(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년04월21일
(11) 등록번호 10-2102728
(24) 등록일자 2020년04월14일

(51) 국제특허분류(Int. Cl.)
G06F 11/08 (2006.01) *G06F 12/00* (2016.01)
G06F 13/00 (2006.01)
(21) 출원번호 10-2013-0100271
(22) 출원일자 2013년08월23일
심사청구일자 2018년08월23일
(65) 공개번호 10-2014-0031112
(43) 공개일자 2014년03월12일
(30) 우선권주장
13/688,654 2012년11월29일 미국(US)
61/696,720 2012년09월04일 미국(US)
(56) 선행기술조사문헌
JP07271522 A*
(뒷면에 계속)
전체 청구항 수 : 총 20 항

(73) 특허권자
엘에스아이 코포레이션
미국 캘리포니아 95131, 새너제이, 라이더 파크 드라이브 1320
(72) 발명자
코헨 얼 티
미국 캘리포니아주 95035 밀피타스 스위트 100 사우스 밀피타스 불러바드 691
퀸 로버트 에프
미국 캘리포니아주 95035 밀피타스 바버 레인 1621
(74) 대리인
특허법인 남앤남

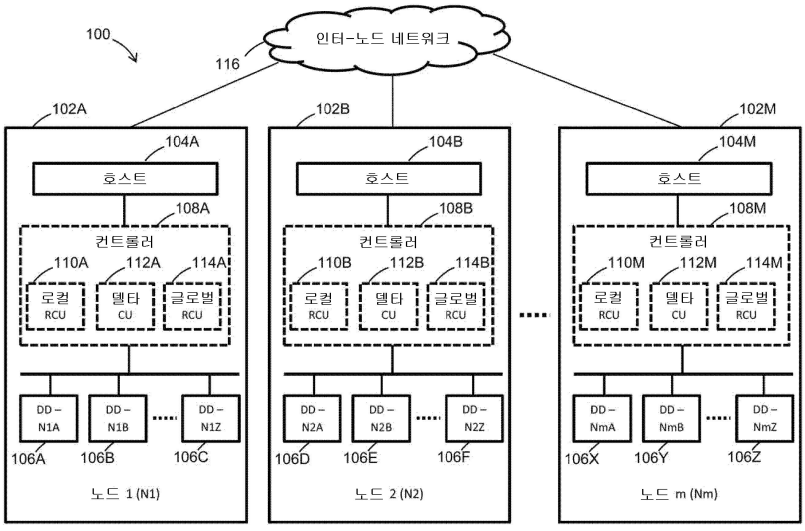
심사관 : 김계준

(54) 발명의 명칭 **스케일러블 스토리지 보호**

(57) 요 약

본 개시 내용은 스케일러블 스토리지 시스템의 데이터를 보호하는 것에 관한 것이다. 스케일러블 스토리지 시스템은 복수의 노드를 포함하며, 각각의 상기 노드는 하나 이상의 하드-디스크 드라이브 및/또는 고체 상태 디스크 드라이브와 같은 직접-부착형 스토리지(DAS)를 갖는다. 노드들은 인터-노드 통신 네트워크를 통해 결합되며, 실질적 전체의 DAS는 각각의 노드에 의해 전역적으로 액세스가능하다. DAS는 노드들 중 한 노드에서 장애가 존재할 때 신뢰성 있고 전역적으로 액세스 가능한 DAS에 저장된 데이터를 유지하는 인트라-노드 보호를 이용하여 보호된다. DAS는 또한 노드들 중 적어도 한 노드가 장애인 경우 신뢰성있고 전역적으로 액세스가능한 DAS에 저장된 데이터를 유지하는 인터-노드 보호를 이용하여 보호된다.

대 표 도



(56) 선행기술조사문헌
US20090210742 A1
US20120023291 A1
US20030188097 A1
US20070050578 A1
*는 심사관에 의하여 인용된 문헌

명 세 서

청구범위

청구항 1

서로 통신하는 복수의 프로세싱 노드들을 포함하는 스토리지 시스템으로서,

각각의 프로세싱 노드는,

복수의 디스크들과,

상기 복수의 디스크들 중 선택된 디스크에 데이터를 기록하도록 구성된 적어도 하나의 호스트와,

상기 적어도 하나의 호스트에 의해 상기 선택된 디스크에 기록된 상기 데이터를 이용하여 로컬 리던던트 데이터를 결정하도록 구성되는 로컬 리던던시 계산 유닛 - 상기 로컬 리던던시 계산 유닛은 또한 상기 로컬 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하도록 구성됨 - 과,

상기 적어도 하나의 호스트에 의해 상기 선택된 디스크에 기록된 상기 데이터를 이용하여 델타 데이터를 결정하도록 구성되는 델타 계산 유닛 - 상기 델타 계산 유닛은 또한 상기 결정된 델타 데이터를 적어도 하나의 다른 프로세싱 노드에 송신하도록 구성됨 - 과,

상기 프로세싱 노드들 중 적어도 하나의 다른 노드로부터 델타 데이터를 수신하도록 구성되는 글로벌 리던던시 계산 유닛 - 상기 글로벌 리던던시 계산 유닛은 또한 상기 수신된 델타 데이터를 이용하여 글로벌 리던던트 데이터를 결정하도록 구성되며, 상기 글로벌 리던던시 계산 유닛은 또한 상기 글로벌 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하도록 구성되고, 상기 로컬 리던던트 데이터는 상기 글로벌 리던던트 데이터를 보호함 - 을 포함하는,

스토리지 시스템.

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

제1항에 있어서,

상기 로컬 리던던시 계산 유닛은 또한 상기 선택된 디스크가 장애(fail)일 때 상기 로컬 리던던트 데이터를 이용하여 상기 선택된 디스크에 기록된 상기 데이터를 복구하도록 구성된

스토리지 시스템.

청구항 22

제1항에 있어서,

상기 글로벌 리던던시 계산 유닛은 또한 적어도 하나의 다른 프로세싱 노드가 장애일 때 상기 글로벌 리던던트 데이터를 이용하여 상기 적어도 하나의 다른 프로세싱 노드의 데이터를 복구하도록 구성된

스토리지 시스템.

청구항 23

제1항에 있어서,

상기 로컬 리던던트 데이터는 제1 로컬 리던던트 데이터이고,

상기 로컬 리던던시 계산 유닛은 또한 상기 글로벌 리던던트 데이터를 이용하여 제2 로컬 리던던트 데이터를 결정하도록 구성되며,

상기 로컬 리던던시 계산 유닛은 또한 상기 제2 로컬 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하도록 구성된

스토리지 시스템.

청구항 24

제23항에 있어서,

상기 로컬 리던던시 계산 유닛은 또한 상기 선택된 디스크가 장애일 때 상기 로컬 리던던트 데이터를 이용하여 상기 선택된 디스크에 기록된 상기 데이터를 복구하도록 구성되고,

상기 글로벌 리던던시 계산 유닛은 또한 상기 적어도 하나의 다른 프로세싱 노드가 장애일 때 상기 글로벌 리던던트 데이터를 이용하여 상기 적어도 하나의 다른 프로세싱 노드의 데이터를 복구하도록 구성되며,

상기 로컬 리던던시 계산 유닛은 또한 상기 글로벌 리던던트 데이터를 저장하는 상기 적어도 하나의 디스크가 장애일 때 상기 제2 로컬 리던던트 데이터를 이용하여 상기 글로벌 리던던트 데이터를 복구하도록 구성되는

스토리지 시스템.

청구항 25

제1항에 있어서,

상기 복수의 프로세싱 노드들은 제1 글로벌 코딩 타입을 통해 보호된 상기 프로세싱 노드들 중에서 상기 복수의 디스크들 중 제1 디스크 셋 및 제2 글로벌 코딩 타입을 통해 보호된 상기 프로세싱 노드들 중에서 상기 복수의 디스크들 중 제2 디스크 셋을 포함하는

스토리지 시스템.

청구항 26

제25항에 있어서,

상기 제1 디스크 셋은 상기 복수의 프로세싱 노드들 중 제1 프로세싱 노드의 적어도 하나의 디스크 및 상기 복수의 프로세싱 노드들 중 제2 프로세싱 노드의 적어도 하나의 디스크를 포함하는

스토리지 시스템.

청구항 27

제1항에 있어서,

상기 로컬 리던던시 계산 유닛은 또한 제1 소거-정정 코드 타입을 이용하여 데이터를 처리하도록 구성되며,

상기 글로벌 리던던시 계산 유닛은 또한 제2 소거-정정 코드 타입을 이용하여 데이터를 처리하도록 구성되며,

상기 제1 소거-정정 코드 타입은 상기 제2 소거-정정 코드 타입과 상이한

스토리지 시스템.

청구항 28

서로 통신하는 복수의 프로세싱 노드들을 포함하는 스토리지 시스템으로서,

각각의 프로세싱 노드는,

복수의 디스크들과,

상기 복수의 디스크들 중 선택된 디스크에 데이터를 기록하도록 구성된 적어도 하나의 호스트와,

상기 복수의 디스크들과 통신하는 컨트롤러를 포함하며, 상기 컨트롤러는,

상기 선택된 디스크에 기록된 상기 데이터를 이용하여 로컬 리던던트 데이터를 결정하고,

상기 로컬 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하고,

상기 선택된 디스크에 기록된 상기 데이터를 이용하여 델타 데이터를 결정하고,

상기 결정된 델타 데이터를 상기 프로세싱 노드들 중 적어도 하나의 다른 프로세싱 노드에 송신하고,

상기 프로세싱 노드들 중 적어도 하나의 다른 노드로부터 델타 데이터를 수신하고,

상기 수신된 델타 데이터를 이용하여 글로벌 리던던트 데이터를 결정하고 -상기 로컬 리던던트 데이터는 상기 글로벌 리던던트 데이터를 보호함 -,

상기 글로벌 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하도록 구성된

스토리지 시스템.

청구항 29

제28항에 있어서,

상기 컨트롤러는 또한,

상기 선택된 디스크가 장애일 때 상기 로컬 리던던트 데이터를 이용하여 상기 선택된 디스크에 기록된 상기 데이터를 복구하고,

상기 적어도 하나의 다른 프로세싱 노드가 장애일 때 상기 글로벌 리던던트 데이터를 이용하여 상기 적어도 하나의 다른 프로세싱 노드의 데이터를 복구하도록 구성된

스토리지 시스템.

청구항 30

제29항에 있어서,

상기 로컬 리던던트 데이터는 제1 로컬 리던던트 데이터이고,

상기 컨트롤러는 또한,

상기 글로벌 리던던트 데이터를 이용하여 제2 로컬 리던던트 데이터를 결정하고,

상기 제2 로컬 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하고,

상기 글로벌 리던던트 데이터를 저장하는 상기 적어도 하나의 디스크가 장애일 때 상기 제2 로컬 리던던트 데이

터를 이용하여 상기 글로벌 리던던트 데이터를 복구하도록 구성된

스토리지 시스템.

청구항 31

제28항에 있어서,

상기 복수의 프로세싱 노드들은 제1 글로벌 코딩 타입을 통해 보호된 상기 프로세싱 노드들 중에서 상기 복수의 디스크들 중의 제1 디스크 셋 및 제2 글로벌 코딩 타입을 통해 보호된 상기 프로세싱 노드들 중에서 상기 복수의 디스크들 중의 제2 디스크 셋을 포함하며, 상기 제1 글로벌 코딩 타입은 상기 제2 글로벌 코딩 타입과 상이한

스토리지 시스템.

청구항 32

제31항에 있어서,

상기 제1 디스크 셋은 상기 복수의 프로세싱 노드들 중 제1 프로세싱 노드의 적어도 하나의 디스크 및 상기 복수의 프로세싱 노드들 중 제2 프로세싱 노드의 적어도 하나의 디스크를 포함하는

스토리지 시스템.

청구항 33

제28항에 있어서,

상기 컨트롤러는 또한 제1 소거-정정 코드 타입을 이용하여 상기 로컬 리던던트 데이터를 결정하고,

제2 소거-정정 코드 타입을 이용하여 상기 글로벌 리던던트 데이터를 결정하도록 구성되며, 상기 제1 소거-정정 코드 타입은 상기 제2 소거-정정 코드 타입과 상이한

스토리지 시스템.

청구항 34

스토리지 보호 방법으로서,

서로 통신하는 복수의 프로세싱 노드들 중 제1 프로세싱 노드의 복수의 디스크들 중 선택된 디스크에 데이터를 기록하는 단계와,

상기 선택된 디스크에 기록된 상기 데이터를 이용하여 로컬 리던던트 데이터를 결정하는 단계와,

상기 로컬 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하는 단계와,

상기 선택된 디스크에 기록된 데이터를 이용하여 제1 델타 데이터를 결정하는 단계와,

상기 제1 델타 데이터를 상기 적어도 하나의 다른 프로세싱 노드에 송신하는 단계와,

적어도 하나의 다른 노드로부터 제2 델타 데이터를 수신하는 단계와,

상기 제2 델타 데이터를 이용하여 글로벌 리던던트 데이터를 결정하는 단계와 -상기 로컬 리던던트 데이터는 상기 글로벌 리던던트 데이터를 보호함 -,

상기 글로벌 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하는 단계를 포함하는

스토리지 보호 방법.

청구항 35

제34항에 있어서,

상기 스토리지 보호 방법은,

상기 선택된 디스크가 장애일 때 상기 로컬 리던던트 데이터를 이용하여 상기 선택된 디스크에 기록된 상기 데이터를 복구하는 단계와,

상기 적어도 하나의 다른 프로세싱 노드가 장애일 때 상기 글로벌 리던던트 데이터를 이용하여 상기 적어도 하나의 프로세싱 노드의 데이터를 복구하는 단계를 더 포함하는

스토리지 보호 방법.

청구항 36

제35항에 있어서,

상기 로컬 리던던트 데이터는 제1 로컬 리던던트 데이터이고,

상기 스토리지 보호 방법은,

상기 글로벌 리던던트 데이터를 이용하여 제2 로컬 리던던트 데이터를 결정하는 단계와,

상기 제2 로컬 리던던트 데이터를 상기 복수의 디스크들 중 적어도 하나의 디스크에 저장하는 단계와,

상기 글로벌 리던던트 데이터를 저장하는 상기 적어도 하나의 디스크가 장애일 때 상기 제2 로컬 리던던트 데이터를 이용하여 상기 글로벌 리던던트 데이터를 복구하는 단계를 더 포함하는

스토리지 보호 방법.

청구항 37

제34항에 있어서,

상기 스토리지 보호 방법은,

제1 글로벌 코딩 타입을 이용하여 상기 프로세싱 노드들의 제1 디스크 셋을 보호하는 단계와,

제2 글로벌 코딩 타입을 이용하여 상기 프로세싱 노드들의 제2 디스크 셋을 보호하는 단계를 더 포함하며, 상기 제1 글로벌 코딩 타입은 상기 제2 글로벌 코딩 타입과 상이한

스토리지 보호 방법.

청구항 38

제37항에 있어서,

상기 제1 디스크 셋은 상기 제1 프로세싱 노드의 적어도 하나의 디스크 및 상기 복수의 프로세싱 노드들 중 제2 프로세싱 노드의 적어도 하나의 디스크를 포함하는

스토리지 보호 방법.

청구항 39

제34항에 있어서,

상기 스토리지 보호 방법은,

제1 소거-정정 코드 타입을 이용하여 상기 로컬 리던던트 데이터를 결정하는 단계와,

제2 소거-정정 코드 타입을 이용하여 상기 글로벌 리던던트 데이터를 결정하는 단계를 더 포함하며, 상기 제1 소거-정정 코드 타입은 상기 제2 소거-정정 코드 타입과 상이한

스토리지 보호 방법.

발명의 설명

기 술 분 야

배 경 기 술

[0001] 직접 부착형 디스크(directly attached disks)를 가진 스케일러블 스토리지 시스템은 데이터 보호를 위한 리던던시 메커니즘(redundancy mechanisms)을 필요로 한다. 단일 서버와 같은 단일 노드에서는 RAID-5, RAID-6, 다른 RAID 레벨, 또는 그의 변형과 같은 직접 부착형 스토리지(directly-attached storage (DAS))의 보호를 위한 다양한 기술이 사용된다. 대형 JBOD 콤플렉스와 같은 분산 시스템 또는 대규모 스토리지 시스템에서는 오류 정정 코딩(error-correction coding)을 복수의 디스크들에 걸쳐 분산시킴으로써 보호하는 소거-코딩 기술(erasure-coding techniques)이 사용된다. 그러나, 소거-코딩은 대량의 데이터의 수송(shipping)(즉, 송신 및 수신)을 필요로 한다. 일부 실시예에서, n개 드라이브 장애 중 r개를 다루기 위해 r개의 개별 디스크에서 데이터가 업데이트되어야 한다. 노드 장애에 대비하여 복원력을 겸비할 때, 전술한 시스템은 리던던시의 양에서 및/또는 업데이트 또는 복구를 위해 노드들 사이에서 수송될 데이터의 양에서 아주 비용이 많이드는 경향이 있다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0002] 개시내용의 실시예는 서로 통신하는 복수의 프로세싱 노드를 포함하는 스토리지 시스템에 관련된다. 각각의 프로세싱 노드는 적어도 하나의 호스트와 통신하는 복수의 디스크들을 포함한다. 호스트는 데이터를 복수의 디스크들 중 선택된 디스크에 기록하기 위해 구성된다. 로컬 리던던시 계산 유닛은 호스트에 의한 선택된 디스크에 기록된 데이터를 이용하여 로컬 리던던트 데이터를 결정하기 위해 구성된다. 로컬 리던던시 계산 유닛은 또한 로컬 리던던트 데이터를 복수의 디스크들 중 적어도 하나의 디스크에 저장하기 위해 구성된다. 델타 계산 유닛은 호스트에 의한 선택된 디스크에 기록된 데이터를 이용하여 델타 데이터를 결정하기 위해 구성된다. 델타 계산 유닛은 또한 델타 데이터를 적어도 하나의 다른 프로세싱 노드에 송신하도록 구성된다. 글로벌 리던던시 계산 유닛은 적어도 하나의 다른 프로세싱 노드로부터 델타 데이터를 수신하기 위해 구성된다. 글로벌 리던던시 계산 유닛은 또한 다른 프로세싱 노드로부터 수신한 델타 데이터를 이용하여 글로벌 리던던트 데이터를 결정하고 글로벌 리던던트 데이터를 복수의 디스크들 중 적어도 하나의 디스크에 저장하기 위해 구성된다.

[0003] 전술한 개괄적인 설명과 다음의 상세한 설명은 모두 반드시 개시 내용을 제한하는 것이 아님은 물론이다. 명세서의 일부에 포함되고 그 일부를 구성하는 첨부 도면은 개시 내용의 실시예를 예시한다.

도면의 간단한 설명

[0004] 개시 내용의 실시예는 첨부 도면을 참조함으로써 본 기술에서 통상의 지식을 가진 자들에게 보다 잘 이해될 수 있다.

도 1은 개시 내용의 실시예에 따라서 스케일러블 스토리지 시스템을 예시하는 블록도이다.

도 2는 개시 내용의 실시예에 따라서 호스트 데이터 기록을 처리하는 방법을 예시하는 흐름도이다.

도 3은 개시 내용의 실시예에 따라서 델타 데이터를 처리하는 방법을 예시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0005] 이제 첨부 도면에 예시된 바와 같은 개시된 실시예가 상세히 설명될 것이다.

[0006] 도 1 내지 도 3은 일반적으로 적어도 하나의 스케일러블 스토리지 시스템을 보호하기 위한 시스템 및 방법의 실시예를 예시한다. 스케일러블 스토리지 시스템에서의 몇 가지 과제는 모든 데이터로의 글로벌 액세스와, 디스크 장애로부터 복원력과, 하나 이상의 프로세싱 노드의 장애를 처리하는 메커니즘의 복합적인 것을 제공하는 것을 포함한다. 전술한 과제 중 적어도 일부는 하드 디스크 드라이브(HDD) 장애와 같은 인트라-노드(intra-node) 장애로부터 보호하는 인트라-노드 레벨에서 리던던시를 인트라-노드 보호의 장애와 같은 노드들 중 하나 이상의 노드의 장애로부터 보호하는 인터-노드 레벨에서 리던던시와 균형을 이루게 함으로써 달성된다. 일부 실시예에서, 노드들에서 분산 방식으로 캐싱하는 것은 노드 각각의 로컬 성능을 더욱 개선하여 주며 데이터 보호를 위해 사용된 기록들의 조기 확인응답(acknowledgement)을 가능하게 해줌으로써 스케일러블 스토리지 시스템의 시스템-레벨 성능을 개선한다.

[0007] 도 1은 스케일러블 직접 부착형 스토리지(scalable directly-attached storage (DAS))와 같은 스토리지 시스템(100)의 실시예를 예시하지만, 이것으로 제한되지 않는다. 시스템(100)은 서버와 같은 복수의 프로세싱 노드(102)를 포함한다. 각각의 프로세싱 노드(102)는 (하나 이상의 프로세서 또는 CPU와 같은) 각자의 (즉, 로컬) 호스트(104) 및 복수의 디스크들 드라이브(106)와 같은 각자의 (즉, 로컬) DAS(106)를 포함한다. 여러 실시예에서, 로컬 DAS(106)는 각자의 하나 이상의 (즉, 로컬) I/O 컨트롤러(108)를 통해 로컬 호스트(104)와 통신가능하게 연결되어 있다. 실질적인 전체의, 이를 테면, 모든 스토리지(106A-106Z)는 프로세싱 노드(102) 모두에게 전역적으로 식별된다. 특정 프로세싱 노드(102A)의 DAS(106A-106C)는 제각기 프로세싱 노드(102)의 "로컬 스토리지(local storage)"라고 불리운다. 다른 프로세싱 노드(102B-102M)의 DAS(106D-106Z)는 제각기 특정 프로세싱 노드(102A)의 "외부 스토리지(foreign storage)"라고 불리운다. 프로세싱 노드(102)는 인터-노드 통신 네트워크(116), 이를 테면, 이것으로 제한되지 않지만, 직렬 부착형 소형 컴퓨터 시스템 인터페이스(serial attached small computer system interface (SAS)) 스위칭 인터커넥트를 통해 서로 통신한다. 프로세싱 노드(102)는 인터-노드 통신 네트워크(116)를 통해 실질적 전체의 스토리지(106A-106Z)에 액세스한다. 그러나, 일부 실시예에서, 특정 프로세싱 노드(102A)의 각각의 로컬 스토리지(106A-106C)는 각각의 외부 스토리지(106D-106Z)에 액세스하는 것 보다 더 빠르며/빠르거나 대역폭이 더 높다. 일부 실시예에서, 인터-노드 통신 네트워크(116)는 적어도 하나의 SAS 패브릭, 이더넷 네트워크, 인피니밴드 네트워크(InfiniBand network), PCIe(PCI-e(Peripheral Component Interconnect express) 상호접속 네트워크, 근거리 통신 네트워크(LAN), 광 대역 통신 네트워크(WAN), 사유 네트워크(proprietary network), 또는 이들의 어떤 조합을 포함하지만, 이것으로 제한되지 않는다.

[0008] 일부 실시예에서, 시스템(100)은 스토리지(106)의 공유를 용이하게 해주는 로킹 및/또는 코히어런시 메커니즘을 더 포함한다. 예를 들면, 디렉토리-기반의 캐싱 메커니즘은 데이터의 소유권 및/또는 변경을 추적할 수 있게 해준다. 일부 실시예에서, 각각의 프로세싱 노드(102)는 자주 액세스되는 데이터를 저장하는 디스크 캐시와 같은 캐시를 포함한다. 여러 실시예에 따르면, 자주 액세스된 데이터의 일부는 프로세싱 노드에 대해 로컬이며/이거나 자주 액세스되는 데이터의 일부는 포린(foreign)이다. 일부 실시예에서, 디스크 캐시는 고체 상태 디스크 드라이브(SSD)를 포함하지만, 이것으로 제한되지 않는다.

[0009] 멀티-노드 스토리지 시스템(100)에서 몇 가지 관심의 장애 시나리오는 다음과 같은 것을 포함한다.

[0010] - 프로세싱 노드(102)의 HDD 또는 SSD(106)와 같은 하나 이상의 입력/출력(I/O) 장치의 장애,

[0011] - 프로세싱 노드(102) 중 하나의 프로세싱 노드 내 I/O 장치(106) 중 하나 이상의 장치에 이르는 경로의 장애,

[0012] - 호스트(104) 또는 인트라-노드 통신 인프라스트럭처와 같은 프로세싱 노드(102)의 일부 또는 전부의 장애,

[0013] - 프로세싱 노드들(102)을 연결시키는 인터-노드 통신 네트워크(116)와 같은 상위 레벨 통신 인프라스트럭처의 장애.

[0014] 이러한 장애는 인트라-노드 장애 또는 인터-노드 장애로 분류된다. 인트라-노드 장애는 프로세싱 노드(102)의

적어도 일부를 사용불능으로 만들지만 프로세싱 노드(102)에 대해 로컬인 데이터로의 글로벌 액세스를 포함하여 프로세싱 노드(102)의 지속적인 동작을 못하게 하지 않는 장애이다. 인터-노드 장애는 프로세싱 노드(102) 또는 프로세싱 노드(102)에 대해 로컬인 데이터의 적어도 일부를 사용불능으로 만드는 장애이다. 일부의 인트라-노드 장애는 장애가 일어난 프로세싱 노드(102)의 레벨에서 탄력적 대처가 가능하며, (가능한 성능 충격을 제외하고) 다른 프로세싱 노드(102)에 전역적으로 식별가능하지 않다.

[0015] 장애는 또한 경(hard) 장애(예를 들면, 잘 풀리지 않는, 반복적인) 또는 연(soft) 장애 (예를 들면, 일회성, 일시적, 파워 사이클 후 사라짐)로서 특징지워진다. 많은 노드 장애는 소프트웨어 충돌과 같은 연 장애이며, 그래서 일시적이거나 지속기간이 짧다. 디스크 장애도 또한 연 장애 (예를 들면, 새로운 데이터를 기록함으로써 복구 가능한 일시적인 수정 불능 오류) 또는 경 장애(예를 들면, 헤드 충돌로 인한 디스크의 장애) 중 어느 하나이다. 장애 지속 기간, 그래서 경 장애 대 연 장애 분류는 얼마나 많은 각종 형태의 동시적인 에러가 고려되는지에 기반하여 장애의 가능성을 계산하는데 관련이 있다. 일부 실시예에서, 대부분의 프로세싱 노드 장애가 연 장애인 경우에 복수의 프로세싱 노드가 동시에 장애를 일으키는 확률은 대부분의 프로세싱 노드 장애가 경 장애인 경우에 복수의 프로세싱 노드가 동시에 장애를 일으키는 확률보다 적다.

[0016] 시스템-레벨의 장애는 어느 프로세싱 노드(102)에 저장된 호스트-기록된(host-written) (즉, 논-리던던트(non-redundant)) 데이터 중 어느 데이터의 복구불능 손실 또는 프로세싱 노드(102)의 특정 개수보다 많은 손실과 같은 멀티-노드 스토리지 시스템의 장애이다. 일부 실시예에서, 시스템(100)은 적어도 부분적으로는 시스템-레벨 장애의 확률을 특정 값보다 적게 줄이도록 설계된다.

[0017] 단순한 소거-코딩 해결책은 다량의 리던던시 및/또는 데이터 수송을 수반하는 경향이 있다. 예를 들면, 각기 n개 (HDD 또는 SSD와 같은) 디스크(106)를 포함하는 m개 노드(102)를 고려해보면, 총 m*n 개 디스크(106)이다. 어느 세개의 디스크(106)를 장애로부터 보호하기 위해, 적어도 세개의 디스크(106)가 리던던트 데이터를 포함하여야 한다. 다른 (m*n-3) 데이터(즉, 논-리던던트) 디스크(106) 중 어느 디스크에 기록하려면 3개의 리던던트 디스크(106)를 업데이트하는 것이 필요하다. 프로세서와 같은 호스트(104)가 데이터 디스크(106) 중 하나의 디스크에 작고 랜덤한 기록(예를 들면, 4KB 또는 8KB)을 수행할 때, 총 네 개의 유사한 크기의 기록이 수행되어야 하며, 네 개 중 세 개의 기록은 계산(즉, 호스트 기록 이전의 구 데이터(old data) 및 호스트에 의해 기록된 새로운 데이터(new data)에 기반하여 리던던트 데이터를 업데이트하는 것)을 수반한다. 더욱이, 만일 하나 이상의 노드 장애가 소거-코딩으로 처리된다면, 세 개의 리던던트 디스크(106)는 가급적이면 상이한 노드(102)에 배치되는 것이 더 좋다. 따라서, 호스트 기록은, 선택된 데이터 디스크(106A)를 포함하는 노드(102A)의 선택된 데이터 디스크로부터 구 데이터를 판독하기와, 새로운 데이터를 선택된 데이터 디스크(106A)에 기록함으로써 구 데이터를 호스트(104)에 의해 제공된 새로운 데이터로 교체하기와, 구 데이터와 새로운 데이터 간의 델타와 같은 함수를 계산하기와, 델타를 상이한 노드(102)에 배치될 수 있는 세개의 리던던트 디스크(106)로 수송하기와, 리던던트 디스크(106) 중 하나를 포함하는 각 노드(102) 상의 리던던트 데이터의 구 버전을 판독하기와, 델타를 이용하여 리던던트 데이터로에 대한 업데이트를 결정하기와, 리던던트 데이터의 새로운 버전으로 다시 기록하기를 필요로 한다. 델타를 복수의 노드(102)에 수송하는 것은 지연 시간과 전력을 둘 다 소모한다. 일부 실시예에서, 호스트 기록 데이터가 "안전(safe)"할 때까지 호스트 기록이 확인응답될 수 없기 때문에 또 다른 지연이 발생하며, 호스트 기록 데이터는 리던던트 데이터 기록이 완료될 때까지 안전하지 않다.

[0018] RAID와 같은 노드(102) 내의 장애에 대해 잘 작동하는 단일 보호 해결책(single protection solution)은 복수의 노드(102) 전체에는 적합하지 않을 수 있다. 전술한 예에서 예시된 소거-코딩과 같은 전역적 해결책은 노드(102) 사이에서 수송되는 데이터 량의 면에서 너무 비싸다. 더욱이, 각종 장애 시나리오마다 상이한 가능성을 갖는다. 전형적으로, 시스템 장애의 확률을 줄이는 것이 디스크 장애의 확률 또는 노드 장애의 확률을 개별적으로 줄이는 것보다 더욱 중요하다. 일부 실시예에서, 시스템(100)은, 노드들 사이에서 더 적은 데이터를 수송하기, 성능 더 높이기, 비용 더 낮추기(예를 들면, 소정의 시스템-레벨 장애 확률에 필요한 리던던시의 양을 낮추기), 전력을 더 낮추기, 지연을 더 낮추기, 및 다른 전력, 비용 및 성능 매트릭스 중 하나 이상을 성취하기 위해 구성된다. 예를 들면, 개개의 하드 디스크 드라이브(106)의 장애는 아주 그럴 수 있다. 그러므로, 일부 실시예에서, 그러므로, 시스템-장애의 확률은 더 많은 리던던시를 하드 디스크 드라이브 장애로부터 보호하는데 제공하고 노드 장애에 대해서는 더 적게 제공함으로써 줄어들며, 그럼으로써 성능을 과도하게 절충하지 않고 또는 높은 데이터 수송이나 리던던시 비용을 필요로 하지 않고 시스템-장애의 확률을 줄일 수 있다.

[0019] 실시예(도 1 참조)에서, 시스템(100)은 노드(102) 내 I/O 장치(106)에 저장된 데이터를 보호하는 제1 형태의 보호(즉, "내부", "로컬", 또는 "인트라-노드" 보호) 및 하나 이상의 노드(102)의 장애로부터 보호하는 제2 형태의 보호(즉, "외부", "글로벌" 또는 "인터-노드" 보호)를 포함한다. 전술한 스케일러블 스토리지 보호 형태는

보호 및 복구를 위해 노드들(102) 사이에서 수송되어야 하는 데이터 량을 줄여준다. 더욱이, 델타-캐싱 메커니즘은 호스트 기록이 안전하게 저장되는 것을 확인응답하는 데 필요한 시간을 줄여준다.

[0020] 일부 실시예에서, 시스템(100)은 로컬(즉, 인트라-노드) 대 글로벌(즉, 인터-노드) 장애로부터 보호하는 별개의 메커니즘(110, 114)을 포함한다. 다른 실시예에서, 로컬 보호 메커니즘(110) 및 글로벌 보호 메커니즘(114)은 각기 각자의 장애 확률을 줄이도록 선택됨으로써, 전체 시스템-레벨 장애 확률을 특정 레벨까지 줄일 수 있다. 여러 실시예에서, 로컬 보호 메커니즘(110) 및 글로벌 보호 메커니즘(110)는 각기 리던던트 데이터 저장 및 장애로부터 복구를 위해 노드(102) 사이에서 수송되는 데이터 량을 줄이도록 선택된다.

[0021] 일부 실시예에서, 스케일러블 스토리지 보호를 갖춘 시스템(100)은 비용 면에서 장점을 제공한다. 예를 들면, 이전에 기술된 m개 노드(102)가 있고, 각 노드가기 n개 디스크(106)를 가지며 3개 디스크 장애로부터 보호하는 요건이 있으며, 또한 리던던트 디스크들(106)이 모두 상이한 노드(102) 상에 존재한다고 가정하는 단순 소거-코딩을 고려해보자. 단순 소거-코딩 접근법은 리던던시를 위해 데이터를 다른 노드(102)에 기록하는 것 만큼의 데이터를 3회에 걸쳐 수송하는 것이다. 스케일러블 스토리지 보호를 갖춘 시스템(100)에 의해 제안된 다층 보호는 융통성 있게 균형을 유지하여 준다. 예를 들면, 일부 실시예에서, 시스템(100)은 다양한 장애 확률(예를 들면, 경 장애 대 연 장애) 또는 비용이 드는 요인(예를 들면, 데이터를 수송하는 비용)을 기반으로 하여 설계된다.

[0022] 전술한 단순 소거-코딩 접근법 대신 스케일러블 스토리지 보호를 갖춘 시스템(100)의 예시적인 실시예에서, 각각의 노드(102)에서 n개 디스크(106) 중 두 개의 디스크는 그 노드(102)의 리던던트 로컬 데이터를 포함하고 있으며, m 개 노드(102) 중 하나 이상의 노드(즉, 리던던트 노드)는 전역적으로 리던던트 데이터를 포함하고 있다. 하나의 리던던트 노드(102)를 가진 실시예에서, 호스트(104)가 데이터 디스크(106) 중 하나의 디스크에 대해 작고 랜덤한 기록(예를 들면, 4KB 또는 8KB 기록)을 수행할 때, 총 네 개의 더 작은 크기의 기록이 수행되어야 하지만, 유사한 크기의 기록 중 세 개(즉, 호스트 기록 데이터 및 두 로컬 리던던트 데이터 기록)는 로컬이다. 더 작은 크기의 기록 중 단지 하나의 기록만이 리던던트 노드(102)에 수송되어야 한다. 단순 소거-코딩과 비교하여 볼 때, 수송될 데이터 량은 줄어든다(예를 들면, 2/3 만큼). 전술한 예에서, 스케일러블 스토리지 보호를 갖춘 시스템(100)은 적어도 세개의 디스크 장애를 처리할 수 있다. 일부 실시예에서, 시스템(100)은 노드(102) 당 두개 디스크 장애를 처리하는 것이 가능해진다.

[0023] 전술한 예에서, 한 개의 노드(102)에서 세개의 디스크 장애는 각 노드(102)의 두 리던던트 디스크(106)가 노드(102)에서 n 개 디스크(106) 중 두 디스크의 장애에 대해서 정정할 수 있을 뿐이기 때문에 노드(102)의 장애와 실질적으로 같거나 유사하다. 일부 실시예에서, 인트라-노드 보호의 장애 확률은 노드(102)의 장애 확률에 포함되며 적어도 부분적으로 인터-노드 보호의 요구된 레벨을 결정하는데 이용된다. 단순, 소거-코딩 접근법은 세 개의 노드 장애까지 다룰 수 있지만, 이 결과는 전역적으로 리던던트한 데이터를 처리하기 위해 더 높은 퍼센티지의 노드(102)들이 사용된다는 것이다. 만일 노드 장애 확률이 디스크 장애 확률과 비교해 적다면, 스케일러블 스토리지 보호의 대안은 대등하거나 또는 I/O 수송 및 리던던시 중 적어도 하나에서 더 낮은 비용으로 더 좋은 보호를 제공한다.

[0024] 전술한 예들은 단순 소거-코딩 보호 시스템과 비교한 스케일러블 스토리지 보호를 갖춘 시스템(100)의 적어도 몇 가지 장점을 보여준다. 그러나, 이 예들은 어떤 형태로든 개시 내용을 제한하려는 것은 아니다. 여러 실시예에 따르면, 시스템(100)은 본 명세서에서 개요적으로 기술된 스케일러블 스토리지 보호 구성을 구현하는 선택된 파라미터 및 구성의 모든 조합을 포함한다. 실시예에서, 시스템(100)은 m개 노드(102)를 포함하고, 각 노드는 n개 디스크(106)를 갖는다. 시스템(100)은 k개 노드 장애(예를 들면, k=2)를 회생시키도록 구성된다. g개 디스크의 각 그룹은 디스크-레벨의 장애를 적절하게 다루는 적어도 h개 리던던트 디스크(106)를 포함한다(예를 들면, g=10 디스크(106) 중 h=3 이 리던던트 디스크이다).

[0025] 일부 실시예에서, h를 적절히 스케일링함으로서 g=n 이 된다. 따라서, 시스템(100)은 총 m*n 디스크(106)를 포함하며 디스크(106) 중 h*m 개가 리던던트 데이터를 저장한다. m개 노드 장애 중 k개를 회생하기 위해, 하나의 코드워드(예를 들면, 하나의 보호 그룹) 내 리던던트 디스크(106)는 적어도 k개의 상이한 노드(102) 상에 있다. m개 노드(102)는 그 어느 노드도 동일한 코드워드에 의해 보호되는 리던던트 디스크(106) 중 h*m/k 보다 많이 가질 수 없다. 그렇지 않다면, k개 노드 장애가 회생되지 않을 수 있다. 그러므로, 실시예에서, n은 h*m/k 보다 크거나 또는 리던던트 데이터는 노드(102) 중 k개 보다 많은 노드에서 존재하여야 한다. 예를 들면, 만일 n=10, m=8, h=3, and k=2 이면, 80 중 24개 리던던트 디스크(106)가 필요하다. 그러나, 노드(102) 당 단지 10 개 디스크(106) 밖에 없고, 그래서 리던던트 디스크는 k가 2일뿐 일지라도 적어도 세개의 노드들에 분산되어

있어야 한다.

[0026] 소거-코딩은 신뢰성 요건을 만족시킬 수 있지만, 다음과 같은 것을 포함하는 복수의 결점을 가지고 있다. g개 소거 코드 중 h개는 계산 상 비용이 든다. 만일 h가 k 보다 크면, 한 노드(102)는 복수의 소거 코드 업데이트를 처리하여야 하고 (이것은 균형이 맞지 않은 계산 노력을 가져온다) 또는 필요한 I/O 수송은 k 보다는 h에 비례한다. 만일 n 이 h*m/k 보다 작으면, I/O 수송은 k에 비례하는 것 보다 더 크다. 일반적으로 단일 디스크 장애로부터 복구하는 것 조차도 I/O 수송을 필요로 한다. 더욱이, 시스템-레벨 성능은 적어도 하나의 장애가 있는 디스크(106)를 갖는 것이 통상적이며 I/O 수송은 흔히 복구에 필요한 것이기 때문에 시스템-레벨의 성능은 전형적으로 열악하다.

[0027] 스케일러블 스토리지 보호를 갖춘 시스템(100)은 로컬 리던던시를 이용하여 디스크 장애와 같은 인트라-노드 장애로부터 보호하는 인트라-노드 보호 메커니즘(110) 및 글로벌 리던던시를 이용하여 노드 장애와 같은 인터-노드 장애로부터 보호하는 인터-노드 보호 메커니즘(110)을 포함한다. 여러 실시예에 따르면, 시스템(100)은, I/O 수송은 선택된 개수의 회생 가능한 노드 장애에 기반하며 디스크 장애의 처리에 직교하고, 하드 디스크 장애는 인트라-노드 보호를 이용하여 복구 가능해지는 특정한 신뢰성 레벨까지 I/O 수송없이 국부적으로 복구가능하고, 특정 레벨의 시스템-장애 확률을 성취하기 위해 더 짧고 더 간단한 코딩 타입이 사용되어 하드웨어를 더욱 효율적이 되게 하며, 다른 성능, 효율 및 또는 스케일러빌리티적인 장점 중의 하나 이상을 포함하는 여러 장점을 제공한다.

[0028] 인트라-노드 보호 메커니즘(110)은 한가지 이상의 코딩 타입, 이를 테면, RAID-1; RAID-2; RAID-3; RAID-4; RAID-5; RAID-6; 어느 다른 RAID 레벨; 리드-솔로몬 코드, 파운틴 코드(fountain code), 랩토 코드(Raptor code), 레이트리스-소거 코드(rate-less erasure code), 또는 온라인 코드(Online code)와 같은 소거 코드; 및 이들의 어떤 조합 중의 하나 이상을 포함한다. 인터-노드 보호 메커니즘(114)은 한가지 이상의 코딩 타입, 이를 테면, RAID-1; RAID-2; RAID-3; RAID-4; RAID-5; RAID-6; 어느 다른 RAID 레벨; 리드-솔로몬 코드, 파운틴 코드(fountain code), 랩토 코드(Raptor code), 레이트리스 소거 코드(rate-less erasure code), 또는 온라인 코드(Online code)와 같은 소거 코드; 및 이들의 어떤 조합 중의 하나 이상을 포함한다.

[0029] 인트라-노드 보호 메커니즘(110) 또는 인터-노드 보호 메커니즘(110) 중 한 사례에 의해 보호된 복수의 디스크들(106)에 저장된 데이터는 코드워드라고 지칭된다. 예를 들면, 다섯 디스크 ◎이중 하나의 디스크는 RAID-5에서와 같이 리던던트임- 에 저장된 데이터는 개별적으로 판독가능하고 교정가능한 데이터의 셋별로 하나의 코드워드를 나타낸다. RAID-5는 바이트 레벨에서 동작가능하며, 반면에 많은 디스크는 데이터의 512B 섹터를 판독할 수 있을 뿐이며, 그래서 그러한 경우, 각각의 코드워드는 다섯 디스크 각각마다 하나의 섹터가 모인 복수의 512B 섹터일 것이다.

[0030] 일부 실시예에서, 인트라-노드 보호 메커니즘(110) 및 인터-노드 보호 메커니즘(114)는 둘 다 동일한 코딩 타입을 위해 구성된다. 예를 들면, 여러 실시예에서, 인트라-노드 보호 메커니즘(110) 및 인터-노드 보호 메커니즘(114)은 둘 다 RAID-6와 같은 2-소거-정정 코드(two-erasure-correcting code)를 사용하거나, 또는 둘 다 RAID-5와 같은 1-소거-정정 코드(one-erasure-correcting code)를 사용할 수 있다. 다른 실시예에서, 인트라-노드 보호 메커니즘(110) 및 인터-노드 보호 메커니즘(114)은 상이한 코딩 타입을 사용한다. 예를 들면, 몇 몇 사용 시나리오에서, 인트라-노드 보호 메커니즘(110)은 2-소거-정정 코드를 사용하며 인터-노드 보호 메커니즘(114)은 RAID-5와 같이 1-소거-정정 코드를 사용한다.

[0031] 인트라-노드 보호 메커니즘(110) 및 인터-노드 보호 메커니즘(114)의 계산은 각자의 코딩 타입을 따른다. 예를 들면, 1-소거-정정 RAID-5 코딩 타입은 XOR 계산을 필요로 하며, RAID-6 코딩 타입은 리드 솔로몬 코드와 같은 2-소거-정정 코드에 따른 계산을 필요로 한다.

[0032] 시스템(100)의 복수의 프로세싱 노드(102)는 각기 각 노드(102)의 복수의 디스크들(106)과 통신하는 프로세서와 같은 적어도 하나의 호스트(104)를 포함한다. 일부 실시예에서, 호스트(104)는 적어도 하나의 싱글 코어 또는 멀티 코어 CPU를 포함하지만, 이것으로 제한되지 않는다. 일부 실시예에서, I/O 컨트롤러(108)는 디스크(106)를 호스트(104)에 결합하도록 구성된다. 각각의 노드(102)는 또한 캐시 메모리 및/또는 DRAM 메모리와 같은 로컬 메모리를 포함한다. 각각의 노드(102)는 또한 하드 디스크 드라이브 및/또는 고체 상태 디스크와 같은 각자 하나 이상의 디스크(106)의 셋을 더 포함한다. 각각의 노드(102)는 또한 네트워크 인터페이스 카드 또는 본 기술에서 공지된 네트워크형 프로세싱 시스템 내에 존재하는 어느 다른 컴포넌트넌트와 같은 인터-노드 통신 네트워크(116)를 통해 노드들과 통신가능하게 결합하는 인터-노드 통신 메커니즘을 포함한다.

[0033] 일부 실시예에서, 호스트(104)는 하나 이상의 멀티 코어 x86-아키텍처 CPU 칩을 포함한다. 일부 실시예에서, I/O 컨트롤러(108)는 레이드-온-칩 컨트롤러(Raid-On-Chip controller (ROC))를 포함하며, 호스트(104)는 PCIe 인터커넥트를 통해 I/O 컨트롤러(108)에 결합된다. 일부 실시예에서, 하나 이상의 디스크 드라이브(106)는 하나 이상의 SAS 및/또는 SATA 하드 디스크 드라이브를 포함한다. 일부 실시예에서, 하나 이상의 디스크 드라이브(106)는 하나 이상의 고체 상태 디스크 드라이브를 포함한다. 일부 실시예에서, 인터-노드 통신 메커니즘은 I/O 컨트롤러(108) 내에 통합된다. 예를 들면, ROC는 로컬 디스크(106)와의 SAS 및/또는 SATA 연결성 및 SAS 패브릭을 통한 다른 프로세싱 노드(102)의 디스크(106)와의 SAS 및/또는 SATA 연결성을 제공한다.

[0034] 각각의 프로세싱 노드(102)는 또한 노드(102)의 디스크(106)에 저장된 데이터의 보호를 위해 리던던트 데이터를 결정하도록 구성된 각자의 인트라-노드 리던던시 계산 유닛(110)을 포함한다. 각각의 프로세싱 노드(102)는 또한 인트라-노드 리던던시 계산 유닛(110)에 의해 국부적으로 사용된 및/또는 노드(102)의 디스크(106)에 저장된 데이터의 기록에 응답하여 다른 노드(102)에 송신된 델타 데이터를 결정하도록 구성된 각자의 델타 리던던시 계산 유닛(112)을 더 포함한다. 각각의 프로세싱 노드(102)는 또한 노드(102)의 디스크(106)에 저장된 데이터의 보호를 위해 리던던트 데이터를 결정하도록 구성된 인터-노드 리던던시 계산 유닛(114)을 포함한다.

[0035] 일부 실시예에서, 리던던시 계산 유닛(110, 112 및 114) 중 하나 이상은 단일 메커니즘으로 결합되며/결합되거나 하나 이상의 컴포넌트를 공유한다. 예를 들면, 여러 실시예에 따르면, 리던던시 계산 유닛(110, 112 및 114)은 하나 이상의 전자 회로 또는 적어도 하나의 프로세서에 의해 캐리어 미디어로부터 실행된 프로그램 명령어와 같은 별개 또는 조합된 하드웨어, 소프트웨어 및/또는 펌웨어 모듈 로 구현된다. 일부 실시예에서, 컨트롤러(108)는 리던던시 계산 유닛(110, 112 및 114) 중 하나 이상을 포함하며/포함하거나 리던던시 계산 유닛(110, 112 및 114)의 하나 이상의 기능을 수행하도록 구성된다.

[0036] 일부 실시예에서, 제1 인트라-노드 보호 메커니즘(예를 들면, RAID-5)은 제1 프로세싱 노드(102A)의 디스크(106)의 제1 서브셋을 보호하며, 제1 인트라-노드 보호 메커니즘과 상이한 제2 인트라-노드 보호 메커니즘(예를 들면, RAID-6)은 제1 프로세싱 노드(102A)의 디스크(106)의 제2 서브셋을 보호한다. 다른 실시예에서, 디스크(106)의 제1 서브셋은 디스크(106)의 제2 서브셋과 상이한 형태의 서브셋이다. 예를 들면, 디스크(106)의 제1 서브셋은 하나 이상의 HDD를 포함할 수 있으며, 디스크(106)의 제2 서브셋은 하나 이상의 SSD를 포함할 수 있다. 일부 실시예에서, 제1 인터-노드 보호 메커니즘은 디스크(106)의 제1 서브셋의 인터-노드 보호를 디스크(106)에 제공하며, (제1 인터-노드 보호 메커니즘과 상이한) 제2 인터-노드 보호 메커니즘은 디스크(106)의 제2 서브셋의 인터-노드 보호를 디스크(106)에 제공한다.

[0037] 일부 실시예 및/또는 사용 시나리오에서, 프로세싱 노드(102) 중 한 노드의 둘 이상의 디스크(106)는 인터-노드 보호 메커니즘(114)의 동일한 코드워드에 의해 보호된다. 다른 실시예 및/또는 사용 시나리오에서, 프로세싱 노드(102) 중 어느 노드의 디스크(106) 중 하나 보다 많은 디스크는 그 어느 것도 인터-노드 보호 메커니즘(114)의 동일한 코드워드에 있지 않는다.

[0038] 일부 실시예에서, 제1 프로세싱 노드(102A)의 호스트(104A)에 의한 제1 프로세싱 노드(104A)의 디스크(106) 중 한 디스크로의 데이터의 기록은 제1 프로세싱 노드(102A)의 다른 디스크(106)에 저장된 제1 로컬(즉, 인트라-노드) 리던던트 데이터의 업데이트를 유발한다. 호스트 데이터 기록 역시 제2 프로세싱 노드(102B)의 디스크(106) 중 적어도 일부의 디스크에 저장된 글로벌(즉, 인터-노드) 리던던트 데이터의 업데이트를 유발한다. 일부 실시예에서, 글로벌 리던던트 데이터의 업데이트는 제2 프로세싱 노드(102B)의 다른 디스크들에 저장된 제2 로컬 리던던트 데이터의 업데이트를 유발한다. 일부 실시예에서, 호스트 데이터 기록은 제1 프로세싱 노드(102A)가 장애일지라도 그 호스트 데이터 기록이 복구 가능할 때와 같이 안전 시점에 도달하는 글로벌 리던던트 데이터의 업데이트 이후 확인응답된다.

[0039] 도 2 및 도 3은 스케일러블 스토리지 보호를 위해 각기 델타 기록을 처리하는 방법(200) 및 델타 데이터를 처리하는 방법(300)을 예시한다. 시스템(100)은 방법(200 및 300)의 시현예이며 시스템(100) 또는 방법(200 또는 300)의 실시예에 관해 기술된 모든 단계 또는 특징은 시스템(100) 및 방법(200 및 300) 둘 다에 적용할 수 있다. 그러나, 방법(200 또는 300)의 하나 이상의 단계는 본 기술에서 공지된 다른 수단을 통해 실행될 수 있음을 알아야 한다. 본 명세서에 기술된 시스템(100)의 실시예는 어떤 방식으로든 방법(200 또는 300)을 제한하는 것으로 해석되지 않아야 한다.

[0040] 단계(202)에서, 데이터는 제1 프로세싱 노드(102A)의 호스트(104A)에 의해 선택된 논리 블록 어드레스(logical block address (LBA))에 기록된다. 단계(204)에서, 데이터를 선택된 LBA에 저장하는 제1 프로세싱 노드(102A)의 적어도 하나의 목적지 디스크(106) 및 목적지 디스크(106)를 위한 인트라-노드 보호 데이터를 저장하는 제

1 프로세싱 노드(102A)의 하나 이상의 리던던트 디스크(106)가 결정된다. 일부의 실시예에서, 목적지 및 인트라-노드 리던던트 디스크(106)는 제1 프로세싱 노드(102A)의 호스트(104A) 및 컨트롤러(108A) 중 적어도 하나에 의해 결정된다. 예를 들면, 제1 프로세싱 노드(102A)의 호스트(104A)에서 실행하는 드라이버 소프트웨어는 목적지 디스크(106)를 결정하며, 컨트롤러(108A)는 리던던트 디스크(106)를 결정한다. 단계(206)에서, 목적지 디스크(106)를 위한 인트라-노드 보호 데이터를 저장하는 하나 이상의 리던던트 프로세싱 노드(102)는 제1 프로세싱 노드(102A)의 호스트(104A) 및 컨트롤러(108A) 중 적어도 하나에 의해 결정된다.

[0041] 단계(208)에서, 구 데이터가 선택된 LBA에서 목적지 디스크(106)로부터 판독된다. 단계(212)에서, 호스트 데이터 기록의 새로운 데이터가 선택된 LBA에서 목적지 디스크(106)에 기록된다. 단계(210)에서, 제1 프로세싱 노드(102A)의 델타 계산 유닛(112A)은 신규 데이터 및 구 데이터를 이용하여 델타 데이터를 결정한다. 단계(214)에서, 제1프로세싱 노드(102A)의 인트라-노드 리던던시 계산 유닛(110A)은 델타 데이터에 따라서 제1 프로세싱 노드(102A)의 리던던트 디스크(106)에 저장된 제1 리던던트 데이터를 업데이트한다.

[0042] 단계(216)에서, 제1 프로세싱 노드 (102A)는 델타 데이터를 적어도 하나의 리던던트 프로세싱 노드(102), 이를테면, 제1 프로세싱 노드(102A)와 상이한 제2 프로세싱 노드(102B)에게 송신한다. 도 3을 참조하면, 단계(302)에서 제2 프로세싱 노드(102B)는 델타 데이터를 수신하며, 단계(304)에서 델타 데이터를 제2 프로세싱 노드(102B)의 디스크 캐시에 저장한다. 일단 델타 데이터가 디스크 캐시에 저장되면, 단계(306)에서, 제2 프로세싱 노드(102B)는 제1 프로세싱 노드(102A)에 델타 데이터 기록의 완료를 확인응답하도록 구성된다. 이때, 제1 프로세싱 노드(102A)가 장애가 있었다면, 제2 프로세싱 노드는 제1 프로세싱 노드(102A)의 호스트(104A)에 의한 선택된 LBA에 기록된 데이터의 복구에 참여할 수 있다. 단계(218)에서, 모든 리던던트 노드(102)가 델타 데이터 기록의 완료를 확인응답하였는지 판단된다. 단계(220)에서, 호스트 데이터 기록의 완료가 제1 프로세싱 노드(102A)의 호스트(104A)에게 확인응답된다.

[0043] 단계(308)에서, 델타 데이터를 제2 프로세싱 노드(102B)의 디스크 캐시에 저장한 후, 델타 데이터는 선택적으로 디스크 캐시로부터 플러시(flush)된다. 디스크 캐시가 작은 경우의 일부 실시예에서, 플러시할 때를 판단하는 데 최소 최근(least-recently) 사용된 알고리즘과 같은 알고리즘을 사용하는 큰 디스크 캐시를 갖는 다른 실시예와 비교하여 볼 때, 단계(308)는 비교적 신속하게 수행된다. 단계(310)에서, 델타 데이터를 디스크 캐시로부터 플러시하도록 하는 플러싱 또는 그렇게 플러시하도록 하는 결정에 응답하여, 델타 데이터에 대응하는 인트라-노드 보호 데이터를 저장하는 제2 프로세싱 노드(102B)의 하나 이상의 인터-노드 리던던시 디스크(106) 및 인터-노드 리던던시 디스크(106)를 위한 인트라-노드 보호 데이터를 저장하는 제2 프로세싱 노드(102B)의 하나 이상의 리던던트 디스크(106)는 제2 프로세싱 노드(102B)의 호스트(104B) 및 컨트롤러(108B) 중 적어도 하나에 의해 결정된다.

[0044] 단계(312)에서, 제2 프로세싱 노드(102B)의 인터-노드 리던던시 계산 유닛(114B)은 델타 데이터에 따라서 제2 프로세싱 노드(102B)의 인터-노드 리던던시 디스크(106)에 저장된 글로벌 리던던트 데이터를 업데이트 한다. 단계(314)에서, 제2 프로세싱 노드(102B)의 인트라-노드 리던던시 계산 유닛(110B)은 인터-노드 노드 리던던시 디스크(106)의 업데이트에 따라서 제2 프로세싱 노드(102B)의 리던던트 디스크(106)에 저장된 제2 로컬 리던던트 데이터를 업데이트한다. 단계(316)에서, 델타 데이터는 제2 프로세싱 노드(102B)의 디스크 캐시로부터 제거된다. 디스크 캐시가 휘발성 캐시인 경우의 일부 실시예에서, 단계(306)는 델타 데이터가 비휘발성 데이터로 저장되는 것을 보장하기 위해 단계(312) 및/또는 단계(314) 중 하나 이상의 단계 이후까지 지연된다.

[0045] 일부 실시예에서, 글로벌 리던던트 데이터의 계산을 위해 노드들(102) 사이에서 운송된 델타 데이터는 (호스트(104)에 의한 데이터의 기록 이전의) 구 데이터와 호스트(104)에 의해 기록된 신규 데이터와의 함수이다. 일부 실시예에서, 델타 데이터는 구 데이터 및 호스트(104)에 의해 기록된 신규 데이터와의 XOR 함수 또는 XNOR 함수를 이용하여 결정된다. 다른 실시예에서, 델타 데이터는 구 데이터 및 신규 데이터를 포함하며, 구 데이터 및 신규 데이터는 둘 다 노드들(102) 사이에서 운송된다. 일부 실시예에서, 델타 데이터는, 어느 노드가 델타 데이터를 생성하였는지와, 델타 데이터가 생성되게 만든 기록의 인터-노드 보호 코드워드 내 위치와, 델타 데이터의 원위치 및/또는 위치와 연관된 다른 정보 중의 적어도 하나를 더 포함한다.

[0046] 일부 실시예에서, 인터-노드 리던던시 계산은 글로벌 리던던트 데이터의 일부를 저장하는 하나 이상의 노드(102)의 각각에서 독립적으로 수행된다. 예를 들면, 2-소거-정정 리드 솔로몬 코드를 이용하는 RAID-6 코딩 타입의 경우, 델타 데이터는 글로벌 리던던트 데이터의 일부를 저장하는 두 프로세싱 노드(102)의 각각으로 송신되며, 두 프로세싱 노드(102)는 각기 독립적으로 글로벌 리던던트 데이터의 일부를 업데이트한다. 2-소거-정정 리드 솔로몬 코드의 경우, 리드 솔로몬 코드의 코드워드 내 델타 데이터의 위치는 델타 데이터와 함께

송신되며, 두 프로세싱 노드(102)는 각기 코드워드 내 델타 데이터의 위치에 있는 데이터가 리드 솔로몬 코드의 생성 다항식에 의해 제산될 때 획득된 나머지 부분에 해당하는 업데이터를 결정함으로써 글로벌 리던던트 데이터의 일부에 대한 업데이트를 독립적으로 계산하도록 구성된다.

[0047] 일부 실시예에서, 델타 데이터는 글로벌 리던던트 데이터의 계산을 위해 노드들 중 다른 노드에 수송되기 전에 감소 및/또는 국부적으로 조합된다. 제1의 일예에서, 프로세싱 노드들 중 제1 프로세싱 노드의 호스트에 의한 데이터의 제1 기록 및 제1 프로세싱 노드의 호스트에 의한 데이터의 제2 기록은 동일한 LBA에 대해 수행되며, 제1 기록 및 제2 기록의 둘 다에 해당하는 단일의 델타 데이터가 수송된다. 예를 들면, 함수가 XOR인 경우, 델타 데이터는 제2 기록의 제2 (최종) 데이터와 XOR 계산된 (제1 기록 이전의) 구 데이터에 해당한다. 제2의 일예에서, 인터-노드 보호 메커니즘의 코드워드는 프로세싱 노드들 중 제1 프로세싱 노드의 둘 이상의 디스크에 저장된 것에 해당하며, 둘 이상의 디스크 중 하나보다 많은 디스크에 기록은 그 기록에 대응하는 단일의 델타 데이터가 생성되게 한다. 인터-노드 보호 메커니즘의 코딩 타입 및 특정한 신뢰도에 따라, 델타 데이터의 크기는 둘 이상의 디스크 중 단지 하나의 디스크에 기록의 크기와 같다.

[0048] 일부 실시예에서, 제1 프로세싱 노드(102A)의 호스트 데이터 기록은 복수의 상이한 델타 데이터를 생성하며, 각각의 델타 데이터는 글로벌 리던던트 데이터의 일부를 저장하는 대응하는 프로세싱 노드(102)에 송신된다. 다른 실시예에서, 제1 프로세싱 노드(102A)의 호스트 데이터 기록은 글로벌 리던던트 데이터의 일부를 저장하는 하나 이상의 프로세싱 노드(102)에 송신된 단일의 델타 데이터를 생성한다.

[0049] 일부 실시예에서, 제1 프로세싱 노드(102A)의 호스트 데이터 기록은 제1 프로세싱 노드(102A)와 상이한 제2 프로세싱 노드(102B)의 디스크들(106) 중 한 디스크에 대해 수행(즉, "외부(foreign)" 데이터 기록)된다. 시스템(100)의 디스크(106)에 대해, 포린 기록은 로컬 기록과 유사하게 수행된다. 그러나, 포린 기록의 데이터는 제1 프로세싱 노드(102A)에서 국부적으로 머물기보다는 제2 프로세싱 노드(102B)에 수송된다. 일부 실시예에서, 또 다른 차이는 제2 프로세싱 노드(102B)가 포린 기록으로 인한 어느 인터-노드 리던던트 기록의 완료를 판단한 이후 포린 기록의 완료의 확인응답이 제2 프로세싱 노드(102B)에 의해 제1 프로세싱 노드(102A)에 리턴된다는 것이다.

[0050] 일부 실시예에서, 프로세싱 노드(102) 중 적어도 일부는 캐시처럼 사용된 고체 상태 디스크와 같은 디스크 캐시를 포함한다. 디스크 캐시는, 프로세싱 노드(102)의 호스트(104)에 의해 액세스된 데이터(예를 들면, 스토리지)와, 다른 프로세싱 노드(102)의 호스트(104)에 의해 액세스된 데이터와, 프로세싱 노드(102)의 로컬 리던던트 데이터와, 프로세싱 노드(102)의 디스크(106)에 저장된 글로벌 리던던트 데이터와, 프로세싱 노드(102)에 의해 계산된 및/또는 다른 프로세싱 노드(102)로부터 수신된 델타 데이터, 및 다른 형식의 데이터 중 하나 이상을 저장한다. 일부 실시예에서, 다른 프로세싱 노드(102)로부터 수신된 델타 데이터를 디스크 캐시에 저장하는 것은 델타 데이터의 안전의 확인응답이 가능해 진 것이며, 그래서 대응하는 호스트 데이터 기록의 안전의 확인응답이 가능해지며, 그런 다음 글로벌 리던던트 데이터 및/또는 그 글로벌 리던던트 데이터를 보호하는 제2 로컬 리던던트 데이터를 업데이트할 수 있다.

[0051] 일부 실시예에서, 프로세싱 노드(102)의 디스크 캐시는 프로세싱 노드(102)의 호스트(104)와, 프로세싱 노드의 ROC와 같은 I/O 컨트롤러(108)와, 전용의 관리 프로세서, 및 전술한 것들의 어떤 조합 중 하나 이상에 의해 관리된다.

[0052] 일부 실시예에서, 디스크 캐시는 다른 형태의 데이터와 상이하게 델타 데이터를 태그(tag)한다. 일부 실시예에서, 델타 데이터는 더티(dirty)인 것으로 및 논-델타 데이터 더티 데이터(non-delta dirty data) 처럼 직접 저장될 수 있는 것과 반대인 델타 포맷으로서 태그된다. 델타 데이터를 제1 프로세싱 노드(102A)의 디스크 캐시로부터 플러시하기 위해, 제1 프로세싱 노드(102A)의 인터-노드 리던던시 계산 유닛(114A)은 델타 데이터에 따라서 제1 프로세싱 노드(102A)의 디스크(106)에 저장된 글로벌 리던던트 데이터를 업데이트한 다음, 델타 데이터가 디스크 캐시로부터 삭제 또는 제거되도록 구성된다. 일부 실시예에서, 제1 프로세싱 노드(102A)의 디스크(106)에 저장된 글로벌 리던던트 데이터를 업데이트하는 것은 글로벌 리던던트 데이터를 보호하는 인트라-노드 리던던트 데이터를 업데이트하는 것을 포함한다. 제1 프로세싱 노드(102A)의 다른 디스크에 저장된 인트라-노드 리던던트 데이터는 제1 프로세싱 노드(102A)의 인트라-노드 리던던시 계산 유닛(110A)을 통해 업데이트된다.

[0053] 델타 데이터가 프로세싱 노드(102)의 디스크 캐시에 저장되는 일부 실시예에서, 델타 데이터를 수신하는 프로세싱 노드(102)는 델타 데이터를 디스크 캐시에 저장하기 전에 델타 데이터에 대해 적어도 일부의 인터-노드 리던던시 계산을 수행하고 변환된 버전의 델타 데이터를 디스크 캐시에 저장한다. 예를 들면, 다중-소거-정정 코드의 경우, 수신된 델타 데이터는 프로세싱 노드(102)에 저장된 글로벌 리던던트 데이터에 바로 조합될 수 있는

형태의 것은 아니다. 수신된 델타 데이터를 인터-노드 리던던시 계산 유닛(114)을 이용하여 변환함으로써, 변환된 델타 데이터가 XOR 함수와 같은 간단한 연산을 통해 나중에 글로벌 리던던트 데이터와 조합될 수 있다. 일부 실시예에서, 또한 변환된 버전의 델타 데이터를 디스크 캐시에 저장하면 나중에 수신한 델타 데이터를 변환된 버전의 델타 데이터에 조합하는 것이 가능해지며, 그럼으로써 유익하게 디스크 캐시의 공간을 절감할 수 있다. 예를 들면, 인터-노드 보호 코딩 타입과 같은 리드 솔로몬 코드의 경우, 델타 데이터는 리드 솔로몬 코드의 생성 다항식에 따라서 글로벌 리던던트 데이터로서 저장된 코드워드 나머지의 일부에 대한 업데이트로(XOR를 통해) 변환된다.

[0054] 일부 실시예에서, 캐시된 델타 데이터는 업데이트되거나 디스크 캐시에서 조합된다. 예를 들면, 제1 프로세싱 노드(102A)의 호스트(103A)에 의한 선택된 논리 블록 어드레스(LBA)에서 제1 기록에 대응하는 제1 델타 데이터는 제2 프로세싱 노드(102B)의 디스크 캐시에 저장되며, 선택된 LBA에서 제2 기록에 대응하는 제2 델타 데이터는 제2 프로세싱 노드(102B)에 의해 수신된다. 제2 프로세싱 노드(102B)의 디스크 캐시는 제2 프로세싱 노드(102B)의 디스크(106)에 저장된 글로벌 리던던트 데이터의 한번의 업데이트만이 제1 기록 및 제2 기록의 양쪽에 필요하도록 제2 델타 데이터에 따라서 제1 델타 데이터를 업데이트하도록 구성된다. 예를 들어, 만일 델타 데이터가 XOR 함수를 이용하여 제1 프로세싱 노드(102A)에서 계산되면, 제1 델타 데이터는 제2 델타 데이터와의 XOR 연산에 의해 업데이트된다.

[0055] 일부 실시예에서, 제1 프로세싱 노드(102A)의 호스트(104A)에 의한 인터-노드 보호 코드워드에 의해 보호된 데이터에의 제1 기록에 대응하는 제1 델타 데이터는 제2 프로세싱 노드(102B)의 디스크 캐시에 저장되며, 인터-노드 보호 코드워드에 의해 보호된 데이터에의 제2 기록에 대응하는 제2 델타 데이터는 제2 프로세싱 노드(102B)에 의해 수신된다. 제2 프로세싱 노드(102B)의 디스크 캐시는 제1 기록 및 제2 기록 둘 다에게 제2 프로세싱 노드(102B)의 디스크(106)에 저장된 글로벌 리던던트 데이터의 한번의 업데이트가 필요하도록 제2 델타 데이터에 따라서 제1 델타 데이터를 업데이트하도록 구성된다.

[0056] 일부 실시예에서, 로컬 리던던트 데이터는 결정된 방식, 이를 테면, CRUSH(Controlled Replication Under Scalable Hashing) 알고리즘 또는 다른 데이터 분배 알고리즘에 따라 프로세싱 노드(102)의 디스크들(106) 사이에 분배된다. 일부 실시예에서, 글로벌 리던던트 데이터는 CRUSH 알고리즘 또는 다른 데이터 분배 알고리즘과 같은 결정된 방식대로 프로세싱 노드들(102) 중 둘 이상의 노드의 디스크(106) 사이에서 분배된다. 예를 들면, 제1의 인터-노드 보호 코드워드는 프로세싱 노드(102)의 제1 서브셋 상의 디스크들(106)에 퍼져 있으며, 제2의 인터-노드 보호 코드워드는 제1 서브셋과 상이한 프로세싱 노드(102)의 제2 서브셋 상의 디스크들(106)에 퍼져 있다. 일부 실시예에서, 제1 서브셋과 제2 서브셋은 중첩한다(예를 들면, 적어도 하나의 프로세싱 노드(102)를 공통으로 포함한다).

[0057] 일부 실시예에서, 인트라-노드 리던던시 계산 유닛(110)은 하나 이상의 디스크(106)의 일부이며/이거나 그 하나 이상의 디스크에 통합된다. 예를 들면, 몇몇 SSD는 SSD의 비휘발성 메모리 칩에 저장된 데이터를 보호하는 RAID-5형 또는 RAID-6형 리던던시 메커니즘을 구현한다. SSD의 리던던시 메커니즘은 SSD에 저장된 데이터를 위한 인트라-노드 리던던시 계산 유닛(110)으로서 기능할 수 있다.

[0058] 일부 실시예에서, 프로세싱 노드들(102)은 실질적으로 동일 또는 유사하게 구성될 수 있다. 다른 실시예에서, 프로세싱 노드들(102)은 모두 호스트(들)(104)의 개수 및/또는 구성, 로컬 메모리의 양, 디스크의 개수, 구성, 형태, 및/또는 용량, 또는 어느 다른 파라미터(들), 컴포넌트(들), 또는 구성(들) 면에서 모두 대칭적이지 않다.

[0059] 일부 실시예에서, 프로세싱 노드(102) 중 적어도 일부는 제한된 또는 프로세싱 능력이 없으며, 사실상 "디스크-전용(disk-only)"이다. 디스크-전용 프로세싱 노드(102)는 글로벌 리던던시의 일부를 저장하는 것과 같은 글로벌 리던던시 계산에 참여한다. 일부 실시예에서, 프로세싱 노드(102) 중 하나의 노드는 디스크-전용 프로세싱 노드(102)의 적어도 일부 스토리지가 여전히 전역적으로 액세스 가능하다면, 각자의 호스트(104)의 충돌로 인해 디스크-전용이 된다. 따라서, 다른 프로세싱 노드(102)로부터 디스크-전용 프로세싱 노드(102)의 스토리지로의 포린 기록은 디스크-전용 프로세싱 노드(102)의 컨트롤러(108)(예를 들면, ROC)가 하는 것처럼 여전히 델타 데이터를 생성되게 할 수 있고 전송되게 할 수 있다.

[0060] 일부 실시예에서, 복수의 인트라-노드 보호 메커니즘(110) 및/또는 인터-노드 보호 메커니즘(114)은, 보호되는 디스크(106)의 형태 및/또는 신뢰도와, 보호되는 디스크(106)에 저장된 데이터의 형태와, 선택된 인터-노드 보호 메커니즘(114)에 의해 보호된 노드(102)의 장애 확률, 및 기타 요인들 중 하나 이상에 따라서 사용된다.

[0061] 일부 실시예에서 본 개시내용 전체에서 기술된 여러 단계들은 단일의 컴퓨팅 시스템 또는 다중 컴퓨팅 시스템에 의해 수행될 수 있다는 것을 인식하여야 한다. 컴퓨팅 시스템은 퍼스널 컴퓨팅 시스템, 메인프레임 컴퓨팅 시스템, 워크스테이션, 이미지 컴퓨터, 병렬 프로세서, 또는 본 기술에서 공지된 어느 다른 장치를 포함할 수 있지만, 이것으로 제한되지 않는다. 일반적으로, "컴퓨팅 시스템"이라는 용어는 메모리 매체로부터의 명령어를 실행하는 하나 이상의 프로세서를 갖는 모든 장치를 망라하는 것으로 넓게 정의된다.

[0062] 본 명세서에서 기술된 실시예들에 의해 명시된 바와 같은 방법을 구현하는 프로그램 명령어는 캐리어 매체를 통해 전송 또는 캐리어 매체에 저장될 수 있다. 캐리어 매체는 전송 매체, 이를 테면, 와이어, 케이블 또는 와이어리스 전송 링크일 수 있지만, 이것으로 제한되지 않는다. 캐리어 매체는 또한 스토리지 매체, 이를 테면, 리드-온리 메모리, 랜덤 액세스 메모리, 자기 또는 광학 디스크, 또는 자기 테이프를 포함하지만, 이것으로 제한되지 않는다.

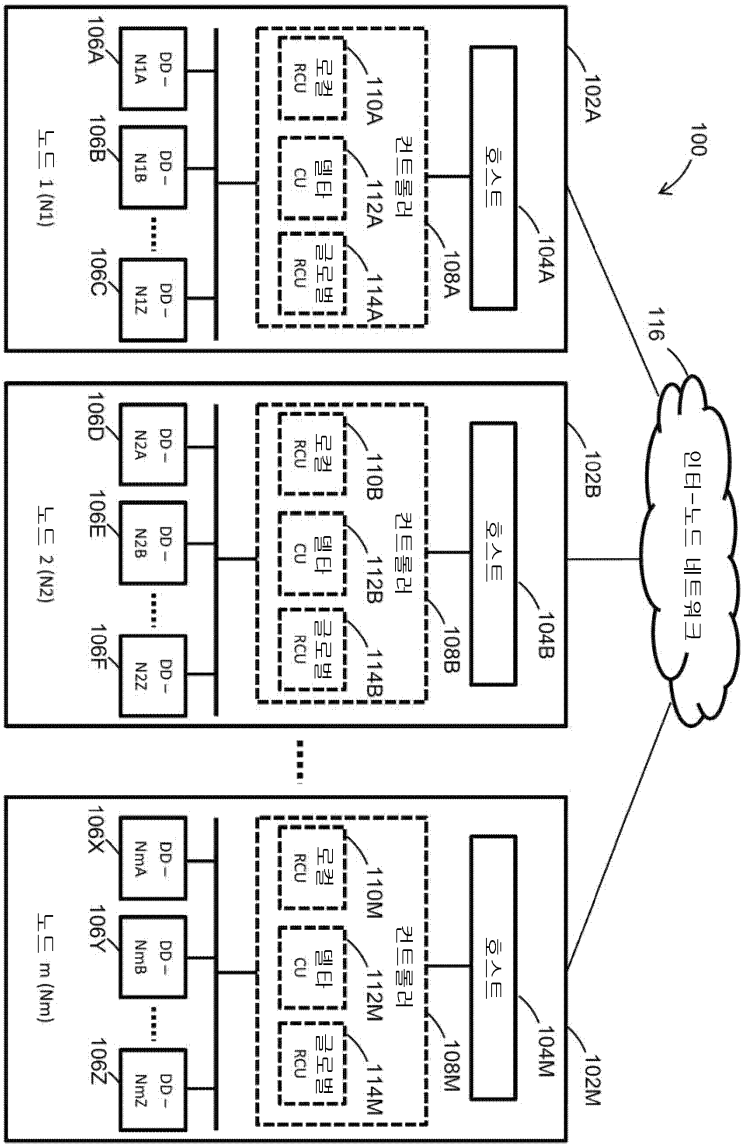
[0063] 본 명세서에서 기술된 방법을 명시하는 실시예는 스토리지 매체에 저장하는 결과물을 포함할 수 있다. 그 결과물이 저장된 후, 그 결과물은 스토리지 메체에서 액세스가능하며 본 명세서에서 기술된 방법 또는 시스템 실시예의 어느 것에 의해 사용되며, 사용자에게 디스플레이하기 위해 포맷되고, 다른 소프트웨어 모듈, 방법 또는 시스템 등에 의해 사용된다. 더욱이, 그 결과물은 "영구적으로", "반영구적으로", "일시적으로", 또는 당분간 동안 저장될 수 있다. 예를 들면, 스토리지 매체는 랜덤 액세스 메모리(RAM)일 수 있으며, 결과물은 반드시 스토리지 매체 내에 무기한으로 지속되어 있지 않을 수 있다.

[0064] 시스템 또는 방법으로서 상기 명시된 개시 내용의 어느 실시예도 본 명세서에서 기술된 어느 다른 실시예의 적어도 일부를 포함할 수 있음을 또한 주목하여야 한다. 본 기술에서 통상의 지식을 가진 자들이라면 본 명세서에서 기술된 시스템 및 방법에 영향을 미칠 수 있는 여러 실시예가 있으며, 그 구현은 개시 내용의 실시예가 배치된 상황에 따라 변할 수 있음을 인식할 것이다.

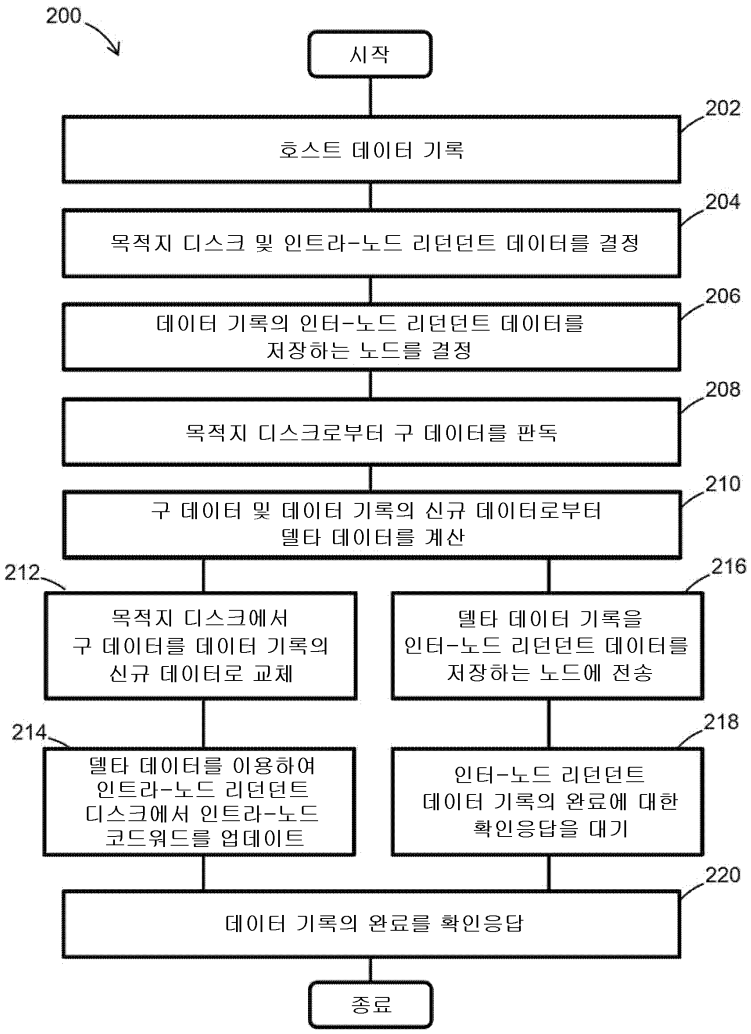
[0065] 더욱이, 본 발명은 첨부의 특허청구범위에 의해 정의됨은 물론이다. 비록 본 발명의 실시예가 예시되었을지라도, 본 기술에서 통상의 지식을 가진 자들에 의해 개시 내용의 범주 및 정신을 일탈함이 없이 여러 변형 예가 이루어질 수 있음은 자명하다.

도면

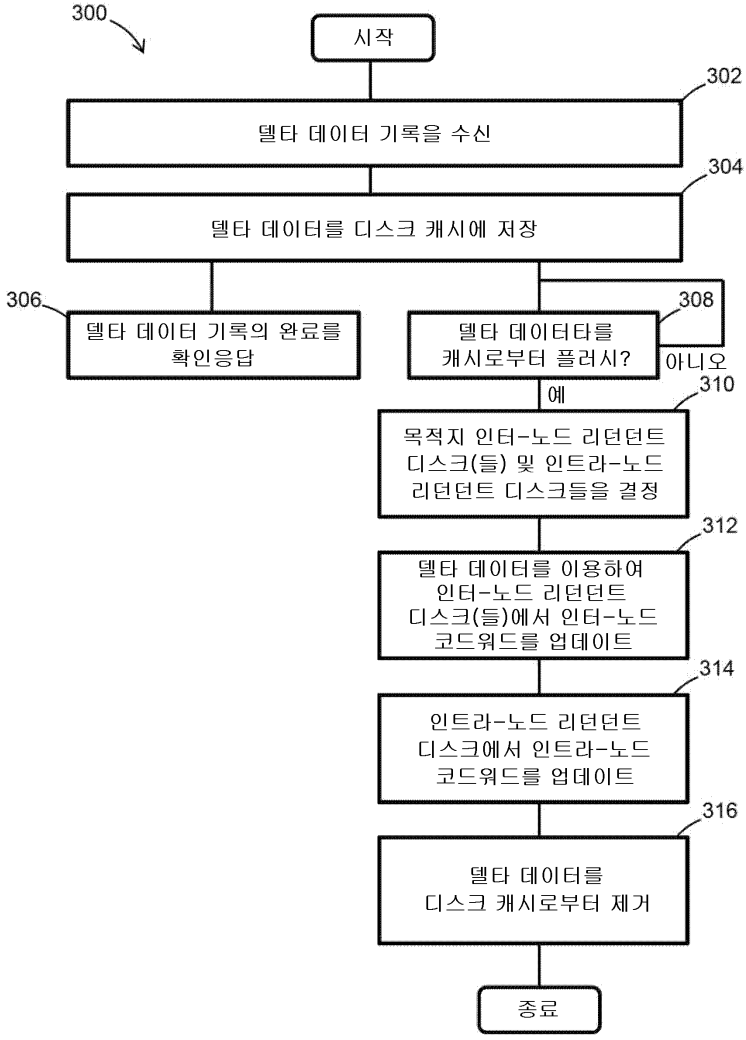
도면1


100
116
인터-노드 네트워크
102A
호스트
104A
108A
컨트롤러
로컬
RCU
110A
델타
CU
112A
글로벌
RCU
114A
노드 1 (N1)
106A
DD-N1A
106B
DD-N1B
106C
DD-N1Z
102B
호스트
104B
108B
컨트롤러
로컬
RCU
110B
델타
CU
112B
글로벌
RCU
114B
노드 2 (N2)
106D
DD-N2A
106E
DD-N2B
106F
DD-N2Z
102M
호스트
104M
108M
컨트롤러
로컬
RCU
110M
델타
CU
112M
글로벌
RCU
114M
노드 m (Nm)
106X
DD-NmA
106Y
DD-NmB
106Z
DD-NmZ

도면2


200
시작
202
호스트 데이터 기록
204
목적지 디스크 및 인트라-노드 리던던트 데이터를 결정
206
데이터 기록의 인터-노드 리던던트 데이터를 저장하는 노드를 결정
208
목적지 디스크로부터 구 데이터를 판독
210
구 데이터 및 데이터 기록의 신규 데이터로부터 델타 데이터를 계산
212
목적지 디스크에서 구 데이터를 데이터 기록의 신규 데이터로 교체
214
델타 데이터를 이용하여 인트라-노드 리던던트 디스크에서 인트라-노드 코드워드를 업데이트
216
델타 데이터 기록을 인터-노드 리던던트 데이터를 저장하는 노드에 전송
218
인터-노드 리던던트 데이터 기록의 완료에 대한 확인응답을 대기
220
데이터 기록의 완료를 확인응답
종료

도면3


300
시작
302
델타 데이터 기록을 수신
304
델타 데이터를 디스크 캐시에 저장
306
델타 데이터 기록의 완료를 확인응답
308
델타 데이터타를 캐시로부터 플러시?
아니오
예
310
목적지 인터-노드 리던던트 디스크(들) 및 인트라-노드 리던던트 디스크들을 결정
312
델타 데이터를 이용하여 인터-노드 리던던트 디스크(들)에서 인터-노드 코드워드를 업데이트
314
인트라-노드 리던던트 디스크에서 인트라-노드 코드워드를 업데이트
316
델타 데이터를 디스크 캐시로부터 제거
종료