(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2014-0083556
(43) 공개일자 2014년07월04일

(51) 국제특허분류(Int. Cl.)
G06F 17/30 (2006.01) *G06F 17/40* (2006.01)
(21) 출원번호 10-2012-0153447
(22) 출원일자 2012년12월26일
심사청구일자 2012년12월26일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
박수영
인천 부평구 부영로 165, 108동 310호 (산곡동, 우성아파트)
이상진
서울 종로구 사직로8길 20, 101동 1503호 (내수동, 파크팰리스)
전상준
경기 성남시 분당구 판교원로82번길 60, 1405동 1301호 (운중동, 경남아너스빌아파트)
(74) 대리인
최관락, 송인호, 민영준

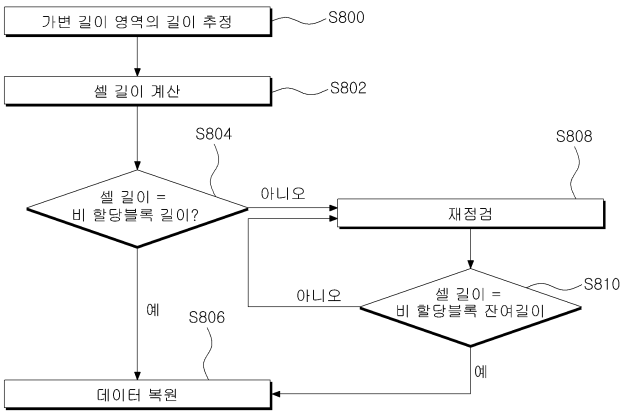
전체 청구항 수 : 총 15 항

(54) 발명의 명칭 **ＳＱＬｉｔｅ 데이터베이스에서 삭제된 데이터의 복원 방법 및 장치**

(57) 요 약

본 발명은 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법 및 장치를 개시한다. 본 발명에 따르면, SQLite 데이터베이스에서 삭제된 데이터의 복원 방법으로서, (a) 상기 데이터베이스의 데이터 페이지를 획득하는 단계; (b) 상기 획득된 데이터 페이지에서 삭제된 데이터를 포함하는 비 할당 블록을 탐색하는 단계; (c) 상기 비 할당 블록의 가변 길이 영역의 길이를 추정하는 단계; (d) 상기 추정된 가변 길이 영역의 길이에 기초하여 상기 가변 길이 영역에 상응하는 제1 셀의 길이를 계산하는 단계; (e) 상기 계산된 제1 셀의 길이와 상기 비 할당 블록의 길이가 일치하는지 판단하는 단계; 및 (f) 상기 제1 셀의 길이와 상기 비 할당 블록의 길이가 일치하는 경우, 상기 비 할당 블록에 포함된 데이터를 복원하는 단계를 포함하는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법이 제공된다.

대 표 도 - 도8



특허청구의 범위

청구항 1

SQLite 데이터베이스에서 삭제된 데이터의 복원 방법으로서,

(a) 상기 데이터베이스의 데이터 페이지를 획득하는 단계;

(b) 상기 획득된 데이터 페이지에서 삭제된 데이터를 포함하는 비 할당 블록을 탐색하는 단계;

(c) 상기 비 할당 블록의 가변 길이 영역의 길이를 추정하는 단계;

(d) 상기 추정된 가변 길이 영역의 길이에 기초하여 상기 가변 길이 영역에 상응하는 제1 셀의 길이를 계산하는 단계;

(e) 상기 계산된 제1 셀의 길이와 상기 비 할당 블록의 길이가 일치하는지 판단하는 단계; 및

(f) 상기 제1 셀의 길이와 상기 비 할당 블록의 길이가 일치하는 경우, 상기 비 할당 블록에 포함된 데이터를 복원하는 단계를 포함하는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 2

제1항에 있어서,

상기 (b) 단계는 상기 데이터 페이지에 포함된 정상적인 셀들 각각의 시작 지점 및 종료 지점을 이용하여 비어있는 영역을 식별하고, 상기 비어있는 영역에 포함된 데이터를 이용하여 상기 비 할당 블록을 탐색하는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 3

제2항에 있어서,

상기 비 할당 블록은 상기 정상적인 셀들 사이에 배치되며,

상기 (b) 단계는 상기 비어있는 영역 중 소정 데이터가 포함된 영역 이전 셀의 종료 지점 및 다음 셀의 시작 지점을 이용하여 상기 비 할당 블록의 시작 지점 및 길이를 결정하는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 4

제1항에 있어서,

상기 가변 길이 영역은 셀 사이즈, Row ID 및 헤더 사이즈를 포함하는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 5

제1항에 있어서,

상기 (c) 단계는 상기 가변 길이 영역의 길이를 미리 설정된 최소 바이트부터 최대 바이트까지 순차적으로 증가시켜 상기 가변 길이 영역의 길이를 추정하는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 6

제5항에 있어서,

상기 최소 바이트는 3이며, 상기 최대 바이트는 27인 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 7

제1항에 있어서,

상기 제1 셀의 길이는 상기 (c) 단계에서 추정된 가변 길이 영역의 길이, 필드의 개수, 상기 데이터의 타입 중

BLOB와 TEXT가 소정 바이트 이상인지 여부를 식별하는 값 및 상기 데이터 영역의 길이 중 적어도 하나를 이용하여 계산되는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 8

제1항에 있어서,

상기 제1 셀의 길이가 상기 비 할당 블록의 길이와 일치하지 않는 경우, 상기 비 할당 블록에서 상기 제1 셀의 길이만큼을 제외한 나머지 영역에서의 제2 가변 길이 영역의 길이를 추정하는 단계;

상기 추정된 제2 가변 길이 영역이 길이에 기초하여 제2 셀의 길이를 계산하는 단계; 및

상기 계산된 제2 셀의 길이와 상기 비 할당 블록의 잔여 길이를 비교하는 단계를 더 포함하되,

상기 비 할당 블록의 잔여 길이는 상기 비 할당 블록의 길이에서 상기 제1 셀의 길이를 차감한 길이인 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 9

제1항에 있어서,

데이터 삭제 시 상기 가변 길이 영역의 적어도 일부는 비 할당 블록의 길이 및 다음 비 할당 블록 오프셋을 포함하는 4 바이트 정보로 갱신되는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법.

청구항 10

제1항 내지 제9항 중 어느 한 항에 따른 방법을 수행하는 프로그램이 기록된 컴퓨터 판독 가능한 기록매체.

청구항 11

SQLite 데이터베이스에서 삭제된 데이터의 복원 장치로서,

복구를 위한 데이터베이스의 데이터베이스 파일 또는 바이너리 데이터를 입력 받아 데이터 페이지를 획득하는 데이터 페이지 획득부;

상기 획득된 데이터 페이지에서 삭제된 데이터를 포함하는 비 할당 블록을 탐색하는 비 할당 블록 탐색부;

상기 비 할당 블록의 가변 길이 영역의 길이를 추정하는 길이 추정부;

상기 추정된 가변 길이 영역의 길이에 기초하여 상기 가변 길이 영역에 상응하는 셀의 길이를 계산하고 상기 계산된 셀의 길이와 상기 비 할당 블록의 길이가 일치하는지 판단하는 검정부; 및

상기 셀의 길이와 상기 비 할당 블록의 길이가 일치하는 경우, 상기 비 할당 블록에 포함된 데이터를 복원하는 데이터 복원부를 포함하는 데이터 복원 장치.

청구항 12

제11항에 있어서,

상기 비 할당 블록 탐색부는 상기 데이터 페이지에 포함된 정상적인 셀들 각각의 시작 지점 및 종료 지점을 이용하여 비어있는 영역을 식별하고, 상기 비어있는 영역에 포함된 데이터를 이용하여 상기 비 할당 블록을 탐색하는 데이터 복원 장치.

청구항 13

제12항에 있어서,

상기 비 할당 블록은 상기 정상적인 셀들 사이에 배치되며,

상기 비 할당 블록 탐색부는 상기 비어있는 영역 중 소정 데이터가 포함된 영역 이전 셀의 종료 지점 및 다음 셀의 시작 지점을 이용하여 상기 비 할당 블록의 시작 지점 및 길이를 결정하는 데이터 복원 장치.

청구항 14

제11항에 있어서,

상기 가변 길이 영역은 셀 사이즈, Row ID 및 헤더 사이즈를 포함하는 데이터 복원 장치.

청구항 15

제11항에 있어서,

상기 검정부는 상기 가변 길이 영역의 길이를 미리 설정된 최소 바이트부터 최대 바이트까지 순차적으로 증가시켜 상기 가변 길이 영역의 길이를 추정하는 데이터 복원 장치.

명 세 서

기 술 분 야

[0001] 본 발명은 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법 및 장치에 관한 것으로서, 보다 상세하게는, 데이터베이스 파일에서 레코드를 저장한 영역을 별도로 추출하며 스키마 테이블에 의존하지 않으면서 삭제된 데이터를 복원할 수 있는 방법 및 장치에 관한 것이다.

배 경 기 술

[0002] SQLite는 MySQL나 PostgreSQL와 같은 데이터베이스 관리 시스템이지만, 서버가 아니라 응용 프로그램에 넣어 사용하는 비교적 가벼운 데이터베이스이다.

[0003] 이와 같은 SQLite는 가볍고 빠른 특성을 가지고 있어 간단한 정보를 체계적으로 저장할 필요가 있는 응용 프로그램 및 임베디드 장치에 탑재되는 소프트웨어에 사용된다.

[0004] 또한, SQLite에 있어서, 데이터베이스 사용에 대한 결과는 하나의 파일에 모두 저장되며, 모든 기능을 사용 가능하도록 설정할 경우 약 250KB 정도의 크기를 가진다.

[0005] 그러나 필수 기능 요소만 사용 가능하도록 할 경우 약 180KB 정도의 크기를 가지며, 최소 스택 공간인 16KB 스택 공간에서 사용가능하도록 제작 가능하다.

[0006] 이와 같은 SQLite는 스마트폰 운영체제(iOS, Android)에서 대부분의 응용 프로그램뿐만 아니라 사파리, 파이어폭스3, 크롬, 스카이프 등의 응용 프로그램에 사용된다.

[0007] 이처럼 광범위한 분야에서 SQLite 데이터베이스가 사용되고 있는 실정에서 데이터베이스 내에 데이터의 복원 필요성이 높아지고 있다 .

[0008] 특히 손상된 중요 데이터의 복원 및 포렌식 조사 시 데이터베이스에서 의도적으로 삭제한 정보들 중 의미 있는 정보를 복원하는 것이 특히 필요하다.

[0009] 종래에 저널 파일에 기반한 복원 기법이 제안되었으나, 저널 파일에 기반한 복원기법의 경우, 복원 가능한 레코드(데이터)가 저널 파일에서 얻을 수 있는 정보에 의존된다는 한계점이 있다.

[0010] 또한, 한국공개특허공보 제10-2012-0097293호(발명의 명칭: 데이터베이스 복원 방법 및 장치)에서는 데이터베이스 파일의 구조적 특성을 이용하여 데이터 베이스 파일의 비 할당 영역에 잔존하는 삭제된 레코드를 복원하는 기법을 제안한다.

[0011] 보다 상세하게, 헤더 페이지에 존재하는 스키마 테이블로부터 필드에 대한 정보를 획득하여 복원에 활용하며, 비 할당 블록(Free Block)으로부터 삭제된 데이터 복원하는 방식을 제안한다.

[0012] 그러나 상기한 종래기술에 따르면, 온전한 데이터베이스 파일에 존재하는 삭제된 레코드는 효율적으로 복원할 수 있지만, 삭제된 데이터베이스 파일이 조각난 경우에는 레코드의 복원이 이루어지지 않는다.

[0013] 데이터베이스 파일은 지속적으로 데이터가 축적되면서 크기가 커지는 특성 때문에, 파일 시스템에 조각난 채로 존재하는 경우가 많다. 또한 스마트폰에서 사용하는 플래시 메모리의 데이터를 물리적으로 접근하여 이미지를 획득한 경우에는 삭제되지 않은 데이터베이스 파일도 조각난 채로 획득되는 경우가 많다. 따라서 데이터베이스의 레코드 영역을 따로 획득한 경우에도 데이터를 추출할 수 있는 방법이 필요하다.

발명의 내용

해결하려는 과제

[0014] 본 발명은 상기한 바와 같은 종래기술의 문제점을 해결하기 위해, 레코드 저장 영역을 별도로 추출할 수 있으며, 스키마 테이블에 의존하지 않도록 데이터를 복원할 수 있는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법 및 장치를 제안하고자 한다.

과제의 해결 수단

[0015] 상기한 목적을 달성하기 위해, 본 발명의 바람직한 일 실시예에 따르면, SQLite 데이터베이스에서 삭제된 데이터의 복원 방법으로서, (a) 상기 데이터베이스의 데이터 페이지를 획득하는 단계; (b) 상기 획득된 데이터 페이지에서 삭제된 데이터를 포함하는 비 할당 블록을 탐색하는 단계; (c) 상기 비 할당 블록의 가변 길이 영역의 길이를 추정하는 단계; (d) 상기 추정된 가변 길이 영역의 길이에 기초하여 상기 가변 길이 영역에 상응하는 제1 셀의 길이를 계산하는 단계; (e) 상기 계산된 제1 셀의 길이와 상기 비 할당 블록의 길이가 일치하는지 판단하는 단계; 및 (f) 상기 제1 셀의 길이와 상기 비 할당 블록의 길이가 일치하는 경우, 상기 비 할당 블록에 포함된 데이터를 복원하는 단계를 포함하는 SQLite 데이터베이스에서 삭제된 데이터의 복원 방법이 제공된다.

[0016] 상기 (b) 단계는 상기 데이터 페이지에 포함된 정상적인 셀들 각각의 시작 지점 및 종료 지점을 이용하여 비어있는 영역을 식별하고, 상기 비어있는 영역에 포함된 데이터를 이용하여 상기 비 할당 블록을 탐색할 수 있다.

[0017] 상기 비 할당 블록은 상기 정상적인 셀들 사이에 배치되며, 상기 (b) 단계는 상기 비어있는 영역 중 소정 데이터가 포함된 영역 이전 셀의 종료 지점 및 다음 셀의 시작 지점을 이용하여 상기 비 할당 블록의 시작 지점 및 길이를 결정할 수 있다.

[0018] 상기 가변 길이 영역은 셀 사이즈, Row ID 및 헤더 사이즈를 포함할 수 있다.

[0019] 상기 (c) 단계는 상기 가변 길이 영역의 길이를 미리 설정된 최소 바이트부터 최대 바이트까지 순차적으로 증가시켜 상기 가변 길이 영역의 길이를 추정할 수 있다.

[0020] 상기 최소 바이트는 3이며, 상기 최대 바이트는 27일 수 있다.

[0021] 상기 제1 셀의 길이는 상기 (c) 단계에서 추정된 가변 길이 영역의 길이, 필드의 개수, 상기 데이터의 타입 중 BLOB와 TEXT가 소정 바이트 이상인지 여부를 식별하는 값 및 상기 데이터 영역의 길이 중 적어도 하나를 이용하여 계산될 수 있다.

[0022] 상기 제1 셀의 길이가 상기 비 할당 블록의 길이와 일치하지 않는 경우, 상기 비 할당 블록에서 상기 제1 셀의 길이만큼을 제외한 나머지 영역에서의 제2 가변 길이 영역의 길이를 추정하는 단계; 상기 추정된 제2 가변 길이 영역이 길이에 기초하여 제2 셀의 길이를 계산하는 단계; 및 상기 계산된 제2 셀의 길이와 상기 비 할당 블록의 잔여 길이를 비교하는 단계를 더 포함하되, 상기 비 할당 블록의 잔여 길이는 상기 비 할당 블록의 길이에서 상기 제1 셀의 길이를 차감한 길이일 수 있다.

[0023] 데이터 삭제 시 상기 가변 길이 영역의 적어도 일부는 비 할당 블록의 길이 및 다음 비 할당 블록 오프셋을 포함하는 4 바이트 정보로 갱신될 수 있다.

[0024] 본 발명의 다른 측면에 따르면, 상기한 방법을 수행하는 프로그램이 기록된 컴퓨터 판독 가능한 기록매체가 제공된다.

[0025] 본 발명의 또 다른 측면에 따르면, SQLite 데이터베이스에서 삭제된 데이터의 복원 장치로서, 복구를 위한 데이터베이스의 데이터베이스 파일 또는 바이너리 데이터를 입력 받아 데이터 페이지를 획득하는 데이터 페이지 획득부; 상기 획득된 데이터 페이지에서 삭제된 데이터를 포함하는 비 할당 블록을 탐색하는 비 할당 블록 탐색부; 상기 비 할당 블록의 가변 길이 영역의 길이를 추정하는 길이 추정부; 상기 추정된 가변 길이 영역의 길이에 기초하여 상기 가변 길이 영역에 상응하는 셀의 길이를 계산하고 상기 계산된 셀의 길이와 상기 비 할당 블록의 길이가 일치하는지 판단하는 검정부; 및 상기 셀의 길이와 상기 비 할당 블록의 길이가 일치하는 경우, 상기 비 할당 블록에 포함된 데이터를 복원하는 데이터 복원부를 포함하는 데이터 복원 장치가 제공된다.

발명의 효과

[0026] 본 발명에 따르면, 특정 응용프로그램이나 파일시스템에 종속되지 않으며 모든 경우에 SQLite 데이터베이스에 대하여 범용적으로 사용 가능하며, SQLite에서 삭제된 데이터를 통해 의미 있는 정보를 추출하는데 효과적으로 활용할 수 있는 장점이 있다.

도면의 간단한 설명

[0027] 도 1은 데이터베이스 파일 중 헤더 페이지에 포함된 정보를 도시한 도면.

도 2는 데이터베이스 파일 중 B-Tree의 구조를 도시한 도면.

도 3은 데이터 페이지의 구조를 도시한 도면.

도 4는 셀의 상세 구조를 도시한 도면.

도 5는 셀이 삭제되는 경우 가변 길이 영역의 4 바이트가 갱신된 상태를 도시한 도면.

도 6은 본 발명의 바람직한 일 실시예에 따른 SQLite 데이터베이스에서 데이터의 복원 과정을 도시한 순서도.

도 7은 본 발명의 바람직한 일 실시예에 따른 SQLite 데이터 페이지 카빙 과정을 도시한 도면.

도 8은 본 발명의 바람직한 일 실시예에 따른 데이터 헤더 위치 추측 이후의 데이터 복원 과정을 도시한 순서도.

도 9는 본 발명에 따른 데이터 복원 장치의 상세 구성을 도시한 블록도.

발명을 실시하기 위한 구체적인 내용

[0028] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.

[0029] 이하, 본 발명의 바람직한 실시예를 첨부한 도면들을 참조하여 상세히 설명하기로 한다. 본 발명을 설명함에 있어 전체적인 이해를 용이하게 하기 위하여 도면 번호에 상관없이 동일한 수단에 대해서는 동일한 참조 번호를 사용하기로 한다.

[0030] 본 발명은 SQLite 데이터베이스의 삭제된 데이터를 복원하는 방법 및 장치에 관한 것으로서, 아래에서 SQLite 데이터베이스 파일 구조를 상세하게 살펴본다.

[0031] SQLite 데이터베이스 파일은 헤더 페이지(Header Page) 및 B-Tree를 포함할 수 있다.

[0032] 도 1은 데이터베이스 파일 중 헤더 페이지에 포함된 정보를 도시한 도면이다.

[0033] 도 1에 도시된 바와 같이, 헤더 페이지는 데이터베이스 헤더 및 스키마 테이블을 포함할 수 있다.

[0034] 데이터베이스 헤더는 데이터베이스에 대한 페이지의 사이즈, 수정된 횟수 등과 같은 일반적인 정보를 포함한다.

[0035] 스키마 테이블은 SQLite 데이터베이스 파일을 구성하는 테이블, 인덱스, 트리거에 대한 정보를 포함하며, 테이블의 이름, 테이블을 구성하는 필드의 개수, 각 필드의 이름과 타입을 포함한다.

[0036] 도 2는 데이터베이스 파일 중 B-Tree의 구조를 도시한 도면이다.

[0037] 도 2에 도시된 바와 같이, B-Tree는 내부 페이지(Internal Page) 및 리프 페이지(Leaf Page)를 포함할 수 있다.

[0038] 내부 페이지는 하위 페이지의 포인트를 가지며, 리프 페이지는 데이터 페이지로서, 실제 데이터(셀)를 포함한다.

[0039] 도 3은 데이터 페이지의 구조를 도시한 도면이다.

[0040] 도 3에 도시된 바와 같이 리프 페이지는 페이지 헤더(Page Header), 셀 오프셋(Cell offset), 비 할당 공간(Free Space), 셀(Cell) 및 비 할당 블록(Free block)을 포함할 수 있다.

[0041] 표 1은 페이지 헤더에 포함된 정보를 나타낸다.

표 1

[0042] 데이터 페이지의 헤더 정보

Offset	내용
0	Page flag: 0x0D
1-2	첫 번째 비 할당 블록 오프셋
3-4	데이터 페이지 내 셀 개수
5-6	첫 번째 나타나는 셀의 오프셋
7	3 바이트 이후 비할당 블록 개수

[0043] 여기서, 셀(Cell)은 SQLite 데이터베이스 파일 내 최소 저장 단위이며, 셀 오프셋은 위에서 차례로, 셀은 아래서 차례로 쓰여진다.

[0044] 비 할당 공간은 아직 셀이 할당되지 않은 영역이며, 셀 오프셋과 마지막 셀 사이의 공간에 위치한다.

[0045] 비 할당 블록은 셀이 할당되었다가 삭제된 영역으로서, 정상적인 셀과 셀 사이에 존재한다.

[0046] 도 4는 셀의 상세 구조를 도시한 도면이다.

[0047] 도 4에 도시된 바와 같이, 정상 셀은 셀 사이즈(Cell Size), Row ID, 헤더 사이즈(Header), 레코드 타입 영역 및 데이터 영역으로 이루어진다.

[0048] 여기서, 상위 세개 요소(셀 사이즈, Row ID, 헤더 사이즈)는 가변 길이 정수(영역)로서, 각각 1 내지 9 바이트의 길이로 셀마다 가변적인 길이를 가질 수 있다.

[0049] 레코드 타입 영역은 데이터 헤더로 정의될 수 있으며, 실제 데이터 각각의 타입 및 길이에 관한 정보를 포함한다.

[0050] 헤더 사이즈는 Value, Data Type, Data Size로 구성된다.

[0051] 여기서, Data Type 중 BLOB와 TEXT의 Value 역시 가변적인 값을 가진다. 기본적으로는 헤더 사이즈에서 1 바이트로 길이가 표현되며, Data Size가 56 바이트를 넘어갈 경우, 헤더 사이즈에 2 바이트로 길이를 표현한다.

[0052] 비 할당 블록은 정상 셀의 사이 사이에 존재하며 체인(chain) 형태를 가진다.

[0053] 셀 삭제 시, 해당 셀을 비 할당 블록으로 구분하기 위해 셀의 첫 4 바이트의 정보가 갱신된다.

[0054] 도 5에 도시된 바와 같이, 첫 4 바이트 중 2 바이트에는 현재 비 할당 블록 이후에 등장하는 다음 비 할당 블록에 대한 오프셋 정보(Next free block offset, 2 Byte)가 포함되며, 마지막 비 할당 블록의 경우 오프셋 값으로 0x0000값을 갖는다.

[0055] 본 발명에 따르면, 비 할당 블록을 포함하는 데이터 페이지를 획득한 이후 데이터 페이지 내에서 삭제되거나 또는 손상된 셀 내의 데이터의 복원을 수행한다.

[0056] 데이터 페이지의 획득 과정에 대해서는 하기에서 상세하게 설명한다.

[0057] 본 발명에 따른 데이터 페이지 복원은 다음의 단계를 포함한다.

[0058] **1. 가변 길이 영역의 길이(셀 사이즈, Row ID, 헤더 사이즈) 추정**

[0059] 상기한 바와 같이, 가변 길이 영역의 각 필드의 최소 길이(m)는 1이며, 최대 길이(M)는 9이다.

[0060] 상기한 가변 길이 영역의 추측 길이를 3으로 시작하여 27까지 차례로 증가시키면서 가변 길이 영역의 길이를 추측한다.

[0061] **2. 추측 길이 검정**

[0062] 추측된 가변 길이 영역의 길이를 토대로 셀의 전체 길이를 계산한다.

[0063] 셀의 전체 길이를 계산하는 식은 다음과 같다.

수학식 1

[0064] $$f(x)=x+F+\alpha+D$$

[0065] f: 셀의 전체 길이 계산식

[0066] x: 추측된 가변 길이 영역의 길이

[0067] P: 필드의 개수

[0068] α: TEXT와 BLOB가 56 바이트 이상인 경우에 발생하는 값

[0069] D: 데이터 영역의 길이

[0070] 여기서, 필드의 개수는 도 4의 Type1~TypeN 구간의 길이를 의미한다.

[0071] 보다 상세하게, 도 4를 참조하면, x(추측된 가변 길이 영역의 길이)는 셀 사이즈, Row ID 및 헤더 사이즈에 해당되며, 그 뒤로 데이터(필드)의 타입(Type 1~Type N)이 나타난다.

[0072] 이때, 각 데이터(필드)의 타입은 1Byte를 사용하여 표현되며, 따라서 N개의 데이터가 있다면 데이터의 타입을 나타내는 구간에서 N byte가 사용된다.

[0073] 단, 데이터(필드)의 타입이 TEXT나 BLOB일 경우, 실제 데이터가 56바이트 이상이 되면 1Byte 이상을 사용하게 됩니다. 이로 인해 추가적으로 늘어나게 되는 길이는 α로 포함시킨다.

[0074] 이후, 현재 비 할당 블록의 길이와 계산된 값을 비교하여 일치하는 경우에는 바로 데이터 추출을 진행한다.

[0075] 비 할당 블록의 길이는 데이터 페이지의 셀 오프셋 정보를 통해 미리 알 수 있다.

[0076] 또한, 비 할당 블록의 길이와 계산된 값이 일치한다는 것은 현재 비 할당 블록이 하나의 셀이라는 것을 의미하며, 이처럼 비 할당 블록의 길이와 계산된 값을 비교하여 일치하는 경우에는 가변 길이 영역 추정을 통해 레코드 타입 영역 및 데이터 영역의 시작 지점과 끝 지점을 알 수 있기 때문에 데이터의 추출, 즉 레코드의 복원이 가능하다.

[0077] 그러나, 비 할당 블록의 길이와 계산된 값이 일치하지 않는 경우에는 상기한 가변 길이 영역 추정 및 검정 과정을 반복 수행한다.

[0078] 비 할당 블록의 길이와 계산된 값이 일치하지 않는 것은 해당 비 할당 블록에 다수의 삭제된 셀(레코드)이 포함되어 있거나 손상된 셀만 존재하는 경우이다.

[0079] 비 할당 블록에 복수의 삭제된 셀이 포함되는 경우, 최초 가변 길이 영역 추정 및 검정 과정을 통해 첫 번째 삭제된 셀의 길이가 추정되며, 재검정 과정을 통해 두 번째 삭제된 셀에 대한 가변 길이 영역 길이 추정 및 검정 과정이 수행된다.

[0080] 재검정 과정에서 계산된 값과 비 할당 블록의 잔여 길이가 일치하는 경우에는 바로 데이터 추출을 진행한다. 여기서, 비 할당 블록의 잔여 길이는 전체 비 할당 블록의 길이 중 첫 번째 삭제된 셀의 길이를 제외한 길이를 의미한다.

[0081] 만일 두 번째 추정 및 재검정을 통해서도 길이가 서로 일치하지 않은 경우에는 나머지 잔여 영역에 대해 추정 및 검정을 반복 수행한다.

[0082] 순차적인 추정 및 검정을 통해서도 데이터가 추출되지 않은 경우에는 해당 영역엔 손상된 셀만 존재하는 것으로 판단한다.

[0083] 하기에서는 도면을 참조하여 SQLite 데이터베이스에서 삭제된 데이터의 복원을 위한 전체적인 과정을 살펴본다.

[0084] 도 6은 본 발명의 바람직한 일 실시예에 따른 SQLite 데이터베이스에서 데이터의 복원 과정을 도시한 순서도이다.

[0085] 도 6의 과정은 통상의 컴퓨팅 장치에서 수행될 수 있다.

[0086] 본 발명에 따른 복원 과정은 우선 데이터베이스의 데이터 페이지(Data page)를 찾는 것으로 시작하며, 복원 장치에 SQLite 데이터베이스 파일 또는 바이너리 데이터가 입력된다(단계 600).

[0087] 이때 복원 알고리즘에 대한 입력 값이 온전한 데이터베이스 파일인 경우와 바이너리 데이터(조립이 불가능한 Nand Flash 이미지나 비 할당 영역과 같은 데이터)인 경우에 따라 데이터 페이지를 찾는 방법이 구분된다.

[0088] 복원 장치는 입력 데이터가 데이터베이스 파일인지 여부를 판단한다(단계 602).

[0089] 온전한 데이터베이스 파일을 입력 값으로 받은 경우에는 스키마 테이블을 확인한 후 데이터베이스 파일 내에서 데이터 페이지를 구분하여 탐색하며(단계 604), 탐색된 데이터 페이지에서 데이터 복원을 수행한다(단계 606).

[0090] 그러나, 입력 값이 바이너리 데이터인 경우, 복원 장치는 전체 데이터베이스 파일을 복원할 수 있는지 여부를 판단하며(단계 608), 전체 데이터베이스 파일을 복원할 수 있는 경우에는 단계 604로 진행한다.

[0091] 반면, 전체 데이터베이스 파일을 복원할 수 있는 없는 경우에는 SQLite 데이터 페이지 카빙(Data Page Carving)을 수행하여 삭제된 채 비 할당 영역에 존재하는 데이터 페이지를 획득하여야 한다(단계 610).

[0092] 여기서, 비 할당 영역에 존재한다는 것은 데이터베이스 전체가 삭제되어 데이터 페이지가 조각난 상태로 존재하는 것을 의미한다.

[0093] 데이터 페이지가 획득되면 데이터 페이지에 대한 데이터(레코드) 복원을 수행한다(단계 606).

[0094] 이후, 마지막 데이터 페이지인지 여부를 판단하여 모든 데이터 페이지에 대한 레코드 복원을 수행한다(단계 612).

[0095] 현재 데이터 페이지가 마지막 데이터 페이지가 아닌 경우, 입력 값이 데이터베이스 파일인지 여부를 판단하며(단계 614), 데이터베이스 파일이면 단계 604로, 그렇지 않은 경우에는 단계 610으로 진행한다.

[0096] 온전한 데이터베이스 파일에 대한 복원을 수행할 경우에는 데이터 페이지 내의 삭제된 레코드만 추출하고 비 할당 영역에서 데이터 페이지만을 따로 획득한 경우에는 정상 레코드를 포함한 데이터 페이지 내의 모든 레코드를 추출할 필요가 있다. 스키마 테이블 없이 레코드를 복원하는 경우에는 각 필드의 구분은 가능하지만 필드의 이름을 알 수 없다는 단점이 있다.

[0097] 데이터 페이지 카빙은 SQLite 데이터베이스의 실제 데이터를 저장하는 데이터 페이지를 대상으로 시행한다. 실제 비 할당 영역에는 과거에 사용되었던 SQLite의 데이터 페이지가 존재하는 것을 확인할 수 있으며, 데이터 페이지 카빙을 이용하면 파일이 삭제된 후 비 할당 영역에 남아있는 SQLite의 데이터 페이지를 복구하여 데이터를 획득할 수 있다.

[0098] 도 7은 본 발명의 바람직한 일 실시예에 따른 SQLite 데이터 페이지 카빙 과정을 도시한 도면이다.

[0099] 도 7을 참조하면, 첫 번째 바이트가 0×0D인지를 판단하며(단계 700), 첫 번째 바이트가 0×0D이 아닌 경우 데이터가 존재하지 않으므로 복원 절차를 종료한다. 첫 번째 바이트가 0×0D인 경우, 데이터가 쓰였다가 삭제된 것이므로 비 할당 블록이 존재하는지 판단하다(단계 702).

[0100] 비 할당 블록이 존재하면, 비 할당 블록의 오프셋이 유효한지 판단하며(단계 704), 오프셋이 유효하지 않은 경우에는 복원 절차를 종료하고, 오프셋이 유효한 경우에는 정상적인 셀의 오프셋 및 사이즈(길이)를 검증한다(단계 706).

[0101] 상기한 단계 702에서 비 할당 블록이 존재하지 않는 경우에도 단계 706으로 진행한다.

[0102] 이후, 셀 오프셋 및 사이즈의 검증이 유효한지 여부를 판단하며(단계 708), 유효한 경우에는 검증된 사이즈를 저장한다(단계 710).

[0103] 사이즈 저장 이후 추가적인 셀이 존재하는지 판단하며(단계 712), 셀이 더 존재하는 경우 단계 706 내지 710을

반복 수행한다.

[0104] 반면 추가적인 셀이 존재하지 않는 경우 검증된 사이즈만큼 데이터 페이지를 획득한다(단계 714).

[0105] 상기한 단계 708에서 셀 오프셋 및 사이즈의 검증이 유효하지 않은 경우, 즉 정상적인 셀 구조가 나타나지 않을 경우 단계 714로 진행한다.

[0106] 리프 페이지(데이터 페이지) 시그니처는 1 바이트로 분별력이 낮다. 따라서 도 7과 같이, 데이터 페이지 헤더의 정보와 셀의 구조 일치 여부를 확인하는 검증 방법을 사용해서 데이터 페이지를 복구(획득)해야 한다.

[0107] 전술한 바와 같이, 비 할당 블록(Free Block)이란 데이터 페이지에서 삭제된 셀이 사용했던 공간을 의미한다. 데이터 페이지 헤더에 첫 번째 비 할당 블록의 위치가 저장되며 비 할당 블록에는 다음 비 할당 블록의 위치가 저장된다.

[0108] 데이터 페이지 카빙은 데이터 페이지의 사이즈를 알지 못한 상태에서 수행하기 때문에 비 할당 블록의 위치가 데이터 페이지의 최대 사이즈인 0x8000보다 작은 경우만을 인정한다.

[0109] 도 3 내지 도 4를 참조하면, 셀 구조 및 위치는 데이터 페이지 헤더에 포함된 가장 가까운 셀의 위치, 셀의 수와 앞서 검증한 비 할당 블록의 위치를 통해서 셀의 사이즈와 연속성 검증한다.

[0110] 하나 이상의 셀이 검증될 경우 검증된 위치를 저장하고 정상적인 셀 구조가 나타나지 않았을 때 저장된 사이즈만큼 획득하여 데이터 페이지를 복구한다.

[0111] 상기한 과정을 통해 정상적으로 데이터 페이지에 접근하였다면 데이터 페이지에 존재하는 레코드를 추출해야한다. 우선 데이터 페이지의 비 할당 블록에 존재하는 삭제된 데이터를 구분하여 추출하여야 하며 파일 시스템의 비 할당 영역과 같은 바이너리 데이터에서 Data page를 추출한 경우에는 정상적인 데이터도 추출할 필요가 있다.

[0112] 이하에서는 상기한 바와 같은 과정을 통해 획득된 데이터 페이지를 이용하여 삭제된 데이터를 복원하는 과정을 상세하게 설명한다.

[0113] **(1) 데이터 페이지에서의 삭제된 영역 구별**

[0114] SQLite 데이터베이스의 데이터 페이지는 생성될 때 기본적으로 0x00으로 모든 데이터가 세팅된 후 데이터가 쓰여진다. 만약 0x00이 아닌 다른 값이 존재한다면 데이터가 쓰였다가 삭제된 것이다. 상기한 표 1에 기재된 바와 같이, SQLite 데이터 페이지의 페이지 헤더에는 페이지 플래그, 첫 번째 비 할당 블록 오프셋, 페이지 내 셀의 개수, 첫 번째 나타나는 셀의 오프셋 등이 포함된다.

[0115] 본 발명에 따르면, 셀 오프셋에는 해당 셀의 시작 지점 및 종료 지점에 대한 정보가 포함되며, 복원 장치는 셋 오프셋을 이용하여 정상적인 셀들 사이에 비어 있는 영역을 식별할 수 있다. 여기서, 비어 있는 영역 중 0x00이외의 데이터가 존재하면 이는 데이터가 생성되었다가 삭제된 것이므로 이를 비 할당 블록으로 결정할 수 있다.

[0116] 이와 같은 방법으로 비 할당 블록을 탐색할 수 있으며, 비 할당 블록은 정상적인 셀들 사이에 배치되기 때문에 비어 있는 영역 중 소정 데이터(0x00이외의 데이터) 포함된 영역 이전의 셀의 종료 지점 및 비어 있는 영역 다음 셀의 시작 지점을 통해 비 할당 블록의 길이를 결정할 수 있다.

[0117] **(2) 삭제된 셀의 각 필드(Field) 구분**

[0118] 상기한 도 4에 도시된 바와 같이, SQLite는 셀이 독립적으로 존재하여도 스키마 테이블 없이도 각 필드를 구분할 수 있는 구조로 되어 있다.

[0119] 셀의 헤더에는 전체 셀의 길이(Cell Size), Row ID 및 헤더 사이즈가 가변 길이 정수(Variable length integer)로 존재한다. 가변 길이 정수는 [표 2]와 같은 구조로 되어있어 정수를 표현하는 바이트 수를 외부에 저장하지 않더라도 정수 값과 이를 표현하기 위해 사용된 바이트 수를 알 수 있다.

표 2

[0120] 가변 길이 정수 포맷

Byte	Value range	Bit pattern
1	7 bit	0XXXXXXX
2	14 bit	1XXXXXXX 0XXXXXXX
3	21 bit	1XXXXXXX 1XXXXXXX 0XXXXXXX
4	28 bit	1XXXXXXX 1XXXXXXX 1XXXXXXX 0XXXXXXX
5	35 bit	1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 0XXXXXXX
6	42 bit	1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 0XXXXXXX
7	49 bit	1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 0XXXXXXX
8	56 bit	1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 0XXXXXXX
9	64 bit	1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX 1XXXXXXX XXXXXXXX

[0121] 데이터 헤더에는 데이터 영역에 존재하는 각 필드의 데이터 유형과 사이즈를 알 수 있는 정보가 아래 표에 따라 포함되어 있다.

표 3

[0122] 데이터 헤더의 표시 형식

Value	Data Type	Data Size
0	NULL	0
N (N=1-4)	Signed Integer	N
5	Signed Integer	6
6	Signed Integer	8
7	IEEE float	8
8-11	Reserved	
N>12 (N:even)	BLOB	(N-12)/2
N>13 (N:odd)	TEXT	(N-13)/2

[0123] 정수와 실수를 표현하는 필드는 데이터 헤더에 1Byte로 데이터 유형과 길이를 표현하고 있으며 BLOB과 TEXT의 경우에는 가변 길이 정수를 사용하고 있다. BLOB과 TEXT는 각각 짝수와 홀수로 데이터 헤더에 기록되어 서로를 구분한다.

[0124] 그러나 셀이 삭제되었을 경우에는 상기한 도 5과 같이 셀 헤더의 상위 4Byte가 덮어 쓰이기 때문에 데이터 헤더의 위치를 알 수 없게 된다. 따라서 삭제된 레코드 영역을 추출하더라도 각 필드를 구분하기 위해서는 데이터 헤더의 위치를 정확히 추측할 수 있어야 한다.

[0125] 데이터 헤더의 위치를 추측하기 위해서는 데이터 헤더가 포함되는 가변 길이 영역의 길이를 추정해야 한다. 추정하는 길이는 최초 3 바이트로 시작하여 각각 1씩 차례로 늘려가며 27 바이트를 넘지 않도록 해야 한다.

[0126] 도 8은 본 발명의 바람직한 일 실시예에 따른 데이터 헤더 위치 추측 이후의 데이터 복원 과정을 도시한 순서도이다.

[0127] 도 8를 참조하면, 복원 장치는 상기한 바와 같은 과정을 가변 길이 영역의 길이의 추정한다(단계 800).

[0128] 가변 길이 정수의 길이 추정 이후, 추정된 가변 길이 정수에 기초하여 셀의 길이를 계산한다(단계 802).

[0129] 다음으로 현재 계산된 셀의 전체 길이와 비 할당 블록의 길이가 일치하는지 여부를 판단한다(단계 804).

[0130] 계산된 셀의 길이와 비 할당 블록의 길이가 일치하는 경우, 바로 데이터 추출(복원)을 수행한다(단계 806).

[0131] 그러나, 길이가 서로 일치하지 않는 경우에는, 비 할당 블록에 다수의 셀이 포함되거나 손상된 셀이 존재하는 경우이므로 단계 802에서 계산된 셀(제1 셀)의 길이의 다음 길이에서부터 가변 길이 정수의 길이 추정 및 셀(제

2 셀)의 길이를 계산하는 재검정 과정을 수행한다(단계 808).

[0132] 이후, 비 할당 블록의 잔여 길이와 단계 808에서 계산된 제2 셀의 길이가 일치하는지 여부를 판단한다(단계 810).

[0133] 여기서, 비 할당 블록의 잔여 길이는 상기한 비 할당 블록 길에서 제1 셀의 길이를 차감한 값을 의미한다.

[0134] 단계 810에서 제2 셀의 길이가 일치하는 경우에는 제2 셀에 포함된 데이터의 추출이 진행된다(단계 806).

[0135] 이와 같은 과정을 반복 수행함으로써 비 할당 블록에 포함된 모든 셀의 데이터 복원이 가능하다.

[0136] 도 9는 본 발명에 따른 데이터 복원 장치의 상세 구성을 도시한 블록도이다.

[0137] 도 9에 도시된 바와 같이, 본 발명에 따른 데이터 복원 장치는 데이터 페이지 획득부(900), 비 할당 블록 탐색부(902), 길이 추정부(904), 검정부(906) 및 데이터 복원부(908)를 포함할 수 있다.

[0138] 데이터 페이지 획득부(900)는 복구를 위한 데이터베이스의 데이터베이스 파일 또는 바이너리 데이터를 입력 받아 데이터 페이지를 획득한다.

[0139] 비 할당 블록 탐색부(902)는 획득된 데이터 페이지에서 삭제된 데이터를 포함하는 비 할당 블록을 탐색한다. 전술한 바와 같이 비 할당 블록의 탐색은 셀 오프셋을 이용하여 비 할당 블록인지 여부를 탐색하고, 비 할당 블록의 길이를 결정하는 과정이다.

[0140] 길이 추정부(904)는 비 할당 블록의 가변 길이 영역의 길이를 추정한다.

[0141] 본 발명에 따르면 가변 길이 영역의 길이는 3 바이트에서 27 바이트까지 순차적으로 증가시키면서 수행될 수 있다.

[0142] 검정부(906)는 추정된 가변 길이 영역의 길이에 기초하여 상기 가변 길이 영역에 상응하는 셀의 길이를 계산하고 상기 계산된 셀의 길이와 상기 비 할당 블록의 길이가 일치하는지 판단한다.

[0143] 데이터 복원부(908)는 계산된 셀의 길이와 비 할당 블록의 길이가 일치하는 경우, 비 할당 블록에 포함된 데이터를 복원한다.

[0144] 만일 계산된 셀의 길이와 비 할당 블록의 길이가 일치하지 않는 경우, 길이 추정부(904)는 이전에 계산된 셀의 다음 위치에서부터 가변 길이 영역의 길이 추정하며, 검정부(906)는 다음 셀의 길이 계산 및 계산된 다음 셀의 길이와 비 할당 블록 잔여 길이(여기서, 비 할당 블록 잔여 길이는 비 할당 블록 전체 길이에서 이전에 계산된 셀의 길이를 차감한 값이다)를 비교한다.

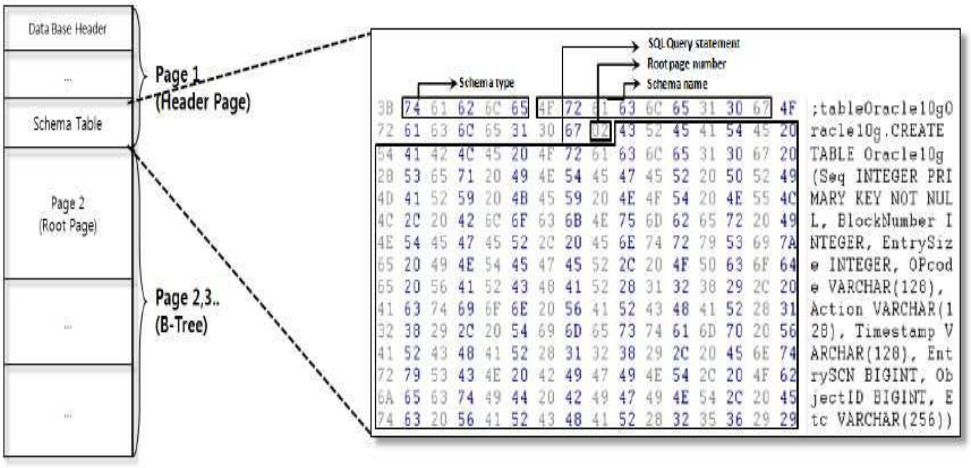
[0145] 이와 같은 과정을 반복 수행함에 따라 비 할당 블록에 포함된 삭제된 데이터를 순차적으로 복원할 수 있다.

[0146] 본 발명의 실시예들은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 본 발명을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 본 발명의 일 실시예들의 동작을 수행하기 위해 적어도 하나의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

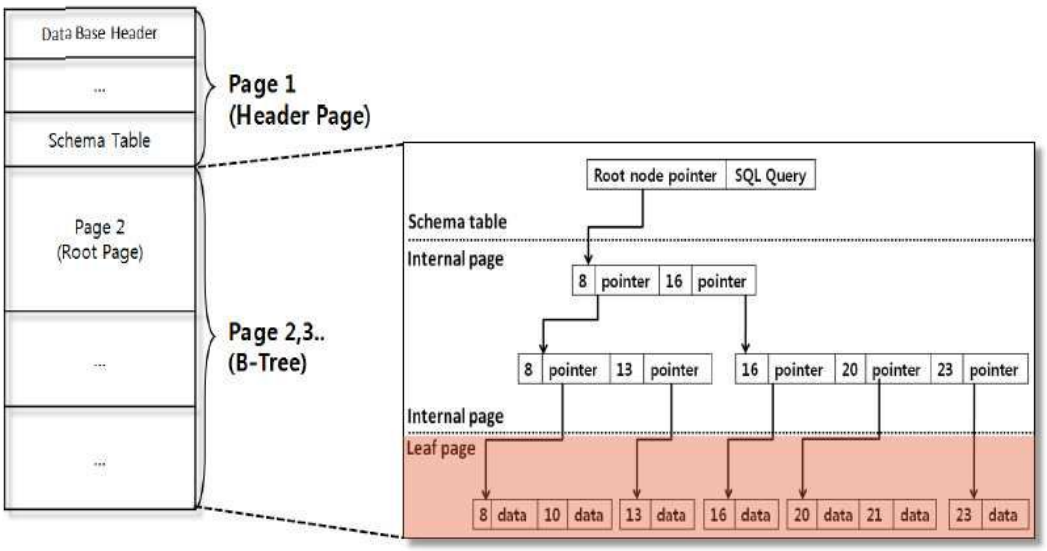
[0147] 상기한 본 발명의 바람직한 실시예는 예시의 목적을 위해 개시된 것이고, 본 발명에 대해 통상의 지식을 가진 당업자라면 본 발명의 사상과 범위 안에서 다양한 수정, 변경, 부가가 가능할 것이며, 이러한 수정, 변경 및 부가는 하기의 특허청구범위에 속하는 것으로 보아야 할 것이다.

도면

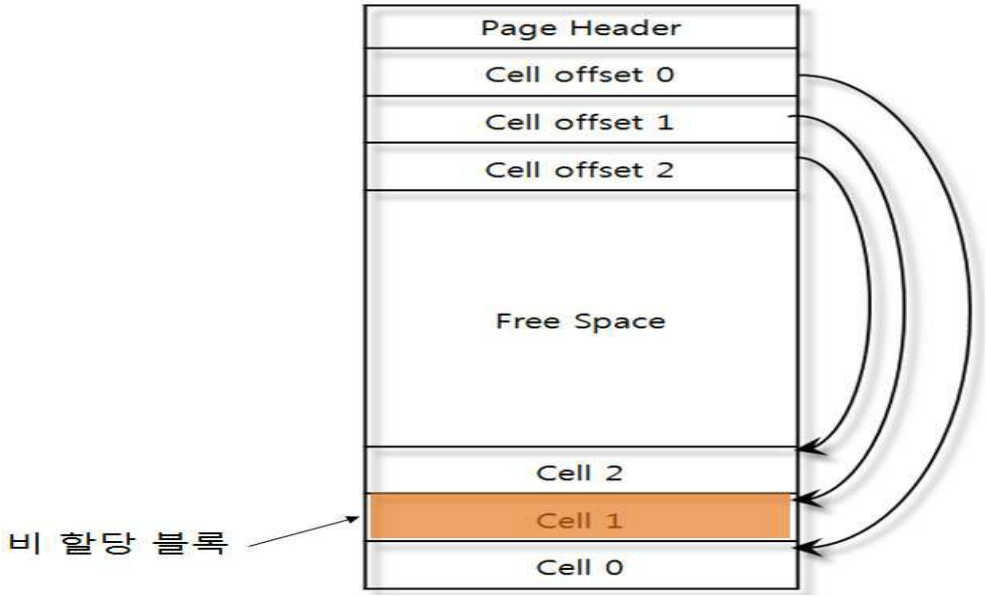
도면1


Data Base Header
...
Schema Table
Page 2
(Root Page)
...
...
Page 1
(Header Page)
Page 2,3..
(B-Tree)
Schema type
SQL Query statement
Root page number
Schema name

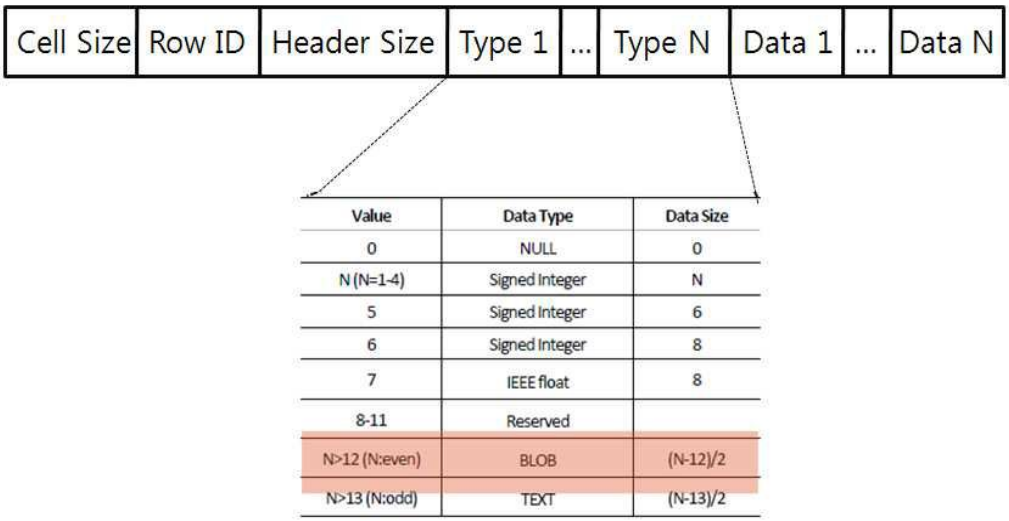
도면2


Data Base Header
...
Schema Table
Page 2
(Root Page)
...
...
Page 1
(Header Page)
Page 2,3..
(B-Tree)
Root node pointer
SQL Query
Schema table
Internal page
8 pointer 16 pointer
8 pointer 13 pointer
16 pointer 20 pointer 23 pointer
Internal page
Leaf page
8 data 10 data
13 data
16 data
20 data 21 data
23 data

도면3

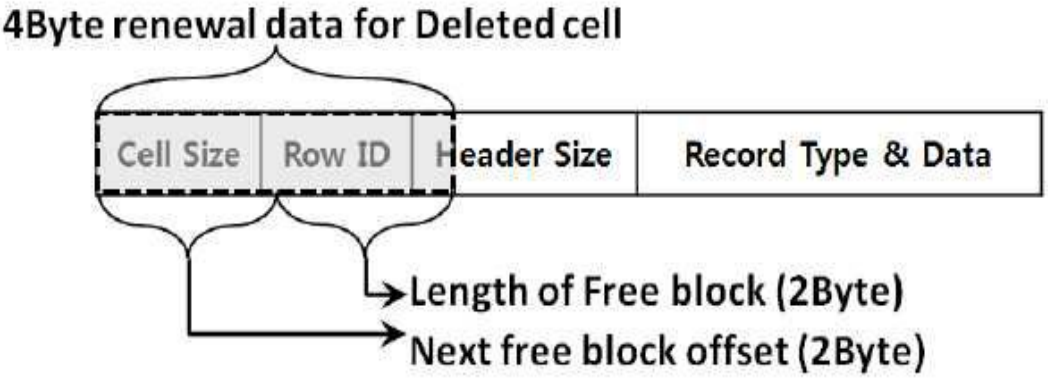


도면4

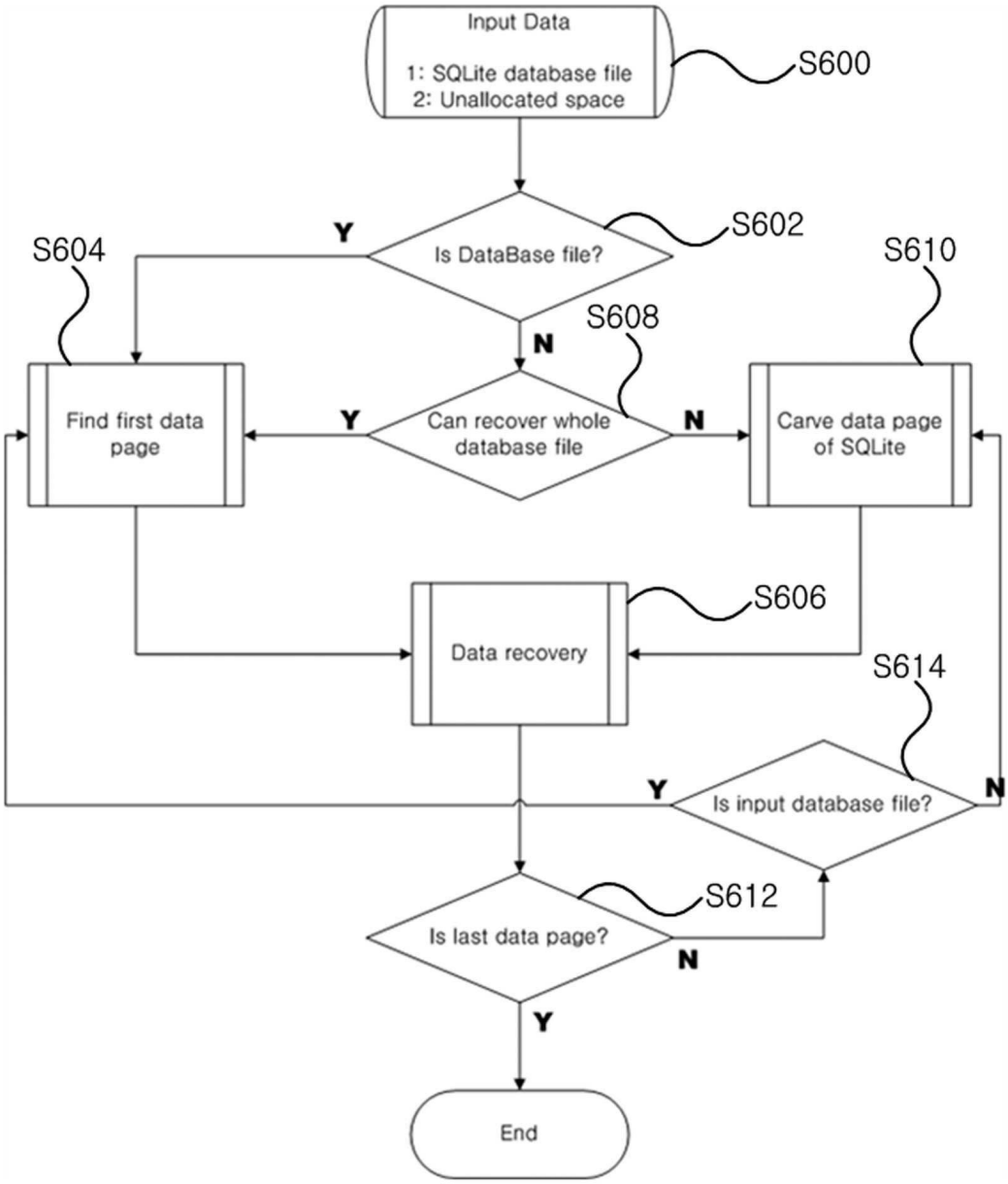


Value	Data Type	Data Size
0	NULL	0
N (N=1-4)	Signed Integer	N
5	Signed Integer	6
6	Signed Integer	8
7	IEEE float	8
8-11	Reserved	
N>12 (N:even)	BLOB	(N-12)/2
N>13 (N:odd)	TEXT	(N-13)/2

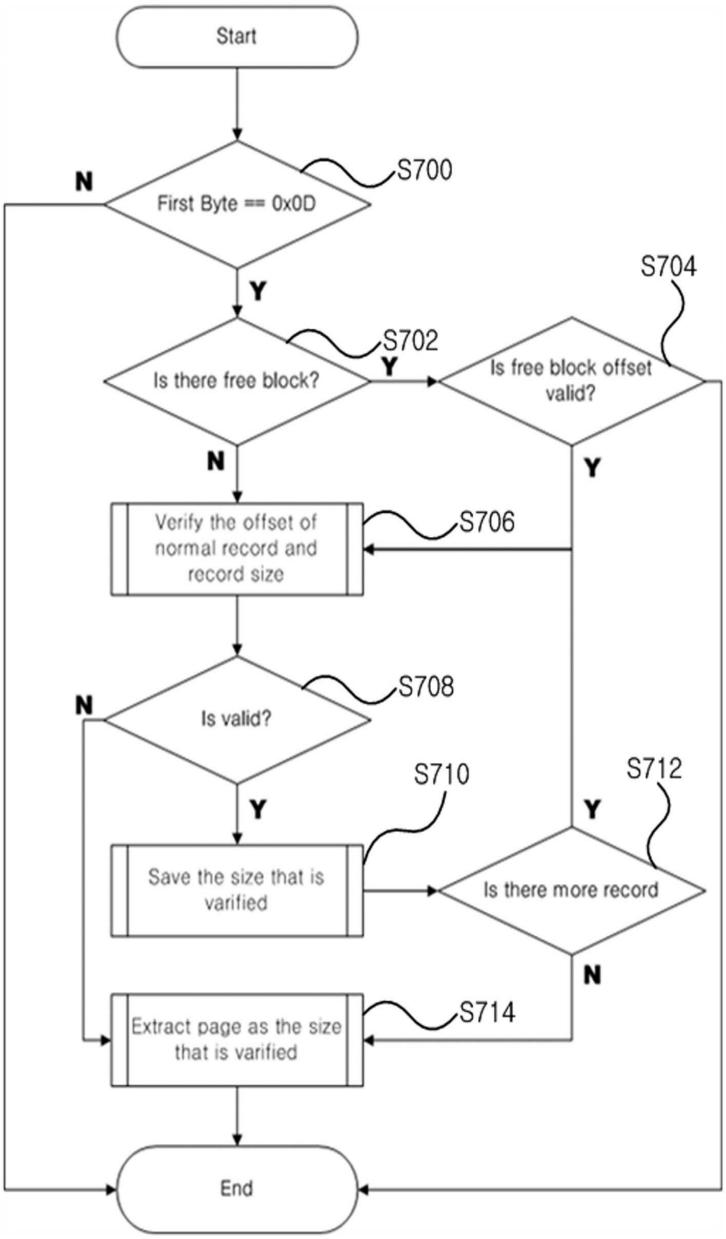
도면5



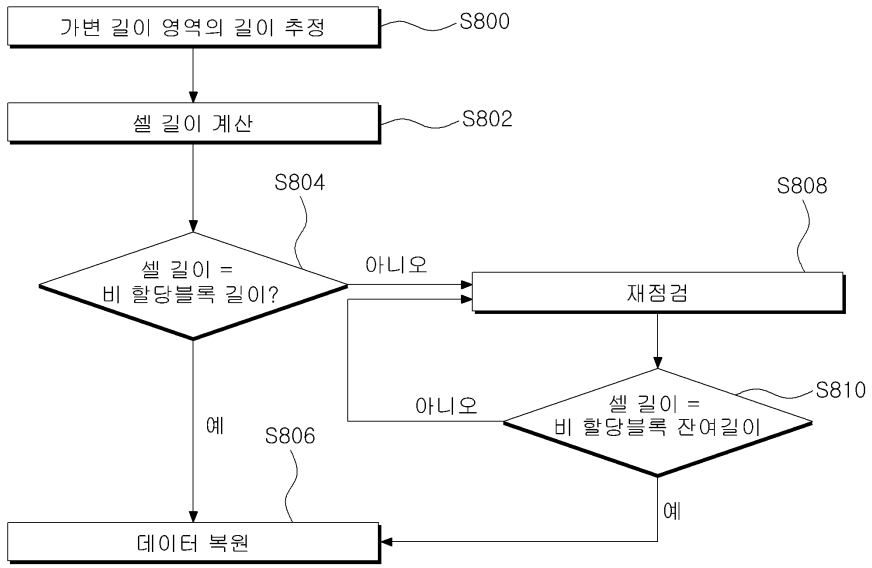
도면6


Input Data
1: SQLite database file
2: Unallocated space
S600
Is DataBase file?
S602
Y
N
S604
S610
S608
Find first data page
Can recover whole database file
Carve data page of SQLite
Y
N
S606
Data recovery
S614
Is input database file?
Y
N
S612
Is last data page?
N
Y
End

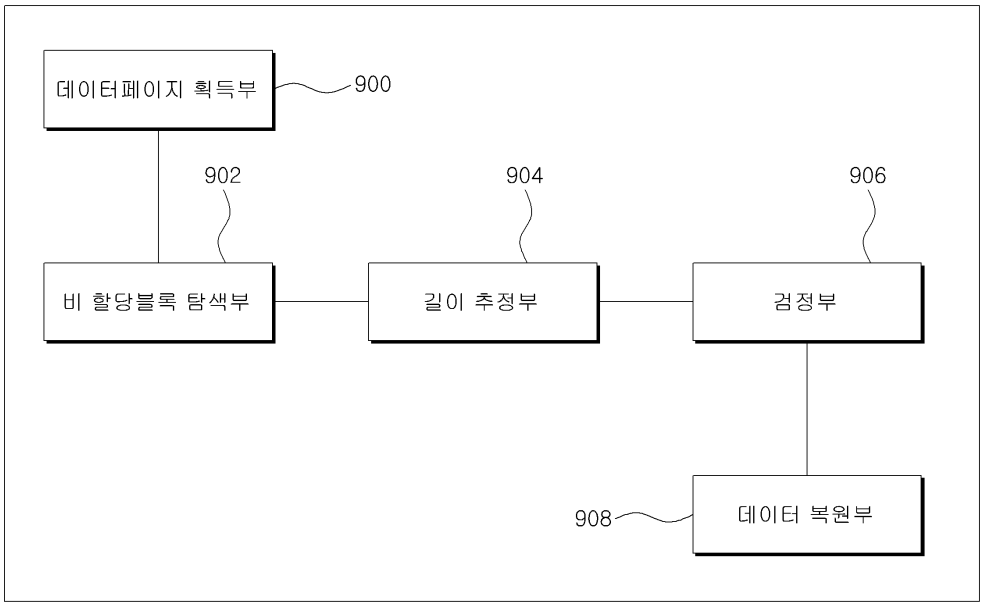
도면7


Start
S700
First Byte == 0x0D
N
Y
S702
Is there free block?
Y
S704
Is free block offset valid?
N
Y
S706
Verify the offset of normal record and record size
S708
Is valid?
N
Y
S710
Save the size that is varified
S712
Is there more record
Y
N
S714
Extract page as the size that is varified
End

도면8


가변 길이 영역의 길이 추정
S800
셀 길이 계산
S802
S804
셀 길이 =
비 할당블록 길이?
아니오
S808
재점검
아니오
셀 길이 =
비 할당블록 잔여길이
S810
예
S806
예
데이터 복원

도면9


데이터페이지 획득부
900
902
904
906
비 할당블록 탐색부
길이 추정부
검정부
908
데이터 복원부