

# 하둡 플랫폼 분석을 위한 DEVS 기반의 하둡 시뮬레이터 개발

김병수\*, 이순주\*, 김탁곤\*

## Development of DEVS based Hadoop Simulator for Hadoop Platform Anlysis

Byeong Soo Kim, Sun Ju Lee, Tag Gon Kim

### Abstract

Hadoop platform is a representative storing and managing platform for big data. Hadoop consists of distributed computing system called MapReduce, and distributed file system called HDFS(Hadoop Distributed File System). It is important to analyse the effectiveness according to the change of node constructions and several parameters. However, since it is hard to construct thousands of nodes and analyse the constructed system, simulation method is required to analyse the system. This paper proposes Hadoop simulator based on DEVS formalism which provides hierarchical and modular modeling. Hadoop simulator provides an experimental environment that is possible to change of various parameters. It is also possible to design input models reflecting the characteristics of Hadoop applications. In this paper, we validate Hadoop Simulator through the comparison with the Hadoop execution results and perform various experiments.

**Key Words** : 하둡, DEVS 형식론, 시뮬레이터 개발

### 1. 서론

최근 다양한 종류의 빅 데이터가 증가하는 추세에 따라 빅데이터를 분석하고 처리하며 활용하는 방안에 대한 관심 역시 증대되고 있다. 이러한 빅데이터를 저장, 관리하기 위한 대표적인 플랫폼으로 하둡이 있다[1]. 하둡은 분산 컴퓨팅을 위한 맵리듀스 프레임워크[2]와 분산 파일 시스템인 HDFS(Hadoop Distributed File System)[3]로 구성되어 있다. 하둡은 빅데이터를 처리하기 위해 통상 수백 대 이상의 노드들로 구성되며, 실제 노드들의 스펙이나 여러 가지 파라미터들에 따라 하둡 플랫폼이 가지는 성능을 분석하는 것이 중요하다. 하지만 실제로 수천 대 이상의 노드 구축 후에 이를 분석하는 것이 경제적, 그리고 공간상의 제약 등 쉽지 않기 때문에 시뮬레이션 기법을 통해 분석하는 것이 필요하다.

기존에도 이미 하둡 시뮬레이션에 대한 연구들이 진행되었지만 많은 연구들이 맵리듀스를 시뮬레이션 하는 것에 초점을 맞추어 개발되었고 HDFS와 함께 하둡 전체를 시뮬레이션을 진행한 연구는 부족하였다[4].

따라서 본 논문에서는 하둡을 구성하는 맵리듀스와 HDFS를 이산 사건 시스템 형식론(DEVS: Discrete Event System Formalism)을 이용하여 모델링하고 구현한다[5]. 제안하는 하둡 시뮬레이터는 하둡 플랫폼을 구성하는 컴포넌트들이 모듈리하게 구현되어 있으며 다양한 파라미터들을 입력으로 받을 수 있다. 또한 실제 하둡 플랫폼을 이용한 실험 결과를 통해 응용 프로그램의 특성을 반영하는 모델을 설계하여 하둡 시뮬레이터의 입력으로 사용한다.

### 2. 하둡 모델 설계

#### 2.1. 하둡 플랫폼 모델

HDFS에서는 네임 노드의 분산 저장 알고리즘을 통해 하나의 파일을 여러 노드로 분산하여 블록 단위로 저장하며, 데이터 노드를 통해 실제 블록을 받아 저장하고 요청하는 파일의 해당 블록을 읽어준다. 이렇게 저장되어 있는 빅데이터를 맵리듀스 알고리즘을 이용하여 처리하는데, 잡트래커에 의해 잡을 여러 태스크로 쪼개어 여러 노드로 할당하고 해당 노드의 태스크 트래커는 태스크를 수행하고 결과를 반환한다. 이 때 네임 노드와 잡 트래커는 마스터 노드에 존재하며, 데이터 노드와 태스크 트래커는 슬레이브 노드에 존재한다. 그렇기 때문에 실제 각각의 요소

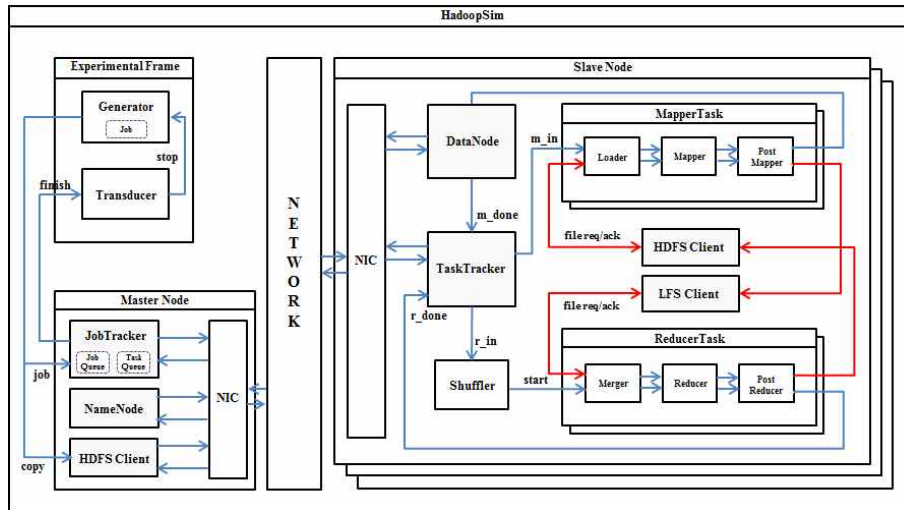


그림 1 하둡 시물레이터 결합 모델

들을 마스터 노드와 슬레이브 노드로 나누어 DEVS로 모델링하고, 그 사이에 네트워크 모델을 두어 노드들 사이의 데이터를 주고받는 역할을 하도록 모델링하였다.

## 2.2. 입력 모델 및 파라미터

입력 모델은 실제 하둡 수행 결과를 통한 맵 수행 시간, 셔플 수행 시간, 정렬 시간, 리듀스 시간 등을 포함하도록 설계하여 각 어플리케이션의 특성을 반영하며, 이 모델을 입력 받아 시물레이션을 진행한다. 시물레이터 파라미터로는 노드의 수, 각 노드에서 처리할 수 있는 맵과 리듀스 수, HDFS에서 블록 복제본의 수, 블록 사이즈 등을 설정해 줄 수 있도록 구현하였다.

## 3. 시물레이션 환경 및 실험 결과



그림 2 노드 수에 따른 실험 결과

설계된 DEVS 모델을 DEVS 실행 엔진인 DEVSsim++를 기반으로 구현하고 실험을 수행하였다. 실제 설치된 16대의 하둡 플랫폼을 이용하여 하둡 어플리케이션인 WordCount를 수행하고, 그 결과를 바탕으로 시물레이터를 검증하였다. 그림 2와 같이 실제 결과와 시물레이션 결과

가 유사함을 확인할 수 있고, 노드 수 증가에 따른 시물레이션 결과 또한 확인 할 수 있었다.

## 4. 결론

본 논문에서는 DEVS 형식론을 이용하여 하둡 플랫폼을 모델링하고 시물레이션 하였다. 응용 프로그램은 실제 하둡 결과를 이용하여 입력 모델로 설계하고, 이를 바탕으로 시물레이션 하여 실제 하둡과 유사한 결과가 나오는 것을 확인하고 검증하였다. 또한 노드의 수를 증가시켜가며 시물레이션을 함으로써 하둡 시물레이터의 확장성을 확인 할 수 있었다. 추후 연구로는 기존의 하둡 시물레이터를 바탕으로 하드웨어 모델들을 설계하고 구현하여 더욱 다양한 파라미터들에 대하여 실험 가능한 틀에 대해 연구하도록 한다.

## 감사의 글

본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었습니다(UD110006MD).

## 참고문헌

- [1] Apache Hadoop, <http://hadoop.apache.org>
- [2] Jeffrey Dean, Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", OSDI, 2004.
- [3] Sanjay Ghemawat, Howard Gobioff, Shun Tak Leung, "The Google File System", SOSP '03 on Operating systems principles, 2003.
- [4] Guanying Wang, Ali R. Butt, "A Simulation Approach to Evaluating Design Decisions in MapReduce Setups", MASCOTS '09, 2009.
- [5] Bernard P. Zeigler, Herbert Praehofer and Tag Gon Kim, Theory of Modelling and Simulation (2nd Edition), Academic Press, 2000.