



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2011년06월28일
(11) 등록번호 10-1044849
(24) 등록일자 2011년06월21일

(51) Int. Cl.
G06F 11/08 (2006.01) G06F 17/30 (2006.01)
G06F 11/00 (2006.01) G06F 11/22 (2006.01)
(21) 출원번호 10-2005-7012466
(22) 출원일자(국제출원일자) 2004년07월29일
심사청구일자 2009년07월23일
(85) 번역문제출일자 2005년06월30일
(65) 공개번호 10-2007-0006542
(43) 공개일자 2007년01월11일
(86) 국제출원번호 PCT/US2004/024565
(87) 국제공개번호 WO 2005/111867
국제공개일자 2005년11월24일
(30) 우선권주장
10/837,932 2004년05월03일 미국(US)
(56) 선행기술조사문헌
US19904945474 A1
US6671699 B1
US6314433 B1

(73) 특허권자
마이크로소프트 코포레이션
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원
마이크로소프트 웨이
(72) 발명자
오크스, 아덤 에이.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트
웨이마이크로소프트 코포레이션 내
코다발라, 하누만타 알.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트
웨이마이크로소프트 코포레이션 내
슬리만, 마틴 제이.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트
웨이마이크로소프트 코포레이션 내
(74) 대리인
주성민, 이중희, 백만기

전체 청구항 수 : 총 27 항

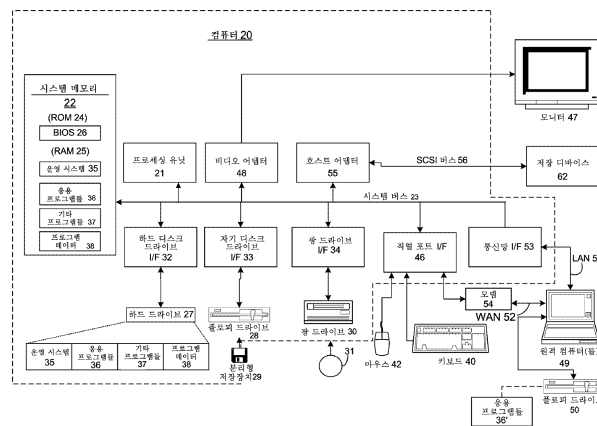
심사관 : 이병수

(54) 자동 데이터베이스 또는 파일 시스템 정비 및 수리를 위한시스템 및 방법

(57) 요약

본 발명은 일반적으로 데이터베이스 및 파일 시스템 관리에 관한 것이고, 더 구체적으로는, 데이터 신뢰성을 보장하기 위한 자동 데이터베이스와 파일 시스템 관리 및 수리에 관한 것이다. 본 발명의 다양한 양태들은 모든 데이터 페이지 유형들에 대한 데이터 페이지 레벨에서 데이터 손상에 응답하고 교정하는 것, 및 인덱스 페이지 손상(클러스터되고 클러스터되지 않음), 데이터 페이지 손상, 및 로그 파일의 페이지 손상을 포함하는, 하지만 이에 제한되지는 않는, 다양한 시나리오들에 대한 복구(재구성 또는 복원 동작들을 포함함)에 관련된다.

대표도



특허청구의 범위

청구항 1

컴퓨팅 장치에 의해 적어도 부분적으로 구현되는, 데이터베이스 파일 시스템(database file system;DBFS)을 위한 자동 데이터 신뢰성 시스템(data reliability system;DRS)으로서,

정책들(policies)의 세트에 기초하여 데이터베이스 관리(database administration;DBA) 작업들을 수행하는 서브시스템;

페이지 레벨에서 데이터 손상들의 세트에 응답하는 서브시스템;

손상된 페이지를 수리하기 위해 시도하는 제1 레벨의 복구를 위한 서브시스템;

상기 손상된 페이지가 수리될 수 없으면 손상된 페이지를 재구성하거나 복원하려고 시도하는 제2 레벨의 복구를 위한 서브시스템;

상기 DBFS와 연관된 데이터베이스가 트랜잭션면에서 일관적(transactionally consistent)인지를 판정하기 위해 상기 데이터베이스를 평가하는 서브시스템; 및

상기 데이터베이스가 트랜잭션면에서 일관적이라면 크래쉬 복구가 완료될 때까지 수리들을 지연시키고, 상기 데이터베이스가 트랜잭션면에서 일관적이지 않다면 지연 없이 수리들을 진행하는 서브시스템

을 포함하는 자동 데이터 신뢰성 시스템.

청구항 2

제1항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하는 제2 레벨의 복구를 위한 서브시스템은 인덱스 페이지 손상들을 해결하는 서브시스템을 포함하는 자동 데이터 신뢰성 시스템.

청구항 3

제1항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하는 제2 레벨의 복구를 위한 서브시스템은 데이터 페이지 손상들을 해결하는 서브시스템을 포함하는 자동 데이터 신뢰성 시스템.

청구항 4

제1항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하는 제2 레벨의 복구를 위한 서브시스템은 로그 파일에서 페이지 손상들을 해결하는 서브시스템을 포함하는 자동 데이터 신뢰성 시스템.

청구항 5

제1항에 있어서,

오류 및 이벤트 통지, 정책, 및 오류/이벤트 핸들링 알고리즘 기능의 그룹 중에서 적어도 하나의 기능을 추가, 삭제, 및 수정하기 위한 인터페이스를 더 포함하는 자동 데이터 신뢰성 시스템.

청구항 6

제1항에 있어서,

상기 DRS는 배경 스레드(background thread)로서 동작하는 자동 데이터 신뢰성 시스템.

청구항 7

데이터베이스 파일 시스템(DBFS)을 위한 데이터 저장소를 부착하고 온라인으로 불러오는 시스템으로서,

상기 데이터 저장소를 부착하는 서브시스템;

크래쉬 복구(crash recovery)를 실행하고, 상기 데이터 저장소를 온라인으로 불러오려고 시도하는 서브시스템 - 데이터 손상들의 세트는 페이지 레벨에서 발생하고, 제1 레벨의 복구는 손상된 페이지를 수리하기 위해 시도하고, 상기 손상된 페이지가 수리될 수 없으면 제2 레벨의 복구가 손상된 페이지를 재구성하거나 복원하려고 시도함 - ;

상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않은지를 판정하는 서브시스템;

상기 데이터 저장소와 연관된 데이터베이스가 트랜잭션면에서 일관적인지를 판정하기 위해 상기 데이터베이스를 평가하는 서브시스템; 및

상기 데이터베이스가 트랜잭션면에서 일관적이면, 상기 크래쉬 복구가 완료될 때까지 수리들을 지연시키고, 상기 데이터베이스가 트랜잭션면에서 일관적이지 않다면 지연 없이 수리들을 진행하는 서브시스템

을 포함하는 시스템.

청구항 8

제7항에 있어서,

페이지 레벨 복원을 시도하고, 상기 페이지 레벨 복원이 실패하면, 상기 실패가 인덱스 페이지 때문인지를 확인하고, 상기 실패가 인덱스 페이지 때문이라면, 상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않은 경우, 그리고 상기 데이터베이스가 트랜잭션면에서 일관적인 경우, 상기 인덱스 페이지를 재구성하는

서브시스템을 더 포함하는 시스템.

청구항 9

제7항에 있어서,

상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않은 경우, 그리고 상기 데이터베이스가 트랜잭션면에서 일관적이지 않은 경우에 상기 데이터베이스를 오프라인으로 하고, 비상 수리를 시도하는 서브시스템을 더 포함하는 시스템.

청구항 10

데이터베이스 파일 시스템(DBFS)을 위한 자동 데이터 신뢰성 시스템(DRS)을 위한, 컴퓨팅 장치에 의해 적어도 부분적으로 구현되는 방법으로서,

정책들의 세트를 개설하는 단계;

상기 정책들의 세트에 기초하여 데이터베이스 관리(DBA) 작업들을 수행하는 단계;

모든 페이지 유형들에 대해 페이지 레벨에서 데이터 손상들의 세트에 응답하는 단계;

손상된 페이지를 수리하려고 시도함으로써 제1 레벨의 복구를 수행하는 단계;

상기 손상된 페이지가 수리될 수 없으면 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 단계;

상기 DBFS와 연관된 데이터베이스가 트랜잭션면에서 일관적인지를 판정하기 위해 상기 데이터베이스를 평가하는 단계; 및

상기 데이터베이스가 트랜잭션면에서 일관적이라면 크래쉬 복구가 완료될 때까지 수리들을 지연시키고, 상기 데이터베이스가 트랜잭션면에서 일관적이지 않다면 지연 없이 수리들을 진행하는 단계;

를 포함하는 방법.

청구항 11

제10항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 단계는 인덱스 페이지

지 손상들을 해결하는 단계를 포함하는 방법.

청구항 12

제10항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 단계는 데이터 페이지 손상들을 해결하는 단계를 포함하는 방법.

청구항 13

제10항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 단계는 로그 파일에서 페이지 손상들을 해결하는 단계를 포함하는 방법.

청구항 14

제10항에 있어서,

상기 DRS는 오류 및 이벤트 통지, 정책, 및 오류/이벤트 핸들링 알고리즘 기능의 그룹 중 적어도 하나의 기능을 추가, 삭제, 및 수정하기 위한 인터페이스를 포함하는 방법.

청구항 15

제10항에 있어서,

상기 DRS는 배경 스레드(background thread)로서 동작하는 방법.

청구항 16

데이터베이스 파일 시스템(DBFS)을 위해 데이터 저장소를 부착하고 온라인으로 불러오는 방법으로서,

상기 데이터 저장소를 부착하는 단계;

크래쉬 복구를 실행하고, 상기 데이터 저장소를 온라인으로 불러오려고 시도하는 단계 - 데이터 손상들의 세트는 페이지 레벨에서 발생하고, 제1 레벨의 복구는 손상된 페이지를 수리하기 위해 시도하고, 상기 손상된 페이지가 수리될 수 없으면 제2 레벨의 복구는 손상된 페이지를 재구성하거나 복원하려고 시도함 - ;

상기 데이터 저장소를 온라인으로 불러오려고 시도하는 것이 성공적이지 않은지를 판정하는 단계;

상기 데이터 저장소와 연관된 데이터베이스가 트랜잭션면에서 일관적인지를 판정하기 위해 상기 데이터베이스를 평가하는 단계; 및

상기 데이터베이스가 트랜잭션면에서 일관적이라면 상기 크래쉬 복구가 완료될 때까지 수리들을 지연시키는 단계

를 포함하는 방법.

청구항 17

제16항에 있어서,

상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않고, 상기 데이터베이스가 트랜잭션면에서 일관적인 경우, 상기 수리들이 상기 크래쉬 복구가 완료될 때까지 지연되며,

상기 방법은, 페이지 레벨 복원을 시도하고, 상기 페이지 레벨 복원이 실패하면, 상기 실패가 인덱스 페이지 때문인지를 확인하고, 상기 실패가 인덱스 페이지 때문이라면, 상기 인덱스 페이지를 재구성하는 단계를 더 포함하는 방법.

청구항 18

제16항에 있어서,

상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않고, 상기 데이터베이스가 트랜잭션면에서 일관적이지 않은 경우,

상기 방법은, 상기 데이터베이스를 오프라인으로 하고, 비상 수리를 시도하는 단계를 더 포함하는 방법.

청구항 19

데이터 파일 시스템(DBFS)을 위한 자동 데이터 신뢰성 시스템(DRS)에 대한 컴퓨터-판독가능 명령어를 포함하는 적어도 하나의 실체적인(tangible) 매체를 갖는 컴퓨터-판독가능 저장 매체로서, 상기 컴퓨터-판독가능 명령어는,

정책들의 집합에 기초하여 데이터베이스 관리(DBA) 작업들을 수행하는 명령어;

모든 페이지 유형들에 대해 페이지 레벨에서 데이터 손상들의 세트에 응답하는 명령어;

손상된 페이지를 수리하려고 시도함으로써 제1 레벨의 복구를 수행하는 명령어;

상기 손상된 페이지가 수리될 수 없으면 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 명령어; 및

상기 DBFS와 연관된 데이터베이스가 트랜잭션면에서 일관적이지를 판정하기 위해 상기 데이터베이스를 평가하고, 트랜잭션면에서 일관적이라면 크래쉬 복구가 완료될 때까지 수리들을 지연시키고, 상기 데이터베이스가 트랜잭션면에서 일관적이지 않다면 지연 없이 수리들을 진행하는 명령어

를 포함하는 컴퓨터-판독가능 저장 매체.

청구항 20

제19항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 명령어는 인덱스 페이지 손상들을 해결하는 명령어를 포함하는 컴퓨터-판독가능 저장 매체.

청구항 21

제19항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 명령어는 데이터 페이지 손상들을 해결하는 명령어를 포함하는 컴퓨터-판독가능 저장 매체.

청구항 22

제19항에 있어서,

상기 손상된 페이지를 재구성하거나 복원하려고 시도함으로써 제2 레벨의 복구를 수행하는 명령어는 로그 파일에서 페이지 손상들을 해결하는 명령어를 포함하는 컴퓨터-판독가능 저장 매체.

청구항 23

제19항에 있어서,

오류 및 이벤트 통지, 정책, 및 오류/이벤트 핸들링 알고리즘의 기능의 그룹 중 적어도 하나의 기능을, 추가, 삭제, 및 수정하기 위한 인터페이스를 위한 명령어를 더 포함하는 컴퓨터-판독가능 저장 매체.

청구항 24

제19항에 있어서,

상기 DRS가 배경 스레드로서 동작하게 하는 명령어를 더 포함하는 컴퓨터-판독가능 저장 매체.

청구항 25

데이터베이스 파일 시스템(DBFS)을 위한 데이터 저장소를 부착하고 온라인으로 불러오는 컴퓨터-판독가능 명령어를 포함하는 적어도 하나의 실체적인(tangible) 매체를 갖는 컴퓨터-판독가능 저장 매체로서, 상기 컴퓨터-판

독가능 명령어는,

상기 데이터 저장소를 부착하는 명령어;

크래쉬 복구를 실행하고, 상기 데이터 저장소를 온라인으로 불러오려고 시도하는 명령어 - 데이터 손상들의 세트는 페이지 레벨에서 발생하고, 복구의 제1 레벨은 손상된 페이지를 수리하기 위해 시도하고, 상기 손상된 페이지가 수리될 수 없으면 복구의 제2 레벨은 손상된 페이지를 재구성하거나 복원하려고 시도함 - ;

상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않은지를 판정하는 명령어;

상기 데이터 저장소와 연관된 데이터베이스가 트랜잭션면에서 일관적인지를 판정하기 위해 상기 데이터베이스를 평가하는 명령어; 및

상기 데이터베이스가 트랜잭션면에서 일관적이면 상기 크래쉬 복구가 완료될 때까지 수리들을 지연시키는 명령어

를 포함하는 컴퓨터-판독가능 저장 매체.

청구항 26

제25항에 있어서,

페이지 레벨 복원을 시도하고, 상기 페이지 레벨 복원이 실패하면, 상기 실패가 인덱스 페이지 때문인지를 확인하고, 상기 실패가 인덱스 페이지 때문이라면, 상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않은 경우, 그리고 상기 데이터베이스가 트랜잭션면에서 일관적이고, 상기 수리들이 상기 크래쉬 복구가 완료될 때까지 지연되는 경우에, 상기 인덱스 페이지를 재구성하는

명령어를 더 포함하는 컴퓨터-판독가능 저장 매체.

청구항 27

제25항에 있어서,

상기 데이터 저장소를 온라인으로 불러오려는 시도가 성공적이지 않고 상기 데이터베이스가 트랜잭션면에서 일관적이지 않은 경우에, 상기 데이터베이스를 오프라인으로 하고, 비상 수리를 시도하는

명령어를 더 포함하는 컴퓨터-판독가능 저장 매체.

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

명세서

[0001] <관련 출원의 상호 참조>

[0002] 본 출원서는 본 명세서에서 그 전체내용이 참조되는, 2004년 5월 3일 출원된, 미국 특허출원 제10/837,932호에 우선권을 주장한다.

[0003] 본 출원서는 본 명세서에 그 전체내용이 병합되는 다음의 공통 양수된 출원서들에서 개시된 발명들의 주제와 관련된다: 미국 특허출원 제10/647,058호(2003년 8월 21일) "SYSTEMS AND METHODS FOR REPRESENTING UNITS OF INFORMATION MANAGEABLE BY A HARDWARE/SOFTWARE INTERFACE SYSTEM BUT INDEPENDENT OF PHYSICAL REPRESENTATION"; 미국 특허출원 제10/646,941호(2003년 8월 21) "SYSTEMS AND METHODS FOR SEPARATING UNITS OF INFORMATION MANAGEABLE BY A HARDWARE/SOFTWARE INTERFACE SYSTEM FROM THEIR PHYSICAL ORGANIZATION"; 미국 특허출원 제10/646,940호(2003년 8월 21일) "SYSTEMS AND METHODS FOR THE IMPLEMENTATION OF A BASE

SCHEMA FOR ORGANIZING UNITS OF INFORMATION MANAGEABLE BY A HARDWARE/SOFTWARE INTERFACE SYSTEM"; 미국 특허출원 제10/646,632호(2003년 8월 21일) "SYSTEMS AND METHODS FOR THE IMPLEMENTATION OF A CORE SCHEMA FOR PROVIDING A TOP-LEVEL STRUCTURE FOR ORGANIZING UNITS OF INFORMATION MANAGEABLE BY HARDWARE/SOFTWARE INTERFACE SYSTEM"; 미국 특허출원 제10/646,645호(2004년 8월 21일) "SYSTEMS AND METHODS FOR REPRESENTING RELATIONSHIPS BETWEEN UNITS OF INFORMATION MANAGEABLE BY A HARDWARE/SOFTWARE INTERFACE SYSTEM"; 미국 특허출원 제10/646,575호(2003년 8월 21일) "SYSTEMS AND METHODS FOR INTERFACING APPLICATION PROGRAMS WITH AN ITEM-BASED STORAGE PLATFORM"; 미국 특허출원 제10/646,646호(2003년 8월 21일) "STORAGE PLATFORM FOR ORGANIZING, SEARCHING, AND SHARING DATA"; 미국 특허출원 제10/646,580호(2003년 8월 21일) "SYSTEMS AND METHODS FOR DATA MODELING IN AN ITEM-BASED STORAGE PLATFORM".

기술분야

[0004] 본 발명은 일반적으로 데이터베이스 및 파일 시스템 정비에 관한 것으로서, 특히, 데이터 신뢰성(data reliability)을 보장하기 위한 자동 데이터베이스 및 파일 시스템 정비 및 수리에 관한 것이다. 본 발명의 다양한 양태들은 모든 데이터 페이지 유형들에 대해 데이터 페이지 레벨에서 데이터 손상에 응답하고 수리하는 것, 및 인덱스 페이지 손상(클러스터 되고 클러스터 되지 않음), 데이터 페이지 손상, 및 로그 파일의 페이지 손상을 포함하는, 하지만 이에 제한되지는 않으며, 다양한 시나리오들에 대한 복구(재구성 또는 복원 동작들을 포함함)에 관련된다.

배경기술

[0005] 클라이언트 데이터베이스 플랫폼들(예를 들어, 가정 및 비즈니스 데스크톱 컴퓨터들)은 서버 플랫폼들에서보다 훨씬 더 낮은 품질의 하드웨어를 사용하는 한편, 서버-클래스 하드웨어(제어기, 드라이버, 디스크 등)도 읽기 동작이 응용 프로그램이 데이터 저장소에 기록한 것을 리턴하지 않는 것과 같은 데이터 손상의 원인이 될 수 있다. 물론, 이것은, 클라이언트 머신이 기대하지 않은 전력 중단 때문에 쓰기 동작 중에 임의적으로 전력이 중단되어, 찢어진 페이지들과 잠재적 데이터베이스 손상으로 이어지는 증대된 가능성을 포함하는, 하지만 이에 제한되지는 않는, 다양한 이유로 클라이언트 데이터베이스 플랫폼들(서버 데이터베이스 플랫폼들에 반해)에서 명백히 더 많은 문제들이 있다. (전력 중단으로부터의 문제들을 완화시키기 위해 연속적 전력 공급기들을 서버 데이터베이스 시스템들이 사용하는 것은 더 일반적임) 물리적 저장 매체가 시간에 걸쳐 문자 그대로 마멸되는 경우, 매체 마멸은 데이터베이스 손상의 다른 소스이다. 또한, 신뢰성에 관한 관심의 다른 소스는 부주의하고(예를 들어, 버그) 악의적인(예를 들어, 바이러스) 모든 소프트웨어 오류들로 인한 손상들로부터 탐지와 복구이다.

[0006] 종래 데이터베이스들의 정비 및 수리는 데이터베이스 시스템들에 대한 잘 개발된 기술 세트 및 깊은 지식을 가진 데이터베이스 관리자, 또는 적어도 데이터베이스들에 익숙하고 정기적으로 사용하는 개인들에게 -데이터베이스 기술들에 대해 비교적 기술이 있는 사람들- 속한 일이었다. 한편, 운영 시스템들과 응용 프로그램들의 통상의 소비자들 및 비즈니스 최종 사용자들은 데이터베이스들과 거의 작업을 하지 않고, 데이터베이스 정비 및 수리 이슈들을 다루기에 잘 준비가 되어있지 않다.

[0007] 이들 2개의 그룹들 간의 분리된 레벨의 기술이 과거에는 크게 관련이 없었지만, 운영 시스템들을 위한 데이터베이스-구현된 파일 시스템 -본 명세서의 처음부분에 "상호참조" 섹션에서 식별된 관련된 미국특허 출원서들에 개시된 운영 시스템과 같은-은 이들 기술이 부족한 최종 사용자들이 그들이 주로 해결할 수 없는 데이터베이스 정비 및 수리 이슈들에 직면할 경우의 시나리오를 생성한다. 그러므로, 비즈니스/소비자 데이터베이스-구현된 운영 시스템 파일 시스템, 또는 간략하게 "DBFS(database file system)"는 손상을 탐지하고 그것의 데이터베이스를 트랜잭션에서 일관적인 상태로 복구할 수 있어야 하고, 복구불능의 데이터 유실인 경우들에서, DBFS는 데이터로의 항목 변경 단위가 관리되는 레벨에서(예를 들어, 항목-기반 DBFS를 위한 "항목" 레벨에서) 데이터 일관성을 보장해야 한다. 더욱이, 느린 위임 모드(commit mode)에서 디폴트로 실행하는 DBFS에 대해, 변칙 셔트다운(abnormal shutdown) 바로 전에 행해진 트랜잭션들의 영속성(durability)은 보장되지 않으므로, 설명되어야 하고, 교정되어야 한다.

[0008] 더욱이, 비즈니스/소비자 최종 사용자가 자동 DBFS 관리 및 복구로부터 크게 이득을 얻을 것인 한편, 데이터베이스 관리자들과 더 깊은 데이터베이스 기술을 가진 이들도 또한 일반 데이터베이스 정비 및 수리에 대한 기술 해결책으로부터 이득을 얻을 것이다. 데이터베이스 관리 분야에서 데이터베이스 툴들(예를 들어, SQL 서버 2000과 함께 제공된 데이터베이스 튜닝 어드바이저)을 사용하는 것은 일반적이지만, 이들 툴들은 직접 신뢰성을

해결하지는 않지만, 대신에 데이터베이스 백업이 관리되는 수단을 제공한다 -대부분-자동의 방식이 아닌, 특히 데이터베이스 백업을 이용할 수 없거나 다른 수리 이슈들이 발생할 때, 실제적 데이터베이스 관리자의 관여를 요구함-. 그러므로, 데이터베이스 신뢰성을 해결하기 위한 자동 해결책은 또한 데이터베이스 관리자와 다른 숙련된 데이터베이스 사용자들에게 이득이 될 것이다. 본 발명은 그런 해결책을 제공할 따름이다.

발명의 상세한 설명

[0009] 본 발명의 다양한 실시예들은 DBFS를 위한 데이터 신뢰성 시스템(data reliability system;DRS)에 관한 것이고, 여기서, DRS는 최종 사용자에게 의한 직접 관여가 없이 또는 관여가 거의 없이 데이터베이스 관리(database administration;DBA) 작업들을 자동으로 수행하는(그러므로, 기본적으로 최종 사용자에게 투명함) 프레임워크와 정책들(policies)의 세트를 포함한다. 여러 실시예들에서, DRS 프레임워크는 오류와 이벤트 통지, 정책, 및 오류/이벤트 핸들링 알고리즘을 DRS로 플러그인하는 메카니즘들을 구현한다. 더 구체적으로, 이들 실시예들에 대해, DRS는 배경에서 DBFS를 관리하고 수리하는 책임이 있는 배경 스레드(thread)이고, 그러므로, 최상위 레벨에서, DRS는 DBFS의 전체 정상상태(health)를 보호하고 관리한다.

[0010] 본 발명의 다양한 실시예들에서, DRS는 다음 특징들을 포함한다: (1) 모든 페이지 유형들에 대해 페이지 레벨의 데이터 손상에 응답하고 교정하는 것; 및 (2) (a) 인덱스 페이지 손상(클러스터되고 클러스터되지 않음); (b) 데이터 페이지 손상; 및 (c) 로그 파일의 페이지 손상에 대해 복구(재구성 또는 복원)의 제2 레벨을 시도하는 것.

[0011] 본 발명의 특정 실시예들은 다음과 같은 것들을 포함하는, 하지만 이에 제한되지는 않은, DRS를 위한 특정 기능을 더 포함한다: (i) 수리/복원 데이터 손상 경우들을 핸들링하는 것; 및 (ii) (iii) 시스템의 신뢰성과 이용성을 개선하는 것; 및 (iv) 필요하면 숙련된 제3자가 데이터베이스 또는 저장 엔진 문제들을 해결하도록 DRS 오류/이벤트 이력표(history table)를 유지하는 것.

실시예

[0024] 본 발명의 요지는 법적인 요건을 충족하도록 기술되었다. 그러나, 설명 그자체는 본 출원의 범위를 제한하고자 하는 것이 아니다. 오히려, 발명자는 본 발명의 요지가 다른 현재 또는 미래의 기술과 결합되어 본 명세서에 기재된 것과 유사한 다른 단계 또는 단계들의 조합을 포함하도록 다른 방식으로 실시됨을 고려한다. 더욱이, 용어 "단계"가 본 명세서에서 채용된 방법의 다른 요소를 나타내는데 사용된다고 할지라도, 용어는 개별 단계의 순서가 명백히 기술될 때까지 그리고 이러한 경우만을 제외하고 본 명세서에 개시된 여러 단계들 사이에서 임의의 특정 순서를 암시하는 것으로 해석되어서는 안된다.

[0025] 상술한 개요는 본 발명의 특징의 개관을 제공한다. 본 발명의 일 실시예의 상세한 설명이 뒤따른다. 이하 기재되는 여러 실시예에서, 본 발명의 특징들은 MICROSOFT SQL SERVER 데이터베이스 시스템(때때로 본 명세서에서 단순히 "SQL"로 언급됨)만으로 구현되거나, 또는 차세대 개인용 컴퓨터 운영 시스템(통상 짧게 얘기하면 "Windows Longhorn" 또는 "Longhorn"으로 언급됨)을 위한 MICROSOFT WinFS 파일 시스템에 통합되는 것으로 기술되고, 후자는 본 명세서 처음에 상호 참조된 특허 출원서들 중의 다수의 주제이다. 전술한 바와 같이, SQL SERVER는 MICROSOFT.NET CLR(common language runtime)을 통합하여, 관리된 코드가 SQL SERVER 데이터베이스의 데이터 저장소 상에서 동작하도록 기록 및 실행될 수 있게 한다. 이하 설명되는 실시예가 이런 컨텍스트에서 동작한다 할지라도, 본 발명이 SQL SERVER 제품에서의 구현에 결코 제한되지 않음을 이해할 것이다. 그 보다는, 본 발명은 데이터베이스 저장소, 예를 들어, 객체지향 데이터베이스 시스템 및 객체 관계 확장을 갖는 관계 데이터베이스 시스템, 에서 동작하도록 객체지향 프로그래밍 코드의 실행을 지원하는 임의의 데이터베이스 시스템에서 구현될 수 있다. 따라서, 본 발명은 이하 설명되는 특정 실시예에 제한되는 것이 아니라, 부착된 청구범위에 의해 정의된 본 발명의 정신 및 범위 내에 있는 모든 수정을 커버하려는 것으로 이해해야 한다.

[0026] 컴퓨터 환경

[0027] 본 발명의 많은 수의 실시예들은 컴퓨터상에서 실행될 수 있다. 도 1 및 다음의 논의는 본 발명이 구현될 수 있는 적당한 컴퓨팅 환경의 간략한 일반적 설명을 제공하고자 하는 것이다. 비록 요구되지 않았다 할지라도, 본 발명은 클라이언트 워크스테이션 또는 서버와 같은 컴퓨터에 의해 실행되는 프로그램 모듈들과 같은 컴퓨터 실행가능한 명령의 일반적인 컨텍스트로 기술될 것이다. 일반적으로, 프로그램 모듈은 특정 작업들을 수행하거나 특정 추상 데이터 유형들을 구현하는 루틴, 프로그램, 객체, 컴포넌트, 데이터 구조 등을 포함한다. 더욱이, 당업자들은 본 발명이 핸드 헬드 디바이스, 멀티 프로세서 시스템, 마이크로프로세서 기반 또는 프로그래밍가능

한 소비자 전자제품, 통신망 PC, 미니컴퓨터, 메인프레임 컴퓨터 등을 포함하는 다른 컴퓨터 시스템 구성으로 실시될 수 있음을 이해할 것이다. 본 발명은 또한 통신망을 통해 링크되는 원격 처리 디바이스에 의해 작업이 수행되는 분산 컴퓨팅 환경에서 실시될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈은 로컬 및 원격 메모리 저장 디바이스 모두에 위치한다.

[0028] 도 1에 도시된 바와 같이, 일반 목적 컴퓨팅 시스템의 예는 프로세싱 유닛(21), 시스템 메모리(22), 및 시스템 메모리(22)를 포함하는 여러 시스템 컴포넌트들을 프로세싱 유닛(21)에 연결하는 시스템 버스(23)를 포함하는 종래의 개인용 컴퓨터(20) 등을 포함한다. 시스템 버스(23)는 메모리 버스 또는 메모리 제어기, 주변 버스, 및 각종 버스 아키텍처들중 임의의 하나를 이용하는 로컬 버스를 포함하는 여러 유형의 버스 구조들 중의 임의의 하나일 수 있다. 시스템 메모리는 ROM(read only memory)(24) 및 RAM(random access memory)(25)을 포함한다. 기동시와 같이, 개인용 컴퓨터(20) 내의 요소들 간의 정보 전달에 도움을 주는 기본 루틴들을 포함하는 BIOS(basic input/output system)(26)은 ROM(24)에 저장된다. 개인용 컴퓨터(20)는, 도시 안된, 하드 디스크로부터 판독 및 기록하기 위한 하드 디스크 드라이브(27), 제거가능한 자기 디스크(29)로부터 판독 및 기록하기 위한 자기 디스크 드라이브(28), 및 CD-ROM 또는 다른 광매체와 같은 제거가능한 광 디스크(31)로부터 판독 및 기록하기 위한 광 디스크 드라이브(30)를 더 포함한다. 하드 디스크 드라이브(27), 자기 디스크 드라이브(28), 및 광 디스크 드라이브(30)는 하드 디스크 드라이브 인터페이스(32), 자기 디스크 드라이브 인터페이스(33) 및 광 디스크 드라이브 인터페이스(34)에 의해 각각 시스템 버스(23)에 연결된다. 드라이브들 및 이들과 관련된 컴퓨터 판독가능한 매체는 컴퓨터 판독가능한 명령, 데이터 구조, 프로그램 모듈 및 개인용 컴퓨터(20)용 다른 데이터의 비휘발성 저장 장치를 제공한다. 본 명세서에 개시된 예시적인 환경이 하드 디스크, 제거가능한 자기 디스크(29) 및 제거가능한 광 디스크(31)를 채용한다 할지라도, 당업자들은, 자기 카세트, 플래시 메모리 카드, 디지털 비디오 디스크, 버블리 카트리지, RAM, ROM 등과 같은 컴퓨터에 의해 액세스가능한 데이터를 저장할 수 있는 컴퓨터 판독가능한 매체의 다른 유형들이 예시적인 컴퓨팅 환경에 또한 사용될 수 있음을 이해해야 한다.

[0029] 많은 수의 프로그램 모듈들은 운영 시스템(35), 하나 이상의 응용 프로그램들(36), 기타 프로그램 모듈들(37) 및 프로그램 데이터(38)를 포함하는 하드 디스크, 자기 디스크(29), 광 디스크(31), ROM(24) 또는 RMA(25) 상에 저장될 수 있다. 사용자는 커맨드 및 정보를 키보드(40) 및 포인팅 디바이스(42)와 같은 입력 디바이스들을 통해 개인용 컴퓨터(20)로 입력한다. 다른 입력 디바이스들(도시되지 않음)은 마이크로폰, 조이스틱, 게임 패드, 위성 접시, 스캐너 등을 포함할 수 있다. 이들 및 다른 입력 디바이스들은 시스템 버스를 통해 접속되는 직렬 포트 인터페이스(46)를 통해 프로세싱 유닛(21)에 종종 연결되지만, 병렬 포트, 게임 패드, 또는 USB(universal serial bus)와 같은 다른 인터페이스들에 의해서도 연결될 수 있다. 모니터(47) 또는 다른 유형의 디스플레이 디바이스는 비디오 어댑터(48)와 같은 인터페이스를 통해 시스템 버스(23)에 또한 연결된다. 모니터(47)에 추가하여, 개인용 컴퓨터들은 전형적으로 스피커 및 프린터와 같은 다른 주변 출력 디바이스들(도시 안됨)을 포함한다. 도 1의 예시적인 시스템은 또한 호스트 어댑터(55), SCSI(small computer system interface)버스(56), 및 SCSI 버스(56)에 연결된 외부 저장 디바이스(62)를 포함한다.

[0030] 개인용 컴퓨터(20)는 원격 컴퓨터(49)와 같은 하나 이상의 원격 컴퓨터들에 대한 논리 연결을 이용하는 통신망 환경에서 동작할 수 있다. 원격 컴퓨터(49)는 다른 개인용 컴퓨터, 서버, 라우터, 통신망 PC, 피어 디바이스, 또는 다른 공통 통신망 노드일 수 있으며, 단지 메모리 저장 디바이스(50)만이 도 1에 예시된다 할지라도, 전형적으로 개인용 컴퓨터(20)에 관한 상술한 요소들중 많은 수 또는 모두를 포함한다. 도 1에 도시된 논리 연결들은 LAN(local area network)(51) 및 WAN(wide area network)(52)을 포함한다. 이런 통신망 환경은 사무실, 기업체 전반 컴퓨터 통신망, 인트라넷 및 인터넷에서 일반적이다.

[0031] LAN 통신망 환경에서 사용될 때, 개인용 컴퓨터(20)는 통신망 인터페이스 또는 어댑터(53)를 통해 LAN(51)에 연결된다. WAN 통신망 환경에서 사용될 때, 개인용 컴퓨터(20)는 전형적으로 모뎀, 또는 인터넷과 같은 WAN(52)을 통해 통신을 구축하는 다른 수단을 포함한다. 내부 또는 외부일 수 있는 모뎀(54)은 직렬 포트 인터페이스(46)를 통해 시스템 버스(23)에 연결된다. 통신망 환경에서, 개인용 컴퓨터(20), 또는 그 일부, 에 관해 묘사된 프로그램 모듈은 원격 메모리 저장 디바이스에 저장될 수 있다. 도시된 통신망 연결이 예시적이며, 컴퓨터들 간에 통신 링크를 구축하는 다른 수단이 사용될 수 있음을 이해할 것이다.

[0032] 본 발명의 수많은 실시예들이 컴퓨터화된 시스템들에서 특히 매우 적합한 것으로 고려된다 할지라도, 본 명세서는 이런 실시예들만으로 본 발명을 제한하고자 하는 것이 아니다. 이와 반대로, 본 명세서에서 사용되는 바와 같이, "컴퓨터 시스템"이란 용어는, 디바이스가 속성상 전자, 기계, 논리 또는 가상적인지에 무관하게, 정보를 저장 및 처리할 수 있고 그리고/또는 디바이스의 행위 또는 실행을 제어하기 위하여 저장된 정보를 이용할 수

있는 임의의 모든 디바이스들을 포함하려고 의도된다.

[0033] DRS의 개관

[0034] 본 발명의 몇몇 실시예들에 있어서, DRS는 데이터베이스를 배경에서 유지 및 보수하여 DBFS의 일반 정상상태를 보호하는 쓰레드이다. 도 2는 DBFS 내의 DRS의 구조를 나타내는 블록도이다. 도면에서, 운영 시스템 레벨 서비스를 복수의 응용 프로그램들(212, 214, 216)에 제공하는 운영 시스템(202)은 지속적 데이터 저장소(232)에 논리적으로 연결된 DBFS(222)를 포함한다. 또한, 운영 시스템(202)은, 지속적 데이터 저장소(232) 내의 복수의 페이지(234, 236, 238)에서 페이지 오류(a page error)(240)가 발견될 때마다 DBFS(222)에 의해 호출되는 DRS(242)를 포함하며, 그 다음, DRS(242)는 페이지 오류(240)에 응답하여 수리 동작을 수행한다.

[0035] 본 발명의 다양한 실시예들에 대해, DRS는 다음 특징들을 포함할 것이다: (1) 모든 페이지 유형들에 대해 페이지 레벨에서 데이터 손상에 응답하고 교정하는 것; 및 (2) (a) 인덱스 페이지 손상(클러스터되고 클러스터되지 않음); (b) 데이터 페이지 손상; 및 (c) 로그 파일의 페이지 손상을 위한 복구(재구성 또는 복원)의 제2 레벨을 시도하는 것. 본 발명의 특정 실시예들은 다음의 것들을 포함하는, 하지만 이에 제한되지는 않는, DRS를 위한 특정 기능을 더 포함한다: (i) 수리/복원 데이터 손상 경우들을 핸들링하는 것; (ii) 시스템의 신뢰성과 이용성을 개선하는 것; 및 (iii) 필요하면 제3자가 데이터베이스 또는 저장 엔진 문제들을 해결하도록 DRS 오류/이벤트 이력표를 유지하는 것.

[0036] 본 발명의 특정 실시예들은 DRS가 확장가능하여, DBFS가 해제된 후에 복구 정책들 및 탐지 메카니즘들이 업데이트될 수 있음을 제공한다. 여러 실시예들은 DBFS 데이터베이스가 온라인으로 유지되는 동안 수리를 실행하는 DRS에 관한 것이다. 여전히 다른 실시예들은 DBFS 저장소로의 전체 액세스(즉, 시스템관리자 권한)를 가지고 실행하는 것에 관한 것이다. 여전히 다른 실시예들은 실시간으로 실패들을 탐지하고 반응하는 능력을 가질 것이다. 여러 실시예들에서, DRS 수리는 데이터로의 변경 단위가 관리되는 레벨에서(즉, 항목-기반 DBFS에 대한 "항목" 레벨에서) 트랜잭션이 될 것이다. 마지막으로, 다양한 실시예들에서, 수리들은 항목을 완전 복구하거나 또는 그것은 그것의 변경들을 취소할 것이고, 재부팅이 그 프로세스를 통해 중간에 실행되더라도 DRS는 복구/복원 작업을 계속하는 능력을 가질 것이다.

[0037] 본 발명의 여러 실시예들에서, DRS는 SQL 이벤트들에 가입하여, SQL이 일반 이벤트를 시작하면, DRS는 그것을 인터셉트하여 반응할 것이다(823/824 이벤트들을 포함하지만, 이에 제한되지는 않는). 추가로, 본 발명의 다른 양태는 DRS가 특정하게 핸들링하는 오류 조건들에 대해 DRS-고유의 이벤트들을 전송하기 위해 수정되는 데이터베이스 엔진에 대한 것이다.

[0038] 본 발명의 다양한 실시예들에 대해, DBFS가 디스크로부터 페이지들을 읽거나 쓸 때마다 손상들이 탐지될 것이고, 이 경우, SQL은 손상의 유형에 따른 오류들 중의 하나를 생성할 것이고, 또한 특정 DRS 이벤트들을 시작하여 그것에게 특정 오류 조건들을 통지할 것이다. DRS는 이들 오류들을 수신하여, 그들을 프로세싱을 위한 인입 큐(incoming queue)에 배치시킬 것이다.

[0039] 본 발명의 여러 실시예들에서, 페이지가 손상되었는지를 확인하는 것은 다음의 것들을 포함하는, 하지만 이에 제한되지는 않는, 다양한 수단들에 의해 성취된다: (a) 페이지에 대해 체크섬(checksum)을 검사하는 것, 체크섬이 무효하면, 페이지는 손상되었다고 고려된다, 또는 (b) LSN(log serial number)이 로그 파일의 끝 이상인지를 검사하는 것(여기서, LSN은 각 트랜잭션마다 증가되는 정수이어서, 로그의 최종 트랜잭션이 LSN 432이고 더 큰 LSN을 가진 페이지가 발견되면, 쓰기 장애 오류가 발생해야 한다). 이 관점에서, DBFS의 동작들에 영향을 미칠 수 있는 4 가지 주요 유형들의 페이지 손상들(버그 등과 같은 다른 소스들에 추가하여)이 존재하고, 이들 4 가지 유형들은 찢어진 페이지, 매체 마멸, 하드웨어 장애, 및 쓰기 장애를 포함한다. 찢어진 페이지는 데이터 페이지가 항목별로 정확하게 쓰이지 않았을 때 발생하므로, 쓰기 동안 단지 페이지의 섹터들 중의 일부만이, 예를 들어, 전력 장애 또는 섹터 쓰기 장애와 같은, 장애 이벤트 전에 디스크에 기록되므로 페이지의 임의의 부분이 손상될 수 있다. 매체 마멸은 데이터 페이지 비트들이 물리적 매체 마멸에 의해 손상될 때 발생한다. 하드웨어 장애는 버스, 제어기, 또는 하드 디스크 디바이스에 관련된 다양한 이유들에 대해 발생할 수 있다. 쓰기 장애에 대해, 이들 오류들은 IDE 드라이브들이 디스크에 쓰는 순서를 보장할 수 없고, 특히 IDE 드라이브는 쓰기-캐싱(write-caching)을 가동하여(턴온됨(turned on)), 데이터 저장소에서의 쓰기는 장애가 발생할 수 있음이 가능하다는 사실로부터 생겨난다. 쓰기 장애의 부분적 연속물이 발생하지만, 예를 들어, 전력 중단으로 인터럽트(interrupt)되면, 예를 들어, 연관된 로그 엔트리들이 기록되기 전에 데이터 페이지가 디스크에 기록되는 것과 같은 여러 오류들이 발생할 수 있다. 장애 오류들이 데이터 페이지들의 LSN(log sequence numbers)을 검사하여 탐지될 수 있는 한편, 모든 페이지를 읽는 것보다 더 쉽게 이것을 수행하는 방법이 없다.

[0040] **페이지 클래스들**

[0041] 본 발명의 목적으로, 모든 페이지들은 다음 페이지들의 클래스들 중의 하나에 따라 분류된다:

[0042] • 데이터 페이지들: 데이터 페이지는, 클러스터된 인덱스 종단(leaf) 페이지들을 포함하는, 사용자 데이터를 가진 임의의 페이지인 것으로 고려된다.

[0043] • 인덱스 페이지들: 이들 페이지들은 단지 인덱스 정보를 포함하고, 그들은 클러스터되지 않은 인덱스 페이지 및 클러스터된 인덱스의 비종단(non-leaf) 페이지들을 모두 포함한다.

[0044] • 시스템 페이지들: 이들 페이지들은 GAM, SGAM, 및 부트 페이지들을 포함하고, DRS가 이들 페이지들을 위해 특정한 복원 지원을 갖지 않을 수 있지만, DRS는 이들 페이지들에 페이지 레벨 복원을 시도할 것이다; 임의의 이벤트에서, 페이지 레벨 복원이 실패하면, DRS는 비상 복원(본 명세서에서 나중에 논의됨)을 시도한다.

[0045] • 복구불능 페이지들: PFS(Page Free Space) 페이지 또는 5개의 시스템 표들(Sysrowsetcolumns, Sysrowsets, Sysallocunits, Syshobtcolumns, Syshobts)로부터의 페이지들은 이 클래스의 페이지들을 포함하고, DRS는 이 경우에 전체 데이터베이스를 복원할 것이다.

[0046] • 로그 페이지들: 이들은 트랜잭션 로그에 속하는 페이지들이고, DRS는 그들이 손상될 때(본 명세서에서 나중에 논의됨) 비상 수리를 시도할 것이다.

[0047] **손상 카테고리들**

[0048] 본 발명의 여러 실시예들에서, 손상이 탐지될 때 정의되는 데이터 페이지 손상들의 3개의 구별되는 카테고리들을 리졸브(resolve)하기 위해 DRS는 디자인되고, 3개의 카테고리들은 다음을 포함한다: (1) 데이터베이스 부착 동안; (2) 정상 온라인 동작들 동안; 및 (3) 트랜잭션 롤백(rollback) 동안.

[0049] 데이터베이스 부착 손상 탐지

[0050] 도 3a는 데이터베이스 FPM에 의해 DBFS로 데이터 저장소들의 부착을 설명하는 블록도이다. 도 3b는 손상된 페이지들이 데이터베이스 부착 동작 동안 탐지되고 교정되는 프로세스를 설명하는 흐름도이다. 일반적으로, 데이터베이스 파일 특성 관리자(DBFPM)(302)는 DBFS 저장소들(304, 306, 및 308)의 부착과 분리를 관리한다. 단계(352)에서, DBFPM(302)가, 예를 들어 DBFS 저장소(304)와 같은 저장소를 부착할 때, 트랜잭션 로그(314)에서의 크래쉬의 시간에 임의의 활성화된 트랜잭션들(324)이 존재하는지(예를 들어, 임의의 완료되지 않은 트랜잭션들)를 단계(356)에서 판정하기 위해 SQL은 단계(354)에서 크래쉬 복구를 실행하고, 만약 그렇다면, 단계(358)에서, SQL은 계속하기 전에 데이터베이스의 크래쉬 복구를 수행한다. 크래쉬 복구 동안, SQL은 정상적으로는 (a) 최종 검사점 후에 트랜잭션 로그를 분석하고, (b) 디스크에 기록된 것으로 발견되지 않는 로그의 임의의 동작들을 재실행(redo)하고, 및 (c) 완료되지 않은 임의의 트랜잭션들을 실행취소(undo)할 것이다. 단계(360)에서, 복구 동안 발견된 오류가 없으면, 데이터베이스는 단계(362)에서 성공적으로 부착된다. 그러나, 단계(360)에서 크래쉬 복구 동안 오류가 발견되면, SQL은 단계(364)에서 데이터베이스 부착을 실패할 것이다.

[0051] 이 "데이터베이스 부착" 시나리오는 그것이 저장소가 부착될 때마다 호출되므로 중요하다. 저장소들은 운영 시스템(예를 들어, 윈도우즈 운영 시스템)이 시작될 때마다, 및 외부 드라이브들(예를 들어, 피어와이어, USB 등)이 컴퓨터로부터 부착되거나 분리될 때마다 부착된다. 데이터베이스 부착 시나리오들은 크래쉬 복구 및 그러므로 찢어진 페이지들(불완전한 기록)의 탐지를 호출하고, 그러므로 사용자가 물리적으로 하드웨어를 제거하는 잠재성 때문에 DRS가 이 경우를 핸들링하는 것이 바람직하다.

[0052] 그러나, 본 발명의 다양한 실시예들에 따라 DSR을 사용할 때, 도 4a의 흐름도에 의해 도시된 바와 같이 행위는 약간 다르다. 먼저, 단계(402)에서, DBFPM은 DBFS 데이터베이스들을 부착하고, 단계(404)에서, SQL은 크래쉬 복구를 실행하고, 단계(408)에서, 데이터베이스를 온라인으로 불러오는 시도를 한다. 데이터베이스가 단계(410)에서 크래쉬 복구로부터의 오류가 없이 부착되면, SQL은 성공을 리턴할 것이고, 데이터베이스는 단계(452)에서 (온라인) 부착될 것이다. 그러나, 오류들이 있으면, 단계(412)에서, DRS는 데이터베이스가 트랜잭션에서 일관적인지를 판정하기 위해 데이터베이스를 평가할 것이다.

[0053] 단지 데이터베이스가 트랜잭션 롤백 동안에 실패를 경험했으면, 즉, 물리적 또는 논리적 실행취소(undo) 오류 - 또는 크래쉬 복구 동안 알지 못하는 오류가 발생되었으면-, 데이터베이스는 일관적이지 않다. 데이터베이스가

단계(412)에서 트랜잭션에서 일관적이면, 단계(414)에서, DRS는 복구 동작이 완료될 때까지 수리를 지연할 것이다 -부연하면, DRS는 데이터베이스 상태를 데이터베이스 부착에 대해 일관적이라고("성공") 선언할 것이고, 그 다음, 복구가 완료된 후에 탐지된 손상들에 대한 수리를 시작하기 위해 기다리고, 데이터베이스는 크래쉬 복구와 임의의 충돌을 피하기 위해 단계(452)에서 부착된다.

[0054] 일단 복구가 완료되고 부착이 단계(452)에서 성공적이면, 단계(478)에서 지연된 수리들이 존재하면, 단계(480)에서 DRS는 수리들을 시작할 것이고, 단계(482)에서, 먼저 손상에 대한 페이지 레벨 복원을 호출하려고 시도할 것이다. 페이지 레벨 복원이 실패하거나, 또는 단계(484)에서 불가능하면, 단계(486)에서, DRS는 그것이 인덱스 페이지의 실패 때문인지를 확인할 것이고, 만약 그렇다면, DRS는 단계(488)에서 인덱스를 재구성할 것이다. 한편, 오류가 단계(490)에서 데이터 페이지에 존재하면, DRS는 단계(492)에서 데이터 페이지 복구를 시도할 것이다. 그러나, DRS가 데이터베이스가 일관적이지 않거나 단계(490)에서 알려지지 않은/지원되지 않은 오류를 가진다고 판정하면, DRS는 즉시 데이터베이스를 오프라인으로 만들고(그러므로, 복구 동작을 종료함), 단계(494)에서 데이터베이스를 "서스펙트(suspect)"로서 표시를 할 것이고, 그 다음, 단계(496)에서, 비상 수리를 시도할 것이다.

[0055] 도 4b는 데이터베이스 부착 동안 DRS에 의해 핸들링되는 오류들을 설명하는 표이다.

[0056] 온라인 동작 손상 탐지:

[0057] 도 5a는 쿼리 엔진이 본 발명의 다양한 실시예들에 대한 오류를 대면하는 온라인 동작들에서 DRS의 사용을 설명하는 흐름도이다. 이 시나리오는, 최종 사용자가 DBFS에 대해 적당히 복잡한 쿼리를 실행하기 위해 운영 시스템 셸을 사용할 때 및 그 쿼리의 실행 동안과 같은, DBFS의 정상 실행 시간 사용 동안의 것이고, 쿼리 엔진은 오류를 보고한다. 이들 온라인 동작 상황들에서, 오류가 발생했을 때, 단계(502)에서, DRS는 트랜잭션을 취소하여 그 오류를 고치려는 시도를 한다. 단계(504)에서, DRS는 먼저 페이지 레벨 복원을 시도할 것이다. 그러나, 페이지 레벨 복원이 불가능하거나 단계(506)에서 실패하면, DRS는 페이지 유형이 단계(508)에서 실패하는지를 판정할 것이다. 단계(510)에서, 그것이 인덱스 페이지의 실패이면, DRS는 단계(512)에서 인덱스 복구를 시도할 것이다. 단계(514)에서 실패가 데이터 페이지에 있으면, DRS는 단계(516)에서 데이터 페이지 복구를 시도할 것이다. 실패가 단계(518)에서 시스템이나 로그 페이지 오류 때문이면, DRS는 단계(520)에서 비상 수리를 시도할 것이다. 실패가 다시 단계(518)에서 복구불능의 오류(5개의 시스템 표, 또는 PFS 페이지)이면, DFS는 단계(522)에서 전체 데이터베이스를 자동 복구하거나, 또는 다른 경우에, 최종 사용자가 전체 데이터베이스를 복구하도록 촉구할 것이다.

[0058] 도 5b는 온라인 동작들 동안 DRS에 의해 핸들링되는 오류들을 설명하는 표이다.

[0059] 트랜잭션 롤백 손상 탐지

[0060] 오류가 트랜잭션 롤백 동안 발생하면, DRS는 데이터베이스를 오프라인으로 하고, 그것을 서스펙트로 표시하고, 및 크래쉬 복구를 호출하기 위해 데이터베이스를 재시작할 것이다. 그 다음, 프로세스는 본 명세서에서 앞서 개시된 크래쉬 복구 손상 탐지를 위한 방법을 재호출할 것이다.

[0061] **복구 기술들**

[0062] 당업자들에게 잘 알려지고 쉽게 이해되는 바와 같이, 트랜잭션 일관성은 단지 위임된 데이터만 디스플레이하는 것(또는 이용가능하도록 하는 것), 및 단지 정확한 데이터를 위임하는 것(지속적 저장소에 기록하는 것 및 트랜잭션을 로깅(logging)하는 것)에 관한 것이다. 이 관점에서, 데이터베이스는 크래쉬 복구 동안 모든 재실행 및 실행취소가 존중되는 한, 트랜잭션에서 일관적이다. 그러므로, 다수의 유형들의 손상이 데이터 및 인덱스 페이지들에 존재할 수 있지만, 여전히 트랜잭션에서 일관적일 수 있다.

[0063] 본 발명의 다양한 실시예들은 이들 트랜잭션들의 다수를 온라인으로 수행하는 것은 종종 바람직하지 않음을 인식할 것이고, 그러므로, 이들 실시예들은 데이터베이스가 오프라인인 동안 모든 수리들을 수행하려고 시도한다.

[0064] SQL 오류가 발생한 때, DRS는 페이지 ID와 데이터베이스 ID만을 수신할 것이다. 그 정보로부터, DRS는 페이지를 조사하여 그것이 확실히 무슨 유형의 페이지인지를 알아낼 것이다. 이들 실시예들은 DRS가 손상 때문에 유실된 DBFS 항목들을 복구하려는 시도를 하는 경우, 데이터 페이지 및 인덱스 손상에 대한 복구 메카니즘들을 가진다. 그러나, 데이터베이스에서 GAM, SGAM, PFS, 부트 페이지 등을 포함하여 유실될 수 있는 다수의 다른 유형들의 페이지들이 존재한다. 통상적 DBFS들이 이들 유형들의 페이지들에 대해 특정 백업 및 복구 메카니즘들을 갖지 않는 한편, DRS는 페이지 레벨 복원을 호출하려고 시도할 것이다.

- [0065] 페이지가 손상되었으므로, DRS는 페이지 레벨 복원을 시도할 것이다. 위임되지 않은 임의의 데이터가 유실될 것임을 사용자에게 보장할 수 있기 때문에 페이지 레벨 복원은 무반응일 것이다. 동일 기술이 임의의 표로부터의 페이지들에 적용될 것이고, 복원되는 페이지나 표의 유형에 무관하게 방법은 변경하지 않는다. 페이지 레벨 복원은 단지 페이지가 가장 최신의 스냅샷에서 존재하면 발생할 것이다. 또한 유효한 트랜잭션 로그를 이용할 수 있어야 한다. 페이지가 가장 최신의 스냅샷에 있지 않으면, 당업자들에게 알려지고 이해되는 항목 레벨 복원 기술들을 사용하여 페이지를 복구해야 한다(백업 동작으로부터의 복원과 같은).
- [0066] 페이지가 가장 최신 스냅샷에 존재하고 유효한 트랜잭션 로그가 있으면, 도 6에 도시된 바와 같이, DRS는 페이지를 복원하기 위해 다음 동작들을 수행할 것이다: (1) 단계(602)에서, 손상된 페이지의 페이지 ID를 결정하고; (2) 단계(604)에서, DBFS의 저장소의 가장 최신 스냅샷으로부터 손상된 페이지를 찾고 복사하고; (3) 단계(606)에서, 그 페이지에 적용하는 트랜잭션들을 롤링 포워드(rolling forward)하여 페이지에 온디스크(on-disk) 트랜잭션 로그를 적용하고; (4) 단계(608)에서, 온라인 데이터베이스에 복원된 페이지를 적용하고; 및 (5) 단계(610)에 복원되는 페이지에 대한 지연된 트랜잭션들이 존재하면, 단계(612)에서 데이터베이스를 재시작해야 할 것이고(그러므로, 크래쉬 복구가 실행되어 지연된 트랜잭션들을 없앴), 그렇지 않으면 동작들은 단계(614)에서 계속한다.
- [0067] 전술된 스냅샷에 대해, 당업자들에게 의해 이해될 것처럼, VSS(Volume Shadowcopy Service)는 특정 볼륨들(예를 들어, NTFS 볼륨들)의 시간 스냅샷들에서 포인트를 관리하기 위한 방식을 제공한다. VSS 스냅샷들이 쓰기에 복사를 사용하여 볼륨의 윈도우 복사를 관리한다, 즉, 디스크 페이지가 수정될 때마다, 그 페이지의 이전-이미지가 가장 최신 스냅샷의 저장 지역에 기록된다. 타임워프(TimeWarp)가 머신에서 가동될 때 -이것은 특정 DBFS와 그들의 대응하는 운영 시스템들에 디폴트일 수 있음-, 스냅샷들을 하루에 두 번의 디폴트 비율로 찍혀지고, 최대 63 스냅샷들이 관리된다. 스냅샷에 저장된 이전 이미지들을 사용하기 위해, 스냅샷이 찍힌 시간으로부터 현재 시간까지의 로그가 관리되고, 로그 절단(log truncation)은 타임워프 스냅샷 시간에만 발생하여서, 항상 최종 스냅샷으로부터 이용가능한 로그가 존재하도록 한다. 페이지 손상이 존재하는 경우, 페이지는 가장 최근의 스냅샷에서 이용가능하고, 로그는 그것을 복원할 그 페이지에 현재 지점까지 스냅샷으로부터 재생될 수 있다.
- [0068] 타임 워프(Time Warp) 스냅샷들 동안, 스냅샷의 DBFS 저장소가 복구된다. 이것은 스냅샷 데이터베이스의 복구에 의해 터치된 페이지들에 대해 페이지 레벨 복원이 불가능하도록 한다. 이 문제를 해결하기 위해, SQL 서버 시점(viewpoint)이 스냅샷 볼륨에 대해 데이터베이스에 취해지고, 그 다음 데이터베이스가 복구된다 -즉, DRS는 페이지 레벨 복원을 위해 시점이 항상 사용될 수 있는 스냅샷 볼륨 및 페이지들에서 데이터베이스에 복구되지 않은 시점을 가짐-.
- [0069] 시스템, 로그, 또는 알려지지 않은 페이지 수리에 대해 -즉, 로그 손상이 발생하거나 또는 DRS가 교정할 수 없는(예를 들어, 데이터 또는 인덱스) 실패들이 존재하면, DRS는 사용자에게 다음 옵션들을 제공할 것이다: (a) 전체 데이터베이스(저장소)를 복원하는 것; 또는 (b) 비상 모드에 데이터베이스를 복구하는 것.
- [0070] 비상 모드의 데이터베이스를 수리하기 위해, 손상-트랜잭션 로그 및 복구불능 데이터베이스 상황들로부터 복구하는 DBCC의 새 능력을 인식하는 것이 우선 중요하다. 그 다음, 데이터베이스가 복구될 수 없고, 사용할 수 있는 백업이 없으면, 도 7에 도시된 다음 액션들의 세트는 본 발명의 특정 DRS 실시예들에서 데이터베이스를 온라인으로 다시 불러올 것이다: (a) 단계(702)에서, 데이터베이스를 비상 모드로 세트하고; (b) 단계(704)에서, (i) 과거 오류들을 데이터베이스 복구가 처리하도록 강제하고(로그로부터 가능한 많은 데이터를 얻지만 트랜잭션에서 비일관적 데이터베이스를 남김), (ii) 손상 로그 파일들을 던져버리고 새 것들을 생성하고, (iii) 전체 데이터베이스 수리를 실행하여 데이터베이스를 구조적으로 일관적인 상태로 가져오는(롤백되거나 수행취소될 수 없는 '항목적인' 일방 동작, 이것은 수동으로 파일들을 편집하지 않고 그런 상황에서 데이터베이스를 복구하는 유일하게 가능한 방식임) 비상 모드에서 특별 의미를 갖는 'DBCC CHECKDB(database, REPAIR_ALLOW_DATA_LOSS)'를 실행하고; 및 (c) 이제 데이터베이스는 물리적으로 일관적이고, DSR은 단계(706)에서 전체 저장소에 CC를 실행한다. 이들 단계들의 성공적 실행은 데이터(항목-기반 DBFS의 항목들)가 일관적일 것을 보장해야 하지만, 이것은 특정 응용 프로그램들이 트랜잭션으로 일관적이지 않을 것임을 의미할 것이다.
- [0071] 2개의 유형들의 인덱스 페이지들이 존재한다: 클러스터되지 않은 페이지들 및 클러스터된 비종단 페이지들. 데이터를 가진 인덱스 페이지들(클러스터된 인덱스 종단 페이지들)은 데이터 페이지들로서 고려된다. 이것을 염두에 두고, 복구가능한 인덱스 실패들에 대해, DRS는 오프라인 인덱스 재구성을 사용하여 인덱스를 수리하려는 시도를 한다(여기서, 데이터베이스는 이 수리 동안 온라인일 것인 한편, 단지 인덱스만이 오프라인일 것이다). 수리가 실패하면, DRS는 인덱스를 드롭(drop)하고, 인덱스를 재생성하려는 시도를 한다. 그 다음, 그것이 또한

실패하면, DRS는 인덱스를 모두 드롭하거나, 또는 다른 경우에, 인덱스를 가동중지하여, 나중에 재구성하려는 시도를 할 것이다(및 아마도 성공할 때까지 무한정 그것을 수행할 것임).

[0072] 데이터 페이지 복구에 대해, DRS가 핸들링되는 오류들 중의 하나를 수신하고 페이지가 데이터 페이지임을 판정하면, 그것은 복구를 시도할 것이다. 이것은 SQL 서버 동작의 임의의 상태(DB 부착, 온라인 동작, 및 롤백) 동안 발생할 수 있다. 도 8에 도시된 바와 같이, DRS가 단계(802)에서 데이터 페이지 오류를 탐지하면, 단계(806)에서 실패하는 단계(804)에서의 페이지 레벨 복원을 시도한 후에, DRS는 다음을 수행할 것이다: (a) 단계(808)에서, 손상된 페이지의 페이지 ID를 수신하고 이전에 논의된 것처럼 페이지 유형을 판정하여(데이터 페이지 손상이 주어지면, DRS는 페이지를 액세스하려고 시도하는 다른 트랜잭션들을 해결할 필요는 없음) 손상된 페이지의 유형을 판정하고; (b) 단계(810)에서, 유실된 항목 ID들의 범위를 얻고(Tx를 시작하고, 클러스터된 키들과 매치하지 않는 클러스터되지 않는 인덱스로부터 모든 인덱스 키들을 리턴하는 DBCC CHECK 표(table, REPAIR_ALLOW_DATA_LOSS)를 실행하고, 클러스터되지 않은 인덱스 키들은 항목 ID들을 포함하여, DRS는 이들을 추출할 수 있고, 내부 표에 이들을 기록할 수 있음), 및 (c) 단계(812)에서, DBFS가 일관적임을 보장하기 위해 WCC(여기서, WCC는 이들 항목 ID들을 수리함)에 항목 ID들과 표 이름들의 리스트를 전달하고(및 저장소의 나머지 부분을 검사함), Tx에 위임하여 DRS가 이들 복구들을 취소하고 재시도하도록 한다. 손상된 페이지들의 세트가 존재하면, DRS는 추가 프로세싱을 위해(백업으로부터 페이지를 복원하여서와 같이) 그들을 핸들링하기 전에 모든 손상된 페이지들로부터 모든 항목 ID들을 알아낸다.

[0073] 페이지 레벨 복원이 잘 동작하면 사용자 데이터는 유실되지 않을 것임을 DRS는 보장할 수 있으므로, 페이지 레벨 복원이 일어나는 동안 사용자는 대부분 단지 기대된 것보다 더 느린 응답만을 알아챌 것이므로 DRS는 단지 명목상으로 사용자에게 알릴 필요가 있다. 특정 실시예들에서, 사용자는 아무런 통지를 수신하지 않을 것이다(이것은 고작 몇 초가 걸릴 것이기 때문이므로); 그러나, 다른 실시예들에 대해, DRS는 DRS가 페이지를 복원했다는 사실을 "이벤트" 로그 엔트리가 캡처하도록 할 것이다. 복원되는 페이지에 대해 미해결의 활성화된 트랜잭션이 존재하는 경우의 최악의 시나리오에서, 데이터베이스가 분리되고 재시작되어야 하고(크래쉬 복구를 실행시키기 위해), 그러므로 데이터베이스로의 모든 접속이 종료될 것이다. 응용 프로그램들은 이 가능성을 염두에 두고 이미 디자인되어야 한다; 그러나, 이것은 서투르게 디자인된 응용 프로그램들이 "공중에 뜨는(hang)" 원인이 될 수 있다. 그러므로, 본 발명의 특정 실시예들은 이 바람직하지 못한 결과를 피하려는 노력에서 모든 그런 응용 프로그램들을 사용자가 단도록 하는 그런 이벤트로만 유도될 수 있는 상황들에 대한 사용자 통지를 제공한다.

[0074] **샘플 최종 사용자 경험**

[0075] DRS의 동작을 설명하기 위해, 본 명세서에서 사용자 경험 및 사용자에게 비가시적인 DRS가 수행하는 것을 일반적으로 특성화하는 몇 개의 상황들이 기재된다.

[0076] 인덱스 손상:

[0077] 애비(Abby: 사람의 성명)는 지난주에 그녀가 수정한 모든 문서들을 찾기 위해 WinFS 쿼리를 수행하고 있다. 이 쿼리 동안, WinFS는 보통 때보다 좀 더 시간이 걸림을 알아챈다. 사실상 그녀의 후속 쿼리들의 일부도 또한 좀 느리다. 그 다음, 그녀는 그녀의 작업줄(task bar)에 작은 풍선을 보게된다. 이 풍선은 신뢰성 이슈가 그녀의 머신에서 발견되었고 윈도우즈가 오류들을 수리할 동안 인내심을 가지고 기다릴 것을 보고한다. 잠시 후, 다른 풍선이 나타나서 애비에게 인덱스가 성공적으로 재구성되었음을 통지한다. 애비는 그녀의 컴퓨터가 이제 성능이 나아진 것 같아서 기쁘다.

[0078] 장면들 뒤에서, DRS는 인덱스 손상을 탐지하고 수리한다. 그것은 인덱스를 오프라인으로 하고(그러므로 성능이 느려짐), 그것을 재구성하고, 및 그 다음, 인덱스를 다시 온라인으로 만든다.

[0079] 찢어진 페이지 쓰기:

[0080] 번개치는 거친 날씨 중에 토비는 숙제로서 아담 스미스에 대한 에세이를 쓰고 있고, 정기적으로 저장하고 있다. 에세이를 거의 90% 정도 썼을 때, 토비의 전체 숙제에 전력이 중단된다. 토비는 랩톱에서 실행하는 중이 아니고, 전지 백업을 가지고 있지 않다. 토비에게 다행히도, 약 30분 후에 전력이 다시 재개된다. 토비는 컴퓨터에 로그인하고, 그의 에세이를 열려고 시도한다. 그것이 보통 때보다 조금 더 시간이 걸리는 것 같아 토비는 머리를 긁적인다.

[0081] 장면들 뒤에서 일어난 일은 토비의 에세이 문서 항목이 전력이 유실되었을 때 그의 하드 디스크 드라이브에 의한 찢어진 페이지 쓰기 때문에 손상된 것이다. DBFS가 재시작되었을 때, DRS는 이 데이터 페이지 손상을 탐지

하고, 그 데이터의 복구를 시도한다. DRS는 자동 스냅샷으로부터 데이터 페이지를 복원하려고 자동으로 시도한다. 토비가 그의 작업을 정기적으로 저장했기 때문에, 가장 최신 스냅샷에 에세이의 복사본이 존재하였다. 그러므로, DBFS는 손상된 페이지들을 자동으로 복원할 수 있었다.

[0082] 비트 룯(Bit Rot) 및 섹터 손상:

[0083] 수잔은 그녀의 디지털 카메라를 사랑하고, 현재까지 최근 2년 동안 그녀의 자식들의 5800장의 사진들을 찍어왔다. 이들 5800장 중에, 그녀는 컴퓨터의 WinFS에 3000장이 넘는 사진들을 보관해왔다. 수잔에게 불행히도, 이들 중요한 사진들을 가진 그녀의 하드 디스크 드라이브가 디스크의 소수의 섹터들을 손상시켰다. 이들 손상된 섹터들은 이제 수잔의 사진들 중의 10장을 손상시켰다. 수잔이 이들 사진들을 일람하려고 시도할 때, 그녀는 운영 시스템 쉘로부터 오류를 얻고, 백업으로부터 이들 사진들을 복원할 것이 촉구된다. 수잔은 그 촉구를 따르고, 그녀의 백업 매체(집 드라이브)를 찾고, 그 다음, 디스크로부터 항목들을 복원한다.

[0084] 장면들 뒤에서, DBFS는 페이지 레벨 복원을 시도하고 있다. 그러나, 이들 사진들이 수정되지 않았기 때문에(전혀), 그들은 가장 최신의 스냅샷에 존재하지 않고, 그러므로, 수잔이 그녀의 백업(단지 이들 항목들에 대해서만)을 집어넣으라고 촉구하는 것은 이 특정 실시예에서 필요하다.

[0085] **결 론**

[0086] 본 명세서에서 기재된 다양한 시스템, 방법, 및 기술은 하드웨어 또는 소프트웨어 또는, 필요한 경우, 그 둘의 조합으로 구현될 수 있다. 그러므로, 본 발명, 또는 그것의 특정 양태들이나 일부, 의 방법들 및 장치는, 플로피 디스켓, CD-ROM, 하드 드라이브, 또는 임의의 다른 기계-관독가능한 저장 매체와 같은, 실제적 매체에 구현된 프로그램 코드(즉 명령들)의 형태를 취할 것이고, 프로그램 코드가, 컴퓨터와 같은, 기계에 의해 로드되고 실행될 때, 기계는 본 발명을 실시하는 장치가 된다. 프로그램가능한 컴퓨터들에서 프로그램 코드의 실행의 경우, 컴퓨터는 일반적으로 프로세서, 프로세서에 의해 관독될 수 있는 저장 매체(휘발성 및 비휘발성 메모리 및/또는 저장 소자들을 포함함), 적어도 한 개의 입력 디바이스, 및 적어도 한 개의 출력 디바이스를 포함할 것이다. 한 개 이상의 프로그램들은 컴퓨터 시스템과 통신하기 위해 고수준의 절차형 프로그래밍 언어 또는 객체지향형 프로그래밍 언어로 구현되는 것이 선호된다. 그러나, 프로그램(들)은, 원하면, 어셈블리어 또는 기계어로 구현될 수 있다. 임의의 경우에, 언어는 컴파일되거나 인터프리트된 언어일 것이고, 하드웨어 구현과 조합될 수 있다.

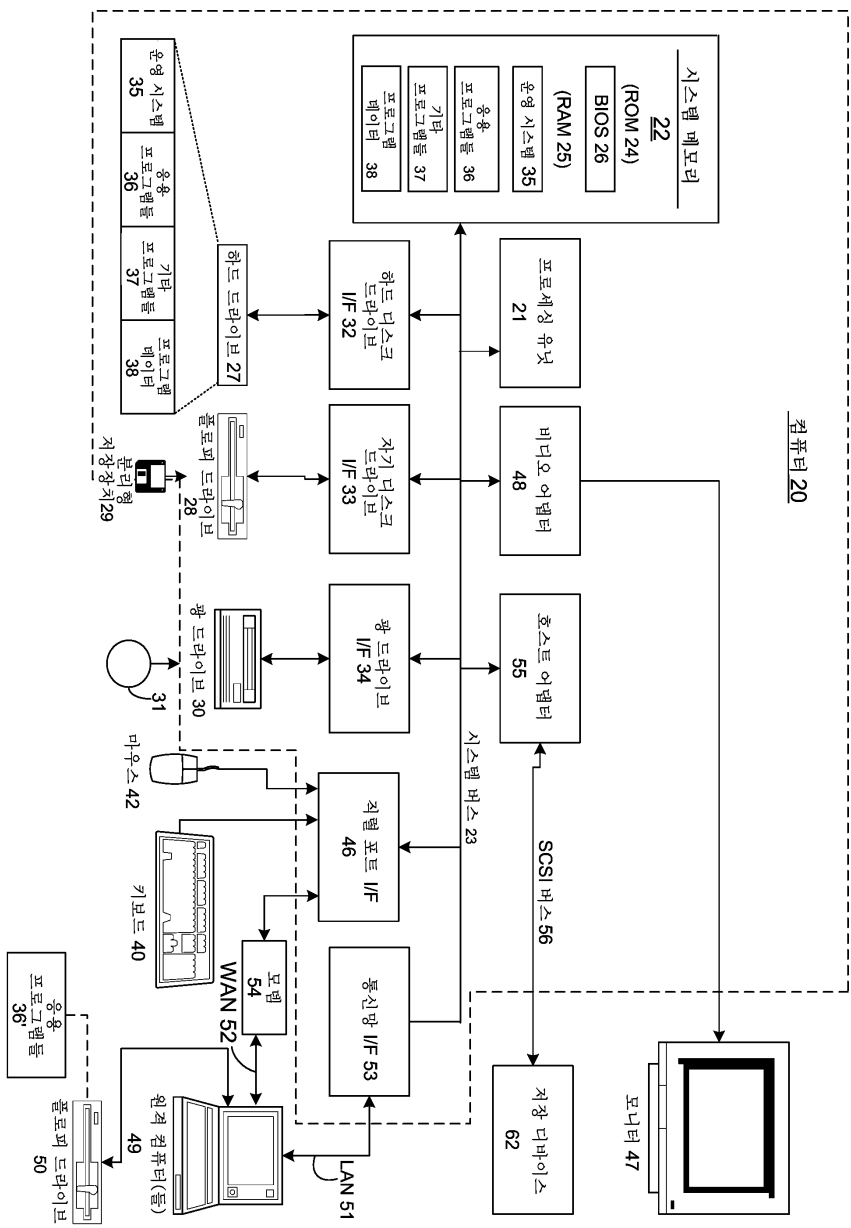
[0087] 본 발명의 방법 및 장치가 또한, 전기 유선이나 케이블을 통해, 광섬유를 통해, 또는 임의의 다른 형태의 전송을 통해서와 같이, 일부 전송 매체를 통해 전송되는 프로그램 코드의 형태로 구현될 수 있고, 프로그램 코드가, EPROM, 게이트 어레이, PLD(programmable logic device), 클라이언트 컴퓨터, 비디오 레코더 등과 같은 기계에 의해 수신되고, 로드되고, 및 실행될 때, 기계는 본 발명을 실시하는 장치가 된다. 일반-목적 프로세서에 구현될 때, 프로그램 코드는 본 발명의 인택싱 기능을 수행하기 위해 동작하는 유일한 장치를 제공하기 위해 프로세서와 결합된다.

[0088] 본 발명이 다양한 도면들의 선호되는 실시예들과 연결하여 기재되는 한편, 다른 유사한 실시예들이 사용될 수 있거나, 또는 본 발명에서 벗어나지 않고 본 발명의 동일 기능을 수행하기 위해 기재된 실시예로 수정 및 추가가 만들어질 수 있음을 이해할 것이다. 예를 들어, 본 발명의 실시예들의 예가 개인용 컴퓨터들의 기능을 모방하는 디지털 디바이스들의 컨텍스트로 기재되는 한편, 당업자는, 본 발명이 본 출원서에 기재된 바와 같이 그런 디지털 디바이스들에 제한되지 않고, 게임 콘솔, 핸드헬드 컴퓨터, 휴대형 컴퓨터 등과 같은, 유선이나 무선에 무관하게, 임의의 수의 기존 또는 새로 생겨나는 컴퓨팅 디바이스들 또는 환경들에 적용될 것이고, 통신망을 통해 접속되어 상호작용하는 임의의 수의 그런 통신망 디바이스들에 적용될 수 있음을 인식할 것이다. 더욱이, 핸드헬드 디바이스 운영 시스템 및 다른 응용 프로그램 고유의 하드웨어/소프트웨어 인터페이스 시스템을 포함하는 다양한 컴퓨터 플랫폼들이, 특히 무선 통신망 디바이스들의 수가 증가하기를 계속함에 따라, 본 명세서에서 고려됨이 강조되어야 한다. 그러므로, 본 발명은 임의의 한 개의 실시예에 제한되어서는 안 되고, 그 보다 는 부착된 청구범위에 따라 넓이와 범위에서 이해되어야 한다.

도면의 간단한 설명

[0012] 전술된 개요 및 선호되는 실시예들에 대한 다음의 상세한 설명은 부착된 도면들과 연결하여 읽혀질 때 더 잘 이해된다. 본 발명을 설명하는 목적으로, 도면들에 본 발명의 예시적 구성들이 도시된다: 그러나, 본 발명은 개시된 특정 방법들 및 도구들에 제한되지는 않는다.

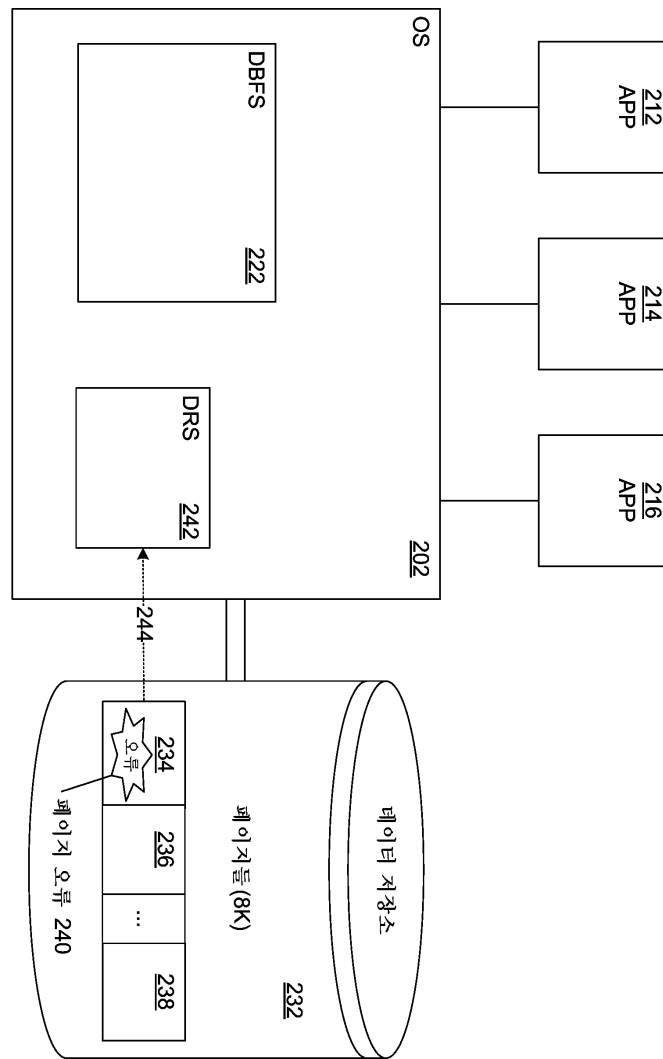
- [0013] 도 1은 본 발명의 양태들이 병합될 수 있는 컴퓨터 시스템을 나타내는 블록도이다.
- [0014] 도 2는 본 발명의 여러 개의 실시예들을 나타내는 데이터베이스 파일 시스템(database file system;DBFS)의 데이터 신뢰성 시스템(data reliability system;DRS)의 구조를 설명하는 블록도이다.
- [0015] 도 3a는 데이터베이스 FPM에 의해 DBFS로 데이터 저장소들의 부착을 설명하는 블록도이다.
- [0016] 도 3b는 본 발명의 여러 개의 실시예들을 위한 데이터베이스 부착 동작 동안 손상된 페이지들이 탐지되고 교정되는 프로세스를 설명하는 흐름도이다.
- [0017] 도 4a는 본 발명의 다양한 실시예들에 따라 스타트-업(start-up)/크래쉬-복구(crash recovery) 컨텍스트로 DRS의 사용을 설명하는 흐름도이다.
- [0018] 도 4b는 데이터베이스 부착 동안 DRS에 의해 핸들링되는 오류들을 설명하는 표이다.
- [0019] 도 5a는 본 발명의 다양한 실시예들에 대해 쿼리 엔진(query engine)이 오류를 대면하는 온라인 동작들에서 DRS의 사용을 설명하는 흐름도이다.
- [0020] 도 5b는 온라인 동작들 동안 DRS에 의해 핸들링되는 오류들을 설명하는 표이다.
- [0021] 도 6은 페이지가 가장 최신의 스냅샷에 존재하고 유효한 트랜잭션 로그(transaction log)가 존재할 때 복원을 수행하는 DRS를 설명하는 흐름도이다.
- [0022] 도 7은 데이터베이스가 복구될 수 없고 사용가능한 백업(backup)이 없는 경우 본 발명의 특정 DRS 실시예들에 대해 데이터베이스를 온라인으로 다시 불러오는 방법을 설명하는 흐름도이다.
- [0023] 도 8은 본 발명의 일 양태에 따라 데이터 페이지를 복구하려고 시도하기 위한 DRS에 대한 방법을 설명하는 흐름도이다.



도면

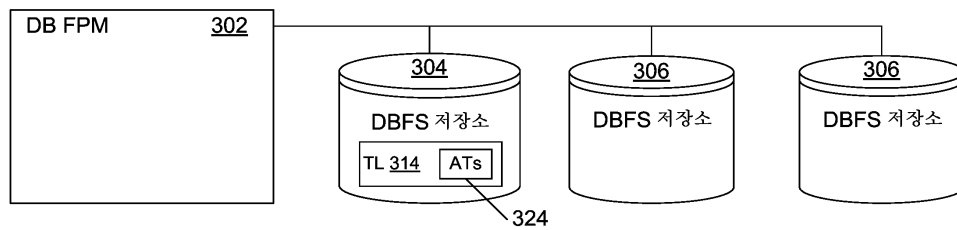
도면1

도면2



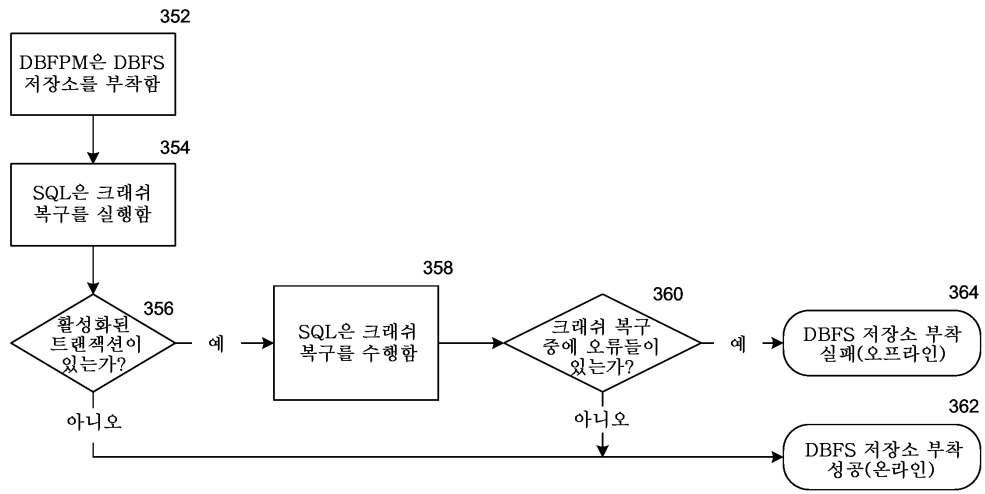
도면3a

(종래 기술)

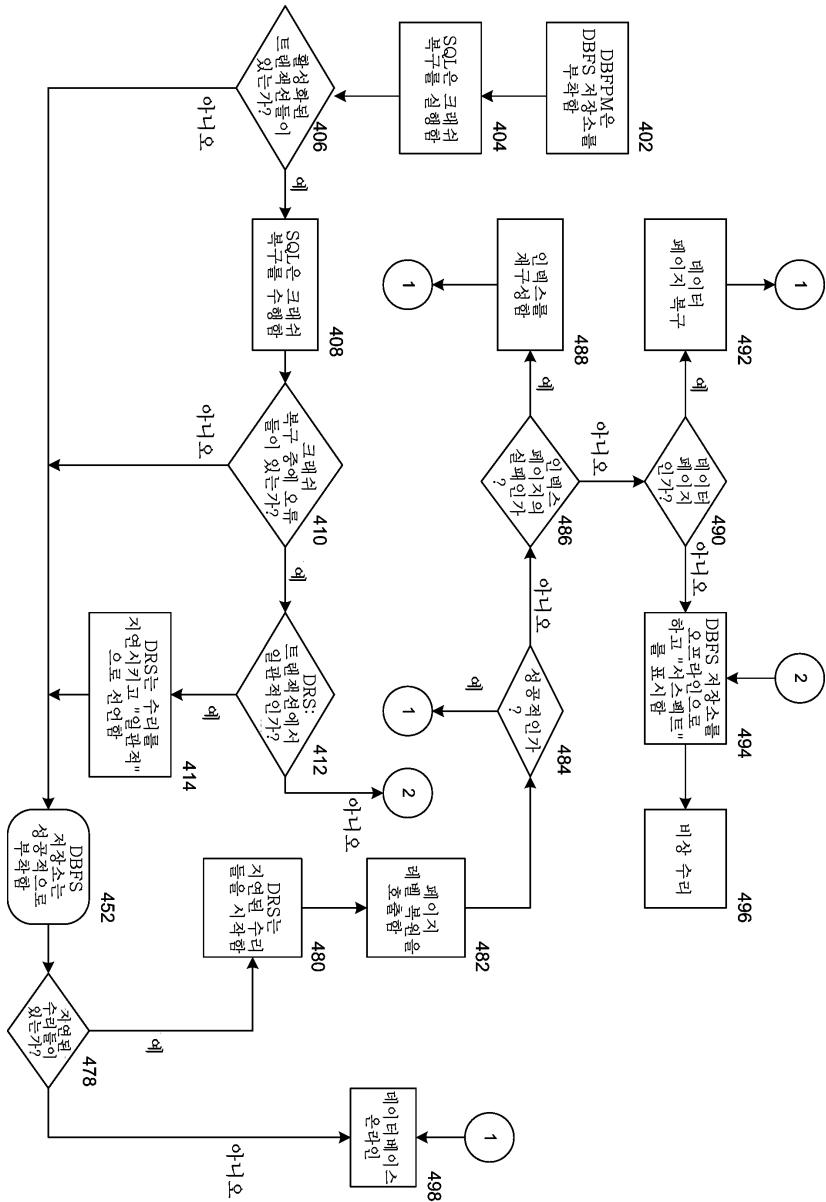


도면3b

(종래 기술)



도면4a

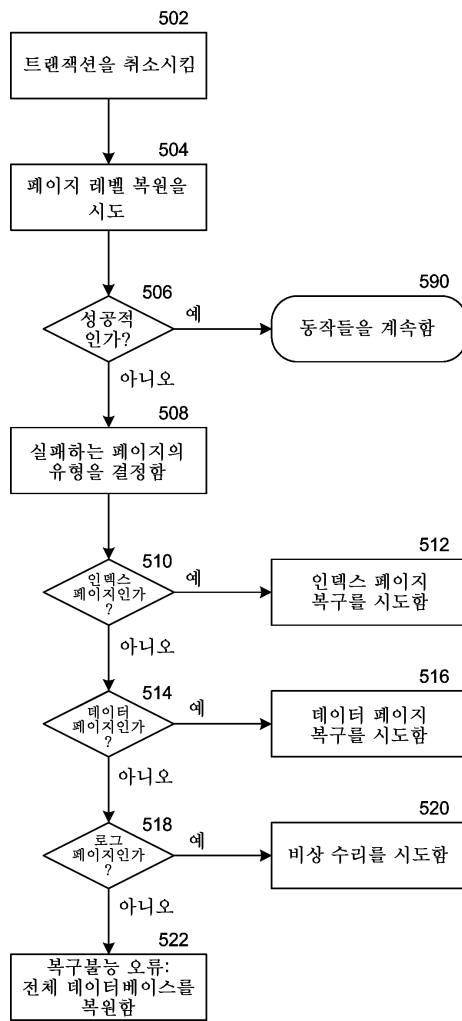


도면4b

데이터베이스 부차 오류 핸들링

오류 번호	설명
824	I/O 오류 %1s가 파일 '%6s'에 오픈셋 %5s에 dbid %4s의 페이지 %3s의 %2s 동안 원도우즈 파일 시스템에 의해 탐지됨
823	I/O 오류 %1s가 파일 '%4s'에서 오픈셋 %3s에 %2s 동안 OS에 의해 탐지됨
605	데이터베이스에 논리 페이지 %1s를 폐치하려고 시도함. %2s는 %5s:%6s가 아닌 할당 유닛 %3s:%4s에 속함

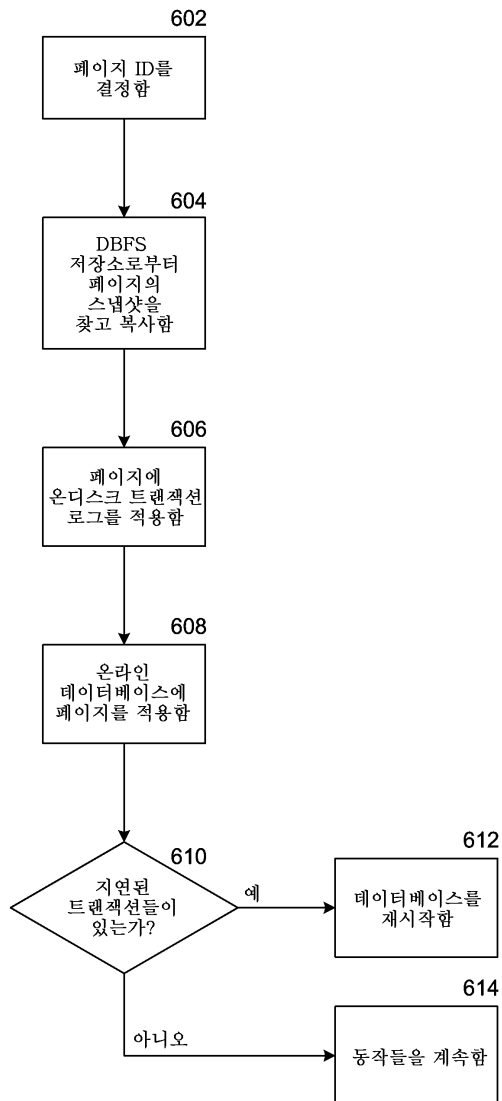
도면5a



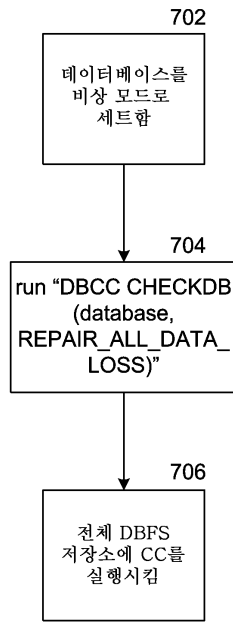
온라인 동작 오류 웬들링

오류 번호	설명
824	I/O 오류 %1s!가 파일 '%6s!'에 오프셋 %5s!에 dbid %4s!의 페이지 %3s!의 %2s! 동안 원도우즈 파일 시스템에 의해 탐지됨
823	I/O 오류 %1s!가 파일 '%4s!'에서 오프셋 %3s!에 %2s! 동안 OS에 의해 탐지됨
605	테이터페이스에 논리 페이지 %1s!를 폐지하려고 시도함. %2s!는 %5s!:%6s!가 아닌 칼럼 유닛 %3s!:%4s!에 속함
615	테이터페이스 포 ID %1s!, 이름 '%2s!'를 찾지 못함

도면6



도면7



도면8

