論文
6-3-18

주기억장치DBMS의트랜잭션성능평가 PerformanceEvaluationofTransactionProcessinginMain MemoryDBMS

이규웅 (Lee, Kyu Woong)[1)]

요약

ALTIBASE™ 시스템은 메인 메모리를 주 저장장치로 사용하는 관계형 주기억장치 DBMS이다. 본 논문에서는 최근 데이터베이스 응용들의 요구사항으로 부각되고 있는 데이터베이스의 고가용성과 실시간 데이터베이스 시스템의 높은 트랜잭션 처리율을 동시에 보장하기 위하여 ALTIBASE™ 시스템의 구조 및 설계에 대하여 기술하고, 시스템의 설계 요소 기술에 대한 성능 분석 및 평가 결과를 보인다. 표준 성능평가 도구인 Wisconsin 벤치마크 테스트 결과 및 TPC-H 성능평가 결과를 통해 전체 시스템의 가용성을 입증한다. 또한 인덱스 관리 기법 및 트랜잭션 처리 기법에 대하여 기존 다른 기법과의 성능 비교를 통해 설계에 적용된 요소기술의 우수성을 입증한다.

Abstract

ALTIBASE is the relational main memory DBMS that enables us to develop the high performance and fault tolerant applications. It guarantees the short and predictable execution time as well as the basic functionality of conventional disk-based DBMS. We present the overview of system architecture and the performance analysis with respect to the various design choices. The assorted experiments are performed under the various environments. The results of TPC-H and Wisconsin benchmark tests are described. We illustrate the various performance comparisons under the various index mechanisms, the replication models, the transaction durabilities, and the application structures. A performance study shows the ALTIBASE system can be applied to the wide area of industrial DBMS fields.

키워드 : 주기억장치DBMS, 트랜잭션(Transaction), 인덱스(Index), 벤치마크(Benchmark), 성능평가(Performance Evaluation)

논문접수 : 2005. 5. 30.
심사완료 : 2005. 7. 1.

1) 정회원 : 상지대학교 컴퓨터정보공학부

1.서론

대부분 디스크 기반(Disk Resident) 데이터베이스 관리 시스템(DRDBMS)들은 개개의트랜잭션에 대한 빠른 응답 시간을보장하기 보다 전체적인 트랜잭션 처리량의 향상을 위해 설계되었다. 즉, 하나의 트랜잭션에 대한 시간제약사항을 만족하기보다는 트랜잭션의 처리율을 높여 전체적인 성능 향상을 위한 방법으로 설계되었다. 따라서 이러한 디스크 기반의 데이터베이스 시스템은 실시간 응용 분야에 적합하지 못하다.

이러한 수요를 만족하기 위해 기존 디스크 기반 데이터베이스 시스템들은 풍부한 메모리 공간을 활용하여 모든 데이터를 메모리에 적재하여 사용할 수 있는 대체방안을 제시하였다. 그러나 이러한 방법은 순수한 주기억 상주 데이터베이스 관리시스템(MMDBMS)과 그 기능 및 성능 면에서 다르다. 예를 들어, 충분한 주기억 공간을 갖는 디스크 기반데이터베이스 시스템의 경우, 데이터 접근을 위해서는 실제 물리적 디스크 주소가 결정되어야 하고, 이 데이터 블록이 버퍼에 위치하는지 검사한 후, 해당 데이터블록이 프로세스에게 전달된다. 반면에 순수한 주기억 상주 데이터베이스 시스템에서는 모든 데이터가 주기억장치에 있음이 보장되므로, 모든 데이터의 요구는단순히 가상 기억장치의 주소로 직접 변환하여 사용하면 된다. 즉, 모든 저장장치가 계층적 구조에서 평면구조로 변환되어 버퍼 관리기를 통할 필요가없으므로, 그 만큼 성능상에서 유리하다. 디스크 공간을 주 저장장치로 설계된 시스템에서는 충분한 주기억장치 공간을 갖는다 하여도 순수 주기억 상주 데이터베이스 시스템에서의 성능 보다 우수하지 못함을 이미 다른 연구에서도 보였다[1, 2].

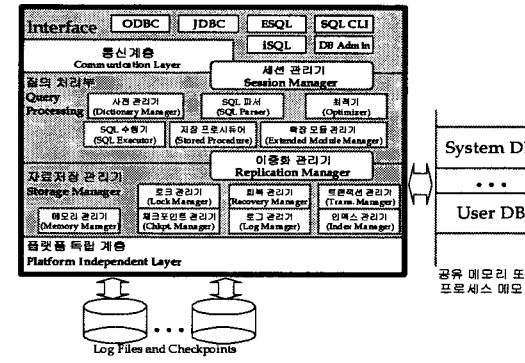
주기억 상주 데이터베이스 시스템은 기존의 데이터베이스 시스템에서 지원되는 병행수행제어, 인덱스 관리, 로그 및 회복 기능 등 모든 기능을 지원해야 하고 아울러 기본 저장장소로서 주기억장치를 사용한다. 그러나, 기존 시스템에서 사용하던 기법들은 기본 저장장소가 기억장치로 변환되면서 직접적으로 적용하기 어려운 점이 많다[3]. 본 ALTIBASE 시스템은 이러한 문제점들을 해결하여 실시간 응용 및 고성능 트랜잭션 처리율을 요구하는 산업 분야에실제 적용될 수 있는 주기억상주 DBMS로 개발되었다.

본 논문에서는 ALTIBASE의 자료저장 관리자가 고성능을 위해 사용한 여러 기법들을 소개하고 적용된 각 기술에 대해 성능평가를 통해 우수성을 보인다. 본 논문의 구성은 다음과 같다. 2장에서 시스템의 전체적인 구조를 설명하고, 적용된 데이터베이스 기법중 평가대상에 대한 기술을 중점적으로 기술한다. 3장에서는 인덱스 및 트랜잭션 처리 기술을 중심으로 타 기법 및 타 시스템과의 성능평가방법 및 결과를 보이고 이를 분석한다. 끝으로 4장에서 본 논문의 결론을 맺는다.

2.시스템구조및평가주요기술

2.1전체적인시스템구조

ALTIBASE는 주기억장치를 사용하는 관계형 데이터베이스 시스템으로서, 범용적 응용뿐만 아니라 특정 실시간 응용에도 적합한 클라이언트-서버 구조와 응용 내장형



[그림 1] ALTIBASE 시스템 구조
[Fig. 1] ALTIBASE System Architecture

구조, 멀티 쓰레드 구조, 그리고 서버 연결 풀 구조를 제공한다. 클라이언트-서버 구조를 기반으로 응용 프로그램은 다양한 통신 방법을 사용하여 서버에 접속할 수 있으며 또한 응용 프로그램을 데이터베이스 서버에 내장함으로써 응용 프로그램과 서버간의 통신 비용을 제거하는 응용 내장형 구조를 지원한다. 멀티 쓰레드 구조를 기반으로 프로세스간 문맥 교환 시간을 제거하였으며, 사용자의 증가에 따른 시스템 자원을 감소할 수 있는 구조를 지원한다. 또한 서버 기능을 갖는 다수의 시스템 쓰레드를 서비스 쓰레드 풀로 구성하여 서버 연결 풀과 연동되게 구성하여 한정된 개수의 쓰레드를 통하여 다수의 동시 사용자에게 서비스를 극대화할 수 있는 서버 연결 풀 구조를 갖는다.

ALTIBASE 시스템의 구조는 <그림 1>과 같이 인터페이스 부분, 통신 계층, 질의 처리부, 그리고 자료저장 관리기로 분류할 수 있다. 그 중 데이터베이스 시스템의 가장 핵심이 되는 서비스를 관리하고 있는 자료저장 관리기는 다시 로크 관리기, 회복 관리기, 트랜잭션 관리기, 메모리 관리기, 체크포인트 관리기, 로그 관리기 및 인덱스 관리기로 구성된다.

2.2평가대상요소기술

2.2.1 인덱스 관리 기법

인덱스 구조는 전체적인 시스템의 성능에 영향을 미치는 주요한 요소이다. 디스크 기반의 데이터베이스시스템 구조에서 가장 일반적으로 많이 사용되는 인덱스 구조는 B-트리 나 B+-트리이지만, 주기억장치 데이터베이스시스템을 위해 개발된 T-트리 인덱스 구조는 실시간 데이터베이스 시스템과 같은 빠른 응답을 요구하는 시스템에 많이 사용된다. 이 인덱스 구조는 B-트리 보다 우수한 성능을 이유로 Starburst[4] 시스템 이나 Dali[3,5] 시스템과 같은 몇몇 상용 시스템에서 사용되고 있다. 주기억장치 시스템 인덱스 구조의 I/O 병목현상이 없기 때문에 병행 접근에 따르는 성능 저하 문제가 성능평가의 가장 주된 요소가 된다. 따라서 T-트리 구조의 병행 접근 성능이 가장 신중하게 평가되어야 함에도 불구하고 기존 B-트리와의 병행 접근에 관한 성능평가가 되지 않았다. 참고문헌 [6]에 따르면 병행접근에 따르는 로킹의 오버헤드 때문에 T-트리가 B-트리보다 좋은 성능을 보이지 않고 있음을 나타냈다.

그러므로, 본 시스템에서는 T-트리 인덱스 구조에 멀티 버전 기술을 제안하고 T-트리의 인덱스 노드에 대하여 버져닝 기술을 통한 동시 접근을 가능하게 하여 인덱스 트리의 변경 작업에 대한 병행성을 증가시키고 로킹에 대한 오버헤드를 없앴다. 따라서 물리적인 인덱스 노드의 버전을 통해 로킹 없이 인덱스 트리를 탐색할 수 있다. 또한 응용의 특성에 따라 선택적으로 적용할 수 있도록 T-트리와 B^+-트리를 모두 제공하고 있다. 3장에서 두 가지 인덱스 기법에 대한 성능평가를 보인다.

2.2.2 트랜잭션의 영속성

로그 버퍼의 내용은 회복시 필수적인 내용이므로 디스크의 동기화를 필수적으로 수행하여야 한다. 기본적인 로그 버퍼로서 메모리 맵드 파일(memory mapped file)을 제공한다. 메모리 맵드 파일은 로그 버퍼로 사용하는 경우 디스크의 입출력이 느리거나 운영체제의 부하가 많은 경우 시스템 전체의 성능 저하를 유발할 수 있게 된다. 따라서 트랜잭션의 영속성 수준에 따라 메모리 버퍼와 메모리 맵드 파일 두 종류의 로그 버퍼를 선택적으로 사용할 수 있게 하여, 성능대비 안정성을 유도할 수 있다. 트랜잭션의 영속성은 모두 5가지 수준으로 제공되며 각 수준에 따른 로그 버퍼 및 기능은 <표 1>과 같다.

<표 1> 트랜잭션 영속성 수준

트랜잭션 영속성 수준	로그버퍼 및 디스크동기화	기능 설명
Level 1	메모리 버퍼	로그는 메모리 버퍼에만 반영되며 변경 내용을 디스크에 수행하지 않는다.
Level 2	메모리 버퍼 디스크 로그 파일 Log Sync Thread 동작	로그는 메모리 버퍼에 반영되고 Log Sync Thread에 의해 로그 파일에도 주기적으로 반영된다. 디스크의 로그 파일에 로그가 반영된 것을 보장하기 전에 트랜잭션의 완료가 선언되므로 완료 트랜잭션의 영속성을 보장하지 않는다.
Level 3	메모리 맵드 파일	모든 로그는 디스크에 반영된다. 운영체제의 파일 버퍼가 적용되므로 트랜잭션의 영속성을 보장한다.
Level 4	메모리 버퍼 메모리 맵드 파일 Log Sync Thread 동작	로그는 메모리 버퍼에 반영되고 Log Sync Thread에 의해 메모리 맵드 파일에도 주기적으로 반영된다. 메모리 맵드 파일에 로그가 기록된 것을 보장하기 전에 트랜잭션의 완료가 선언되므로 완료 트랜잭션의 영속성을 보장하지 않는다
Level 5	메모리 버퍼 디스크 로그 파일 Log Sync Thread 동작	로그는 메모리 버퍼에 반영되고 Log Sync Thread에 의해 로그 파일에도 주기적으로 반영된다. 트랜잭션의 완료는 완료 로그를 포함한 모든 로그가 디스크 로그 파일에 기록된 후에 선언된다.

각 트랜잭션 영속성 수준에 따른 성능평가의 결과 및 분석을 통해 영속성이 성능에 미치는 영향을 알아본다.

3.트랜잭션처리기술성능평가

본 장에서는 ALTIBASE 시스템의 트랜잭션 수행에 대한 성능평가를 보인다. 특히 여러 종류의 시스템 부하에 따른 TPS(Transaction per Second) 값을 집중적으로 평가하여 시간제약적인 응용 분야에 적합한 시스템임을 보인다. 본 절에서 수행한 모든 실험은 750MHz CPU 4개와 4G 바이트 메모리를 보유한 "HP RP5470" 플랫폼과 "HP.UX 11.0" 운영체제 하에서 수행하였다. 또한 실험 트랜잭션은 시스템에서 제공하는 저장 프로시듀어 인터페이스를 사용하여 구현하였다. 따라서 트랜잭션의 데이터베이스 요구는 모두 질의 처리기를 거쳐 최적 수행 계획에 따라 수행되며, 네트워크 지연에 따른 간섭은 실험에 평가되지 않는다. 실험에 사용된 트랜잭션은 모두 4종류, 검색, 삽입, 변경 그리고 삭제 트랜잭션이며, 대상 테이블은 *"number"*, *"real"*, *"varchar"* 등의 여러 가지 속성들로 구성되는 총 20개의 속성을 갖는 단일 테이블이다. 동시 사용자 수는 실험에 따라 단일 사용자에서 50명의 사용자로 변화를 주었으며, 레코드의 개수는 실험에 따라 총 10,000개에서500,000개의 레코드로 구성하였다. 검색 트랜잭션의 경우 모든 속성을 검색하도록 수행되었으며, 삽입 및 변경 트랜잭션들도 모든 속성들에 대해 삽입 및 변경 작업을 수행하도록 구성하였다. 모든 트랜잭션의 조건식은 인덱스를 갖는 속성에 대해 조건식을 작성하였다.

3.1시스템전체의성능평가

시스템 전체의 대표적인 실험결과는 아래 <표 2>에 보인 바와 같이 단일 사용자 환경에서 수행한 4종류의 트랜잭션들에 대한 TPS 결과이다.

<표 2> 단일 사용자 환경에서의 TPS 측정값

삽입 트랜잭션	변경 트랜잭션	검색 트랜잭션	삭제 트랜잭션
6,134.97	4,405.29	29,411.76	12,345.68

단위 : TPS(Transactions Per Second)

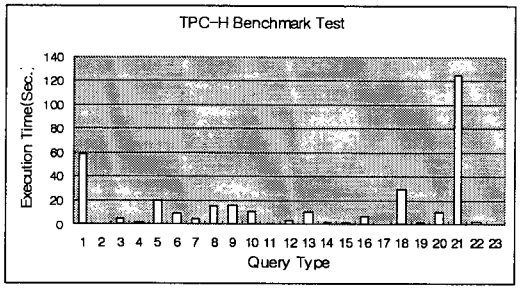
[그림 2]는 표준 데이터베이스 성능평가기관의 TPC-H 실험결과를 보이고 있다. HP 플랫폼상에서 TPC-H의 질의어 22개를 실험하여

그 결과값을 측정하였다. 또한 [그림 3]은 벤치마크 테스트 기관인 위스콘신 대학의 벤치마크 테스트를 통한 트랜잭션 성능 평가 결과를 보이고 있다. [그림 2]및 [그림 3]에서 보인 트랜잭션의 실행시간은 디스크 기반의 시스템보다 당연히 우수한 성능을 보이고 있으며, 주기억 상주형 타 시스템 보다 빠른 실행시간을 제시하고있다.

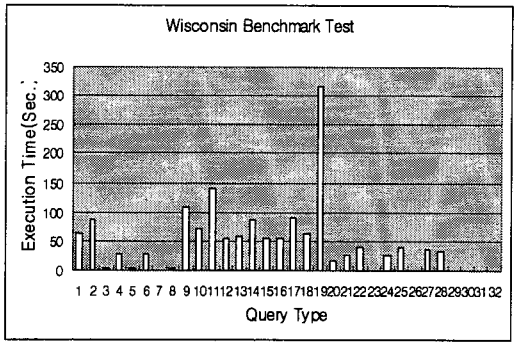
3.2인덱스기법에대한성능평가및분석

제안하는 시스템은 B^+-트리와 T-트리를 모두 지원한다. 따라서 본 절에서는 각각의 인덱스 기법을 제안하는 시스템에 적용하였을 때 성능을 측정한다.

인덱스의 동시 접근성능을 중점적으로 평가하기 위해 동시 사용자 수를 증가하였을 때, 다양항 종류의 트랜잭션에 대한 성능을 B^+-트리와 T-트리 인덱스 구조에 대하여 측정하였다. 실험의 본 목적을 위해 인덱스 생성은 테이블의 가장 기본적인 *Char* 타입에 대하여 하였으며, 다른 파라미터들도 기본적인 값으로 설정하였다. [그림 3]은 동시 사용자수가 16으로 증가할 때, 각 인덱스 기법을 적용한 트랜잭션의 초당 트랜잭션 실행 개수를 나타내고 있다. 점선으로 나타난 그래프는 B^+-트리 인덱스를 사용한 성능 결과이고, 실선은 T-트리 구조를 사용한 결과이다. 참고문헌 [6]에서 지적된 바와 같이, 동시 접근성을 고려한 성능 평가에서 T-트리는 B-트리보다 우수한 성능을 보이지 못하고 있다.

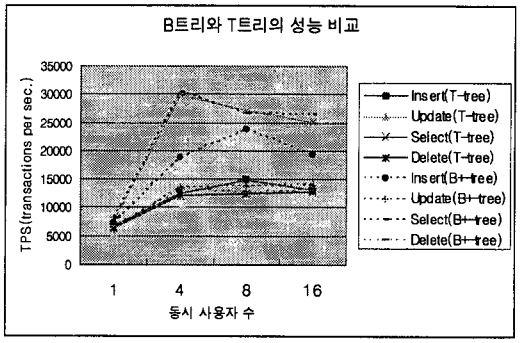


[그림2] TPC-H 성능평가 결과
[Fig. 2] TPC-H Performance Results



[그림2] Wisconsin 벤치마크 성능평가 결과
[Fig. 3] Wisconsin Benchmark Results

특히 적용된 T-트리 기법은 앞 장에서 설명한 바와 같이 동시 접근성을 증가시키기 위하여 인덱스 노드에 대한 멀티 버전 기법을 적용하였음에도 불구하고 [그림 3]의 성능 결과 처럼 다양한 종류의 모든 트랜잭션 경우에도 다소 낮은 성능을 보이고 있다.



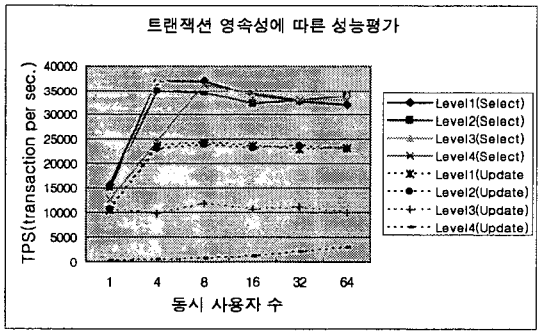
[그림3] T-트리와 B-트리 인덱스의 비교
[Fig. 3] T-tree and B^+-tree Index Structure

3.2트랜잭션영속성의성능평가

데이터베이스 시스템의 특성에 따라 로그 버퍼의 기능으로 인한 디스크 입출력의 오버헤드가 전체 시스템의 성능을 크게 좌우할 수 있다. 본 시스템에서는 앞서 설명하였듯이 5가지 수준의 서로 다른 영속성을 제공하므로써, 이러한 로그 버퍼의 기능을 제어하여 성능을 향상시킬 수 있다. 제공되는 트랜잭션 영속성 수준이 시스템 성능에 미치는 성능 영향을 [그림

4]에 보이고 있다.

이 성능평가에서는 검색과 변경 두가지 종류의 트랜잭션과 4가지 영속성 수준에 대해 총 8개의 성능을 측정하였으며, 동시 사용자수에 대한 변화율를 측정하였다. 영속성수준 5는 4와 유사한 성능을 보이므로 실험에서 제외하였다. 성능 평가 결과에 따르면 로그 버퍼를 메모리 버퍼로 사용하는 경우가 디스크 버퍼를 사용하는 경우보다 월등한 성능 우세를 보이는 것을 알 수 있으며, 메모리 맵드 파일을 이용하여 로그 버퍼를 사용하는 경우와는 유사한 성능을 보이고 있다. 또한 검색 트랜잭션은 로그 기록의 내용이 디스크에 동기화 되지 않아도 되므로 로그 버퍼의 종류에 따라 성능 차이의 변화가 심하지 않음을 알 수 있으며, 동시 사용자 수에 대한 성능 저하만이 있음을 알 수 있다.



[그림4] 트랜잭션 영속성에 따른 성능평가
[Fig.4] Performance of Transaction Durability

4.결론및향후연구

본 논문에서는 주기억 상주 데이터베이스 시스템인 ALTIBASE 시스템에 대한 설계 고려사항과 주요 평가 대상 기술을 언급하였으며, 각 요소 기술들의 성능을 측정하고 분석하였다. 시스템 전체성능에 많은 영향을 미치는 인덱스 구조와 트랜잭션 처리 방법, 이중화 구조에 대하여 다양한 기술을 적용하였을 때의 변화를 측정하여 각 요소기술이 시스템에 미치는 영향을 분석하였다. 또한 표준 벤치마크 테스트 도구들을 활용하여 전체 시스템의 성능을 측정하여 타 시스템과의 비교를 유도하였고 결과적으로 비교우위에 있음을 알 수 있었다. 본 시스템은 대용량 데이터 저장을 위해 디스크 기반 시스템과의 혼합을 향후과제로 두고 있다.

참고문헌

[1] P. Bohannon, J. Parker, R. Rastogi, S. Seshadri, A. Sil-berschatz, and S. Sudarshan, "Distributed Multi-Level Re-covery in Main-Memory Databases," Proc. of the Interna-tional Conference on Parallel and Distributed Information Systems, 1996.

[2] H. Garcia-Molina and K. Salem, "Main Memory Database Systems : An Overview," IEEE Transactions on Knowl-edge and Data Engineering, 4(6), 1993.

[3] P. Bohannon, D. F. Lieuwen, R. Rastogi, A. Silberschatz, S. Seshadri, and S. Sudarshan, "The Architecture of the Dali Main-Memory Storage Manager," Multimedia Tools and Applications, 4(2), 1997.

[4] T. J. Lehman and M. J. Carey. "A study of index structures for main memory database management systems" In Proceedings of the 1992 International Conference on Very Large Database, pages 294303, 1992.

[5] Philip Bohannon, James Parker, Rajeev Rastogi, S. Seshadri, Abraham Silberschatz, and S. Sudarshan, "Distributed multi-level recovery in mainmemory databases", In Proc. of the International Conference on Parallel and

Distributed Information Systems, pages 4455, 1996.

[6] Hongjun Lu, Yuet Yeung Ng, and Zengping Tian, "T-tree or b-tree: Main memory database index structure revisited", In Australasian Database Conference, pages 6573, 2000.

[7] Bettina Kemme and Gustavo Alonso, "A New Approach to Developing and Implementing Eager Database Replication Protocols", ACM Transactions on Database Systems, 25(3), Sep., 2000.

[8] Jim Gray, Pat Helland, Patrick O'Neil, and Dennis Shasha, "The Dangers of Replication and a Solution",Proc. of the AMC SIGMOD International Conference on Mangement of Data, 1996.

이규웅

1990년 한국외국어대학교 전자계산학과(이학사)
1992년 서강대학교 대학원 전자계산학과(공학석사)
1998년 서강대학교 대학원 전자계산학과(공학박사)
1998년-2000년 8월 한국전자통신연구원 인터넷서비스 연구부 선임연구원
2000년 9월 ~ 현재 상지대학교 컴퓨터.정보공학부 조교수
관심분야 : 자료저장 시스템, 분산및 실시간 DB