



(19) 대한민국특허청(KR)
 (12) 등록특허공보(B1)

(45) 공고일자 2010년12월06일
 (11) 등록번호 10-0998299
 (24) 등록일자 2010년11월29일

(51) Int. Cl.

G06F 15/167 (2006.01) G06F 15/173 (2006.01)

G06F 9/54 (2006.01) G06F 13/38 (2006.01)

(21) 출원번호 10-2007-7013199

(22) 출원일자(국제출원일자) 2005년11월17일

심사청구일자 2008년06월25일

(85) 번역문제출일자 2007년06월12일

(65) 공개번호 10-2007-0094742

(43) 공개일자 2007년09월21일

(86) 국제출원번호 PCT/EP2005/056040

(87) 국제공개번호 WO 2006/061316

국제공개일자 2006년06월15일

(30) 우선권주장

11/009,937 2004년12월10일 미국(US)

(56) 선행기술조사문헌

IEEE computer vol.27 no.3 pp.75-83 march 1994

Ornan Kreiger et al "The alloc stream facility a redesign of application-level stream I/O"

US20040028063 A1

US5212772 A

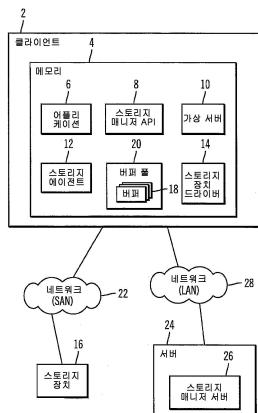
전체 청구항 수 : 총 10 항

심사관 : 황승희

(54) 공유 베퍼에서의 시스템과 스토리지 사이의 데이터 전송

(57) 요 약

공유된 베퍼에서 시스템과 스토리지(storage) 사이에 데이터를 전송하기 위한 방법, 시스템 및 프로그램을 기록한 컴퓨터 판독가능 기록매체가 제공된다. 어플리케이션은 컴퓨트로부터 베퍼를 요청한다. 컴퓨트는 베퍼를 할당하고 어플리케이션에 어플리케이션 데이터를 위한 베퍼의 제1 오프셋 및 제1 헤더를 위한 베퍼의 제2 오프셋을 반환한다. 어플리케이션은 베퍼의 제1 오프셋에 어플리케이션 데이터를 기록하고, 베퍼의 제2 오프셋에 제1 헤더를 기록한다. 제2 헤더는 베퍼의 제3 오프셋으로 기록된다.

대 표 도 - 도1

특허청구의 범위

청구항 1

공유 버퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법으로서,

어플리케이션에 의해, 제1 컴포넌트로부터 버퍼를 요청하는 단계;

상기 제1 컴포넌트에 의해, 상기 버퍼를 할당하는 단계;

상기 제1 컴포넌트에 의해, 상기 버퍼의 제1 및 제2 오프셋을 계산하는 단계로서, 상기 제1 오프셋은, 상기 버퍼의 제1 오프셋 앞에 제1 및 제2 헤더를 기록하도록 상기 제1 오프셋 앞에 상기 버퍼에 충분한 크기의 제1 공간을 제공하기 위해 상기 버퍼 내의 위치를 가리키도록 계산되는 것이고, 상기 제2 오프셋은, 상기 버퍼의 제2 오프셋 앞에 상기 제2 헤더를 기록하도록 상기 버퍼에 충분한 크기의 제2 공간을 제공하기 위해 계산되는 것이며, 상기 제1 및 제2 오프셋은, 이전에 기록된 데이터 앞에 데이터를 기록하는 것을 가능하게 하기 위해 상기 버퍼에 충분한 크기의 제1 및 제2 공간을 제공하도록 계산되는 것인, 상기 제1 및 제2 오프셋 계산 단계;

상기 제1 컴포넌트에 의해, 어플리케이션 데이터에 대한 상기 버퍼의 제1 오프셋 및 상기 제1 헤더에 대한 상기 버퍼의 제2 오프셋을 상기 어플리케이션으로 반환하는 단계;

상기 어플리케이션에 의해, 상기 어플리케이션 데이터를 상기 버퍼의 제1 오프셋에 기록하는 단계;

상기 어플리케이션에 의해, 상기 제1 헤더를 상기 버퍼의 제2 오프셋에 기록하는 단계;

상기 제2 헤더를 상기 버퍼의 제3 오프셋에 기록하는 단계;

상기 제2 헤더에 대한 포인터를 제2 컴포넌트로 전달하는 단계; 및

상기 제2 컴포넌트에 의해, 상기 버퍼를 어떻게 처리할 것인지를 결정하기 위해 상기 포인터의 수신에 응답하여 상기 버퍼의 제2 헤더를 처리하는 단계

를 포함하는, 공유 버퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 2

제1항에 있어서, 상기 제1 오프셋은 상기 제2 오프셋보다 크고, 상기 제2 오프셋은 상기 제3 오프셋보다 큰 것인, 공유 버퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 3

제1항에 있어서, 상기 제1 컴포넌트는 상기 제2 헤더를 상기 버퍼에 기록하는 것이고,

제3 헤더를 상기 버퍼의 제4 오프셋에 기록하는 단계를 더 포함하며,

상기 제1 오프셋은 상기 제2 오프셋보다 크고, 상기 제2 오프셋은 상기 제3 오프셋보다 크고, 상기 제3 오프셋은 상기 제4 오프셋보다 큰 것인, 공유 버퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 4

제1항에 있어서,

상기 제2 컴포넌트에 의해, 스토리지 장치로 버퍼 컨텐츠를 전송하는데 사용되는 정보로 상기 제2 헤더를 대체시키는 단계를 더 포함하는, 공유 버퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 5

제1항에 있어서, 상기 제3 오프셋의 제2 헤더는 상기 어플리케이션 데이터와 상기 제1 헤더 및 제2 헤더를 포함하는 버퍼 컨텐츠가 제1 장치 또는 제2 장치로 전송되는 것을 나타내는 것이고,

상기 제1 컴포넌트에 의해, 상기 제2 헤더에 대한 포인터를 상기 제2 컴포넌트로 전달하는 단계;

상기 제2 컴포넌트에 의해, 상기 버퍼 컨텐츠를 제1 장치 또는 제2 장치로 전송할 것인지 여부를 결정하기 위해 상기 제2 헤더를 처리하는 단계; 및

상기 제2 헤더가 상기 베퍼 컨텐츠를 상기 제1 장치로 전송하는 것을 나타내는 것으로 결정되는 것에 응답하여, 상기 베퍼 컨텐츠를 상기 제1 장치로 전송하는데 사용되는 장치 정보를 상기 제2 헤더에 기록하는 단계로서, 상기 장치 정보는 상기 제1 컴포넌트에 의해 기록된 상기 제2 헤더 정보를 대체하는 것인, 상기 제2 헤더에 기록하는 단계

를 포함하는, 공유 베퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 6

제1항에 있어서,

상기 제1 컴포넌트에 의해, 상기 어플리케이션 데이터 및 상기 제1 헤더의 기록 다음에, 상기 제2 헤더를 기록하기 위한 상기 제3 오프셋 앞에 상기 베퍼에 충분한 크기의 공간을 제공하기 위하여 상기 제3 오프셋을 계산하는 단계를 더 포함하는, 공유 베퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 7

제1항에 있어서,

상기 어플리케이션 데이터가 고정된 숫자의 배수인지 여부를 결정하는 단계를 더 포함하고,

상기 제1 및 제2 오프셋은, 상기 어플리케이션 데이터가 상기 고정된 숫자의 배수가 아닌 것으로 결정되는 것에 응답하여 빈 공간과 상기 어플리케이션 데이터가 상기 고정된 숫자의 배수를 차지하도록 하기 위해 상기 제1 오프셋에 기록된 상기 어플리케이션 데이터 앞에 상기 빈 공간이 유지되도록 계산되는 것이고,

상기 제1 헤더는 상기 제2 오프셋부터 상기 빈 공간의 시작까지 확장되는 것인, 공유 베퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 8

제1항에 있어서,

상기 어플리케이션에 의해, 복수의 베퍼들을 요청하는 단계;

상기 제1 컴포넌트에 의해, 상기 복수의 베퍼들을 상기 어플리케이션에 반환하는 단계;

상기 어플리케이션에 의해, 어플리케이션 데이터 및 제1 헤더를 상기 복수의 베퍼들에 동시에 기록하는 단계를 더 포함하는, 공유 베퍼에서 시스템과 스토리지 사이에 데이터를 전송하기 위한 방법.

청구항 9

프로세서;

상기 프로세서에 접속 가능하고, 베퍼들을 구비하는 메모리;

어플리케이션 및 컴포넌트를 포함하는 컴퓨터로 관독가능한 매체 내의 코드를 포함하고,

상기 프로세서는 상기 어플리케이션 및 상기 컴포넌트가 상기 청구항 제1항 내지 제8항 중 어느 한 항의 단계들을 수행하도록 상기 코드를 실행시키는 것인 시스템.

청구항 10

베퍼들을 포함하는 메모리와 통신하는 어플리케이션 및 컴포넌트를 포함하고,

상기 어플리케이션 및 상기 컴포넌트는 상기 청구항 제1항 내지 상기 제8항 중 어느 한 항에 따른 방법의 단계들을 수행하도록 실행될 수 있는 것인 컴퓨터 프로그램을 기록한 컴퓨터 관독가능 기록매체.

명세서

기술 분야

[0001] 본 발명은 공유 베퍼에서의 시스템과 스토리지(storage) 사이에서 데이터를 전송하기 위한 방법, 시스템 및 프

로그램을 기록한 컴퓨터 관독가능 기록매체에 관한 것이다.

배경기술

[0002] 네트워크 백업(backup) 환경에서, 클라이언트 시스템은 스토리지 장치에 데이터를 백업하고, 백업을 백업 서버와 통합(coordinate)시킬 수 있다. 예를 들어, 아이비엠("IBM"®) 티볼리(Tivoli®) 스토리지 매니저 제품은 클라이언트 및 서버 시스템이 클라이언트 데이터를 백업하기 위한 소프트웨어를 제공한다. (IBM과 Tivoli는 IBM의 등록 상표이다.) 클라이언트 머신은 클라이언트의 어플리케이션과 스토리지 장치 사이에서 전송되는 데이터를 처리하는 다수의 프로그램 컴포넌트들을 포함할 수 있다. 각 컴포넌트는 이전 컴포넌트에 의해 사용된 버퍼로부터 데이터를 독출하고, 데이터를 각 컴포넌트에 할당된 버퍼로 복사하는 메모리 복사 작업을 수행한다. 데이터 처리 과정(prcessing)의 일부로서 데이터가 컴포넌트에 할당된 버퍼들 사이를 통과할 때, 프로그램 컴포넌트들은 데이터에 헤더 정보를 부가할 수 있다. 프로그램 컴포넌트는 이전 컴포넌트의 버퍼로부터 데이터를 독출하고, 상기 데이터를 처리하고, 상기 데이터에 새로운 헤더를 부가하고, 그 후에 이 데이터 및 새로운 헤더를 상기 컴포넌트에 할당된 버퍼에 기록한다. 데이터를 처리하는 일련의 컴포넌트들 중에서 다음 프로그램 컴포넌트는 전술한 바와 같이 동작한다.

발명의 상세한 설명

[0003] 공유 버퍼에서 시스템과 스토리지(storage) 사이에 데이터를 전송하기 위한 방법, 시스템 및 프로그램을 기록한 컴퓨터 관독가능 기록매체가 제공된다. 본 발명의 제1 관점에서, 어플리케이션은 컴포넌트로부터 버퍼를 요청한다. 컴포넌트는 버퍼를 할당하고 어플리케이션 데이터를 위한 버퍼의 제1 오프셋 및 제1 헤더를 위한 버퍼의 제2 오프셋을 어플리케이션에 반환한다. 어플리케이션은 버퍼의 제1 오프셋에 어플리케이션 데이터를 기록하고, 제1 헤더를 버퍼의 제2 오프셋에 기록한다. 제2 헤더는 버퍼의 제3 오프셋에 기록된다.

[0004] 본 발명의 제2 관점에 따르면, 제1 컴포넌트는 어플리케이션으로부터 버퍼 헌들을 수신한다. 제1 컴포넌트는 제3 오프셋을 계산하고, 제2 헤더를 버퍼의 제3 오프셋에 기록한다. 제1 컴포넌트는 버퍼 컨텐츠가 스토리지 장치 또는 서버로 전송되어야 하는지 여부를 표시하는 제3 헤더를 버퍼의 제4 오프셋에 기록하고, 제3 헤더에 대한 포인터를 제2 컴포넌트로 전달한다. 제2 컴포넌트는 제3 헤더가 버퍼 컨텐츠를 서버로 전송하도록 지시하는지 또는 스토리지 장치로 전송하도록 지시하는지를 결정하고, 서버에 대해 의도된 경우 버퍼 컨텐츠를 스토리지 서버로 전송하고, 스토리지 장치에 대해 의도된 경우 버퍼에 대한 제2 포인터를 제3 컴포넌트로 전송한다.

실시예

[0008] 이하의 상세한 설명에서, 본 명세서의 일부를 구성하는 첨부된 도면들에는 참조번호를 붙였고, 첨부된 도면은 본 발명의 다양한 실시예를 나타낸다. 다른 실시예들이 이용될 수 있고, 본 발명의 범위를 벗어나지 않고 구조 및 동작의 변형이 있을 수 있음을 자명하다.

[0009] 도 1은 메모리(4)를 포함하는 스토리지 클라이언트(2)를 구비한 네트워크 컴퓨팅 환경을 도시한 것이다. 스토리지 클라이언트(2)는 프로그램 컴포넌트로서 데이터베이스 어플리케이션과 같은 어플리케이션(6), 스토리지 매니저 어플리케이션 프로그램 인터페이스(application program interface, API, 8), 가상 서버(10), 스토리지 애플리케이션(12) 및 스토리지 장치(16)에 기록하는 것에 대한 인터페이스를 제공하는 스토리지 장치 드라이버(14)를 포함한다. 스토리지 매니저 API(8)는 버퍼 풀(20)의 버퍼들(18)들을 관리한다.

[0010] 일 실시예에서, 스토리지 장치 드라이버(14)는 스토리지 영역 네트워크(Storage Area Network, SAN)와 같은 네트워크(22)를 통하여 스토리지 장치(16)로 데이터를 전달한다. 대안적인 실시예에서, 클라이언트(2)는 PCI(Peripheral Component Interconnet) 버스, SCSI(Small Computer SYstem Interface) 버스 등과 같은 버스 인터페이스를 통하여 스토리지 장치(16)에 접속할 수 있다.

[0011] 일 실시예에서, 가상 서버(10)는 LAN(Local area Network)과 같은 제2 네트워크(28)를 통하여 스토리지 장치(16)에 기록된 데이터에 관한 정보를 스토리지 매니저 서버(26)를 갖는 서버(24)로 전달할 수 있다. 이러한 방식으로, 데이터는 SAN과 같은 제1 네트워크를 통하여 스토리지로부터 또는 스토리지로 전송되고, 전송되고 있는 데이터에 관한 제어 정보는 LAN과 같은 다른 네트워크를 통하여 전송된다. 대안적인 실시예에서, 데이터 및 정보와 데이터 전송 동작을 위한 명령들 모두는 도 1에 도시된 것처럼 개별적인 네트워크들이 아니라, 동일한 네트워크를 통하여 전송될 수 있다. 서버(26)는 백업 데이터 세트들을 관리하기 위해 백업 데이터베이스 내의 스토리지 장치(16)에 저장되어 있는 데이터에 관한 정보를 유지한다.

[0012]

스토리지 매니저 API(8)는 어플리케이션(6)과 스토리지 장치(16) 사이에서 전송되는 데이터를 저장하는데 사용하기 위해 버퍼 풀(20) 내의 버퍼(18)를 클라이언트(2) 내의 컴포넌트들에 할당한다. 도 2는 어플리케이션(6)에 의해 기록된 어플리케이션 데이터(40)와 어플리케이션 헤더(44), 스토리지 매니저 API(8)에 의해 기록된 API 헤더(46), 스토리지 매니저 API(8)에 의해 기록된 버퍼를 처리하는 방법을 지시하는 verb 정보(48)를 포함하는 컴포넌트들(6, 8, 10, 12 및 14)이 버퍼(18)에 부가한 정보를 보여준다. 일 실시예에서, 스토리지 에이전트(12)는 스토리지 장치 드라이버(14)가 데이터를 스토리지 장치(16)에 전송할 수 있도록 동사 정보(48)를 장치 특유 정보(46)로 덮어쓰기(overwrite)한다. 설명된 실시예들에서, 컴포넌트(예컨대, 6, 8, 10, 12 and 14)는, 컴포넌트가 버퍼에 기록한 헤더 또는 다른 정보를 포함하는 버퍼 컨텐츠(content)에 대한 포인터(pointer)를 상기 버퍼 컨텐츠를 처리하는 다음 컴포넌트(예컨대, 8, 10, 12 and 14)로 전달한다. 이러한 방식으로, 버퍼(18)는 컴포넌트들에 의해 공유되고, 따라서 버퍼에서 데이터를 처리하고 헤더 정보를 부가하는 각 컴포넌트는 컨텐츠를 복사하는 부가적인 버퍼를 할당할 필요가 없다. 대신에, 각 컴포넌트는 데이터를 처리하고 헤더 및 다른 정보를 동일한 공유된 버퍼에 부가할 수 있고, 그것에 의해 별개의 버퍼들로부터 또는 버퍼들로 데이터의 메모리 복사 작업을 수행할 필요가 없다.

[0013]

도 3 내지 도 5는 스토리지 장치(16)로의 전송을 위해 컴포넌트들(예컨대, 6, 8, 10, 12 and 14) 사이에서 데이터를 전송하도록 컴포넌트들에 의해 수행되는 동작들을 보여준다. 도 3과 관련하여, 어플리케이션(6)은 스토리지 매니저 API(8)로부터 하나 이상의 버퍼들(18)을 요청함으로써(102 단계) 스토리지 장치(16)로 어플리케이션 데이터를 전송하기 위한 동작을 개시한다(100 단계). 이 요청은 어플리케이션(6)이 기록하려고 의도한 헤더의 최대 길이를 포함할 수 있고, 이 길이는 어플리케이션(6)이 헤더를 기록하지 않을 경우 영(zero)이 될 수 있다. 응답으로, 스토리지 매니저 API(8)는 버퍼 풀(20)에서 자유 버퍼들로 표시된 하나 이상의 버퍼(18)를 어플리케이션에 할당한다(104 단계). 제1 컴포넌트, 즉 스토리지 매니저 API(8)는 제1 오프셋으로부터의 어플리케이션 데이터(40)의 기록 다음에 제1 및 제2 헤더, 예컨대 어플리케이션 헤더(42) 및 API 헤더(44)를 기록하는데 있어서 제1 오프셋 앞에 버퍼(18)에서 충분한 양의 공간을 제공하기 위해 어플리케이션 데이터(40)에 대한 제1 오프셋을 계산한다(106 단계). 제1 오프셋은 요청에 의해 표시된 헤더의 길이에 대해 버퍼(18)에서 충분한 공간을 제공할 수 있다. 따라서 제1 오프셋은 버퍼(18)에서 어플리케이션 데이터가 시작하는 부분에 주소가 정해진다. 제1 컴포넌트(8)는 어플리케이션 데이터(40) 및 어플리케이션 헤더(44)의 기록 다음에 제2 헤더 예컨대, API 헤더(44)를 기록하는데 있어서 제2 오프셋 앞에 버퍼(18)에서 충분한 공간을 제공하기 위해 제2 오프셋을 계산한다(108 단계). 이러한 방식으로, 오프셋들은 하나의 컴포넌트가 이전 컴포넌트에 의해 기록된 데이터 앞에 데이터를 기록할 수 있도록 계산되며, 그에 따라 컴포넌트들이 이전에 기록된 어플리케이션 데이터(40)와 헤더들 앞에 헤더들을 부가할 수 있다. 더욱이, 임의의 컴포넌트들(6, 8, 10, 12 및 14)이 버퍼(18)에서 헤더들과 데이터의 위치를 결정할 수 있도록 버퍼(18)내의 오프셋 및 각 헤더의 길이를 포함하여, 모든 헤더들(예컨대, 48, 46, 44)과 데이터(40)에 관한 정보를 갖는 버퍼(18)의 탑(top)에 데이터 구조가 있을 수 있다.

[0014]

설명된 실시예들에서, 스토리지 매니저 API(8)는 데이터가 4K 경계와 같이 고정된 경계 영역 상에 존재하는 것을 보장하기를 원할 수 있다. 어플리케이션(6)이 스토리지 장치(16)로 입출력(I/O)할 때, 버퍼(18)는 I/O에 대한 커널(kerne1)로 전송된다. 직접 메모리 어드레싱(Direct Memory Addressing, DMA)을 수행하기 위해 커널은 버퍼(18)를 물리적 메모리(4)(가상 메모리와 반대로)로 “핀(pin)”할 수 있고, 그 결과 I/O 장치(PCI 버스, SCSI 버스, SAN 등)는 데이터로의 접속을 획득할 수 있고, 그것을 적절한 장치로 전송할 수 있다. 이 버퍼(18)의 “피닝(pining)”은 (버스 상에서) I/O 전송이 진행 중인 동안 다른 어플리케이션들이 데이터를 사용하는 것을 방지한다.

[0015]

포인터 어플리케이션 데이터(40)가 고정된 숫자(예컨대, 4Kb)의 배수가 아니라면(110 단계), 제1 컴포넌트(스토리지 매니저 API(8))는 제1 오프셋에 기록된 어플리케이션 데이터 앞에 빈 공간이 유지되도록 제1 및 제2 오프셋을 계산하고(110 단계), 그 결과 빈 공간과 어플리케이션 데이터가 고정된 숫자 즉, 4Kb의 배수를 차지하게 된다. 예를 들어, 제2 오프셋은 4Kb 경계 상에 어플리케이션 데이터를 배열(align)하기 위해 필요한 빈 공간(42)의 크기와 동일한 제1 오프셋으로부터의 오프셋과, 제2 오프셋에서 시작하여 기록되는 어플리케이션 헤더(44)의 길이를 포함할 수 있다. 제1 컴포넌트(8)는 제1 및 제2 오프셋과 할당된 버퍼(18)에 대한 핸들(handle)을 어플리케이션(6)으로 반환한다(114 단계).

[0016]

오프셋들에 대한 응답으로, 어플리케이션(6)은 버퍼(18)의 제1 오프셋에 어플리케이션 데이터(40)를 기록하고(116 단계), 버퍼(18)의 제2 오프셋에 어플리케이션 헤더(44)와 같은 제1 헤더를 기록한다(118 단계). 어플리케이션 헤더(44)의 기록은 선택적일 수 있다. 기록 후에, 어플리케이션(6)은 버퍼 핸들을 제1 컴포넌트(스토리지 매니저 API(8))로 전달한다(120 단계).

[0017]

도 4와 관련하여, 어플리케이션(6)으로부터 버퍼 헨들을 수신하면(130 단계), 어플리케이션 데이터(40) 및 제1 헤더(어플리케이션 헤더(44))의 기록 다음에 제1 컴포넌트(스토리지 매니저 API(8))는 제2 헤더(API 헤더(46))를 기록하기 위하여 제3 오프셋 앞에 버퍼내의 충분한 공간을 제공하기 위해 제3 오프셋을 계산한다(132 단계). 일 실시예에서, 제1 헤더(어플리케이션 헤더(44))가 고정된 숫자 예컨대, 4Kb의 배수라면, 스토리지 매니저 API(8)은 제1 헤더 앞에 빈 공간을 유지하도록 제3 오프셋을 선택하고(136 단계), 그 결과 제1 헤더와 빈 공간은 고정된 숫자(4Kb)의 배수를 차지한다. 제2 헤더(API 헤더(46))는 버퍼(18)의 제3 오프셋에 기록된다(138 단계). 스토리지 매니저 API(8)는 버퍼 컨텐츠가 스토리지 장치(16) 또는 서버(24)로 전송되는지 여부를 나타내는 제3 헤더(verb 헤더(48))를 버퍼의 제4 오프셋에 더 기록한다(138 단계). 스토리지 매니저 API(8)는 스토리지 장치(16)를 위해 예정된 데이터를 스토리지 장치(16)로 전송하고 전송 백업 명령들과 관련 정보를 클라이언트(2) 백업 동작을 관리하는 서버(24)로 전송하도록 지시한다. 스토리지 매니저 API(8)는 제3 헤더에 대한 포인터를 가상 서버(10)와 같은 제2 컴포넌트로 전달한다(142 단계).

[0018]

제3 헤더에 대한 포인터를 수신에 응답하여, 가상 서버(10)는 제3 헤더(예컨대, verb 헤더(48))가 버퍼 컨텐츠를 스토리지 장치(16)나 서버(24)로 전송하도록 지시하는지 여부를 결정한다(144 단계). 144 단계에서 컨텐츠가 서버(24)에 대해 의도된 경우, 가상 서버(10)는 네트워크(28)를 통하여 버퍼 컨텐츠를 서버(24)로 전송한다(146 단계). 한편, 144 단계에서 컨텐츠가 스토리지 장치(16)에 대해 의도된 경우, 가상 서버(10)는 스토리지 에이전트(12)와 같은 제3 컴포넌트로 버퍼(18)에 대한 제2 포인터를 전송한다(148 단계).

[0019]

도 5와 관련하여, 제2 포인터 수신(150 단계)에 응답하여, 제3 컴포넌트(스토리지 에이전트(12))는 제3 헤더에 버퍼 컨텐츠를 제1 장치로 전송하는데 사용되는 장치 정보를 기록한다(152 단계). 이 장치 특유 정보는 제1 컴포넌트(스토리지 매니저 API(8))에 의해 기록된 제2 헤더 정보를 대체한다(replace). 스토리지 에이전트(12)는 버퍼(18)에 대한 제3 포인터를 스토리지 장치 드라이버(14)로 전송한다(154 단계). 응답하여, 스토리지 장치 드라이버(14)는 제3 포인터를 사용하여 버퍼(18) 컨텐츠에 접속하고, 스토리지 장치(16) 내의 스토리지 매체(예를 들어, 기술 분야에서 알려진 하드디스크 드라이브(HDD), 마그네틱 테이프 등의 마그네틱 스토리지 매체, 광학 스토리지 매체 또는 다른 적절한 비-휘발성 스토리지 매체)에 버퍼(18) 컨텐츠를 전송한다.

[0020]

추가적인 실시예에서, 일부 헤더들은 기록되지 않을 수 있다. 예를 들어, 어플리케이션(6) 또는 스토리지 매니저 API(8)가 하나의 헤더(44, 46)를 기록하지 않았다면, 제2 헤더는 verb 데이터(48)를 포함할 것이다. 대안적으로, 컴포넌트는 오프셋 계산을 더 필요로 하는 부가적인 헤더들을 기록할 수 있다. 어떤 기록된 헤더 앞의 오프셋은 빈 공간과 기록된 헤더가 4Kb 경계 상에 위치하도록 빈 공간을 이전에 기록된 헤더 전에 부가하는 것을 고려할 것이다. 이러한 방식으로, 빈 공간(42) 뒤에 오는 오프셋에 기록된 헤더(44)는 오프셋으로부터 빈 공간(42)의 시작까지 확장한다.

[0021]

일 실시예에서, 제1 오프셋은 제1 오프셋보다 크고, 제1 오프셋은 제3 오프셋보다 크며, 제3 오프셋은 제4 오프셋보다 크다. 이러한 방식으로, 데이터를 처리하는 각각의 다음 컴포넌트는, 부가적인 헤더를 이전에 기록된 어플리케이션 데이터와 버퍼에 기록된 헤더들 앞에 위치시키도록 하기 위해, 데이터에 대한 헤더를 오프셋에 기록할 수 있다.

[0022]

다른 실시예에서, 어플리케이션(6)은 스토리지 장치(16)로부터 데이터를 요청할 수 있다. 응답으로, 스토리지 장치 드라이버(14)는 데이터에 접속하고, 데이터를 스토리지 에이전트(12)에 반환한다. 그 후에, 스토리지 에이전트(12)(또는 어떤 다른 컴포넌트)는 검색된 데이터를 부가하기 위하여 스토리지 매니저 API(8)로부터 버퍼(18)를 요청할 수 있다. 그 후에, 컴포넌트들이 검색된 어플리케이션 데이터에 대한 헤더들을 버퍼(18)에 부가하도록 허용하는 버퍼에서의 공간(room)을 제공하는 오프셋을 제공하도록 하기 위해, 컴포넌트는 오프셋 계산을 수행할 수 있고, 그 결과 버퍼에 대한 포인터는 결국 데이터 접속을 위해 어플리케이션(6)으로 전달된다.

[0023]

설명된 실시예들에서는, 상이한 컴포넌트들이 버퍼를 공유하고 이 동일한 버퍼로부터 데이터에 접속하고 헤더 정보를 이 동일한 버퍼에 기록하고, 다른 컴포넌트들이 버퍼에 접속할 수 있도록 하기 위해 버퍼에 대한 포인터들을 다른 컴포넌트에 전달한다.

[0024]

다른 실시예들에서, 어플리케이션은 스토리지 매니저 API(8)로부터 복수의 버퍼들을 요청하고, 동시에 어플리케이션 데이터(40)와 어플리케이션 헤더(44) 데이터를 다중 버퍼들에 기록하고, 버퍼들에 대한 헨들을 후속 처리를 위하여 스토리지 매니저 API(8)로 전달할 수 있다.

[0025]

설명된 동작들은 소프트웨어, 펌웨어, 하드웨어, 또는 이들의 조합을 생산하기 위하여 표준 프로그래밍 및/또는 엔지니어링 기술들을 사용하는 제조 방법, 장치 또는 물품으로서 구현될 수 있다. 여기서 사용되는 것처럼 “

제조 물품(article of manufacture)"이라는 용어는 하드웨어 로직(예컨대, 접적회로 칩, 프로그램가능한 게이트 어레이(Programmable Gate Array, PGA), 주문형반도체(Application Specific Integrated Circuit, ASIC 등) 또는 마그네틱 스토리지 매체(예컨대, 하드 디스크 드라이브, 플로피 디스크, 테이프 등)와 같은 컴퓨터가 판독 가능한 매체, 광학 스토리지(CD-ROMs, 광 디스크 등), 휘발성 및 비휘발성 메모리 소자(예컨대, EEPROM, ROM, PROM, RAM, DRAM, SRAM, 프로그램가능한 로직 등)에서 구현되는 코드 또는 로직을 지칭한다. 컴퓨터가 판독 가능한 매체에서의 코드는 프로세서에 의해 접속되고 실행된다. 바람직한 실시예에서 구현되는 코드는 전송 매체를 통하여거나 또는 네트워크를 통한 파일 서버로부터 접속 가능하도록 구현될 수 있다. 그런 경우에, 코드가 구현되는 제조 물품은 네트워크 전송선과 같은 전송 매체, 무선 전송 매체, 공간을 통하여 전파하는 신호, 전자파(radio wave), 적외선 신호 등을 포함할 수 있다. 따라서, "제조 물품"은 코드가 구현되는 매체를 포함할 것이다. 추가적으로, "제조 물품"은 코드가 구현되고, 처리되고, 및 실행되는 하드웨어와 소프트웨어 컴포넌트의 조합을 포함할 수 있다. 물론, 본 발명의 범위를 벗어나지 않고, 이 구성에 다양한 변형들을 가할 수 있으며, 제조 물품은 이 기술 분야에서 알려진 모든 정보 보유 매체(information bearing medium)를 포함할 수 있음을 당업자에게 자명할 것이다.

[0026] 도 3 내지 도 5의 도시된 동작들은 특정 사건들이 특정 순서로 발생하는 것을 보여준다. 대안적인 실시예에서, 특정 동작은 변형되거나 또는 제거된 상이한 순서로 수행될 수 있다. 더욱이, 단계들이 앞에서 설명된 실시예들에 추가될 수 있고, 여전히 설명된 실시예들에 순응할 수도 있다. 또한, 여기서 설명된 동작들은 연속적으로 발생할 수 있고, 특정 동작들은 병렬적으로 처리될 수도 있다. 그러나 또한, 동작들은 하나의 단일 공정에 의해 또는 분산된 여러 공정 유닛들에 의해 수행될 수 있다.

[0027] 본 발명의 다양한 실시예들에 관한 설명은 예시와 설명의 목적을 위해 제공되었으며, 개시된 것과 동일한 형태로 제한되지는 않는다. 다양한 변형과 변화가 상기 개시된 것에 기초하여 가능하다. 본 발명의 범위가 여기서 상세히 설명된 것에 의해 제한되는 것이 아니라, 여기서 첨부된 청구항들에 의해 제한되어야 한다. 앞에서의 설명, 예제 및 데이터는 본 발명의 구성들의 제조 및 사용에 관한 완전한 설명을 제공한다. 본 발명의 다양한 실시예들이 본 발명의 범위와 정신을 벗어나지 않는 범위에서 본 발명의 다양한 실시예들이 구현될 수 있으며, 본 발명은 여기에서 첨부되는 특허청구범위에 속한다.

[0028] (부기 1) 어플리케이션에 의해, 컴포넌트로부터 버퍼를 요청하는 단계; 컴포넌트에 의해, 버퍼를 할당하는 단계; 상기 컴포넌트에 의해, 어플리케이션 데이터에 대한 버퍼의 제1 오프셋 및 제1 헤더에 대한 버퍼의 제2 오프셋을 어플리케이션으로 반환하는 단계; 상기 어플리케이션에 의해, 상기 어플리케이션 데이터를 상기 버퍼의 제1 오프셋에 기록하는 단계; 상기 어플리케이션에 의해, 상기 제1 헤더를 상기 버퍼의 제2 오프셋에 기록하는 단계; 및 제2 헤더를 상기 버퍼의 제3 오프셋에 기록하는 단계를 포함하는 방법.

[0029] (부기 2) 제1항에 있어서, 상기 제1 오프셋은 상기 제2 오프셋보다 크고, 상기 제2 오프셋은 상기 제3 오프셋보다 큰 것인 방법.

[0030] (부기 3) 제1항에 있어서, 상기 컴포넌트는 상기 제2 헤더를 상기 버퍼에 기록하며, 제3 헤더를 상기 버퍼의 제4 오프셋에 기록하는 단계를 더 포함하고, 상기 제1 오프셋은 상기 제2 오프셋보다 크고, 상기 제2 오프셋은 상기 제3 오프셋보다 크고, 상기 제3 오프셋은 상기 제4 오프셋보다 큰 것인 방법.

[0031] (부기 4) 제1항에 있어서, 상기 컴포넌트는 제1 컴포넌트를 포함하고, 제2 컴포넌트에 의해, 상기 제2 헤더를 버퍼 컨텐츠를 스토리지 장치로 전송하는데 사용되는 정보와 대체하는 단계를 더 포함하는 방법.

[0032] (부기 5) 제1항에 있어서, 상기 컴포넌트는 제1 컴포넌트를 포함하고, 상기 제3 오프셋의 제2 헤더는 어플리케이션 데이터 및 헤더를 포함하는 버퍼 컨텐츠가 제1 장치 또는 제2 장치로 전송되어야 하는지를 나타내고, 상기 제1 컴포넌트에 의해, 상기 제2 헤더에 대한 포인터를 제2 컴포넌트로 전달하는 단계; 상기 제2 컴포넌트에 의해, 상기 버퍼 컨텐츠를 제1 장치 또는 제2 장치로 전송할 것인지 여부를 결정하기 위해 상기 제2 헤더를 처리하는 단계; 및 상기 제2 헤더가 상기 버퍼 컨텐츠를 상기 제1 장치로 전송하는 것을 나타내는 것으로 결정되는 것에 응답하여, 상기 버퍼 컨텐츠를 상기 제1 장치로 전송하는데 사용되는 장치 정보를 상기 제2 헤더에 기록하는 단계를 포함하고, 상기 장치 정보는 상기 제1 컴포넌트에 의해 기록된 상기 제2 헤더 정보를 대체하는 것인 방법.

[0033] (부기 6) 제5항에 있어서, 상기 제1 장치는 스토리지 장치를 포함하고, 상기 스토리지 장치를 위한 스토리지 장치 드라이버에 의해, 상기 버퍼 컨텐츠가 상기 스토리지 장치로 전송되어야 하는 것으로 결정되는 것에 응답하여, 상기 버퍼 컨텐츠를 상기 버퍼로부터 상기 스토리지 장치로 전송하기 위해, 상기 제2 헤더의 상기 장치 정

보를 사용하는 단계를 더 포함하는 방법.

- [0034] (부기 7) 제6항에 있어서, 상기 제1 컴포넌트에 의해 전달되는 상기 포인터는 제1 포인터를 포함하고, 상기 제2 컴포넌트에 의해, 상기 버퍼 컨텐츠가 상기 스토리지 장치로 전송되어야 하는 것으로 결정되는 것에 응답하여, 상기 버퍼에 대한 제2 포인터를 상기 제2 헤더에 상기 장치 정보를 기록하는 제3 컴포넌트로 전송하는 단계; 및 상기 제3 컴포넌트에 의해, 상기 버퍼에 대한 제3 포인터를 상기 스토리지 장치의 스토리지 매체에 상기 버퍼의 컨텐츠를 기록하기 위해 상기 제3 포인터를 사용하는 상기 스토리지 장치 드라이버로 전송하는 단계를 더 포함하는 방법.
- [0035] (부기 8) 제7항에 있어서, 상기 제1 컴포넌트는 스토리지 매니저의 어플리케이션 프로그램 인터페이스(API)를 포함하고, 상기 제3 컴포넌트는 스토리지 에이전트를 포함하는 것인 방법.
- [0036] (부기 9) 제5항에 있어서, 상기 제2 장치는 스토리지 서버를 포함하고,
- [0037] 상기 제2 컴포넌트에 의해, 상기 제2 헤더가 상기 버퍼 컨텐츠를 상기 스토리지 서버에 전송하는 것을 나타내는 것으로 결정되는 것에 응답하여, 상기 버퍼 컨텐츠를 상기 스토리지 서버로 전송하는 단계를 더 포함하는 방법.
- [0038] (부기 10) 제1항에 있어서, 상기 컴포넌트에 의해, 상기 제1 오프셋으로부터의 어플리케이션 데이터의 기록 다음에, 상기 제1 및 제2 헤더를 기록하기 위한 상기 제1 오프셋 전에 버퍼에서 충분한 양의 공간을 제공하기 위하여 제1 오프셋을 계산하는 단계; 및 상기 컴포넌트에 의해, 상기 어플리케이션 데이터 및 상기 제1 헤더의 기록 다음에, 상기 제2 헤더를 기록하기 위한 상기 제2 오프셋 전에 버퍼에서 충분한 양의 공간을 제공하기 위하여 제2 오프셋을 계산하는 단계를 더 포함하는 방법.
- [0039] (부기 11) 제10항에 있어서, 상기 컴포넌트에 의해, 상기 어플리케이션의 데이터 및 상기 제1 헤더의 기록 다음에, 상기 제2 헤더를 기록하기 위한 상기 제3 오프셋 전에 버퍼에서 충분한 양의 공간을 제공하기 위하여 상기 제3 오프셋을 계산하는 단계를 더 포함하는 방법.
- [0040] (부기 12) 제1항에 있어서, 상기 어플리케이션 데이터가 고정된 숫자의 배수인지 여부를 결정하는 단계를 더 포함하고, 상기 제1 및 제2 오프셋은, 상기 어플리케이션 데이터가 상기 고정된 숫자의 배수가 아닌 것으로 결정되는 것에 응답하여 상기 어플리케이션 데이터와 빈 공간이 상기 고정된 숫자의 배수를 차지하도록 하기 위해 상기 제1 오프셋에 기록된 어플리케이션 데이터 앞에 빈공간이 유지되도록 선택되고, 상기 제1 헤더는 상기 제2 오프셋부터 상기 빈 공간의 시작까지 확장되는 것인 방법.
- [0041] (부기 13) 제12항에 있어서, 상기 제1 헤더가 상기 고정된 숫자의 배수인지 여부를 결정하는 단계를 더 포함하고, 상기 제3 오프셋은, 상기 제1 헤더가 상기 고정된 숫자의 배수가 아닌 것으로 결정되는 것에 응답하여, 상기 어플리케이션 데이터와 빈 공간이 상기 고정된 숫자의 배수를 차지하도록 하기 위해 상기 제1 헤더 앞에 빈 공간이 유지되도록 선택되고, 상기 제2 헤더는 상기 제3 오프셋부터 상기 제1 헤더 앞의 상기 사의 빈 공간의 시작까지 확장되는 것인 방법.
- [0042] (부기 14) 제12항에 있어서, 상기 고정된 숫자는 4Kb를 포함하고, 상기 제1 오프셋에서 상기 버퍼에 기록된 상기 어플리케이션 데이터가 상기 버퍼에서 4Kb 경계 상에 배치되지 않은 경우, 상기 어플리케이션 데이터는 상기 고정된 숫자의 배수가 아닌 것인 방법.
- [0043] (부기 15) 제1항에 있어서, 상기 어플리케이션에 의해, 복수의 버퍼들을 요청하는 단계; 상기 컴포넌트에 의해, 상기 복수의 버퍼들을 상기 어플리케이션에 반환하는 단계; 상기 어플리케이션에 의해, 어플리케이션 데이터 및 제1 헤더를 상기 복수의 버퍼들에 동시에 기록하는 단계를 더 포함하는 방법.
- [0044] (부기 16) 프로세서; 상기 프로세서에 접속 가능하고, 버퍼들을 구비하는 메모리; 및 어플리케이션 및 컴포넌트를 포함하는 컴퓨터로 판독 가능한 매체 내의 코드를 포함하고, 상기 프로세서는 상기 컴포넌트 및 상기 어플리케이션이 상기 청구항 제1항 내지 제15항 중 어느 한 항의 단계들을 수행하도록 상기 코드를 실행시키는 것인 시스템.
- [0045] (부기 17) 버퍼들을 포함하는 메모리와 통신하는 어플리케이션 및 컴포넌트를 포함하고, 상기 어플리케이션 및 상기 컴포넌트는 상기 청구항 제1항 내지 상기 제15항 중 어느 한 항에 따른 방법의 단계들을 수행하도록 실행될 수 있는 것인 컴퓨터 프로그램.

도면의 간단한 설명

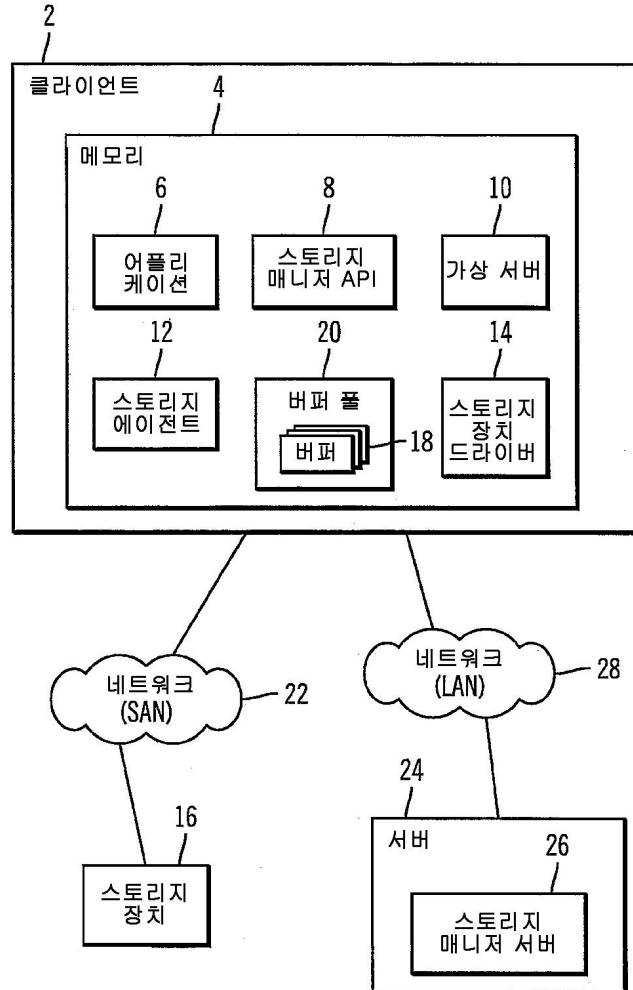
[0005] 도 1은 네트워크 컴퓨팅 환경의 실시예를 나타낸다.

[0006] 도 2는 버퍼 포맷의 실시예를 나타낸다.

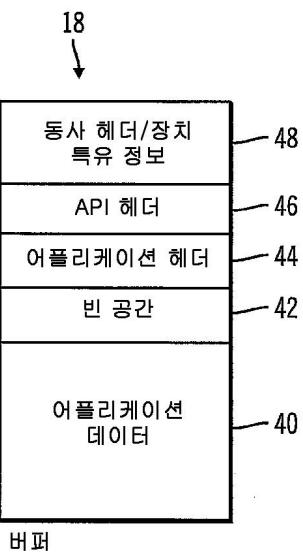
[0007] 도 3 내지 도 5는 데이터를 어플리케이션으로부터 스토리지 장치로 전송하는 동작들의 실시예를 나타낸다.

도면

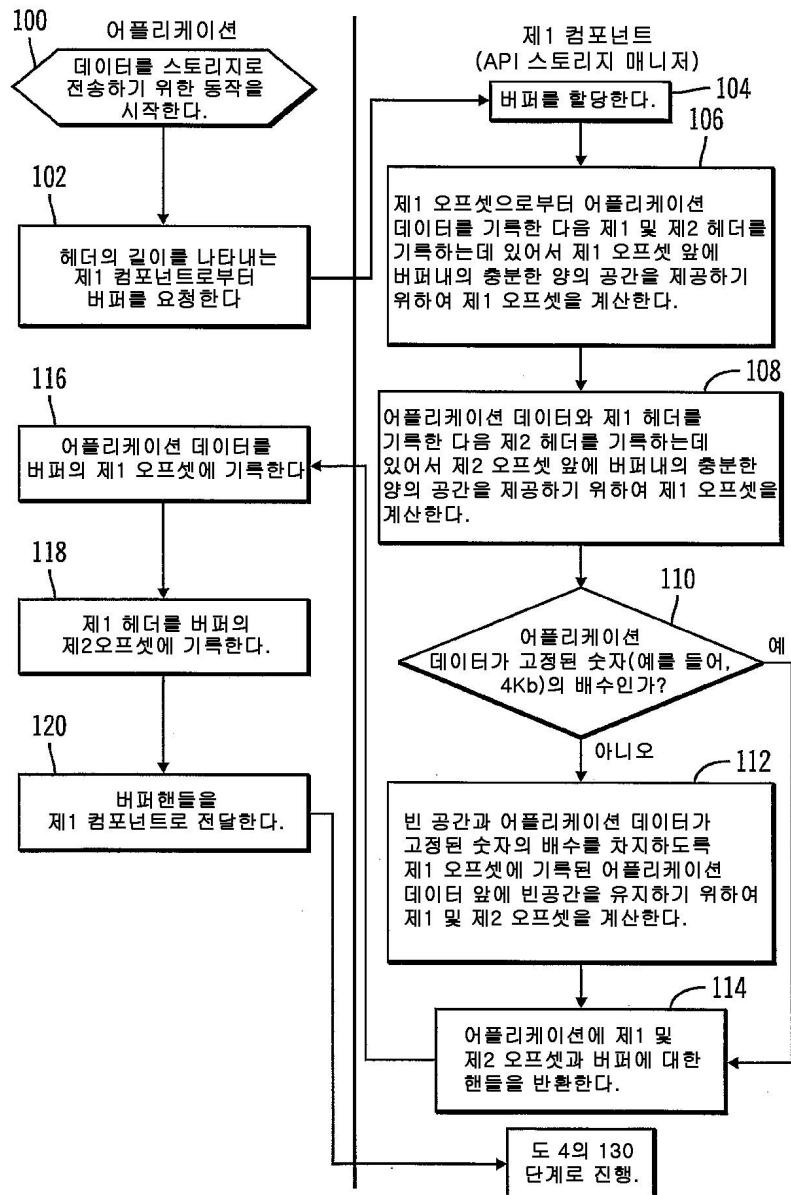
도면1



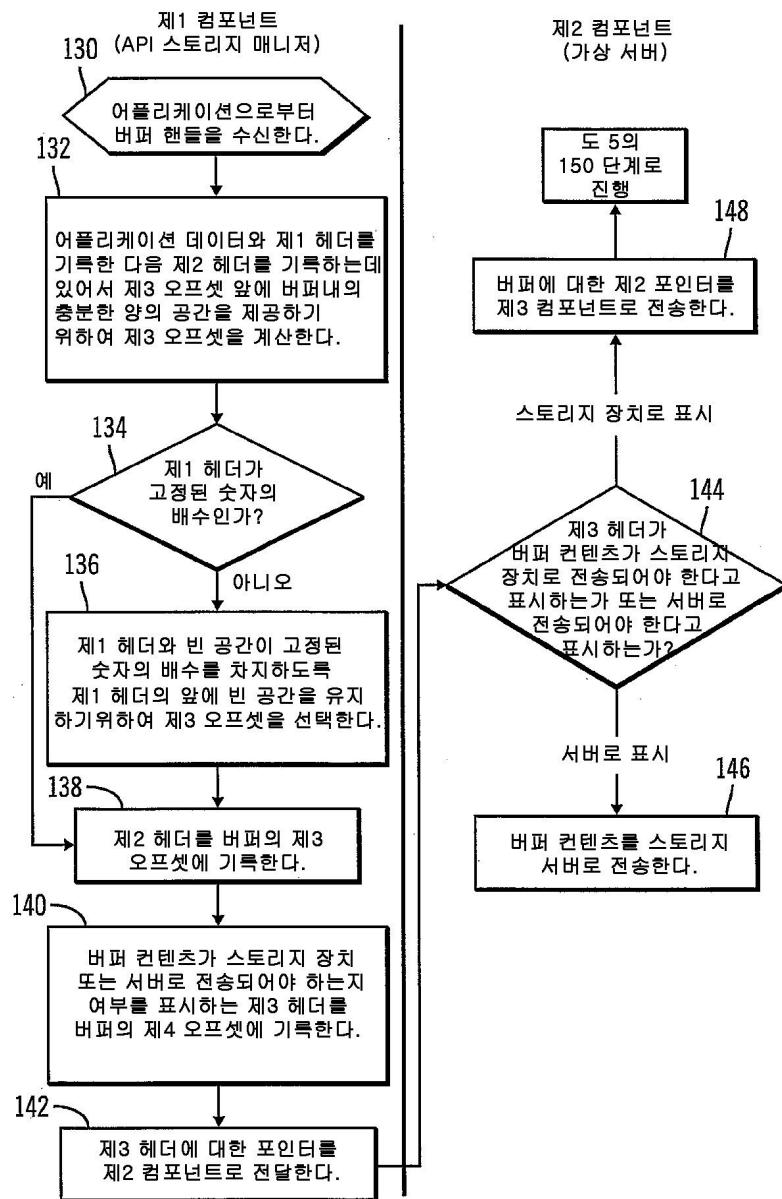
도면2



도면3



도면4



도면5

