클라우드에서 다중 자원의 성능을 고려한 가상머신 할당 기법

채송화*, 이화민*
*순천향대학교 컴퓨터학과
e-mail:shchae@sch.ac.kr, leehm@sch.ac.kr

A Virtual Machine Allocation Scheme based on Performance of Multi Resource in Cloud

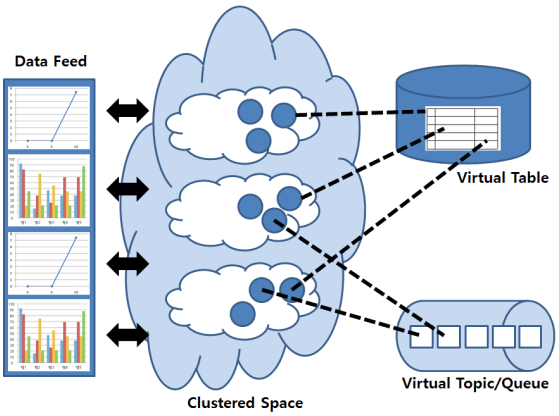
Song-Hwa Chae*, HwaMin Lee*
*Dept of Computer Science&Engineering, Soonchunhyang University

요 약

클라우드 컴퓨팅은 기존의 컴퓨팅 환경에 제약을 받지 않고 가상머신을 활용하여 고사양 컴퓨팅 환경을 제공해준다. 고사양의 컴퓨팅 환경을 제공해주는 서버는 분산 파일 시스템을 통해 자원을 관리하고 매치메이킹, Haizea 등의 가상머신 스케줄러를 통해 노드에 가상머신을 할당한다. 본 논문은 노드가 가지는 CPU, 메모리, 스토리지 자원의 처리 속도를 반영하여 최적의 노드에 가상머신을 할당 할 수 있는 다중 자원의 성능을 고려한 가상머신 할당 기법을 제안한다. 노드 성능 분석 결과를 가진 데이터베이스에서 하드웨어 성능이 지원되는 노드들 중 성능 분석 결과가 좋은 노드에 가상머신을 할당해주는 방법으로 자원의 단편화를 최소화하고 신속한 할당이 가능하다.

1. 서론

클라우드 컴퓨팅은 저사양의 컴퓨팅 환경에서 고사양의 컴퓨팅 자원을 제공 받아 고사양의 서비스를 이용할 수 있도록 해주는 기술로서 IaaS(Infrastructure as a Service), SaaS(Software as a Service), PaaS(Platform as a Service)의 서비스 플랫폼을 제공한다. IaaS는 스토리지와 컴퓨팅 자원을 제공하며, SaaS는 원격 제어 등을 통해 애플리케이션의 실행 환경을 제공하고, PaaS는 애플리케이션 개발 플랫폼을 제공한다[1]. (그림1)은 클라우드 컴퓨팅의 개념도이다.



(그림 1) 클라우드 컴퓨팅 개념도[1]

클라우드 컴퓨팅은 가상화(Virtualization)을 통해 컴퓨팅 자원을 추상화하며 Xen, VM Ware 등을 통해 운영체제 수준의 가상화를 할 뿐 아니라 하드웨어 수준의 가상화도 가능하다[2]. 클라우드 컴퓨팅 환경을 구축해주는 플랫폼 서비스로 유칼립투스(Eucalyptus), 님버스(Nimbus), 오픈니블라(OpenNebula) 등이 있다[3]. 각 플랫폼 서비스에서는 가상화를 통해 할당된 가상머신들의 할당 관리를 위한 스케줄러가 존재하며 대표적으로 유칼립투스의 라운드 로빈(Round-Robin)[4]과 오픈니블라의 매치메이킹(MatchMaking)[5] 등이 있다.

라운드로빈은 자원 사용을 요구하는 노드들이 번갈아가며 일정 시간동안 자원을 사용하는 스케줄링이며 모든 노드에게 공정한 할당이 가능하지만 우선순위 부여가 불가능하여 사용자의 지불 비용에 따른 서비스 제공에 어려움이 있다. 매치메이킹은 각 노드의 CPU 사용 여부와 노드에 가상머신을 할당 후 남은 양을 이용한 가상머신 할당 기법으로 노드의 부하가 발생할 수 있다.

기존의 가상머신 할당을 위한 스케줄링 기법은 노드들의 상태에 따라 가상머신을 할당함으로서 비교적 공정한 할당이 가능하며 이로 인해 부하를 최소화할 수 있다. 그러나 우선순위에 대한 조정이 불가능하기 때문에 사용자 요구 사항에 적합한 성능을 가지는 노드 선택이 불가능하며 단편화로 인한 적재 공간 부족 등의 문제가 발생할 수 있다.

본 논문에서는 노드가 가지는 CPU, 메모리, 스토리지 등 성능을 반영하여 최적의 노드에 가상머신을 할당 할 수 있는 다중 자원의 성능을 고려한 가상머신 할당 기법

을 제안한다. 2장에서는 관련 연구를 통해 기존에 가상머신을 할당을 위한 연구들을 살펴보고 3장에서는 본 논문에서 노드의 성능을 분석하는 방법을 설명하고 4장에서는 노드 성능 분석이 적용된 다중 자원의 성능을 고려한 가상머신 할당 기법을 소개하며 5장에서 결론을 맺는다.

2. 관련 연구

가상머신에 지역 컨트롤러와 전역 컨트롤러라는 자원 관리 컨트롤러를 두 단계로 두어 가상머신을 동적으로 할당하기 위한 연구[6, 7]에서는 지역 컨트롤러가 각각 가상머신의 자원 사용량과 응용 프로그램의 자원 요구량을 파악하고 전역 컨트롤러는 파악된 자원 상황을 바탕으로 요청에 응한다.

가상머신을 동적으로 할당하기 위한 방법으로 전자 공학을 이용한 방법[6]과 담금질 기법을 이용하여 자원이 요구하는 양을 파악하는 방법[7]들이 제시되었다. [6]와 [7]에서는 환경에 제약받지 않는 지역 컨트롤러를 다양하게 구현할 수 있다는 장점을 가지지만 CPU 스케줄링을 고려하지 않아 지연을 방지하거나 해결할 수 있는 방법이 모색되지 않았다는 단점이 있다.

[8]은 Xen의 SEDF(Simple Early Deadline First) 스케줄러를 수정한 SEDF CPU 스케줄러를 제안하였으며 가상머신의 처리가 끝난 CPU를 idle 상태로 보내는 것이 아니라 다른 CPU에 대기 중인 가상머신을 가져와 처리함으로서 지연을 최소화하고자 하였다.

[9]은 시스템의 CPU 사용률을 데이터베이스에 기록하고 최적의 노드를 검색하는 레이어와 CPU 사용률을 관리하는 레이어로 구분하고 CPU의 사용 여부를 기록하여 과부화를 최소화하고자 하는 기법을 제시하였다. 이 방법은 자주 사용되지 않는 CPU를 갖는 노드에 가상머신을 할당함으로서 부하 분산을 가능토록 하였지만, CPU 등 다중 자원의 성능에 대한 우선순위가 없어 가상머신을 처리하는 속도가 사용자마다 다르며 이에 대한 처리가 어렵다.

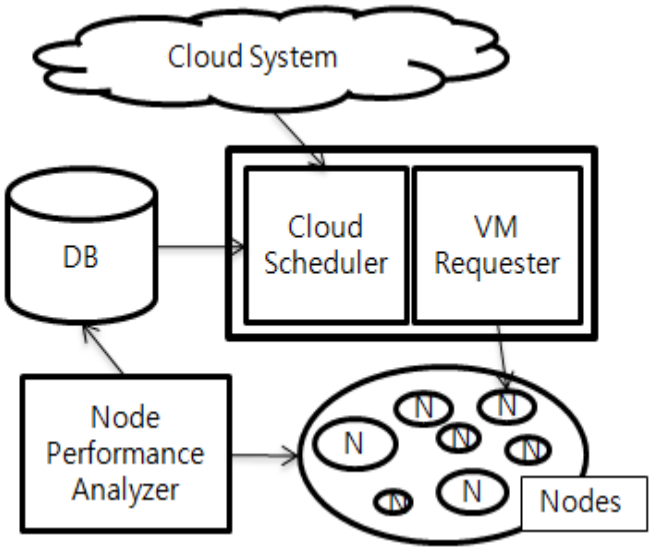
3. 노드의 성능 분석

본 논문에서 제안하는 다중 자원 성능을 고려한 가상머신 할당 기법은 시스템 정보 초기화 시 연산 처리 능력을 기반으로 노드들의 성능을 평가한다. 연산 처리 능력은 CPU가 문제 해결을 위해 소요되는 시간을 기준으로 처리 속도가 빠른 노드를 연산 처리 능력이 뛰어나다 평가하였다. 성능평가를 위해 본 논문에서는 LU Decomposition을 이용하여 행렬의 역행렬을 구하는 것으로 연산 처리를 위해 소요되는 시간을 계산하여 연산 처리 능력을 평가하였다.

LU Decomposition을 이용하여 역행렬을 구하는 방법은 행렬 A를 하 삼각행렬 L과 상 삼각 행렬 U로 분해하여 U는 Gaussian Elimination을 이용하여 피봇 성분과 행 연산이 될 성분과의 배수, 승수를 구해 초기값이 대각 요소가 1인 L 행렬에 승수를 곱한 L의 결과에 L * D = B(B는 단위 행렬의 i번째 요소) 방정식을 통해 D를 구하고 구해진 D를 이용하여 U * X = D를 이용하여 역행렬의 i열을 구한 후 모든 열의 합을 구하면 된다.

4. 다중 자원의 성능을 고려한 가상머신 할당 기법

본 논문에 제안하는 다중 자원 성능을 고려한 가상머신 할당 기법은 노드들의 성능 분석이 기록된 데이터베이스의 내용을 기반으로 가상머신 생성 요청에 신속하게 반응하고 수행할 수 있도록 한다. 본 논문에서 제안하는 시스템 구조는 (그림 2)와 같다.



(그림 2) 다중 자원 성능을 고려한 가상머신 할당 시스템 구조

각 구성 요소가 하는 일은 다음과 같다.

· Node Performance Analysis Result DB : 노드들의 성능을 분석한 결과를 가지는 데이터베이스로, 시스템에서 이 데이터베이스의 데이터를 이용하여 노드 리스트를 추출한다.

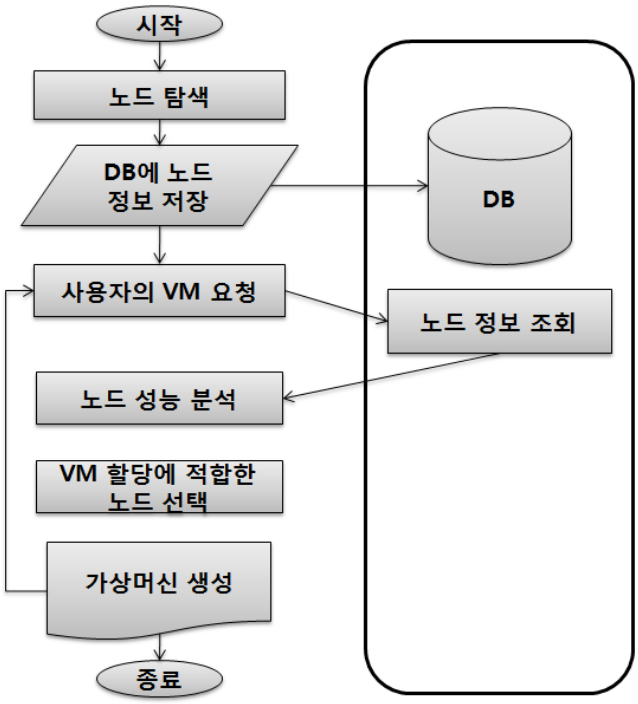
· Node Performance Analyzer : 노드들의 정보를 가져와 성능 분석을 한 결과를 데이터베이스에 저장한다.

· Cloud Scheduler : 클라우드 시스템의 요청에 응답하는 것으로 데이터베이스에서 노드 성능 정보를 가져와 Extraction Node List를 작성하며, 리스트 중 가장 적합한 노드를 선택하는 역할을 한다.

· VM Requester : Cloud Scheduler에서 선택된 적합한 노드에 가상머신 할당을 요청한다.

시스템이 시작되면 클라우드가 가지는 전체 노드의 하드웨어 스펙과 성능을 평가하여 데이터베이스에 저장하고 가상머신의 할당이 요청되면 시스템은 노드 성능 분석 결과 데이터베이스에 등록된 전체 노드들 중 가상머신이 요

구하는 CPU, 메모리, 스토리지가 수용 가능한 노드들의 리스트를 요청한다. 데이터베이스에서 반환받은 노드 리스트 중 요청된 가상머신 할당을 위한 조건에 성립하는 노드들을 선정한다. (그림 3)은 다중 자원 성능을 고려한 가상머신 할당 기법의 흐름을 나타낸다.



(그림 3) 다중 자원 성능을 고려한 가상머신 할당 흐름도

다중 자원 성능을 고려한 가상머신 할당 기법은 크게 3가지 단계로 분류된다. 첫 번째 단계로 Node Performance Analysis로 노드들의 성능을 평가하는데 이는 각 노드들에 LU Decomposition을 이용한 역행렬을 구하는 방법을 통해 연산 처리 성능을 평가하게 된다. 두 번째 단계는 Node List Request로 성능이 평가된 노드들의 정보를 가지는 데이터베이스에서 CPU, 메모리, 스토리지의 성능이 요구 자원을 충족하는 노드들을 리스트화하여 해당 리스트를 성능 평과 결과를 기준으로 정렬하여 반환을 해준다. 세 번째 단계 Node Select는 반환받은 노드 리스트들 중 가장 처리속도가 빠른 노드를 선택하여 할당 예정 노드로 선정하고 선정된 노드에 가상머신의 할당을 요청한다. (그림 4)는 다중 자원 성능을 고려한 가상머신 할당 알고리즘이다.

```
[Node Information Search]
for (Node Performance Analysis Result 전체)
    if cpu, memory, storage의 성능이 요구 자원을 충족하면 than
    ExtractionNodeList.add();
ExtractionNodeList.sort();
return ExtractionNodeList
```

```
[Node Performance Analysis]
for ( VM 할당이 가능한 노드들 )
{
    노드에서 LU Decomposition로 연산처리;
    연산처리 소요 시간 측정;
    DB.add(성능평가결과);
}

[Node Select]
for (ExtractionNodeList 전체)
    if Node에 할당중인 VM이 없다면 than
        if 할당 예정인 노드보다 가장 처리속도가 빠르다면 than
            할당 예정 노드로 선정;
할당 예정 노드에 VM 할당
```

(그림 4) 다중 자원 성능을 고려한 가상머신 할당 알고리즘

5. 결론 및 향후 연구과제

클라우드 컴퓨팅은 컴퓨팅 환경의 새 패러다임으로 사용하는 기기에 제약 없이 고사양의 컴퓨팅 환경에서 작업을 할 수 있도록 한다. 본 논문에서는 클라우드 컴퓨팅 환경에서 노드의 자원이 통일되지 않을 때 각 노드들의 성능을 비교하여 가상머신 할당 요청을 신속하게 처리할 수 있는 할당 기법을 제안하였다.

향후 연구과제로 노드의 네트워크 상태가 불안정할 경우 안정적인 클라우드 서비스 제공을 위한 방법과 하며 자원의 성능을 정확하게 분석할 수 있는 알고리즘에 대한 연구가 필요하다. 또한 QoS(Quality of Service)를 반영하여 사용자의 요구 사항을 충족할 수 있는 방법에 대한 연구가 필요하다.

참고문헌

[1] Cloud Computing 시장 동향 및 전망, SW 산업동향 KIPA, 2008.

[2] 김창환, 클라우드 컴퓨팅 기술 시장 동향, 정보통신산업진흥원, 2010.

[3] Sempolinski, P., Thain, D., A Comparison and Critique of Eucalyptus, OpenNebula and Nimbus, Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on, pp.417-426, 2010

[4] Johannes Fürnkranz, Round robin classification, The Journal of Machine Learning Research, vol.2, pp.721-747, 2002

[5] Raman, R., Livny, M., Solomon, M., Matchmaking: Distributed Resource Management forHigh Throughput Computing, High Performance Distributed Computing 1998, The Seventh International Symposium, pp.140-146, Chicago, 1998.

[6] Xu, J., Zhao, M., Fortes, J., Carpenter, R., and Yousif, M., Autonomic Resource Management in Virtualized Data Centers Using Fuzzy Logic-Based Approaches, P_roceedings of Cluster Computing, vol.11 no.3, pp.213-227, 2008.
[7] X. Zhu, D. Young, B. J. Watson, Z. Wang, J, Rolia, S. Sighal, B. McKee, C. Hyser, D. Gmach, R. Gardner, T. Christian, and L. Cherkasova, 1000 Islands: Integrated Capacity and Workload Management for the Next Generation Data Center, Proceedings of International Conference on Auto-nomic Computing (ICAC '08), 2008
[8] 양은지, 최현식, 한세영, 박성용, Xen 환경에서 스케줄링 지연을 고려한 가상머신 우선순위 할당 기법, 정보과학회논문지 : 시스템 및 이론 제 27권, pp.195-355, 2010.
[9] 클라우드 컴퓨팅에서 CPU 사용률을 고려한 가상머신 할당 기법, 산업진흥원, 2010.