



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2012년05월18일  
(11) 등록번호 10-1143241  
(24) 등록일자 2012년04월27일

(51) 국제특허분류(Int. Cl.)  
*G06Q 50/00* (2006.01)  
(21) 출원번호 10-2005-0102989  
(22) 출원일자 2005년10월31일  
심사청구일자 2010년10월25일  
(65) 공개번호 10-2006-0070413  
(43) 공개일자 2006년06월23일  
(30) 우선권주장  
11/018,914 2004년12월20일 미국(US)  
(56) 선행기술조사문헌  
JP2003123382 A\*  
US06567826 B1  
\*는 심사관에 의하여 인용된 문헌

(73) 특허권자  
마이크로소프트 코포레이션  
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원  
마이크로소프트 웨이  
(72) 발명자  
존스, 브라이언 엠.  
미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이마이크로소프트 코포레이션 내  
리우, 캐롤 엘.  
미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이마이크로소프트 코포레이션 내  
(뒷면에 계속)  
(74) 대리인  
제일특허법인

전체 청구항 수 : 총 15 항

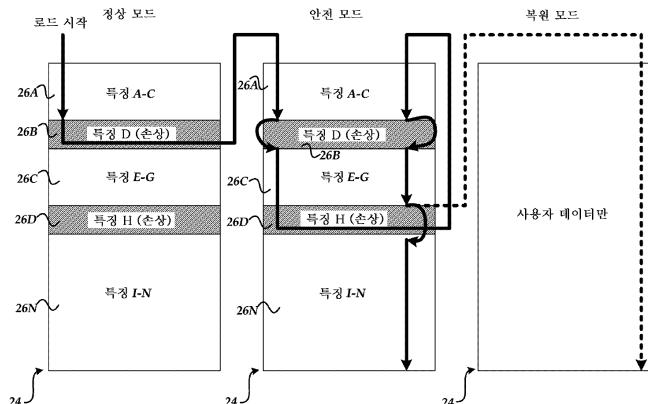
심사관 : 김석호

(54) 발명의 명칭 데이터 파일의 콘텐츠를 로딩하기 위한 방법 및 컴퓨터판독가능 매체

### (57) 요약

전자 데이터 파일의 콘텐츠를 로딩하기 위한 방법 및 컴퓨터 판독가능 매체를 제공한다. 본 방법에 따르면, 데이터 파일의 손상된 부분을 식별하고, 이러한 부분의 복구(repair)를 시도한다. 손상된 부분을 수리할 수 없는 경우, 이러한 부분의 로딩은 스kip(skip)한다. 데이터 파일의 손상되지 않은 부분 및 수리된 부분은 메모리에 로드(load)한다. 데이터 파일의 부분을 수리 또는 스kip 할 수 없는 경우, 데이터 파일에 포함된 사용자 데이터만을 로드하려고 시도한다. 이러한 방식으로, 데이터 파일에 포함된 사용자 데이터는, 심한 손상이 발생하는 경우에도 로드할 수도 있다.

### 대 표 도



(72) 발명자

로쓰뮐러, 채드 비.

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이마이크로소프트 코포레이션 내

맥코헤이, 로버트 알.

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이마이크로소프트 코포레이션 내

---

빌래론 손 에이.

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이마이크로소프트 코포레이션 내

우, 수-피아오 빌

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이마이크로소프트 코포레이션 내

## 특허청구의 범위

### 청구항 1

적어도 하나의 부분(portion)을 포함하는 데이터 파일을 로딩하기 위한, 컴퓨터에 의하여 수행되는 방법으로서, 상기 방법은,

제1 로딩 모드에서 상기 데이터 파일의 상기 적어도 하나의 부분의 로드를 시도하는 단계 - 상기 적어도 하나의 부분의 로드를 시도하는 단계는,

스킵될 수 있는 상기 적어도 하나의 부분 각각을 식별하는 스kip 카운터(a skip counter)를 초기화하는 단계, 및

상기 적어도 하나의 부분 각각에 대하여 무결성 검사(integrity checking)를 수행하는 단계를 포함함 -,

상기 제1 로딩 모드에서 상기 적어도 하나의 부분 중 어떤 부분이 로드 불가능한지 여부를 판정하는 단계 - 상기 제1 로딩 모드에서 상기 적어도 하나의 부분 중 어떤 부분이 로드 불가능한지 여부를 판정하는 단계는, 현재 모드가 정상 모드(normal mode) 및 안전 모드(safe mode) 중 적어도 하나에 해당하는지 여부를 판정하는 단계를 포함함 -, 및

상기 적어도 하나의 부분 중 어떤 부분이 로드 불가능하다는 판정에 응답하여,

로드 불가능한 부분이 스kip될 수 있음을 나타내도록 상기 스kip 카운터에 플래그(flag)하는 단계,

제2 로딩 모드로 전환하는 단계, 및

상기 제2 로딩 모드에서 상기 데이터 파일의 로드를 시도하는 단계 - 추가적인 무결성 검사가 상기 적어도 하나의 부분 각각에 대하여 수행되고, 로드 불가능한 부분의 로딩 및 상기 로드 불가능한 부분에 관련된 부분의 로딩이 스kip되고, 상기 제2 로딩 모드에서 상기 데이터 파일의 로드를 시도하는 동안에 상기 로드 불가능한 부분이 복구(repair)될 수 있는지 여부를 판정하고, 상기 로드 불가능한 부분이 복구될 수 있다는 판정에 응답하여 상기 로드 불가능한 부분을 복구하고, 상기 스kip 카운터의 플래그를 해제(unflag)하며, 복구된 부분을 로드함 -

를 포함하는, 데이터 파일 로딩 방법.

### 청구항 2

제1항에 있어서,

로드 불가능한 부분이 복구될 수 없다는 판정에 응답하여, 상기 로드 불가능한 부분의 로딩을 스kip하는 단계를 더 포함하는, 데이터 파일 로딩 방법.

### 청구항 3

제2항에 있어서,

상기 로드 불가능한 부분의 로딩이 스kip될 수 있는지 여부를 판정하는 단계, 및

상기 로드 불가능한 부분의 로딩이 스kip될 수 없다는 판정에 응답하여, 제3 로딩 모드에서 상기 데이터 파일의 로드를 시도하는 단계 - 상기 제3 로딩 모드에서는 사용자 데이터에 대응하는 상기 데이터 파일의 부분만을 로드함 -

를 더 포함하는, 데이터 파일 로딩 방법.

### 청구항 4

제3항에 있어서,

상기 사용자 데이터는 사용자에 의하여 입력된 텍스트 데이터를 포함하는, 데이터 파일 로딩 방법.

**청구항 5**

제3항에 있어서,

상기 사용자 데이터는 사용자에 의하여 입력된 수치 데이터(numerical data)를 포함하는, 데이터 파일 로딩 방법.

**청구항 6**

컴퓨터 실행가능 명령어를 저장한 컴퓨터 판독가능 기록매체로서,

상기 컴퓨터 실행가능 명령어가 컴퓨터에 의하여 실행될 때 상기 컴퓨터는,

하나 이상의 부분을 포함하는 데이터 파일을 로딩하기 위한 제1 로딩 모드를 제공 - 상기 제1 로딩 모드를 제공하는 것은 스kip될 수 있는 상기 적어도 하나의 부분 각각을 식별하는 스kip 카운터를 초기화하고, 상기 제1 로딩 모드에서 로딩하는 경우 상기 적어도 하나의 부분 각각에 대해 무결성 검사를 수행함 - 하고,

상기 데이터 파일의 상기 적어도 하나의 부분을 로딩하기 위한 제2 로딩 모드를 제공 - 상기 적어도 하나의 부분 각각에 대해 추가적인 무결성 검사를 수행하고, 로드 불가능한 부분의 로딩 및 상기 로드 불가능한 부분에 관련된 부분의 로딩을 스kip함 - 하고,

상기 제1 로딩 모드에서 상기 데이터 파일의 로딩을 시작 - 상기 데이터 파일을 로딩하는 동안 현재 모드가 정상 모드 및 안전 모드 중 적어도 하나에 해당하는지 여부를 판정함 - 하고,

상기 제1 로딩 모드에서 동작하는 경우, 상기 데이터 파일의 어떤 부분이 로드 불가능한지 여부를 판정하고, 상기 어떤 부분이 로드 불가능하다는 판정에 응답하여, 로드 불가능한 부분이 스kip될 수 있음을 나타내도록 상기 스kip 카운터에 플래그하고 상기 제2 로딩 모드로 전환하며,

상기 제2 로딩 모드에서 상기 데이터 파일의 로드를 시도 - 상기 제2 로딩 모드에서 상기 데이터 파일의 로드를 시도하는 동안에 상기 로드 불가능한 부분이 복구될 수 있는지 여부를 판정하고, 상기 로드 불가능한 부분이 복구될 수 있다는 판정에 응답하여 상기 로드 불가능한 부분을 복구하고, 상기 스kip 카운터의 플래그를 해제하며, 복구된 부분을 로드함 - 하는

컴퓨터 판독가능 기록매체.

**청구항 7**

제6항에 있어서,

상기 제2 로딩 모드에서 로드 불가능한 부분이 복구될 수 없다는 판정에 응답하여, 상기 로드 불가능한 부분의 로딩을 스kip하는, 컴퓨터 판독가능 기록매체.

**청구항 8**

제7항에 있어서,

상기 컴퓨터에 의하여 실행될 때 상기 컴퓨터로 하여금,

제3 로딩 모드를 제공 - 상기 제3 로딩 모드에서 사용자 데이터에 대응하는 상기 데이터 파일의 부분만을 로드함 - 하게 하고,

상기 제2 로딩 모드에서 동작하는 동안, 상기 로드 불가능한 부분의 로딩이 스kip될 수 있는지 여부를 판정하게 하며,

상기 로드 불가능한 부분의 로딩이 스kip될 수 없다는 판정에 응답하여, 상기 제3 로딩 모드로 전환하게 하는

컴퓨터 실행가능 명령어를 더 포함하는, 컴퓨터 판독가능 기록매체.

**청구항 9**

제8항에 있어서,

상기 사용자 데이터는 사용자에 의하여 입력된 텍스트 데이터를 포함하는, 컴퓨터 판독가능 기록매체.

**청구항 10**

제8항에 있어서,

상기 사용자 데이터는 사용자에 의하여 입력된 수치 데이터를 포함하는, 컴퓨터 관독가능 기록매체.

**청구항 11**

적어도 하나의 레코드(record)을 포함하는 데이터 파일을 로딩하기 위한, 컴퓨터에 의하여 수행되는 방법으로서,

상기 방법은,

사용자에 의한 입력에 기초하여 로드 모드(a load mode)를 선택하는 단계;

레코드가 스kip될 수 없음을 나타내는 스kip 카운터를 초기화하는 단계;

스kip될 레코드 개수를 나타내는, 스kip될 레코드 개수 카운터를 초기화하는 단계;

제1 모드에서 상기 데이터 파일을 로딩하는 단계 - 상기 데이터 파일을 로딩하는 단계는 상기 적어도 하나의 레코드의 각 레코드에 대하여 무결성 검사를 수행하는 단계를 포함함 - ;

상기 제1 모드에서 상기 데이터 파일의 레코드가 로드 불가능한지 여부를 판정하는 단계;

상기 데이터 파일의 상기 레코드가 로드 불가능하다는 판정에 응답하여, 상기 제1 모드에서 현재 모드가 정상 모드인지 여부를 판정하는 단계 - 로드 불가능한 레코드가 스kip될 수 있음을 나타내도록 상기 스kip 카운터에 플래그하고, 상기 현재 모드를 안전 모드로 전환하고, 상기 데이터 파일을 처음부터 로딩함 - ;

상기 현재 모드가 상기 정상 모드가 아니라는 상기 제1 모드에서의 판정에 응답하여, 상기 현재 모드가 상기 안전 모드인지 여부를 판정하는 단계 - 상기 현재 모드가 상기 안전 모드가 아닌 경우에는 에러를 리턴함 - ;

제2 모드에서 상기 데이터 파일의 로드를 시도하는 단계 - 상기 제2 모드에서 상기 데이터 파일의 로드를 시도하는 동안 상기 로드 불가능한 레코드를 복구하는지 여부를 판정함 - ;

상기 로드 불가능한 레코드를 복구하는지 여부에 대한 판정에 응답하여, 상기 스kip 카운터의 플래그를 해제하고, 상기 레코드가 복구된 경우 상기 레코드를 로딩하고, 상기 레코드가 복구되지 않은 경우 상기 스kip될 레코드 개수를 판정하는 단계; 및

상기 스kip될 레코드의 개수에 대한 판정에 응답하여, 상기 스kip될 레코드 개수가 0과 일치하면 상기 스kip 카운터에 플래그하고, 상기 스kip될 레코드 개수가 0과 일치하지 않으면 상기 레코드의 로딩을 스kip을 시도하는 단계를 포함하는, 데이터 파일 로딩 방법.

**청구항 12**

제11항에 있어서,

로드 불가능한 레코드가 복구될 수 없다는 판정에 응답하여, 상기 로드 불가능한 레코드의 로딩을 스kip하는 단계를 더 포함하는, 데이터 파일 로딩 방법.

**청구항 13**

제12항에 있어서,

상기 로드 불가능한 레코드의 로딩이 스kip될 수 있는지 여부를 판정하는 단계, 및

상기 로드 불가능한 레코드의 로딩이 스kip될 수 없다는 판정에 응답하여, 제3 모드에서 상기 데이터 파일의 로드를 시도하는 단계 - 상기 제3 로딩 모드에서는 사용자 데이터에 대응하는 상기 데이터 파일의 레코드만을 로드함 -

를 더 포함하는, 데이터 파일 로딩 방법.

**청구항 14**

제3항에 있어서,

상기 사용자 데이터는 사용자에 의하여 입력된 텍스트 데이터를 포함하는, 데이터 파일 로딩 방법.

청구항 15

제3항에 있어서,

상기 사용자 데이터는 사용자에 의하여 입력된 수치 데이터를 포함하는, 데이터 파일 로딩 방법.

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

## 명세서

### 발명의 상세한 설명

#### 발명의 목적

#### 발명이 속하는 기술 및 그 분야의 종래기술

[0011]

관련 출원에 대한 상호 참조

[0012]

본 출원은 그 전반적인 내용이 본 명세서에 참조로서 포함되어 있는 미국 특허 출원번호 제11/018,916호(대리인 참조번호 60001.0465US01, 명칭 "Method and Computer-Readable Medium For Verifying and Saving an Electronic Document")와 관련이 있다.

[0013]

오늘날의 사회는 널리 보급된 컴퓨터를 사용하여 다양한 목적의 작업 및 엔터테인먼트를 실행한다. 예컨대, 오늘날의 컴퓨터는, 게임, 통신, 리서치, 및 사실상 끝이 없는 다양한 애플리케이션으로 사용한다. 사업 및 개인적인 용도에 있어서, 컴퓨터의 가장 일반적인 사용 중 하나는 전자 문서(document)의 생성 및 인쇄이다. 스프레드시트, 프레젠테이션, 워드 프로세싱 문서, 다이어그램 및 디지털 영상과 같은 그래픽 문서, CAD(computer-aided design) 문서, 및 수많은 다른 타입의 전자 문서를 포함하는 모든 종류의 전자 문서를 생성하기 위한 컴퓨터 애플리케이션 프로그램이 존재한다.

[0014]

흔히 전자 문서는 매우 중요한 콘텐츠를 포함한다. 또한, 수많은 경우에 있어서, 손상된 전자 문서의 콘텐츠는 재생성하기가 어렵거나, 불가능하다. 예컨대, 흔히 생성되는 매우 복잡한 법률, 사업, 마케팅 및 기술 문서는, 그 문서를 저장하는 데이터 파일이 손상 또는 파손되는 경우, 용이하게 재생성할 수 없다. 문서의 콘텐츠를 용이하게 재생성할 수 있는 경우라도, 데이터의 작은 부분이라도 잃어버린다는 점은 사용자를 매우 실망스럽게 할 수 있다. 따라서, 전자 문서에 포함된 데이터를 파손 및 손상으로부터 보호한다는 점은 매우 중요하다.

[0015]

현재의 컴퓨터 시스템은 의도하지 않은 파손 또는 손상으로부터 데이터 파일을 보호하는 여러 검사 및 다른 메

커니즘을 포함한다. 불행하게도, 이러한 메커니즘을 적절하게 이용하더라도, 전자 문서를 저장하는 데이터 파일이 손상되는 것은 매우 일반적이다. 데이터 파일을 대용량 저장부(storage)에 저장하는 동안에 손상이 발생할 수도 있고, 또는 고장이 난 저장 디바이스 또는 컨트롤러 하드웨어의 결과로서 손상이 발생할 수도 있다. 데이터 파일은, 네트워크 전송 동안, 또는 수많은 타입의 이벤트의 발생에 의해서도 손상될 수도 있다. 작은 양의 데이터 손실에도 사용자가 실망할 수 있기 때문에, 그리고 손상된 문서를 재생성하는데 많은 시간 및 노력이 필요하기 때문에, 손상된 데이터 파일로부터 가능한 한 많은 데이터를 복원하는 것이 매우 중요하다.

[0016] 본 발명의 다양한 실시예는 이러한 점을 고려하여 이루어진다.

### 발명이 이루고자 하는 기술적 과제

[0017] 본 발명에 따르면, 전자 데이터 파일의 콘텐츠를 로딩하기 위한 방법 및 컴퓨터 판독가능 매체는 상술한 문제들을 해결한다. 본 발명의 다양한 실시예를 이용함으로써, 데이터 파일의 로드(load) 동안, 파일의 손상된 부분(portion) 또는 레코드(record)를 식별하고, 이러한 부분의 수리(repair, 또는 복구)를 시도한다. 손상된 부분을 수리할 수 없는 경우, 이러한 부분의 로딩은 스킵(skip)한다. 데이터 파일의 손상되지 않은 부분 및 수리된 부분은 메모리에 로드한다. 데이터 파일의 부분을 수리 또는 스킵할 수 없는 경우, 데이터 파일에 포함된 사용자 데이터만을 로드하려고 시도한다. 이러한 방식으로, 데이터 파일에 포함된 사용자 데이터는, 데이터 파일의 나머지 부분이 심하게 손상되는 경우에도 로드할 수도 있다.

[0018] 본 발명의 양상에 따르면, 하나 이상의 부분을 포함한 데이터 파일을 로딩하기 위한 방법을 제공한다. 본 방법에 따르면, 많은 로드 모드(mode)를 제공한다. "정상" 로드 모드는 데이터 파일의 각 부분을 정상적인 방식으로 로드하려고 시도한다. 정상 로드 모드는 데이터 파일의 각 부분에 대한 최소 무결성 검사(minimal integrity checking)를 포함하여, 데이터 파일을 신속하게 로드할 수 있다. 정상 모드 동안, 데이터 파일의 부분이 손실 또는 손상되는 경우, "안전" 로드 모드로 명명한 제2 모드를 이용해 데이터 파일의 부분을 로드하려고 시도한다. 데이터 파일의 부분이, 그 부분을 로드하려고 시도하는 애플리케이션 프로그램에서의 에러 또는 충돌의 원인이 되는 경우, 그 부분이 예기치 않은 데이터 값을 포함하는 경우, 그 부분이 손실 데이터인 경우, 그 부분이 무효 레코드 또는 무효 XML(extensible markup language)인 경우, 및 다른 원인 때문에, 데이터 파일의 부분은 손상되어 로드 불가능한 것으로 고려할 수도 있다.

[0019] 안전 로드 모드는 데이터 파일의 각 부분에 대해 광범위한 무결성 검사를 수행한다. 안전 로드 모드는 데이터 파일의 손상된 부분의 수리를 또한 시도할 수도 있다. 수리될 수 있는 부분은 로드한다. 안전 로드 모드에서, 데이터 파일의 부분이 손실 또는 손상되며, 또한 수리 불가능하게 되는 경우, 수리 불가능한 부분의 로딩은 스킵한다. 데이터 파일의 부분이 수리 불가능하게 되고, 로딩을 스킵할 수 없는 경우, "복원" 로드 모드로 명명한 제3 로드 모드를 이용해 데이터 파일의 부분을 로드하려고 시도한다.

[0020] 복원 로드 모드는 사용자 데이터를 포함하는 데이터 파일의 부분만을 로드한다. 예컨대, 사용자 데이터는 사용자가 입력한 텍스트 데이터 또는 수치(numerical) 데이터를 포함할 수도 있다. 예로서, 데이터 파일이 스프레드시트를 포함하는 경우, 복원 로드 모드는 스프레드시트의 셀(cell)에 포함된 데이터만을 로드하려고 시도한다. 복원 모드는, 데이터 파일에 포함할 수도 있는 다른 타입의 데이터, 예를 들어 피벗 테이블(pivot table), 리스트 오브젝트, 지정 범위(named range), 자동 필터, 스타일, 포맷팅, 및 애플리케이션 또는 사용자 선호도의 로드를 시도하지는 않는다.

[0021] 본 발명의 다른 실시예에 따르면, 컴퓨터 실행가능 명령을 저장한 컴퓨터 판독가능 매체를 또한 제공한다. 컴퓨터가 컴퓨터 실행가능 명령을 실행하는 경우, 컴퓨터 실행가능 명령으로 인하여, 컴퓨터는, 하나 이상의 부분을 포함한 데이터 파일을 로딩하기 위한 제1 로딩 모드를 제공한다. 제1 로딩 모드는, 데이터 파일의 부분을 로드할 때, 데이터 파일의 부분에 대해 최소 무결성 검사를 수행한다. 컴퓨터 실행가능 명령으로 인하여, 컴퓨터는, 데이터 파일을 로딩하기 위한 제2 로딩 모드를 또한 제공하여, 데이터 파일의 부분에 대해 제1 로딩 모드보다 더 광범위한 무결성 검사를 수행한다. 제2 로딩 모드는 로드 불가능한 부분의 수리를 또한 시도할 수도 있다. 더욱이, 제2 로딩 모드는 로드 불가능한 부분의 로딩을 스킵한다.

[0022] 컴퓨터 실행가능 명령으로 인하여, 컴퓨터는 제1 로딩 모드에서 데이터 파일을 로딩하기 시작한다. 제1 로딩 모드에서 데이터 파일의 부분이 로드 불가능한 것으로 판정되는 경우, 컴퓨터는 제2 로딩 모드로 전환하고, 이로딩 모드에서 데이터 파일을 로드하려고 시도한다. 제2 로딩 모드에서 로드 불가능한 부분을 수리할 수도 있는 경우, 로드 불가능한 부분을 수리하여 로드한다. 로드 불가능한 부분을 수리할 수 없는 경우, 로드 불가능한 부분의 로딩은 스킵한다.

[0023] 본 발명의 실시예에 따르면, 컴퓨터 실행가능 명령으로 인하여, 컴퓨터는, 사용자 데이터를 포함하는 데이터 파일의 부분만을 로드하는 제3 로딩 모드를 제공한다. 제2 로딩 모드에서 데이터 파일의 부분이 로드 불가능하고, 로드 불가능한 부분을 수리 또는 스kip할 수 없는 경우, 제3 로딩 모드는 데이터 파일의 콘텐츠를 로드하려고 시도한다.

[0024] 본 발명은, 컴퓨터 프로세스, 컴퓨팅 시스템, 또는 컴퓨터 프로그램 제품 또는 컴퓨터 판독가능 매체와 같은 제조 물품으로서 구현할 수도 있다. 컴퓨터 프로그램 제품은, 컴퓨터 시스템에 의해 판독가능하고, 컴퓨터 프로세스를 실행하기 위한 명령들의 컴퓨터 프로그램을 인코딩함으로써 판독가능한 컴퓨터 저장 매체일 수도 있다. 또한, 컴퓨터 프로그램 제품은, 컴퓨팅 시스템에 의해 판독가능하고, 컴퓨터 프로세서를 실행하기 위한 명령들의 컴퓨터 프로그램을 인코딩함으로써 판독가능한 캐리어(carrier)상의 전파 신호(propagted signal)일 수도 있다.

[0025] 본 발명을 특정하는 이와 같은 다양한 특징뿐만 아니라 장점은, 관련 도면을 참조하여 이하의 상세한 설명을 통해 명백해질 것이다.

### 발명의 구성 및 작용

[0026] 이제, 비슷한 숫자는 비슷한 엘리먼트를 표현하는 도면들을 참조하여, 본 발명의 다양한 양상을 기술한다. 특히, 도 1 및 이에 대한 설명은, 본 발명의 실시예를 구현할 수도 있는 적합한 컴퓨팅 환경에 대해 간략하면서도 일반적으로 기술한다. 본 발명은 퍼스널 컴퓨터의 오피레이팅 시스템에서 실행하는 프로그램 모듈의 일반적인 환경하에서 기술하지만, 본 기술분야의 숙련자는 다른 타입의 컴퓨터 시스템 및 프로그램 모듈의 조합으로도 본 발명을 구현할 수도 있음을 인식할 것이다.

[0027] 일반적으로, 프로그램 모듈은, 루틴, 프로그램, 컴포넌트, 데이터 구조, 및 특정 작업을 수행하거나, 특별한 추상적인 데이터 타입을 구현하는 다른 타입의 구조를 포함한다. 또한, 본 기술분야의 숙련자는, 핸드헬드 디바이스, 멀티프로세서 시스템, 마이크로프로세서 기반 또는 프로그램가능 소비자 가전제품, 미니컴퓨터, 메인프레임 컴퓨터 등을 포함하는 다른 컴퓨터 시스템 구성은 이용해 본 발명을 실행할 수도 있음을 인식할 것이다. 본 발명은, 통신 네트워크를 통해 링크되는 원격 프로세싱 디바이스에 의해 작업이 실행되는 분산형 컴퓨팅 환경에서 또한 실행할 수도 있다. 분산형 컴퓨팅 환경에 있어서, 프로그램 모듈은 로컬 및 원격 메모리 저장 디바이스에 위치할 수도 있다.

[0028] 이제 도 1을 참조하여, 본 발명의 다양한 실시예에 사용되는 컴퓨터(2)에 대한 예시적인 컴퓨터 아키텍처를 기술한다. 도 1에 도시한 컴퓨터 아키텍처는, CPU(5), RAM(9) 및 ROM(11)을 포함한 시스템 메모리(7), 및 메모리를 CPU(5)와 연결하는 시스템 버스(12)를 포함하는 통상적인 데스크톱 또는 랩톱 컴퓨터를 나타낸다. 컴퓨터 내 엘리먼트 간의 정보 전송을 돋는 기본 루틴을 포함하는 기본 입/출력 시스템은, 예를 들어 스타트업(startup) 동안에 ROM(11)에 저장한다. 컴퓨터(2)는, 오피레이팅 시스템(16), 애플리케이션 프로그램 및 기타 프로그램 모듈을 저장하기 위한 대용량 저장 디바이스(14)를 더 포함하는데, 이하에서 상세하게 기술한다.

[0029] 대용량 저장 디바이스(14)는 버스(12)와 연결되는 대용량 저장 컨트롤러(도시하지 않음)를 통해 CPU(5)와 연결된다. 대용량 저장 디바이스(14) 및 그와 관련된 컴퓨터 판독가능 매체는 컴퓨터(2)에 비휘발성 저장부를 제공한다. 본 명세서에 포함되는 컴퓨터 판독가능 매체에 대한 설명은, 하드디스크 또는 CD-ROM 드라이브와 같은 대용량 저장 디바이스를 나타내지만, 본 기술분야의 숙련자는, 컴퓨터 판독가능 매체는 컴퓨터(2)가 액세스할 수 있는 어떤 이용가능한 매체일 수 있음을 인식해야 한다.

[0030] 예로서, 컴퓨터 판독가능 매체는 컴퓨터 저장 매체 및 통신 매체를 포함할 수도 있지만, 이에 한정하지는 않는다. 컴퓨터 저장 매체는, 컴퓨터 판독가능 명령, 데이터 구조, 프로그램 모듈 또는 다른 데이터와 같은 정보의 저장을 위한 어떤 메소드(method) 또는 기술로 구현하는, 휘발성 및 비휘발성 매체 그리고 분리형 및 비분리형 매체를 포함한다. 컴퓨터 저장 매체는, RAM, ROM, EPROM, EEPROM, 플래시 메모리 또는 기타 고체 상태의 메모리 기술, CD-ROM, DVD(digital versatile disk), 또는 기타 광 저장, 마그네틱 카세트, 마그네틱 테이프, 마그네틱 디스크 저장 또는 다른 마그네틱 저장 디바이스, 또는 소정의 정보를 저장하기 위해 사용할 수 있고, 컴퓨터(2)가 액세스할 수 있는 어떤 다른 매체를 포함하지만, 이에 한정하지는 않는다.

[0031] 본 발명의 다양한 실시예에 따르면, 컴퓨터(2)는, 인터넷과 같은 네트워크(18)를 통해 원격의 컴퓨터와 논리적으로 연결해 이용하는 네트워크형 환경에서 동작할 수도 있다. 컴퓨터(2)는 버스(12)와 연결되는 네트워크 인터페이스 유닛(20)을 통해 네트워크(18)와 연결할 수도 있다. 네트워크 인터페이스 유닛(20)을 이용해 다른 타입의 네트워크 및 원격 컴퓨터 시스템과 또한 연결할 수도 있음을 인식해야 한다. 컴퓨터(2)는, 키보드, 마우

스, 또는 전자 스타일러스(도 1에 도시하지 않음)를 포함하는 수많은 다른 디바이스로부터의 입력을 수신 및 처리하기 위한 입/출력 컨트롤러(22)를 또한 포함할 수도 있다. 유사하게, 입/출력 컨트롤러(22)는, 표시 스크린, 프린터, 또는 다른 타입의 출력 디바이스에 출력을 제공할 수도 있다.

[0032] 간략하게 상술한 바와 같이, 수많은 프로그램 모듈 및 데이터 파일은, 위상된 주, 레드몬드에 위치한 MICROSOFT CORPORATION의 WINDOWS XP 오퍼레이팅 시스템과 같은, 네트워크형 퍼스널 컴퓨터의 동작 제어에 적합한 오퍼레이팅 시스템(16)을 포함하는 컴퓨터(2)의 대용량 저장 디바이스(14) 및 RAM(9)에 저장할 수도 있다. 대용량 저장 디바이스(14) 및 RAM(9)은 하나 이상의 프로그램 모듈을 또한 저장할 수도 있다. 특히, 대용량 저장 디바이스(14) 및 RAM(9)은 스프레드시트 애플리케이션 프로그램(10)을 저장할 수도 있다. 본 기술분야의 숙련자에게 공지된 바와 같이, 스프레드시트 애플리케이션 프로그램(10)은 전자 스프레드시트를 생성 및 편집하기 위한 기능을 제공하도록 동작한다.

[0033] 본 발명의 실시예에 따르면, 스프레드시트 애플리케이션 프로그램(10)은 MICROSOFT CORPORATION의 EXCEL 스프레드시트 애플리케이션 프로그램을 포함한다. 그러나 다른 제조업체의 다른 스프레드시트 애플리케이션 프로그램을 이용해 본 발명의 다양한 양상을 구현할 수도 있음을 인식해야 한다. 본 명세서에서 기술하는 본 발명의 실시예는 스프레드시트 애플리케이션 프로그램의 환경에서 제공하지만, 데이터 파일로부터 데이터를 로드(load)하는 임의의 타입의 애플리케이션 프로그램을 본 발명에 이용할 수도 있음을 또한 인식해야 한다. 예컨대, 본 명세서에서 기술하는 본 발명의 실시예는, 워드 프로세싱 애플리케이션 프로그램, 프레젠테이션 애플리케이션 프로그램, 드로잉(drawing) 또는 CAD(computer-aided design) 애플리케이션 프로그램, 또는 데이터베이스 애플리케이션 프로그램에서 활용할 수도 있다.

[0034] 스프레드시트 문서의 생성 및 편집에 있어서, 스프레드시트 애플리케이션 프로그램(10)은 데이터 파일(24)에 있는 스프레드시트 문서를 대용량 저장 디바이스(14)상에 저장하기 위한 기능을 제공한다. 데이터 파일(24)은, 스프레드시트 셀의 콘텐츠, 애플리케이션 선호도, 포맷팅 정보를 포함하는 사용자 데이터, 및 스프레드시트 애플리케이션 프로그램(10)이 제공하는 다양한 특징에 대응하는 다른 데이터와 같은 스프레드시트 문서의 다양한 양상을 표현하는 데이터를 포함한다. 데이터 파일(24)의 구조에 관한 추가 설명은 도 2를 참조하여 이하에서 제공한다.

[0035] 본 발명의 실시예에 따르면, 스프레드시트 애플리케이션 프로그램(10)은, 컴퓨터(2)의 대용량 저장 디바이스(14)로부터 RAM(9)으로 데이터 파일(24)을 로딩함으로써, 스프레드시트 문서의 콘텐츠를 재저장하기 위한 기능을 제공한다. 도 2 내지 3(b)를 참조하여 이하에서 상세하게 기술하는 바와 같이, 스프레드시트 애플리케이션 프로그램(10)이 사용하는 데이터 파일(24)을 로딩하기 위한 방법은, 데이터 파일(24)에서의 손상 가능성을 설명하고, 데이터 파일(24)이 손상된 경우에도, 데이터 파일(24)로부터 로드하는 데이터의 양을 최대화하도록 시도한다.

[0036] 이제, 도 2를 참조하여, 데이터 파일(24)의 구조와, 스프레드시트 애플리케이션 프로그램(10)이 사용하는 로딩 메커니즘의 동작에 관해 추가로 설명한다. 도 2에 도시한 바와 같이, 데이터 파일(24)은 수많은 부분(portion)(26A ~ 26N)으로 서브분할된다. 각 부분(26A ~ 26N)은 스프레드시트 애플리케이션 프로그램(10)이 지원하는 하나 이상의 특징에 관한 정보를 저장하기 위해 사용한다. 또한, 상이하지만 관련된 특징들에 대한 정보는 부분(26A ~ 26N) 중 하나의 부분에 저장할 수도 있다. 예컨대, 도 2에 도시한 바와 같이, 특징 A ~ C에 대한 데이터는 부분 26A에 저장한다. 특징 D에 대한 데이터는 부분 26B에 저장한다. 특징 E ~ G에 대한 데이터는 부분 26C에 저장한다. 사용자 데이터는 임의의 부분(26A ~ 26N)에 저장할 수도 있다.

[0037] 간략하게 상술하고, 도 2에 도시한 바와 같이, 부분(26A ~ 26N)에 포함되는 데이터가 손상될 가능성이 존재한다. 데이터 파일을 대용량 저장부에 저장하는 동안에 손상이 발생할 수도 있고, 또는 고장이 난 저장 디바이스 또는 컨트롤러 하드웨어의 결과로서 손상이 발생할 수도 있다. 데이터 파일은, 네트워크 전송 동안, 또는 수많은 다른 타입의 이벤트 발생에 의해서 또한 손상될 수도 있다. 특정 부분에 대한 데이터가 손실되도록 또한 결정할 수도 있다. 데이터 파일의 부분이, 그 부분을 로드하려고 시도하는 애플리케이션 프로그램에서의 에러 또는 충돌의 원인이 되는 경우, 그 부분이 예기치 않은 데이터 값을 포함하는 경우, 그 부분이 손실 데이터인 경우, 그 부분이 무효 레코드(record) 또는 무효 XML(extensible markup language)을 포함하는 경우, 및 다른 원인 때문에, 데이터 파일의 부분은 손상되어 로드 불가능한 것으로 고려할 수도 있다. 도 2에 도시한 예시적인 데이터 파일(24)에 있어서, 부분 26B 및 26D는 손상된 부분이다.

[0038] 본 명세서에서 기술하는 바와 같이, 부분 26B 및 26D가 손상되었음에도, 스프레드시트 애플리케이션 프로그램(10)은 데이터 파일(24)의 부분들을 로드할 수 있다. 도 2는 예시적인 데이터 파일(24)을 이용하는 이러한

딩 프로세스를 또한 나타낸다. 특히, 스프레드시트 애플리케이션 프로그램(10)은 정상 로딩 모드에서 데이터 파일(24)을 로딩하기 시작한다. 정상 로딩 모드에서 데이터 파일의 부분(26A~26N)에 대해 최소 무결성 검사(minimal integrity checking)를 수행한다. 정상 모드에서 로딩하는 동안, 데이터 파일(24)의 손상된 부분을 만나는 경우, 스프레드시트 애플리케이션 프로그램(10)은 안전 로딩 모드로 전환하고, 데이터 파일(24)을 처음부터 로딩하기 시작한다. 예컨대, 도 2에 도시한 바와 같이, 정상 로딩 모드에 있어서, 손상된 부분 26B와 만나는 경우, 로딩 모드는 안전 로딩 모드로 전환하고, 데이터 파일(24)의 처음부터 다시 로딩하기 시작한다.

[0039] 안전 로딩 모드에서 데이터 파일의 부분(26A~26N)에 대해 정상 로딩 모드와 비교되는 추가 무결성 검사를 수행한다. 추가로, 안전 모드에서 로딩하는 동안, 손상된 부분을 만나는 경우, 그 손상된 부분의 수리(repair)를 시도한다. 손상된 부분을 수리할 수 있는 경우, 그 부분은 로드한다. 손상된 부분을 수리할 수 없는 경우, 손상된 부분의 로딩은 스킵(skip)한다. 예를 들어, 도 2에 도시한 부분 26B는 손상되어 수리할 수 없다. 그러므로 부분 26B의 로딩은 스킵하고, 부분 26C는 로드한다.

[0040] 부분 26C를 로드한 이후, 부분 26D의 로드를 시도한다. 그러나 도 2에 도시한 바와 같이, 부분 26D는 손상된 부분이다. 따라서, 부분 26D의 수리를 시도한다. 부분 26D를 로드할 수 없는 경우, 부분 26D의 로딩은 스킵하고, 이 프로세스는, 남아 있는 부분들을 로드 또는 스킵할 때까지 계속한다. 본 발명의 실시예에 따르면, 로드 불가능한 부분을 만나고, 그 부분을 수리 불가능한 것으로 판정한 이후, 데이터 파일(24)의 로딩은 데이터 파일(24)의 처음으로 되돌아갈 수도 있다. 이는 도 2에서 도시한다. 이러한 방식으로 데이터 파일(24)의 처음으로 되돌아감으로써, 로드 불가능한 부분과 관련되는 데이터 파일(24)의 다른 부분들의 로딩은, 관련된 부분들이 손상되지 않았을 수도 있지만 스킵한다.

[0041] 데이터 파일(24)의 로딩 동안, 로드 불가능하고 수리 불가능한 부분을 만나는 경우, 스프레드시트 애플리케이션 프로그램(10)은, 복원 로딩 모드로 명명한 제3 로딩 모드로 전환할 수도 있다. 복원 로딩 모드는 데이터 파일로부터 사용자 데이터만을 로드하려고 시도한다. 특히 텍스트 문서에 관해서는, 문서의 텍스트만을 로드하려고 시도한다. 스프레드시트 문서에 관해서는, 사용자가 입력한 데이터, 공식(formula) 및 공식으로 생성된 데이터를 포함하는, 스프레드시트 셀들의 콘텐츠를 로드하려고 시도한다. 이러한 방식에 있어서, 데이터 파일의 부분들이 손상되는 경우라도, 사용자 데이터의 일부 또는 전부를 복원 및 로드할 수도 있다. 이러한 프로세서는, 도 2에서 점선으로 나타내고 있으며, 부분 26D가 로드 불가능하고 수리 불가능한 것으로 판정되는 경우에 수행되며, 그 부분 없이는 파일을 조금도 로드해서는 안 된다는 점을 나타낸다. 이러한 프로세스에 관해서는 도 3(a) 및 3(b)를 참조하여 이하에서 추가로 설명한다.

[0042] 이제, 도 3(a) 및 3(b)를 참조하여, 데이터 파일(24)의 콘텐츠를 로딩하기 위해 스프레드시트 애플리케이션 프로그램(10)이 실행하는 프로세스를 나타내는 루틴(300)을 기술한다. 본 명세서에서 제공하는 루틴에 대한 설명을 판독할 때, 본 발명의 다양한 실시예의 논리적 동작은, (1) 컴퓨팅 시스템상에서 동작하는, 컴퓨터로 구현되는 행동(act)들 또는 프로그램 모듈들의 시퀀스로서, 및/또는 (2) 컴퓨팅 시스템 내의 상호연결된 머신 논리 회로들 또는 회로 모듈들로서 구현함을 인식해야 한다. 그러한 구현은 본 발명을 구현하는 컴퓨팅 시스템의 성능 요구조건에 좌우되는 선택의 문제이다. 따라서, 도 3(a) 및 3(b)에 도시한 논리적 동작과, 본 명세서에서 기술하는 본 발명의 실시예의 구성은, 다양한 동작, 구조적 디바이스, 행동 또는 모듈로서 간주한다. 본 기술분야의 숙련자는, 이러한 동작, 구조적 디바이스, 행동 및 모듈은, 본 명세서의 청구범위에서 기술하는 본 발명의 사상 및 범위를 벗어나지 않는 소프트웨어, 펌웨어, 특정 목적의 디지털 로직, 및 그들의 조합으로 구현할 수도 있음을 인식할 것이다.

[0043] 루틴(300)은 그 동작에 있어서 복수의 변수를 사용함을 인식해야 한다. 특히, "모드" 변수는 현재 로딩 모드의 트랙(track)을 유지한다. 이 변수는, "안전", "정상" 또는 "복원"으로 설정될 수도 있다. "스킵 카운터(skip counter)" 변수는, 로딩 동안 스킵해야 하는 각 부분을 개별적으로 식별하기 위해 사용하는 데이터 구조이다. "스킵 할 레코드의 수" 변수는, 현재의 로드 시도 시 스킵해야 하는 섹션들의 현재 수를 기술한다. "현재 레코드" 변수는, 처리되는 데이터 파일 내의 현재 섹션을 식별한다. 더 많거나 적은 변수를 사용하여 동일한 작업을 수행할 수도 있음을 인식해야 한다. 또한, 도 3에 도시한 루틴(300)은 단지 본 발명의 하나의 가능한 구현을 표현한다는 점과, 다른 많은 구현이 본 기술분야의 숙련자에게 명백해질 것이라는 점을 인식해야 한다.

[0044] 루틴(300)은 동작 302, 304 또는 306에서 시작한다. 특히, 본 발명의 실시예에 따르면, 사용자 인터페이스를 제공함으로써, 사용자가, 데이터 파일이 정상적으로 로드되는지(동작 304), 안전 로딩 모드에서 로드되는지(동작 302), 또는 복원 로딩 모드에서 로드되는지(동작 306)를 선택할 수도 있다. 이 사용자 인터페이스는, 사용자가 파일의 로드를 요청하는 경우에 사용자에게 제공할 수도 있다. 사용자 인터페이스에서의 사용자의 선택에

기초하여, 루틴(300)은 동작 302, 304 또는 306에서 자신의 동작을 시작한다.

[0045] 안전 로딩 모드에서 로딩을 시작하는 경우, 루틴(302)은 모드 변수가 "안전"으로 설정되는 동작 302에서 시작한다. 다음으로, 루틴(300)은 동작 308을 계속한다. 정상 로딩 모드에서 로딩을 시작하는 경우, 루틴(300)은 모드 변수가 "정상"으로 설정되는 동작 304에서 시작한다. 다음으로, 루틴(300)은 동작 304에 후속하여 동작 308을 계속한다. 복원 로딩 모드에서 로딩을 시작하는 경우, 루틴(300)은 모드 변수가 "복원"으로 설정되는 동작 306에서 시작한다. 동작 306에 후속하여, 루틴(300)은 이하에서 기술하는 동작 348을 계속한다.

[0046] 동작 308에서, 스kip 카운터는, 스kip해야 하는 레코드가 없다는 점을 나타내도록 초기화된다. 다음으로, 루틴(300)은 현재 레코드를 데이터 파일에서의 제1 레코드로 설정하는 동작 310을 계속한다. 스kip할 레코드의 수 변수도 마찬가지로 초기화된다. 제1 경로상에서, 이는, 스kip할 레코드의 수가 0과 동일하다고 설정한다. 동작 310에 후속하여, 루틴(300)은 동작 312를 계속한다.

[0047] 동작 312에서, 현재 레코드를 현재 모드에서 로드하려고 시도한다. 예컨대, 모드 변수가 "정상"과 동일한 경우, 로드되고 있는 섹션에 대해 최소 무결성 검사를 수행한다. 모드 변수가 "안전"과 동일한 경우, 추가 무결성 검사를 수행한다. 동작 312에 후속하여, 루틴(300)은, 현재 레코드가 로드 불가능한 것인지(즉, 손상 또는 손실)를 판정하는 동작 314를 계속한다. 현재 레코드가 로드 가능한 경우, 루틴(300)은, 더 많은 레코드가 여전히 로드되는지를 판정하는 동작 316으로 분기한다. 더 많은 레코드가 존재하는 경우, 루틴(300)은 동작 316에서 동작 318로 분기하여, 현재 레코드 변수를 데이터 파일에서의 그 다음 레코드로 설정한다. 다음으로, 루틴(300)은 그 다음 레코드를 로드하는 동작 321을 계속한다. 동작 316에서, 로드되는 추가 레코드가 존재하지 않는 것으로 판정되는 경우, 루틴(300)은 그 루틴을 종료하는 동작 320으로 분기한다. 이러한 방식으로, 손상 또는 손실 레코드가 존재하지 않는 경우, 현재 모드에서 모든 레코드를 로드한다.

[0048] 본 발명의 실시예에 있어서, 일부 무결성 검사는 레코드 레벨과는 대조적인 특징 레벨에서 수행할 수도 있다. 그러한 특징 레벨 무결성 검사를 수행하기 위하여, 특별한 특징에 대한 모든 레코드를 로드한다. 다음으로, 그 특징에 대한 데이터가 유효한지를 판정한다. 데이터가 유효한 경우, 스kip 데이터 구조는 스kip할 특징에 대한 레코드를 이용해 업데이트하고, 파일은 리로드(reload)한다. 파일-레벨 일관성(consistency) 검사도 유사한 방식으로 또한 수행할 수도 있다.

[0049] 동작 314에서, 현재 레코드가 로드 불가능한 것으로 판정되는 경우, 루틴(300)은, 현재 모드가 정상 모드인지를 판정하는 동작 322를 계속한다. 현재 모드가 정상 모드인 경우, 루틴(300)은 동작 324로 분기하여, 데이터 파일의 부분을 식별하여 스kip할 필요가 있음을 나타내는 스kip 카운터 변수에 그 레코드를 플래그(flag)한다. 다음으로, 루틴(300)은, 모드 변수가 "안전"으로 설정되는 동작 326을 계속한다. 이러한 방식으로, 데이터 파일의 로드 불가능한 부분을 만나는 경우, 로딩 모드는 정상에서 안전으로 전환한다. 다음으로, 루틴(300)은 데이터 파일의 처리가 처음으로 되돌아 가는 동작 310으로 다시 되돌아 간다.

[0050] 동작 322에서, 현재 로딩 모드가 정상 모드가 아닌 것으로 판정되는 경우, 루틴(300)은, 현재 로딩 모드가 안전 모드인지를 판정하는 동작 328을 계속한다. 루틴(300)의 이 부분에서는 정상 또는 안전 로딩 모드만이 가능한 값이기 때문에, 현재 로딩 모드가 안전 모드가 아닌 경우, 루틴(300)은 에러를 보고하는 동작 330으로 분기한다. 다음으로, 루틴(300)은 동작 330에 후속하여 그 루틴을 종료하는 동작 320을 계속한다. 그러나 동작 328에서, 현재 모드가 안전 모드로 판정되는 경우, 루틴(300)은 동작 332를 계속한다.

[0051] 동작 332에서, 현재 레코드의 수리를 시도한다. 동작 334에서, 현재 레코드가 수리 가능한지를 판정한다. 레코드가 수리 가능한 경우, 루틴(300)은 현재 레코드를 로드하는 동작 336으로 분기한다. 동작 336에서, 레코드가 수리 가능하기 때문에, 현재 레코드의 로딩을 스kip하지 않아야 한다는 점을 나타내도록 스kip 카운터 변수를 또한 업데이트한다. 동작 336에 후속하여, 루틴(300)은 동작 316으로 다시 분기하여, 데이터 파일의 잔여 레코드를 상술한 방식으로 처리한다.

[0052] 동작 334에서, 현재 레코드가 수리될 수 없는 것으로 판정되는 경우, 루틴(300)은 동작 338로 분기한다. 동작 338에서, 스kip할 레코드의 수가 0과 동일한지를 판정한다. 이는, 정상 모드에서 로딩을 시작하는 경우이고, 제1 손상 레코드를 만나게 된 경우이며, 그 레코드는 수리 불가능하다. 이 경우, 루틴(300)은 동작 340으로 분기하여, 레코드를 스kip해야 함을 나타내도록 스kip 카운터 변수를 업데이트한다. 다음으로, 루틴(300)은 동작 310으로 되돌아 가서 상술한 방식으로 데이터 파일을 처리한다.

[0053] 동작 338에서, 스kip할 레코드의 수가 0과 동일하지 않은 경우, 루틴(300)은 현재 레코드의 로딩을 스kip하려고 시도하는 동작 342를 계속한다. 동작 344에서, 현재 레코드의 로딩을 스kip할 수도 있는지를 판정한다. 현재

레코드의 로딩을 스kip할 수 있는 경우, 루틴(300)은 레코드를 스kip 레코드 변수에 플래그하는 동작 346으로 분기한다. 다음으로, 루틴(300)은 상술한 동작 316을 계속한다.

[0054] 동작 344에서, 현재 레코드를 스kip할 수 없는 것으로 판정되는 경우, 루틴(300)은 모드 변수가 "복원"으로 설정되는 동작 306을 계속한다. 다음으로, 루틴(300)은 복원 모드에서 데이터 파일을 로드하려고 시도하는 동작 348을 계속한다. 상술한 바와 같이, 복원 모드는 사용자 데이터만을 로드한다. 또한, 사용자 데이터가 손상되는 경우, 가능한 한 많은 사용자 데이터를 로드하는 시도가 이루어진다. 다음으로, 루틴(300)은 그 루틴을 종료하는 동작 320을 계속한다.

[0055] 상술한 바에 기초하여, 본 발명의 다양한 실시예는, 데이터 파일의 콘텐츠를 로딩하기 위한 방법, 시스템, 장치 및 컴퓨터 판독가능 매체를 포함한다는 점을 인식해야 한다. 상술한 사양, 예 및 데이터는 본 발명의 합성물에 대한 제조 및 사용에 대해 완전하게 기술한다. 본 발명의 사상 및 범위를 벗어나지 않는 본 발명의 많은 실시예가 이루어질 수 있기 때문에, 본 발명은 이하에 첨부하는 청구범위에 귀속한다.

### 발명의 효과

[0056] 본 방법에 따르면, 데이터 파일의 손상된 부분을 식별하고, 이러한 부분의 수리를 시도한다. 손상된 부분을 수리할 수 없는 경우, 이러한 부분의 로딩은 스kip한다. 데이터 파일의 손상되지 않은 부분 및 수리된 부분은 메모리에 로드한다. 데이터 파일의 부분을 수리 또는 스kip할 수 없는 경우, 데이터 파일에 포함된 사용자 데이터만을 로드하려고 시도한다. 이러한 방식으로, 데이터 파일에 포함된 사용자 데이터는, 심한 손상이 발생하는 경우에도 로드할 수도 있다.

### 도면의 간단한 설명

[0001] 도 1은 본 발명의 다양한 실시예가 사용 및 제공하는 컴퓨터 시스템을 도시한 컴퓨터 시스템 구조도.

[0002] 도 2는 본 발명의 실시예가 제공하는 데이터 파일의 양상 및 다양한 로딩 모드를 도시한 블록도.

[0003] 도 3(a) 및 3(b)는 본 발명의 다양한 실시예에 따른 데이터 파일을 로딩하기 위한 예시적인 프로세스를 도시한 흐름도.

[0004] <도면의 주요 부분에 대한 부호의 설명>

[0005] 5 : CPU

[0006] 7 : 시스템 메모리

[0007] 14 : 대용량 저장 디바이스

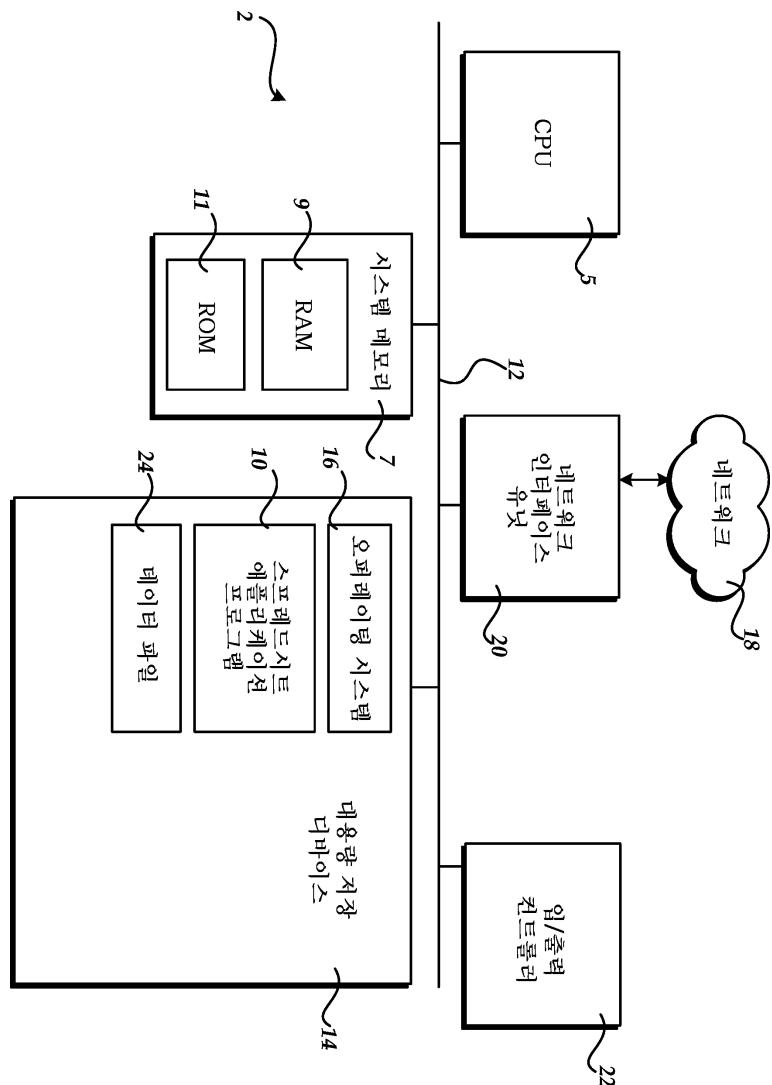
[0008] 18 : 네트워크

[0009] 20 : 네트워크 인터페이스 유닛

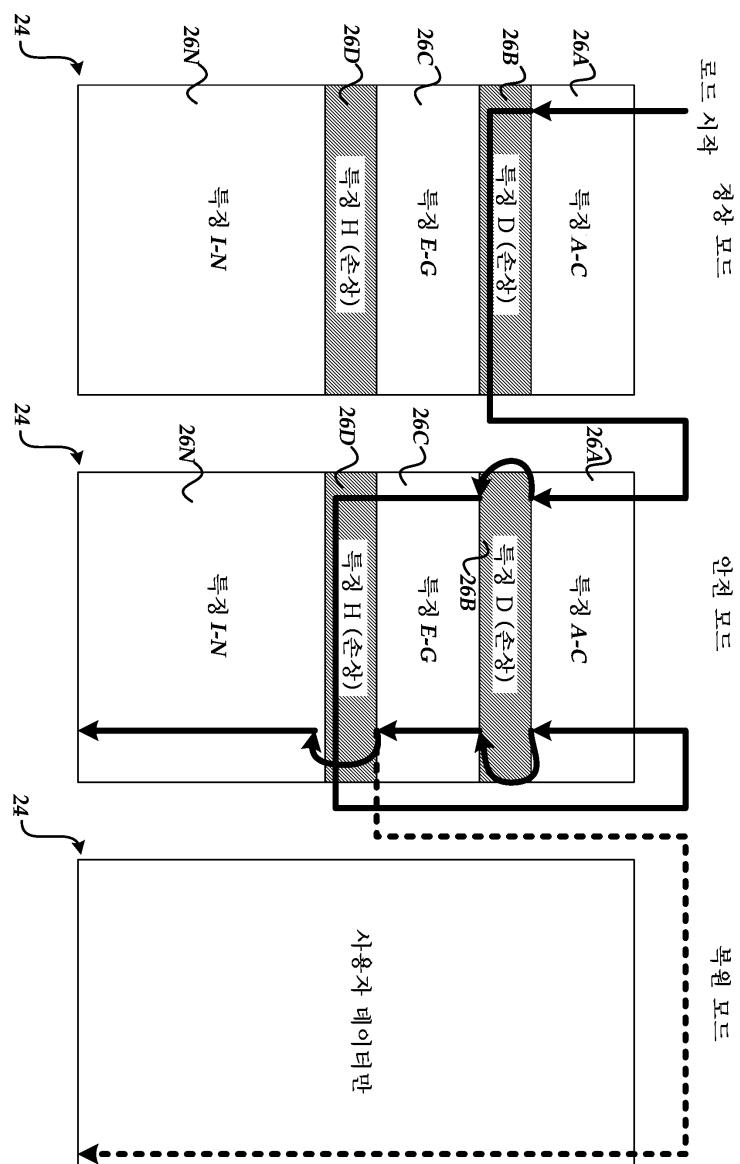
[0010] 22 : 입/출력 컨트롤러

도면

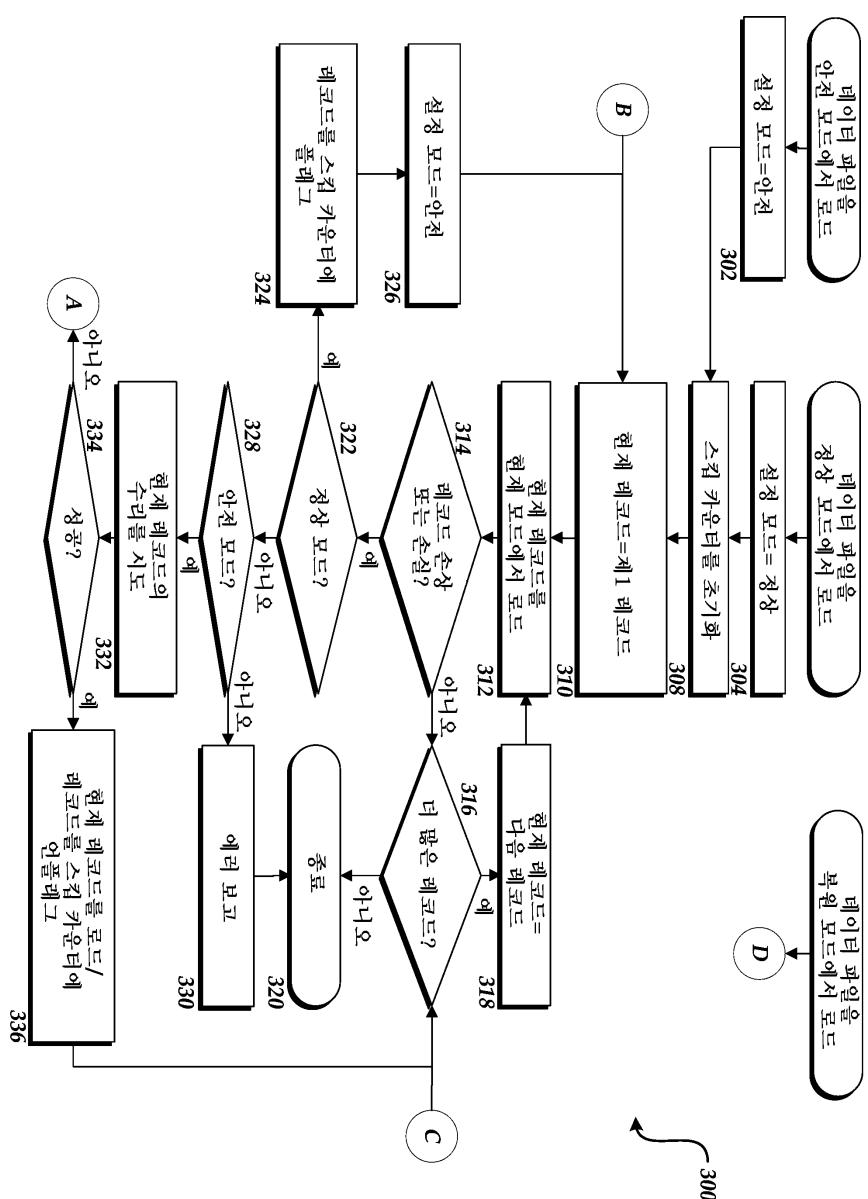
도면1



도면2



도면3a



도면3b

