

ARIES 논문 요약

학번: 201984-010010

이름: 김 동 훈

1. 소개

복구, 동시성 제어, 완충 관리 등과 관련된 기본 개념을 소개한다.

1.1 Logging, Failures, and Recovery Methods

트랜잭션 및 데이터 복구 보증을 충족하기 위해 ARIES는 트랜잭션의 진행률과 복구할 수 있는 데이터 개체에 대한 변경사항을 로그에 기록한다. 로그는 다양한 유형의 실패에도 불구하고 트랜잭션의 커밋된 액션이 데이터베이스에 반영되거나 커밋되지 않은 액션이 실행 취소됨(즉, 롤백 됨)을 확인하는 소스가 된다. 모든 로그 레코드는 해당 레코드가 로그에 추가될 때 고유한 LSN(Log Sequence Number)이 할당되고 WAL(Write Ahead Logging) 프로토콜을 사용한다. 트랜잭션 상태를 로그에 저장하여 복구 처리를 할 수 있다.

1.2 Latches and Locks

Latches와 Locks는 공유 정보에 대한 접근을 제어하는 데 사용된다. Latches는 세마포어와 같다. Latches는 데이터의 물리적 일관성을 보장하는 데 사용되는 반면 Locks는 데이터의 논리적 일관성을 보장하는 데 사용된다.

1.3 Fine-Granularity Locking

높은 동시성을 지원하는 것이 중요해지고 객체 지향 시스템이 인기를 끌면서, 데이터에 대한 동시성 제어와 복구를 위해 Fine-Granularity Locking을 사용한다.

1.4 Buffer Management

Buffer Manager는 버퍼 풀을 관리하는 트랜잭션 시스템의 구성요소로서 데이터베이스의 비휘발성 저장소의 read/write 페이지 I/O를 처리한다. Buffer Manager는 페이지 Latches를 지원하고 제어는 버퍼 슬롯의 버퍼 제어 블록(BCB)을 사용한다. 그리고 Buffer 관리 정책은 시스템 별로 다르다.

2. 목표

- Simplicity: 주요 알고리즘이 간단하다.
- Operation logging: 복구 방법으로 작업 로깅이 필요하다. 운영 로깅을 지원하기 위해서는 LSN과 같은 개념을 사용하여 상태를 추적하고 롤백은 compensation log records(CLRs)를 통해 로깅한다. 운영 로깅을 사용하면 논리 로깅을 수행할 수 있고 로그 공간을 절약할 수 있다.
- 유연한 스토리지 관리: 다양한 길이 데이터의 저장 및 조작에 대한 효율적인 지원이 중요하다.
- 부분 롤백: savepoint 개념과 savepoint 롤백이 필수적이다.
- 유연한 버퍼 관리: 유연성은 동시성 증가, I/O 감소, 버퍼 스토리지의 효율적인 사용할 수 있다.
- Recovery independence: 이미지 복사(아카이브 덤프)를 할 수 있어야 하며, 전체 데이터베이스 레벨에서가 아니라 다른 세분화에서 미디어 복구를 수행하거나 복구를 재시작할 수 있어야 한다.

- Logical undo: Logical undo 수행을 할 수 있다는 것은 특히 검색 구조에서 더 높은 수준의 동시성을 지원할 수 있다.
- Parallelism과 빠른 복구: 멀티프로세서가 매우 보편화되고 데이터 가용성이 점점 더 중요해지면서 복구 방법은 복구 재시작의 다른 단계와 미디어 복구 중에 병렬 처리를 이용할 수 있어야 한다.
- 최소 오버헤드: 성능과 안정성을 제공하기 위해 오버헤드(로그 데이터 볼륨, 스토리지 소비 등)는 최소화해야 한다.

3. ARIES 개요

ARIES는 프로세스, 트랜잭션, 시스템 및 미디어 장애의 사실에서 트랜잭션의 원자성과 내구성을 보장한다. 이를 위해 ARIES는 로그를 사용하여 데이터베이스의 변경사항을 추적하고 WAL(Write-ahead Logging)을 수행한다. ARIES는 로그 기록 외에도, 영향을 받는 페이지 단위로, 거래의 전방 처리 동안에 수행된 업데이트 활동도 기록한다. 일반적으로 CLR를 사용하여, 정상 처리와 재시작 처리 동안에 트랜잭션의 부분 또는 전체 롤백 중에 수행된 업데이트도 기록한다. 로그 레코드의 실행 취소가 CLR을 작성하는 경우, CLR은 재실행 목적의 보상 조치에 대한 설명을 포함하는 것 외에 방금 실행 취소된 로그 레코드의 이전 버전을 가리키는 UndoNxtLSN 포인터를 포함하도록 만들어진다. 그리고 각 페이지의 단일 LSN을 사용하여 페이지 상태를 추적하며 정기적으로 정상적인 처리 중에 체크포인트를 취한다. 재시동 복구시 ARIES는 먼저 마지막 체크포인트의 첫 번째 기록부터 로그 끝까지 로그를 스캔한다. 다음 로그 패스는 모든 실패자 트랜잭션의 업데이트가 로그의 단일 스윕에서 역순으로 롤백되는 실행 취소 패스다. 시스템 장애 당시 이미 롤백되고 있던 트랜잭션의 경우 ARIES는 아직 실행 취소되지 않은 작업만 롤백한다.

4. 데이터 구조

ARIES가 사용하는 주요 데이터 구조를 설명한다.

4.1 Log Records

- LSN: 계속 증가하는 로그 주소 공간에 있는 로그 레코드의 첫 번째 바이트 주소
- Type: compensation record 지시자
- TransID: 트랜잭션 아이디
- PrevLSN: 트랜잭션 선행 로그 레코드
- PageID: 업데이트가 적용된 페이지의 식별자
- UndoNxtLSN: 롤백 중에 처리할 이 트랜잭션의 다음 로그 레코드의 LSN
- Data: redo and/or undo 데이터

4.2 Page Structure

데이터베이스의 모든 페이지에 있는 필드 중 하나는 page_LSN 필드다. 페이지에 대한 최신 업데이트를 설명하는 로그 레코드의 LSN을 포함하고 있다.

4.3 Transaction Table

트랜잭션 테이블이라는 테이블은 복구 재시작 중에 활성 트랜잭션의 상태를 추적하기 위해 사용된다.

- TransID(Transaction ID)
- State
- LastLSN
- UndoNxtLSN

4.4 Dirty-Pages Table

정상적인 처리 중 더티 버퍼 페이지에 대한 정보를 나타내는 데 사용된다.

5. NORMAL PROCESSING

정상적인 트랜잭션 처리에 대해 논의한다.

5.1 Updates

정상적인 처리 중 트랜잭션은 전방 처리, 부분 롤백 또는 전체 롤백에 있을 수 있다. 롤백은 시스템 또는 응용 프로그램 시작일 수 있다. 롤백의 원인은 교착 상태, 오류 조건, 무결성 제약 조건 위반, 예상치 못한 데이터베이스 상태 등이 될 수 있다.

5.2 Total or Partial Rollbacks

트랜잭션 롤백의 범위를 제한하는 유연성을 제공하기 위해, 세이브포인트의 개념을 지원한다. 거래 실행 중 어느 시점에서나 세이브포인트를 설정할 수 있다.

5.3 Transaction Termination

트랜잭션 종료는 two-phase commit 규약을 사용한다.

5.4 Checkpoints

주기적으로 체크포인트를 취하여 복구 재시동 중에 수행해야 할 작업량을 줄인다.

6. RESTART PROCESSING

트랜잭션 시스템 재시작 시 트랜잭션의 원자성과 내구성을 보장하기 위해 복구를 실시할 필요가 있다. 고가용성을 위해 재시동 처리 기간은 최대한 짧아야 한다.

6.1 Analysis Pass

복구 재시작 중에 발생하는 로그의 첫 번째 패스는 Analysis Pass다.

6.2 Redo Pass

복구 재시작 중에 발생하는 로그의 두 번째 패스는 Redo Pass다.

6.3 Undo Pass

복구 재시작 중에 발생하는 로그의 세 번째 패스는 Undo Pass다.

6.4 Selective or Deferred Restart

일부 복구 작업은 나중에 수행하도록 연기할 수 있다. 일반적으로 일부 중요한 데이터를 사용할 수 없는 시간을 줄이기 위해 수행된다.

7. CHECKPOINTS DURING RESTART

복구 처리를 재시작하는 여러 단계에서 선택적으로 체크포인트를 취함으로써 CPU 처리 및 I/O에 대한 장애의 영향을 어떻게 줄일 수 있는지 설명한다.

- Analysis pass: Analysis pass가 끝날 때 체크포인트를 찍음으로써 복구 중 장애가 발생하면 작업을 절약할 수 있다.

- Redo pass: Redo pass가 시작될 때, 버퍼 관리자(BM)가 Redo pass가 진행되는 동안 수정된 페이지를 비휘발성 저장소에 기록할 때마다 RecLSN을 해당 로그 레코드의 LSN과 동일하게 하여 해당 페이지의 더티 페이지 테이블 항목을 다시 시작하도록 변경하도록 통보한다.
- Undo pass: 실행 취소 패스의 시작에서 다시 시작 더티 페이지 테이블은 BP 더티 페이지 테이블이 된다.

8. MEDIA RECOVERY

퍼지 이미지 복사 작업이 시작되면 가장 최근에 완료된 체크포인트의 begin_chkpt 레코드의 위치가 이미지 복사 데이터와 함께 기록되고 기억된다. 이 체크포인트를 이미지 카피 체크포인트라고 부른다. 미디어 복구가 필요한 경우 엔티티의 이미지 복사 버전이 다시 로드된 다음 미디어 복구 재실행 지점부터 재실행 검사를 시작한다. Page_oriented 로깅은 개체 간에 복구 독립성을 제공한다. ARIES에서는 임의 데이터베이스 페이지가 비휘발성 저장소에서 손상되어 복구해야 하는 경우에도 모든 데이터베이스 페이지의 업데이트가 별도로 기록되기 때문에 이미지 복사본에서 해당 페이지의 이전 복사본을 추출하여 위에서 설명한 로그를 사용하여 해당 페이지의 버전을 롤포워드하면 복구가 쉽게 이루어질 수 있다.

9. NESTED TOP ACTIONS

거래의 최종적인 커밋 여부와 상관없이 거래의 일부 업데이트가 커밋되기를 원할 때가 있다. 중첩된 상위 작업을 정의하는 일련의 작업을 수행하는 트랜잭션 실행은 다음 단계로 구성된다.

- (1) 현재 거래의 마지막 로그 기록의 위치 확인
- (2) 중첩된 상위 액션의 액션과 관련된 redo 및 실행 취소 정보 기록
- (3) 중첩된 상단 동작 완료 시 UndoNxtLSN이 단계(1)에서 위치가 기억된 로그 레코드를 가리키는 더미 CLR을 작성한다.

10. RECOVERY PARADIGMS

트랜잭션 롤백의 잠금 및 처리와 관련된 몇 가지 문제를 설명한다.

- 복구 재시작 중 선택적 재실행.
- 복구 재시작 중 재실행 전 작업을 실행 취소하십시오.
- 트랜잭션 롤백 중에 수행된 업데이트 기록 없음(즉, CLRS 없음)
- 인덱스 및 공간 관리 정보 변경 기록 없음.
- 페이지 자체에서 페이지 상태를 추적하여 기록된 업데이트와 관련시키지 않음(즉, 페이지에 LSNS 없음)

10.1 Selective Redo

트랜잭션 시스템이 장애 발생 후 재시작할 경우 일반적으로 로그의 2회 통과 후 데이터베이스 복구 업데이트를 수행한다. redo 패스 동안, 시스템 R은 커밋되고 준비된(즉, 의심스러운) 트랜잭션의 동작만 재실행한다.

10.2 Rollback State

장애가 발생했을 때 트랜잭션이 롤백될 수 있고 롤백 중에 수행된 업데이트의 일부 영향이 비휘발성 스토리지에 기록될 수 있으므로 트랜잭션 롤백의 진행 상태를 추적할 수 있는 방법이 필요하다. 논리적으로 실행 취소 정보의 로깅을 허용하면 높은 동시성이 지원될 수 있다. CLRS를 사용하여 롤백 중에 수행된 작업을 로깅하여 이 문제를 처리한다.

10.3 Space Management

유연한 스토리지 관리로 레코드 잠금을 할 때 해결해야 할 문제는 데이터 페이지의 레코드 삭제 또는 업데이트 중에 트랜잭션에 의해 해제된 공간이 공간 공개 트랜잭션이 커밋될 때까지 다른 트랜잭션에 의해 소비되지 않도록 하는 것이다. 유연한 스토리지 관리를 위해 한 페이지 내에서 물리적인(즉, 바이트 지향적인) 데이터 잠금 및 로깅을 수행하는 것은 바람직하지 않았다.

10.4 Multiple LSNs

LSN을 각 미니 페이지와 연결하여 각 미니 페이지의 상태를 별도로 추적하며, 리프 페이지 전체에 대한 LSN을 가지고 있다. 미니 페이지가 업데이트될 때마다 해당 로그 레코드의 LSN이 미니 페이지 LSN 필드에 저장된다. LSN 페이지는 미니 페이지 LSNs의 최대값과 동일하게 설정된다. 실행 취소 중에 로그 레코드의 업데이트를 실행 취소해야 하는지 여부를 결정하기 위해 로그 레코드의 LSN과 비교하는 것은 LSN 페이지가 아니라 미니 페이지 LSN이다.

11. OTHER WAL-BASED METHODS

- Buffer management
- Normal checkpointing
- Partial rollbacks
- Compensation log records
- Log record contents
- Page overhead
- Log passes during restart recovery
- Page forces during restart
- Restart checkpoints
- Restrictions on data

12. ARIES 속성

- (1)페이지 수준의 동시성 제어 및 여러 세부적인 잠금 기능보다 더 미세한 기능 지원.
- (2)재시작 및 정상 처리 중 유연한 버퍼 관리
- (3)최소 공간 오버헤드 - 페이지당 하나의 LSN만 가능.
- (4)재실행 또는 기록된 조치의 실행 취소를 보장하기 위해 데이터에 제약이 없다.
- (5)업데이트를 실행 취소하는 동안 수행된 작업이 원래 업데이트 중에 수행된 작업의 정확한 inverse일 필요는 없다.
- (6)작동 기록 및 새로운 잠금 모드 지원.
- (7)재실행 전용 및 실행 취소 전용 레코드도 수용할 수 있다.
- (8)부분 및 전체 트랜잭션 롤백 지원
- (9)여러 페이지에 걸친 객체 지원
- (10)언제든지 운영 체제에서 또는 운영 체제로 파일을 가져오거나 반환할 수 있도록 허용
- (11)거래의 일부 활동은 거래가 전체적으로 롤백되더라도 실행될 수 있다.
- (12)효율적인 체크포인트(복구 재시작 중 포함)
- (13)동일한 페이지에 액세스하는 전방 처리 and/or 롤백에서 여러 트랜잭션을 동시에 처리
- (14)트랜잭션 롤백 중 잠금 또는 교착 상태 없음
- (15)반복적인 실패 또는 중첩된 롤백에도 불구하고 재시작 중 한계 로깅입니다.
- (16)병렬 처리 및 선택적/지연적 처리를 통해 보다 빠른 재시작 가능

- (17)미디어 복구를 위한 퍼지 이미지 복사(아카이브 덤프)
- (18)시스템 재시작 후 실패자 트랜잭션 계속
- (19)재시작 또는 미디어 복구 중에 로그의 역방향 통과 한 번만
- (20)보상 로그 기록의 정보만 다시 실행하면 된다.
- (21)분산 트랜잭션 지원
- (22)부분 롤백을 사용한 트랜잭션 롤백 및 교착 상태 해결 중 잠김의 조기 해제

13. 요약

유연성과 효율성을 제공하는 특정 속성

- (1)이동된 레코드를 잠그지 않고 이동을 기록하지 않고도 스토리지 조각화를 방지할 수 있도록 레코드 레벨 잠금을 지원하고 한 페이지 내에서 레코드를 이동하십시오.
- (2)페이지당 상태 변수(로그 시퀀스 번호)를 하나만 사용하십시오.
- (3)동일한 거래의 나중의 행위에 대해 또는 전자가 일단 커밋한 후에 다른 거래의 행위에 대해 한 거래에 의해 발매된 스토리지를 재사용함으로써, 기록의 클러스터링 보존과 스토리지의 효율적 이용으로 이어진다.
- (4)트랜잭션을 전진 처리하는 동안 처음 수행되는 동작의 역행위는 원래 동작의 실행 취소(예: 스페이스 맵 페이지의 클래스 변경) 동안 수행된 동작과 다르다.
- 즉, 회복의 독립성을 가진 논리적 해지가 가능하다.
- (5)여러 개의 트랜잭션이 동일한 페이지에서 동시에 실행 취소될 수 있다.
- (6)특히 미디어 복구 중에 다른 페이지 또는 트랜잭션 상태와 관련된 로그 레코드와 독립적으로 각 페이지 복구.
- (7)필요한 경우, 시스템 장애 시 진행 중이던 거래의 계속.
- (8)데이터 가용성을 향상시키기 위해 새로운 트랜잭션 처리와 동시에 선택적 또는 지연된 재시작 및 실패자의 실행 취소.
- (9)트랜잭션의 부분 롤백.
- (10)페이지 내 변경사항의 작업 기록 및 논리적 기록.

Chaining, UndoNxtLSN 필드를 사용하여 정방향 처리 중에 작성된 레코드를 기록하는 CLR은 반복 기록 프로토콜도 준수할 경우 다음을 허용한다.

- (1)CLR의 실행 취소를 방지하여 CLR에 대한 CLR 쓰기를 방지한다. 이것은 또한 실행 취소 정보를 CLR에 저장할 필요가 없게 만든다.
- (2)두 번 이상 정방향 처리 중에 작성된 동일한 로그 레코드의 실행 취소 방지.

(3)트랜잭션이 롤백될 때 해당 개체에 대한 모든 업데이트가 취소되었을 때 개체의 잠금을 해제하는 기능. 이는 긴 트랜잭션을 롤백하거나 피해자를 부분적으로 롤백하여 교착 상태를 해결하는 동안 중요할 수 있다.

(4)시스템 R에서와 같이 로그 패치 적용과 같은 특별한 작업 없이 부분 롤백 처리.

(5)중첩된 상위 작업을 통해 필요할 경우, 거래 자체가 후속적으로 롤백되는지 커밋되는지 여부에 관계없이 거래에 의해 이루어진 일부 변경사항을 영구적으로 만든다.

기록을 반복하기 전에 분석 통과를 수행하면 다음이 허용된다.

(1)복구 패스를 다시 실행하고 실행 취소하는 동안 언제든지 사용할 수 있는 체크포인트.

(2)운영 체제에 파일을 동적으로 반환하여 데이터베이스 개체와 파일 간의 동적 바인딩 허용

(3)파일 관련 정보의 복구와 동시에 사용자 데이터의 복구

(4)재실행(redo)이 필요한 페이지를 식별하여 재실행(redo) 패스가 시작되기도 전에 비동기 병렬 I/O를 시작할 수 있도록 한다.

(5)예를 들어, 일부 빈 페이지가 해제되었다는 것을 알아채는 데 관한 dirty_pages 테이블에서 해당 페이지를 제거함으로써 일부 페이지의 재고를 피할 수 있는 기회를 활용한다.

(6)예를 들어, dirty_page가 비휘발성 저장소에 작성된 후 end_write 레코드를 작성하고 end_write 레코드가 발견될 때 dirty_page 테이블에서 해당 페이지를 제거함으로써 redo 중에 일부 페이지를 읽지 않도록 기회를 활용한다.

(7)선택적 또는 지연된 재시작, 재시작 후 실패자 트랜잭션의 지속, 새로운 트랜잭션 처리와 병행하여 실패자 트랜잭션의 실행 취소를 지원하기 위해 재실행 패스워드 동안 해당 트랜잭션에 대한 잠금이 다시 획득될 수 있도록 의심 상태 및 진행 중인 상태의 트랜잭션을 식별한다.

"끝"