공동 활용 컴퓨팅 시스템에서의 사용자 작업별 전력 사용량 생성 기능 구현

권민우*, 고동건*, 윤준원*, 홍태영*
*한국과학기술정보연구원 슈퍼컴퓨팅인프라센터
mwkwon81@kisti.re.kr

Implementation of power consumption generation for each user job of shared utilization computing system

Min-Woo Kwon*, Dong-Gun Koh*, JunWeon Yoon*, TaeYoung Hong*
*Dept. of Supercomputing Infrastructure Center, KISTI

요 약

한국과학기술정보연구원(KISTI)에서는 슈퍼컴퓨터 5호기인 누리온과 보조시스템인 뉴론 시스템을 이용하여 공동 활용 서비스를 진행하고 있다. 다수의 사용자가 접속하여 사용하는 이와 같은 컴퓨팅 시스템들은 구축비용 뿐 아니라 운영 중에 발생하는 전력 소비 비용이 상당하다. 본 논문에서는 뉴론에서 수행되는 사용자의 작업별 전력 사용량을 생성하는 기법에 대해 소개하고 전력 소비량을 조절할 수 있는 방안에 대해 논의한다.

1. 서론

최근 AI, 빅데이터 연구가 활성화되면서 전 세계적으로 해당 연구에 특화된 다양한 아키텍처의 시스템이 개발되고 있다. 이러한 시스템들은 높은 구축비용 뿐 아니라 운영 중에 발생하는 전력 소비에 따른 운영비용이 상당하다. 이를 해결하기 위해 아키텍처 개발 시에 저 전력 시스템 개발이 대형 컴퓨팅 시스템에서 중요한 화두가 되고 있다. KISTI에서 운영 중인 슈퍼컴퓨터 5호기 누리온 역시 8,432대의 계산노드로 구성되어 있으며, 보조 시스템인 뉴론 시스템도 GPU 152개가 장착된 서버 76대로 구성되어 있다[1]. 이러한 시스템들은 전력 소비에 따른 운영비용의 부담이 크다.

본 논문에서는 전력 소비량을 조절하기 위한 기초데이터로 활용될 사용자의 작업별 전력사용량을 생성하는 기법에 대해 소개한다. 계산노드에서 제공하는 순간 전력 값을 로그 형태로 저장한 후에, 뉴론에서 사용 중인 SLURM 작업 스케줄러에서 제공해주는 종료된 작업에 대한 상세한 정보를 활용하여 사용자 작업별로 전력사용량을 계산하게 된다. 구체적으로 작업이 수행된 모든 계산노드의 전력사용량을 수집하여 최종적인 사용량을 계산하여 로그 형태로 생성한다.

2. 계산노드 별 전력 사용량 수집

계산노드에서 사용되는 전력 사용량은 하드웨어를 원격으로 관리하는데 사용되는 인터페이스인 IPMI (intelligent Platform Management Interface)를 통해 수집할 수 있다.

```
 $ ipmitool sdr | grep Watts
Pwr Consumption  | 88 Watts          | ok
```

(그림 1) IPMI를 이용한 전력 사용량 조회

그림 1과 같이 ipmitool 명령어를 이용하여 SDR (Sensor Data Record) 정보를 획득할 수 있으며, 이 중에서 'Pwr Consumption' 필드에서 전력 사용량인 'Watts' 값을 획득할 수 있다. 서버에 장착된 Power Supply(PS) 개수만큼 입력된 순간 전력 값이 획득된다.

```
queue:skl,node:skl10,20200921113105,uPower:352
queue:skl,node:skl10,20200921113205,uPower:418
queue:skl,node:skl10,20200921113305,uPower:352
queue:skl,node:skl10,20200921113405,uPower:352
queue:skl,node:skl10,20200921113505,uPower:418
queue:skl,node:skl10,20200921113605,uPower:418
queue:skl,node:skl10,20200921113705,uPower:418
queue:skl,node:skl10,20200921113805,uPower:352
```

(그림 2) 공유디렉터리에 저장된 전력 사용량 정보

그림 2와 같이 IPMI를 통해 획득된 PS별 순간 전력 사용량 정보를 리눅스의 crond 데몬을 이용하여 1분마다 수행하여 작업 스케줄러가 동작하고 있는 인프라노드가 접근할 수 있는 공유 디렉터리의 파일에 일자별로 기록한다. 뉴론 시스템은 모든 인프라, 계산노드가 고성능의 Infiniband 네트워크로 병렬 스토리지를 공유하고 있으므로 병렬 스토리지 내부에 노드별 전력 사용량 정보를 취합하여 모니터링할 수 있다.

3. SLURM 작업 스케줄러의 과금 로그를 이용한 작업별 전력 사용량 생성

뉴론은 SLURM 작업 스케줄러의 DB 데이터를 추출하여 일자별로 과금 로그를 생성하고 있다. 스케줄러 DB에는 작업의 시작, 변경, 종료 시의 기록이 남게 된다[2]. 본 논문에서는 종료 시에 기록되는 정보를 이용하여 전력 사용량을 계산한다. 표 1은 과금 로그에서 제공하는 종료된 작업의 정보이다.

<표 1> 과금 로그의 종료된 작업 정보

명칭	정보
user	사용자 ID
jobid	작업 ID
jobname	작업 명칭
queue	작업이 수행되는 큐
nodelist	작업이 수행되는 노드리스트
start	작업이 시작된 시간
end	작업이 종료된 시간

SLURM 제외한 다른 작업 스케줄러(PBS)에서도 과금 로그를 제공하고 있다[3]. 본 논문에서 제안한 방식과 유사한 방식으로 GPU 카드의 사용 통계정보를 획득하는 연구가 이미 수행되었다[4]. 그림 3은 Perl 스크립트로 구현한 작업별 전력 사용량 생성 기능의 Pseudo코드이다.

```
FOR each node in nodelist
  FOR from startday to endday
    Open power usage file of each server
    IF starttime <= timestamp <=endtime THEN
      Accumulate power usage
Calculate average power usage
```

(그림 3) Pseudo코드

뉴론과 같은 클러스터 시스템에서는 사용자의 작업이 MPI 라이브러리를 사용하여 멀티노드에서 수행된다. 그림 2와 같이 큐별, 계산노드별, 일자별로 저장되어 있는 전력 사용량 정보 파일에서 작업이 시작한 일자와 시간으로부터 작업이 종료한 일자와 시간 사이의 전력 사용량을 누적 시키고 최종적으로 평균값을 구함으로써 작업 수행 시간 동안 사용한 전력 사용량을 계산할 수 있게 된다.

```
userid:mwkwon queue:skl jobid:42871 jobname:BMT uPower:384.4 time:1033.0
userid:mwkwon queue:skl jobid:42872 jobname:BMT uPower:376.9 time:891.0
userid:mwkwon queue:skl jobid:42873 jobname:BMT uPower:368.5 time:733.0
userid:mwkwon queue:skl jobid:42874 jobname:BMT uPower:403.3 time:597.0
userid:mwkwon queue:skl jobid:42875 jobname:BMT uPower:394.4 time:889.0
userid:mwkwon queue:skl jobid:42876 jobname:BMT uPower:345.2 time:734.0
userid:mwkwon queue:skl jobid:42879 jobname:BMT uPower:386.9 time:1032.0
```

(그림 4) 작업별 전력 사용량 정보

작업 스케줄러를 사용하지 않는 시스템의 경우는 대부분 다수의 사용자가 사용하는 공동 활용 시스템이 아닌 개인이나 소규모 그룹에서 사용하는 시스템일 가능성이 높고 이런 환경에서는 사용자 본인이 수행되는 서버의 리스트를 직접 작성하여 인터랙티브하게 작업을 수행할 것이다. 이런 환경에서는 Python이나 Perl 스크립트 등을 이용하여 서버 리스트와 작업 수행 정보를 직접 입력하여 본 논문에서 제안하는 기능을 충분히 구현할 수 있다.

3. 결론 및 향후

사용자의 작업별 전력사용량 정보는 시스템 사용 비용을 책정하는 과금 정책에 반영하여 사용자 스스로 전력사용에 대한 책임을 지게 함으로서 전체적인 전력 사용비용을 절감시키는 방안이 있다. 향후에는 본 논문에서 제안한 서버별 전력 사용량을 모니터링하고 제어하는 툴을 개발하여 전체 클러스터 시스템의 전력 사용량을 제어하는 기능을 구현하고자 한다. 작업 스케줄러를 사용하지 않는 일반적인 클러스터 시스템에서도 이러한 기법을 활용해 시스템의 전력 사용량을 제어할 수 있을 것으로 기대한다.

참고문헌

[1] National Supercomputing Center, KISTI, http://ksc.re.kr
[2] slurm workload manager, SchedMD, https://slurm.schedmd.com
[3] PBS Professional 19.2 Admin Guide, Altair, https://www.altair.co.kr/pbs-works-documentation
[4] 권민우, 윤준원, 홍태영 “클러스터 시스템에서 GPU 사용 통계정보 획득 방안에 대한 연구” 2018년 추계학술발표대회, 476-477, 2018.