

유틸리티 컴퓨팅 시대를 여는 가상화 기술 동향

김진미* 배승조** 정영우** 심규호*** 고광원*** 우영춘****

컴퓨팅 기술이 빠른 속도로 발전하여 시스템들이 네트워크를 통해 서로 연결되고 시스템과 업무들이 연계되면서 그에 맞춰 시스템 구조나 관련 소프트웨어는 엄청나게 복잡해지게 되었다. 가상화는 사용자에게 컴퓨팅 자원을 사용함에 있어 단순하고 일관성 있으며 편리한 논리 구조를 갖게 하여 비용 및 인프라 관리를 향상시킨다. 궁극적으로 이러한 가상화 기술은 자율 컴퓨팅, 그리드 컴퓨팅 및 고정적인 컴퓨팅 비용을 가변적으로 바꿀 수 있는 유틸리티 컴퓨팅 기술에 포함되어 양질의 서비스를 제공할 수 있는 기반 기술로 다시 자리 매김하고 있다. IT 가상화 솔루션은 소프트웨어와 하드웨어간의 밀착된 관계를 느슨하게 만들어 IT의 관리와 운영에 있어 보다 관리하기 쉽고 효율적인 컴퓨팅 인프라스트럭처를 구성할 수 있는 기술임에 틀림없다. ☐

목 차

- I. 서 론
- II. 가상화 기술 동향
- III. 가상화 소프트웨어
- IV. 결 론

I. 서 론

기계 중심의 컴퓨팅 환경에서 사람 중심의 유비쿼터스 컴퓨팅 환경으로 이전하면서 사용자는 요구에 따라 언제 어디서나 의식하지 않는 상태에서 자유롭게 네트워크에 접속하여 서비스를 받고자 하는 요구가 생겨나고 있다. 이러한 다양한 서비스 중에는 고성능 컴퓨팅을 필요로 할 수 있으며 이러한 서비스의 요구를 효율적으로 처리할 수 있는 방안이 마련되어야 한다. 이러한 요구는 언제 어디서나 어떤 사용자들에게도 고품질의 맞춤형 서비스를 투명하게 제공할 수 있는 가상의 서비스 인프라 혹은 서비스 전달망을 구축하고 관리하는 기술을 필요로 한다. 최근 이에 대한 요구로 성능 및 안정성에 중점을 둔 자원 가상화에 대한 기술 개발이 이루어져 왔으

* ETRI 공개 S/W 솔루션연구팀/책임연구원
 ** ETRI 공개 S/W 솔루션연구팀/선임연구원
 *** ETRI 공개 S/W 솔루션연구팀/연구원
 **** ETRI 공개 S/W 솔루션연구팀/팀장

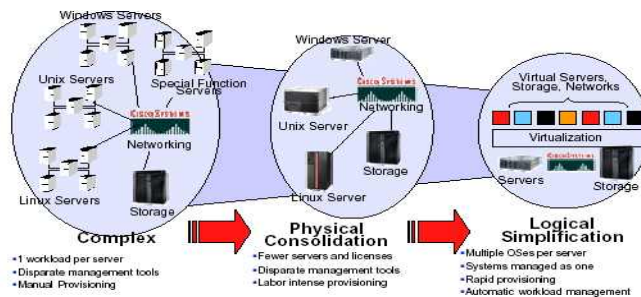
며 이를 이용하여 비용 관리와 인프라 관리를 자동으로 해주는 궁극적인 서비스 형태인 유틸리티 컴퓨팅 단계 중심으로 연구되어 왔다. 유틸리티 컴퓨팅은 온디맨드 컴퓨팅, 정책 기반 컴퓨팅 서비스, 서비스 중심 컴퓨팅이라는 용어와 혼용되어 사용되고 있다[1].

본 고에서는 이러한 컴퓨팅 환경의 실제 기반 기술인 가상화 기술에 대한 동향 및 기술들에 대해 설명하였다. 2 장에서는 가상화 기술 발전 배경 및 가상 인프라를 가능하게 하는 각 단계별 가상화 기술 동향에 대해 알아보고자 한다. 3 장에서는 공개 소스 기반의 가상화 소프트웨어 및 가상화 기술이 적용된 상용 가상화 기술에 대해 기술하고, 마지막 4 장에서 결론을 맺는다.

II. 가상화 기술 동향

컴퓨팅 기술이 빠른 속도로 발전하여 시스템들이 네트워크를 통해 서로 연결되고 시스템과 업무들이 연계되면서 그에 맞춰 시스템 구조나 관련 소프트웨어는 엄청나게 복잡해지게 되었다. 이러한 진화의 결과로 생산성의 향상은 이루어 졌으나 복잡해진 시스템을 운영하기 위한 비용의 증가와 IT 환경의 유연성 부족으로 사용되고 있지 않은 시스템 자원의 낭비가 발생하고 있다. 이를 해결하기 위해서는 다양한 컴퓨팅 자원을 보다 유연하게 사용할 수 있도록 하는 환경이 마련되어야 하며 이 중 기반이 되는 기술이 가상화 기술이라고 볼 수 있다.

가상화란 사전적인 의미에서 사실이 아니거나 사실 여부가 분명하지 않은 것을 사실이라고 가정하여 생각하는 것으로 컴퓨팅 기술에서의 가상화란 실제로 지니고 있는 물리 구조를 적당한 계층을 개입시킴으로써 사용자에게 일관성 있고 편리한 더 좋은 논리 구조를 갖게 하는 것으로 볼 수 있다. 이는 하드웨어부터 서비스에 이르기까지 여러 계층에 걸쳐 적용할 수 있으며 ((그림 1)은 복잡한 컴퓨팅 시스템 환경을 논리적으로 가상화시켜 단순화한 환경을 보여준다.



<참조>: Greg Hintermeister, Putting the you in Virtualization, IBM, 2004

(그림 1) 컴퓨팅 시스템 가상화 환경

가. 가상화 분류

가트너 그룹에 의하면 엔터프라이즈 컴퓨팅 환경이 현재는 성능 및 안정성에 중점을 둔 선택적 가상화 및 IT 통합 단계에서 자원을 가상화하고 공유하며 비용 관리 및 인프라 관리를 자동으로 해주고 신속한 변경 및 최적화를 지원하는 유틸리티 컴퓨팅 단계를 거쳐 2010 년경에는 정책기반 컴퓨팅 서비스 단계로 진화해 갈 것으로 예측하고 있다[2]. 또한 서버 시장의 미래를 결정하는 서버 유형의 변화에 있어 서버 가상화, 블레이드 서버, 그리고 그리드 컴퓨팅을 그 중요 유형으로 하고 있다[3].

가상화는 스토리지, 서버, 네트워크 및 기타 자원들에 대해 논리적인 통합으로 적용 업무의 중단 없이 IT 자원을 효율적으로 사용할 수 있도록 한다. 초기에 IT 에서 폭넓게 사용되고 있는 가상화라는 개념은 스토리지, 서버 및 네트워크 도메인에 이르기 까지 물리적인 자원을 가상적으로 묶어 큰 풀을 형성하고, 자원이 동적으로 추가, 삭제, 변경될 수 있도록 하여 애플리케이션에 할당 및 반환되어 사용할 수 있게 하는 기술이다. 이 단계에서 발전된 가상화 기술은 단순히 물리적 자원의 가상화뿐만 아니라 분산된 이기종 환경의 모든 IT 자원을 가상화된 자원으로서 정의하고 조작하는 것을 목표로 하고 있다. 이를 지원하기 위해서는 개방 및 표준화된 인프라가 필요하며 OGSA 와 같은 공개 표준 서비스 기반에서 가상화된 자원을 서비스처럼 운영할 수 있게 되고, 성능, 확장성, 데이터 가용성, 애플리케이션 가용성, 호환성 및 자원 최적화를 가능하게 한다. 다음은 일반적으로 분류되는 스토리지, 서버, 네트워크 및 서비스 가상화에 관한 설명이다.

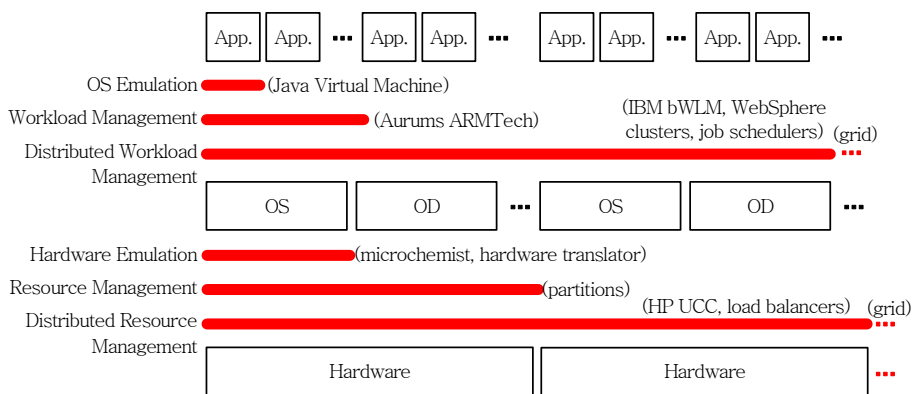
(1) 스토리지 가상화

스토리지 가상화는 시스템들이 스토리지의 물리적인 위치를 파악할 필요가 없이 여러 대의 스토리지를 하나의 스토리지 저장 풀로 만들어서 여러 기기에 걸친 볼륨 구성, 기기 간 자유로운 데이터 마이그레이션 및 볼륨 복제, 이기종 서버간 블록 단위의 데이터 공유 등을 가능할 수 있게 하는 기술이다. 일반적인 스토리지 가상화 솔루션은 스토리지와 서버 사이에 가상화 계층을 두고 서로 간의 접근을 제어한다. 가상화를 가능하게 하기 위해서는 크게 두 레벨에서 처리되어야 하는데, 하나는 스토리지 어드레스를 재매핑, 재설정하여 가상의 스토리지 용량 풀을 만들어 주는 접근 수준에서의 처리, 다른 하나는 해당 애플리케이션과 스토리지 사이의 데이터 경로를 개발, 제공, 유지시키는 기능을 하게 되는 컨트롤 및 관리 수준에서의 처리이다. 스토리지 가상화의 가장 큰 이점은 복잡한 스토리지 구성을 단순화하여 관리 효과를 얻을 수 있고 스토리지의 구성 변경이 서버에 영향을 주지 않으므로 서버의 가용 시간을 늘릴 수 있는 것이다. 또한 전체 스토리지 시스템 운용에 필요한 비용 절감을 얻을 수 있다. 스토리지 가상화 솔루션은 물

리적인 위치에 따라 서버 기반, 네트워크 기반, 스토리지 기반으로 나눌 수가 있으며 서버 기반 솔루션은 서버에 스토리지 가상화 소프트웨어를 설치하여 이 계층을 통해 가상화된 스토리지 볼륨에 접근하는 방식이다. 네트워크 기반 솔루션은 모든 스토리지가 상호 연결된 네트워크 장비를 지나면서 가상화되어 서버에 전달되는 방식으로 많은 부가 기술을 필요로 한다. 스토리지 기반 솔루션은 스토리지 컨트롤러에 가상화 솔루션을 탑재하는 방식으로 가상화 범위가 스토리지에 제한되므로 범용성이 떨어진다.

(2) 서버 가상화

서버 가상화는 이미 현재 운용 중인 운영체제 안에 또 하나의 가상머신을 만들어 마치 컴퓨터가 여러대 있는 것처럼 시스템을 구축하여 동시에 운영할 수 있으며 이는 다중 운영체제와 하드웨어 사이에서 가상화된 유형이다. 일반적으로 서버 가상화는 CPU, 메모리, 입출력 등의 다양한 서버 자원을 하나의 집합으로 묶어서 사용자의 필요에 따라 자유롭게 나누어 사용하는 개념으로 작은 여러 대의 서버들을 보다 쉽게 관리하고, 재할당하여 효율을 높이는데 더욱 유용하게 사용된다. 가트너에서는 ((그림 2)와 같이 서버 가상화의 유형을 여섯 가지로 나누었다. 운영체제가 가상화의 한 계층이 되어 현대의 대칭형 멀티 프로세싱 (SMP) 서버에서 애플리케이션을 보다 효율적으로 구동하는 형태, 자바 가상머신과 같은 애플리케이션과 운영체제 사이에서의 가상화, 파티션과 같이 하드웨어와 운영체제 사이에서 가상화한 형태, 또한 다중 운영체제와 하드웨어 사이 등이 가상화 계층으로 정의될 수 있다. 이러한 서버 가상화로서 얻을 수 있는 이점은 각각의 서버에 대한 이용효율을 증대시킬 수 있다는 것이다.



(그림 2) 서버 가상화

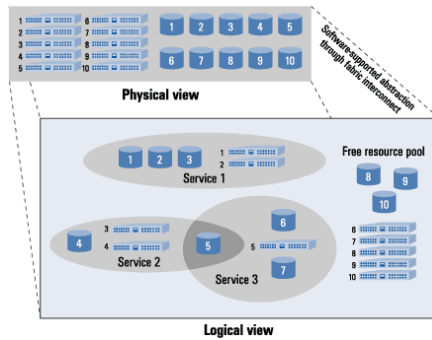
(3) 네트워크 가상화

네트워크 가상 환경은 여러 사용자들을 네트워크를 통해서 하나의 공통된 가상 환경으로 묶어주는 기술이다. 이는 여러 명의 사용자가 같은 가상 공간을 공유하면서 협동 작업을 가능하게 한다. 네트워크에서의 공유되어 사용할 수 있는 자원은 라우터, 스위치, 로드 발란서, 방화벽과 서버와 연결되어서는 VPNs, HiperSocket, VLANs, VSWITCH, Virtual IP, Ethernet 어댑터 등의 자원들이 될 수 있다. 이러한 자원들을 효율적으로 사용하기 위해서는 사용자에 대한 물리적인 자원을 분리할 수 있어야 하고 오류 발생시에 가용성을 제공하여 융통성 있는 복구가 행해져야 한다. 온디맨드 컴퓨팅을 제공하기 위해 네트워크 가상화는 서비스 계층에서 관리되어야 한다. 애플리케이션이나 서비스에서 요구되는 네트워크 자원이 대역폭이나 우선순위를 고려하여 자동으로 적용되어야 하고, 이러한 과정들이 QoS 를 보장하는 정책 기반으로 단순하게 관리될 수 있어야 한다.

(4) 서비스 가상화

최근 컴퓨팅 환경이 서비스 중심으로 발전하면서 사용자들은 표면에 들어난 가상화된 서비스만을 호출하여 서비스를 받고자하고 있다. 이를 가능하게 하기 위해서 서비스 가상화는 앞서 설명한 스토리지, 서버, 네트워크 가상화 기술과 복합적인 모델을 지원하여 서비스 중심 컴퓨팅 동작이 가능하게 구현되어야 한다. 최근 이를 지원하기 위한 환경으로 적합하게 적용되는 기술이 웹서비스이며 이질적인 시스템들간에 표준화란 연결고리를 사용하여 흩어진 자원들을 사용할 수 있도록 한다.

기존의 웹 접속이 사람에 의해 접속되어 사용된 것이라면 웹서비스는 가상화된 서비스가 웹에 접속하여 정보를 요청하고 요청된 서비스는 관련되는 서비스들에 의해 요청된 정보의 응답을 받게된다. 즉, 웹서비스는 인터넷상에서 서로 통신할 수 있는 서비스를 지칭한다. 웹서비스의 기술은 서비스 지향 구조인 SOA(Service Oriented Architecture)를 기반으로 서비스 단위의 통합을 이루어 유연하고, 능동적인 웹 애플리케이션 수행 환경을 제공하는 기술이라고 정의될 수 있다. 이런 측면에서 앞으로 웹서비스는 가상화된 서비스를 전달하는 주요 요소라고 할 수 있으며 가상화된 서비스는 ((그림 3)에서와 같이 서로 이질적인 시스템 자원들을 투명성있게 사용할 수 있도록 한다. 서비스 가상화를 이루기 위해서는 웹서비스를 활용하여 흩어진 서비스들을 구성하고 활용할 수 있도록 해야 할 뿐만 아니라 서비스들을 배치하고 관리하는 작업들이 이루어져야 한다.

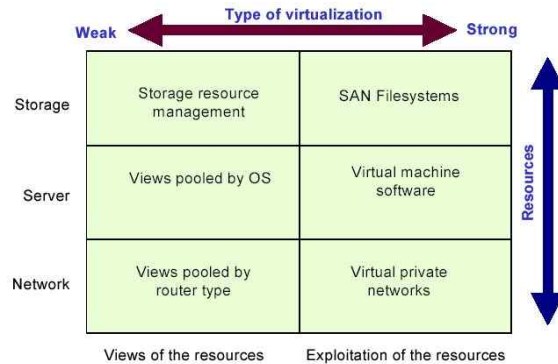


<참조>: Dell 2003.

(그림 3) 하드웨어를 기반으로 하는 Service 레벨 가상화

(나) 가상화 기술

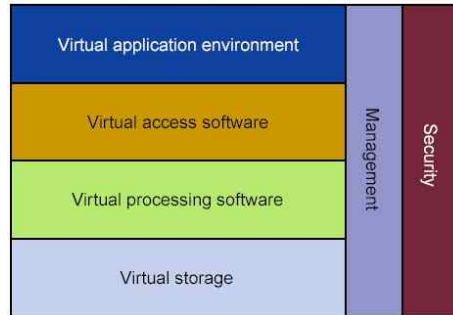
다음은 IDC 기술 문서를 참조하여 정의한 가상화 기술 및 프로비저닝 기술에 대해 기술하고자 한다[4]. 가상화 기술은 시스템의 거의 전 분야에서 활용될 수 있으며 ((그림 4)는 대표적인 각 단계에서 초기 단계의 약-가상화 유형과 발전 단계에 있는 강-가상화 유형을 보여주고 있다.



<참조>: IDC 2003.

(그림 4) 가상화의 약/강 단계

초기 단계의 약-가상화 유형이 자원을 물리적인 관점 및 유형 수준에서 단순히 풀 형식으로 관리하는 수준임에 반해 강-가상화 유형은 정책 기반의 자원 프로비저닝과 관리에 중점을 둔다. 즉, 약-가상화가 단지 자원의 가상화된 면모만 보여준다면 강-가상화는 자원을 이용하기 위해 가상화된 방법을 제공하는 것을 포함하는 것으로 볼 수 있다. IDC 에서 분류하는 강-가상화 단계는 ((그림 5)의 모델로 설명된다



<참조>: IDC 2003.

(그림5) 가상 환경 모델

가상 스토리지 (Virtual Storage) 소프트웨어는 시스템들이 스토리지의 물리적인 위치를 파악할 필요가 없이 여러 대의 스토리지를 하나의 스토리지 저장 풀로 만들어서 여러 기기에 걸친 볼륨 구성, 기기 간 자유로운 데이터 마이그레이션 및 볼륨 복제, 이기종 서버간 블록 단위의 데이터 공유 등을 가능할 수 있게 하는 기술이다. 현재 Veritas Volume Manager/Volume Replicator, DataCore, FalconStor, Pirus Networks, Fujitsu Softek 등의 소프트웨어가 해당된다.

가상 처리(Virtual processing) 소프트웨어는 OS 수준에서 처리되는 개념이다. 일반적으로 서버 가상화는 CPU, 메모리, 입출력 등의 다양한 서버 자원을 하나의 집합으로 묶어서 사용자의 필요에 따라 자유롭게 나누어 사용하는 개념으로 여러 대의 서버들을 보다 쉽게 관리하고, 재할당하여 효율을 높이는데 더욱 유용하게 사용된다. 이러한 서버 가상화로서 얻을 수 있는 이점은 각각의 서버에 대한 이용효율을 증대시킬 수 있다는 것이다. 또한 이미 널리 사용되고 있는 클러스터링 및 가용성 소프트웨어도 가상 처리 소프트웨어에 해당된다. 현재 IBM 의 z/VM, VMware 의 GSX, HP MC/Serviceguard, Sun SunCluster, Microsoft 의 cluster services, Veritas 의 ClusterServer, 그리드 등의 소프트웨어가 해당된다. 서버 가상화의 궁극적인 모습은 유틸리티 컴퓨팅을 가능하게 하는 정책 기반의 가상화된 프로비저닝 모델로 온디맨드 환경에서 자원을 실제적으로 가상화하는 기술이다.

가상 접근(Virtual access) 소프트웨어는 사용자 인터페이스를 가상화 할 수 있도록 하는 개념으로 사용자는 애플리케이션이 실제 어디서 구동되는지 몰라도 수행 가능하도록 해준다. 현재 이러한 소프트웨어로는 Citrix, Microsoft, Aspell, GraphOn, Tarantella 등에서 개발되고 있다

가상 애플리케이션 환경(Virtual application environment)은 하나의 애플리케이션이 한 서버에서만 동작하는 기존의 애플리케이션 수행 환경과는 달리 하나의 애플리케이션이 다수의 독립

적인 시스템들에서 컴포넌트들로 구성되어 실행할 수 있는 가상화된 애플리케이션 환경을 가능하게 하는 모델이다. 현재 표준을 기반으로 하는 웹서비스 구조 등이 해당되며 IBM의 애플리케이션 서버인 WepSphere 환경이 대표적이다.

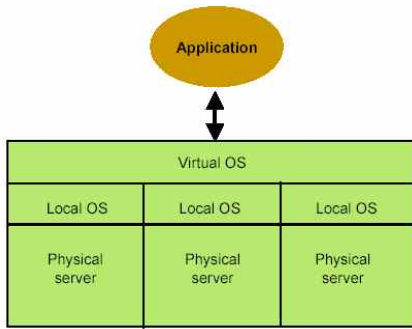
(다) 가상화 특성

가상화는 애플리케이션의 요구에 대해 다양한 목적을 가진다.

- 성능 (Performance) - 자원들을 하나의 풀로 만들어 최대한 컴퓨팅 파워를 사용하고자 성능을 향상시킴
- 확장성 (Scalability) - 서비스 요구량 및 트래픽의 폭발적인 증가에도 유연하게 대처할 수 있는 서비스의 확장성
- 가용성 (Availability), 신뢰성 (reliability), 회복성 (resiliency)- 하루 24 시간 1년 365 일 동안 중단 없이 서비스를 제공할 수 있는 데이터 및 서비스의 가용성
- 탄력성 (flexibility), 민첩성 (agility) - 긴급한 문제 발생 시에도 빠르게 대처하여 실시간으로 장애 복구가 가능한 서비스 인프라의 탄력성과 어떠한 상황 변화(가령, 시스템 자원의 증설/축소, 서비스의 추가/삭제 등)에도 빠르게 대처할 수 있는 서비스 인프라의 민첩성
- 자원 최적화 (resource optimization) - 물리적인 시스템의 재구성 없이도 다양한 비즈니스 요구에 맞는 자원의 활용을 최적화하게 구성

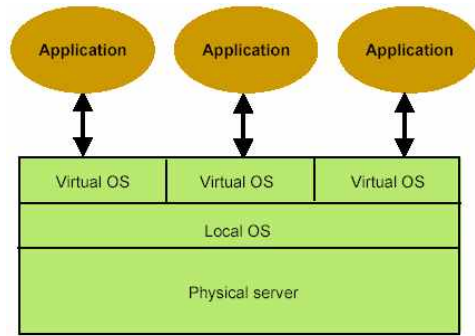
이러한 가상화의 목적들을 부합하기 위해서 가상화는 서버 및 서비스 인프라 환경에서 가상화의 개념적인 모델을 지원할 수 있다. 가상화를 서버에 적용시키기 위해서는 2 가지 개념으로 나눌 수가 있다. 첫 번째로는 여러 대의 이질적으로 연결된 물리적인 서버를 현대의 서버로 가상화해서 운용되는 Server Virtualizing Up 개념으로 ((그림 6)에 해당한다. 이는 공개소스 프로젝트로 진행중인 Squid 나 HP의 workload manager와 같은 부하 분산용 소프트웨어를 사용하여 여러 시스템을 클러스터링하여 사용자 요구를 분산시켜 운용함으로써 서버 가상화를 구현할 수가 있다. HP의 MC/Serviceguard나 MS의 Cluster Server와 같은 소프트웨어는 클러스터링되어 운영되는 시스템 장애 시에도 나머지 서버들에 의해 운용되므로 신뢰성 및 가용성을 높일 수가 있다. 두 번째로는 현대의 물리적인 서버를 여러 OS 상에서 운용되는 애플리케이션들을 정상 동작하기 위해 여러 OS로 파티션 하여 사용하는 Virtualizing Down의 개념으로 (그림 7)에 해당한다. 이러한 가상화는 일반적으로 서비스 인프라의 탄력성(flexibility)과 민첩성

(agility)을 향상시키고 자원의 최적화에 목적을 둔다. Virtualizing Down 에 해당하는 시스템으로는 주로 IBM 의 z/VM OS 가 많이 알려져 있다.



Source: IDC, 2003

(그림 6) Server Virtualizing Up Model

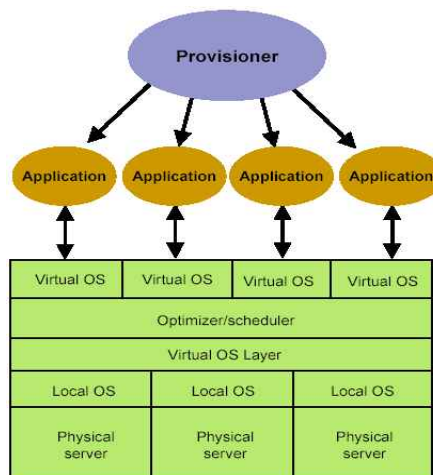


Source: IDC, 2003

(그림 7) Server Virtualizing Down Model

이러한 가상화의 개념들은 궁극적으로 복합적인 모델을 가능하게 한다. 서비스 중심 컴퓨팅, 유틸리티 컴퓨팅, 온디맨드 컴퓨팅 환경에서는 이러한 복합 모델을 지원 가능해야 한다. 온디맨드 컴퓨팅에서 이상적으로 요구되는 기능은 (

(그림 8)에서 보여지는 바와 같이 컴퓨팅 자원이 느슨하게 연결되어 있든 아니든 실질적인 컴퓨팅 환경에서도 가상화가 확장성 있게 제공되어야 하고, 언제 어디서나 가상 머신에서 자원의 추가 및 제거가 자유롭게 이루어져야 하며, 이로 인해 자원의 할당이 동적으로 이루어 질 수 있는 능력을 가져야 한다. 또한 비즈니스 목적에 맞게 최적화되고 스케줄링 되는 것이 필요하다.



Source: IDC, 2003

(그림 8) On Demand Computing의 Virtualized Provisioning Model

가상화의 목적이 성능, 확장성, 가용성, 호환성 및 자원 최적화임에 반해 운용의 입장에서는 이러한 가상화 서버를 자동으로 관리하고 구성하는 기술이 필수적이며 이에 해당하는 기술이 서버 프로비저닝 기술이다. IDC 에서는 가상화의 상위 수준에서 관리 역할을 하는 소프트웨어를 SPM(server provisioning and management) 소프트웨어라고 정의하였다. SPM 은 서버 시스템 과 애플리케이션 스택 이미지를 배치(deployment), 구성(configuration)하고 관리(management) 하는 일반적인 기능을 가진다.

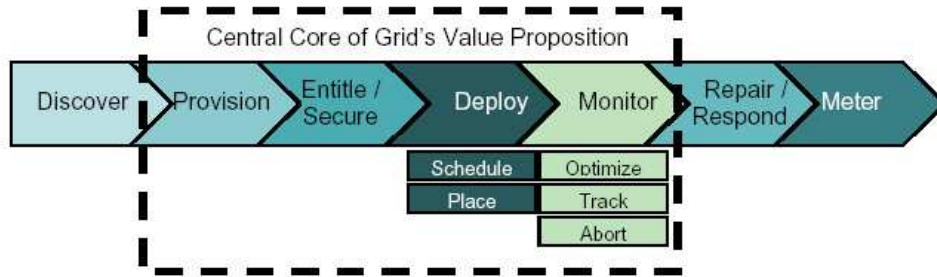
- 어디서나 한번에 자동 설치 및 배치가 가능한 소프트웨어(operating system, middleware, tools, applications) 패키징 기능
- SPM 서버의 편리한 중앙 관리
- 수정된 버전 및 패치의 분배, 관리, 설치 등을 포함하는 소프트웨어 분배 및 관리
- 웹 기반 관리
- 정책 기반에 따른 자원의 자동 준비 및 배치
- 클라이언트 시스템의 모니터링 및 기본적인 경고 전달 기능
- 하드웨어 소프트웨어의 자산 관리

살펴본 바와 같이 가상화 기술은 유틸리티 컴퓨팅의 실제 구현 기술로서 소프트웨어와 하드웨어간의 밀착된 관계를 느슨하게 만들어 관리하기 쉽고 효율적인 컴퓨팅 인프라스트럭처를 구성할 수 있는 기술로 서비스 인프라 구축 관리에 또한 필요한 기술이 될 것이다.

(라) 그리드 컴퓨팅의 가상화 및 프로비저닝

오늘날 컴퓨팅 환경은 이기종 시스템들이 인터넷에 분산되어 있고 사용자의 요구에 의해 분산된 환경에서 다양한 형태로 서로 연결되는 구조를 가진다. 이러한 환경에서 인터넷에 바탕을 둔 하나의 도메인 내 혹은 다른 외부 도메인과 애플리케이션, 데이터 및 자원을 동적으로 공유하기 위한 구조를 가지는 그리드 컴퓨팅은 인터넷에 분산된 웹 기반의 서비스를 사용할 수 있는 좋은 인프라가 된다. 초창기 그리드 컴퓨팅은 사용되지 않고 있는 컴퓨팅 파워를 활용하는 컴퓨팅 그리드 환경으로 과학 기술 분야에서 많이 사용되어 왔으나 최근에 들어 상용분야에서의 요구가 활발해지자 현재 그리드 기술은 서비스 지향 방식의 분산 컴퓨팅 환경에서 컴퓨팅 인프라로 활용되고 있다. 특히 차세대 인터넷 서비스를 위한 그리드 기술은 초고속 인터넷 인프라가 잘 갖추어져 있는 한국에서 이동성에 대한 보완과 사용의 용이성을 적극적으로 대처함으로써 적응적인 유비쿼터스 환경을 제공하기 위한 컴퓨팅 기반 인프라로 좋은 도구가 될 수 있으며 이를 통해 웹 기반의 정보 강대 국가로 더욱 굳건히 자리매김할 수 있을 것이다. 나아가 유비쿼터스 컴퓨팅 기술도 그리드 지식 기반 인프라 위에서 더욱 발전할 수 있다[5].

그리드 컴퓨팅 기술은 자원 공유를 위한 풀링, 가상화, 필요한 자원을 동적으로 분배하는 동적 프로비저닝, 자체 적응형 시스템, 통합 관리 등의 기본 기술을 목표로 한다. 최근 그리드는 GGF(Global Grid Forum)의 주도하에 웹서비스 표준을 통합하여 OGSA(Open Grid Service Architecture)로 불리는 개방형 표준에 의해 제시되고 있다. 최근에는 기존의 웹서비스 인프라에 사용할 수 있게 보완된 OGS(Open Grid Service Infrastructure)를 재정립하여 WSRF(Web Service Resource Framework)를 정의하여 그리드 서비스 개념 대신 WS-resource 개념이 도입되었다. 이러한 OGSA는 궁극적으로는 유틸리티 컴퓨팅, 온디맨드 컴퓨팅, 시스템 관리, 협업 컴퓨팅과 같은 요구 사항들을 충족시키고자 만들어졌다. 그리드는 다음 ((그림 9)에서 보는 바와 같이 discover, provision, entitle/secure, deploy, monitor, repair/respond 와 meter 와 같은 7가지 주요 기능으로 기술을 개발하고 있다[6]. 이 그림에서 보는 바와 같이 그리드는 크게 시스템의 자원과 관련된 기능들을 의미하는 가상화와 언제 어디서나 자원을 가용하게 하는 프로비저닝 기술을 기반으로 한다.



Source: The Tabb Group

(그림 9) 그리드 컴퓨팅의 주요 기술 방향

현재 그리드는 웹서비스 표준안을 기반으로 효율적인 자원 관리를 위해 확장된 표준안인 WSRF 를 구현한 GT(Globus Toolkit) 4.0 이 2005 년 4 월에 공개 소스 미들웨어로 발표되었다. GT 4.0 에는 Java WS Core 와 Security Infrastructure(GSI), GridFTP, Execution Service(GRAM), Information Service(MDS4)와 같은 기본 서비스 모듈로 구성되어 있다[7]. 또한 가상화를 통해 자원을 관리하며 대규모 자원의 공유를 가능하게 하고 이는 ((그림 10)의 자율컴퓨팅을 이루는 기반이 된다[8].



<참조>: IBM 2005.

(그림 10) 그리드 컴퓨팅의 발전 방향

IBM, HP, 오라클을 비롯하여 최근에 들어서 상용분야에서의 요구가 발전되면서 그리드는 서비스 통합에 유용하게 쓰여지는 웹서비스 기술과 융합하여 서비스 그리드라는 이름으로 유틸리티 방식의 온디맨드 컴퓨팅 모델의 기초로써 본격적으로 적용되게 될 것이다.

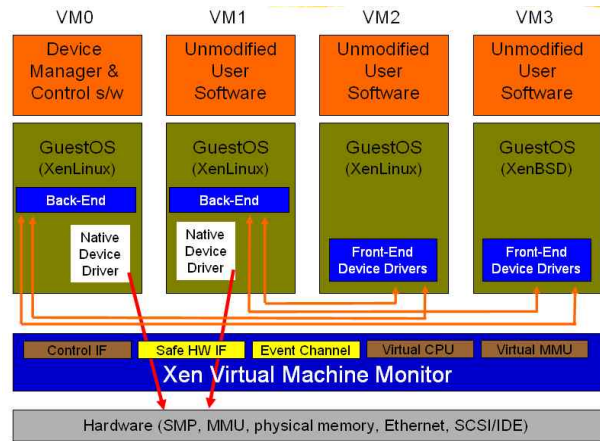
III. 가상화 소프트웨어

이 장에서는 주로 서버 가상화에 해당하는 가상화 소프트웨어에 대해 소개하고자 한다. 공개 소스 기반 가상화 기술에는 상용 제품으로 많이 알려진 VMWare 를 대체 할 수 있는 가상화 소프트웨어에 대해 소개하고, 상업용 가상화 기술에는 대표 기업들의 대표 브랜드 가상화 기술에 대해 소개하였다.

(가) 공개 소스 기반 가상화 기술

(1) Xen

XenSource 사에서 펀드를 받아 영국의 캠브리지 대학에서 개발한 Xen 은 운영체제 가상화이며 공개소스 기반의 가상머신이다[10][11]. VirtualPC, VMWare 가 가상머신 상에서 다수의 운영체제들이 수정하지 않고도 작동될 수 있는 전가상화(Full-virtualization)인 반면 Xen 은 특정 하드웨어 구조에 맞게 운영체제를 수정해야 하는 반가상화(Para-virtualization)란 기술을 사용하여 단일 컴퓨터에서 동시에 여러 개의 운영체제를 사용할 수가 있다. Xen 2.0 의 기능에서 VM relocation 기능은 하나의 클러스터 내에서 동작중인 VM 들의 풀을 관리할 수 있고 부하 분산 기능을 가진다. 2005 년 3 분기에 계획하고 있는 Xen 3.0 은 SMP 운영체제 및 x86/64 구조를 지원하고 인텔의 벤더풀(VT-x) 확장을 지원하며 운영체제를 수정하지 않고도 사용할 수 있다. 이 경우 다소 성능은 떨어질 수 있으나 공개되지 않는 윈도우 운영체제 조차도 사용할 수 있음을 시사한다. 또한 클러스터 관리 도구 기능을 확장하고 새로운 GUI 관리 도구를 가지며 x86 프로세서뿐만 아니라 다른 플랫폼에 적용될 수 있도록 하드웨어 호환성이 향상된다. 이에 따라 수세리눅스 및 레드햇 등에서도 Xen 을 서버 버전에 포함할 계획을 가지고 진행하고 있다. ((그림 11)은 현재 발표된 Xen 2.0 의 구조이다. Xen 은 현재 썬 마이크로시스템즈, HP, 노벨, 레드햇, 인텔, AMD, IBM 등의 지지를 받고 있으며 이들은 프로그래밍 지원이나 소프트웨어 제공 등의 방법으로 Xen 을 지원하고 있다.



(그림 11) Xen 2.0 Architecture

(2) Linux VServer

독립적인 가상 서버 (VPS: Virtual Private Server)를 구축할 수 있도록 하고 각 가상 서버는 리눅스 서버와 같은 대부분 동일한 운영 환경에서 자체 사용자 계정을 가지는 데이터베이스 및 루트 패스워드를 가진다[12].

(3) Cooperative Linux (coLinux)

일본 AIST 연구소에서 지원받아 진행중인 coLinux 는 윈도우상에서 리눅스를 수행할 수 있도록 하는 공개 소스 소프트웨어로서 VMWare 와 같은 에뮬레이터를 사용하지 않고도 윈도우에서 리눅스가 하나의 프로세스인 애플리케이션 형태로 동작할 수 있도록 이식된다[13].

(4) QEMU

QEMU 는 프랑스 Fabrice Bellard 의 공개 소스 프로세서 에뮬레이터로 리눅스에서 부팅한 채로 윈도우를 포함하는 다른 운영체제를 에뮬레이트 할 수 있다[14]. QEMU 는 두 개의 동작 모드를 지원하는 데 하나는 전체 시스템 에뮬레이션(Full system emulation) 모드로 한 프로세서와 다양한 주변장치를 포함하여 전체 시스템을 에뮬레이션 함으로써 컴퓨터를 재 부팅하지 않고도 다른 운영체제를 사용할 수 있다. 다른 모드는 사용자 모드 에뮬레이션(User mode emulation)으로 리눅스 호스트상에서 한 CPU 에서 다른 CPU 를 컴파일 할 수 있도록 에뮬레이트한다. 최적화 및 동적 변환기를 사용하여 에뮬레이션 속도 면에서 빠르다는 것을 장점으로 한다. 하지만 이도 상용화 제품에 비해서는 속도가 느리다.

(5) Bochs

VMWare 및 VirtualPC 와 같은 개념으로 동작을 하며 x86 에뮬레이터이다[15]. 실행할 수 있는 플랫폼은 IBM PC, Windows, Linux 와 Apple Macintosh MacOS 에서 실행 가능하며 IBM PC 와 Mac 에서도 똑 같은 x86 IBM PC 를 에뮬레이트 해서 가상의 PC 를 만들어 낼 수 있다. 즉 윈도우나 리눅스 등의 x86 에서 동작되는 운영체제를 설치하여 사용할 수 있다.

(6) LTSP

LTSP 는 Linux Terminal Server Project 로 리눅스 서버에 썬 클라이언트 터미널로 접속할 수 있도록 한다[16]. 애플리케이션은 썬 클라이언트형태에서 전형적으로 서버에서 수행되며 디스크 없는 워크스테이션을 구현할 수 있다.

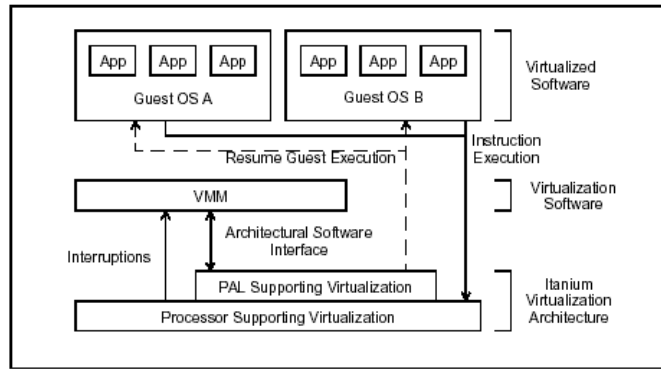
(나) 상업용 가상화 기술

(1) EMC VMware

VMware 는 소프트웨어 방식으로 하드웨어를 에뮬레이션 하는 가상 머신 소프트웨어이다 [17]. 따라서 속도가 느리고 높은 하드웨어 사양을 단점을 가지고 있으나 오랜 개발 과정을 거치면서 최적화가 상당히 많이 진행되고 하드웨어의 성능이 향상됨에 따라 최근의 VMWare 는 속도도 빠르고 안정성이 높다. 현재 최대 2 개의 물리적 프로세서를 사용하는 가상 머신을 지원한다. 또한 2005 년 말까지 4 프로세서 지원이 가능하도록 개발 중에 있다.

(2) Intel Vanderpool

코드네임 Vanderpool 이라는 이 기술은 ((그림 12)와 같이 하나의 컴퓨터 상에서 다수의 운영체제를 운영하게 하고, 동일 기기에서 독립적인 파티션에 애플리케이션을 별도로 운영하게 하는 기술이다[18]. 인텔은 밴더풀 기술 보급과 관련하여 소프트웨어 개발을 추진하기 위해서 밴더풀 기술 외부 아키텍처 스펙을 공개하였다[19]. 이는 복수의 운영체제에서 개별 애플리케이션을 구동 시에 하드웨어 자원들을 복수의 운영체제에 선택적으로 할당할 수 있다. 인텔은 아이테니엄 기반 플랫폼에 밴더풀 기술을 선보일 예정으로 올해 데스크톱 프로세서와 칩셋 제품에도 해당 기술을 제공할 계획이라고 밝혔다.



<참조>: Intel 2005.

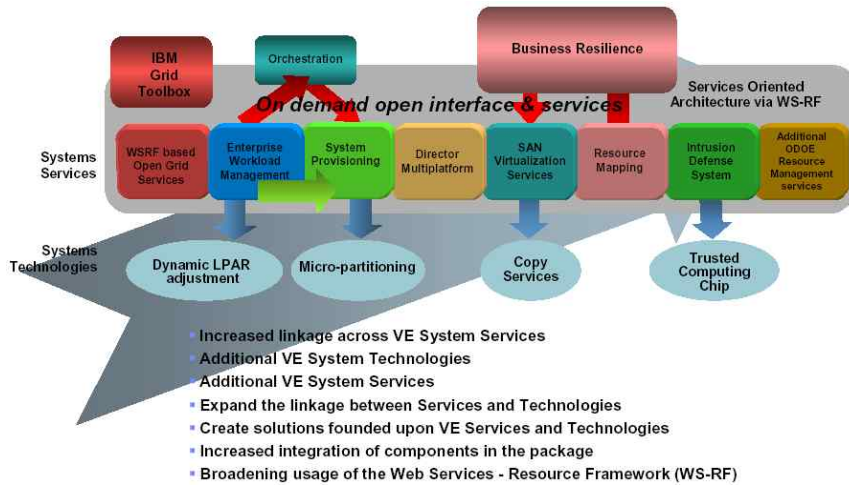
(그림 12) Intel Virtual Server

(3) Microsoft Virtual Server

마이크로소프트는 Connectix 로부터 Virtual Server 를 인수하여 마이크로소프트의 서버환경에 맞게 향상시켜 윈도우 서버 환경에서 여러 개의 다른 플랫폼의 운영체제를 동시에 운용할 수 있도록 하고 있다[20]. 마이크로소프트는 이를 사용하여 애플리케이션의 플랫폼 별 적용, 서버 간의 통합, 또한 소프트웨어 개발 및 테스트 환경의 자동화 등을 간편하게 적용하여 관리 및 하드웨어 자원 이용 측면에서 효율적으로 해결할 수 있다고 한다. 자체 하이퍼바이저를 개발 중에 있으며 이를 포함한 버전은 2007년 출시할 예정으로 있다.

(4) IBM Virtualization Engine

IBM의 가상화 엔진은 자원, 서버, 스토리지, 네트워크의 통합 가상화 솔루션을 지원한다 [21]. 이는 IBM의 가상화 기술인 마이크로 파티셔닝 기술, 티볼리의 프로비저닝과 관리 툴, 웹 스피어, OGSA 기반의 그리드 가용성 등에서 파생된 기술을 표준화 한 것이다. ((그림 13)은 가상화 엔진을 실현하기 위한 IBM의 소프트웨어 및 기술들을 보여준다. 가상화를 하기 위한 기본 기술인 Hypervisor, VLAN, Virtual I/O 를 기반으로 시스템을 가상화하고, IBM Grid Toolbox, Tivoli Provisioning Manager, IBM Enterprise Workload Manager, IBM Director Multiplatform, VE Console 로 서버의 가상화를 가능하게 하며 IBM TotalStorage SAN Volume Controller 및 San File System, Productivity Center 로 스토리지 가상화가 가능하다. IBM은 가상화 엔진이 제공하는 가치로 가상화를 통한 온 디맨드 비즈니스 구현을 실현 할 수 있다고 강조한다.

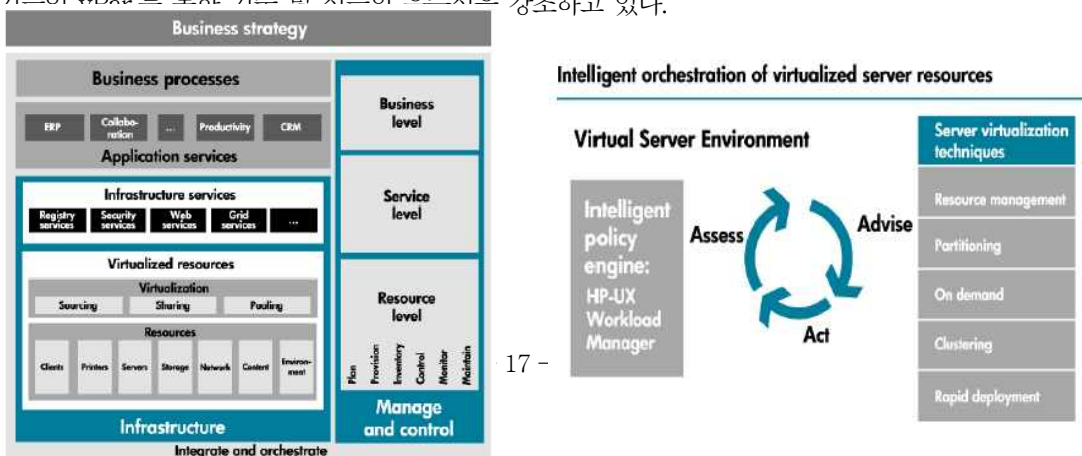


<참조>: IBM 2004.

(그림 13) IBM Virtualization Engine

(5) HP Virtual Server Environment (VSE)

HP VSE 를 적용할 수 있는 솔루션에는 가용성, 물리적으로 서버를 분리하는 파티셔닝 부분 및 통합 관리 기능으로 볼 수 있다[22]. 솔루션들을 살펴보면 가용성 부분은 여러 서버를 하나의 그룹으로 묶어서 시스템 가용성을 높여주는 Serviceguard Clustering 기능, 파티셔닝을 지원하는 HP Virtual Partitioning 인 vPars 와 Hard Partitioning 인 nPar, 서버에서의 프로세스 자원관리 툴 등 애플리케이션의 운영을 최적화하는 PRM(Process Resource Manager)과 pSets(processor sets), 고객의 시스템 사용 정도에 따른 서비스를 차별화하여 온 디맨드를 지원하는 iCAP(Instant Capacity) 등이 있다. ((그림 14)와 같은 기본 솔루션들과 워크로드 매니저의 통합으로, 클러스터링으로 구성된 서버간 또는 파티셔닝으로 분할된 서버간에 자원이 실시간으로 재분배되어 전반적인 서비스 레벨을 만족시킬 수 있을 것이라는 전망이다. HP 는 파티셔닝 기능이 vPar 를 통한 가상 머신 서비스의 강조하고 있다.



<참조>: HP 2005.

(그림 14) HP Virtual Server Environment

IV. 결론

최근에 서버기반 컴퓨팅 혹은 썬 클라이언트 컴퓨팅이 다시 거론되고 있다. 서버 기반 컴퓨팅이란 서버에 설치된 데이터 및 프로그램을 인터넷 연결을 통해 사용하는 컴퓨팅이다. 이는 저장장치가 없는 썬클라이언트 혹은 접속 가능한 단말을 통해 서버를 접속하여 서버 상에서 모든 작업을 할 수 가 있다. 이는 1997 년에 등장하여 열악한 네트워크 인프라 등의 이유로 사그라진 네트워크 컴퓨팅 모델과 일맥상통하다. 최근 국내 삼성전자의 일부 부처의 컴퓨팅 환경을 네트워크 컴퓨팅 환경으로 교체한다고 발표한 적이 있다. 이와 같이 SBC 환경이 다시 주목을 받는 가장 큰 이유는 보안의 문제라고 대두되고 있으며 SBC 컴퓨팅은 이러한 문제를 서버에서 통합 관리함으로써 해결하고 있다. 썬 클라이언트의 경우에는 최소한의 CPU 와 메모리 장치를 가지며 대부분 자원을 서버시스템을 활용하므로 이 경우 가상화된 시스템 환경은 사용자에게 일관성 있고 투명한 환경을 제공할 수 있다.

가상화는 사용자에게 컴퓨팅 자원을 사용함에 있어 단순하고 일관성 있으며 편리한 논리 구조를 갖게 하여 비용 및 인프라 관리를 향상시킨다. 궁극적으로 이러한 가상화 기술은 자율 컴퓨팅, 그리드 컴퓨팅 및 고정적인 컴퓨팅 비용을 가변적으로 바꿀 수 있는 유틸리티 컴퓨팅 기술에 포함되어 양질의 서비스를 제공할 수 있는 기반 기술로 다시 자리 매김하고 있다. 이러한 가상화 기술을 사용한 서비스는 사용자가 컴퓨터를 의식하지 않고 자원을 사용할 수 있는 환경을 지향하는 유비쿼터스 서비스에서부터 사용만큼 지불한다는 유틸리티 컴퓨팅에 이르기까지 앞으로 활용가치가 높은 서비스로 예측하고 있으나 표준화 측면 등 아직 많은 변수들이 있다. 하지만 IT 가상화 솔루션은 소프트웨어와 하드웨어간의 밀착된 관계를 느슨하게 만들어 IT 의 관리와 운영에 있어 보다 관리하기 쉽고 효율적인 컴퓨팅 인프라스트럭처를 구성할 수 있는 기술임에는 틀림없다.

<참 고 문 헌>

- [1] 김진미, 배승조, 박유현, 김학영. 유비쿼터스 시대의 온디맨드 기술 동향. 주간 기술동향, 통권 1158호, 정보통신연구진흥원, 2004. 8.
- [2] The Future of Server Virtualization, Garther Research, July 2003.
- [3] The Future of the Server, Garther Research, July 2003.

- [4] Paul Mason, Dan Kusnetzky, Server Provisioning, Virtualization, and the On-demand Model of Computing: Addressing Market Confusion, IDC, June 2003.
- [5] 이상호, 2004년도 정보산업부문별 경기전망 및 마케팅 전략: 그리드 컴퓨팅, 한국정보산업연합회, 2003. 12. 13.
- [6] Larry Tabb, Grid Computing in Financial Markets: Moving Beyond Computer-Intensive Applications, Tabb, November 2003.
- [7] <http://www.globus.org/>
- [8] Matt Haynos, Perspectives on grid: Using automation effectively within a grid infrastructure, IBM, 2005.
- [9] J. Craig Lowery, Building the Virtual Data Center, Dell Inc., 2003.
- [10] <http://xen.sf.net/>
- [11] A XenSource White Paper, Xen-the Art of Virtualization, XenSource, Inc., 2005.
- [12] <http://linux-vserver.org/>
- [13] <http://www.colinux.org/>
- [14] <http://fabrice.bellard.free.fr/qemu/>
- [15] <http://bochs.sourceforge.net/>
- [16] <http://www.ltsp.org/>
- [17] <http://www.vmware.com>
- [18] <http://www.intel.com/technology/computing/vptech>
- [19] Vanderpool Technology for the Intel Itanium Architecture(VT-i) Preliminary Specification, Intel, 2005.
- [20] <http://www.microsoft.com/windowsserversystem/virtualserver>
- [21] 윤병훈, IBM Virtualization Engine-Server Technology, IBM, 2004.
- [22] White paper: HP Virtual Server Environment (VSE) for HP-UX-automated, real-time server resource allocation, HP, 2005.
- [23] 진기성, IT 가상화 솔루션의 가치와 미래, 한국 HP, 2004.
- [24] 디지털타임즈, <http://www.dt.co.kr>
- [25] ZDNet Korea, <http://www.zdnet.co.kr>