

□ 특별기고 □

NVC(Networked Virtual Computer)의 개념과 구조

양 통*, 박영택*, 임영환*, 정일영*, 이경호**, 박치항*

1. 서 론

앞으로 컴퓨터 관련 패러다임 변화가 있을 것으로 예상된다. 공중 통신망의 속도가 증가하며 그에 연결된 컴퓨터가 거대한 하드웨어 계층으로 간주되는 컴퓨팅 환경의 광역화가 이루어질 것이다. 또 인간 중심의 컴퓨팅 환경의 중요성에 따라 착용이 가능한 단말기가 등장하리라 예견된다. 그리고 컴퓨터 개념의 구조적 변화가 예상된다.

현재 가상 컴퓨팅 기술 연구는 개념 정립 단계로 확고한 기술은 없다. 따라서 장기적인 진화 방향에 대한 일치된 개념 모델 및 통신과 컴퓨터와의 융합 방식의 정립이 필요하며, 앞으로 도래할 초고속통신망과 통합된 통신망의 초고속성을 제대로 활용할 수 있는 여러 분야의 연구가 필요하다.

본 연구에서는 이와 같은 환경 변화에 적응하는 네트워크 기반 가상컴퓨터(Networked Virtual Computer)란 개념을 제시하고 사용자 및 기능, 동작환경에서의 요구 사항을 분석하여 NVC의 구체적인 구조의 정립하고, 구조의 타당성을 보이기 위하여 계층별 구조 및 동작 모델을 제시하였다.

2. NVC의 개념

2.1 NVC의 정의

NVC란 앞으로 통신망의 속도가 계속 빠르

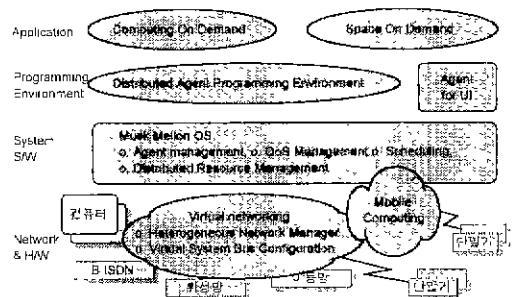


그림 1 NVC의 개념도

게 되어 시스템 버스로 사용될 수 있을 만큼 충분히 빠르게 되었을 때 통신망을 시스템 버스로 하는 컴퓨터 구조로 요구 및 서비스에 필요한 만큼의 컴퓨팅 파워와 메모리 스페이스를 확보 제공하는 것을 기본 개념으로 하며 본 연구에서는 정의는 다음과 같다.

NVC란 네트워크를 기반으로 세계 어디서나 어떤 컴퓨터와도 연결되며 사용자가 필요로 하는 컴퓨팅 파워 및 메모리 공간과 응용 서비스를 제공해 주는 컴퓨터이다. 사용자는 간단한 단말 장치를 이용하여, NVC의 내부 구조를 의식할 필요없이 하나의 컴퓨터처럼 사용한다. NVC는 초고속 통신망을 시스템 버스로 하여 네트워크 상의 컴퓨터 자원을 필요에 따라 재구성할 수 있다. 이를 위해 이동성 에이전트를 이용하여 구성 및 운영을 관리하는 NVC O&M (Operation & Management) manager가 존재한다.

2.2 NVC 개념

NVC는 개념적으로 네트워크 및 하드웨어, 시스템 소프트웨어, 프로그래밍 환경, 응용 소

* 중신회원
** 정 회원

소프트웨어의 4단계의 계층으로 구성되며, 각각은 다양한 네트워크를 융통성있게 재구성하여 사용자를 지원해 주며, 사용자 스케줄링 기술, 분산 자원 관리 기술 등을 지원하며, 분산 환경에서 효과적인 사용자의 프로그래밍을 도와 주며, 필요한 컴퓨팅 파워와 메모리 공간을 원하는 시간에 활용할 수 있게 지원한다.

2.3 NVC의 요구 사항

요구 사항은 NVC의 구조 정립을 위해 사용자 및 기능, 동작 환경의 관점에서 요구 사항을 정리하였다.

2.3.1 사용자 관점의 요구 사항

탁상형 또는 휴대형 단말기를 이용하여 소형부터 슈퍼컴퓨터 이상의 성능을 가진 컴퓨터 시스템을 구성하며, 실시간으로 한개의 컴퓨터 처럼 이용하여 기존 응용 프로그램의 수행 및 프로그래밍을 할 수 있는 환경이 제공되어야 한다. 정보는 단말기의 기능에 맞추어진 인터페이스를 통하여 멀티미디어 정보의 입출력 가능해야 한다. 여러개의 컴퓨터가 통신망에 연결되어 제공된다고 하더라도 사용자는 내부 모습을 자세히 알 필요없이 사용자는 원하는 서비스를 언제, 어디서나 사용할 수 있어야 한다.

2.3.2 기능적인 요구 사항

네트워크 상에 연결된 컴퓨터 시스템을 재구성하여 1개의 Von Neumann 구조를 가진 컴퓨터로 구성할 수 있을 뿐만 아니라, 여러 개의 프로세서 구조를 가진 병렬 컴퓨터도 구성할 수 있는 기능이 있고, 동적 재구성 기능이 있으며 Seamless Operation을 제공해야 한다. 또한 초고속 통신망을 시스템 버스로 사용할 수 있는 기능과 프로그램을 개발할 수 있는 기능, 음성이나 gesture 등으로 사용할 수 있는 멀티 모달 인터페이스 기능, 보안 기능이 있어야 한다.

2.3.3 동작 환경의 요구 사항

정지 상태 및 이동 상태에 NVC를 사용할 수 있어야 하며, 이종 시스템도 포함되어 사용

될 수 있어야 한다. 또 구성된 NVC를 다수의 사용자가 동시에 사용할 수 있을 뿐만 아니라 사용자들간에 공유 작업 환경이 제공되어야 하고, NVC에 참여하고 있는 자원을 균형있게 사용할 수 있는 기능이 제공되어야 한다.

3. NVC 구성 모델

3.1 NVC 시스템의 구성

NVC 시스템은 컴퓨팅 파워와 메모리 등의 환경이 사용자의 요구에 의하여 형성된 사용자 사용 공간을 말하며, 그림 2와 같이 하나의 컴퓨터 시스템만으로 구성되거나, 동종 또는 이종의 컴퓨터들로 구성될 수 있고, 각각의 컴퓨터로부터 NVC에 제공된 processor, memory, storage, io device 등의 자원들을 취하여 시스템을 형성할 수도 있다.

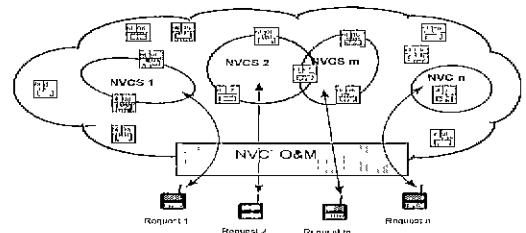


그림 2 NVC 시스템의 구성 형태

3.2 NVC O&M Manager

NVC O&M Manager의 기능은 자원의 등록, 관리, 요구 NVC 시스템 구성 및 load balancing 조절, NVC 자원 사용료 부과 및 자원 제공자 대가 지불 등의 일을 담당한다.

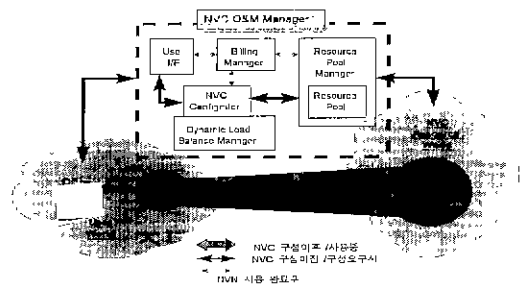


그림 3 NVC O&M Manager 동작 모델

3.3 NVC 동작 모델

NVC 동작 모델은 4단계로 구분된다.

1) NVC 시스템 구성 이전 단계

이 단계는 NVC resource world를 구성하여 O&M Manager가 NVC 시스템 자원 등록 및 자원 관리, 사용자 요구 대기 단계이다. O&M Manager는 자원 등록과 사용자의 요구를 대기한다.

2) 사용자요구에 의한 NVC 시스템 구성단계

O&M Manager Configurator는 사용자 요구에 따라 resource pool에 있는 자원을 활용하여 NVC 시스템을 구성하고 UI agent를 사용자의 단말기에 보냄으로 NVC 시스템을 사용하도록 연결한다. 이후 O&M Manager는 다른 사용자의 요구가 있는지 대기하는 상태로 된다.

3) NVC 시스템 사용 단계

사용자는 단말기에 있는 UI agent를 통하여 자기에게 할당된 NVC 시스템을 단독 컴퓨터인 것처럼 프로그램의 개발 또는 응용 소프트웨어를 수행하는데 사용한다. 이런 중에도 O&M Manager는 사용자의 업무량에 따라 NVC 시스템 구성에 사용된 resource들간의 load balancing을 담당한다.

4) NVC 시스템 사용 완료 단계

이 단계에서는 O&M Manager는 NVC 시스템 사용자에게 청구할 사용료와 자원의 제공자에게 지불할 대여료를 계산하여 정기적으로 발급하는 billing 및 accounting을 담당한다.

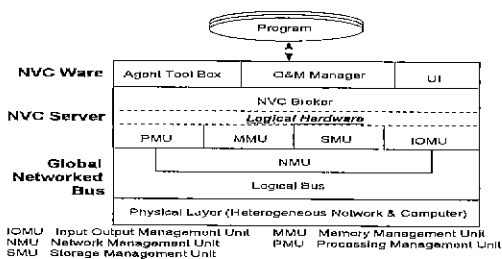


그림 4 NVC 시스템

4. NVC 시스템 구조

하위에는 heterogeneous network에 해당하

는 물리 계층이 있고 그 위에 NVC 시스템을 구성하는 각 요소들이 있다. 또한, 이러한 NVC 구조를 활용하여 생성된 application이나 tool 들은 NVCware로 정의하였다. 시스템 구성에 필요한 요소들은 resource pool로부터 가져와 하드웨어를 구성한 후에, 각 구성 요소들을 내부적으로 연결하고, 각 구성 요소들 사이를 bus 구조로 연결한다. NVC broker는 NMU로 연결된 여러 개의 logical hardware resource 를 한개의 시스템으로 묶어 주는 일을 담당하며 또한 사용자 프로그램이나 상위 계층의 요구시 적당한 자원들로 일을 분배하는 역할을 담당한다.

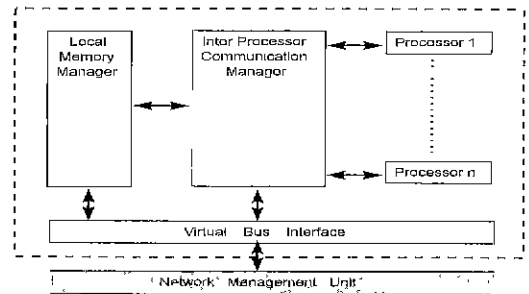


그림 5 PUM 동작모델

4.1 PMU(Processor Management Unit)

PMU의 주요 기능은 사용자가 요구한 CPU power를 제공하기 위하여 resource pool에 있는 하나의 적당한 CPU resource를 선택하거나 여러 개의 CPU를 모아 병렬 processor의 구성 기능을 담당한다.

4.2 MMU(Memory Management Unit)

NVC 시스템 내에서는 NVC Processor를 제

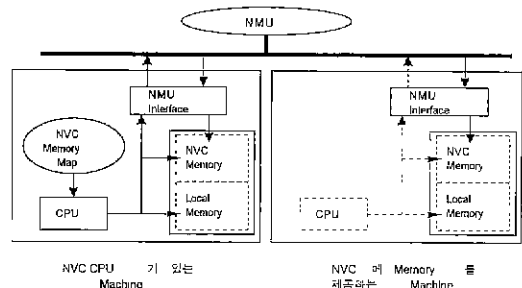


그림 6 MMU 동작 모델

공하는 Machine과 Processor 이외의 하드웨어 자원을 제공하는 Machine 들로 구분된다. NVC Processor를 제공하는 Machine들은 NVC Resource Map을 download 받아서 NVC 자원의 위치를 파악하고 있다가 필요시 NVC Resource Map을 이용하여 원거리에 있는 자원을 활용하게 된다.

MMU의 동작 모델은 그림 6과 같이 Processor가 있는 곳의 기억장치를 이용하다가 기억장치가 부족한 경우는 NVC Resource Map을 이용하여 다른 Machine에 있는 기억장치를 이용할 준비를 하고, 그 내용을 NMU(Network Management Unit) 인터페이스를 통해서 NMU에 요구하게 된다. NMU는 이 요구를 전달받고 해당 Machine에 이를 전달하여 해당 Machine의 NMU 인터페이스를 통해 NMU 기억장치를 이용하게 된다. SMU(Storage Management Unit)와 IOMU(Input Output Management Unit)의 동작 모델도 이와 유사하다.

4.3 NMU(Network Management Unit)

NMU는 PMU/NMU/IOMU/SMU의 요청에 의해 CPU/Memory/IO/Storage의 주소 정보를 look-up한다(look-up address table).

NMU는 NVC 시스템에 속해 있는 CPU, Memory, Storage 시스템 및 I/O 시스템간의 통신 기능을 원활히 수행할 수 있도록 관리 및 제어하는 기능을 수행하게 한다. NVC의 자원들을 최적으로 매핑하기 위해서는 이들 각 자원들에 관한 규격, 가용 용량 및 가용 처리 능력 등을 사전에 파악하여 효과적인 네트워킹이 이루어지도록 하여야 할 것이다. 이를 위하여 NMU는 NVC 시스템내의 PMU, MMU, IOMU 및 SMU들이 호환성 있는 통신 기능을 제공하게 되며, 효과적인 통신 기능을 위해서 각 unit간의 접속 및 요구되는 정보의 고속 전송 등을 관리 제어하게 된다(그림 7). NVC 시스템 내에서의 효과적인 네트워킹을 위해서 NMU는 필요에 따라 다음의 기능을 가지게 된다.

- CPU Resource Address Resolution Protocol
- Memory Resource ARP
- I/O Resource ARP

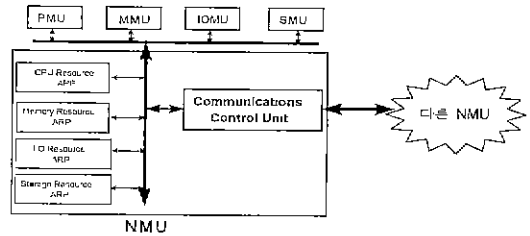


그림 7 NMU의 동작 모델

- Storage Resource ARP
- Communications Control Unit(CCU)

NMU는 NVC 시스템내의 각 Resource들에 대한 주소를 지닌 table 제공하고 관리하는 ARP 기능을 가지고 있으며, 초고속 통신망의 NVC 시스템 버스 역할을 수행하는 통신 제어 기능인 CCU를 가진다. NVC 시스템 내에서 Resource들간의 네트워킹에서 NVC 시스템의 각 Resource들간의 통신에서 고정적인 Physical networking 기능으로는 NVC 시스템의 확장성 및 다양성을 제공할 수 없다. 따라서 NVC 시스템 내에서 Resource들의 프로세싱, 저장 및 I/O등이 필요에 따라 가장 빠른 시간 안에 동적으로 Resource간의 통신 기능을 제공한다. 즉, NVC 시스템 요구 사항 등을 고려할 때 NVC 시스템 Resource들이 필요할 때, 적절한 대상끼리 효과적으로 가상 회선 설정을 설정하고, 고속으로 필요한 데이터를 전달할 수 있는 동적인 Virtual Networking 기능이 효과적으로 제공한다. 이를 위해 기능적으로 초고속 네트워크 플랫폼이 제공되고, 네트워킹 기능을 위한 주소 관리 및 고속의 전송 프로토콜 등이 Virtual Networking에서 제공된다. 이를 바탕으로 NVC 시스템의 각종 정보들은 상호 교환되는 기능을 갖게 된다. 특히, 전송 프로토콜에서 기존의 TCP 혹은 새로운 기능의 VLTP(Very Light-weight Transport Protocol) 등이 NVC

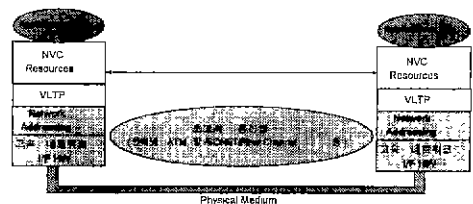


그림 8 통신 프로토콜 플랫폼 구조

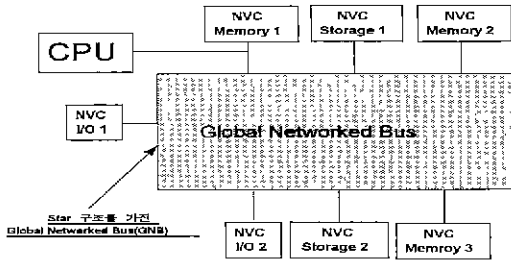


그림 9 virtual networking에 의한 GNB 구성

통신 인터페이스 기능에서 제공된다. 기능적인 측면에서의 Virtual Networking 통신 프로토콜 구조는 다음과 같다.

4.4 Global Networked Bus

NVC에서의 초고속 통신망은 내부 버스 기능으로서의 역할을 하며 GNB(Global Networked Bus)로 볼 수 있다. 이를 통해서 필요한 정보를 교환하고 전달하는 기능이 수행된다. 즉, 원격지의 컴퓨터 자원들이 serial 데이터 전송 방식을 사용하는 통신망이 단일 혹은 복수 개의 네트워크 정보 채널을 이용하여 정보를 교환하게 된다.

이러한 기능은 일종의 네트워크 기능에 의한 BUS 기능을 수행하는 것이다. 그림 10~그림 14는 GNB를 이용한 NVC 시스템의 구성 및 NVC 시스템의 GNB 구성을 위한 virtual connection 및 resource availability checking 절차, GNB 구성을 위한 resource 및 virtual connection 정보 Table 작성 과정, 초고속망의 네트워킹에 의한 NVC 시스템의 구축 예, NMU interface와 memory/storage와의 Data Flow 절차를 도해한 것이다.

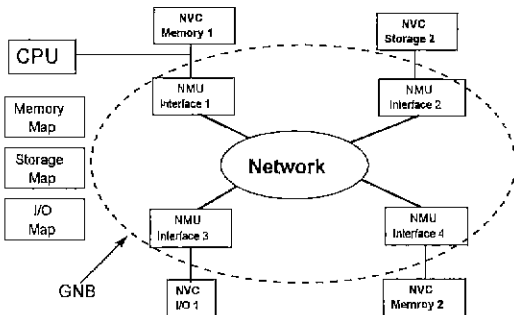


그림 10 GNB를 이용한 NVC 시스템의 구성

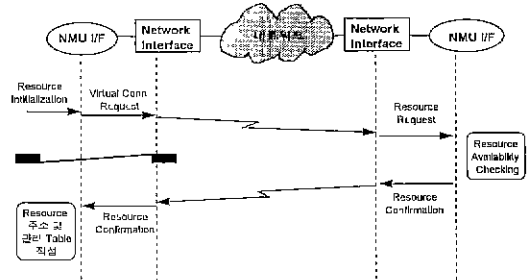


그림 11 NVC 시스템의 GNB 구성을 위한 virtual connection 및 resource availability checking 절차

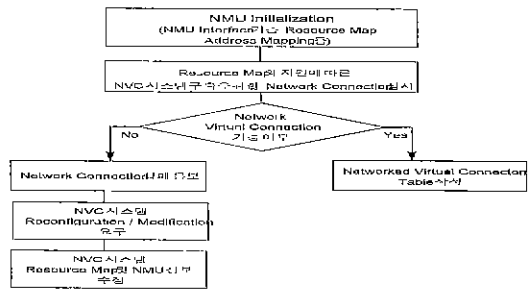


그림 12 GNB 구성을 위한 resource 및 virtual connection 정보 Table 작성 과정

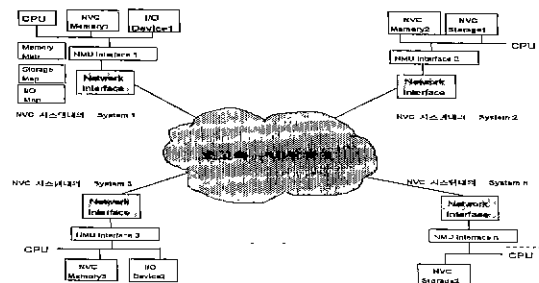


그림 13 초고속망의 네트워킹에 의한 NVC 시스템의 구축 예

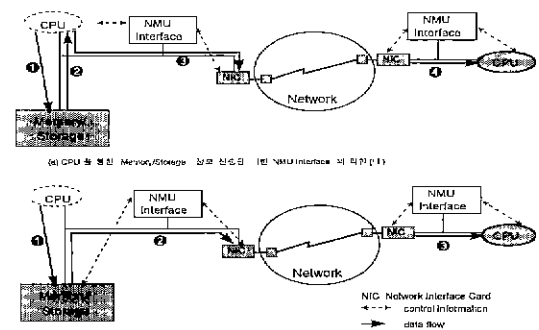


그림 14 NMU interface와 memory/storage와의 Data Flow 절차

4.5 NVCware 및 NVC의 동작 체계

NVCware는 NVC 시스템을 사용하는 이용자에게보다 편리하게 NVC를 구성할 수 있도록 해주며, 네트워크 상의 하드웨어, 소프트웨어 및 physical 네트워크 구조에 무관하게 NVC를 구성할 수 있도록 해준다. 따라서, NVCware는 사용자와 상호작용 하면서 NVC를 구성하고 이를 관리하는 기능을 갖는다. NVCware는 NVC 시스템을 구성하는 NVC O&M의 기능을 가지고 있는데, 이러한 NVCware가 가지는 기능은 ①NVC 프로세싱 unit 구성, ②NVC 메모리 구성, ③NVC 보조기억 장치 구성, ④NVC 네트워크 구성, ⑤NVC 응용 소프트웨어 구성, ⑥NVC 단말 구성, ⑦NVC 시스템 사용료 청구이다.

NVC 시스템의 종류는 NVC를 구동하는 프로그램에 따라서 NVC의 computing power를 결정하고 reconfiguration 기능에 따라서 3가지로 구분된다.

- 1) User Defined Model : 사용자가 NVC를 구동하기 전에 필요한 h/w resource를 define하고 이를 바탕으로 NVC가 configuration한다.
- 2) Program Adaptive Model : NVC가 사용자의 program을 보고 자동으로 h/w resource를 define하고 configuration한다.
- 3) Process Dynamic Model : NVC가 사용자의 프로그램을 보고 h/w resource를 define 및 configuration하고 프로그램 수행중에 프로세스에 맞추어 NVC의 각 component를 reconfiguration한다.

사용자는 시스템을 구동하는 과정에서 컴퓨

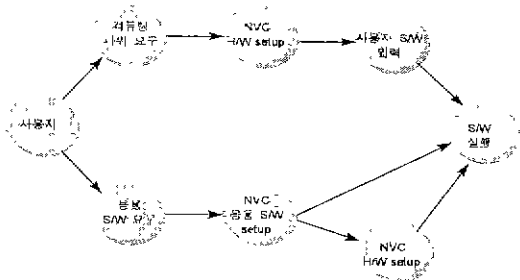


그림 15 사용자의 요구에 따른 동작 체계

팅 파워 또는 응용 소프트웨어 기능을 요구할 수 있다. 이를 위해서 NVC는 하드웨어 셋업과 NVC 응용소프트웨어 셋업 과정이 필요하다.

하드웨어 셋업은 사용자가 요구하는 컴퓨팅 파워를 만족하는 NVC 시스템을 구성하기 위해 여러 종류의 이동 에이전트가 이용된다. 즉 필요한 컴퓨팅 파워를 가지는 프로세서를 구성하기 위해서 CPU 이동 에이전트는 네트워크 상에서 적절한 프로세서들을 수집한다. 같은 방식으로 메모리 이동 에이전트와 I/O 이동 에이전트가 동작하게 된다.

NVC 응용 소프트웨어 셋업은 사용자가 요구하는 응용 소프트웨어를 네트워크 상에서 발견하기 위해서 이동 에이전트가 이용된다. 이동 에이전트는 사용자가 요구하는 응용 소프트웨어를 탐색하기 위해서 네트워크에 연결된 컴퓨터들을 이동한다. 이때 발견된 응용 소프트웨어가 기존의 시스템에 의해서 실행 가능한 상태이면 그대로 실행하고 여러 가지 상황에 의해서 발견된 응용 소프트웨어가 탑재된 컴퓨터에서 실행되기가 어려운 상황인 경우에는 NVC 하드웨어를 셋업하여 NVC 시스템에서 실행되도록 한다.

4.6 NVC Resource Mapping 동작 체계

그림 16은 NVC의 Dynamic Resource Mapping 구조를 나타내고 있다. 분산 환경에서 효과적인 Mapping을 위해서 Hierarchical 구조를 지닌 Agent들이 필요하며, 이 Agent들은 분산 네트워크에 산재하고 있는 컴퓨팅 자원들을 찾아서 가장 적절한 NVC 시스템을 구축하게 한다. 따라서, 최종적인 Physical Mapping이 이루어지기까지 상위 계층의 Agent

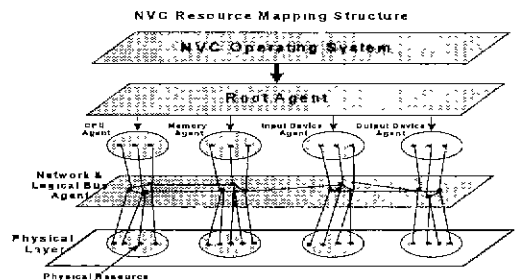


그림 16 NVC의 Resource Mapping 구조도

들어 해야 하는 주요 기능, 관련 절차 및 관리 기능 구조에 관한 세부적인 연구가 중점적으로 이루어져야 할 것이다.

5. 결 론

본 논문은 앞으로의 컴퓨터의 진화 방향을 네트워크를 기반으로 하는 가상컴퓨팅으로 보고 이의 기술 개발 방안을 연구하였다. NVC의 구체적인 모델을 위하여 서비스 측면의 요구 사항 분석, 환경 분석, 기능 분석을 하고 NVC의 구조를 정립하였다. 또 이 구조의 타당성을 보이기 위해 사용자 이용 환경인 NVC 시스템에서 PMU, MMU, NMU, GNB의 동작 모델을 보였고 사용자의 요구에 따른 NVC의 동작 체계와 자원의 맵핑 동작 체계를 보였다.

NVC의 기술 개발은 NVC사용자가 요구하는 서비스를 처리하는 시스템(Service on Demand)과 NVC사용자가 요구하는 컴퓨팅 파워를 제공하는 시스템(Computing on Demand)의 두 측면에 대하여 이루어져야 한다,

참고문헌

[1] Keith D. Kotay and David Kotz, "Transportable Agents", Dartmouth College, Nov. 1994.

[2] Mike Addlesee, Chris Turner and Andy Hopper, "Displaying the Future", ORL Technical Report 94. 13.

[3] Tristan Richardson, "Teleporting-Mobile X Sessions", ORL Technical Report 95. 7.

[4] A Beguelin, J. Dongarra, A. Geist and R. Manchek, "HeNCE : A Heterogeneous Network Computing Environment", Scientific Programming, Vol. 3. No. 1, 1994.

[5] S. Casselman, "Virtual Computing and the Virtual Computer", Proceedings of IEEE Workshop on FPGAs for Custom Computing Machines, April 1993.

[6] Al Geist et al, "User's Guide and Tech-

nical for Networked Parallel Computing", Technical Advisory for PVM, June 1995.

[7] Colin G. Harrison, David M. Chess and Aaron Kershenbaum, "Mobile Agents : Are they a good idea?", IBM Confidential, Mar. 95.

[8] ANSI, "Fiber Channel Standards-FC Protocol for HTTPPI", andi x 11, 1994.

[9] David Chess, Henjamin Grosf, Colin Harrison, David Levine, Colin Parris and Gene Tsudik, "Itinerant Agent for Mobile Computing", IBM Tech. RC20010, Mar. 1995.

[10] Katy Ring and Philip Carnelly, "Distributed Objects-Creating the Virtual Mainframe", Ovum Ltd, 1995.

[11] Don Tolmie and Don Flanagan, "HIPPI : It's not Just for Supercomputers Anymore", Data Communications, May 1996.

[12] Saurab Nog, Sumit Chawla and David Kots, "An RPC Mechanism for Transportable Agents", Dartmouth Technical Report PCS-TR96-280, Dartmouth College, Mar. 96.

[13] Robert S. Gray, "Agent Tcl : A Transportable Agent System", Dartmouth College, Nov. 1995.

[14] Christine Guilfoyle and Ellie Warner, "Intelligent Agents : the New Revolution in Software", Ovum Ltd, 1994.

[15] Kenneth E. Harker, "TIAS : A Transportable Intelligent Agent System", Dartmouth Technical Report : PCS-TR.95-258, Dartmouth College, Jun. 1995.

[16] Robert S Swain, "A Race Mobile Project Line Assembly Vision", UMTS Report.

양 통



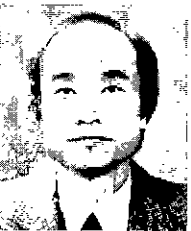
1972 한국항공대학교(공학사)
 1974~1979 동의공업전문대학 교수
 1979~현재 인하공업전문대학 전자계산과 교수
 1980 동아대학교 대학원(공학 석사)
 1990 단국대학교 대학원 컴퓨터 전공(공학박사)
 1991~1994 포스테이타(주) 기술고문, 연구소장
 관심분야: 컴퓨터구조, 멀티미디어, HCI

박 영택



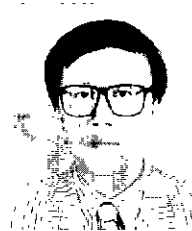
1978 서울대학교 전기공학과(공학사)
 1980 한국과학기술원 진산학과(공학석사)
 1981~현재 숭실대학교 정보과학대학 컴퓨터학부 교수
 1992 University of Illinois at Urbana-Champaign 전산학과(공학박사)
 관심분야: 인공지능

임 영환



1977 경북대학교 수학과(이학사)
 1979 한국과학기술원 전산학과(이학석사)
 1979~1996 한국전자통신연구소 책임연구원
 1985 Northwestern University(이학박사)
 1996~현재 숭실대학교 정보과학대학 컴퓨터학부 교수
 관심분야: 멀티미디어

정 일영



1980 경북대학교 전자공학과(공학사)
 1980~1996 한국전자통신연구소 실장
 1990 University of Massachusetts 전산학과(공학석사)
 1993 University of Massachusetts 전산학과(공학박사)
 1996~현재 한국의국어대학교 정보통신과 교수
 관심분야: 고속통신프로토콜, ATM

이 경호



1991 한국방송통신대학 전산학과(이학사)
 1994 한국과학기술원 정보및통신공학과(공학석사)
 1994~1996 (주)미디어랜드 연구실장
 1996~현재 한라공과대학교 전기공학부 교수
 관심분야: 멀티미디어, 음성인식, 인공지능

박 치 항



1974 서울대학교 응용물리학과(이학사)
 1974~1978 한국과학기술연구소 연구원
 1978~1985 한국전자기술연구소 선임연구원
 1980 한국과학기술원 전산학과(이학석사)
 1985~현재 한국전자통신연구소 책임연구원, 컴퓨터연구단 단장
 1987 파리 6대학 전산학과(공학박사)
 관심분야: 멀티미디어, 분산시스템, 그룹웨어, 네트워크컴퓨터, 에이전트 아키텍처, 가상현실