

대규모 클러스터 서버의 성능 모델링 및 예측 방법론

(A Methodology for Performance Modeling and Prediction of Large-Scale Cluster Servers)

장혜천[†] 진현욱^{**}
(Hye-Churn Jang) (Hyun-Wook Jin)

김학영^{***}
(Hag-Young Kim)

키워드 : 클러스터, 성능 모델링, 성능 예측, 성능 측정

Abstract Clusters can provide scalable and flexible architectures for parallel computing servers and data centers. Their performance prediction has been a very challenging issue. Existing performance measurement methodologies are able to measure the performance of servers already constructed. Thus they cannot provide a way to predict the overall system performance in advance when designing the system at the initial phase or adding more nodes for more capacity. Therefore, the performance modeling and prediction methodology for large-scale clusters is highly required. In this paper, we suggest a methodology to predict the performance of large-scale clusters, which consists of measurement, modeling and prediction steps. We apply the methodology to a real cluster server and show its usefulness.

Key words : Cluster, Performance modeling, Performance prediction, Performance measurement

요약 클러스터는 병렬 컴퓨팅 및 데이터 센터에 적합한 구조를 제시하지만 설계 및 확장을 할 때 성능에 대한 예측이 쉽지 않다. 또한 기존의 클러스터 성능 분석은 이미 구성된 시스템만을 그 대상으로 한다는 문제점을 가지고 있으며 클러스터의 확장 및 대용량 클러스터에 대한 성능 예측을 지원하지 못한다. 그러므로 기존에 대규모 클러스터를 평가하던 방법들과는 다른, 시스템 구성 전 대규모 클러스터를 위한 모델링 및 예측 방법을 필요로 한다. 이러한 작업은 클러스터의 구조적 특성이 잘 반영되어야 하며 실제 시스템 적용 시 나타나는 문제에 관해서도 분석이 쉽게 되어야 한다. 본 논문에서는 대규모 클러스터의 성능 모델링을 위한 방법론을 제시하고 실제 시스템에서 수행한 측정 및 예측 결과로 방법론의 유용성을 보인다.

1. 서론

클러스터는 뛰어난 확장성과 가격 대비 성능의 우수성으로 인해 대규모 서버를 위해서 많이 사용되고 있는 서버 구조이며 병렬 컴퓨팅[1]과 데이터센터[2]를 위하여 널리 사용되고 있다. 이러한 클러스터는 대규모 사용자에 대한 지원과 여러 기관의 웹 기반 서비스들을 제공해주는 것을 목표로 하기 때문에 최근 1천대 이상의 대형 클러스터 구성을 시도하고 있다[3,4]. 하지만 이와 같이 대규모 클러스터를 구성하기 위해서는 비용과 시간이 많이 든다. 따라서 실제 시스템을 구성하기 전에 소규모의 시스템으로 대규모 시스템 구성 시 목표 성능치를 얻을 수 있는 지에 대한 검증 단계를 거칠 수 있다면 이러한 비용과 시간을 상당히 줄일 수 있다. 하지만 현재까지는 이미 구성된 클러스터의 일부 특성만을 거시적으로 고려하여 조율하는 것이 일반적이다. 그러나 이와 같은 방법은 목표치를 설정하고 그 성취 가능성을 예상하기 어려우며 투자될 비용, 인력, 시간에 대한 결정이 힘들어 진다는 문제를 야기한다. 본 논문에서는 이러한 문제를 효과적으로 다루기 위하여 대규모 클러스터 구성을 위한 성능 모델링 및 예측 방법의 필요성을 제시하고 기존의 클러스터 평가 방법이 아닌 성능 모델링과 예측을 위한 방법론을 제시하며 실제 시스템의 성능 측정 결과를 가지고 제안된 방법론의 유용성을 보인다.

본 논문은 다음과 같이 구성되어 있다. 서론에 이어서 2장에서는 기존 클러스터 평가 방법과 문제점에 대하여 살펴본다. 그리고 3장에서는 클러스터 성능 모델링 및 예측을 위한 방법론에 관하여 기술하며, 4장에서는 방법론을 실제 시스템에 적용한 사례 분석 결과를 살펴본다.

· 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원 사업(NIPA-2010-C1090-1031-0003), 교육과학기술부의 세계수준의 연구중심대학 육성사업(WCU-2009-A419-0115), 2009년도 한국전자통신연구원원의 위탁과제 지원(7010-2009-0016)의 연구 결과로 수행되었음
· 이 논문은 2010 한국컴퓨터종합학술대회에서 '대규모 클러스터 서버의 성능 모델링 및 예측 방법론'의 제목으로 발표된 논문을 확장한 것임

[†] 학생회원 : 건국대학교 컴퓨터공학부
comfact@konkuk.ac.kr

^{**} 종신회원 : 건국대학교 컴퓨터공학부 교수
jinh@konkuk.ac.kr
(Corresponding author)

^{***} 종신회원 : 한국전자통신연구원 컴퓨터시스템연구부 책임연구원
h0kim@etri.re.kr

논문접수 : 2010년 8월 10일
심사완료 : 2010년 10월 5일

Copyright©2010 한국정보과학회 : 개인 목적이거나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨팅의 실제 및 레터 제16권 제11호(2010.11)

마지막으로 5장에서 본 논문의 결론을 내린다.

2. 관련 연구

현재 클러스터 시스템의 성능을 측정하기 위한 도구들은 많이 개발되어 있다. 우선 병렬 컴퓨팅 클러스터의 경우는 Linkpack[5]을 사용하여 성능을 측정한다. Linkpack은 선형 대수 계산을 위한 라이브러리를 제공하고 시스템의 실수 연산 능력을 출력하여준다.

클러스터 기반 데이터 센터의 성능은 일반적으로 전통적인 웹 서버 성능 측정 도구인 TPC 벤치마크를 사용한다[6]. TPC 벤치마크는 여러 가지 시스템의 특성을 갖는 워크로드를 생성하고 그에 따른 성능을 단위 시간당 트랜잭션 수로 출력한다. 이 외에도 최근 라이스 대학교에서 개발된 RUBiS(Rice University Bidding System)는 전자 상거래 워크로드를 생성하여 시스템의 성능을 측정한다[7].

지금까지 언급된 측정 도구들은 구성되어 있는 시스템에 대하여 성취 가능한 객관적인 성능을 제시하여 준다. 하지만 이들은 구성되어 있는 시스템을 기반으로 더욱 큰 시스템으로 확장했을 때의 성능 또는 목표치에 대한 가능성 지표는 제공해 주지 못한다. 따라서 이들 벤치마크들은 소규모의 시스템으로 대규모 시스템 구성 시 기대되는 목표치 성능 성취 가능성을 도출할 수 있는 측정 도구로는 부족하다.

3. 측정 방법론

제안되는 방법론의 기본 바탕은 컴퓨팅 노드 및 그룹이 증가하면 전체 시스템 자원 활용이 증가한다는 것에서 시작된다. 이것은 자원 활용의 변화를 측정해야 함을 의미하며, 측정된 내용은 성능 모델을 추출하기 위한 자료로 사용된다. 추출된 성능 모델은 구성되지 않은 클러스터의 성능 예측을 위한 도구로서 사용된다.

일반적으로 클러스터 시스템은 계산을 수행하는 컴퓨팅 노드와 그룹을 관리하는 마스터 노드로 구성되어 있으며 컴퓨팅 노드와 마스터 노드를 묶어서 그룹으로 구성할 수 있다. 이 때 그룹은 다수의 그룹이 모여 상위 그룹을 구성하여 상위 마스터 노드의 관리를 받을 수 있다.

그림 1은 앞서 설명한 일반적인 클러스터 구조의 예를 보여주고 있다. 그림의 구조는 레벨 1의 마스터 노드가 레벨 2의 그룹들을 관리하고 있으며, 레벨 2의 마스터 노드는 그룹 내의 컴퓨팅 노드를 관리하고 있다. 본 논문에서는 이와 같이 두 레벨의 구조를 언급하나 그 이상의 다중레벨도 쉽게 확장할 수 있다. 이러한 구조는 하위 그룹의 수와 컴퓨팅 노드 수의 변화가 상위 그룹에 영향을 줄 수가 있다. 따라서 클러스터의 성능 변

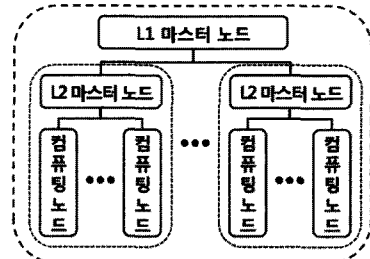


그림 1 일반적인 클러스터의 구조

화를 관찰하기 위해서는 그룹 내의 변화 요인이 다양으로 발생하면 안 된다. 예를 들어 그룹 내의 컴퓨팅 노드와 그룹의 수가 동시에 증가하는 동안에 발생한 성능 변화가 컴퓨팅 노드 수의 변화로 발생한 것인지 그룹 수의 변화로 발생한 것인지 분간할 수 없게 된다. 따라서 클러스터의 성능 측정은 컴퓨팅 노드가 증가할 경우와 그룹이 증가 하는 두 가지 경우로 나눠 질 수 있다. 이 때 측정하는 시스템 환경 및 측정을 위한 도구 설정은 모델화에서 예측을 어렵게 만들지 않게 하기 위한 중요한 요소이다. 이러한 요소는 고려할 것이 많이 있는데, 우선 응용수준 벤치마크에 대한 튜닝이 필요하다. 응용수준 벤치마크를 위한 클라이언트가 병목지점이 되거나 충분한 부하를 생성하지 못하면 이는 클러스터의 제한적인 확장성으로 잘못 해석될 수 있다. 클러스터 시스템의 튜닝 역시 필요하다. 기본적인 구성에서 이미 마스터 노드들이 병목지점이 된다면 성능 예측단계까지 갈수가 없다. 따라서 실측 단계에서 마스터 노드들이 병목지점이 되지 않을 개수에 대한 고려가 필요하다.

앞서 살펴 본 내용을 기반으로 방법론은 3단계로 구성되며 1단계와 2단계는 클러스터 성능 모델을 추출하기 위한 측정 단계이고 마지막 단계는 추출된 모델을 활용하여 성능 예측을 하는 단계이다. 각 단계에 대한 설명은 본 장의 각 절에서 다룬다.

3.1 컴퓨팅 노드 증가에 따른 성능 측정(1단계)

컴퓨팅 노드 수의 증가에 따른 컴퓨팅 노드와 마스터 노드들의 CPU, 메모리, 네트워크 자원 활용률을 측정한다. 이러한 측정은 컴퓨팅 노드 수의 증가에 따른 응용수준의 성능 향상 정도를 파악하는 것을 목적으로 한다. 또한 컴퓨팅 노드의 증가로 인해서 마스터 노드들에 발생하는 부하의 증가 추이를 분석하기 위한 것이다. 클러스터에서 마스터 노드들은 구조상 병목지점이 될 수 있기 때문에 이들의 확장성을 분석하는 것은 중요하다.

3.2 그룹의 증가에 따른 성능 측정(2단계)

그룹 수의 증가에 따른 응용수준 벤치마크 성과와 마스터 노드들의 CPU, 메모리, 네트워크 자원 활용률을 측정한다. 이러한 측정은 그룹 수의 증가에 따른 응용수

준의 성능 향상 정도를 파악하기 위한 것이다. 또한 그룹 수의 증가로 인해서 마스터 노드들에 발생하는 부하의 증가를 분석하기 위한 것이다.

3.3 성능 모델링 및 예측(3단계)

3.1절과 3.2절에 의해서 컴퓨팅 노드와 그룹 수의 증가에 따른 컴퓨팅 노드들과 마스터 노드들의 자원 활용률과 응용수준 벤치마크 성능 특성을 분석할 수 있다. 이와 같이 분석된 특성을 바탕으로 성능 모델을 제안할 수 있다. 성능 모델은 컴퓨팅 노드 수와 그룹의 수를 인자 값으로 받아들인다. 모델의 출력은 컴퓨팅 노드 또는 마스터 노드들의 자원 활용률(CPU, 메모리, 네트워크)과 응용수준 벤치마크의 예측 결과 값이다. 성능 모델은 상수 값을 가지거나 1차 방정식의 형태가 된다. 이것은 성능 증가가 이전 성능 증가율의 연장선상에서 이루지는 것을 의미한다. 하지만 모델링을 통하여 표현하고자 하는 성능 증가 요소가 최대 자원 활용의 한계를 가지고 있을 경우는 성능 모델이 제한적인 형태로 나타날 수 있다. 예를 들어 CPU 사용률의 경우는 최대 100%의 성능 사용이라는 제한을 두고 있다. 이러한 요소는 양적 증가의 개념보다는 비율의 변화의 추이를 살펴야 한다. 결국 최대 자원의 한계를 가지고 있을 경우 1차 방정식의 형태가 아닌 경우 무조건 성능의 병목지점으로 인하여 나타나는 현상으로 취급되어지면 안 된다. 이와 다르게 성능 증가의 제한요소가 없이 1차 방정식이나 상수형 모델링이 아닌 경우는 외부 요인에 의하여 성능 제한을 받은 것이라 볼 수 있기 때문에 성능 모델링을 위한 자료로 적절하지 않다.

이와 같이 작성된 모델은 실제 구성 가능한 여러 시스템의 측정 성능과 비교하여 그 오차율을 분석하며 이 과정을 통해서 모델의 정확성을 향상시킬 수 있다. 모델의 정확성 분석이 완료되면 구성되어 있지 않은 가상의 대규모 클러스터 시스템에 대한 인자 값을 입력으로 성능을 예측할 수 있다. 예측된 모델 기반의 성능은 시스템의 확장성 분석과 병목지점 명시에 활용될 수 있을 것으로 기대된다.

4. 사례연구

앞서 설명한 방법론의 유용성을 검증하기 위해서 본 장에서는 실제 시스템에 방법론을 적용해 본다. 본 사례 연구는 실제 시스템에서 자원을 늘려가며 실험되었고 예측된 값을 기반으로 성능 모델을 제시하고 클러스터 서버의 확장성에 대하여 고찰해 본다.

4.1 시스템 구성 및 측정

방법론을 실제 수행하기 위해 표 1과 같은 시스템을 구성하였다. 시스템은 클라이언트 노드가 작업의 요청을 L1 마스터 노드에게 요청하고 라운드 로빈 방식으로 하위 L2 마스터 노드에게 보낸다. L2 마스터 노드는 해당 작업이 처리 가능한 컴퓨팅 노드를 찾아 처리 요청을 전달하여 준다. 컴퓨팅 노드는 요청을 실제로 처리하여 클라이언트 노드에 작업 결과를 알려준다.

클라이언트 노드의 경우 성능측정 결과의 병목지점이 되는 것을 막기 위하여 서버처리 성능의 3배수를 고려하였으며 네트워크의 경우 총 15Gbps의 대역폭 지원이 가능하게 하였다. 측정은 초당 1만번의 정수 계산을 요구하는 웹 응용 수행과 500KByte의 HTML데이터를 요청하는 웹 데이터 요청의 두 가지를 측정 하였다. 해당 요청을 실제 발생시키는 응용수준 벤치마크로는 WebStone[8]을 사용하였다.

그림 2에서 볼 수 있는 바와 같이 웹 응용 수행에 대해서 시스템은 상당히 확장성이 좋은 것으로 관찰되었다. 그룹을 구성하는 L1 마스터, L2 마스터 및 컴퓨팅 노드들의 네트워크 사용률은 RPS가 증가함에 따라 함께 증가하고 있음이 관찰되며 총 15Gbps의 대역폭까지 증가가 이루어질 수 있을 것이라 알 수 있다. CPU 활용률의 경우에서 컴퓨팅 노드는 그 개수에 상관없이 모

표 1 측정 시스템의 노드 구분

종류	개수	역할
L1 마스터 노드	1	1차 처리요청 분배
L2 마스터 노드	1	2차 처리요청 분배
컴퓨팅 노드	20	클라이언트요청 실제 처리
클라이언트 노드	60	처리요청 생성

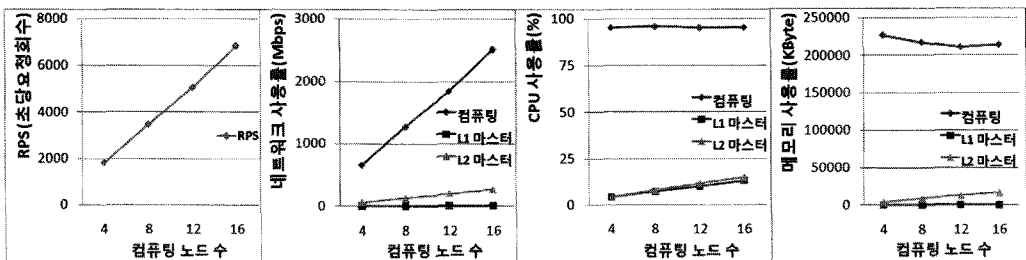


그림 2 웹 응용 수행에 대한 측정 결과

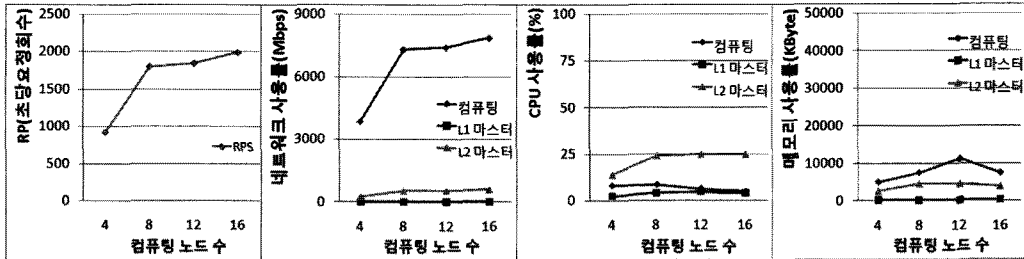


그림 3 웹 데이터 요청에 대한 측정 결과

두 100% 가까이 이르고 있다. 이는 사용된 웹 응용이 계산 집중적이기 때문이다. 반면 L1 마스터와 L2 마스터 노드들은 컴퓨팅 노드의 수가 증가하면 그에 따라 처리할 클라이언트의 요청을 많이 수용하게 되어 CPU 활용률이 점차 증가하고 있음을 알 수 있다. 마지막으로 서버와 L1 마스터 노드의 메모리 활용률은 컴퓨팅 노드 개수에 크게 영향을 받지 않는 반면 L2 마스터 노드는 선형적으로 비례하여 증가하고 있음을 관찰할 수 있다.

그림 3에서 볼 수 있는 바와 같이 웹 데이터 요청은 실험 시스템에서 확장성이 좋지 못한 것으로 관찰되었다. 시스템 그룹을 구성하는 L1 마스터, L2 마스터 및 컴퓨팅 노드들의 네트워크 사용률은 컴퓨팅 노드의 개수가 8개 이상일 경우 큰 변화가 없음을 알 수 있다. 이 경우 이론적으로 15Gbps의 네트워크 환경을 가지지만 TCP/IP 및 HTTP 프로토콜 등의 오버헤드 등에 의해서 약 8Gbps까지 밖에 활용을 못 하는 것으로 보인다 [9]. 그 결과 컴퓨팅 노드의 수가 8개를 넘어가면서부터 네트워크 성능이 병목지점이 되어 RPS도 더 이상 증가하지 못하고 있다. 같은 이유로 인해서 서버의 노드 수가 증가해도 더 많은 클라이언트의 요청을 처리할 수 없기 때문에 CPU와 메모리 자원의 활용률도 모든 노드에서 큰 변화를 보여주지 않고 있다.

4.2 시스템 성능 모델링 및 예측

앞서 측정된 데이터를 기반으로 방법론을 적용하면 해당 측정 실험의 모델링 결과를 얻을 수 있다. 모델링은 단순 선형 회기 분석(Simple Linear Regression Analysis)을 사용하였다. 웹 응용 수행의 경우 16대까지 측정된 자료를 통하여 모델링을 수행한 후 20대의 성능을 예측하여 실측값과 비교하였을 때 약 5% 미만의 오차를 보였다. 표 2에서는 20대 까지 측정된 자료를 가지고 모델링한 결과를 보여주고 있다. 표에서 확인할 수 있듯이 컴퓨팅 노드의 CPU 활용률과 컴퓨팅 및 L1 마스터 노드의 메모리 활용률은 컴퓨팅 노드의 개수에서 영향을 받지 않으므로 상수로 모델링된 것을 볼 수 있다.

표 3에서는 성능 측정 항목 중에 가장 절편이 커서 병목지점에 빠르게 도달할 모델링에 대하여 성능 증가

표 2 웹 응용 수행에 대한 모델링

구분	노드종류	20대 모델링	단위
RPS	-	$399 \times N + 270$	RPS
네트워크	컴퓨팅	$152.26 \times N + 50.82$	Mbps
	L1	$0.59 \times N + 0.82$	
	L2	$15.63 \times N + 0.89$	
CPU	컴퓨팅	95	%
	L1	$0.7 \times N + 2$	
	L2	$0.8 \times N + 1$	
메모리	컴퓨팅	214.25	KB
	L1	0.2	
	L2	$1012.1 \times N + 347$	

표 3 웹 응용 수행의 성능 예측

구분	병목 노드	한계 노드수
네트워크	컴퓨팅	88
CPU	L2	122
메모리	L2	197
RPS	$399 \times 88 + 270 = 35382$	

의 한계치를 표시하였다. 결론적으로 해당 실험환경은 컴퓨팅 노드가 88대가 되었을 때 네트워크가 병목지점이 되어 더 이상 성능 증가가 이루어지지 않는다는 것을 알 수 있다. 이 때 측정된 클러스터 시스템이 해당 웹 응용에 대하여 35382 RPS의 성능을 낼 수 있다는 것을 응용 수준 모델링을 통하여 알 수 있다. 만약 네트워크가 40Gbps로 확장된다면 L2노드의 CPU가 병목지점이 되기 때문에 그룹을 증가시켜야 성능 증가가 이루어 질 수 있다.

웹 데이터 요청의 경우 네트워크가 병목지점이 되어 더 이상 성능 증가가 이루어지지 않았으므로 8대까지의 측정 데이터를 사용하여 네트워크가 40Gbps로 확장되었다고 가정하고 모델링을 한 결과를 표 4에 표시하였다. 이렇게 모델링한 자료를 가지고 웹 응용 수행의 예측과 같은 방법으로 가장 절편이 큰 모델링 자료의 성능 증가의 한계를 살펴보면 표 5와 같다. 웹 데이터 요청의 경우 컴퓨팅 노드가 32대까지 증가하였을 때 L2의 CPU가 병목지점이 되어 더 이상 성능 증가가 이루어지

표 4 웹 데이터 요청에 대한 모델링

구분	노드종류	8대 모델링	단위
RPS	-	$219 \times N + 11$	RPS
네트워크	컴퓨팅	$861.86 \times N + 104.76$	Mbps
	L1	$0.3199 \times N + 0.1333$	
	L2	$69.767 \times N - 8.3231$	
CPU	컴퓨팅	6	%
	L1	$0.43 \times N + 0.2$	
	L2	$2.515 \times N + 1$	
메모리	컴퓨팅	7300	KB
	L1	273	
	L2	$497 \times N + 616$	

표 5 웹 데이터 요청의 성능 예측

구분	병목 노드	한계 노드수
네트워크	컴퓨팅	48
CPU	L2	32
메모리	L2	4022
RPS		$219 \times 32 + 11 = 7019$

지 않는다. 이를 해결하기 위하여 그룹의 수를 증가시켰을 경우 다시 네트워크가 한계가 되어 총 48대의 노드까지 확장할 수 있다.

5. 결론 및 향후 계획

본 논문에서 제안된 대규모 클러스터 시스템 평가 방법론은 소규모의 실제 시스템을 가지고 아직 구성되지 않은 대규모의 클러스터 시스템이 보일 수 있는 성능 요소를 유추 할 수 있게 하여 비용, 인력, 시간의 절약은 물론이고 시스템 소프트웨어의 개발과 하드웨어 구성에 중요한 지침을 제공 할 수 있다. 본 논문은 제안된 시스템 평가 방법론을 기반으로 응용 수준 벤치마크를 수행하여 실험적으로 구성된 클러스터 시스템에서 활용 예시를 보여주었다. 결과적으로 본 논문에서 제안한 평가 방법론은 향후 그 요구가 점차 급격하게 증가할 대규모 클러스터 시스템 구성에 꼭 필요한 기술로서 관련 산업 및 연구 분야에 크게 기여할 수 있을 것이다.

향후 계획으로는 동일하지 않은 노드로 구성된 클러스터 시스템에 방법론을 확장하여 적용할 것이다. 또한 논문에서 제안한 방법론을 쉽게 측정하고 모델링 및 수치 분석을 자동화할 수 있는 도구를 제작할 계획이다. 이 또한 클러스터 분야에서 시스템 구조 설계의 비용 및 자원을 절약할 수 있는 도구로서 사용될 것이라 기대된다.

참 고 문 헌

[1] T. E. Anderson, D. E. Culler, D. A. Patterson, and

the NOW Team, "A Case for Network soft Workstation: NOW," *IEEE Micro*, February 1995.

[2] H. Shah, D. Minturn, A. Foong, G. McAlpine, R. Madukkarumukumana, and G. Regnier, "CSP: A Novel System Architecture for Scalable Internet and Communication Services," *Proceedings of USENIX Symposium on Internet Technologies and Systems*, 2001.

[3] Top500 Supercomputer Sites, <http://www.top500.org/>.

[4] NASA Advanced Supercomputing Division, Columbia Supercomputer, <http://www.nas.nasa.gov/>.

[5] Innovative Computing Laboratory (ICL), "HPC Challenge Benchmark," <http://icl.cs.utk.edu/hpcc>.

[6] Transaction Processing Performance Council, TPC Benchmarks, <http://www.tpc.org/>.

[7] Cristiana Amza, Emmanuel Cecchet, Anupam Chanda, Alan Cox, Sameh Elnikety, Romer Gil, Julie Marguerite, Karthick Rajamani and Willy Zwae-nepoel, "Specification and Implementation of Dynamic Web Site Benchmarks," WWC-5, November 2002.

[8] The Benchmark for Web Servers, Mind Craft. <http://www.mindcraft.com/webstone/>

[9] H.-W. Jin, Y.-J. Yun, and H.-C. Jang, "TCP/IP Performance Near I/O Bus Bandwidth on Multi-Core Systems: 10-Gigabit Ethernet vs. Multi-Port Gigabit Ethernet," *Proceedings of International Workshop on Parallel Programming Models and Systems Software for High-End Computing (P2S2)*, pp.87-94, September 2008.