된다. 또한 사용자에게 많은 터미날을 연결할 수 있는 concentrator 역할도 한다.

데이터 프로세싱 시스템이 없는 경우 concentrator가 노드 프로세서와 여러 대의 터미날을 연결시켜 줄 수 있다. 그리고 컴퓨터와 컴퓨터의 커뮤니케이션을 담당할 뿐만 아니라 컴퓨터와 터미날의 통신까지 담당한다.

2. 네트워크 콘트롤 센터

네트워크 시스템에서 데이터 프로세싱 시스템이 담당하는 주요한 기능이 이 기능이다.

네트워크 콘트롤 센터에서 담당하는 기능은

- System initialization
- Status report 및 이상 경고
- Network control
- Accounting
- 통제처리 등 여러 가지가 있다.

V. 인터페이스와 인터넷워크

이 장에서는 노드 프로세서와 데이터 프로세싱 시스템 간의 인터페이스 뿐만 아니라 local area network과의 연결에 대해서 설명하고자 한다.

1. 노드 프로세서 데이터 프로세싱 시스템과의 연결

여기에 필요한 여러 가지 방법으로는 다음과 같다.

1) X. 25

X. 25는 CCITT에 의해서 정의된 것으로 DTE (data terminal equipment)라는 host와 DCE(data circuit-terminating equipment)라는 프로세서 사이의 연결 방법을 정한 것이다.

이것에 관한 상세한 설명은 참고문헌[1]과 [4]에 나타나 있다.

2) SDLC

IBM이 현재 SNA(system network architecture)에 사용되고 있는 SDLC(synchronous data link control)을 개발하여 ANSI와 ISO에 국제적인 standard로 하기 위하여 제출한 바 ANSI는 그것을 수정하여 ADDCP(advanced data communication control procedure)가 되었고 ISO는 수정하여 HDLC(high-level data link control)을 개발하였다.

그 이후 CCITT가 HDLC를 수정하여 LAP(link-access procedure) 및 LAPB를 제정했다. 이 모든 protocol은 bit oriented 라는 점이 공통적이다.

2. 인터넷네트워크

현존하는 컴퓨터 네트워크이나 앞으로 개발될 네트워크를 서로 연결시킴으로 더 많은 효과를 노릴 수 있지만 서로 다른 특성을 지니고 있기 때문에 여러 가지 문제가 발생한다. 그러므로 인터넷네트워크의 경향은 네트워크 아키텍처를 표준화하는 방향으로 전개되고 있다.

그 예로서 CCITT의 X. 75 모델이 있다.

1) Local Area Network(LAN)

한 건물내에 간단하면서도 편리하게 사용할 수 있는 네트워크 시스템이 지난 세기에 개발되어 PABX(private automatic branch exchange)의 발전과 더불어 급속도로 발전하고 있다. LAN에 관한 소개는 참고문헌[18]과[4]에 잘 나타나 있다.

컴퓨터 분야의 resource sharing과 multiprocessing에 대한 추구의 일환으로 LCN이 급속도로 발전되어 왔다. Distributed system이나 로칼 컴퓨터 네트워크, 그리고 long-haul network의 차이점은 서로가 완전히 다른 시스템이라기 보다 단지 정도의 차이에 불과하다고 볼 수 있다. 즉 프로토콜이나 알고리즘 등은 로칼 네트워크이나 long-haul network에

다같이 존재하나 설계에 있어서 로칼 네트워크이 훨씬 간단하다고 볼 수 있다.

그러므로 로칼 컴퓨터 네트워크는 값싼 transmission medium과 modem, high data rate, 그리고 커뮤니케이션 프로토콜과 high-level protocol에 의한 각 node와 device 간의 높은 기능적인 연관성을 특징으로 한다. 이 분야에 대한 자세한 설명은 참고문헌[1], [2], [3], [4], [16], [18] 등을 참조하면 된다.

2) LAN과 Long-haul Network

Long-haul network은 processing time을 희생시키면서 communication channel의 bandwidth를 효율적으로 사용하기 위하여 복잡한 프로토콜을 채택하고 있는 반면 로칼 네트워크는 프로세싱 타임을 줄이기 위해서 간단한 프로토콜을 사용한다. 그러므로 로칼 네트워크를 long-haul network에 연결하기 위한 gateway를 사용하여 서로 다른 특성을 만족시켜 주며 gateway에서는 X. 25와 같은 국제적인 표준 프로토콜을 사용하여 서로의 프로토콜을 번역해 준다.

여기에 관한 자세한 설명은 참고문헌[12], [14], 에 나타나 있다.

3) Long-haul Network과 Long-haul Network

이 분야의 좋은 예로서 미국에서 계획중인 CSNet project를 들 수 있다. 이 project는 Arpanet와 Telenet, Phonnet을 연결하는데 PDP11이나 VAX11을 gateway로 사용하여 연결하고 있다. 이 분야에 관한 설명은 참고문헌[9]에 있다.

VI. 어플리케이션

1. Electronic Mail

이 기능은 엘렉트로닉스 메시지를 전달하는 기능으로 지역적으로 떨어져 있는 사용자간의 유대를 긴밀히 할 수 있다.

2. File Transfer

메모나 프로그램을 교환함으로써 정보 전달을 용이하게 함과 동시에 소프트웨어를 개발하는데 서로의 tool이나 소프트웨어 모듈을 공유할 수 있다.

3. 데이터 베이스 액세스

서로의 데이터 베이스를 관리하면서 다른 여러 곳에서도 액세스하며 사용할 수 있는 기능이다.

4. Virtual Terminal

다양한 터미널 간의 incompatibility를 해결하기 위하여 서로 다른 종류의 터미널 사이의 차이를 없애 줄 수 있는 프로토콜이 개발되었다.^[1]

5. 기 타

1) Distributed Data Base Systems

데이터 베이스를 지리적으로 분산시켜 처리함으로써 문제를 해결해 갈 수 있다.

2) Distributed Computation

컴퓨테이션 파워를 분산시켜서 지역적인 특성을 최대한 활용한다.

Ⅶ. 결 론

이제까지 노드 프로세서와 메이타 프로세싱 시스템 및 그것의 연결과 응용분야를 살펴 보았다.

네트워크 시스템의 여러 가지 경향중에도 표준화하려는 노력은 더욱 활발히 진행 될 것이다. 그리고 노드 프로세서도 점점 여러 개의 마이크로프로세서를 이용하여 좀 더 수행 능력이 좋은 것을 만들게 될 것이다.

그리고 노드 프로세서에 필요한 시스템 소프트웨어도 high level language를 이용하게 되므로 processing power도 높아질 것이다.

또 다른 분야의 관심은 기존 네트워크 시스템을 서로 연결하는 인터네트워킹 분야에 더 한층 노력을 집중할 것이다.

이와 같은 상황에 복잡한 사회의 요구를 충족하기 위한 하나의 방법으로 이 분야에 대한 연구가 한층 더 진행되어야 한다.

參 考 文 獻

[1] D. W. Davies, D. L. A. Barber, W. L. Price and C. M. Solomoniders, *Computer Networks and Their Protocols*. John Wiley & Sons, 1979.
 [2] Leonard Kleinrock, *Queueing Systems, Vol. II: Computer Applications*. John Wiley & Sons 1976.
 [3] J. Martin, *Computer Networks and Distributed Processing: Software, Techniques, and Architecture*, Prentice-Hall, 1981.

[4] Andrew Tanenbaum, *Computer Networks*. Prentice-Hall, 1981.
 [5] BBN, Building block network components from the leader in packet switching: BBN Computer, *BBN Computer Corp.* September 1981.
 [6] DARPA, DoD Standard, *Internet Protocol*. January 1980.
 [7] DARPA, DoD Standard, *Transmission Control Protocol*. January 1980.
 [8] SYTEK, Inc., *Capabilities and Experience*. November 1979.
 [9] National Science Foundation, *CSNet-The Computer Science Research Network*. June 1981.
 [10] 3 Com Corp., *UNET Networking Software*. November 1980.
 [11] David C. Walden and Alexander A. Mckensize, "The evolution of host-to-host protocol technology," *IEEE Computer*, September 1979.
 [12] Vinton G. Cerf and Perter T. Kirstein, "Issues in Packet-Network Interconnection," *Proceedings of the IEEE*, vol. 66, No. 11, November 1978.
 [13] Masatoshi Iwayama and Atsumu Fujiware "Computer communication in NTT remote computing services," *National Computer Conference*, 1980.
 [14] Jonathan B. Postel, "Internetwork protocol approaches," *IEEE Transction on Communitons*, vol. COM-28, no. 4, April 1980.
 [15] David R. Boggs, John F. Shoch, Edward A. Taft, and Robert M. Metcalfe, "Pup: An inter-network architecture," *IEEE Transctions on Communitons*, vol. COM-28, no. 4, April 1980.
 [16] Computer Standard Committee, IEEE 802 Local network standard (Draft 1), *IEEE Computer Society*, July 4, 1981.
 [17] GTE Telenet, *TP4000 Host/Terminal Interface Processor and Packet Switch*. August 1981.
 [18] David D. Clark, Kenneth T. Torgan, and David P. Reed, "An Introduction to Local Area Network," *Proceedings of the IEEE*, vol. 66, no. 11, November 1978.
 [19] TYMNET, *The Intelligent Network People*. TYMNET Inc., August 1981.