NEC 시스템의 효율적인 활용을 위한 작업관리 큐 설계 및 구현

이영주, 성진우, 이상동, 김중권
한국과학기술정보연구원
e-mail:yjlee@kisti.re.kr

The NQS Queue Design and Implement in NEC System for Efficient Practical Use

Young-Joo Lee, Jin-Woo Sung, Sang-Dong Lee,
Joong-Kwon Kim
Korea Institute of Science Technology Information

요 약

시스템의 한정된 자원을 다수의 사용자들에게 효율적으로 분배하기 위해서 작업관리 시스템을 사용한다. 작업관리 시스템은 그 종류가 여러 가지 있는데, 시스템의 종류나 작업의 특성에 따라서 적당한 작업관리 시스템을 사용한다. NEC 시스템에서는 작업관리 시스템으로 NQS를 사용하고 있으며, 이 작업관리 시스템을 어떻게 잘 설계하느냐에 따라 시스템 자원의 활용율이 달라지기 때문이다. 따라서 한정된 시스템의 자원을 다양한 사용자들의 작업 특징에 따라 적절히 자원을 배분 할 수 있도록 차등 큐를 설계하고 구현하였다, 그리고 작업관리 시스템 각각의 큐에서 처리된 작업의 turnaround time을 분석 하였다.

1. 서론

작업관리 시스템은 다수의 사용자가 이용하는 시스템의 자원을 효율적으로 분배하고 활용될 수 있도록 작업을 관리하는 시스템이다. 이러한 기능의 시스템에는 대표적으로 NQS, LoadLeveler, PBS, LSF, Condor 등 많은 작업관리 프로그램이 있다. NEC 시스템은 주로 작업관리 시스템으로 NQS(Network Queuing System)을 사용하여 사용자들에게 자원을 할당하고 있다. 일반적으로 작업관리 시스템을 사용자의 작업에 적용하는 경우, 그 설계 방식에 따라 시스템 전체의 작업처리 효율에 많은 영향을 줄 수 있기 때문에 작업관리 시스템을 설계하고자 할 때는 시스템의 환경이나 특성 그리고 사용자 작업의 패턴 등을 충분히 분석하여 반영하여야 한다.

본 논문에서는 작업관리 시스템의 각각의 큐를 설계하고 구현된 시스템에서 CPU 사용율에 따라서 turn around time이 어떻게 변화하는지 분석하고자 한다. 그 결과로서 작업관리 시스템의 설계가 잘 되었는지 검증할 수 있고 CPU의 사용율이 높아짐에 따라 작업의 처리속도가 어떻게 변화하는지 분석하고자 한다.

작업의 turnaround time을 분석하기 위하여 NQS 로그의 데이터 중에서 작업의 대기 시간, 경과시간, CPU 시간을 이용하여 각각의 큐별의 성능을 비교 분석하였다.

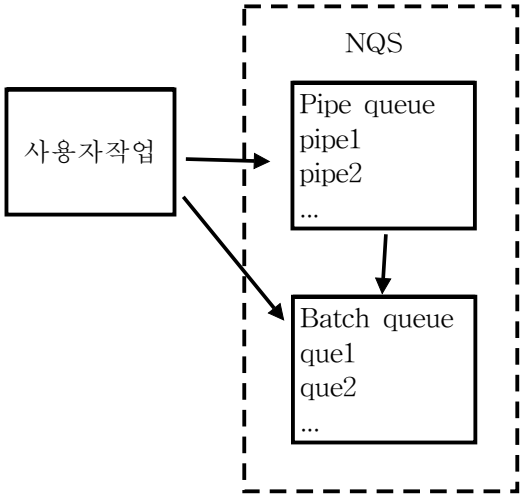
2. 관련 연구

2.1 NQS 구조

NQS의 큐는 크게 파이프 큐와 배치 큐가 있다. 파이프 큐는 사용자가 보낸 작업을 일시적으로 저장하였다가 적당한 배치 큐에 할당하는 역할을 하는 큐이고, 배치 큐는 작업이 실제 실행되는 큐로서 배치 큐에 정의한 자원을 가지고 작업이 처리된다. 하나의 파이프 큐는 여러 개의 배치 큐를 가질 수 있다.

배치 큐에 작업을 보내는 방법은 크게 두가지가 있으며, 하나는 작업을 파이프를 통하여 적당한 배

치 큐로 할당하는 방법이며, 다른 하나는 사용자가 작업을 직접 배치큐로 보내는 방식이다.



(그림 1) NQS 구성도

2.2 NQS 큐의 속성

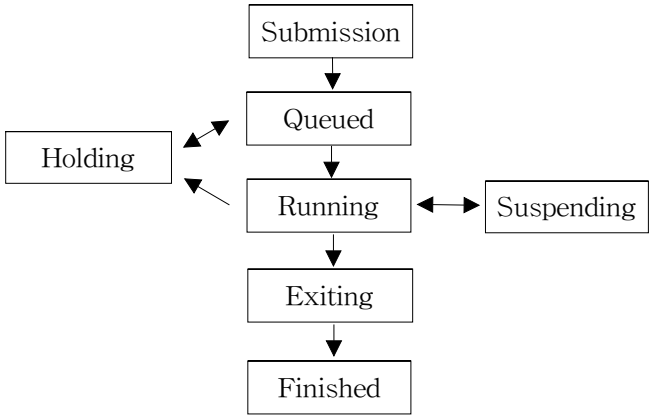
NQS의 큐의 속성을 정의하는 NQS 환경 변수는 여러 가지가 있으며 그 중에서 작업 처리의 속도와 크게 관련된 NQS 변수는 Base priority와 Time slice 등이 있다. Base priority는 프로세스의 할당에 관련되는 변수로서 그 수치가 적을수록 프로세스의 할당을 빨리 받게 된다. Time slice는 실행 프로세스를 할당 받아서 작업이 실행되는데 지속되는 시간을 의미하며 해당 시간 동안 계속해서 실행 프로세스를 점유하게 된다. 해당 변수가 크면 클수록 해당 큐의 속성이 그 만큼 유지된다는 것을 의미한다. 그래서 큐에 할당된 처리속도가 빠르거나 느릴 경우에 해당 변수의 시간을 길게 주면 그 영향이 오랫동안 지속되어 진다. Aging 변수는 프로세서 스케쥴링 과정에서 실행 priority가 변화하는 어떤 전환점의 수치값이다. 프로세서의 처리 중 해당 CPU 시간이 지나감에 따라 계속 증가하다가 해당 변수가 Aging 변수보다 커지면 실행 priority 변수는 다시 하강하게 된다. 이 기간 동안에 프로세서는 계속 작업을 처리하게 된다.

2.3 NQS 작업 처리 과정

다음 (그림 2)은 NQS에 작업이 보내져서 해당 작업이 종료되기까지의 과정을 나타내고 있다. NQS에 의한 작업이 처리 순서는 단계별로 보면 크게 2단계로 구분할 수 있다. 하나는 사용자가 보낸 작업이 실행 배치 큐를 할당받기까지 파이프큐에서 작업이 대기하고 있는 대기시간(Waiting time)과 작업이 적당한 배치큐에 할당되어 해당 작업이 종료될 때까지의 경과시간(Elapsed time)이다. CPU 시간은 작업이 처리되기까지의 실행 시간이다.

작업이 큐에 보내져서 실행이 종료될 때까지 이러한 과정을 거치게 된다.

사용자가 직접 배치큐로 작업을 보내는 것이 가능할 경우는 작업이 배치큐에서 대기함으로 대기시간이 발생하고 그 기록이 NQS 로그에 남지만, 파이프큐를 통하여 배치 큐에 작업을 할당하여 처리할 경우 배치 큐에서는 작업의 대기 시간이 발생하지 않게 된다. 이것은 실행 가능한 배치큐가 있을 때 파이프큐에서 대기 하던 작업이 할당되기 때문이다. NQS의 통계는 NQS의 로그 데이터를 이용하여 작성하는데 파이프큐에서 작업이 대기한 시간은 NQS 로그에 남지 않는다. NQS 로그는 배치큐에 대한 정보만 기록되기 때문이다. 따라서 파이프큐에서 발생한 대기시간을 구하기 위해서 별도의 스크립트 프로그램을 만들어서 작업의 대기 시간을 얻어야 한다.



(그림 2) NQS 작업 처리 흐름도

2.4 NQS Load balancing

NQS에서 파이프큐를 통하여 작업이 각각의 노드에 있는 배치큐로 할당하는 방법은 아래와 같이 3가지 방식이 있다.

- Round Robin
  가장 단순한 방법으로서 각 노드에 대하여 일정한 순서에 의하여 작업을 하나씩 할당한다.
- Load Information Collection
  각 노드에 대한 CPU usage의 정보를 일정한 간격으로 참조하여 가장 사용율이 낮은 노드에 작업을 할당하다.
- Demand Delivery Method

작업이 NQS에 보내지면 작업이 우선 파이프 큐에 들어가서 대기하고 있다가 적당한 배치 큐를 찾아 할당하는 방법이다.

3. NQS 설계

3.1 NEC 시스템 구성

NEC 시스템은 <표 1>에서 보는바와 같이 SX-5 1대, SX-6 2대의 모두 3노드로 구성되어 있으며 이들 시스템은 벡터시스템으로서 3대의 노드가 클러스터 형태로 연결되어 있다. 하나의 노드는 8개의 CPU로 구성되어 있으며, 8개의 CPU가 하나의 메모리를 공유한다. 로긴 노드는 SX-6a 시스템이고 홈 디렉토리는 로컬로 연결되어 있으며 다른 노드는 GFS(Global File System)을 이용하여 홈디렉토리를 공유한다.

각각의 노드는 3중의 기가네트워크로 연결되어 있으며 첫 번째 네트워크는 외부 접속네트워크로 사용하고, 두번째는 GFS 파일을 공유하기 위하여 사용하고, 세 번째는 백업 등을 위한 NFS 연결을 위하여 사용하고 있다.

SX-6 시스템은 서로 고속의 IXS(Internode Crossbar Switch)로 연결되어 있어서 MPI 등의 작업이 가능하다. 각각의 노드마다 사용자의 작업 공간을 위하여 스크래치 디스크가 있으며, 3노드에서 공유할 수 있는 별도의 스크래치 디스크가 SX-6a에 로컬로 연결되어 있다.

<표 1> KISTI의 NEC SX 시스템 사양

구분	내 용	
모델명	SX-5/8B	SX-6/16M * 2대
운영체제	SUPER-UX R13.1	SUPER-UX R13.1
CPU	8개	16개
이론성능	80Gflops	160Gflops
메모리	128GB	128GB
디스크 용량	2.85TB	2.85TB

3.2 NQS 큐 구성

사용자들에게 다양한 서비스를 제공하기 위하여 작업의 처리 속도 차이에 의한 서비스 레벨을 두고 이 서비스 레벨에 따라서 각각의 차등큐를 만들었다.

차등큐는 사용자가 작업의 처리에 대한 특성과 긴급성을 고려하여 큐의 등급은 0.5 ~ 2까지 4단계로 구분하였다. 이렇게 함으로써 작업의 특징에 따라서 적당한 큐에서 작업의 실행이 가능하도록 하였다.

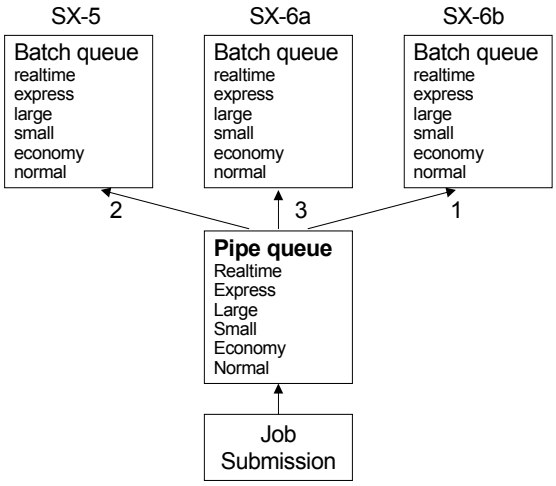
일반적으로 큐의 설계는 CPU와 메모리 등을 이용하여 설계하지만 이 시스템에서는 메모리의 크기가 충분히 크기 때문에 차등큐의 설계에 메모리의 변수는 반영하지 않았다.

<표 2 > 서비스 레벨 큐 구성

큐 이름	CPU Time	서비스 레벨	Base Pri	Time Slice	Aging
dedicated	무제한	2	60	2000	160
realtime	무제한	2	60	500	109
express	무제한	1.5	75	1000	109
small	180분	1	80	1000	109
large	무제한	1	80	1000	109
economy	무제한	0.5	85	1000	160

3.3 NQS의 작업 처리

(그림 3)은 사용자가 보낸 작업이 일단 파이프큐에 머물렀다가 파이프큐에서 실행 가능한 적당한 큐에 작업을 할당하는데, 이 때 파이프큐가 각각의 노드를 찾는 순서는 SX-6b, SX-5, SX-6a로 하였다. 이것은 전체 노드의 load balancing을 고려한 것으로서 작업이 먼저 할당되는 노드가 다른 노드에 비하여 부하가 많이 발생하기 때문이다.



(그림 3) 노드간의 NQS 구성

로긴 노드는 사용자가 작업을 하는 곳으로서 여러

가지 서버의 역할을 하기 때문에 부하가 많이 발생함으로 이를 고려하여 작업이 가장 나중에 할당되도록 고려하였다.

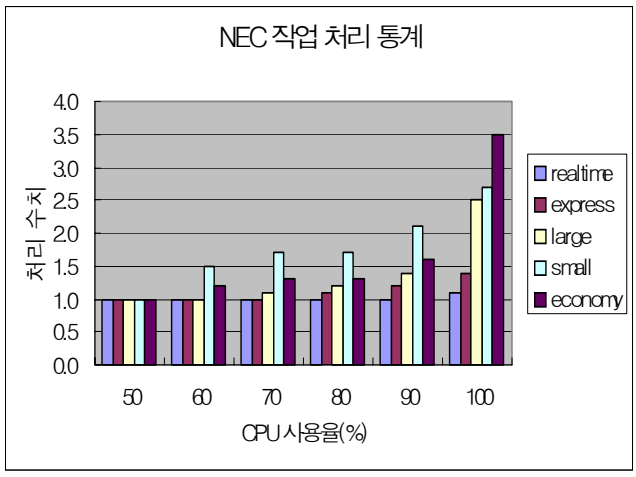
작업은 어느 노드에서나 큐로 보내는 것이 가능하고 사용자가 큐로 보낸 작업은 로긴 노드인 SX-6a에 대기하였다가 실행 가능한 배치큐로 할당된다.

4. 구현 결과

NQS 작업의 통계는 NQS에 로그에 기록된 데이터 중에서 아래와 같은 세가지의 시간 사이의 관계식을 만들고 비교 분석하였다. 이들의 관계식은 다음과 같은 식을 만들어 이용하였다.

Execution factor = (Waiting + Elapsed) / CPU

(그림 4)는 CPU의 사용율에 따른 작업처리 시간의 변화에 대한 통계이다. 통계는 1년의 사용기간 중에서 해당 사용율을 선택하여 3노드에 각각의 평균치를 통계를 산출하였다. 1년의 사용율 통계 중에서 하루만 어떤 사용율을 나타낼 경우 그에 대한 작업처리 시간 통계를 구한 결과 그 수치의 변동이 매우 심하였다. 그래서 1년의 사용율 통계 중에서 적어도 사용율이 2일 이상 경과한 날의 통계를 모두 취합하여 평균치를 구하였다. NQS의 작업 중에는 병렬 작업도 있는데 여기서는 통계 데이터의 일관성을 유지하기 위하여 일반 순차 작업만 선택하였다. 그래서 로그 데이터 중에서 CPU 시간이 작업의 경과시간보다 큰 데이터는 병렬 작업으로 간주하고 통계 데이터에서 제외하였다.



(그림 4) 작업의 turnaround time

(그림 4)에서 처리 수치는 Execution factor를 나타내며 그 수치가 1에 근접할수록 작업처리의 효율이 좋다는 것을 의미한다. 각각의 CPU 사용율에 따른 작업 처리 수치를 보면 큐의 성능이 낮을수록 작업처리 효율이 낮아지는 것을 알 수 있다. 큐 중에서 small 큐가 다른 큐에 비하여 작업처리 효율이 낮은 것은 어느 특정한 작업을 small 큐를 이용하여 계속 이어지는 작업 과정에서 대기 시간이 많이 발생하였기 때문이다.

5. 결론

CPU의 사용율이 50%이상이 되면서 실행 중인 작업들 간에 경쟁이 발생한다. 이때부터 작업 처리 factor가 낮은 큐에서부터 작업 간의 경쟁이 시작되어 CPU의 사용율이 점점 높아짐에 따라 반대로 작업처리 효율이 점점 낮아져서 CPU의 사용율이 100%에 근접하면 작업처리 시간은 약 2배 이상 소요되는 것을 알 수 있었다.

이의 결과로서 작업처리 시스템의 설계가 잘 이루어진 것을 알 수 있다.

시스템의 사용율이 증가할수록 시스템에서 작업처리 하는 CPU 시간보다 시스템의 작업을 스케쥴링 하는데 소요되는 시간이 더 많아짐으로 작업관리 시스템의 설계가 시스템에 많은 영향을 주는 것을 알 수 있다.

향후에는 NQS 시스템 외에 여러 가지의 작업관리 시스템을 사용하여 작업관리 시스템의 성능에 대한 비교 연구를 하고자 한다.

참고문헌

[1] NEC, "SUPER-UX NQS User Guide". NEC Corporation

[2] NEC, "SUPER-UX System Design Guide". NEC Corporation

[3] NEC, "Guide to System Operation of Super Computer SX-5 for KISTI". NEC Corporation

[4] 우준 "NEC SX-5 시스템에서 NQS 차등화 서비스 레벨 큐 구현 및 최적화" 한국과학기술정보인프라워크숍 2002. 12

[5] 이영주 외 "NEC 시스템에서 NQS 및 Load balancing 최적화" 정보처리학회 2003 추계학술지

[6] 이영주 외 "멀티노드에서의 NQS Node Balancing 설계 및 구현" 정보처리학회 2004 춘계학술지