



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2018년09월17일
(11) 등록번호 10-1899719
(24) 등록일자 2018년09월11일

(51) 국제특허분류(Int. Cl.)
G06F 9/455 (2018.01) G06N 99/00 (2010.01)
(52) CPC특허분류
G06F 9/45558 (2013.01)
G06N 99/005 (2013.01)
(21) 출원번호 10-2017-0053459
(22) 출원일자 2017년04월26일
심사청구일자 2017년04월26일
(56) 선행기술조사문헌
KR1020090063852 A*
(뒷면에 계속)

(73) 특허권자
세종대학교산학협력단
서울특별시 광진구 능동로 209 (군자동, 세종대학교)
(72) 발명자
노재춘
서울특별시 광진구 능동로 209, 율곡관 302호 컴
퓨터공학과 학과사무실 (군자동, 세종대학교)
김태훈
서울특별시 광진구 능동로 209, 율곡관 302호 컴
퓨터공학과 학과사무실(군자동, 세종대학교)
(뒷면에 계속)
(74) 대리인
민영준

전체 청구항 수 : 총 6 항

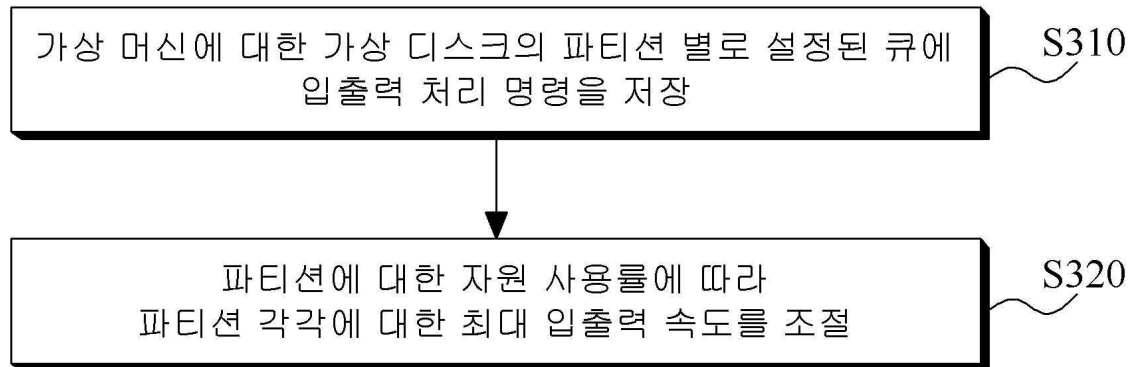
심사관 : 유진태

(54) 발명의 명칭 가상 머신에 대한 입출력 성능 제한 방법

(57) 요약

가상 머신들에게 효율적으로 자원을 분배하기 위한 가상 머신에 대한 입출력 성능 제한 방법이 개시된다. 가상 머신에 대한 입출력 성능 제한 방법은 가상 머신에 대한 가상 디스크의 파티션 별로 설정된 큐에, 입출력 처리 명령을 저장하는 단계; 및 상기 파티션에 대한 자원 사용률에 따라, 상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계를 포함한다.

대표도 - 도3



(52) CPC특허분류

G06F 2009/45579 (2013.01)

(72) 발명자

송광혁

서울특별시 광진구 능동로 209, 율곡관 302호 컴퓨
터공학과 학과사무실(군자동, 세종대학교)

박정민

서울특별시 광진구 능동로 209, 율곡관 302호 컴퓨
터공학과 학과사무실(군자동, 세종대학교)

김영우

서울특별시 광진구 능동로 209, 율곡관 302호 컴퓨
터공학과 학과사무실 (군자동, 세종대학교)

최훈하

서울특별시 광진구 능동로 209, 율곡관 302호 컴퓨
터공학과 학과사무실(군자동, 세종대학교)

박성순

경기도 안양시 동안구 시민대로327번길 11-31, 파
낙스 R&D센터 (관양동)

(56) 선행기술조사문헌

KR1020150101763 A*

KR1020110010697 A

US20100133840 A1

US20120304171 A1

*는 심사관에 의하여 인용된 문헌

이 발명을 지원한 국가연구개발사업

과제고유번호 2016R1D1A1B03930683

부처명 교육부

연구관리전문기관 한국연구재단

연구사업명 개인기초연구

연구과제명 초고성능컴퓨팅플랫폼의 비용절감과 구성노드 경량화를 위한 도커와 가상머신 기반의 하이
브리드 가상화에 관한 연구

기 여 율 1/2

주관기관 세종대학교

연구기간 2018.09.01 ~ 2019.10.31

이 발명을 지원한 국가연구개발사업

과제고유번호 R7117-16-0232

부처명 과학기술정보통신부

연구관리전문기관 정보통신기술진흥센터

연구사업명 SW컴퓨팅산업원천기술개발사업

연구과제명 32Gbps 데이터 서비스를 위한 엑스트림 스토리지 입출력 기술 개발

기 여 율 1/2

주관기관 (주)글루시스

연구기간 2016.04.01 ~ 2018.12.31

명세서

청구범위

청구항 1

하이퍼바이저의 가상 머신에 대한 입출력 성능 제한 방법에 있어서,
가상 머신에 대한 가상 디스크의 파티션 별로 설정된 큐에, 입출력 처리 명령을 저장하는 단계; 및
상기 파티션에 대한 자원 사용률에 따라, 상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계를 포함하
며,
상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계는
상기 자원 사용률에 따라서, 현재 사용중인 데이터 블록에 대한 작업이 종료된 이후 새로운 데이터 블록에 대한
작업이 개시되기 전에, 상기 최대 입출력 속도를 조절하는
가상 머신에 대한 입출력 성능 제한 방법.

청구항 2

제 1항에 있어서,
상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계는
제1파티션에 대한 입출력 속도가 제1임계값 이상인 경우, 제1파티션에 대한 최대 입출력 속도를 증가시키고, 입
출력 속도가 제2임계값 이하인 제2파티션에 대한 최대 입출력 속도를 감소시키는
가상 머신에 대한 입출력 성능 제한 방법.

청구항 3

제 2항에 있어서,
상기 제2파티션은
입출력 속도가 제2임계값 이하인 파티션 중, 입출력 속도가 가장 느린 파티션인
가상 머신에 대한 입출력 성능 제한 방법.

청구항 4

삭제

청구항 5

삭제

청구항 6

제 1항에 있어서,
상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계는
상기 입출력 처리 명령이 상기 큐에 대기하는 시간을 조절하는

가상 머신에 대한 입출력 성능 제한 방법.

청구항 7

제 6항에 있어서,

상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계는

상기 최대 입출력 속도에 따라 결정된 최대 대기 시간과, 이전 입출력 처리 명령에 따라 데이터가 처리된 시간의 차이만큼, 현재 입출력 처리 명령을 상기 큐에 대기시키는

가상 머신에 대한 입출력 성능 제한 방법.

청구항 8

제 1항에 있어서,

상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계는

상기 큐의 길이를 조절하는

가상 머신에 대한 입출력 성능 제한 방법.

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

발명의 설명

기술 분야

[0001] 본 발명은 가상 머신에 대한 입출력 성능 제한 방법에 관한 것으로서, 더욱 상세하게는 가상 머신들에게 효율적으로 자원을 분배하기 위한 가상 머신에 대한 입출력 성능 제한 방법에 관한 것이다.

배경 기술

[0003] 리눅스 기반의 하이퍼바이저인 KVM(Kernel-Based Virtual Machine)에서는 가상머신의 디스크 용량 할당 외에 최대 성능을 지정할 수 있는 옵션이 존재한다. 사용자는 가상머신 매니지먼트 프로그램인 Virt-Manager를 통해 손쉽게 입/출력(Input/Output)의 성능을 제한(limits)할 수 있다. 입출력 성능이란 초당 사용할 수 있는 최대 입/출력 속도를 의미하며, KB/sec와 IOPS, 읽기, 쓰기로 나뉘어 설정할 수 있다

[0004] 가상머신 설치 시 I/O 성능 제한은 필수로 설정해야하는 옵션이지만, 저장 공간 설정과 다르게 컴퓨터 구동에 필요한 사양이 아니기 때문에 기본(default)으로 설정되지 않는다.

[0006] 도 1은 가상 머신에 대한 입출력 성능 제한을 설정하지 않은 경우 발생할 수 있는 문제점을 나타내는 도면이다.

[0007] 하이퍼바이저(호스트)는 단일 스토리지 자원을 통해 가상머신의 저장 공간(이미지)을 나누어 사용한다. 도 1에 서와 같이, 스토리지의 최대 성능이 1GB/sec이고, 특정 가상머신(VM1)에서 많은 I/O 작업을 수행할 경우, 가상 머신에 대한 입출력 성능 제한이 설정되지 않았기 때문에, 가상 머신(VM1)은 디스크의 최대 성능인 1GB/sec를 독점하여 사용할 수 있다. 하지만 이 경우, 다른 가상머신(VM2, VM3, VM4)의 I/O 작업은 안정적인 성능을 보장

받을 수 없다.

- [0009] 도 2는 가상 머신에 대한 입출력 성능 제한을 설정할 경우 발생할 수 있는 문제점을 나타내는 도면이다.
- [0010] 가상 머신에 대한 입출력 성능 제한을 설정할 경우, 모든 가상 머신이 최대 입출력 속도로 데이터를 송수신하지 않기 때문에, 유휴 자원이 발생할 수 있다.
- [0011] 가상 머신(VM1)은 200MB/sec로 제한된 성능 안에서 자원을 최대로 사용하고 있지만 다른 가상머신들(VM2, VM3, VM4)은 제한된 성능보다 매우 낮은 자원 사용률(10MB/sec, 5MB/sec, 20MB/sec)을 보이고 있다. 이 경우, 스토리지 사용률은 235MB/sec이며, 따라서 최대 자원인 1GB/sec보다 약 750MB/sec의 자원이 낭비되고 있다.
- [0012] 한편, 가상 머신은 할당된 가상 디스크를 복수개의 파티션으로 나눠 이용할 수 있는데, 이 경우, 전술된 문제점이 동일하게 발생한다. 즉, 파티션 별로 입출력 성능을 제한할 경우, 파티션에 따라서 유휴 자원이 발생할 수 있다.
- [0013] 또한 파티션의 개수와 무관하게 가상 디스크별로 할당된 하나의 큐(queue)에 데이터 블록 처리 명령이 저장되어 데이터 블록 처리 명령이 처리되는데, 제1파티션에 대한 프로세스의 입출력 작업이 수행중이고 제1파티션에 대한 프로세스가 가상 머신에 설정된 최대 입출력 속도를 이용한다면, 다른 파티션에 대한 입출력 작업이 처리될 수 없는 병목 현상이 발생할 수 있다.
- [0014] 관련 선행문헌으로 특허 문헌인 대한민국 공개특허 제2011-0010697호가 있다.

발명의 내용

해결하려는 과제

- [0016] 본 발명은 가상 머신들에게 효율적으로 자원을 분배하기 위한 가상 머신에 대한 입출력 성능 제한 방법을 제공하기 위한 것이다.

과제의 해결 수단

- [0018] 상기한 목적을 달성하기 위한 본 발명의 일 실시예에 따르면, 가상 머신에 대한 가상 디스크의 파티션 별로 설정된 큐에, 입출력 처리 명령을 저장하는 단계; 및 상기 파티션에 대한 자원 사용률에 따라, 상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계를 포함하는 가상 머신에 대한 입출력 성능 제한 방법이 제공된다.
- [0019] 또한 상기한 목적을 달성하기 위한 본 발명의 다른 실시예에 따르면, 가상 머신에 대한 가상 디스크의 파티션 별로 설정된 큐에, 입출력 처리 명령을 저장하는 단계; 및 상기 파티션에 대한 자원 사용률의 학습 결과를 이용하여, 상기 파티션 각각에 대한 최대 입출력 속도를 조절하는 단계를 포함하는 가상 머신에 대한 입출력 성능 제한 방법이 제공된다.

발명의 효과

- [0021] 본 발명에 따르면, 가상 머신에 대한 입출력 성능 제한을 수동으로 설정할 필요없이, 가상 머신의 자원 사용률에 따라 가상 머신의 파티션 별로 입출력 성능이 실시간으로 조절될 수 있다.
- [0022] 본 발명에 따르면, 파티션 별 입출력 성능을 제한함으로써 인해 발생할 수 있는 병목 현상을 제거할 수 있으며, 파티션 별 최대 입출력 속도가 효율적으로 설정될 수 있다.

도면의 간단한 설명

- [0024] 도 1은 가상 머신에 대한 입출력 성능 제한을 설정하지 않은 경우 발생할 수 있는 문제점을 나타내는 도면이다.
- 도 2는 가상 머신에 대한 입출력 성능 제한을 설정할 경우 발생할 수 있는 문제점을 나타내는 도면이다.
- 도 3은 본 발명의 일실시예에 따른 가상 머신에 대한 입출력 성능 제한 방법을 설명하기 위한 흐름도이다.
- 도 4는 가상 디스크의 파티션에 대한 최대 입출력 속도를 조절하는 일예를 도시하는 도면이다.
- 도 5는 본 발명의 일실시예에 따른 하이퍼바이저(Hypervisor)의 블록도를 도시하는 도면이다.
- 도 6은 본 발명의 구체적 실시예에 따른 입출력 성능 제한 플랫폼을 도시하는 도면이다.

도 7은 본 발명의 다른 실시예에 따른 가상 머신에 대한 입출력 성능 제한 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0025] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.
- [0027] 본 발명은 사용자가 가상 머신에 대한 입출력 성능 제한을 수동으로 설정할 필요없이, 가상 머신의 자원 사용률에 따라 실시간으로 가상 머신에 대한 입출력 성능을 제한하는 방법을 제안한다. 즉, 본 발명에 따르면, 가상 머신에 대한 입출력 성능이, 가상 머신의 자원 사용률에 따라 실시간으로 조절될 수 있다.
- [0028] 특히 본 발명은, 가상 머신의 가상 디스크가 일반적으로 복수의 파티션으로 분할되어 사용된다는 점에 착안하여, 파티션 별로 입출력 처리 명령을 저장하는 큐를 생성하고, 이러한 큐를 이용하여 파티션 별 입출력 성능을 제한한다. 파티션 별로 입출력 처리 명령을 저장하는 큐를 이용함으로써, 파티션 별 입출력 성능을 제한함으로써 인해 발생할 수 있는 병목 현상을 제거할 수 있으며, 파티션 별 최대 입출력 속도가 효율적으로 설정될 수 있다.
- [0029] 본 발명에 따른 입출력 성능 제한 방법은 가상 머신이 구동되는 논리적 플랫폼인 하이퍼바이저에서 수행될 수 있다.
- [0030] 이하에서, 본 발명에 따른 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다.
- [0032] 도 3은 본 발명의 일실시예에 따른 가상 머신에 대한 입출력 성능 제한 방법을 설명하기 위한 흐름도이며, 도 4는 가상 디스크의 파티션에 대한 최대 입출력 속도를 조절하는 일예를 도시하는 도면이다. 도 5는 본 발명의 일 실시예에 따른 하이퍼바이저(Hypervisor)의 블록도를 도시하는 도면이다.
- [0033] 도 3을 참조하면, 본 발명에 따른 하이퍼바이저는 가상 머신에 대한 가상 디스크의 파티션 별로 설정된 큐에, 입출력 처리 명령을 저장(S310)한다. 그리고 파티션에 대한 자원 사용률에 따라, 파티션 각각에 대한 최대 입출력 속도를 조절(S320)한다.
- [0034] 여기서, 자원은 CPU, 메모리, 네트워크, 스토리지, 입출력 속도 등일 수 있으며, 일실시예로서, 하이퍼바이저는 입출력 속도 자원에 대한 사용률에 따라 파티션 각각에 대한 최대 입출력 속도를 조절할 수 있다. 이 때, 자원 사용률은 최대 입출력 속도 대비 파티션의 입출력 속도의 비율을 의미한다.
- [0035] 하이퍼바이저는 단계 S320에서 제1파티션에 대한 입출력 속도가 제1임계값 이상인 경우, 제1파티션에 대한 최대 입출력 속도를 증가시키고, 입출력 속도가 제2임계값 이하인 제2파티션에 대한 최대 입출력 속도를 감소시킬 수 있다. 그리고 제2파티션은 입출력 속도가 제2임계값 이하인 파티션 중, 입출력 속도가 가장 느린 파티션일 수 있다.
- [0036] 도 4를 참조하여 보다 자세히 설명하면, 도 4는 최초 3개의 파티션에 대한 최대 입출력 속도가 모두 100MB/s로 제한된 상태에서 최대 입출력 속도가 조절되는 과정을 도시하고 있다.
- [0037] 도 4(b)와 같이, 파티션 b의 입출력 속도가 제1임계값을 넘어 최대 입출력 속도에 다다를 경우, 하이퍼바이저는 입출력 속도가 제2임계값 이하를 나타내는 파티션 a, c 중에서 입출력 속도를 줄일 파티션 하나를 결정한다. 파티션 c의 입출력 속도(60)보다 파티션 a의 입출력 속도(40)가 보다 느리므로, 하이퍼바이저는 도 4(b) 및 (c)에 도시된 바와 같이, 파티션 a의 최대 입출력 속도를 100에서 60으로 감소시키고, 파티션 b의 최대 입출력 속도를 100에서 140으로 증가시킨다. 이후, 도 4(c)와 같이, 파티션 b의 입출력 속도가 제1임계값 이하로 낮아지면, 하이퍼바이저는 파티션 b의 입출력 속도를 100으로 낮추고, 파티션 a의 입출력 속도를 100으로 증가시킨다.
- [0038] 하이퍼바이저는 파티션들 중에서 평균 입출력 속도가 최대 입출력 속도의 90%이상인 파티션이 존재할 경우, 최대 입출력 속도를 조절할 수 있다. 하이퍼바이저는 로그(log) 데이터에 저장된 파티션에 대한 입출력 속도를 평균하여 평균 입출력 속도를 계산할 수 있다.
- [0039] 또는 하이퍼바이저는 파티션에 대한 자원 사용률을 모니터링하는 자원 모니터링 모듈로부터의 최대 입출력 속도 조절 요청에 따라, 최대 입출력 속도를 조절할 수 있다. 자원 모니터링 모듈은 가상 머신에 대한 자원 사용률을 모니터링하며, 평균 입출력 속도가 임계값 이상인 파티션이 존재할 경우, 최대 입출력 속도 조절을 요청할 수

있다.

- [0040] 한편, 본 발명에 따른 하이퍼바이저는 파티션 간의 병목 현상을 줄이기 위해 전송된 바와 같이, 파티션 별로 설정된 큐를 이용하여, 파티션 별 최대 입출력 속도를 조절한다. 파티션 별로 설정된 큐는 가상 디스크에 할당된 큐와 다른 별도의 큐로서, 본 발명에서 새로 추가되는 큐이다. 본 발명에 따른 하이퍼바이저는 일실시예로서, 입출력 처리 명령이 큐에 대기하는 시간을 조절할 수 있다. 여기서, 대기 시간은 파티션별 최대 입출력 속도에 따라 결정될 수 있다.
- [0041] 도 5를 참조하여 보다 상세히 설명하면, 도 5는 2개의 파티션이 설정된 경우의 하이퍼바이저 블록도로서, 본 발명에 따르면 도 5에 도시된 바와 같이, 파티션 각각에 대한 입출력 처리 명령을 저장하는 2개의 큐(510, 520)가 생성된다.
- [0042] 메인 프로세스(530)에 의해 생성된 파티션별 입출력 명령은 각각 파티션별로 할당된 큐에 저장된다. 즉, 제1파티션에 대한 입출력 명령은 제1큐(510)에 저장되고, 제2파티션에 대한 입출력 명령은 제2큐(520)에 저장된다.
- [0043] 이후, 큐들에 저장된 입출력 명령은 입출력 스레드에 의해 처리되어 쓰기 또는 읽기 시스템-콜이 수행된다. 쓰기 시스템-콜이 수행되면, 가상 디스크(540)에 데이터 블록이 쓰여지며, 읽기 시스템-콜이 수행되면 가상 디스크(540)로부터 데이터 블록이 읽혀진다.
- [0044] 파티션별 최대 입출력 속도가 서로 다를 수 있으며, 큐에 저장된 입출력 명령의 양도 다를 수 있는데, 순차적으로 입출력 명령이 처리되면 파티션들간에 병목 현상이 발생한다. 이에 본 발명에 따른 하이퍼바이저는 파티션별 최대 입출력 속도에 따라 대기시간을 결정하여, 입출력 처리 명령이 큐에 대기하는 시간을 조절한다.
- [0045] 보다 구체적으로 하이퍼바이저는 최대 입출력 속도에 따라 결정된 최대 대기 시간과, 이전 입출력 처리 명령에 따라 데이터가 처리된 시간의 차이만큼, 현재 입출력 처리 명령을 큐에 대기시킨다. 여기서, 데이터가 처리된 시간은 전송된 시스템-콜이 수행된 시간에 대응되며, 최대 대기 시간은 최대 입출력 속도에 반비례하도록 결정될 수 있다. 따라서 제1파티션의 최대 입출력 속도가 제2파티션의 최대 입출력 속도보다 높다면, 제1큐에 대한 최대 대기 시간은 제2큐에 대한 최대 대기 시간보다 짧다.
- [0046] 예를 들어, 이전 입출력 명령에 따라 10MB의 데이터가 처리되는 데 걸리는 시간이 0.3초이고, 제1큐에 대한 최대 대기 시간이 1초라면, 하이퍼바이저는 0.7초간 현재 입출력 명령을 제1큐에 대기시킨다. 즉, 이전 입출력 명령이 처리되고 0.7초 이후에 현재 입출력 명령이 처리된다.
- [0047] 만일 데이터가 처리되는데 걸리는 시간이 0.1초보다 길다면, 하이퍼바이저는 최대 대기 시간과 데이터 처리 시간의 시간 차이를 저장하고, 다음 입출력 명령이 큐에 대기하는 시간에 반영한다. 예컨대, 현재 입출력 명령에 따라 데이터가 처리되는데 걸리는 시간이 0.3초, 최대 대기 시간이 1초, 저장된 시간 차이가 0.2초면, 하이퍼바이저는 0.7초에서 0.2초를 뺀 0.5초만큼의 시간동안 다음 입출력 명령을 큐에 대기시킨다.
- [0048] 본 발명에 따른 하이퍼바이저는 또 다른 실시예로서, 파티션 별로 설정된 큐의 길이를 조절하여, 최대 입출력 속도를 조절할 수 있다. 하이퍼바이저는 최대 입출력 속도를 늘릴 필요가 있는 파티션에 대해 큐의 길이를 증가시키고, 최대 입출력 속도를 줄일 필요가 있는 파티션에 대해 큐의 길이를 감소시킬 수 있다.
- [0049] 한편, 전송된 바와 같이, 본 발명에 따른 하이퍼바이저는 가상 머신에 설정된 파티션 별 최대 입출력 속도를 조절하는데, 데이터의 입출력이 지속적으로 발생하는 상황에서의 최대 입출력 속도 조절은 오버헤드가 될 수 있다. 따라서, 본 발명에 따른 하이퍼바이저는 파티션 별 자원 사용률에 따라서 최대 입출력 속도 조절 시점을 조절한다.
- [0050] 하이퍼바이저는 일실시예로서, 현재 최대 입출력 속도와 코루틴 큐(co-routine queue)에 저장된 데이터 처리 명령의 크기를 비교하여 최대 입출력 속도 조절 시점을 조절할 수 있다. 현재 최대 입출력 속도와 코루틴 큐의 크기에 따라서 데이터의 입출력 처리 작업이 많은지 적은지 판단될 수 있다.
- [0051] 하이퍼바이저는 자원 모니터링 모듈의 최대 입출력 속도 조절을 요청받고 임계 시간 이후 최대 입출력 속도를 조절한다.
- [0052] 데이터의 입출력 처리 작업이 적은 경우, 즉 자원 사용률이 높지 않은 경우, 임계 시간 이후 최대 입출력 속도를 조절한다.
- [0053] 반면, 데이터의 입출력 처리 작업이 많은 경우, 즉 자원 사용률이 높은 경우, 하이퍼바이저는 임계 시간을 대기하고, 현재 사용중인 데이터 블록에 대한 작업이 종료된 이후 새로운 데이터 블록에 대한 작업이 개시되기

전에, 최대 입출력 속도를 조절한다. 예를 들어, a 파일에 대한 작업이 진행중일 때 임계 시간이 종료되더라도, 하이퍼바이저는 a 파일에 대한 작업이 종료되고 b 파일에 대한 작업이 개시되기 전에 최대 입출력 속도를 조절한다.

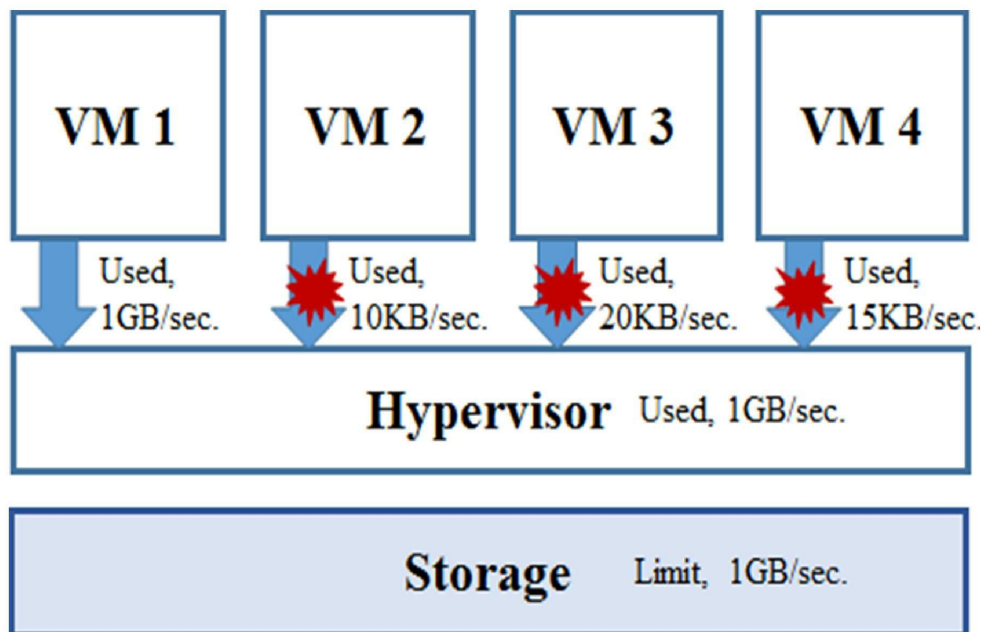
- [0055] 도 6은 본 발명의 구체적 실시예에 따른 입출력 성능 제한 플랫폼을 도시하는 도면으로서, 도 6에서는 리눅스 기반으로 동작하는 플랫폼이 도시된다.
- [0056] 도 6에서는 KVM 하이퍼바이저(630) 상에 2개의 가상 머신이 구동되고, 제1가상 머신(610)은 3개의 파티션, 제2가상 머신(620)은 2개의 파티션을 사용한다.
- [0057] KVM 하이퍼바이저(630)는 가상 머신 별로 할당된 쓰로틀 컨트롤러(Throttle Controller, 631, 632)와 파티셔닝 풀(Partitioning Pool, 633, 634)을 포함한다.
- [0058] 쓰로틀 컨트롤러(631, 632)는 파티셔닝 풀(633, 634)에서 입출력 처리 명령이 대기하는 시간을 조절하여 파티션 각각에 대한 최대 입출력 속도를 조절한다. 쓰로틀 컨트롤러(731, 732)는 파티션에 대한 자원 사용률을 모니터링하여 최대 입출력 속도를 조절하거나 또는 자원 모니터링 모듈(740)의 최대 입출력 속도 조절 요청에 따라 최대 입출력 속도를 조절한다.
- [0059] 파티셔닝 풀(633, 634)은 각 가상머신에서 이용되는 파티션의 개수에 따라 할당된 큐로 구성된다. 파티셔닝 풀(633, 634)은, 리눅스 시스템에서 입출력 처리 명령이 저장되는 ThreadPool이 파티션 별로 할당된 형태이다. 도 7에서 제1파티셔닝 풀(633)은 3개의 큐로 구성되며, 각 큐에서의 대기 시간은 파티션 별 최대 입출력 속도인 30, 20, 50MB/sec에 따라서 조절된다. 제2파티셔닝 풀(634)은 2개의 큐로 구성되며, 각 큐에서의 대기 시간은 파티션별 최대 입출력 속도인 30, 120MB/sec에 따라 조절된다.
- [0060] 자원 모니터링 모듈(640)은 하이퍼바이저(630)와 가상 머신(610, 620)의 자원 사용률을 실시간으로 모니터링하여 데이터베이스(650)에 저장한다. 자원 모니터링 모듈(640)은 가상 머신들의 파티션별 자원 사용률이 임계값 이상일 경우, 쓰로틀 컨트롤러(631, 632)로 최대 입출력 속도 조절을 요청한다.
- [0061] 자원 모니터링 모듈(640)에서 수집된 정보, 제어 정보는 후술되는 학습을 위한 레퍼런스 데이터로 이용될 수 있다.
- [0062] 또한 자원 모니터링 모듈(640)은 주기적으로 핑(ping) 신호를 스로틀 컨트롤러(631, 632)로 전송하여, 스로틀 컨트롤러(631, 632)가 정상적으로 동작하는지 확인할 수 있다.
- [0064] 도 7은 본 발명의 다른 실시예에 따른 가상 머신에 대한 입출력 성능 제한 방법을 설명하기 위한 흐름도이다.
- [0065] 본 발명에 따른 하이퍼바이저는 도 3에서 설명된 입출력 성능 제한 방법과 달리 파티션에 대한 자원 사용률의 학습 결과를 이용한다. 학습은 다양한 머신 러닝 알고리즘을 통해 수행될 수 있다.
- [0066] 본 발명에 따른 하이퍼바이저는 가상 머신에 대한 가상 디스크의 파티션 별로 설정된 큐에, 입출력 처리 명령을 저장(S710)하고, 파티션에 대한 자원 사용률의 학습 결과를 이용하여, 파티션 각각에 대한 최대 입출력 속도를 조절(S720)한다.
- [0067] 여기서, 학습 결과는 이용되는 데이터 블록에 따른 상기 파티션의 입출력 속도의 학습 결과일 수 있다. 본 발명에 따른 하이퍼바이저는 파티션 별로 읽거나 쓰는 데이터 블록에 따라서 파티션의 입출력 속도가 어떻게 변하는지를 학습한다. 그리고, 현재 파티션이 읽어들이는 데이터 블록이나 쓰는 데이터 블록이 어떠한 데이터 블록인지에 따라서 파티션 각각에 대한 최대 입출력 속도를 조절한다.
- [0068] 예를 들어, 제1주소의 데이터 블록이 읽혀질 때의 입출력 속도가 높았던 것으로 학습되었으며, 현재 파티션에서 읽혀지는 데이터 블록이 제1주소의 데이터 블록일 경우, 하이퍼바이저는 해당 파티션의 최대 입출력 속도를 증가시켜 설정할 수 있다.
- [0069] 또는 실시예에 따라서, 학습 결과는 프로세스에 따른 파티션의 입출력 속도의 학습 결과일 수 있다. 이 경우, 하이퍼바이저는 프로세스 별로 파티션의 입출력 속도가 어떻게 변하는지를 학습하고, 현재 수행되는 프로세스에 따라 파티션 각각에 대한 최대 입출력 속도를 조절할 수 있다.
- [0070] 다시 설명하면, 본 발명에 따른 하이퍼바이저는 특정 상황에서의 파티션에 대한 자원 사용률의 패턴을 분석하고, 현재 상황이 학습된 상황에 대응되는지를 판단하여 분석 결과에 따라 파티션 각각에 대한 최대 입출력 속도를 조절할 수 있다.

[0072] 앞서 설명한 기술적 내용들은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예들을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 하드웨어 장치는 실시예들의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

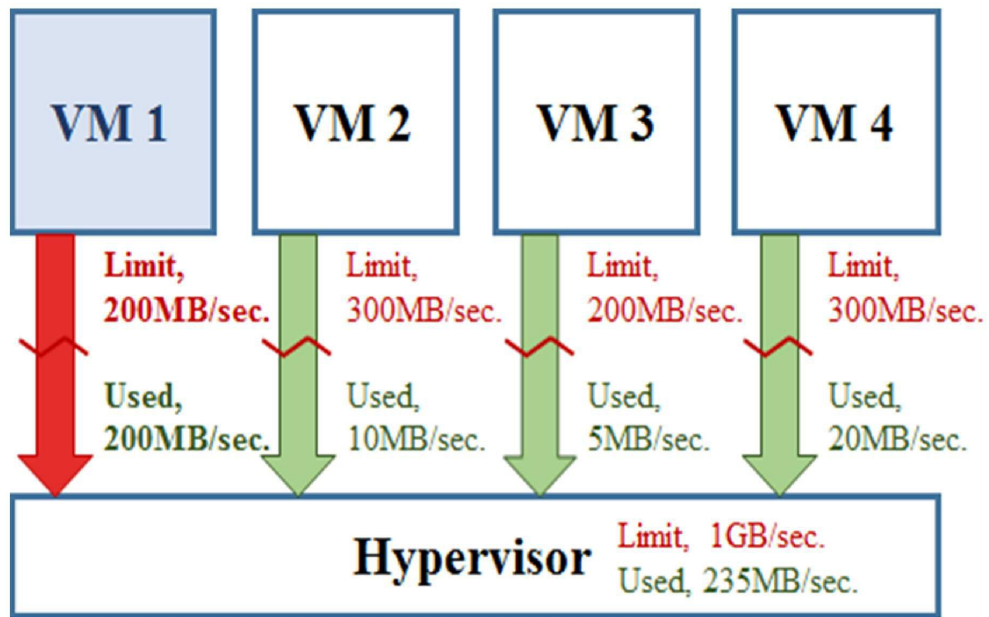
[0074] 이상과 같이 본 발명에서는 구체적인 구성 요소 등과 같은 특정 사항들과 한정된 실시예 및 도면에 의해 설명되었으나 이는 본 발명의 보다 전반적인 이해를 돕기 위해서 제공된 것일 뿐, 본 발명은 상기의 실시예에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상적인 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능하다. 따라서, 본 발명의 사상은 설명된 실시예에 국한되어 정해져서는 아니되며, 후술하는 특허청구범위뿐 아니라 이 특허청구범위와 균등하거나 등가적 변형이 있는 모든 것들은 본 발명 사상의 범주에 속한다고 할 것이다.

도면

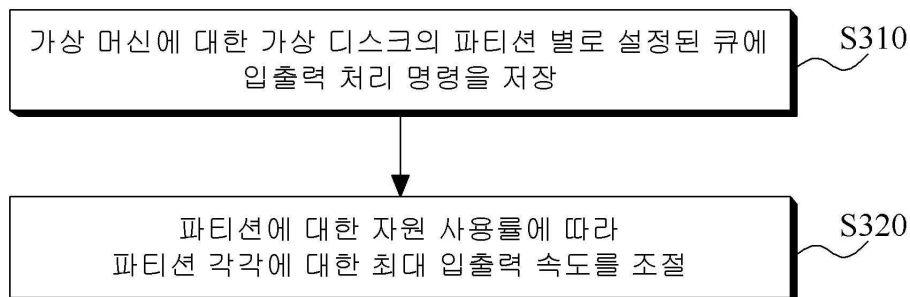
도면1



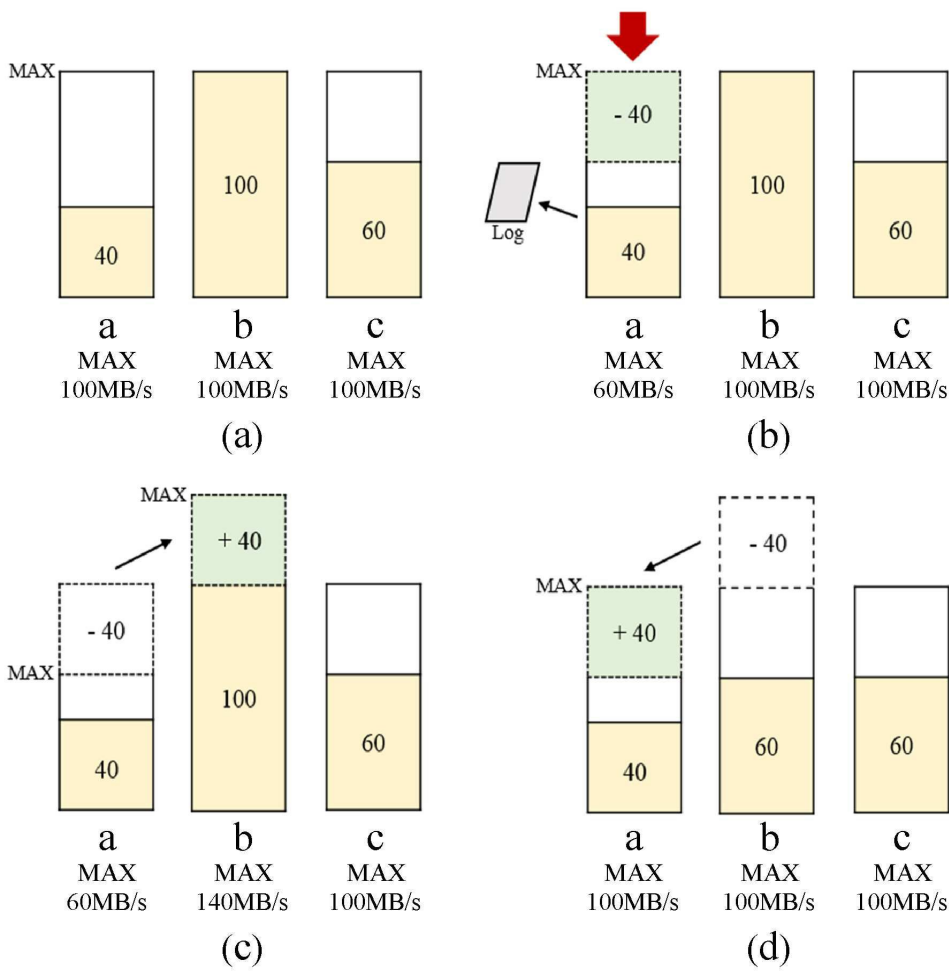
도면2



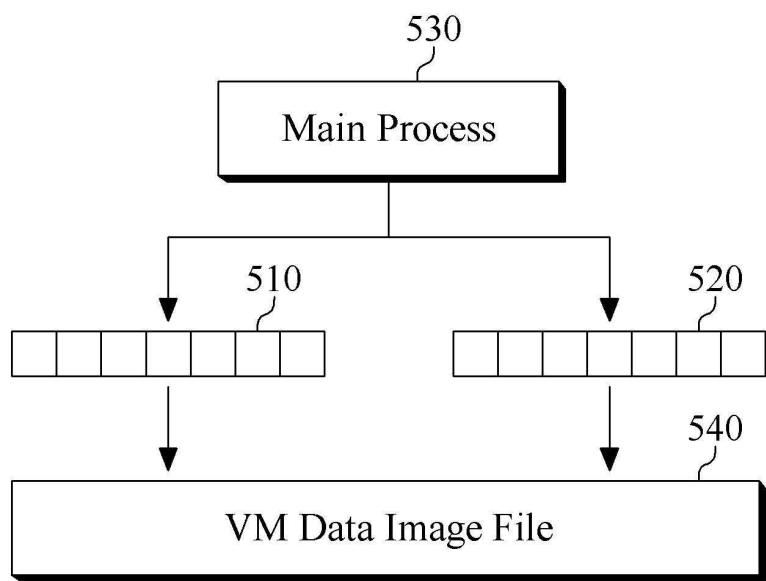
도면3



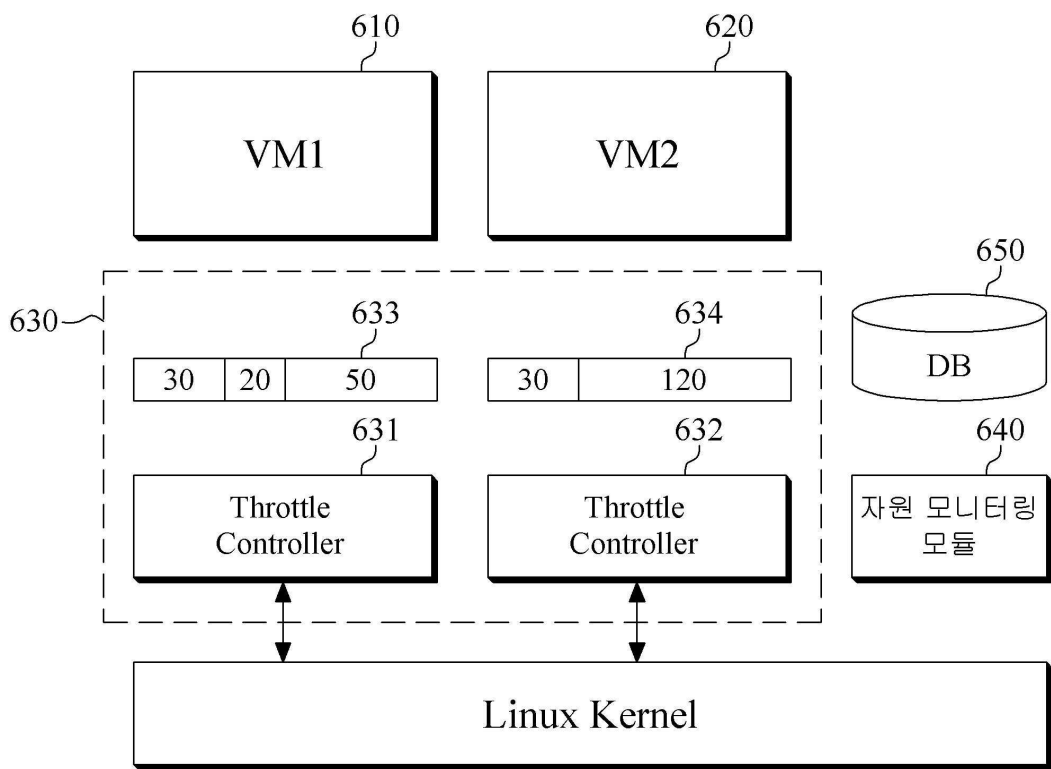
도면4



도면5



도면6



도면7

