

 (19) 대한민국특허청(KR) (12) 공개특허공보(A)	(11) 공개번호 10-2014-0056743 (43) 공개일자 2014년05월12일
(51) 국제특허분류(Int. Cl.) G06F 9/28 (2006.01) G06F 9/38 (2006.01) G06F 9/44 (2006.01) (21) 출원번호 10-2012-0122305 (22) 출원일자 2012년10월31일 심사청구일자 2012년10월31일	(71) 출원인 서강대학교산학협력단 서울특별시 마포구 백범로 35 (신수동, 서강대학교) (72) 발명자 박성용 서울 송파구 중대로 24, 229동 1103호 (문정동, 올림픽훼밀리타운) 이권용 서울특별시 마포구 신수동 167번지 메트로디오빌 2704호 (74) 대리인 특허법인충현

전체 청구항 수 : 총 12 항

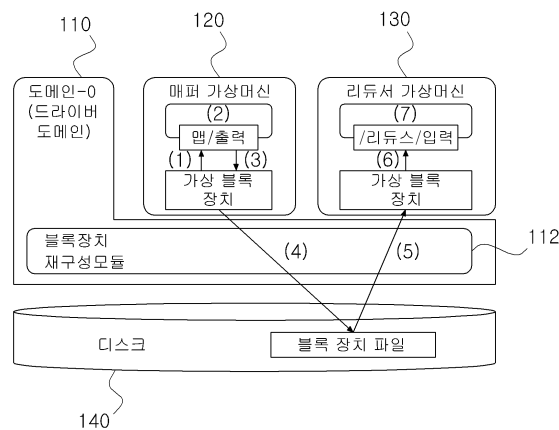
(54) 발명의 명칭 **가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법**

(57) 요약

본 발명은 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법에 관한 것으로, 보다 구체적으로는 맵퍼 가상머신에 의해 구동되는 맵퍼함수의 출력값을 포함하는 블록 장치 파일을 상기 맵퍼 가상머신으로부터 분리하여 리듀서 가상머신에 부착하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터를 포함한다.

이러한 구성에 의해, 본 발명의 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법은 맵퍼 함수의 출력값을 포함하는 블록 장치 파일을 맵퍼 가상머신으로부터 분리하여 리듀스 함수의 입력으로 사용되도록 리듀스 가상머신에 부착함으로써, 리듀서가 네트워크를 이용하여 맵퍼함수의 출력값에 접근할 필요가 없어지게 되어, 네트워크의 부하를 감소시킬 수 있는 효과가 있다.

대표도 - 도3



이 발명을 지원한 국가연구개발사업

과제고유번호	1415123192
부처명	지식경제부
연구사업명	대학 IT 연구센터 육성지원사업
연구과제명	소셜 미디어 서비스를 위한 클라우드 플랫폼 및 응용서비스 기술 개발
기 여 율	1/1
주관기관	건국대학교 산학협력단
연구기간	2012.01.01 ~ 2012.12.31

특허청구의 범위

청구항 1

가상 클라우드 환경 내 맵리듀스 클러스터에 있어서,

매퍼 가상머신에 의해 구동되는 매퍼함수의 출력값을 포함하는 블록 장치 파일을 상기 매퍼 가상머신으로부터 분리하여 리듀서 가상머신에 부착하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 2

제1항에 있어서,

상기 맵리듀스 클러스터는

상기 블록 장치 파일을 송수신하여 가상머신으로 전송하는 드라이버 도메인;

상기 블록 장치 파일에 대하여 임의의 키값을 읽어 이를 필터링하거나, 다른값으로 변환하는 맵 함수를 수행하는 매퍼 가상머신;

상기 키값을 기준으로 하여, 상기 블록 장치 파일을 정렬하고 병합하는 리듀스 함수를 수행하는 리듀서 가상머신;을 포함하며,

상기 드라이버 도메인은

상기 맵 함수의 출력단으로부터 블록 장치 파일을 분리한 후, 분리한 상기 블록 장치 파일을 상기 리듀스 함수의 입력단에 부착하는 블록 장치 재구성 모듈;

을 포함하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 3

제2항에 있어서,

상기 맵리듀스 클러스터가 하나의 노드에서 구성된 클러스터일 때, 블록 장치 파일의 재구성이 수행되는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 4

제1항에 있어서,

상기 맵리듀스 클러스터는

상기 블록 장치 파일에 대하여 임의의 키값을 읽어 이를 필터링 또는 다른값으로 변환하는 맵 함수를 수행하는 매퍼 가상머신;

상기 맵 함수의 출력값을 포함하는 블록 장치 파일을 입력받아 출력하는 매퍼 가상머신용 드라이버 도메인;

상기 가상머신용 드라이버 도메인으로부터 출력된 블록 장치 파일을 입력받아 리듀서 가상머신으로 전송하는 리듀서 가상머신용 드라이버 도메인; 및

상기 블록 장치 파일을 수신하여 키값을 기준으로 상기 블록 장치 파일을 정렬하고 병합하는 리듀스 함수를 수행하는 리듀서 가상머신;

을 포함하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 5

제4항에 있어서,

상기 매퍼 가상머신용 드라이버 도메인은

상기 블록 장치 파일을 마운트하고, 마운트된 위치의 콘텐츠를 상기 리듀서 가상머신용 드라이버 도메인으로 전송하는 제1 블록 장치 재구성 모듈;

을 더 포함하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 6

제4항에 있어서,

상기 리듀서 가상머신용 드라이버 도메인은

마운트된 블록 장치 파일을 마운트 해제하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 7

제6항에 있어서,

상기 리듀서 가상머신용 드라이버 도메인은

마운트 해제한 블록 장치 파일을 상기 리듀서 가상머신에 부착하는 제2 블록 장치 재구성모듈;

을 더 포함하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 8

제4항에 있어서,

상기 맵리듀스 클러스터가 다수의 노드에 걸쳐 구성된 클러스터일 때의 블록 장치 파일의 재구성이 수행되는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터.

청구항 9

가상 클라우드 환경 내 맵리듀스 클러스터 설계방법에 있어서,

매퍼 가상머신 내 매퍼가 마운트 포인팅 지점에 부착된 블록 장치를 마운트하는 단계;

상기 매퍼가 맵 태스크를 수행하고, 상기 마운트된 위치에 상기 맵 함수의 출력을 생성하는 단계;

상기 맵 태스크의 수행이 완료되면, 상기 매퍼는 상기 맵 함수의 출력이 생성된 위치에서 마운트를 해제하는 단계;

상기 드라이버 도메인 내 위치하는 블록 장치 재구성 모듈이 상기 매퍼 가상머신으로부터 블록 장치 파일을 분리하는 단계;

상기 블록 장치 재구성 모듈이 분리한 상기 블록 장치 파일을 리듀스 가상머신에 부착하는 단계;

상기 리듀서 가상머신 내 리듀서는 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트하는 단계; 및

상기 리듀서가 모든 맵 함수의 출력을 받고, 리듀스 함수를 수행하는 단계;

를 포함하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터의 데이터 병렬 처리 방법.

청구항 10

제9항에 있어서,

상기 리듀서가 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트하는 단계는

상기 리듀서가 상기 매퍼 가상머신의 모든 블록 장치 파일들이 마운트될 때까지 기다리는 것을 더 포함하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터의 데이터 병렬 처리 방법.

청구항 11

가상 클라우드 환경 내 맵리듀스 클러스터 설계방법에 있어서,

매퍼 가상머신 내 매퍼가 마운트 포인팅 지점에 부착된 블록 장치를 마운트하는 단계;

상기 매퍼가 맵 태스크를 수행하고, 상기 마운트된 위치에서 상기 맵 함수의 출력을 생성하는 단계;

상기 맵 태스크의 수행이 완료되면, 상기 매퍼는 상기 맵 함수의 출력이 생성된 위치에서 마운트를 해제하는 단계;

상기 드라이버 도메인 내 위치하는 블록 장치 재구성 모듈이 상기 매퍼 가상머신으로부터 블록 장치 파일을 분리하는 단계;

상기 매퍼 가상머신의 드라이버 도메인이 분리된 블록 장치 파일을 마운트하는 단계;

상기 블록 장치 재구성 모듈이 상기 블록 장치 파일이 마운트된 위치의 콘텐츠를 상기 리듀서 가상머신의 드라이버 도메인으로 전송하는 단계;

상기 리듀서 가상머신의 부분 드라이버 도메인이 마운트된 블록 장치 파일을 마운트 해제하는 단계;

상기 블록 장치 재구성 모듈이 상기 리듀서 가상머신의 입력단에 블록 장치 파일을 부착하는 단계;

상기 리듀서 가상머신 내 리듀서는 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트하는 단계; 및

상기 리듀서가 모든 맵 함수의 출력을 받고, 리듀서 함수를 수행하는 단계;

를 포함하는 것을 특징으로 하는 가상 클라우드 환경 내 맵리듀스 클러스터의 데이터 병렬 처리 방법.

청구항 12

제9항 내지 제11항 중 어느 한 항에 따른 방법을 컴퓨터로 실행하기 위한 프로그램이 기록된 컴퓨터 판독가능 기록매체.

명세서

기술분야

[0001] 본 발명은 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법에 관한 것으로, 특히 대용량 데이터 처리를 위한 맵리듀스 수행 시, 발생하는 네트워크 트래픽 양의 발생을 감소시켜 데이터 처리속도를 향상시킬 수 있는 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법에 관한 것이다.

배경기술

[0002] 급속도로 증가하는 데이터량으로 인하여, 데이터 병렬 또는 분산 처리를 위한 클러스터 컴퓨팅이 널리 사용되고 있으며, 이러한 클러스터 컴퓨팅 시스템은 다수의 물리 머신을 연결하여 이용하고 있다.

[0003] 최근에는, 가상머신 기반의 클라우드 서비스들이 출현함에 따라, 기존의 클러스터 시스템들이 점차 클라우드 상으로 이동하고 있다.

- [0004] 이처럼, 클라우드 상에서 동작하는 클러스터 시스템은 관리의 유용성, 신뢰성, 가용성, 그리고 클러스터 구성의 용이성 등의 장점을 갖는다. 하지만 이러한 클라우드 기반의 클러스터 시스템이 많이 이용되면서 분산 파일 시스템이나 통신 프로토콜과 같은 클러스터 시스템의 기반 기술들에 있어서, 병목현상이 발생하는 문제점이 발생했다.
- [0005] 상술한 바와 같이, 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법을 살펴보면 다음과 같다.
- [0006] 선행기술 1은 한국등록특허공보 제10-0980599호(2010.09.01)로서, 입출력 공유장치를 이용한 클라우드 서비스 시스템에 관한 것이다. 이러한 선행기술 1은 인터넷에 연결되어 데이터 통신을 수행하는 네트워크 스위치 허브; 네트워크 스위치 허브를 통해 인터넷과 데이터 통신을 수행하는 호스트 네트워크 인터페이스 카드를 구비하며 입출력 공유 장치의 제어에 따라 다수의 가상머신을 실행시키며 실행되는 다수의 가상머신이 호스트 네트워크 인터페이스 카드를 통해 인터넷의 다수의 사용자 단말과 통신을 수행하는 복수의 호스트 컴퓨터; 적어도 하나의 가상머신 이미지를 저장하는 저장장치; 및 네트워크 스위치 허브를 통해 인터넷과 데이터 통신을 수행하는 공유 네트워크 인터페이스 카드를 구비하며 인터넷에 연결된 사용자 단말의 요청에 따라 저장장치 내의 적어도 하나의 가상머신 이미지 중 하나를 선택하여 복수의 호스트 컴퓨터 중 어느 하나에서 실행시키는 입출력 공유장치를 포함한다.
- [0007] 또한, 선행기술 2는 한국공개특허공보 제10-2012-0071979호(2012.07.03)로서, 클라우드 컴퓨팅 시스템의 자원관리장치 및 방법에 관한 것이다. 이러한 선행기술 2는 사용자로부터 사용할 컴퓨팅 자원의 사용시점 및 자원사용량을 포함하는 개인 스케줄링 정보를 입력받아, 입력받은 개인 스케줄링 정보를 기초로 사용자가 사용할 가상머신이 생성될 위치를 배정하고, 사용시점에 자원사용량만큼 컴퓨팅 자원을 사용할 수 있는 가상머신을 배정된 위치에 생성하여 사용자에게 제공함으로써, 사용자의 시간대별로 사용할 컴퓨팅 자원에 관한 개인 스케줄링 정보를 고려하여, 컴퓨팅 자원의 할당 및 관리를 최적화할 수 있다.

발명의 내용

해결하려는 과제

- [0008] 상기와 같은 종래 기술의 문제점을 해결하기 위해, 본 발명은 대용량 데이터에 대한 맵리듀스 처리를 위해 발생하는 네트워크 트래픽의 양을 감소시켜 보다 빠른 데이터 분산 처리를 수행하도록 하는 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법을 제공하고자 한다.

과제의 해결 수단

- [0009] 위와 같은 과제를 해결하기 위한 본 발명의 한 실시 예에 따른 가상 클라우드 환경 내 맵리듀스 클러스터는 매퍼 가상머신에 의해 구동되는 맵 함수의 출력값을 포함하는 블록 장치 파일을 상기 매퍼 가상머신으로부터 분리하여 리듀서 가상머신에 부착하는 것을 특징으로 한다.
- [0010] 보다 바람직하게는 상기 블록 장치 파일을 송수신하여 가상머신으로 전송하는 드라이버 도메인; 상기 블록 장치 파일에 대하여 임의의 키값을 읽어 이를 필터링하거나, 다른값으로 변환하는 맵 함수를 수행하는 매퍼 가상머신; 상기 키값을 기준으로 하여, 상기 블록 장치 파일을 정렬하고 병합하는 리듀스 함수를 수행하는 리듀서 가상머신;을 포함하며, 상기 드라이버 도메인은 상기 맵 함수의 출력단으로부터 블록 장치 파일을 분리한 후, 분리한 상기 블록 장치 파일을 상기 리듀스 함수의 입력단에 부착하는 블록 장치 재구성 모듈;을 포함하는 맵리듀스 클러스터를 포함할 수 있다.
- [0011] 특히, 상기 맵리듀스 클러스터가 하나의 노드에서 구성되었을 때, 블록 장치 파일의 재구성이 수행될 수 있다.
- [0012] 보다 바람직하게는 상기 블록 장치 파일에 대하여 임의의 키값을 읽어 이를 필터링 또는 다른값으로 변환하는 맵 함수를 수행하는 매퍼 가상머신; 상기 맵 함수의 출력값을 포함하는 블록 장치 파일을 입력받아 출력하는 매퍼 가상머신용 드라이버 도메인; 상기 가상머신용 드라이버 도메인으로부터 출력된 블록 장치 파일을 입력받아 리듀서 가상머신으로 전송하는 리듀서 가상머신용 드라이버 도메인; 및 상기 블록 장치 파일을 수신하여 키값을 기준으로 상기 블록 장치 파일을 정렬하고 병합하는 리듀스 함수를 수행하는 리듀서 가상머신;을 포함하는 맵리듀스 클러스터를 포함할 수 있다.

- [0013] 보다 바람직하게는 상기 블록 장치 파일을 마운트하고, 마운트된 위치의 콘텐츠를 상기 리듀서 가상머신용 드라이버 도메인으로 전송하는 제1 블록 장치 재구성 모듈; 을 더 포함하는 매퍼 가상머신용 드라이버 도메인을 포함할 수 있다.
- [0014] 특히, 마운트된 블록 장치 파일을 분리하는 리듀서 가상머신용 드라이버 도메인을 포함할 수 있다.
- [0015] 보다 바람직하게는 분리한 블록 장치 파일을 상기 리듀서 가상머신에 부착하는 제2 블록 장치 재구성 모듈; 을 더 포함하는 리듀서 가상머신용 드라이버 도메인을 포함할 수 있다.
- [0016] 특히, 상기 맵리듀스 클러스터가 다수의 노드에 걸쳐 구성되었을 때의 블록 장치 파일의 재구성이 수행되는 것을 특징으로 한다.
- [0017] 위와 같은 과제를 해결하기 위한 본 발명의 다른 실시 예에 따른 가상 클라우드 환경 내 맵리듀스 클러스터 설계방법은 매퍼 가상머신 내 매퍼가 마운트 포인팅 지점에 부착된 블록 장치를 마운트하는 단계; 상기 매퍼가 맵 태스크를 수행하고, 상기 마운트된 위치에서 상기 맵 함수의 출력을 생성하는 단계; 상기 맵 태스크의 수행이 완료되면, 상기 매퍼는 상기 맵 함수의 출력이 생성된 위치에서 마운트를 해제하는 단계; 상기 드라이버 도메인 내 위치하는 블록 장치 재구성모듈이 상기 매퍼 가상머신으로부터 블록 장치 파일을 분리하는 단계; 상기 블록 장치 재구성 모듈이 분리한 상기 블록 장치 파일을 리듀스 가상머신에 부착하는 단계; 상기 리듀서 가상머신 내 리듀서는 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트하는 단계; 상기 리듀서가 모든 맵 함수의 출력을 받고, 리듀서 함수를 수행하는 단계;를 포함하는 것을 특징으로 한다.
- [0018] 보다 바람직하게는 상기 리듀서가 상기 매퍼 가상머신의 모든 블록 장치 파일들이 마운트될 때까지 기다리는 것을 더 포함하는 리듀서가 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트하는 단계를 포함할 수 있다.
- [0019] 위와 같은 과제를 해결하기 위한 본 발명의 다른 실시 예에 따른 가상 클라우드 환경 내 맵리듀스 클러스터 설계방법은 매퍼 가상머신 내 매퍼가 마운트 포인팅 지점에 부착된 블록 장치를 마운트하는 단계; 상기 매퍼가 맵 태스크를 수행하고, 상기 마운트된 위치에서 상기 맵 함수의 출력을 생성하는 단계; 상기 맵 태스크의 수행이 완료되면, 상기 매퍼는 상기 맵 함수의 출력이 생성된 위치에서 마운트를 해제하는 단계; 상기 드라이버 도메인 내 위치하는 블록 장치 재구성 모듈이 상기 매퍼 가상머신으로부터 블록 장치 파일을 분리하는 단계; 상기 매퍼 가상머신은 분리된 블록 장치 파일을 마운트하는 단계; 상기 블록 장치 재구성 모듈이 상기 블록 장치 파일이 마운트된 위치의 콘텐츠를 상기 리듀서 가상머신의 부분 드라이버 도메인으로 전송하는 단계; 상기 리듀서 가상머신의 부분 드라이버 도메인이 마운트된 블록 장치 파일을 분리하는 단계; 상기 블록 장치 재구성 모듈이 상기 리듀서 가상머신의 입력단에 블록 장치 파일을 부착하는 단계; 상기 리듀서 가상머신 내 리듀서는 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트하는 단계; 및 상기 리듀서가 모든 맵 함수의 출력을 받고, 리듀서 함수를 수행하는 단계;를 포함하는 것을 특징으로 한다.

발명의 효과

- [0020] 본 발명의 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법은 맵 함수의 출력값을 포함하는 블록 장치 파일을 매퍼 가상머신으로부터 분리하여 리듀스 함수로 입력되도록 리듀스 가상머신에 부착함으로써, 리듀서가 네트워크 전송을 통해 맵 함수의 출력값에 접근할 필요가 없어지게 되어, 네트워크의 부하를 감소시킬 수 있는 효과가 있다.
- [0021] 또한 본 발명의 가상 클라우드 환경 내 맵리듀스 클러스터 및 이를 위한 설계방법은 가상머신들간에 자원 경쟁으로 인해, 데이터 복제 시, 성능 병목현상이 발생하는 것을 미연에 방지할 수 있는 효과가 있다.

도면의 간단한 설명

- [0022] 도 1은 맵리듀스 프레임워크를 나타낸 도면이다.
- 도 2는 일반적인 가상머신 기반의 클라우드 노드를 나타낸 도면이다.
- 도 3은 본 발명의 일 실시 예에 따른 하나의 노드에서 구성된 클러스터에 대한 구조를 나타낸 도면이다.
- 도 4는 본 발명의 다른 실시 예에 따른 다수의 노드에 걸쳐 구성된 클러스터에 대한 구조를 나타낸 도면이다.

도 5는 블록 장치 재구성 모듈이 없는 경우의 단어 계수 맵리듀스의 실행시간을 나타낸 그래프이다.

도 6은 블록 장치 재구성 모듈이 있는 경우의 단어 계수 맵리듀스의 실행시간을 나타낸 그래프이다.

도 7은 두 개의 물리노드를 교차하는 가상머신간 데이터 전송시간을 나타낸 그래프이다.

도 8은 리듀서 가상머신이 탑재된 노드의 가상머신들의 CPU 활용도를 나타낸 누적 그래프이다.

도 9는 리듀서 가상머신이 탑재되지 않은 노드의 가상머신들의 CPU 활용도를 나타낸 누적 그래프이다.

도 10은 리듀서 가상머신이 탑재된 노드의 가상머신들의 디스크 입출력 지연대기시간을 나타낸 누적 그래프이다.

도 11은 리듀서 가상머신이 탑재되지 않은 노드의 가상머신들의 디스크 입출력 지연대기시간을 나타낸 누적 그래프이다.

발명을 실시하기 위한 구체적인 내용

[0023] 이하, 본 발명을 바람직한 실시 예와 첨부한 도면을 참고로 하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며, 여기에서 설명하는 실시 예에 한정되는 것은 아니다.

[0024] 먼저, 본 발명에 대해 설명하기에 앞서, 대용량 데이터의 분산 또는 병렬 처리를 수행하기 위해 사용되는 맵리듀스(MapReduce) 알고리즘에 대하여 자세히 살펴보도록 한다.

[0025] 상기 맵리듀스(MapReduce)알고리즘은 구글(Google)이 공개한 기술로서, 저가의 컴퓨터로 구성된 클러스터 환경에서 페타바이트 이상의 대용량 데이터를 병렬로 처리하기 위한 분산 프로그래밍 모델을 나타낸다. 이에 따라, 맵리듀스의 오픈소스 구현체인 하둡(Hadoop)을 중심으로 아마존, 링크드인, 페이스북과 같은 대용량 데이터를 다루는 서비스에서 맵리듀스 모델을 채용하고 있으며 대용량 데이터를 처리하는 많은 시스템에서 주목을 받고 있다.

[0026] 도 1은 맵리듀스 프레임워크를 나타낸 도면이다.

[0027] 도 1에 도시된 바와 같이, 맵리듀스 수행 시 파일 시스템은 큰 파일을 블록 단위로 분할하여 맵(Map) 함수를 수행하는 매퍼(Mapper) 태스크들에 할당한다. 맵 함수는 임의의 키-값 쌍을 읽어 이를 필터링하거나 다른 값으로 변환하는 작업을 수행한다. 맵 함수의 수행 결과는 매퍼가 수행된 노드의 로컬 디스크에 기록되며 모든 매퍼가 작업을 종료하면 맵리듀스의 스케줄러가 리듀서(Reducer) 태스크를 수행한다. 리듀서는 여러 노드들로부터 키 값을 기준으로 각 리듀서가 처리해야 할 키-값 쌍을 이전 매퍼들의 로컬 디스크로부터 읽어온다. 이후 키(key) 값을 기준으로 정렬 연산을 수행하고 이렇게 정렬된 키-값 쌍 등을 동일 키 값을 기준으로 병합 연산을 수행하여 그 결과를 출력한다.

[0028] 도 2는 일반적인 가상머신 기반의 클라우드 노드를 나타낸 도면이다.

[0029] 도 2에 도시된 바와 같이, 가상머신 기반의 클라우드 환경에서, 하나의 동일한 물리 노드 상에 가상머신이 네트워크를 통해 각각 통신할 때, 상기 가상머신은 드라이버 도메인을 교차하여 복사하도록 하고, 또한 다른 물리 노드와 교차하는 곳에 위치하는 가상머신들 간의 통신은 각 노드의 드라이버 도메인에 의해 중개되어 이루어진다. 이러한 중개과정의 수행 시, 입출력 가상화 오버헤드와, 가상머신들간에 입출력 간섭이 발생된다.

[0030] 가상화된 클라우드에서의 맵리듀스 클러스터는 상기 가상머신들을 포함하여 수행하는데, 각 가상머신은 매퍼 가상머신 또는 리듀서 가상머신을 통해, 개별적인 맵 또는 리듀스 태스크를 수행한다.

[0031] 이와 같이, 가상화된 클라우드 환경에서 맵리듀스 클러스터는 디바이스 드라이버의 입출력 중개와, 가상머신들간에 자원 경쟁으로 인한 병목현상이 발생한다. 특히, 매퍼와 리듀서간 데이터 전달 시, 많은 양의 네트워크 트래픽이 발생하여 맵 리듀스 처리량을 감소시키게 된다.

[0032] 도 3은 본 발명의 일 실시 예에 따른 하나의 노드에서 구성된 클러스터에 대한 구조를 나타낸 도면이다.

[0033] 도 3에 도시된 바와 같이, 본 발명의 일 실시 예에 따른 하나의 노드에서 구성된 클러스터는 드라이버 도메인(110), 매퍼 가상머신(120), 리듀서 가상머신(130)을 포함한다.

[0034] 매퍼 가상머신(120)은 상기 블록 장치 파일에 대하여 임의의 키값을 읽어 이를 필터링하거나, 다른값으로 변환

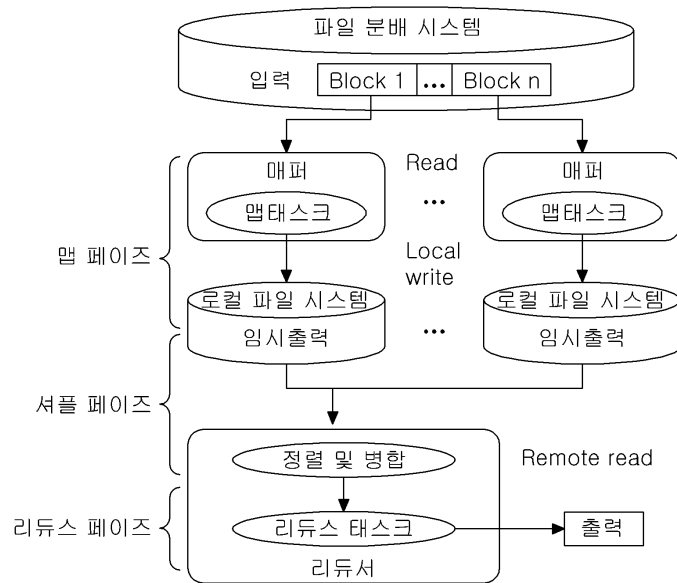
하는 맵 함수를 수행한다.

- [0035] 리듀서 가상머신(130)은 상기 키값을 기준으로 하여, 상기 블록 장치 파일을 정렬하고 병합하는 리듀스 함수를 수행한다.
- [0036] 드라이버 도메인(110)은 매퍼 가상머신의 상기 블록 장치 파일을 리듀서 가상머신으로 전달한다. 이러한 드라이버 도메인은 상기 맵 함수의 출력단으로부터 블록 장치 파일을 분리한 후, 분리한 상기 블록 장치 파일을 상기 리듀스 함수의 입력단에 부착하는 블록 장치 재구성 모듈(112)을 포함한다.
- [0037] 이러한 맵리듀스 클러스터의 구조를 설계하는 방법은 다음과 같다. 먼저, 매퍼 가상머신 내 매퍼가 마운트 포인팅 지점에 부착된 블록 장치를 마운트하고(1), 상기 매퍼가 맵 태스크를 수행하고, 상기 마운트된 위치에서 상기 맵 함수의 출력을 생성한다(2). 이후, 상기 맵 태스크의 수행이 완료되면, 상기 매퍼는 상기 맵 함수의 출력이 생성된 위치에서 마운트를 해제하고(3), 상기 드라이버 도메인 내 위치하는 블록 장치 재구성 모듈이 상기 매퍼 가상머신으로부터 블록 장치 파일을 분리한다(4). 이처럼, 상기 블록 장치 재구성 모듈이 분리한 상기 블록 장치 파일을 리듀스 가상머신에 부착한다(5).
- [0038] 이에 따라, 상기 리듀서 가상머신 내 리듀서는 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트하고(6), 상기 리듀서가 모든 맵 함수의 출력을 받아, 리듀서 함수를 수행한다(7).
- [0039] 도 4는 본 발명의 다른 실시 예에 따른 다수의 노드에 걸쳐 구성된 클러스터에 대한 구조를 나타낸 도면이다.
- [0040] 도 4에 도시된 바와 같이, 본 발명에 따른 다수의 노드에 걸쳐 구성된 클러스터는 매퍼 가상머신(220), 매퍼 가상머신용 드라이버 도메인(210), 리듀서 가상머신용 드라이버 도메인(240) 및 리듀서 가상머신(250)을 포함한다.
- [0041] 매퍼 가상머신(220)은 블록 장치 파일에 대하여 임의의 키값을 읽어 이를 필터링 또는 다른값으로 변환하는 맵 함수를 수행한다.
- [0042] 매퍼 가상머신용 드라이버 도메인(210)은 상기 맵 함수의 출력값을 포함하는 블록 장치 파일을 입력받아 출력한다. 이러한 매퍼 가상머신용 드라이버 도메인(210)은 상기 블록 장치 파일을 마운트하고, 마운트된 위치의 콘텐츠를 상기 리듀서 가상머신용 드라이버 도메인으로 전송하는 제1 블록 장치 재구성 모듈(212)을 포함할 수 있다.
- [0043] 리듀서 가상머신용 드라이버 도메인(240)은 상기 매퍼 가상머신용 드라이버 도메인으로부터 출력된 블록 장치 파일을 입력받아 리듀서 가상머신으로 전송하며, 마운트된 블록 장치 파일을 마운트 해제한다. 이러한 리듀서 가상머신용 드라이버 도메인(240)은 마운트 해제한 블록 장치 파일을 상기 리듀서 가상머신에 부착하는 제2 블록 장치 재구성 모듈(242)을 포함할 수 있다.
- [0044] 리듀서 가상머신(250)은 상기 블록 장치 파일을 수신하여 키값을 기준으로 상기 블록 장치 파일을 정렬하고 병합한다.
- [0045] 이러한 다수의 노드에 걸쳐 구성된 클러스터를 설계하는 방법은 다음과 같다. 먼저, 매퍼 가상머신 내 매퍼가 마운트 포인팅 지점에 부착된 블록 장치를 마운트한다(1). 상기 매퍼가 맵 태스크를 수행하고, 상기 마운트된 위치에서 상기 맵 함수의 출력을 생성한다(2).
- [0046] 상기 맵 태스크의 수행이 완료되면, 상기 매퍼는 상기 맵 함수의 출력이 생성된 위치에서 마운트를 해제한다(3).
- [0047] 상기 드라이버 도메인 내 위치하는 블록 장치 재구성 모듈이 상기 매퍼 가상머신으로부터 블록 장치 파일을 분리한다(4).
- [0048] 상기 매퍼 가상머신은 분리된 블록 장치 파일을 마운트한다(5).
- [0049] 상기 블록 장치 재구성 모듈이 상기 블록 장치 파일이 마운트된 위치의 콘텐츠를 상기 리듀서 가상머신의 부분 드라이버 도메인으로 전송한다(6).
- [0050] 상기 리듀서 가상머신의 부분 드라이버 도메인이 마운트된 블록 장치 파일을 마운트 해제한다(7).
- [0051] 상기 블록 장치 재구성 모듈이 상기 리듀서 가상머신의 입력단에 블록 장치 파일을 부착한다(8).
- [0052] 상기 리듀서 가상머신 내 리듀서는 상기 블록 장치 파일을 리듀스 함수의 입력위치로 마운트한다(9).

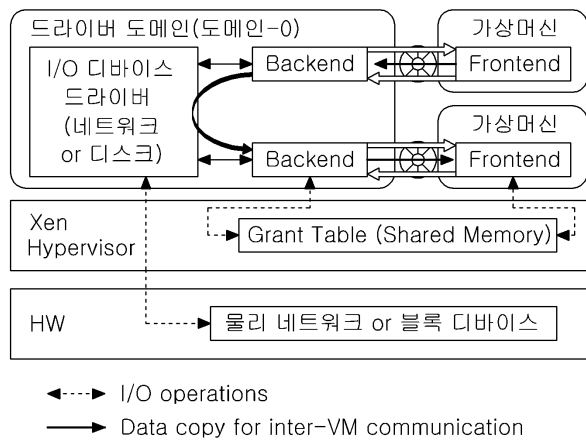
- [0053] 상기 리듀서가 모든 맵 함수의 출력을 받아, 리듀서 함수를 수행한다(10).
- [0054] 이하에서는 본 발명에 따른 가상 클라우드 환경 내 맵리듀스 클러스터를 이용하여 데이터 분산 및 병렬 처리를 수행하는 실험에 대하여 자세히 살펴보도록 한다. 실험조건으로는 두 개의 물리 머신과 두 개의 intel xeon 2.3GHz 쿼드코어 프로세서 및 16GB 메모리를 사용하며, 이때 상기 물리 머신은 Ubuntu 11.10과 커널 3.0.0-12 및 Xen hypervisor 4.1.1에서 구동한다. 또한, 각각의 물리 머신은 7개의 가상머신을 포함하고, 각각의 가상머신은 1개의 VCPU와 1GB 메모리를 포함한다.
- [0055] 특히, 본 실험은 xen의 기본 관리 툴인 xm 툴을 사용하여, 블록 장치를 분리 및 부착하도록 한다.
- [0056] 또한, "윌리엄 셰익스피어 전집"을 테스트 데이터로서 사용하여, 약 300MB 테스트 데이터를 300회 복제하여 생성한다. 이때, 상기 데이터는 각 맵퍼의 입력으로 사용된다. 예를 들어, 맵퍼가 두 개와 세 개가 존재한다면, 전체 입력 데이터의 크기는 각각 600MB 및 900MB이다. 이때, 맵함수의 출력 크기는 약 400MB이고, 상기 400MB 데이터는 각 맵퍼로부터 리듀서로 전송된다. 이러한 실험과정을 10번 반복수행하였고, 이후 실험결과의 평균을 연산하여 맵리듀스의 시행횟수에 대한 그래프를 작성하였다.
- [0057] 도 5는 블록 장치 재구성 모듈을 사용하지 않는 경우의 단어 계수 맵리듀스의 실행시간을 나타낸 그래프이고, 도 6은 본 발명에 따른 블록 장치 재구성 모듈을 사용하는 경우의 단어 계수 맵리듀스의 실행시간을 나타낸 그래프이다.
- [0058] 맵리듀스 클러스터가 하나의 물리노드의 가상머신에서 구동될 때, 상기 가상머신 중 하나의 가상머신이 리듀스 태스크를 수행하고, 다른 가상머신들이 맵 태스크를 수행한다.
- [0059] 도 5 내지 도 6에 도시된 바와 같이, 맵퍼 가상머신의 수와 입력 데이터의 전체 크기가 증가할수록 데이터 전송 시간이 이에 비례하여 증가하는 것을 알 수 있다. 이에 따라, 맵퍼 가상머신에서 네트워크 전송이 많아지거나, 데이터 전송을 위한 디스크 입출력이 시행될 때, 드라이버 도메인 상에서 입출력 간섭이 발생하는 것을 알 수 있다.
- [0060] 하지만 본 발명에 따라, 블록 장치 재구성을 사용하는 경우, 맵퍼로부터 리듀서로의 맵 함수의 출력전송 소요시간이 급격하게 감소하는 것을 알 수 있다. 특히, 도 5와 비교하여 살펴보면, 맵퍼의 수와 입력 데이터의 전체 크기가 증가할 때, 데이터 전송시간이 거의 일정하게 유지되는 것을 알 수 있다.
- [0061] 또한, 다수의 노드에 걸쳐 구성된 클러스터를 이용한 데이터 전송시간을 평가할 수 있다.
- [0062] 도 7은 두 개의 물리노드에서 구성된 클러스터의 가상머신간 데이터 전송시간을 나타낸 그래프이다.
- [0063] 데이터 전송결과를 평가하기 위해, 맵리듀스 클러스터가 다수의 물리노드 상에서 구성되어 구동할 때, 하나의 가상머신은 리듀스 태스크를 수행하고, 물리노드 각각에서 세 개의 가상머신은 맵 태스크를 수행하며, 나머지 가상머신들(이들을 D가상머신이라고 한다.)은 일반적인 맵 테스트와 유사한 디스크 입출력 구동을 수행한다. 이러한 D가상머신들은 맵리듀스 클러스터에 포함되지 않고, dbench를 수행한다. 이때, 상기 dbench는 디스크 입출력 벤치마크 툴이고, 디폴트 구성을 사용하였으며, 하나의 dbench 클라이언트를 이용하였다.
- [0064] 도 7에 도시된 바와 같이, 맵퍼 가상머신의 제1 그룹은 리듀서 가상머신과 동일한 물리노드에 위치하고, 맵퍼 가상머신의 제2 그룹은 다른 물리노드에서 구동된다. 따라서, 제2 그룹 내 가상머신들은 제1 그룹에서 블록 장치 파일들이 분리되고 부착되는 동안, 드라이버 도메인간에 네트워크를 통해 맵 함수의 출력이 전송된다.
- [0065] 이에 따라, 제1 및 제2 그룹에서 작업속도가 향상되는 것을 알 수 있으며, 본 발명에 따른 블록 장치 재구성에 의해, 제1 그룹의 맵퍼 가상머신의 데이터 전송이 약 97%가 향상되었다. 또한, 다른 물리 노드에서 구동되는 제2 그룹의 맵퍼 가상머신의 데이터 전송이 약 15% 향상된 것을 알 수 있다.
- [0066] 도 8은 리듀서 가상머신이 탑재된 노드의 가상머신들의 CPU 활용도를 나타낸 누적 그래프이고, 도 9는 리듀서 가상머신이 탑재되지 않은 노드의 가상머신들의 CPU 활용도를 나타낸 누적 그래프이다.
- [0067] 이처럼, 도 8 내지 도 9에 도시된 두 개의 그래프는 세 개의 페이지로 구성되는데, 본 발명에 따라 블록 장치 재구성을 사용하는 첫 번째 전송 페이지와, 블록 장치 재구성을 사용하지 않는 두 번째 전송 페이지, 그리고 나머지 안정적 페이지를 나타낸다.
- [0068] 도 8 내지 도 9에 도시된 바와 같이, 블록 장치 재구성을 갖는 데이터 전송의 경우에는 블록 장치 재구성이 이루어지지 않는 데이터 전송보다 CPU 자원을 덜 사용되는 것을 알 수 있다. 이러한 현상은 블록 장치 재구성을

도면

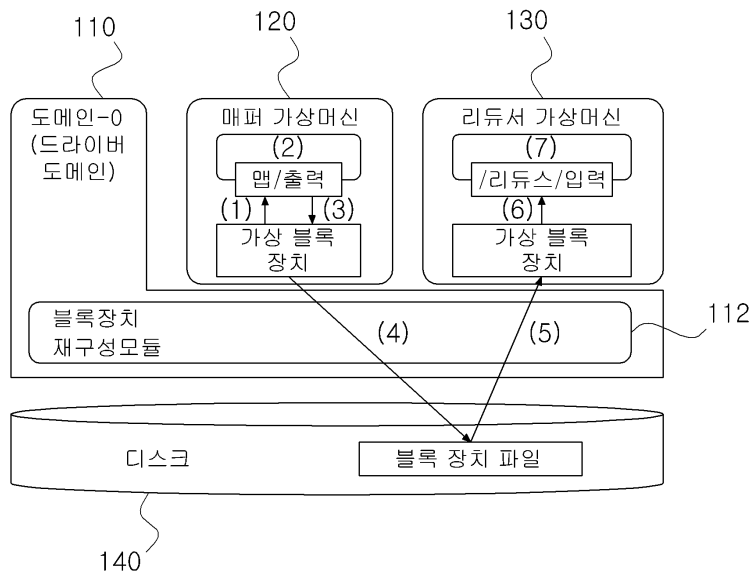
도면1



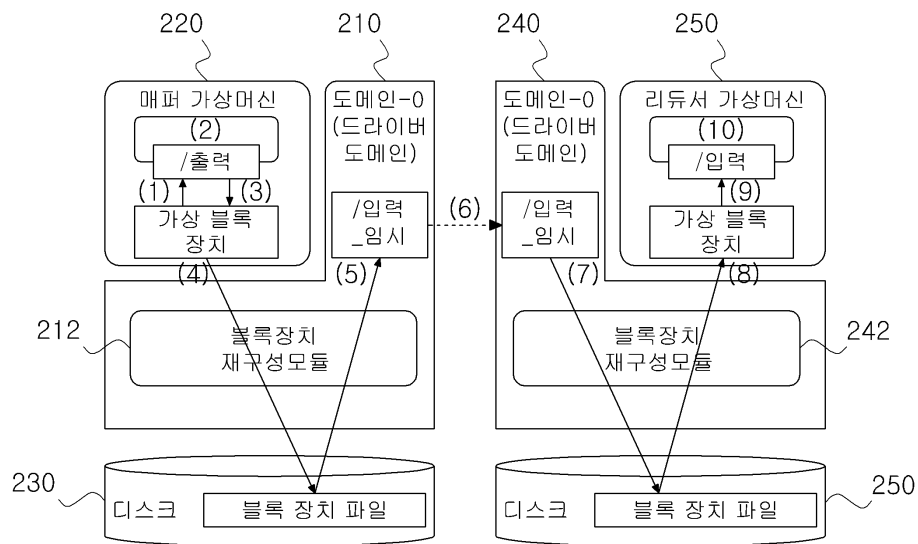
도면2



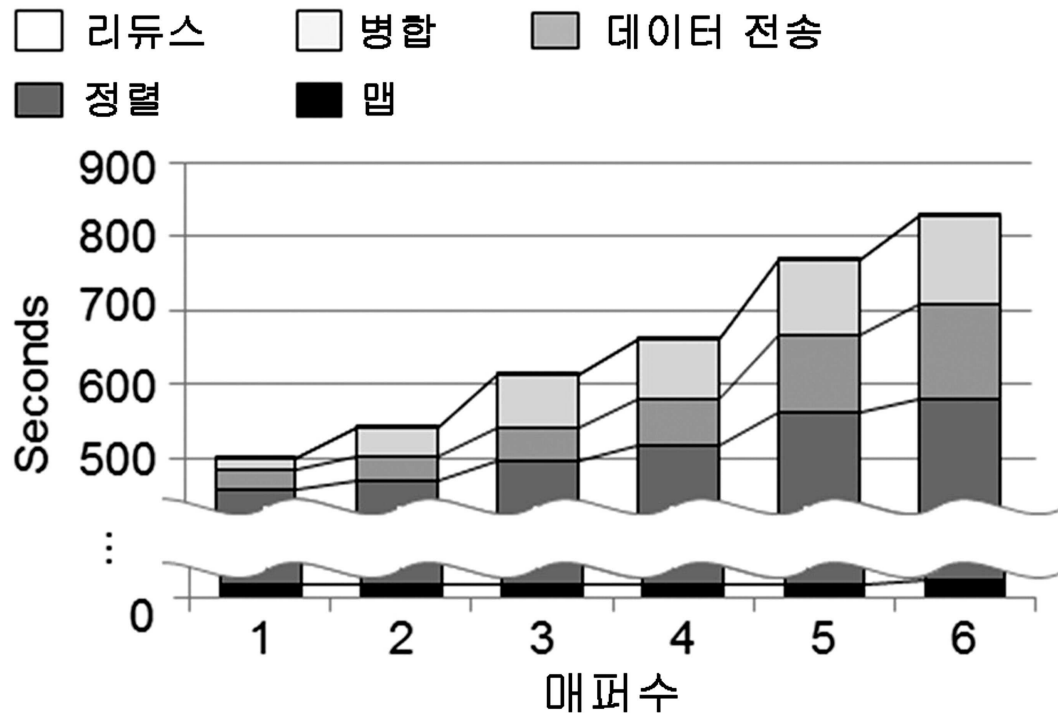
도면3



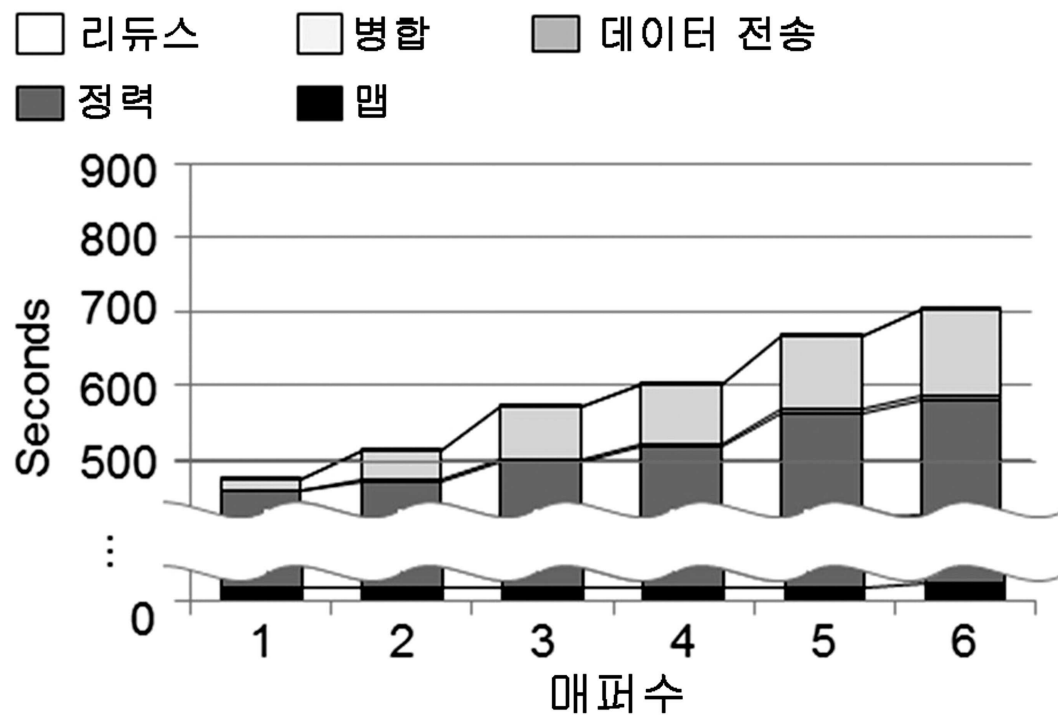
도면4



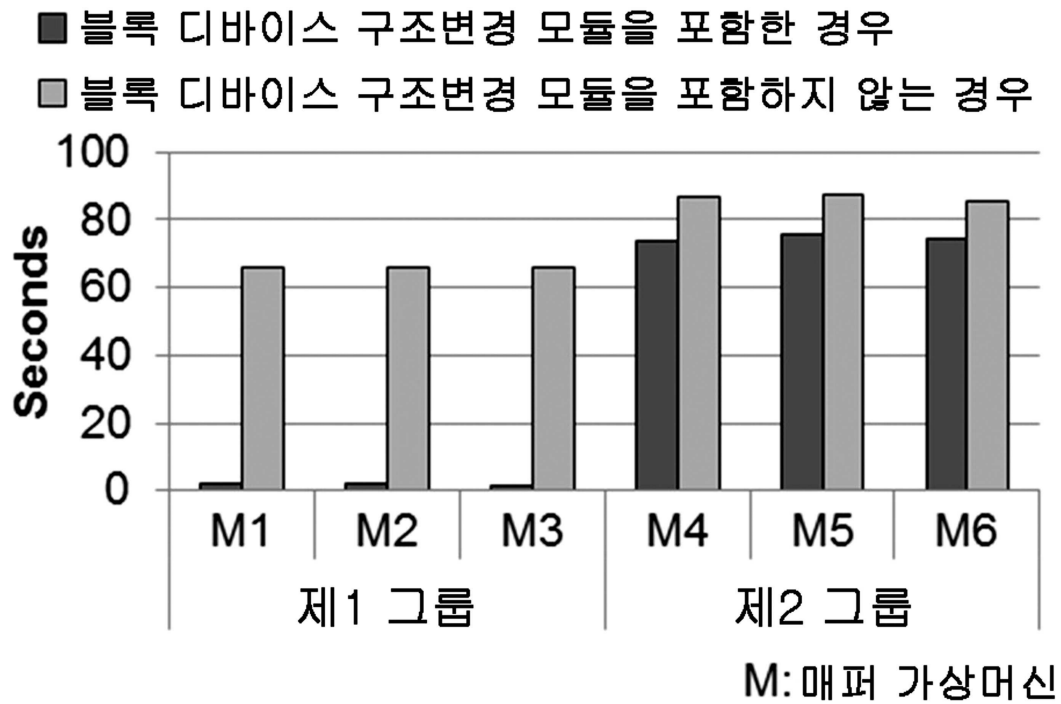
도면5



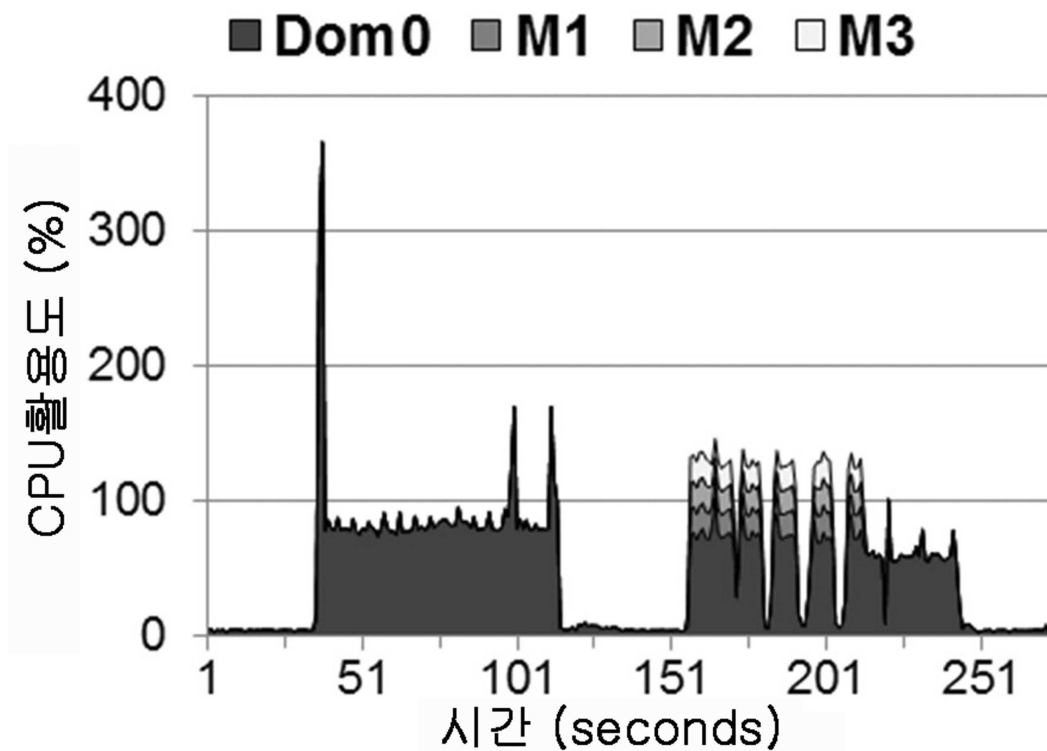
도면6



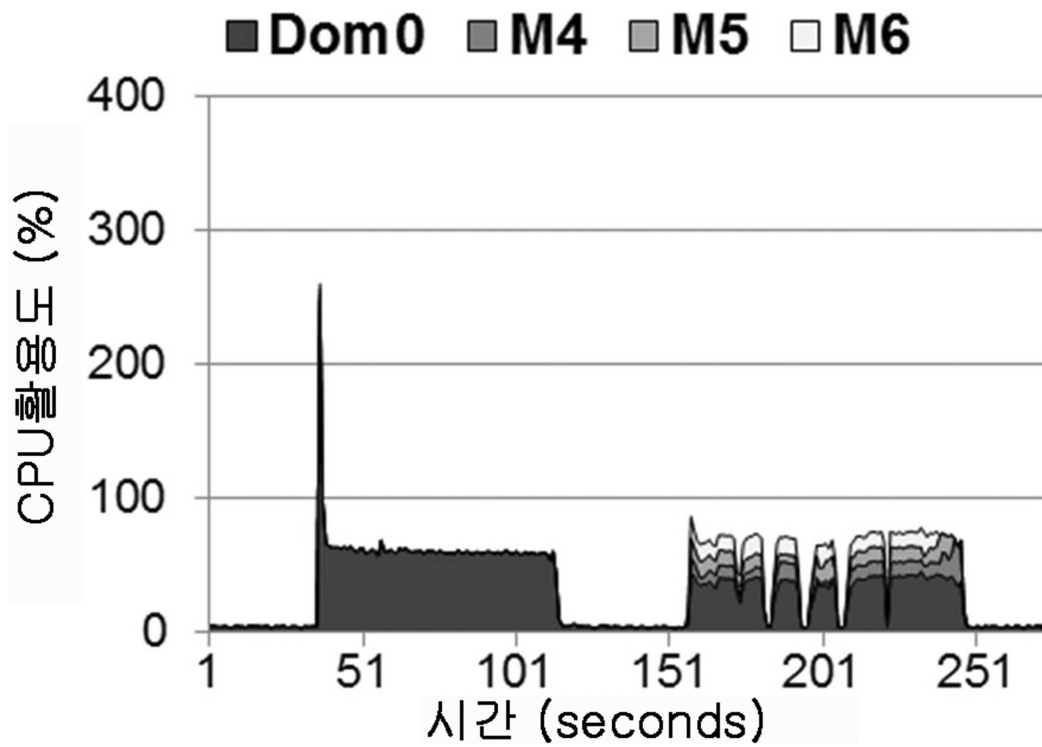
도면7



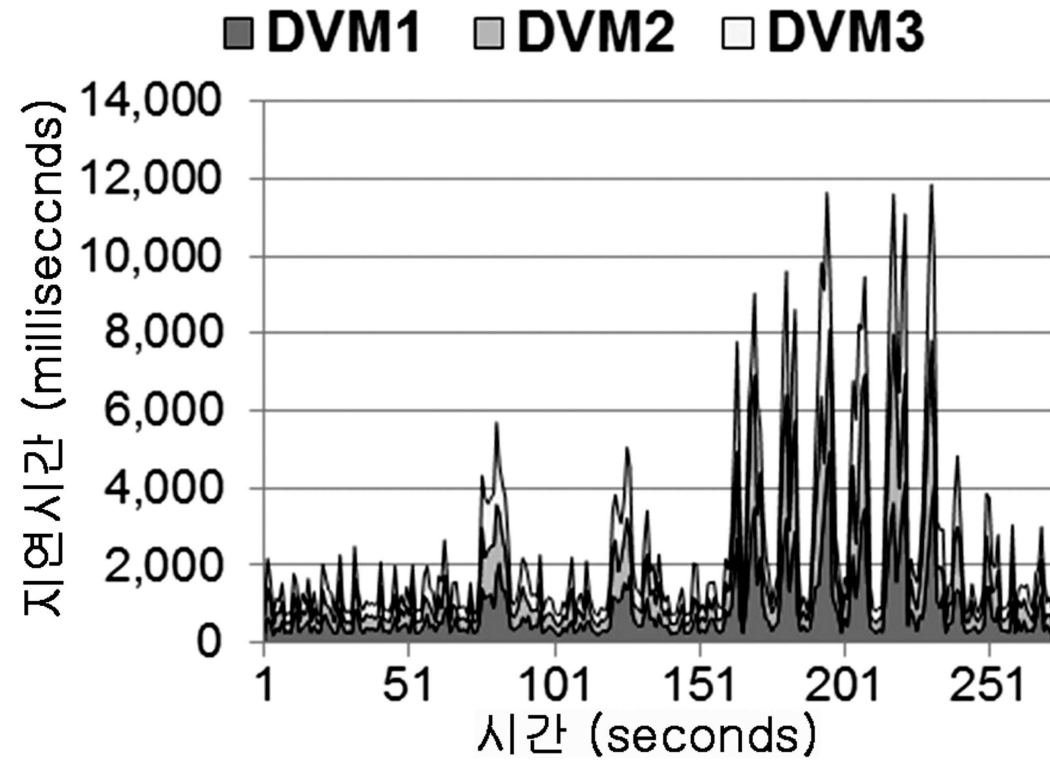
도면8



도면9



도면10



도면11

