대규모 HPC 클러스터 모니터링 시스템의 설계 및 구현

조혜영*, 홍태영*, 임용관**, 김성호*, 이식*
*한국과학기술정보연구원 슈퍼컴퓨팅본부, **나리넷
e-mail:{chohy, tyhong}@kisti.re.kr, yglim@narinet.com,
{sungho, siklee,}@kisti.re.kr

Design and Implementation of A HPC Cluster Monitoring System

Hyeyoung Cho*, Taeyoung Hong*, YongGwan Lim**, Sungho Kim*, Sik Lee,
*Supercomputing Center,
Korea Institute of Science and Technology Information
**Narinet

1. 서 론

초고속 네트워크 및 고성능 마이크로프로세서의 발전과 함께 클러스터 시스템이 더욱 대중화 되고 있다. 수백에서 수만대에 이르는 클러스터 시스템을 효율적으로 활용하기 위해서는 클러스터 시스템의 상태를 정확하게 모니터링하고, 시스템의 상태에 따라 능동적으로 대처할 수 있는 모니터링 시스템이 필수적이다. 중대규모 HPC 클러스터 시스템을 효율적으로 관리하기 위해서 모니터링 시스템은 노드수의 증가에 능동적으로 대처할 수 있는 scalability와 클러스터 시스템의 노드 혹은 네트워크의 failure에 대처할 수 있는 stability를 가져야 하며, 계산 자원 및 네트워크 자원의 소비를 최소화(low-overhead)할 수 있어야 한다. 본 논문에서는 대규모 HPC 클러스터 시스템을 효율적으로 관리하기 위한 클러스터 모니터링 시스템을 설계 및 구현하였다.

2. 본 론

클러스터 시스템은 일반적으로 수십 대에서 수백 대 이상의 노드로 구성되어 있기 때문에, 단일 프로세서 시스템에 비해 관리의 어려움이 많다. 클러스터 시스템의 관리를 효율적으로 수행하기 위해서, 클러스터 모니터링 시스템에 대한 다양한 연구들이 진행되어 왔다[1,2,3].

자체 설계 개발한 클러스터 모니터링 시스템(Cluster Monitoring System)인 Blank&Bunny 시스템은 크게 단일 노드에 대해 전체 모니터링 시스템의 서버 역할을 수행하는Black, 모니터링을 수행하는 단일 시스템 모니터링 에이전트인 Bunny로 구분된다. 여기에 각 노드의 상태와 성능, 에러 및 히스토리 등을 저장하는 DB가 있으며, 관리자에게 Web-Interface를 통하여 UI를 제공한다.

그림 1에 클러스터 모니터링 시스템 Black&Bunny의 전체 구조도를 도식하였다. 단일시스템 모니터링 에이전트(Bunny)는 단일 노드에 대해 모니터링을 수행하고 이를 서버로 전하는 역할을 담당한다. 이는 개별 노드에 대한 모니터링 에이전트로서 독립적으로 주요 프로세스에 대한 장애 및 하드웨어 레벨 에러를 감지하며, 필요한 경우 서버가 요청한 명령을 수행할 수 있는 인터페이스를 가지고 있다. 전체 모니터링 시스템의 서버 역할을 수행하는 Black은 전체 Bunny들을 모니터링하고 관리하며 Bunny가 전달한 모니터링 정보를 가공하여 DB에 저장하는 역할을 수행한다. Bunny는 멀티 스레드 방식으로 구성되어 있으며, Agent의 생사 여부를 주기적으로 체크하여 알리는 프로세스와 시스템의 오류 및 프로세스 장애를 감지하는 프로세스, 그리고 시스템 상태를 모니터링하는 프로세스 등으로 구성되어 있다. 이 프

로세스를 통해 수집된 정보는XML 메시지로 변환하여 네트워크 모듈을 통해TCP 기반으로 전송된다.

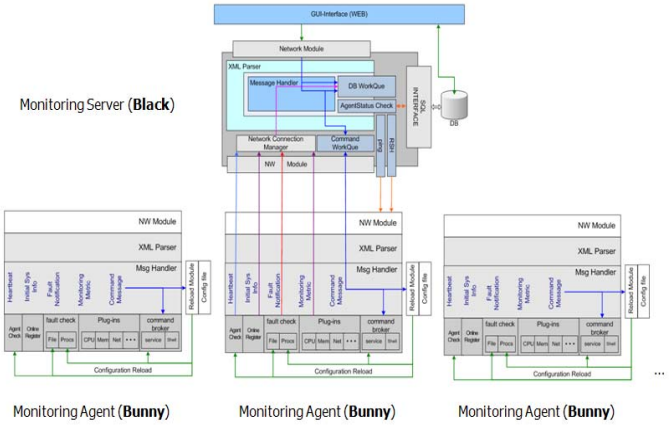

그림 1. Black&Bunny 전체 구조도

또한 Web-Interface를 통해 전달받은 사용자 명령을 Bunny로 relay하는 역할도 수행한다. 기본적으로 MySQL을 사용하고 있는 DB는 각 노드의 상태와 성능, 에러 그리고 히스토리 등을 저장하며 또한 에이전트 자체의 상태정보를 저장하여Black이 Bunny의 상태에 따라 유연하게 대처할 수 있도록 설계 되어 있다. 마지막으로 Web-Interface는 관리자에게 UI를 제공하는 역할을 하며 모니터링을 할 서버 및 그룹 등을 지정하여 할당할 수 있고, 사용자가 필요한 모니터링 metric을 선택, 추가, 생성할 수 있도록 구성되어 있으며, 클러스터 시스템 및 서버팜의 현재 상태에 대해 직관적으로 파악할 수 있는 인터페이스를 제공하고 있다. 또한 명령 콘솔을 통해 직접 Bunny에서 명령을 내릴 수 있다.

클러스터 모니터링 시스템은 서비스 중인 클러스터 시스템에서 수행되어야 하기 때문에 오버헤드를 최소화 해야한다. 본 클러스터 모니터링 시스템인 Black&Bunny를 16노드 클러스터(CPU 1.4 GHz, 2GB 메인 메모리, Opteron 2-way servers)에서 수행했을 경우, CPU 사용량은 모니터링 서버와 에이전트 모두 0.1%이하였으며, 메모리는 모니터링 서버는 2.0MB(0.1%), 모니터링 에이전트는 1.8MB(0.9%)를 사용하였다.

3. 결 론

최소 수백에서 수만대에 이르는 중대형 클러스터를 효율적으로 활용하기 위해서클러스터 시스템의 상태를 모니터링하고 시스템의 상태에 따는 능동적으로 대처할 수 있는 모니터링 시스템이 필수적이다. 이에 본 논문에서는 클러스터 시스템의 대표적인 모니터링 도구를 비교 및 분석하였고대규모 HPC 클러스터 시스템을 효율적으로 관리하기 위한 클러스터 모니터링 시스템을 설계 및 구현하였다.

향후 본 논문의 클러스터 모니터링 시스템을 바탕으로 모니터링 Metrics 추가, Job 스케줄러와 연동, Process level의 자원 모니터링 등을 비롯한 지속적인 기능 추가를 계획하고 있으며노드 수 증가에 따른 scalability 문제, 오버헤드 최소화 등에 초점을 맞추어 지속적으로 연구 및 개발할 계획이다.

참고문헌

[1] Matthew J. Sottile Ronald G. Minnich, "Supermon: A high-speed cluster monitoring system," Cluster '02, 2002.

[2] Robbert van Renesse, Kenneth P. Birman, and Werner Vogels. “Astrolabe: A Robust and Scalable Technology for Distributed System Monitoring, Management, and Data Mining,” ACM Transactions on Computer Systems, Vol. 21, No. 2, May 2003.

[3] Matthew L. Massie, Brent N. Chun, and David E. Culler, "The ganglia distributed monitoring system: design, implementation, and experience," Parallel Computing, Vol. 30, Issue 7, July 2004.