

# 서버 클러스터 환경에서 에너지 절약을 위한 서버 전원 모드 제어에서 동적 종료

함치환, 김호연, 김동준, \*곽후근, \*권희웅, \*김영종, 정규식  
송실대학교 정보통신전자공학부, \*팜킨네트웍스(주)  
e-mail : hch@q.ssu.ac.kr

## Dynamic Shutdown at Server Power Mode Control for Saving Energy in a Server Cluster Environment

Chihwan Ham, Hoyeon Kim, Dongjun Kim, Hukeun Kwak,  
Huiung Kwon, Youngjong Kim, Kyusik Chung  
School of Electronics Engineering, Soongsil University

### 요 약

기존의 서버 전원 모드 제어에는 서버를 Off할 때 정적 종료 방식을 사용하는 관계로 서버가 사용자 요청을 모두 처리하는 최적의 종료 시간을 찾는 데 시간이 많이 걸리는 단점을 가진다. 이 시간이 짧게 되면 사용자 QoS를 보장할 수 없고, 반대로 이 시간이 길게 되면 전력 절감을 기대할 수 없다. 본 논문에서는 정적 종료 방식의 단점을 극복하는 동적 종료 방식을 제안한다. 제안된 방식은 최적의 종료 시간을 찾을 필요 없이 각 서버가 사용자의 요청을 모두 처리하였을 때 자동적으로 서버를 Off한다. 제안된 방법은 최적의 시간을 자동적으로 찾아내기 때문에 사용자 QoS를 보장하고, 전력을 절감한다. 실험은 30대의 PC 클러스터를 이용하여 수행되었고, 실험을 통하여 제안하는 동적 종료 방법이 기존의 정적 종료 방법에 비해 운영자의 수고 없이 자동적으로 전력 절감 및 사용자 QoS에 기여함을 확인하였다.

### 1. 서론

서버, 스토리지 및 네트워크 장비들을 직접 운영하는 데이터 센터는 "전기 먹는 하마"로 불릴 정도로 전력 소비량이 많은 곳으로 그린 IT를 실현하는 데 있어 우선적인 고려 대상이다. 데이터 센터는 크게 IT 장비와 이를 안정적으로 운용하기 위한 기반 설비로 나눌 수 있다. IT 장비는 데이터 센터마다 약간의 차이는 있으나 데이터 센터에서 사용하는 전체 에너지 소비량의 약 50% 이상을 차지한다. IT 장비의 에너지 효율은 데이터 센터의 기반 시설에 비해 낮은 편이다. 플랫폼에 따라 차이는 있겠지만 서버의 평균 사용률이 20%를 넘지 않는다는 것은 에너지 사용량을 줄일 수 있는 여지가 상당히 있다는 것을 의미한다[1].

데이터 센터에서 서버들의 에너지 소모를 줄이기 위해서는 1) 에너지 절감형 서버 하드웨어를 준비하는 방법 [2], 2) 멀티코어/멀티프로세서를 사용하는 서버를 사용하는 경우 한 서버 내 CPU 전력 소모를 최소화하도록 서버 운영체제에서 에너지 절감형 스케줄링을 사용하는 방법 [3], 3) 많은 수의 서버들로 구성된 서버 클러스터에서 서버들 전체 총 전력 소모를 최소화하도록, 필요하다면 일부 서버를 Off하는 에너지 절감형 부하 분산기를 사용하는 방법[4]들이 있다.

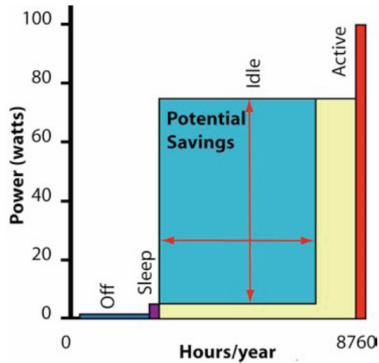
본 논문은 데이터 센터에서 서버들의 에너지 소모를 줄이는 방법 중 3)에 해당된다. 데이터 센터에는 규모에

따라 서버가 수백/수천/수만대로 구성되어 있고, 이 서버들이 담당하는 서비스별 그룹으로 나뉘는데 이 그룹을 각각 하나의 서버 클러스터라고 부른다. 서버 클러스터 규모에 따라 하나의 부하 분산기가 하나의 서버 클러스터 또는 여러 서버 클러스터의 부하 분산을 담당한다. 한 대의 부하 분산기가 수십/수백/수천대의 서버를 담당한다는 점을 고려할 때 부하 분산기를 통한 에너지 절감효과는 매우 클 것이다.

(그림 1)은 미국 로렌스 버클리 국가 연구소에서 분석한 PC에서 에너지 사용 패턴을 보여준다. PC 에너지 사용 패턴을 보면 PC가 idle한 상태에서도 전력 사용률은 60%가 넘으며, Sleep 모드에서는 약 5%의 전력 사용률을 가진다. 또한 상업적인 서버의 활용률은 15~20% 정도로 분석되었다[5]. PC 에너지 사용 패턴을 기반으로 상업 서버의 에너지 사용 패턴을 유추해보면 비슷할 것으로 보이므로 많은 서버들이 들어있는 서버 클러스터에서 시간별 부하 특성에 따라 다른 부하 분산 정책을 적용할 수 있다. 전체 부하가 많은 경우에는 모든 서버를 On하여 기존 방식처럼 부하를 균등하게 배분하며 전체 부하가 작을 경우에는 일부 서버들은 Off하고 나머지 서버들에 한해서 부하를 균등하게 배분할 수 있다. 이렇게 함으로써 같은 부하 분산 정책을 항상 적용하는 것보다 서버들의 에너지 소모를 줄일 수 있다.

서버를 Off할 때 중요한 것은 언제 서버를 Off할 것인가에 대한 문제이다. 사용자 요청을 모두 처리하는 순간에

Off 하여야 하는데 기존 방법의 경우 이 값이 고정되어 있다. 문제는 이 시간이 너무 길면 전력을 절감할 수 없고, 반대로 이 시간이 너무 짧으면 사용자 QoS를 보장할 수 없다. 본 논문의 목적은 서버를 Off할 때 최적의 시간을 자동적으로 찾아내는 동적 종료 방법을 사용하는 것이다. 이를 위해 Graceful Shutdown 방법[6]을 에너지 절약을 위한 서버 종료에 적용한다. Graceful Shutdown 방법은 클러스터 상에서 서버들의 사용자 요청 관리를 위해 주로 사용한 방법으로 아직까지 이를 에너지 절약을 위한 서버 On/Off에 적용한 기존 연구는 없다.



(그림 1) PC 에너지 사용 패턴

본 논문의 구성은 다음과 같다. 2장에서는 에너지 절약을 위한 서버 전원 모드 제어에서 기존의 종료 방법에 대한 연구와 문제점을 소개한다. 3장에서는 기존의 종료 방법의 문제점에 대한 해결 방법에 대해 소개한다. 4장에서는 실험 및 토론을, 5장에서는 결론 및 향후 연구 방향을 제시한다.

## 2. 기존 연구

### 2.1 정적 종료 방법[7]

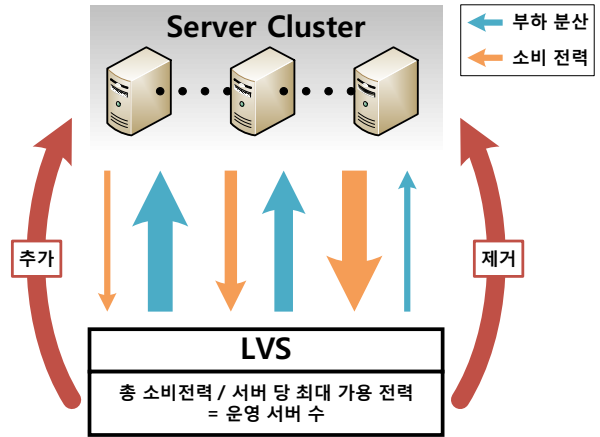
기존의 서버 클러스터는 성능만을 보장하기 위해, 실시간 요청 수량에 관계없이 정해진 서버 수를 항상 최대 운영하고 있다. 이는 QoS는 보장되었지만, 소비 전력의 효율성이 떨어짐으로 인해 소비 전력이 낭비되는 문제점이 있다. 이를 해결하기 위해 클라이언트의 요청 수량에 따라 동작될 서버의 전원 모드를 제어함으로써 소비 전력의 효율성을 높이는 알고리즘이 구현되었다. 이 알고리즘의 동작 과정은 다음과 같다.

(1) 현재 운영 중인 각 서버의 현재 소비 전력을 계산하고, 소비 전력이 많은 서버는 부하를 적게 주고, 소비 전력이 적은 서버는 부하를 많이 주어 균등한 부하 분산을 유지한다.

(2) 총 소비 전력, 현재 운영 중인 서버의 수, 서버 당 최대 가용 전력을 이용하여 서버의 전원 모드 제어의 방향을 결정한다. 이는 총 소비 전력을 서버 당 최대 가용 전력으로 나누어 운영되어질 서버의 수를 결정하여 현재 운영 중인 서버의 수에서 추가/제거를 결정하게 된다.

(3) 서버의 추가는 서버의 전원 모드를 Off 모드에서 On 모드로 전환한다는 의미이며, Off 모드의 서버 중에서 소비 전력이 적은 서버를 On 모드로 전환한다.

(4) 서버의 제거는 서버의 전원 모드를 On 모드에서 Off 모드로 전환한다는 의미이며, 이 때 바로 종료하지 않고 기존의 연결된 부하를 모두 처리할 때까지 90초를 기다린 후, Off 모드로 전환한다.



(그림 2) 기존 시스템의 동작 과정

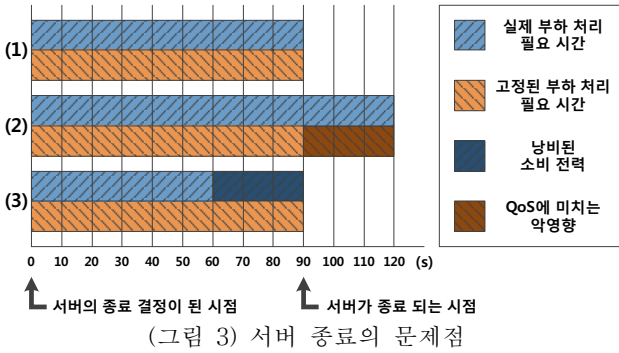
### 2.2 기존 연구의 문제점

클라이언트의 요청 수량에 따라 동작될 서버의 전원 모드를 제어함으로써 소비 전력의 효율성을 높이는 기존의 연구 방법은 전원 모드가 On 모드에서 Off 모드로 변경될 때 바로 실행되는 것이 아니고, 기존에 연결된 부하 처리를 위한 시간을 모두 기다린 후 실행하게 된다. 그 시간은 여러 번의 실험을 통해 90초로 정해졌고, 이 방법으로 인해 기존 클러스터와 같은 성능을 유지하면서, 소비 전력은 29% 절감하게 되었다. 물론 이 방법은 기존 클러스터보다 우수한 소비 전력 절감 효과를 내지만, 서버에 연결된 부하 처리 시간이 90초로 정해져 있다는 단점이 있다. 그 단점의 예를 들면, 부하 처리를 90초보다 빨리 끝낸 서버는 90초까지 기다리는 동안의 소비 전력을 낭비하게 된다. 또는 부하 처리를 90초보다 늦게 끝내야 하는 서버는 부하 처리를 끝까지 마치지 못하고 종료됨으로써 QoS를 보장할 수 없다. 문제점의 쉬운 이해를 위해 (그림 3)을 통해 설명한다. (그림 3)은 기존의 에너지 절약을 위한 서버 전원 모드 제어의 경우를 나타내고, 그 중 전원 모드를 Off 모드로 전환할 시 문제점을 보여준다. 0초는 서버의 종료 결정이 되었을 때를 말하고, 90초간 기다렸다가 90초가 되는 시점은 서버의 종료가 이루어지는 시점을 말한다.

(1) 실제 부하 처리 필요 시간과 고정된 부하 처리 필요 시간이 같을 때이며, 이 상황에서는 아무 문제가 없을 것이다.

(2) 실제 부하 처리 필요 시간이 고정된 부하 처리 필요 시간보다 클 때이다. 이 상황에서는 실제 부하 처리 필요 시간이 더 많기 때문에 90초 만에 서버를 종료하게 되면 QoS를 보장할 수 없는 상황이다.

(3) 실제 부하 처리 필요 시간이 고정된 부하 처리 필요 시간보다 작을 때이며, 이 상황에서는 부하 처리를 다 끝냈는데도 90초가 되기 전까지 어떠한 부하 처리를 하지 않은 채 소비 전력을 낭비하게 된다.



이런 문제점을 보완하기 위해 제안하는 종료 방법은 어떠한 상황이든지 부하 처리 필요 시간을 동적으로 운영하여 소비 전력과 QoS가 모두 만족되는 결과를 볼 수 있는 방법을 제시한다.

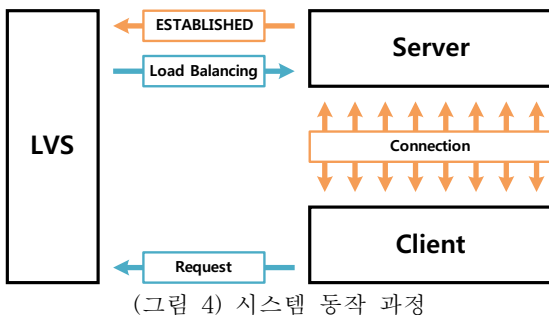
### 3. 동적 종료 방법

우선 QoS를 만족하기 위해서는 각 서버의 연결되어진 부하 처리가 모두 끝났을 때 종료되어야 한다. 그러기 위해서 각 서버의 네트워크 상태 정보를 LVS가 알아야 종료될 서버의 동적인 종료를 할 수 있다. 실제로 'netstat'이라는 명령어를 통해 서버의 상태를 알아 볼 수 있으며, 연결 중 부하 처리를 하는 동안에는 ESTABLISHED의 상태 정보를 나타낸다. 즉, ESTABLISHED 정보가 없다면 모든 부하 처리를 마친 상태이기 때문에 종료를 해도 QoS를 보장할 수 있다. 서버의 네트워크 상태 정보를 이용하여 종료 시점을 결정하는 방법은 다음과 같다.

(1) 각 서버의 상태 정보 중 ESTABLISHED 정보를 LVS로 매초 전송한다. 이 정보는 서버가 클라이언트의 요청에 대한 부하처리의 개수 즉, 실제로 연결된 개수를 뜻하며, 연결 중인 서버에서는 다수의 연결이 존재한다.

(2) LVS는 종료해야 할 서버를 선택하고, 그 서버의 연결 정보를 계속 체크하면서, 종료 시점을 기다린다.

(3) 선택된 서버의 연결 정보 즉, 연결된 개수가 0이 되는 시점에 서버를 종료하게 된다. 이 때, 시간이 얼마나 걸리든지 QoS는 만족하게 된다.

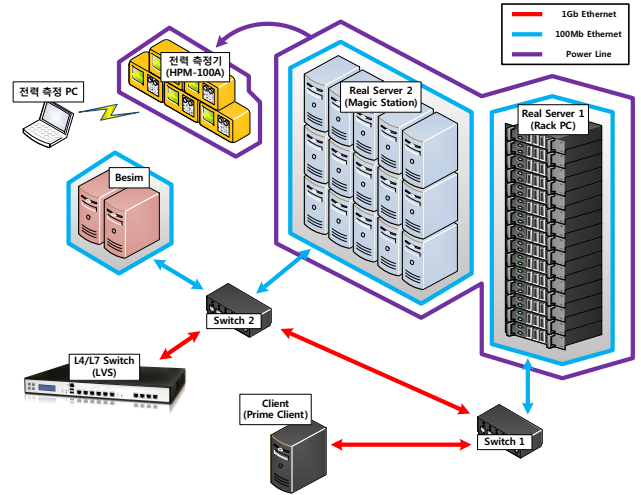


## 4. 실험 및 토론

### 4.1 실험 환경

(그림 5)는 실험 전체 구성을 나타낸다. 전력 측정기는 전체 Real Server의 전력 값을 측정하여, Zigbee를 통해 전

력 측정 PC로 전송한다. Switch 2대를 이용하여 Real Server 1, Real Server 2, Client, Besim을 각각 분리된 네트워크를 구성하였다. 스위치-LVS-Client간의 네트워크 연결은 1Gb Ethernet을 사용하였고, 나머지는 100Mb Ethernet을 사용하였으며, 네트워크 구성은 Direct Routing을 사용하여 네트워크 내 병목을 최소화하였다.



<표 1>은 실험에 사용된 하드웨어와 소프트웨어를 나타낸다.

<표 1> 실험용 하드웨어 & 소프트웨어

	하드웨어		소프트웨어	개수	
	CPU (Hz)	RAM (MB)			
LVS	Dual Core 2.93G	DDR3 2048	RR/WRR	1	
SPEC web	Client (Prime Client)	Quad Q6600 2.4G	DDR2 4096	SPECweb	1
	Besim	P-IV 1.8G	DDR 512	apache/SPECweb	2
	Real Server 1	P-IV 2.26G	DDR2 2048	apache	15
	Real Server 2	P-IV 1.8G	DDR 512	apache	15

### 4.2 실험 방법

각 실험은 <표 2>와 같이 진행되었으며, 요청 패턴은 SPECweb의 Banking Design[8]을 이용하였고, 실험은 60분씩 반복하였다. 실험 1은 기존의 시스템인 RR과 제안정적 종료 시스템 WRR1과 제안하려는 동적 종료 시스템 WRR2의 3가지 방식을 비교하고, 실험 2는 정적 종료 시스템 WRR1의 90초 종료와 60초 종료, 120초 종료를 실험을 추가하여 3가지 방식을 비교한다.

<표 2> 각 실험 진행 방법

	적용 시스템	부하 처리 시간(s)	요청 패턴	시간
실험 1	RR	X	SPECweb Banking Design	60분
	WRR1	90		
	WRR2	Dynamic		
실험 2	WRR1	60		
	WRR1	120		

### 4.3 실험 결과

<표 3>은 실험 1의 결과이다. 실험 1에서 본 논문에서 제안하는 동적 종료 시스템은 소비 전력 절감 부분에 있어서 기존의 시스템 RR보다 정적 종료 시스템과 마찬가지로

지로 동일한 성능과 소비 전력을 절감하는 효과를 볼 수 있었다. 그리고 정적 종료 시스템 WRR1보다 소비 전력 절감 효과는 조금밖에 보지 못했다. 이유는 다음의 계산된 값을 통해 알 수 있다.

<표 3>의 (절감 시간)은 (부하 처리 시간) \* (종료 횟수) 이다. WRR2는 WRR1보다 약 383초를 절감하였다.

- (1)  $WRR1 = 90 * 42 = 3780$
- (2)  $WRR2 = 79 * 43 = 3397$
- (3)  $WRR1 - WRR2 = 3780 - 3397 = 383$

RR의 2238Wh라는 것은 1시간동안 30대의 소비 전력이 2238W이고, 1시간동안 1대의 평균 소비 전력은 74.9W이다.

(4)  $2238 / 30 = 74.9$

즉, 1시간(3600초) 동안 75W를 쓴다는 결론과 383초 동안은 8W를 쓴다는 결론이 나온다.

(5)  $3600 : 75 = 383 : x$ ,  $x=7.9999...$

즉, WRR1과 WRR2의 평균 소비 전력은 약 8W의 차이를 보이므로 실제로 절감되는 소비 전력은 크지 않다. 즉, 소비 전력 절감 효과는 기존의 정적 종료 시스템에 최적화되어 있기 때문에 별 차이가 없는 것이다. 하지만 Fail의 수는 기존 방법이 3개인데 반해 제안된 방법은 0이 됨으로써 사용자 QoS를 확실하게 보장함을 알 수 있다.

<표 3> 실험 1 결과

적용 시스템	부하 처리 시간 (s)	누적 소비 전력 (wh)	Time Tolerable	Fail	종료 횟수	절감 시간
RR	x	2238	0	0	x	x
WRR1	90	1704	2	3	42	3780
WRR2	79	1697	1	0	43	3397

<표 4>는 실험 2의 결과이다. 부하를 처리하는 시간을 90초보다 작게 주었을 때는 Fail이 많아지게 된다. 그리고 서버종료시간을 90초보다 크게 주었을 때는 소비 전력이 늘어나게 되는 단점이 생긴다. 즉, 정적 종료 시스템은 시간을 최적화해야 하는 번거로움과 다른 환경의 서버 클러스터에서는 전력 절감 및 QoS를 보장할 수 없는 단점이 있다.

<표 4> 실험 2 결과

적용 시스템	부하 처리 시간 (s)	누적 소비 전력 (wh)	Time Tolerable	Fail
WRR1	60	1614	2	175
	90	1704	2	3
	120	1726	2	0

#### 4.4 토론

본 논문에서 제안된 동적 종료 시스템은 기존의 정적 종료 시스템에 비해 전력 절감 및 QoS를 보장할 수 있다. 기존의 정적 종료 시스템에서는 연결이 남아 있는 경우에 종료하게 되면 사용자 QoS를 보장할 수 없고, 반대로 연결이 남아 있지 않은데도 종료하지 않는다면 전력을 절감할 수 없다. 또한 이 최적의 종료 시간을 찾는다는 것은 많은 노력이 필요하고, 다양한 상황에서 항상 동작하는 것

은 아니다. 반면 동적 종료 시스템의 다양한 상황에서 최적으로(자동적으로) 서버를 종료하므로 전력 절감 및 사용자 QoS를 보장할 수 있다.

#### 5. 결론

기존의 정적 종료 시스템은 종료 시점이 정해져 있어 어떤 상황에서도 QoS 및 전력 절감을 보장하기 힘들다. 문제는 최적의 종료 시간을 찾기도 힘들 뿐만 아니라 찾았다 하더라도 해당 시간이 모든 사용자 요청을 Fail 없이 처리할 수 없고 또한 그 시간이 길어져 전력 절감을 할 수 없다는 점이다. 하지만 본 논문에서 제안된 동적 종료 시스템은 운영자의 노력 없이 자동적으로 최적의 종료 시간을 찾고(각 서버에 사용자 요청이 끝나는 시점을 실시간으로 파악하여) 이를 에너지 절약을 위한 서버 Off에 적용함으로써 항상 QoS 및 전력 절감을 보장할 수 있다.

#### 참고문헌

- [1] 이상학, 문성준, 김진환, 신상용, 서용원, 최영진, "공공 부문의 그린 데이터 센터 구현 방안에 관한 연구", 한국정보과학회지, 제 27권, 제 11호, pp. 48-57, 2009.
- [2] Chenguang Liu, Jianzhong Huang, Qiang Cao, Shenggang Wan, Changsheng Xie, "Evaluating Energy and Performance for Server-Class Hardware Configurations", 6th IEEE International Conference on Networking, Architecture and Storage, 2011.
- [3] J. Mair, K. Leung, Z. Huang, "Metrics and task scheduling policies for energy saving in multicore computers", 11th IEEE/ACM International Conference on Grid Computing (GRID), 2010.
- [4] Xinying Zheng, Yu Cai, "Markov Model Based Power Management in Server Clusters", IEEE/ACM Int'l Conference on Green Computing and Communications (GreenCom), 2010.
- [5] Bruce Nordman, "What the Real World Tells Us about Saving Energy in Electronics", Symposium on Energy Efficient Electronic Systems, 2009.
- [6] Graceful Shutdown, [http://msdn.microsoft.com/en-us/library/ms738547\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms738547(v=vs.85).aspx)
- [7] 김호연, 함치환, 곽후근, 권희용, 김영종, 정규식, "서버 클러스터 환경에서 에너지 절약을 위한 동적 서버 전원 모드 제어", 한국정보처리학회 논문지, 제 19-C권 제 3호, 2012 6월 게재예정.
- [8] SPECweb Banking Design, <http://www.spec.org/web2005/docs/BankingDesign.html>