한국컴퓨터정보학회 하계학술대회 논문집 제18권 제2호 (2010. 7)

Hadoop 클러스터를 위한 모니터링 툴

금태훈*, 이원주°, 전창호*
*한양대학교 컴퓨터공학과,
°인하공업전문대학 컴퓨터정보과
e-mail: hotangs@gmail.com, wonjoo2@inhatc.ac.kr, chj5193@hanyang.ac.kr

Monitoring Tool for Hadoop Cluster

Taehoon Keum*, Won Joo Lee°, Chang-Ho Jeon*
*Dept. of Computer Science & Engineering, Hanyang University,
°Dept. of Computer Science, Inha Technical College

● 요 약 ●

최근 이슈가 되고 있는 클라우드 컴퓨팅은 다수의 노드를 이용한 클러스터를 사용한다. 이러한 클러스터를 효율적으로 관리하기 위해 모니터링 툴을 사용하고 있다. 하지만, 기존의 모니터링 툴은 클러스터를 구성하는 노드의 가용성과 오버헤드, 데이터 수집/전송 방식에 중심을 둔 모니터링 툴이기 때문에 클라우드 클러스터의 세부 정보까지 모니터링 할 수 없다. 따라서 본 논문에서는 클라우드 컴퓨팅을 구축할 수 있는 플랫폼인 Hadoop을 위한 모니터링 툴을 제안한다.

키워드: Hadoop 클러스터(Hadoop Cluster), 모니터링 툴(Monitoring Tool)

I. 서론

일반적인 클러스터를 위한 모니터링 툴은CARD, PARMON, Ganglia[1-3] 등이 있다. 이러한 모니터링 툴들은 클러스터를 구성하는 노드의 가용성과 오버헤드, 데이터 수집/전송 방식에 중심을 둔 모니터링 툴이기 때문에 Hadoop의 상태 정보까지 모니터링 할 수 없다는 문제점이 있다. 따라서 Hadoop 클러스터를 위한 모니터링 툴은 HDFS(Hadoop Distributed File System)의 정보까지 모니터링 할 수 있어야 한다.

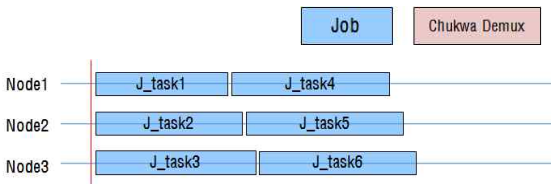
본 논문에서는 Hadoop[4] 클러스터를 위한 기존의 모니터링 툴이 가진 문제점을 지적하고, 이를 해결하기 위한 모니터링 툴을 제안한다.

II. 관련 연구

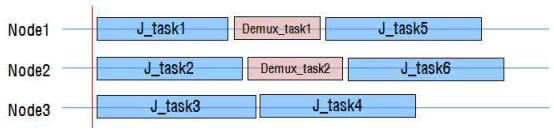
Hadoop의 세부 정보까지 모니터링 할 수 있는 툴로는 Chukwa[5]와 Mochi[6]가 있다.

Chukwa는 기존의 모니터링 툴에서 제공하는 모든 노드의 상태 정보뿐만 아니라, HDFS및 MapReduce의 정보까지 모니터링 하는 툴이다. Chukwa는 모든 노드에서 생성하는 로그파일을 수집하여 하나의 아카이브파일로 HDFS에 저장하게 되고, 이를 MapRuduce를 통해 분석(DeMux) 후, 사용자에게 보여준다. 이러한 구조는 대용량 Hadoop 클러스터에서 발생한 로그 파일을 분석하는데 효율적이지만, 분석과정에서 Hadoop의 MapReduce를 사용함으로써 Chukwa와 Hadoop이 높은 결합도를 가지게 되고, 분석과정의 오버헤드가 Hadoop의 성능에 영향을 끼친다. 그림 1은 Chukwa의 분석과정이 태스크로 등록되어 Hadoop의 성능에 영향을 주는 것을 보여주고 있다.

Mochi는 Hadoop 애플리케이션 디버깅을 위한 모니터링 툴로써 MapReduce의 로그 파일을 수집하고 노드별로 수행한 Task를 Timeline 형태로 보여준다. Chukwa와 달리 로그파일의 분석을 Hadoop을 통해 하지 않기 때문에 Hadoop과 결합도가 낮지만, 노드의 정보를 수집하지 않음으로써 하드웨어의 오버헤드와 관련된 애플리케이션 디버깅에는 적합하지 않다.



(a) 애플리케이션만 실행될 경우



(b) 애플리케이션 실행 도중, Chukwa 분석작업이 실행될 경우

그림 1. Chukwa 분석작업에 따른 오버헤드

이처럼 Hadoop 클러스터를 위한 모니터링 툴은 Hadoop의 작업에 영향을 주지 않아야 하며, Hadoop 애플리케이션의 자세한 분석을 위해 노드의 상태까지 보여줄 수 있어야 한다.

III. Hadoop 클러스터를 위한 모니터링 툴

본 논문에서 제안하는 모니터링 툴의 구조는 그림 2와 같다. 그림 2에서 Collector와 Monitor는 클라이언트-서버 방식으로 통신한다. Monitor는 Hadoop 클러스터를 구성하고 있는 Name Node와 Data Node 모두에 설치한다. 그리고 Collector는 Hadoop 클러스터에 포함된 노드가 아닌, 가용성이 높은 별도의 모니터링 서버에 설치한다. Monitor는 Hadoop의 로그파일과 함께 ps, cpuinfo 등의 시스템 명령을 이용하여 노드의 정보를 수집하여 Collector로 전송한다. Collector는 전송받은 정보와 로그파일을 분석하여 데이터베이스에 저장한다.

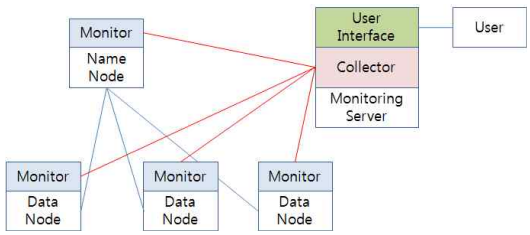


그림 2. 제안하는 모니터링 툴의 구조

User Interface는 사용자가 모니터링 정보를 쉽게 파악할 수 있도록 모든 노드의 정보를 가용성에 따른 색상으로 표현한다. 또한 Hadoop의 작업정보를 열람할 경우, 해당 작업에 관련된 태스크의 하드웨어 오버헤드를 쉽게 파악할 수 있도록 그림 3과 같이 태스크 내 CPU 사용률, 메모리 사용률 등의 하드웨어 정보를 함께 보여준다.

이처럼 제안하는 모니터링 툴은 별도로 구축된 모니터링 서버가 수집된 데이터를 분석함으로써 Hadoop의 성능에 영향을 미치지 않으며, 기존의 모니터링 툴보다 결합도를 낮출 수 있다. 또한, 수집된 정보를 표현할 때 작업에 참여한 태스크의 정보와 그 태스크를 수행한 노드의 오버헤드를 함께 보여줌으로써 Hadoop 애플리케이션 디버깅에 더욱 유용하게 사용될 수 있다.

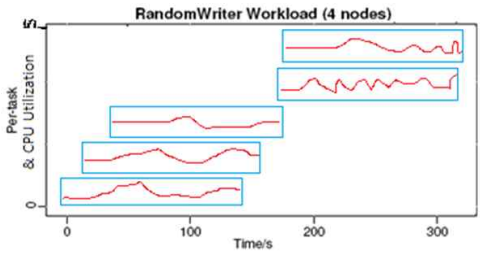


그림 3. 각 태스크의 수행시간과 해당 태스크를 수행한 노드 정보의 일괄적인 표현

IV. 결론 및 향후 연구 방향

본 논문에서는 Hadoop 클러스터를 위한 모니터링 툴을 제안하였다. 기존의 모니터링 툴보다 결합도가 낮으며, Task 정보와 노드의 오버헤드를 일괄적으로 보여줌으로써 Hadoop 애플리케이션 디버깅에 더욱 유용하게 사용될 수 있다. 또한, 향후 연구로 본 논문에서 제안한 모니터링 툴을 구현할 것이다.

참고문헌

[1] E. Anderson and D. Patterson, “Extensible, Scalable Monitoring for Clusters of Computers,” Proceedings of the 11th USENIX conference on System administration., pp. 9-16, Oct. 1997.

[2] R. Buyya, “PARMON: a portable and scalable monitoring system for clusters,” International Journal on Software-Practice and Experience., pp. 30:1-17, June. 2000.

[3] M. L. Massie, B. N. Chun and D.E. Culler, “The Ganglia Distributed Monitoring System: Design, Implementation, and Experience,” In Parallel Computing, vol. 30, Issue. 7, pp. 817-840, Jul. 2004.

[4] Hadoop, *http://hadoop.apache.org, 2009*

[5] J. Boulon, A. Konwinski, R. Qi, A. Rabkin, E. Yang and M. Yang, “Chukwa: A large-scale monitoring system,” Proceeding of international conference on Cloud Computing and Its Applications, pp. 1-5, Oct. 2008.

[6] J. Tan, X. Pan, S. Kavulya, R. Gandhi and P. Narasimhan, “Mochi: Visualizing Log-Anlaysis Based Tools for Debugging Hadoop,” In USENIX Workshop on Hot Topics in Cloud Computing(HotCloud), SanDiego, CA, Jun. 2009.