



등록특허 10-2435046



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2022년08월22일
(11) 등록번호 10-2435046
(24) 등록일자 2022년08월17일

- (51) 국제특허분류(Int. Cl.)
G06F 3/06 (2006.01) *G06F 3/0482* (2022.01)
H03M 7/30 (2006.01) *HO4L 9/40* (2022.01)
- (52) CPC특허분류
G06F 3/0608 (2013.01)
G06F 3/0482 (2022.01)
- (21) 출원번호 10-2018-0052088
- (22) 출원일자 2018년05월04일
심사청구일자 2021년05월03일
- (65) 공개번호 10-2018-0138127
- (43) 공개일자 2018년12월28일
- (30) 우선권주장
62/522,423 2017년06월20일 미국(US)
(뒷면에 계속)
- (56) 선행기술조사문현
미국공개특허 제2013-0254441호(2013.09.26.) 1
부.*
미국공개특허 제2014-0215170호(2014.07.31.) 1
부.*
일본공개특허 제2009-070361호(2009.04.02.) 1
부.*
미국공개특허 제2005-0210054호(2005.09.22.) 1
부.*

*는 심사관에 의하여 인용된 문현

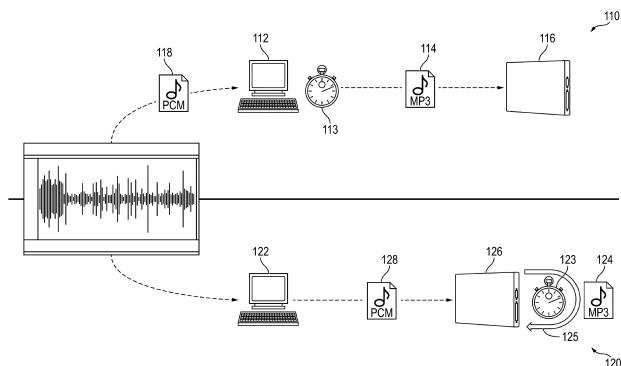
전체 청구항 수 : 총 15 항

심사관 : 김종기

(54) 발명의 명칭 손실 압축 드라이브**(57) 요약**

손실 데이터 압축 방법은, 스토리지 장치에서 원시 데이터를 수신하는 단계, 압축하는 요청을 플래그를 수신하는 단계, 손실 데이터 압축 방식들에 각각 대응하는 다양한 데이터 압축 알고리즘들을 포함하는 온보드(onboard) 데이터 압축 알고리즘 라이브러리에 액세스하는 단계, 파라미터들의 수에 근거해서 상기 데이터 압축 알고리즘들

(뒷면에 계속)

대 표 도

중 하나를 선택하는 단계, 선택된 데이터 알고리즘을 온라인 또는 오프라인 중 하나로 실행하는 단계를 포함하여, 상기 온라인은 상기 원시 데이터가 수신될 때 상기 원시 데이터를 압축된 데이터로서 상기 스토리지 장치에 저장하여 상기 원시 데이터가 상기 스토리지 장치에 의해서 압축되는 것이고, 상기 오프라인은 상기 원시 데이터가 상기 스토리지 장치에 저장되며, 선택된 데이터 압축 알고리즘에 따라서 상기 스토리지 장치에 의해 나중에 압축되며, 압축된 데이터로서 상기 스토리지 장치에 재저장되는 것이다.

(52) CPC특허분류

G06F 3/0659 (2013.01)

H03M 7/6064 (2013.01)

H04L 69/04 (2022.05)

(30) 우선권주장

62/542,008 2017년08월07일 미국(US)

15/728,261 2017년10월09일 미국(US)

명세서

청구범위

청구항 1

스토리지 장치에서 원시 데이터를 수신하는 단계;

압축 요청을 수신하는 단계;

손실 데이터 압축 방식들에 각각 대응하는 다양한 데이터 압축 알고리즘들을 포함하는 데이터 압축 알고리즘 라이브러리에 액세스하는 단계;

상기 스토리지 장치 상의 엔트로피 추정 알고리즘 라이브러리에 액세스하는 단계;

상기 원시 데이터의 엔트로피를 테스트하여 상기 원시 데이터의 압축률 레벨을 판별하는 단계;

상기 스토리지 장치의 장치 성능 프로파일의 분석에 기반하여 상기 스토리지 장치에 대응하는 가용 시간 또는 자원들을 추정하는 단계;

상기 원시 데이터에 대응하는 키-값 쌍의 키와 관련된 메타데이터에 근거해서 상기 데이터 압축 알고리즘들 중 하나를 선택하는 단계;

선택된 데이터 압축 알고리즘을 실행하는 단계를 포함하되;

상기 스토리지 장치는, 상기 수신된 원시 데이터가 상기 스토리지 장치에 의해서 압축되고, 압축된 데이터로서 상기 스토리지 장치에 저장되도록 상기 선택된 데이터 압축 알고리즘이 온라인으로 실행되는 제 1 모드에 따라 상기 데이터 압축 알고리즘을 수행하거나, 또는 상기 원시 데이터가 상기 스토리지 장치에 저장되고, 상기 선택된 데이터 압축 알고리즘에 따라 상기 스토리지 장치에 의해 압축되고, 압축된 데이터로서 상기 스토리지 장치에 재저장되도록 상기 선택된 데이터 압축 알고리즘이 오프라인으로 실행되는 제 2 모드에 따라 상기 데이터 압축 알고리즘을 수행하고,

상기 장치 성능 프로파일은 상기 스토리지 장치의 진행 중인 물리적 성능의 기록에 대응하는 정보를 포함하고, 그리고

상기 선택된 데이터 압축 알고리즘이 상기 제 1 모드에 따라 수행될 것인지 또는 상기 제 2 모드에 따라 수행될 것인지 여부는 상기 추정된 가용 시간 또는 자원들에 기반하여 결정되고, 그리고

상기 스토리지 장치에 대응하는 가용 시간 또는 자원들을 추정하는 단계는 머신 러닝 예측들에 더 기반하는 손실 데이터 압축 방법.

청구항 2

제 1 항에 있어서,

상기 메타데이터를 제공하기 위해 상기 원시 데이터를 압축하도록 사용자 요청을 유도하는 단계를 더 포함하는 손실 데이터 압축 방법.

청구항 3

제 1 항에 있어서,

객체 서명 또는 객체 확장자에 대응하는 상기 원시 데이터와 관련된 상기 메타데이터를 읽어오는 단계;

상기 스토리지 장치 상의 객체 서명 사전 또는 상기 스토리지 장치 상의 객체 확장 사전을 액세스하여 상기 객체 서명 또는 상기 객체 확장자에 각각 근거해서 상기 원시 데이터의 데이터 탑입을 판별하는 단계; 및

상기 판별된 데이터 탑입에 근거해서 상기 데이터 압축 알고리즘을 선택하는 단계를 더 포함하는 손실 데이터 압축 방법.

청구항 4

스토리지 장치에서 원시 데이터를 수신하는 단계;

상기 원시 데이터에 대응하는 키-값 쌍의 키의 메타데이터를 읽어오는 단계;

상기 스토리지 장치 상의 엔트로피 추정 알고리즘 라이브러리에 액세스하는 단계;

상기 원시 데이터의 엔트로피를 테스트하여 상기 원시 데이터의 압축률 레벨을 판별하는 단계;

상기 스토리지 장치의 장치 성능 프로파일의 분석에 기반하여 상기 스토리지 장치에 대응하는 가용 시간 또는 자원들을 추정하는 단계;

상기 메타데이터에 따라서 상기 스토리지 장치 상의 데이터 압축 알고리즘 라이브러리로부터 데이터 압축 알고리즘을 선택하는 단계;

상기 추정된 가용 시간 또는 자원들에 기반하여 상기 원시 데이터를 압축할지 여부를 결정하는 단계;

상기 스토리지 장치와 함께, 상기 선택된 데이터 압축 알고리즘에 따라서 상기 원시 데이터를 압축하기 위한 데이터 압축 방식을 수행하여 압축된 데이터를 생성하는 단계; 및

상기 압축된 데이터를 상기 스토리지 장치 상에 저장하는 단계를 포함하되,

상기 장치 성능 프로파일은 상기 스토리지 장치의 진행 중인 물리적 성능의 기록에 대응하는 정보를 포함하고,

상기 스토리지 장치에 대응하는 가용 시간 또는 자원들을 추정하는 단계는 머신 러닝 예측들에 더 기반하는 데이터 압축 방법.

청구항 5

삭제

청구항 6

제 4 항에 있어서,

압축 방식들 및 파라미터들에 관한 정보를 포함하는 서비스 품질(QoS) 우선 데이터 파일을 마련하는 단계;

상기 스토리지 장치에 의해서, 상기 원시 데이터에 대응하는 객체 파일을 식별하는 단계; 및

상기 정보 및 상기 객체 파일에 따라서 상기 원시 데이터를 압축하기 위한 데이터 압축 방식을 결정하는 단계를 더 포함하는 데이터 압축 방법.

청구항 7

제 4 항에 있어서,

상기 원시 데이터는 객체 서명 또는 객체 확장자를 포함하고,

상기 방법은,

상기 스토리지 장치 상의 객체 서명 사전 또는 상기 스토리지 장치 상의 객체 확장자 사전에 액세스하여 상기 객체 서명 또는 상기 객체 확장자 각각에 근거해서 상기 원시 데이터의 데이터 타입을 판별하는 단계; 및

상기 판별된 데이터 타입에 근거해서 상기 데이터 압축 방식을 수행하는 단계를 더 포함하는 데이터 압축 방법.

청구항 8

제 4 항에 있어서,

사용자 요청에 따라서 호스트 컴퓨터로부터 상기 스토리지 장치로 상기 원시 데이터를 전송하는 단계를 더 포함하는 데이터 압축 방법.

청구항 9

제 8 항에 있어서,

상기 사용자가 정보를 제공하도록 유도하기 위한 메뉴를 제공하는 단계; 및

상기 사용자 제공 정보에 따라 상기 원시 데이터를 압축하기 위한 데이터 압축 알고리즘을 결정하는 단계를 더 포함하는 데이터 압축 방법.

청구항 10

제 4 항에 있어서,

상기 스토리지 장치에서 상기 원시 데이터를 수신하는 단계 이후에 상기 스토리지 장치 상에 상기 원시 데이터를 저장하는 단계를 더 포함하는 데이터 압축 방법.

청구항 11

호스트; 및

상기 호스트로부터 원시 데이터를 수신하도록 구성된 스토리지 장치를 포함하며,

상기 스토리지 장치는,

메모리; 및

명령들을 실행하도록 구성된 프로세서를 포함하며,

상기 명령들이 상기 프로세서에 의해 실행될 때 상기 프로세서로 하여금:

상기 원시 데이터에 대응하는 키-값 쌍의 키의 메타데이터를 읽어오도록 하고,

상기 스토리지 장치 상의 엔트로피 추정 알고리즘 라이브러리에 액세스하도록 하고,

상기 원시 데이터의 엔트로피를 테스트하여 상기 원시 데이터의 압축률 레벨을 판별하도록 하고,

상기 스토리지 장치의 장치 성능 프로파일의 분석에 기반하여 상기 스토리지 장치에 대응하는 가용 시간 또는 자원들을 추정하도록 하고,

상기 메타데이터에 따라서 상기 스토리지 장치 상의 데이터 압축 알고리즘 라이브러리로부터 데이터 압축 알고리즘을 선택하도록 하고,

상기 추정된 가용 시간 또는 자원들에 기반하여 상기 원시 데이터를 압축할지 여부를 결정하도록 하고,

상기 선택된 데이터 압축 알고리즘에 따라서 상기 원시 데이터를 압축하기 위한 데이터 압축 방식을 수행하여 압축된 데이터를 생성하도록 하고, 그리고

상기 압축된 데이터를 상기 스토리지 장치 상에 저장하도록 야기하되,

상기 장치 성능 프로파일은 상기 스토리지 장치의 진행 중인 물리적 성능의 기록에 대응하는 정보를 포함하고,

상기 스토리지 장치에 대응하는 가용 시간 또는 자원들을 추정하는 것은 머신 러닝 예측들에 더 기반하는 데이터 압축 시스템.

청구항 12

삭제

청구항 13

제 11 항에 있어서,

상기 명령들은, 상기 프로세서에 의해 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가

압축 방식들 및 파라미터들에 관한 정보를 저장하는 서비스 품질(QoS) 우선 데이터 파일을 액세스하고;

상기 원시 데이터에 대응하는 객체 파일을 식별하고; 그리고

상기 정보 및 상기 객체 파일에 따라서 상기 원시 데이터를 압축하기 위한 상기 데이터 압축 방식을 결정하도록 더 야기하는 데이터 압축 시스템.

청구항 14

제 11 항에 있어서,

상기 원시 데이터는 객체 서명 또는 객체 확장자를 포함하고, 그리고

상기 명령들은, 상기 프로세서에 의해 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가

상기 스토리지 장치 상의 객체 서명 사전 또는 상기 스토리지 장치 상의 객체 확장자 사전을 액세스해서 상기 객체 서명 또는 상기 객체 확장자 각각에 근거해서 상기 원시 데이터의 데이터 타입을 결정하고; 그리고

상기 결정된 데이터 타입에 근거해서 상기 데이터 압축 방식을 수행하도록 더 야기하는 데이터 압축 시스템.

청구항 15

제 11 항에 있어서,

상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가 사용자 요청에 따라서 호스트 컴퓨터로부터 상기 원시 데이터를 수신하도록 더 야기하는 데이터 압축 시스템.

청구항 16

제 15 항에 있어서,

상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서로 하여금

압축될 상기 원시 데이터를 요청하는 사용자가 정보를 제공하도록 유도하는 메뉴를 호스트가 제공하도록, 그리고

상기 사용자 제공 정보에 따라서 상기 스토리지 장치가 상기 원시 데이터를 압축하기 위한 데이터 압축 알고리즘을 결정하도록 더 야기하는 데이터 압축 시스템.

청구항 17

제 11 항에 있어서,

상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가

상기 원시 데이터를 수신한 이후 상기 원시 데이터를 상기 스토리지 장치 상에 저장하도록 더 야기하는 데이터 압축 시스템.

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

발명의 설명

기술 분야

[0001] 본 발명은 데이터 압축 기능을 포함하는 드라이브 장치에 관한 것이다.

배경 기술

[0002] 데이터 압축은 하드 스토리지 장치와 같은 데이터 스토리지 용량의 효율적인 사용을 증가시키기 위한 잘 알려진 유용한 기술이다. 그러나, 일부 데이터 압축 방식들 또는 알고리즘들은 다른 것들 보다 특정 타입들의 데이터에 대해 더 유리하게 작용한다. 데이터 압축은 압축된 버전으로부터 정확한 원본 데이터가 획득될 수 있는 "무손실(lossless)" 데이터 압축 또는 압축된 버전으로부터 정확한 입력 데이터가 복원될 수 없는 "손실(lossy)" 데이터

터 압축 중 하나일 수 있다.

[0003] 일반적인 데이터 압축은 호스트에서 발생하고, 압축된 데이터는 압축된 데이터 파일로서 드라이브(예를 들면, 하드 스토리지 드라이브(hard storage device, HDD) 또는 솔리드 스테이트 드라이브(solid state drive, SSD)로 전송된다. 그러나 온-드라이브(on-drive) 데이터 압축은 드라이브가 원시(raw) 압축되지 않은 데이터 파일을 호스트로부터 수신하고, 수신된 데이터를 압축한 다음 압축된 데이터 파일을 저장하는 것이다.

[0004] 종래의 온-드라이브 데이터 압축 방법들은 데이터를 "덤리(dumbly)" 압축하거나, 드라이브에 의해 수신되는 모든 원시 데이터 파일이 동일한 데이터 압축 방법 또는 알고리즘을 사용하여 압축되는 비 지능형(unintelligent)/과도하게 단순화된(oversimplified) 데이터 압축을 수행한다. 그 다음, 드라이브는 압축된 데이터를 압축된 데이터 파일로서 저장한다. 즉, 비 지능형 온-드라이브 데이터 압축은 드라이브가 호스트로부터 수신된 모든 데이터를 가져와서 동일한 데이터 압축 알고리즘으로 모든 데이터를 압축한 다음 드라이브에 압축된 데이터 파일을 저장한다는 것을 의미한다.

[0005] 비 지능형 데이터 압축을 사용할 때, 드라이브는, 데이터가 호스트에 의해 이미 압축되었는지의 여부, 데이터가 이전에 암호화되었는지의 여부, 드라이브의 사용자가 어떤 등급 또는 어떤 파라미터들에 따라서 데이터가 압축되기를 원하는지의 여부, 및 다양한 다른 요소들 등을 나타내는 특성들과 같은 데이터의 구별되는 특성을 무시한다. 또한 비 지능형 데이터 압축이 사용될 때, 드라이브는, 특정 타입의 데이터는 다른 타입의 데이터 보다 하나 이상의 특정 데이터 압축 방식들에 따라 더 쉽게 압축된다는 사실을 무시한다.

[0006] 이전에 구현된 "덤(dumb)" 또는 비 지능형 데이터 압축(예를 들어, 일반적인 데이터 압축)의 버전들은 드라이브 상에 데이터를 저장하고자 하는 사용자가 중요한 데이터의 손실을 초래하는 손실 데이터 압축 방식을 원하지 않는다고 가정하여 동작하였다. 따라서 이전에 구현된 지능형 데이터 압축 버전들은 손실 데이터 압축과 달리 무손실 데이터 압축을 항상 포함한다. 즉, 손실 데이터 압축은 특정 타입들의 데이터(예를 들면, 데이터베이스, 작업 데이터 파일들 등)에 해로울 수 있으며, 잠재적으로 데이터 손상을 초래할 수 있으므로 데이터의 무결성을 유지하기 위해 무손실 데이터 압축 방식들 만이 구현되었다.

[0007] 또한, 이전에 구현된 손실 데이터 압축 방식들은 항상 호스트에 의해 수행되었고, 데이터는 드라이브에 의해 압축되지 않았다. 대신, 드라이브는 호스트에서 전송된 압축된 데이터 파일을 단순히 수신하고, 저장한다. 호스트는 일반적으로 데이터 압축 및 저장을 시도하는 사용자의 중앙 처리 장치(CPU)/컴퓨터 프로세서에 대응한다. 불행하게도, 그러한 손실 데이터 압축과 관련된 컴퓨팅 동작들은 일반적으로 많은 수의 CPU 사이클들을 필요로 하며, 또한 대응하는 많은 양의 에너지를 사용한다.

[0008] 상기 정보는 단지 개시의 배경에 대한 이해를 높이기 위해서 개시된 것이며, 종래 기술을 구성하지 않는 정보를 포함할 수 있다.

발명의 내용

해결하려는 과제

[0009] 본 발명은 호스트의 도움 없이 손실 데이터 압축을 지능적으로 수행할 수 있는 드라이브를 제공한다.

과제의 해결 수단

[0010] 본 개시의 실시예들은 호스트의 도움 없이 손실 데이터 압축을 지능적으로 수행할 수 있는 드라이브를 제공한다.

[0011] 일 실시예에 따르면, 손실 데이터 압축 방법은 스토리지 장치에서 원시 데이터를 수신하는 단계, 압축을 요청하는 플래그를 수신하는 단계, 손실 데이터 압축 방식들에 각각 대응하는 다양한 데이터 압축 알고리즘들을 포함하는 온보드(onboard) 데이터 압축 알고리즘 라이브러리에 액세스하는 단계, 파라미터들의 수에 근거해서 상기 데이터 압축 알고리즘들 중 하나를 선택하는 단계, 선택된 데이터 알고리즘을 온라인 또는 오프라인 중 하나로 실행하는 단계를 포함한다. 상기 온라인은 상기 원시 데이터가 수신될 때 상기 원시 데이터를 압축된 데이터로서 상기 스토리지 장치에 저장하여 상기 원시 데이터가 상기 스토리지 장치에 의해 압축되는 것이고, 상기 오프라인은 상기 원시 데이터가 상기 스토리지 장치에 저장되며, 선택된 데이터 압축 알고리즘에 따라서 상기 스토리지 장치에 의해 나중에 압축되며, 압축된 데이터로서 상기 스토리지 장치에 재저장되는 것이다.

[0012] 상기 파라미터들은 상기 원시 데이터에 대응하는 키-값 쌍의 향상된 키 내 또는 상기 스토리지 장치와 연관된

서비스 품질(QoS) 우선 데이터 파일 내 포함될 수 있다.

[0013] 상기 방법은 상기 파라미터들을 제공하기 위해 상기 원시 데이터를 압축하도록 사용자 요청을 유도하는 단계를 더 포함할 수 있다.

[0014] 상기 방법은 객체 서명 또는 객체 확장자에 대응하는 상기 원시 데이터 내 포함된 메타데이터를 읽어오는 단계, 상기 스토리지 장치 상의 객체 서명 사전 또는 상기 스토리지 장치 상의 객체 확장 사전을 액세스하고, 상기 객체 서명 또는 상기 객체 확장자에 각각 근거해서 상기 원시 데이터의 데이터 타입을 판별하는 단계 그리고 상기 판별된 데이터 타입에 근거해서 상기 데이터 압축 알고리즘을 선택하는 단계를 더 포함할 수 있다.

[0015] 다른 실시예에 따르면, 데이터 압축 방법은, 스토리지 장치에서 원시 데이터를 수신하는 단계, 상기 스토리지 장치와 함께 손실 데이터 압축 방식에 따라서 상기 원시 데이터를 압축하고, 압축된 데이터를 생성하는 단계 그리고 상기 압축된 데이터를 상기 스토리지 장치 상에 저장하는 단계를 포함한다.

[0016] 상기 방법은, 상기 스토리지 장치와 함께, 상기 원시 데이터 압축을 위한 데이터 압축 방식을 결정하는 단계를 더 포함할 수 있다.

[0017] 상기 방법은 압축 방식들 및 파라미터들에 관한 정보를 포함하는 서비스 품질(QoS) 우선 데이터 파일을 마련하는 단계, 상기 스토리지 장치에 의해서, 상기 원시 데이터에 대응하는 객체를 식별하는 단계 및 상기 정보 및 상기 객체 파일에 따라서 상기 원시 데이터 압축을 위한 데이터 압축 방식을 결정하는 단계를 더 포함할 수 있다.

[0018] 상기 방법은 상기 원시 데이터에 대응하는 키-값 쌍의 키의 메타데이터를 읽어오는 단계, 상기 메타데이터에 따라서 상기 스토리지 장치 상의 데이터 압축 알고리즘 라이브러리로부터 데이터 압축 알고리즘을 선택하는 단계 및 상기 선택된 데이터 압축 알고리즘에 따라서 상기 원시 데이터 압축을 위한 데이터 압축 방식을 수행하는 단계를 더 포함할 수 있다.

[0019] 상기 원시 데이터는 객체 서명 또는 객체 확장자를 포함하고, 상기 방법은, 상기 스토리지 장치 상의 객체 서명 사전 또는 상기 스토리지 장치 상의 객체 확장자 사전에 액세스하고, 상기 객체 서명 또는 상기 객체 확장자 각각에 근거해서 상기 원시 데이터 타입을 판별하는 단계 및 상기 판별된 데이터 타입에 근거해서 데이터 압축 방식을 수행하는 단계를 더 포함할 수 있다.

[0020] 상기 방법은 사용자 요청에 따라서 호스트 컴퓨터로부터 상기 스토리지 장치로 상기 원시 데이터를 전송하는 단계를 더 포함할 수 있다.

[0021] 상기 방법은 사용자가 정보를 제공하도록 유도하기 위한 메뉴를 제공하는 단계 및 상기 사용자 제공 정보에 따라 상기 원시 데이터를 압축하기 위한 데이터 압축 알고리즘을 결정하는 단계를 더 포함할 수 있다.

[0022] 상기 방법은 상기 스토리지 장치 상에 상기 원시 데이터를 저장하는 단계, 상기 스토리지 장치에 대응하는 가용 가능한 시간 또는 자원들을 추정하는 단계 및 충분한 시간 및 자원들이 가용한 것으로 판별되면 상기 원시 데이터를 압축하는 단계를 더 포함할 수 있다.

[0023] 다른 실시예에 따르면, 데이터 압축 시스템은, 호스트 및 상기 호스트로부터 원시 데이터를 수신하도록 구성된 스토리지 장치를 포함하며, 상기 스토리지 장치는, 메모리 및 명령들을 실행하도록 구성된 프로세서를 포함하며, 상기 명령들이 상기 프로세서에 의해 실행될 때 상기 프로세서로 하여금, 상기 스토리지 장치가 손실 데이터 압축 방식에 따라서 상기 원시 데이터를 압축해서 압축된 데이터를 생성하도록 하고 그리고 상기 스토리지 장치가 상기 압축된 데이터를 상기 스토리지 장치 상에 저장하도록 야기하는 것을 특징으로 하는 데이터 압축 시스템.

[0024] 상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가 상기 원시 데이터를 압축하기 위한 데이터 압축 방식을 결정하도록 더 야기한다.

[0025] 상기 명령들은, 상기 프로세서에 의해 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가 압축 방식들 및 파라미터들에 관한 정보를 저장하는 서비스 품질(QoS) 우선 데이터 파일을 액세스하고, 상기 원시 데이터에 대응하는 객체 파일을 식별하고 그리고 상기 정보 및 상기 객체 파일에 따라서 상기 원시 데이터를 압축하기 위한 상기 데이터 압축 방식을 결정하도록 더 야기할 수 있다.

[0026] 상기 명령들은, 상기 프로세서에 의해 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가 상기 원시 데이터에 대응하는 키-값 쌍의 키의 메타데이터를 읽어오고, 상기 메타데이터에 따라서 데이터 압축 알고리즘 라

이브리리로부터 데이터 압축 알고리즘을 선택하고 그리고 상기 선택된 데이터 압축 알고리즘에 따라서 상기 원시 데이터를 압축하기 위한 상기 데이터 압축 방식을 수행하도록 더 야기할 수 있다.

[0027] 상기 원시 데이터는 객체 서명 또는 객체 확장자를 포함하고, 그리고 상기 명령들은, 상기 프로세서에 의해 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가 상기 스토리지 장치 상의 객체 서명 사전 또는 상기 스토리지 장치 상의 객체 확장자 사전을 액세스해서 상기 객체 서명 또는 상기 객체 확장자 각각에 근거해서 상기 원시 데이터의 데이터 탑입을 결정하고, 그리고 상기 결정된 데이터 탑입에 근거해서 상기 데이터 압축 방식을 수행하도록 더 야기할 수 있다.

[0028] 상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가 사용자 요청에 따라서 호스트 컴퓨터로부터 상기 원시 데이터를 수신하도록 더 야기 할 수 있다.

[0029] 상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서로 하여금 사용자가 압축될 상기 원시 데이터를 요청하여 정보를 제공하도록 유도하는 메뉴를 호스트가 제공하도록, 그리고 상기 사용자 제공 정보에 따라서 상기 스토리지 장치가 상기 원시 데이터를 압축하기 위한 데이터 압축 알고리즘을 결정하도록 더 야기할 수 있다.

[0030] 상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서로 하여금 상기 스토리지 장치가 상기 원시 데이터를 상기 스토리지 장치 상에 저장하고 상기 스토리지 장치에 대응하는 가용 시간 및 자원들을 추정하고 그리고 충분한 시간 및 자원들이 가용가능 할 때 상기 원시 데이터를 압축하도록 더 야 할 수 있다.

[0031] 개시된 실시예들에 따른 드라이브는 원시 데이터를 수신하고, 원시 데이터를 수신할 수 있고, 원시 데이터에 대해 손실 데이터 압축을 수행함으로써, 사용자가 데이터 압축에 대응하는 특정 파라미터들에 대한 다양한 제어 레벨들을 허용한다. 따라서 드라이브는 손실 데이터 압축 작업들을 오프로딩(offloading)하여 호스트 CPU 사이클들을 자유롭게 할 수 있고, 그렇지 않으면 사용자 시스템에 상대적으로 많은 부하가 생길 수 있다. 일부 실시예들에서, 드라이브는 드라이브 상에 추가 또는 증가된 하드웨어 지원들을 구현할 수 있거나, 오퍼레이팅 시스템(OS) 레벨 및/또는 드라이버 레벨 변경을 구현할 수 있다.

발명의 효과

[0032] 본 개시의 실시예들은 데이터 압축의 실행에서 대응하는 호스트의 도움 없이 손실 데이터 압축 및 저장을 수행하여 다른 사용자 활동들에 대한 호스트의 CPU를 자유롭게 할 수 있는 SSD(solid state drive)와 같은 스토리지 장치를 제공할 수 있다.

도면의 간단한 설명

[0033] 일부 실시예들은 첨부된 도면들과 함께 다음의 설명으로부터 보다 상세히 이해 될 수 있다.

도 1은 종래의 손실 데이터 압축 방법의 예시와 본 개시의 실시예에 따른 손실 데이터 압축 방법의 예시를 비교해서 도시한다.

도 2는 본 개시의 실시예에 따른 호스트의 도움없이 드라이브에서 지능형 손실 데이터 압축을 수행할 수 있는 손실 압축 드라이브를 도시한다.

발명을 실시하기 위한 구체적인 내용

[0034] 본 발명의 개념 및 그 실시 방법의 특징들은 다음의 실시예들 및 첨부 도면들의 상세한 설명을 참조하면 보다 쉽게 이해될 수 있다. 이하, 첨부된 도면들을 참조하여 실시예들은 상세히 설명될 것이며, 동일한 참조 번호는 동일한 요소를 지칭한다. 그러나, 본 발명은 다양한 형태들로 구체화될 수 있으며, 여기에 도시된 실시예들에만 한정되는 것으로 해석되어서는 안된다. 오히려, 이들 실시예들은 본 개시들이 철저하고 완전하게 이루어 지도록 당업자에게 본 발명의 양상들 및 특징들을 충분히 전달할 수 있도록 예로서 제공된다. 따라서, 본 발명의 양상들 및 특징들의 완전한 이해를 위해 당업자에게 불필요한 프로세스들, 요소들, 및 기술들은 설명되지 않을 수 있다. 다른 언급이 없는 한, 첨부된 도면들 및 상세한 설명 전반에 걸쳐 동일한 참조 부호들은 동일한 요소들을 나타내며, 따라서 그 설명은 반복되지 않을 수 있다. 도면들에서, 엘리먼트들, 충들 및 영역들의 상대적 크기들은 명확성을 위해 과장될 수 있다.

[0035] 이하 설명에서, 설명의 목적으로, 많은 특정 상세들이 다양한 실시예들의 완전한 이해를 제공하기 위해 설명된다. 그러나, 다양한 실시예들이 이러한 특정 세부 사항없이 또는 하나 이상의 등가의 구성없이 실시될 수 있음

은 자명하다. 다른 예들에서, 공지된 구조들 및 장치들은 불필요하게 다양한 실시예들을 모호하게 하는 것을 피하기 위해 블록도 형태로 도시된다.

[0036] 엘리먼트, 계층, 영역 또는 구성 요소가 다른 엘리먼트, 계층, 영역 또는 컴포넌트에 "연결된" 또는 "결합된" 것으로 언급될 때, 이는 다른 엘리먼트, 계층, 영역 또는 구성 요소에 직접적으로 연결되거나, 결합될 수 있거나, 또는 하나 이상의 개체된 엘리먼트들, 계층들, 영역들 또는 구성 요소들이 존재할 수 있다. 그러나, "직접적으로 연결된/직접적으로 결합된"은 하나의 구성 요소가 중간 구성 요소없이 다른 구성 요소에 직접적으로 연결 또는 결합되는 것을 가리킨다. 한편, "사이에", "바로 사이에" 또는 "인접한" 그리고 "직접적으로 인접한"과 같은 구성 요소들 간의 관계를 설명하는 표현들은 유사하게 해석될 수 있다. 또한, 하나의 엘리먼트 또는 계층이 2 개의 엘리먼트들 또는 계층들의 "사이에" 있는 것으로 언급될 때, 2 개의 엘리먼트들 또는 층들 사이의 유일한 엘리먼트 또는 계층, 또는 하나 이상의 중간에 있는 엘리먼트들 또는 계층들이 존재할 수도 있다.

[0037] 본 명세서에서 사용되는 용어는 특정 실시 양태를 설명하기 위한 것일 뿐 본 발명을 제한하려는 것은 아니다. 본원에서 사용된 단수 형태는 문맥 상 다르게 지시하지 않는 한 복수 형태를 포함하는 것으로 의도된다. 본 명세서에서 사용된 "포함하는", "포함된", "구비하는" 및 "구비된"의 용어는 명시된 특징들, 정수들, 단계들, 동작들, 엘리먼트들 및/또는 구성 요소들의 존재를 특정하며, 하나 이상의 다른 특징들, 정수들, 단계들, 동작들, 엘리먼트들, 구성 요소 및/또는 이들의 그룹의 존재 또는 추가를 배제하지 않음이 잘 이해할 것이다. 본 명세서에 사용된 바와 같이, "및/또는"이라는 용어는 하나 이상의 관련 열거 된 항목의 임의 및 모든 조합을 포함한다. "적어도 하나"와 같은 표현들은 엘리먼트들 앞에서 엘리먼트들의 전체 목록을 수식하고 목록의 개별 엘리먼트들을 수식하지 않는다.

[0038] 본 명세서에 사용된 바와 같이, 용어 "실질적으로", "약" 및 유사한 용어들은 근사의 용어로서 사용되며 정도 (degree)의 용어로서 사용되지 않으며, 측정된 또는 계산된 값의 고유한 변화를 설명하기 위한 것이며, 이는 당업자에 의해 인식될 것이다. 본 명세서에서 사용되는 "약" 또는 "대략"은 언급된 값을 포함하며, 당해 분야의 당업자에 의해서 판별되는 특정 값에 대한 편차의 허용 가능한 범위 이내를 의미한다. 본 명세서에서 사용된 "대략 (about)" 또는 "대략 (approximately)"은 명시된 값을 포함하며, 불확실한 측정 및 특정 양의 측정과 관련된 오차(예를 들면, 측정 시스템의 한계)를 고려하여, 당해 분야의 당업자에 의해 결정된 특정 값에 대한 허용 편차 이내를 의미한다. 예를 들어, "약"은 하나 이상의 표준 편차들 내에서 또는 명시된 값의 ±30%, 20%, 10%, 5% 이내를 의미할 수 있다. 또한 "~일 수 있다"의 사용은 본 발명의 설명하는 실시예들이 "본 발명의 하나 이상의 실시예들"을 나타낸다. 본 명세서에서 사용된 바와 같이, 용어 "사용하다", "사용하는" 및 "사용되는"은 "활용하다", "활용하는" 및 "활용되는"과 동의어로 간주될 수 있다. 또한 "예시적인"의 용어는 사례 또는 예시를 나타낸다.

[0039] 특정 실시예가 다르게 구현될 수 있을 때, 특정 프로세스 순서는 설명된 순서와 다르게 수행될 수 있다. 예를 들어, 설명된 2개의 연속하는 프로세스들은 실질적으로 동시에 또는 설명된 순서와 반대 순서로 수행될 수 있다.

[0040] 다양한 실시예들이 실시에 및/또는 중간 구조들의 개략도인 단면도를 참조하여 여기에 설명된다. 결과로서의 도면들의 형상들로부터의 변화는 예를 들어, 제조 기술들 및/또는 허용 오차와 같이 예상되어야 한다. 따라서, 본 명세서에 개시된 실시예들은 도시된 특정 형태들의 형상에 한정되는 것으로 해석되어서는 안되며, 예를 들어 제조 상의 형상 편차들을 포함해야 한다. 예를 들어, 직사각형으로 도시된 주입 영역은 전형적으로, 주입된 영역에서부터 주입되지 않은 영역으로의 이진 변화 라기보다는, 둥근 또는 곡선의 특성들 및/또는 모서리에서의 주입 농도의 기울기를 가질 것이다. 마찬가지로, 주입에 의해 형성된 매립 영역은 매립 영역과 주입이 일어나는 표면 사이의 영역에 약간의 주입을 초래할 수 있다. 따라서, 도면들에 도시된 영역들은 본질적으로 개략적이며, 그 형상들은 장치의 영역의 실제 형상을 예시하는 것이 아니며, 제한하려는 것은 아니다. 또한, 당업자가 알 수 있는 바와 같이, 설명된 실시 예들은 본 개시의 사상 또는 범위를 벗어나지 않고 다양한 다른 방식으로 수정될 수 있다.

[0041] 본 명세서에 설명된 본 발명의 실시예들에 따른 전자 또는 전기 장치들 및/또는 임의의 다른 관련 장치들 또는 구성 요소들은 임의의 적합한 하드웨어, 펌웨어(예를 들어, 주문형 집적 회로), 소프트웨어 또는 소프트웨어, 펌웨어 및 하드웨어의 조합을 이용하여 구현될 수 있다. 예를 들어, 이들 장치들의 다양한 구성 요소들은 하나의 집적 회로(IC) 칩 상에 또는 개별 IC 칩들 상에 형성될 수 있다. 또한, 이들 장치들의 다양한 구성 요소들은 플렉시블 인쇄 회로 필름, 테이프 캐리어 패키지(TCP), 인쇄 회로 기판(PCB) 또는 하나의 기판 상에 구현될 수 있다. 또한, 이들 장치의 다양한 구성 요소들은 하나 이상의 컴퓨팅 장치들 내 하나 이상의 프로세서에서 실행

되고, 컴퓨터 프로그램 명령들을 실행하고, 여기에 설명된 다양한 기능들을 수행하기 위해 다른 시스템 구성 요소들과 상호 작용하는 프로세스 또는 스레드(thread)일 수 있다. 컴퓨터 프로그램 명령들은, 예를 들어 RAM(random access memory)과 같은 표준 메모리 장치를 사용하는 컴퓨팅 장치에서 구현될 수 있는 메모리에 저장된다. 컴퓨터 프로그램 명령은 또한 예를 들어 CD-ROM, 플래시 드라이브 등과 같은 다른 비일시적인 컴퓨터 판독 가능 매체에 저장될 수 있다. 또한, 당업자는 본 발명의 예시적인 실시 예들의 사상 및 범위를 벗어나지 않는 한 다양한 컴퓨팅 장치들의 기능이 단일 컴퓨팅 장치에 결합되거나 통합될 수 있으며, 또는 특정 컴퓨팅 장치의 기능이 하나 이상의 다른 컴퓨팅 장치들에 걸쳐 분산될 수 있음을 인식해야 한다.

[0042] 달리 정의되지 않는 한, 본 명세서에서 사용되는 모든 용어(기술 용어 및 과학 용어 포함)는 본 발명이 속하는 기술 분야의 당업자에 의해 일반적으로 이해되는 것과 동일한 의미를 갖는다. 또한, 일반적으로 사용되는 사전에서 정의된 용어들과 같은 용어들은 관련 기술 및/또는 본 명세서와 관련하여 그 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 이상적이거나 지나치게 형식적인 의미로 해석되지 않아야 한다.

[0043] 상술한 바와 같이, 호스트의 도움 없이 하드 스토리지 장치에 의해서 수행되는 종래의 비 지능형 데이터 압축은 항상 비손실 압축의 형태였다. 더욱이, 종래의 모든 손실 데이터 압축은 드라이브에 의해서가 아닌 호스트/호스트 컴퓨터/호스트 CPU에 의해서 수행되었다. 그러나, (예를 들어, 음악 데이터 파일, 비디오 데이터 파일 또는 다른 멀티-미디어 데이터 파일을 편집 및/또는 저장할 때) 손실 데이터 압축이 수용 가능하거나 심지어 바람직 할 때, 다른 타입들의 데이터가 높은 수준의 압축에 적용될 수 있는 시나리오가 있을 수 있다. 또한 호스트 CPU가 압축을 수행하는데 필요한 리소스를 비우는 동안 드라이브에서 원시 데이터 파일을 압축하는 것이 유용할 수 있다.

[0044] 본 개시의 실시예들은 데이터 압축의 실행에서 대응하는 호스트의 도움 없이 손실 데이터 압축 및 저장을 수행하여 다른 사용자 활동들에 대한 호스트의 CPU를 자유롭게 할 수 있는 SSD(solid state drive)와 같은 스토리지 장치를 제공한다.

[0045] 도 1은 종래의 손실 데이터 압축 방법의 예시 및 본 개시의 실시예에 따른 손실 데이터 압축 방법의 예시의 비교를 도시한다.

[0046] 도 1을 참조하면, 앞서 설명된 바와 같이, 종래의 드라이브들(116)은 호스트(112)의 도움 없이 손실 데이터 압축을 수행할 수 없다. 대신에, 도 1에 도시된 바와 같이, 손실 데이터 압축 방식(110)은 호스트(112)가 데이터를 압축(113)하고, 압축된 데이터 파일(114)을 저장하기 위해 드라이브(116)로 전송하기 전에, 원시 데이터 파일(118)을 호스트(112)로 제공하는 것에 의해 수행되므로 호스트(112)의 시스템 자원들이 고갈된다. 즉, 종래에는, 만일 원시 데이터 파일(118)(예를 들면, PCM 포맷의 원시 사운드/오디오 데이터 파일)이 압축되도록 진행될 때, 사용자는 호스트(112)가 원시 데이터 파일(118)을 압축된 데이터 파일(114)(예를 들면, MP3 데이터 파일)로 압축(113)하도록 지시하고, 호스트(112)는 데이터 압축(113)을 수행해서 압축된 데이터 파일(114)을 생성한다. 그 후, 호스트(112)는 압축된 데이터(114)를 저장하기 위해 드라이브(116)로 전송할 것이다.

[0047] 따라서, 비록 이전 기술들이 드라이브(116)가 데이터 충실도를 유지할 원시 데이터(118)에 대해 무손실 데이터 압축(113)을 수행할 수 있게 하지만, 드라이브(116)는 손실 데이터 압축을 수행할 수 없다. 데이터 충실도의 손실이 사용자에 의해 용인될 수 있고, 허용되거나 요구될 때(예를 들어, 음악 및 비디오 편집의 경우와 같이, 사용자가 다양한 손실 데이터 압축을 선호하는 경우), 손실 데이터 압축은 통상적으로 대응하는 호스트 장치들(예를 들면, 호스트(112)) 상의 많은 프로세서 사이클들을 점유한다.

[0048] 상술한 관점에서, 만일 손실 데이터 압축의 작업이 사용자의 기본 CPU(들) 내에 위치한 호스트(들)로부터 오프로드(offload)될 수 있다면 유용할 수 있을 것이다. 따라서, 본 개시의 실시예들에 따른 드라이브(126)(예를 들면, 하드 스토리지 드라이브 또는 SSD(solid state drive))는 호스트/호스트 컴퓨터/호스트 CPU(122)로부터 비 압축된 데이터/원시 데이터 파일(128)을 수신하고, 원시 데이터 파일(128)을 압축(123)한 후 압축된 데이터 파일(124)을 생성하는 하나 이상의 손실 데이터 압축 방식들(120)을 수행한다. 압축된 데이터 파일(124)은 드라이브(126) 상에 보관/저장된다.

[0049] 본 실시예에 따르면, 사용자가 (예를 들면, CPU가 "압축 요청" 플래그를 설정하도록 명령하는 것에 의해서) 호스트에게 원시 데이터(128)를 압축(123)하도록 명령할 수 있으며(예를 들면, 사용자가 드라이브(126) 상에 압축된 형태로 저장될(125) 비디오 또는 오디오 데이터 파일의 편집을 완료했을 때), 이에 의해 드라이브(126)가 손실 데이터 압축을 사용하여 원시 데이터 파일(128)을 자동적으로 압축(123)해서 호스트(122)의 자원들의 사용없이 드라이브(126) 상에 압축된 데이터 파일(124)을 생성하도록 한다. 지능형 손실 데이터 압축 드라이브(126)는

데이터 분석에 의해서(예를 들면, 데이터 파일의 타입, 가능한 데이터 압축 방법들 및 알고리즘들, 사용자 파라미터들 및 서비스 품질(quality of service, QoS) 요구들 등의 분석에 의해서) 자동 데이터 압축을 허용하고, 손실 데이터 압축(123)이 적절한 경우를 확인한다. 그 다음, 드라이브(126)는 선택된 데이터 압축 방식에 따라서 손실 데이터 압축(123)을 수행할 수 있고, 압축된 데이터 파일(124)을 자신에 저장할 수 있다. 드라이브(126)는 데이터를 수신하는 대로 데이터(128)를 압축(123)하거나 또는 데이터를 압축(123)하고 압축된 데이터(124)를 저장하기 전에 일정 시간동안 비압축된 포맷으로 데이터를 저장할 수 있다.

[0050] 도 2는 본 개시의 실시예에 따라서 호스트의 도움없이 드라이브에서 지능형 손실 데이터 압축을 수행할 수 있는 손실 압축 드라이브를 도시한다.

[0051] 도 2를 참조하면, 상술한 바와 같이, 사용자는 호스트(122)로부터의 원시 데이터 파일(128)(예를 들면, PCM 포맷의 원시 사운드 데이터)을 드라이브(126)로 직접적으로 전송할 수 있다. 그 후, 드라이브(126)는 원시 데이터 파일(128)이 손실 압축 데이터 파일(124)로 적절하게 변환할 수 있는 원시 데이터 파일임을 식별할 수 있다. 그 다음, 드라이브(126)는 손실 데이터 압축(123)을 수행해서 드라이브(126) 내 압축된 데이터 파일(124)(예를 들면, MP3 데이터 파일)을 생성할 수 있으므로, 원시 데이터 파일(128)은 압축된 손실 데이터 파일(124)로 변환되고, 압축된 손실 데이터 파일(124)은 드라이브(126)에 보관/저장된다. 따라서, 본 실시예는 자동적으로 또는 반-자동적으로 잘 제어되는 손실 데이터 압축(123)을 수행할 수 있는 저장 장치(예를 들어, 드라이브)이며, 따라서 다른 활동들에 대해 그렇지 않으면 데이터 압축을 수행하기 위해 호스트(122)에 의해 사용되었을 CPU 클럭 사이클들 및 자원들을 자유롭게 할 수 있다.

[0052] "스마트 압축(smart compress)" 드라이브(126)는 데이터 관리를 위한 명령 및 제어 프로토콜, 그리고 상대적으로 향상된 온-디스크(예를 들면, on-SSD) 자원들을 포함할 수 있다. 예를 들어, 종래의 하드 디스크와 달리, 드라이브(126)는 다양한 손실 데이터 압축 메커니즘들을 포함하는 데이터 압축 알고리즘 라이브러리(17)를 포함할 수 있다. 온-보드 인코딩 라이브러리/데이터 압축 알고리즘 라이브러리(17)는 다양한 손실 데이터 압축 방식들 및 인코딩 방식들 수행하기 위한 다양한 데이터 압축 알고리즘들에 대응하는 코드들을 포함할 수 있다. 또한 드라이브(126)는 손실 데이터 압축 파라미터들을 포함하도록 확장된 서비스 품질(QoS) 우선 데이터 파일(14)을 포함할 수 있다. 이러한 파라미터들은 원시 데이터(128)가 어떻게 압축되는지를 판별하기 위해 드라이브(126)에 의해서 사용될 수 있다.

[0053] 사용자로부터의 각 손실 데이터 압축 요청은 궁극적으로 압축될(123) 원시 데이터 파일(128)과 관련된다. 데이터 압축 요청은 압축(123)의 결과가 되는 파일의 포맷을 분명하게 나타내는 또는 압축(123)을 수행하는 동안 드라이브(126)를 안내하는 파라미터들을 분명하게 나타내는 직접 명령을 포함할 수 있다. 대안적으로, 데이터 압축 요청은 키-메타데이터에 포함된 내장 명령에 대응할 수 있으며, 드라이브(126)는 키-메타데이터에 근거해서 압축(123)을 어떻게 수행할지를 결정할 수 있다.

[0054] 드라이브(126)가 손실 방식으로 무언가를 압축(123)하기 위해서, 드라이브(126)는 요청된 데이터 압축 알고리즘 또는 최종 포맷과 같은 정보를 사용하여 어떤 압축 방식을 사용할지(예를 들면, 데이터 압축 알고리즘 라이브러리(17)로부터 어느 데이터 압축 알고리즘을 선택할지) 결정할 수 있다. 즉, 드라이브(126)는, 드라이브(126)가 특정 손실 데이터 압축 알고리즘을 사용하여 원시 데이터 파일(128)을 압축하거나 궁극적으로 특정 손실 저장 포맷(예를 들면, MP3 데이터 파일, JPEG 데이터 파일, MPEG 데이터 파일 등)에 따라서 압축된 데이터 파일(124)를 저장해야 한다는 것을 이해할 수 있도록 다양한 정보들을 사용할 수 있다.

[0055] 대안적으로, 드라이브(126)는 (예를 들면, QoS 우선 데이터 파일(14)에 저장된) 파라미터들 및 세팅들을 사용하여 언제 어떻게 데이터를 손실 압축할지를 결정할 수 있다. 예를 들어, QoS 파일은 특정 확장자의 모든 파일들이 JPEG 데이터 파일로 압축되어야 함을 나타낼 수 있다. QoS 파일은 예를 들면, 사용자가 특정 데이터 압축 방식 또는 품질 설정을 선택하거나 조절하거나, 서브-샘플링 양을 설정하거나, 또는 다양한 다른 파라미터들을 제어하는 것을 허용하는 압축 알고리즘들에 대한 압축 세부 사항들을 더 포함할 수 있다.

[0056] 따라서, 드라이브(126)는 정보를 사용하여 수신된 주어진 원시 데이터 파일에 대해 요구되는 데이터 압축 포맷 및 관련된 파라미터들을 이해할 수 있다. 본 개시의 실시예들에 따르면, 데이터 압축 포맷 및 파라미터들은 적어도 3개의 다른 옵션들 중 하나 이상의 옵션으로 전달 및 판별되어 이하 설명되는 바와 같이 드라이브(126)가 정보들을 학습하는 것을 가능하게 한다.

[0057] 드라이브(126)의 향상된 특징들의 다른 예시들에 따르면, 드라이브(126)는 인코딩 가속 하드웨어(encoding acceleration hardware)를 포함할 수 있으며, 인라인 또는 오프라인으로 데이터 중복제거를 수행할 수 있으며,

또한 온라인 또는 오프라인 중 하나로 무손실 데이터 압축을 할 수 있다. 손실 데이터 압축(123)이 온라인 동작으로 수행될 때, 원시 데이터 파일(128)이 드라이브(126)에 의해 수신되는 그 때 또는 그 근방에서 데이터는 압축된다. 손실 데이터 압축(123)이 오프라인 모드로 수행될 때, 비압축된 데이터(128)는 드라이브(126) 상에서 무기한의 시간동안 저장될 수 있다. 드라이브(126)가 충분한 시간/자원들을 가지고 있는 것으로 판별할 때 또는 드라이브(126) 상에 저장된 비압축된 데이터(128)를 작업하는 것을 결정할 때, 데이터 파일(128)은 압축되고 (123) 그리고 압축된 데이터 파일(124)로서 드라이브(126)에 재저장(125)될 수 있다. 드라이브(126)가 온라인 또는 오프라인으로 동작하는 것을 결정할 지는 압축을 위해 필요한 사용 시간 그리고/또는 시스템 자원들의 추정된 양에 대응하는 드라이브(126)의 예측에 의존할 수 있다.

[0058] 그러한 예측들은 머신 러닝 예측들에 근거할 수 있으며, 드라이브(126)의 온보드 장치 성능 프로파일(15)의 분석에 근거해서 결정될 수 있다. 예를 들어, 장치 성능 프로파일(15)은 드라이브(126)의 진행중인 물리적 성능의 기록(예를 들면, 사용 메모리, 프로세서 능력 등)에 대응하는 정보를 포함할 수 있고, 압축 알고리즘 성능 차트(16)는 주어진 작업에 의해 요구될 수 있는 사용 자원들의 총량을 예측하기 위해 사용될 수 있다. 장치 성능 프로파일(15) 및 압축 알고리즘 성능 차트(16)에 포함된 정보에 근거해서, 드라이브(126)는 압축될 원시 데이터 파일(128)의 타입의 결정에 근거해서 데이터 압축 알고리즘 라이브러리(17)로부터의 어느 압축 알고리즘이 사용 장치 자원들을 가장 효율적으로 이용할 수 있는지를 판별할 수 있다.

[0059] 압축 알고리즘 성능 차트(16)는 복수의 압축 알고리즘들에 대한 정보(예를 들면, 각 압축 알고리즘에 대한 평균 압축 비율, 각 압축 알고리즘에 대한 처리량, 각 압축 알고리즘에 대한 메모리 자원 요구들 등)를 포함할 수 있고, 파일 타입, (QoS 우선 데이터 파일(14)에 의해 결정될 수 있는) 가능한 처리량 및 장치 성능 프로파일(15)에 근거한 장치 성능들에 근거해서 데이터 압축 알고리즘 라이브러리(17)로부터 특정 데이터 압축 알고리즘의 선택을 허용할 수 있다.

[0060] 드라이브(126)의 다른 향상된 특징들은 원시 미디어 데이터 파일들의 상이한 타입들을 알 수 있게 하는(예를 들면, 원시 데이터 파일들의 확장자들 및 서명들을 인식하는 것에 의해) 엔트로피 추정 알고리즘 라이브러리(6), 객체 서명 사전(5) 그리고/또는 객체 확장자 사전(4)을 포함할 수 있다.

[0061] 엔트로피 추정 알고리즘 라이브러리(6)는 임의의 샘플 데이터의 엔트로피를 테스트하기 위한 코드들을 포함하고 있어서 데이터 파일(128)의 압축률 레벨을 판별할 수 있다. 엔트로피-기반 계산들은 파일이 압축 가능한 정도를 예측하기 위해 값(2)의 일부를 검색하도록 사용될 수 있다. 엔트로피-기반 계산들은 엔트로피 또는 랜덤 데이터가 압축하기 더 어렵다는 원리로 기능한다(예를 들면, 이미 압축된 파일들은 높은 엔트로피 특성들을 가지며, 압축되지 않은 파일은 더 압축 가능한 순서 데이터를 가질 가능성이 더 많다). 객체 서명 사전(5)은 각각의 압축 기술들에 대응하는 알려진 객체/파일 서명들을 포함할 수 있다. 객체 확장자 사전(4)은 다양한 압축 기술들 각각에 대해 이들 객체들의 타입들이 얼마나 압축 가능한지 평가된 알려진 객체/파일 확장자의 리스트를 포함할 수 있다. 이러한 정보는 압축을 용이하게 하기 위해 객체 확장자 사전(4)에 의해서 데이터 압축 예측기(3)로 제공된다.

[0062] 따라서, 원시 데이터 파일(128)이 수신될 때, 드라이브(126)의 다양한 구성 요소들은 압축될 원시 데이터 파일(128)의 타입을 식별하기 위해 사용될 수 있고, 이는 데이터 압축 예측기(3) 및 압축기(7)의 동작에 영향을 미치고, 데이터 압축 알고리즘 라이브러리(17)로부터 적절한 데이터 압축 알고리즘을 선택하는데 사용될 수 있다. 예를 들어, 데이터 압축 예측기(3)는 객체 확장자 사전(4), 객체 서명 사전(5) 그리고/또는 엔트로피 추정 알고리즘 라이브러리(6)에 포함된 정보를 사용하여 값(2)의 압축률을 결정하거나 또는 결정하는데 도움을 줄 수 있다.

[0063] 그 후, 압축기(7)는 지시된 파라미터들에 근거해서 데이터 압축을 수행할 수 있다. 데이터 압축 예측기(3) 및 압축기(7)는 데이터 파일을 식별하고, 그것이 어떤 타입의 데이터 파일인지 인식하며 그리고 즉각적인 사용자 요구들, 드라이브(126)의 즉각적인 성능들, QoS 우선 데이터 파일(14)에 따른 QoS 요구들(이는 어떤 데이터 압축 방식이 데이터 압축에 사용되어야 하는지를 결정하기 위해 사용될 수 있음), 압축 알고리즘 성능 차트(16) 내 저장된 히스토리컬 데이터 압축 알고리즘 성능 등과 같은 팩터들에 근거해서 가장 효율적인 손실 데이터 압축 알고리즘을 지능적으로 제공할 수 있다. 또한 데이터 압축 예측기(3) 및 압축기(7)는 이용 가능한 더 양호한 데이터 압축 알고리즘의 존재(예를 들면, 만일 더 많은 프로세서 공간이 이용 가능해지거나, 더 많은 버퍼 공간이 이용 가능해지는 등)를 나중에 판별할 수 있다.

[0064] 원시 데이터 파일(128)을 어떻게 압축할 지에 관해서 데이터 압축 예측기(3)에 의한 결정이 이루어지면, 수신 스위치(11)로 정보가 제공되고, 수신 스위치(11)는 데이터(예를 들어, 값(2))를 압축기(70)로 출력하는 것과

(손실 압축이 필요한 것으로 확인되지 않은 파일 타입이 수신되는 때와 같이, 압축하지 않는 것으로 결정되면) 출력(13)을 위해 데이터를 출구 스위치(12)로 전송하는 것에 의해서 압축기(7)를 바이패스 하는 것 사이에 스위치한다. 그러므로, 데이터 압축 예측기(3)는 어떻게 압축이 발생되어야 하는지를 결정하고, 수신 스위치(11)는 데이터를 압축기(7) 또는 출력(13)을 위한 출구 스위치(12) 중 하나로 전송할 것이다.

[0065] 아래는 드라이브(126)에 의해 수신된 원시 데이터 파일(128)을 압축하기 위한 적절한 압축 방식들 및 파라미터들을 결정하기 위한 3 가지 옵션들을 나타낸다.

[0066] 제1 옵션으로서, 데이터 압축 포맷 및 파라미터들은 강화된 저장 명령의 명시적인 부분으로서 결정될 수 있다.

[0067] 종래의 시스템에서, 드라이브(126)에 저장되는 각 원시 데이터 파일(128)은 드라이브(126)로의 궁극적인 저장 명령을 포함할 수 있고, 그에 따라 드라이브(126)에 대한 명시적인 지시를 표시하여 특정 데이터 압축 타입 및 특정 데이터 압축 파라미터들과 함께 특정 원시 데이터 파일(128)을 저장한다.

[0068] 대조적으로, 본 옵션에서, 사용자가 호스트에게 드라이브(126)가 원시 데이터 파일(128)을 압축된 데이터 파일(124)로서 저장하도록 명령을 하라고 지시할 때, 데이터 압축 포맷 및 파라미터들은 저장 명령의 일부로서 명료하게 판별될 수 있다. 본 명세서의 3가지 옵션들 중, 이 옵션은 일반적으로 (블록 기반 스토리지 장치는 아래에서 설명하는 제2 옵션에 따라 키 값 메타 데이터 옵션을 갖지 않으므로) 블록-기반 스토리지 장치들과 함께 사용될 수 있다. 그러나, 이 제1 옵션의 구현은 전통적인 블록-기반 데이터 파일 시스템의 드라이버/OS 계층의 변경을 요구할 수 있다.

[0069] 드라이버/OS 계층의 변경은 원시 데이터 파일(128)을 손실 압축(123)하는 의도 또는 기회의 존재를 판별하기 위해 드라이브(126)로의 쓰기 명령을 초래하는 모든 사용자 활동들(예를 들어, 사용자가 의도적으로 비트맵(bitmap)을 JPEG로 저장하거나, 비트맵을 압축되지 않은 드라이브에서 드라이브(126)로 이동하는 것)을 스캐닝하는 것을 포함할 수 있다. 드라이버/OS 계층의 다른 변경은 사용자의 의도를 확인하기 위해 그리고 사용자가 압축(123)의 파라미터들에 관한 정보를 제공하도록 유도하기 위해 데이터 압축 요청을 인터럽트하는 것을 포함할 수 있다(예를 들어, 사용자는 호스트(122)에서 사용자가 손실 데이터 압축(123)에 관한 파라미터들을 선택할 수 있는 메뉴와 상호작용한다). 더욱이, 부가적인 수정 명령들은 아이덴티티(identity) 및 파라미터 정보를 전송하기 위해 그리고 드라이브(126)가 어느 논리적 블록 어드레스들(al block addresses, LBAs)이 목표 데이터 파일과 연관되는지 식별할 수 있도록 하기 위해 사용될 수 있다.

[0070] 드라이버/OS 계층에 대한 하나 이상의 제안된 변경과 관련된 정보를 획득한 다음, 원시 데이터 파일(128)은 드라이브(126)로 전송될 수 있고, 오퍼레이팅 시스템 및 드라이버는 호스트(122)로부터 드라이브(126)로 사용자에 의해 선택된 정보를 전달할 수 있다. 이 옵션은 드라이브(126)와의 통신의 특정의 향상된 프로토콜들을 더 구현할 수 있다.

[0071] 제2 옵션으로서, 데이터 압축 포맷 및 관련된 파라미터들은 키(1) 및 값(2)((KV) 쌍)의 키(1)에 명시적으로 인코딩될 수 있다. 드라이브(126)는 개선된 명령어 또는 명령어 세트에 응답하거나, 원시 데이터(128)와 관련된 놓축된 키(1)를 검출할 수 있으며, 손실-압축 가능 데이터의 쓰기는 "압축 요청" 플래그, 데이터 압축 포맷 식별 및/또는 데이터 압축 포맷 파라미터들에 의해 수반된다.

[0072] 본 옵션에 따른 KV-기반 시스템에서, 데이터 압축 포맷 및 파라미터들의 관련 정보는 데이터가 어떻게 저장되어야 하는지를 나타내는 키(1)의 메타데이터 부분에 저장될 수 있다. 예를 들어, 키(1)는 압축된 데이터 파일(124)이 특정 데이터 압축 설정 및 특정 서브샘플링 양을 갖는 JPEG 데이터 파일로서 저장되어야 하는 것을 나타내는 메타데이터를 포함할 수 있다. 예를 들어, 드라이브(126)의 데이터 압축 예측기(3) 및 압축기(7)는, 데이터 파일(128)이 수정/압축되어야 하는지, 데이터 압축이 어떻게 수행되어야 하는지를 판별하기 위해 키(1)에 포함된 정보(예를 들면, 메타데이터)를 사용할 수 있다.

[0073] 더욱이, 키(1)는 데이터 파일(128)의 타입을 나타내는 파일/객체 확장자를 포함할 수 있으며, 이는 압축이 수행될지/어떻게 수행되는지에 관한 정보를 제공할 수 있다. 알려진 객체 확장자들의 리스트는 객체 확장자 사전(4)에 유지될 수 있고, 객체 확장자 사전(4)으로부터의 정보는 데이터 압축(123)의 타입을 판별하기 위해 데이터 압축 예측기(3)에 의해서 사용될 수 있다.

[0074] 또한 키(1)는 파일/객체 서명을 포함할 수 있고, 이는 KV 쌍의 값(2) 내 저장된 데이터 파일(128)의 타입에 관한 정보를 제공하는 코드 라인이다. 데이터 타입들에 관한 알려진 서명들의 리스트는 객체 서명 사전(5)에 유지될 수 있으며, 객체 서명 사전(5)으로부터의 정보는 사용할 압축(123)의 타입을 결정하기 위해 데이터 압축 예

측기(3)로 제공될 수 있다.

[0075] 따라서, 수신된 데이터가 원시 미디어 데이터 파일임을 나타내도록 키(1)는 드라이브(126)에 의해 수신될 수 있고, 키(1)는 데이터 압축이 어떻게 수행되어야 하는지의 파라미터들을 포함할 수 있다. 따라서, 객체 서명 사전(5) 또는 객체 확장자 사전(4)은, 키(1)의 정보/메타데이터를 읽어오는 것에 의해, 값(2)에 대응하는 수신된 원시 데이터 파일(128)이 압축되어야 함을 검출할 수 있고, 데이터 압축 알고리즘 라이브러리(17)로부터 어느 데이터 압축 알고리즘이 사용되어야 하는지 판별하기 위해 메타데이터를 사용할 수 있다.

[0076] 데이터를 어떻게 압축(132)하는지를 결정하기 위해 드라이브(126)는 키(1)의 메타데이터 부분을 읽을 수 있다. 데이터 파일(28)이 KV 객체로서 KV 스토어로 전송될 때, KV 스토어는 키(1)를 읽을 수 있고, 메타데이터 부분을 인식하고 읽을 수 있으며, 드라이브(126)가 표시된 데이터 압축 포맷 및 파라미터들을 사용함으로써 손실 데이터 압축(123)의 선택된 타입으로 압축된 데이터 파일(124)을 저장하도록 명령 받았음을 판별할 수 있다. 메타데이터 부분에 저장된 정보의 사용에 의해서, 드라이브(126)는 지정된 데이터 압축 알고리즘 및 파라미터들로 연관된 원시 데이터 파일(128)을 압축(123)할 수 있다.

[0077] 따라서, 데이터 압축 포맷 및 파라미터들은 KV 스토리지 시스템의 명확한 부분이 될 수 있다. KV 쌍의 키(1)는 숨겨진 정보를 포함하는 메타데이터 부분을 포함할 수 있고, 파라미터들은 드라이브(126)에 의해 압축 가능한 데이터의 식별이 가능하도록 인코딩될 수 있다. 그러한 메타데이터 정보의 사용자 레벨 엔트리는 KV 스토리지 소프트웨어 스택으로 탈성될 수 있다. 즉, 사용자는 어느 데이터 파일들(128)이 손실 압축(123)되는지 KV 쌍의 키(1) 내 저장된 태그에 대응하는 정보를 태그 할 수 있다.

[0078] 본 옵션은 "세밀한(fine-grained)" 제어를 사용자에게 제공하고, 구현하기 쉽고, 사소한 변경으로 구현될 수 있다. 이 옵션에 따르면, 사용자는 소정 타입의 원시 데이터 파일이 드라이브(126)에 저장될 때, 데이터가 압축될 것임을 이해할 수 있다.

[0079] 제3 옵션으로서, 데이터 압축 포맷 및 파라미터들이 보다 큰 서비스 품질(QoS) 우선 데이터 파일(14)의 일부로서 암시적으로 행해질 수 있다. QoS 우선 데이터 파일(14)은 (예를 들어, 사용자에 의해) 발생될 수 있으며 정확한 또는 일반적인 손실 압축 및 저장 옵션들을 명시하기 위해 드라이브(126)와 함께 설정된다. QoS 우선 데이터 파일(14)은 (예를 들어, 디풀트 데이터 압축 타입 및 주어진 타입의 모든 데이터 파일들(128)에 적용될 파라미터들을 나타내고, 보존될 원시 데이터 파일(128)의 백업 복사본을 요구하고, 주어진 상황 하에서의 손실 데이터 압축(123)을 금지 또는 지연시키기 위해 등) 데이터 압축 정책들의 특정 및/또는 글로벌 구현을 위한 설정들을 포함할 수 있다.

[0080] QoS 우선 데이터 파일(14)은 드라이브(126)가 객체 원시 데이터 파일들(128)을 식별하기 위해 사용하는 정보(예를 들면, 키(1) 내 정보, 데이터 파일 확장자들, 헤더 서명들 등)를 포함할 수 있다. QoS 우선 데이터 파일(14)에 맞는 미디어 데이터 파일의 드라이브(126)에 의한 인식하는 즉시, QoS 우선 데이터 파일(14)에 포함된 옵션들은 지능형 드라이브(126)에 의해 자동으로 구현될 수 있다. 드라이브(126)는 QoS 선호 데이터 파일(14) 내 정보를 사용해서 적절한 데이터 압축 알고리즘 및 파라미터들을 선택하고 자동적으로 적용한다. 일부 실시예들에서, 글로벌 QoS 요구는 충돌하는 사용자 명령들 또는 키-필드 지정을 무시 override(하거나 지연(delay)시킬 수 있다.

[0081] 따라서, 사용자가 호스트(122)가 특정 타입의 원시 데이터 파일(128)을 드라이브(126)로 전송하게 할 때, 드라이브(126)는 QoS 우선 데이터 파일(14)에 따른 대응하는 방식으로 데이터를 압축하고(123) 저장(125)할 수 있도록 구성될 수 있다. 이 옵션은 이론적으로 수천 개의 데이터 파일들(128)에 적용될 수 있는 일반 규칙들의 초기 구성 후에 사용자에게 보다 적은 제어를 제공하여 사용자로부터의 세세한 제어의 양을 제거한다. 그러나, 이 옵션은 글로벌 데이터 압축 제어를 허용해서, QoS 우선 데이터 파일(14)이 데이터 압축 메커니즘들의 글로벌 구성 을 허용할 수 있고, 침입하는 사용자 상호 작용을 감소시킬 수 있다. 본 실시예에 따른 드라이브(126)는 더 빠른 컨트롤러 프로세서, 더 큰 쓰기 버퍼들, 더 큰 컨트롤러 프로세서 RAM 할당 및/또는 데이터 압축/인코딩 가속기 중 하나 이상과 같은 부가적인 하드웨어 자원들을 포함할 수 있음을 알아야 한다.

[0082] 따라서, 상술한 실시예들은 지능형 손실 데이터 압축이 가능한 자족적(self-sufficient) 스토리지 장치를 제공한다.

[0083] 상술한 내용은 예시적인 실시예를 설명하기 위한 것이며, 본 발명을 제한하는 것으로 해석되어서는 안된다. 비록 몇몇 예시적인 실시예들이 설명되었지만, 당업자는 예시적인 실시예들의 신규한 교시 및 이점으로부터 실질적으로 벗어나지 않고도 예시적인 실시예에서 많은 수정이 가능하다는 것을 용이하게 이해할 것이다. 따라서,

그러한 모든 수정들은 청구항에 정의된 예시적인 실시예들의 범위 내에 포함되도록 의도된다. 청구 범위에서, 수단과 기능이 결합된 표현은 구조적 등가물뿐만 아니라 등가의 구조를 열거하여 여기에 설명된 구조를 포함하고자 한다. 따라서, 상기 설명은 예시적인 실시예들을 설명하기 위한 것이며, 개시된 특정 실시예들에 한정되는 것으로 해석되어서는 안되며, 개시된 예시적인 실시예들뿐만 아니라 다른 예시적인 실시예들에 대한 수정이 포함되는 것으로 이해되어야 한다. 본 발명의 개념은 다음의 청구 범위에 의해 정의되고, 청구 범위의 균등률도 포함된다.

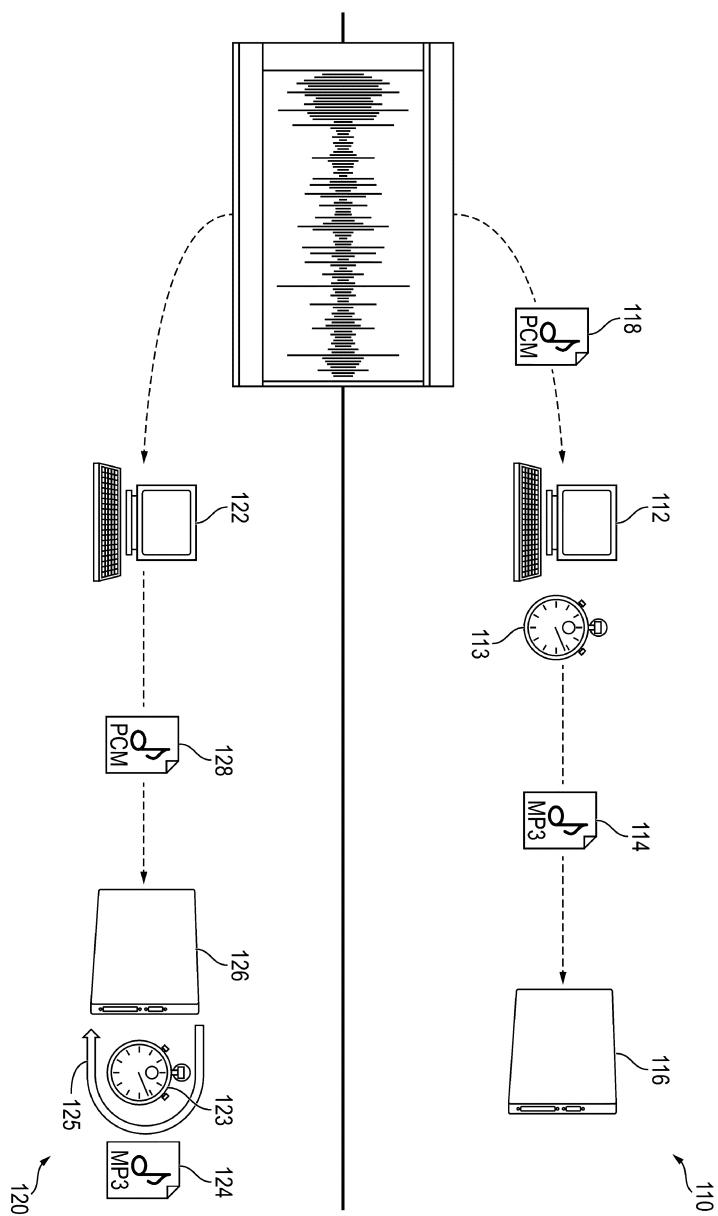
부호의 설명

[0084]

- 110: 종래 손실 데이터 압축 방식
- 120: 본 발명 손실 데이터 압축 방식
- 112, 122: 호스트
- 113, 123: 데이터 압축
- 114, 124: 압축된 데이터 파일
- 116, 126: 드라이브
- 118, 128: 원시 데이터 파일
- 125: 저장

도면

도면1



도면2

