

# 데이터베이스 워크로드 분석 : 실험적 연구

오 정 석\* · 이 상 호\*\*

## 요 약

데이터베이스 시스템을 효율적으로 사용하기 위해 데이터베이스 관리자는 데이터베이스 시스템의 성능 특징들을 알아야 한다. 데이터베이스 시스템에서 자원 사용은 워크로드에 따라 다르게 나타난다. 본 논문의 목적은 상이한 워크로드에서 데이터베이스 튜닝에 도움이 될 수 있도록 데이터베이스 시스템의 성능 특징 분석하고 식별하는 것이다. 이를 위해, OLTP 환경을 나타내는 TPC-C 워크로드와 웹기반의 전자상거래 환경을 나타내는 TPC-W 워크로드에서 14개의 성능 지표에 의해 결정되는 자원 사용 형태를 분석하고 4개(데이터 버퍼, 개인 메모리, I/O 프로세스, 공유 메모리)의 자원 할당 변경에 따른 자원 사용 형태의 변화를 분석한다. 분석에 대한 결과로서 14개 중 8개의 성능 지표는 워크로드에서 성능차이를 보이고, 데이터 버퍼 자원의 변경은 데이터베이스 시스템에 영향을 준다. 본 논문의 결과는 데이터베이스 시스템 자동 튜닝의 기초 자료로서 사용될 수 있다.

## Database Workload Analysis : An Empirical Study

Jeong Seok Oh\* · Sang Ho Lee\*\*

### ABSTRACT

Database administrators should be aware of performance characteristics of database systems in order to manage database system effectively. The usages of system resources in database systems could be quite different under database workloads. The objective of this paper is to identify and analyze performance characteristics of database systems in different workloads, which could help database tuners tune database systems. Under the TPC-C and TPC-W workloads, which represent typical workloads of online transaction processing and electronic commerce respectively, we investigated usage types of resource that are determined by fourteen performance indicator, and are behaved in response to changes of four tuning parameters (data buffer, private memory, I/O process, shared memory). Eight out of the fourteen performance indicators clearly show the performance differences under the workloads. Changes of data buffer parameter give a influences to database system. The tuning parameter that affects the system performance significantly is the database buffer size in the both workloads.

**키워드 :** 데이터베이스 워크로드(Database Workload), 성능 지표(Performance Indicator), 데이터베이스 파라미터(Database Parameter)

### 1. 서 론

효율적인 데이터베이스 시스템의 관리는 데이터베이스 시스템 튜닝에 의해 수행될 수 있다. 데이터베이스 튜닝은 지식 집약적인 분야로서 데이터베이스 시스템 자원이 활용되는 방식을 개선하여 보다 향상된 성능을 낼 수 있는 작업들을 의미한다[10]. 데이터베이스 시스템의 성능은 성능 지표(performance indicator)를 이용하여 표시할 수 있다. 데이터베이스 튜닝이나 성능평가를 수행할 때, 성능 지표로는 주로 응답시간(response time), 단위당 처리량(throughput), 자원 사용(resource usage) 정보 등이 이용되었다[1, 6, 11-13]. 응답 시간과 단위당 처리량은 시스템의 전반적인 성능을 판단할 때 유용하게 사용될 수 있으나, 시스템의 전

체 성능에 영향을 주는 국지적인 문제를 효과적으로 표현하지 못한다. 자원 사용 정보는 시스템의 전체적인 성능을 표현하지 않지만, 시스템의 특정 부분에 국한되는 지역적인 문제를 효과적으로 파악하게 한다[3].

데이터베이스 응용 분야가 다양화되고 복잡해짐에 따라, 관리자는 응용 분야의 독특한 워크로드를 고려하여 시스템을 관리하여야 한다[2, 14]. 일반적으로 워크로드는 시스템에 부하를 가할 수 있는 요소의 집합이며, 데이터베이스 워크로드는 데이터베이스 질의 집합을 의미한다. 데이터베이스 워크로드를 구성하는 데이터 연산 및 데이터 크기 등은 시스템 자원에 상이한 부하를 줄 수 있다.

데이터베이스 워크로드에 관한 연구는 문헌상에서 발견할 수 있다. [14]는 관계형 데이터베이스에서 워크로드 특성을 분석하는 도구인 REDWAR(relational database workload analyzer)를 개발하였다. REDWAR는 워크로드가 수행되는 동안 SELECT, DELETE, UPDATE, OPEN, CLOSE

\* 본 연구는 숭실대학교 교내 연구비 지원으로 이루어졌습니다.

† 준 회원 : 숭실대학교 대학원 컴퓨터학과

\*\* 정 회원 : 숭실대학교 컴퓨터학부 교수

논문접수 : 2004년 1월 9일, 심사완료 : 2004년 5월 3일

등의 정적 및 동적 SQL 문장들의 사용 비율과 SELECT, FROM, WHERE, ORDER BY 등의 구문에서 사용되는 열(column)에 대한 통계를 제공한다. [9]는 운송, 전화, 은행 업무에 사용되는 실제 워크로드에 대하여 잠금 수행 및 잠금 해제에 대한 빈도, 잠금의 종류, 잠금 간의 연관 관계를 측정하여 동시성 제어에 관한 통계적인 분석을 제공하였다. [2]는 복잡한 워크로드에서 메모리 풀 크기 및 다중 프로그래밍 단계(multiprogramming level)를 동적으로 할당하는 알고리즘을 설계하였다. 이 알고리즘은 가상 실험을 통해 미리 정의된 응답시간 안에 질의/트랜잭션이 수행되도록 동적으로 메모리 할당량과 다중 프로그래밍 단계를 조절한다. 기존 연구는 워크로드의 구성 분포 분석 또는 국부적인 자원에 대한 사용 형태 분석이 주류를 이룬다.

데이터베이스 관리자는 워크로드 종류에 따라 독특한 시스템 부하를 활용하면 데이터베이스 시스템 관리 및 튜닝을 효과적으로 수행할 수 있다[7]. 예를 들어, [4]는 워크로드에 따라 정렬 및 커서 정보가 적재되는 영역인 개인 메모리(private memory) 크기의 조절 방법을 제시한다. 구체적으로 OLTP 환경에서는 개인 메모리 크기를 작업 메모리(working memory) 크기의 20%로 설정하고, DSS 환경에서의 개인 메모리 크기는 작업 메모리 크기의 80%로 설정하라고 권고한다. 작업 메모리는 질의에 대한 구문 분석 및 실행계획을 적재하는 공유 메모리(shared memory), 개인 메모리, 기타 부분을 포함하는 메모리 자원을 일컫는다.

본 논문의 목적은 데이터베이스 튜닝을 효과적으로 수행하기 위하여 특정 워크로드 환경에서 시스템 자원 사용 형태를 정량적으로 분석하는 것이다. 표준화된 데이터베이스 워크로드 환경을 구축하기 위하여 본 논문에서는 국제 표준 데이터베이스 성능평가론으로 인정받고 있는 TPC-C와 TPC-W 워크로드를 사용한다[11, 12]. TPC-C는 도매 업체의 재고 관리 시스템을 가상 운영할 수 있는 OLTP 환경의 대표적인 워크로드를 제공하는 성능평가이며, TPC-W는 인터넷 전자 서점의 전자 상거래 시스템을 모델링 하는 웹 기반 환경의 워크로드를 제공하는 성능평가이다.

본 논문에서 우리는 TPC-C와 TPC-W 워크로드 환경에서 성능 지표를 이용하여, 첫째로 다수의 시스템 자원 사용 형태를 분석하고, 둘째로 자원 할당 변경에 따른 자원 사용 형태의 변화를 분석하였다. 워크로드 분석은 총 14개의 성능 지표가 사용되었으며, 자원 사용 특성에 차이점을 보이는 8개의 성능 지표를 중심으로 수행되었다. 본 논문의 결과로는 상이한 데이터베이스 워크로드에서 자원 사용 형태와 그 변화를 식별해 하는 성능지표들을 제공하며 튜닝 기초 자료로 활용될 수 있음을 이야기한다.

본 논문의 구성은 다음과 같다. 2장은 TPC-C와 TPC-W 성능평가를 워크로드 관점에서 설명한다. 3장은 실험 환경 및 전략에 대해 기술하며, 4장은 분석 실험에서 수집된 성

능지표들을 통해 워크로드 특징을 분석한다. 5장은 분석된 내용에 대해 결론을 맺고, 향후계획을 제시한다.

## 2. TPC-C와 TPC-W 분석

TPC-C는 9개의 테이블에 대해 다섯 종류의 트랜잭션을 수행하며 웨어하우스의 수를 변경하여 시스템의 부하를 조절한다. TPC-W는 8개의 테이블에 대해 14개의 웹 상호작용을 수행하며 에뮬레이터 브라우저(emulated browser)의 수를 통해 시스템 부하를 조절한다.

TPC-C의 트랜잭션과 TPC-W의 웹 상호작용에 포함된 질의 집합은 질의 형태와 구성 분포가 다르다. 각 성능평가를 구성하는 질의어의 비율을 분석하기 위하여 각 성능평가를 5회 모의 시험을 하였다. <표 1>은 TPC-C와 TPC-W에서 발생하는 총 질의어에 대하여 검색, 갱신, 데이터 정의(data definition) 질의의 구성 비율을 보인다. TPC-C 워크로드는 전체 질의에 대해 검색 질의와 갱신 질의가 비슷한 분포를 나타낸다. TPC-W 워크로드는 검색 질의가 갱신 질의보다 약 50% 이상 출현하며, 검색 질의가 전체 질의의 약 3/4을 차지하고 있다.

<표 1> 수행된 질의 비율

워크로드	검색 질의	갱신 질의	DDL
TPC-C	45.1%~51.8%	48.2%~54.8%	-
TPC-W	76.2%	23.76%	0.04%

<표 2>는 TPC-C와 TPC-W에서 발생하는 총 데이터 처리량에서 읽기 누적량과 쓰기 누적량에 대한 비율을 보인다. TPC-C 워크로드는 총 데이터 처리량에서 약 19% 정도의 쓰기 누적량이 발생되나 TPC-W 워크로드는 5% 정도의 쓰기 누적량이 발생된다. 또한 읽기 누적량에 대한 쓰기 누적량의 비율은 TPC-C에서 약 23%를 보이며 TPC-W에서 약 5.2%를 보인다.

<표 2> 총 데이터 처리량에서 읽기/쓰기 누적량의 비율

워크로드	쓰기 누적량의 비율	읽기 누적량의 비율
TPC-C	18.7%	81.3%
TPC-W	5%	95%

## 3. 실험 환경 및 전략

워크로드 특성 분석은 각 워크로드 환경에서 성능 지표에 의한 자원 사용량과 파라미터 변경에 따른 성능지표들의 변화에 초점을 둔다. 이를 위해, 실험 대상 자원의 선정, 자원의 크기 조절 방법, 성능 지표의 선별 등에 대한 전략이 필요하다.

실험 대상 자원은 시스템의 성능에 많은 영향을 미치며,

자원의 크기를 변경할 수 있는 것으로 한다. 본 논문에서는 이러한 자원들 중에서 데이터 버퍼, 개인 메모리, 공유 메모리, I/O 프로세스를 대상으로 한다[4, 5, 15].

메모리 관련 자원들은 [2]를 참조하여 선정하였다. [2]는 데이터베이스 성능에 영향을 주는 메모리 자원을 데이터 버퍼와 작업 메모리로 구분하였다. 데이터 버퍼는 자주 사용되는 데이터를 오랫동안 보관한다. 적절한 버퍼의 크기는 디스크 접근 횟수를 감소시켜 질의 결과에 대한 응답시간을 감소시킨다. 작업 메모리는 사용자 질의에 대한 처리 정보 등을 오랫동안 보관하여 질의 구문 분석의 최소화 및 질의 데이터 처리의 최적화를 목적으로 사용된다.

작업 메모리를 사용 목적 및 공유 여부에 따라 개인 메모리와 공유 메모리로 세분화할 수 있으므로 본 논문에서는 메모리 자원을 데이터 버퍼, 개인 메모리, 공유 메모리로 구분하였다. 공유 메모리는 일반적으로 자주 사용되는 질의에 대한 구문 분석 및 실행 계획 등이 저장되고, 공유 메모리의 데이터는 다른 사용자에게 공유된다. 개인 메모리는 질의에서 처리되는 조인, 정렬, 커서 등의 정보가 저장되고 개인 메모리의 데이터는 다른 사용자에게 공유되지 않는다.

데이터 버퍼, 공유 메모리, 개인 메모리, I/O 프로세스 자원에 대한 크기 조절은 실험에 사용된 데이터베이스 시스템에서 제공하는 db\_cache\_size, shared\_pool\_size, pga\_aggregate\_target, dbwr\_io 파라미터를 사용하여 수행하였다.

또한 파라미터의 변경은 성능에 영향을 줄 수 있으므로 주의 깊게 설정해야 한다. 예를 들면, 버퍼 및 캐시 관련 파라미터들은 단편화(fragmentation)를 감소시키기 위해 블록 단위로 확장하는 것이 좋다. 본 논문에서 파라미터의 초기값 및 증가값은 데이터베이스 시스템에서 제공하는 파라미터 기본값을 참조하여 설정 및 변경하였으며, 자세한 변경 범위는 <표 3>에 기술되어 있다.

<표 3> 파라미터 값의 변경 범위

파라미터 항목	초기값	증가값	최대값
db_cache_size	32MB	32MB	480MB
shared_pool_size	32MB	32MB	480MB
pga_aggregate_target	20MB	20MB	300MB
dbwr_io	1	1	15

파라미터 변경 방식은 변경 대상이 되는 파라미터만이 변경되며 다른 파라미터들은 초기값으로 설정된다. 예를 들어, 데이터 버퍼의 크기 변경에서 데이터 버퍼 크기를 64MB로 변경할 때 db\_cache\_size의 값은 64MB로 설정되고 다른 3개의 파라미터의 값은 초기값으로 설정된다.

성능 지표는 실험에 사용된 데이터베이스 시스템에서 제공하는 다양한 지표 중 선정된 파라미터의 변경에 영향을

받을 수 있는 지표를 14개 선별하였으며 종류는 다음과 같다. 참고로 자세한 성능 지표의 설명과 계산 원리는 [4, 8]에 설명되어 있다.

- 버퍼 적중률 : 검색하는 데이터가 데이터 버퍼에 존재할 확률을 의미한다.
- 공유 메모리 적중률 : 검색하는 질의 정보가 공유 메모리에 존재할 확률을 의미한다.
- 시스템 카탈로그 적중률 : 검색하는 카탈로그 정보가 시스템 카탈로그에 존재할 확률을 의미한다.
- 래치 경합 비율 : 파싱을 위해 수행된 CPU 시간을 파싱 수행에 경과된 시간으로 나눈 비율을 의미한다.
- 메모리 정렬 비율 : 메모리에서 데이터가 정렬될 비율을 의미한다.
- 메모리 파싱 비율 : 파싱을 수행할 때 공유 메모리에 존재하는 데이터를 읽어 파싱을 수행할 비율을 의미한다.
- 데이터 변경률 : 디스크에서 메모리로 읽은 데이터가 변경될 비율을 의미한다.
- 데이터 버퍼 읽기량 : 디스크에서 데이터 버퍼로 읽는 데이터의 용량을 의미한다.
- 데이터 비버퍼 읽기량 : 디스크에서 데이터 버퍼가 아닌 메모리의 특정 부분으로 읽는 데이터의 용량을 의미한다.
- 데이터 버퍼 쓰기량 : 데이터 버퍼에서 디스크로 쓰는(write) 데이터 용량을 의미한다.
- 데이터 비버퍼 쓰기량 : 데이터 버퍼가 아닌 다른 부분에서 디스크로 쓰는 데이터의 용량을 의미한다.
- 디스크 쓰기량(체크포인트) : 체크 포인트가 발생했을 때 DBWR(database writer)프로세스에 의해 디스크로 쓰이는 데이터의 용량을 의미한다.
- 디스크 쓰기량(비체크포인트) : 체크 포인트가 아닌 다른 이유로 디스크로 쓰이는 데이터의 용량을 의미한다.
- 로그 데이터량 : 생성되는 리두(redo) 로그 데이터 용량을 의미한다.

분석 실험은 TPC-C와 TPC-W 워크로드를 적용하고 4개의 파라미터에 15개씩 값을 변경하여 총 120회(한 파라미터에서 변경 회수 × 파라미터 수 × 적용되는 워크로드 개수)를 수행하였다. 또한 한회의 실험 당 14개의 성능 지표가 측정되므로 분석되는 총 성능 지표의 개수는 1680개가 산출된다.

TPC-W의 실험 환경은 데이터베이스, 웹/응용프로그램, 이미지 서버를 별개의 시스템에 장착하였다. 서버에 사용되는 데이터베이스는 오라클 9i 버전을 이용하였으며 웹 서버는 웹 로직(WebLogic)과 아파치 웹 서버를 이용하였다. TPC-C의 실험 환경은 클라이언트와 데이터베이스 시스템(오라클 9i)이 동일 기계에서 구동된다. 또한 실험 기간 동안 시스템에

최대 부하를 주기 위해, TPC-C는 웨어하우스의 수를 15개로 설정하였고, TPC-W는 EB의 수를 90개로 설정하였다.

#### 4. 워크로드 및 자원 사용 형태 분석

##### 4.1 데이터 변경률

(그림 1)과 (그림 2)는 TPC-W와 TPC-C의 파라미터 실험에서 기록된 데이터 변경률의 변화를 보인다. TPC-C와 TPC-W에서 데이터 변경률은 약 20% 정도의 차이를 보이기 때문에 두 워크로드를 구분하는 요소가 된다. 데이터 변경률은 TPC-W 워크로드 보다 TPC-C 워크로드에서 높은 수치를 보이며, 파라미터 값의 변경에 영향을 받지 않고 일정 범위 내에서 변화하였다.

##### 4.2 버퍼 적중률

(그림 3)과 (그림 4)는 TPC-W와 TPC-C의 파라미터의

변경 실험에서 기록된 버퍼 적중률의 변화를 보인다. 버퍼 적중률은 공유 메모리, 개인 메모리, I/O 프로세스 파라미터 값의 확장에 영향을 받지 않았으며, TPC-W 환경에서 더 낮은 범위의 적중률을 보였다. 데이터 버퍼 파라미터 값의 확장은 버퍼 적중률을 향상시킬 수 있었으며, TPC-W와 TPC-C 환경에서 다른 형태의 버퍼 적중률을 보이고 있다.

##### 4.3 데이터 버퍼 읽기량

(그림 5)와 (그림 6)은 TPC-C와 TPC-W의 파라미터 변경 실험에서 기록된 데이터 버퍼 읽기량의 변화를 보인다. 데이터 버퍼 읽기량은 공유 메모리, I/O 프로세스, 개인 메모리에 관련된 파라미터 값의 확장에 영향을 받지 않았으며, TPC-W가 TPC-C에 비해 약 40GB 많은 데이터를 읽는다. 데이터 버퍼 파라미터 값의 확장은 버퍼 읽기량을 감소시킬 수 있으며 TPC-W와 TPC-C가 다른 형태의 버퍼 읽기량을 보이고 있다.

(그림 1) TPC-W에서 데이터 변경률의 변화

(그림 2) TPC-C에서 데이터 변경률의 변화

(그림 3) TPC-W에서 버퍼 적중률의 변화

(그림 4) TPC-C에서 버퍼 적중률의 변화

(그림 5) TPC-W에서 데이터 버퍼 읽기량의 변화

(그림 6) TPC-C에서 데이터 버퍼 읽기량의 변화

버퍼 적중률과 데이터 버퍼 읽기량은 데이터 버퍼 자원의 문제를 판단하는 성능지표 중 하나이다. 버퍼 적중률과 데이터 버퍼 읽기량은 워크로드에 따른 데이터 버퍼 크기의 설정이 적절한지 판단할 수 있다. 높은 버퍼 적중률이나 적은 데이터 버퍼 읽기량은 워크로드 환경에서 데이터를 처리하기 위한 데이터 버퍼 크기가 충분함을 의미하며, 낮은 버퍼 적중률과 많은 버퍼 읽기량은 워크로드 환경에서 데이터를 처리하기 위한 데이터 버퍼의 크기가 작음을 의미한다. 낮은 버퍼 적중률과 많은 데이터 버퍼 읽기량은 데이터 버퍼 크기의 확장으로 적중률의 향상과 읽기량의 감소를 시도할 수 있지만, 데이터 변경률에 따라 적중률과 읽기량의 변화 폭이 다르게 나타난다. 데이터 변경률이 높은 경우에 버퍼 적중률의 향상 폭과 버퍼 읽기량의 감소 폭은 적게 나타나고, 데이터 변경률이 낮은 경우에 버퍼 적중률의 향상 폭과 버퍼 읽기량의 감소 폭은 크게 나타난다.

이러한 현상들을 본 실험에서 살펴보면, TPC-W에서 데이터 변경률이 10% 미만(그림 1 참조)을 보여 6단계 이전 변경까지 버퍼 적중률의 향상 폭과 데이터 버퍼 읽기량의 감소 폭이 크며, 7단계 이후 변경부터 버퍼 적중률의 향상 및 데이터 버퍼 읽기량의 감소가 보이지 않는다. TPC-W에서 데이터 버퍼의 크기는 6단계 미만의 변경까지 데이터를 처리하기에 적고 6단계 이상의 변경이면 충분히 데이터를 처리할 수 있으며 7단계 이상의 변경으로 확장할 필요가 없음을 의미한다. 반면에, TPC-C에서 데이터 변경률이

약 30%(그림 2 참조)를 보여 버퍼 적중률 향상 폭과 데이터 버퍼 읽기량 감소 폭이 크지 않았으며 모든 변경 단계에서 버퍼 적중률이 향상되고 데이터 버퍼 읽기량은 감소되었다. TPC-C에서 데이터 버퍼 크기는 15단계의 변경으로도 일정한 버퍼 적중률과 데이터 버퍼 읽기량을 보이지 않았다.

#### 4.4 데이터 버퍼 쓰기량

(그림 7)과 (그림 8)은 파라미터 변경 실험에서 기록된 데이터 버퍼 쓰기량을 워크로드별로 보인다. 버퍼 쓰기량은 두 워크로드에서 다른 범위를 보이기 때문에 워크로드를 구분하는 요소가 되며 TPC-C 워크로드의 공유 메모리와 데이터 버퍼 파라미터 변경에 의해서 영향을 받는다.

#### 4.5 디스크 쓰기량(비체크포인트)

(그림 9)와 (그림 10)은 파라미터 변경 실험에서 체크 포인트가 아닌 다른 경우에 디스크에 쓰이는 용량을 워크로드별로 보인다. 디스크 쓰기량(비체크포인트)은 두 워크로드에서 차이를 보이므로 워크로드를 구분할 수 있는 요소가 되며, TPC-C 워크로드의 공유 메모리와 데이터 버퍼 파라미터 변경에 의해서만 영향을 받는다.

데이터 버퍼 쓰기량과 디스크 쓰기량(비체크포인트)은 데이터 버퍼에서 디스크로 쓰이는 데이터 용량에 관한 성능 지표이며 데이터 변경률과 밀접한 관계를 가진다. 데이터

(그림 7) TPC-W에서 데이터 버퍼 쓰기량의 변화

(그림 8) TPC-C에서 데이터 버퍼 쓰기량의 변화

(그림 9) TPC-C에서 디스크 쓰기량(비체크포인트)의 변화

(그림 10) TPC-W에서 디스크 쓰기량(비체크포인트)의 변화

변경률이 높으면 더티(dirty) 데이터가 많이 발생되어 데이터 버퍼 쓰기량과 디스크 쓰기량(비체크포인트)이 많아질 수 있다. 반대로 데이터 변경률이 적으면 더티 데이터가 적게 발생되어 데이터 버퍼 쓰기량과 디스크 쓰기량(비체크포인트)이 적어질 수 있다.

데이터 버퍼 크기 확장은 버퍼내의 더티 데이터 보존 수를 증가시켜 쓰기량을 감소시킨다. 그러나 데이터 변경률이 낮은 경우에 버퍼 크기 확장은 더티 데이터의 수가 많지 않아 쓰기량이 감소되지 않고 일정하다. 본 실험에서 살펴보면, TPC-W 환경에서 버퍼 쓰기량과 디스크 쓰기량(비체크포인트)은 10% 미만의 변경률(그림 1 참조)로 인해 메가바이트 단위의 용량을 보이며, 버퍼 크기의 확장으로도 일정한 쓰기량을 보인다. TPC-C 환경에서 버퍼 쓰기량과 디스크 쓰기량(비체크포인트)은 30% 정도의 변경률(그림 2 참조)로 인해 기가 바이트 단위의 용량을 보이며, 버퍼 크기의 확장으로 쓰기량이 감소된다.

#### 4.6 공유 메모리 적중률

파라미터 값의 변경 실험에 대한 공유 메모리 적중률의 변화는 (그림 11)과 (그림 12)에서 보인다. 워크로드 별로 적중률 범위가 구분되기 때문에 공유 메모리 적중률은 워크로드를 구분하는 요소가 된다. TPC-W에서 적중률은 거의 100%에 가까운 적중률을 보임으로서 대부분의 질의 정보가 공유 메모리에 존재함을 의미하며, TPC-C에서 적중률은 약 80% 정도를 보였다. 한편, 파라미터 변경은 공유

(그림 11) TPC-W에서 공유 메모리 적중률의 변화

(그림 13) TPC-W에서 메모리 파싱 비율의 변화

메모리 적중률에 영향을 주지 않았다.

#### 4.7 메모리 파싱 비율

(그림 13)과 (그림 14)는 파라미터 변경 실험에서 메모리 파싱 비율의 변화를 보인다. 이 비율은 TPC-C와 TPC-W에서 다른 비율 범위를 나타내어 워크로드를 구분하는 요소가 된다. 메모리 파싱 비율은 파라미터 값의 변화에 영향을 받지 않았다.

공유 메모리 적중률과 메모리 파싱 비율은 공유 메모리 자원의 문제를 판단하는 성능지표이다. 공유 메모리 적중률과 메모리 파싱 비율이 높을수록 질의 및 파싱에서 처리되는 데이터가 공유 메모리에 대부분 존재하며, 공유 메모리 적중률과 메모리 파싱 비율이 낮을수록 질의 및 파싱에서 처리되는 대부분의 데이터가 공유 메모리에 존재하지 않는다.

TPC-W 환경에서 공유 메모리 적중률과 메모리 파싱 비율은 약 100%를 보인다. 이러한 현상은 질의 관련 데이터의 대부분이 공유 메모리에 존재하고, 질의 파싱에 관련된 데이터도 공유 메모리에 존재함을 의미한다. TPC-C 환경에서 공유 메모리 적중률은 약 80%를 보이고 메모리 파싱 비율은 약 20% 정도를 보인다. 이 현상은 질의 관련 데이터가 약 80% 정도 공유 메모리에 존재하나 질의 파싱에 필요한 데이터는 대부분이 디스크에서 읽어 수행됨을 의미한다. 데이터베이스 튜닝 관점에서 볼 때, 공유 메모리 적중률과 메모리 파싱 비율에 대한 향상은 자원의 크기 조절 보다 워크로드 성향을 고려하는 것이 바람직하다.

(그림 12) TPC-C에서 공유 메모리 적중률의 변화

(그림 14) TPC-C에서 메모리 파싱 비율의 변화

(그림 15) TPC-W에서 로그 데이터량의 변화

(그림 16) TPC-C에서 로그 데이터량의 변화

(그림 17) TPC-W에서 래치 경합 비율의 변화

(그림 18) TPC-C에서 래치 경합 비율의 변화

#### 4.8 로그 데이터량

(그림 15)와 (그림 16)은 파라미터 변경 시험에서 생성되는 로그 데이터량의 변화를 보인다. 로그 데이터량은 TPC-C와 TPC-W에서 차이를 보이기 때문에 워크로드를 구분하는 요소가 된다. 두 워크로드 중 TPC-C가 쓰기량이 많기 때문에 TPC-W에 비해 상대적으로 큰 로그 데이터량이 발생된다. 한편, 공유 메모리 파라미터 변경만이 로그 데이터의 량에 영향을 주었다.

#### 4.9 기타 지표들

메모리 정렬 비율은 두 워크로드에서 모두 100%를 보인다. 다른 5개의 지표들은 두 워크로드에서 래치의 경합정도 ((그림 17)과 (그림 18) 참조)처럼 성능지표 값의 범위가 유사하게 나타났다. 즉, 기타 지표는 TPC-C와 TPC-W 워크로드의 성향을 구분하는 요소가 되지 못하였다.

### 5. 결론 및 향후과제

본 논문은 TPC-C와 TPC-W 환경에서 14개의 성능지표를 이용하여 데이터베이스 시스템에 대한 자원 사용 형태를 분석하고, 데이터 버퍼, 공유 메모리, 개인 메모리, I/O 프로세스 자원 크기 변화에 따라 자원 사용 형태의 변화를 분석하였다. 적용된 총 14개의 성능 지표 중 8개의 성능 지표는 두 워크로드에 대한 특징을 구분하였고, 6개의 성능지표는 두 워크로드의 특징을 명확히 구분하지 못하였다.

TPC-W 환경에서 분석 결과는 데이터 변경률이 10% 미만으로 데이터 버퍼 쓰기량과 디스크 쓰기량이 메가바이트 정도의 용량을 보였다. 버퍼 적중률과 데이터 버퍼 읽기량은 데이터 버퍼 파라미터의 크기 변화로 높은 버퍼 적중률과 낮은 버퍼 읽기량을 유지할 수 있었다.

TPC-C 환경에서 분석 결과는 데이터 변경률이 약 30%로 데이터 버퍼 쓰기량과 디스크 쓰기량이 기가 바이트 이상의 용량을 보였다. 버퍼 적중률과 데이터 버퍼 읽기량은 총 15단계의 데이터 버퍼 파라미터 크기 변화로도 일정한 적중률과 읽기량을 보이지 않았다. 이는 수행된 실험보다 더 큰 버퍼 크기로 설정해야 일정한 적중률과 읽기량이 유지됨을 의미한다. 공유 메모리 적중률과 메모리 파싱 비율은 워크로드 성향을 고려한 튜닝 기술이 요구되었다.

본 논문의 결과인 자원 사용 형태와 자원 사용 형태의 변화에 대한 분석은 데이터베이스 시스템 튜닝을 위해 효과적으로 사용될 수 있다. 자원 사용 형태는 성능지표를 이용하여 복잡한 워크로드에서 성능 특징을 식별할 수 있으며, 이는 워크로드 성능 특징을 고려하는 튜닝 기법으로 데이터베이스 시스템의 성능을 향상시킬 수 있다. 자원 사용 형태의 변화는 성능 지표를 이용하여 워크로드에 영향을 주는 자원을 판별할 수 있으며, 워크로드에 따른 적절한 자원 크기를 설정하여 시스템의 성능 향상을 유도할 수 있다. 본 논문의 향후 연구는 성능 지표들을 이용하여 워크로드 환경에 적절한 데이터베이스 시스템의 자원 크기를 추천하는 방법론 수립하고 상이한 워크로드 환경에서 자동으로 적절한 자원의

크기를 조절하는 도구를 개발한다.

### 참 고 문 헌

[1] D. G. Benoit, "Automated Diagnosis and Control of DBMS Resources," Conference on Extending Database Technology (EDBT) Ph.D. Workshop, 2000.

[2] K. P. Brown, M. Mehta, M. J. Carey and M. Livny, "Towards Automated Performance Tuning For Complex Workloads," International Conference on Very Large Data Bases(VLDB), pp.72-84, 1994.

[3] M. Calzarossa and G. Serazzi, "Workload Characterization : A Survey," Proceedings of the IEEE, 8(13), pp.1136-1150, 1993.

[4] M. Cyran, 'Oracle 9i : Database Performance Guide and Reference, Release 1 (9.0.1),' Oracle Corporation, 2001.

[5] 'INFORMIX : Universal Server Administrator's Guide I and II (version 9.1),' Informix press, 1997.

[6] P. Martin, H. Y Li, M. Zheng, K. Romanufa and W. Powley, "Dynamic Reconfiguration Algorithm : Dynamically Tuning Multiple Buffer Pools," International Conference on Database and Expert Systems Applications (DEXA) Conference, pp.92-101, 2000.

[7] P. Martin, W. Powley, H. Y. Li and K. Romanufa, "Managing Database Server Performance to Meet QoS Requirements in Electronic Commerce Systems," International Journal on Digital Libraries, 3(4), pp.316-324, 2002.

[8] T. Morals, and D. Lorentz, 'Oracle 9i : Database Reference, Release 1 (9.0.1)', Oracle Corporation, 2001.

[9] V. Singhal and A. J. Smith, "Characterization of Contention in Real Relational Databases," Technical Reports : CSD-94-801, University of California at Berkely, 1994.

[10] D. E. Shasha, 'Database Tuning : a Principled Approach', Prentice Hall PTR, 1992.

[11] "TPC Benchmark C Specification (Revision 5.0)," <http://www.tpc.org/tpcc/default.asp>, 2001.

[12] "TPC Benchmark W (Web Commerce) Specification (version 1.8)," <http://www.tpc.org/tpcw/default.asp>, 2002.

[13] G. Weikum, A. C. Konig, A. Kraiss, and M. Sinnewell, "Towards Self-Tuning Memory Management for Data Servers," Bulletin of the Technical Committee on Data Engineering, IEEE Computer Society, 22(2), pp.3-11, 1999.

[14] P. S. Yu, M. S. Chen, H. U. Heiss and S. H. Lee, "On Workload Characterization of Relational Database Environments," IEEE Transactions on Software Engineering, 18(4), pp.347-355, 1992.

[15] 'UniSQL : Database Administration Guide (version 4.0),' UniSQL Inc., 1998.

### 오 정 석

e-mail : dbstar@nate.com

1996년 서경대학교 정보처리학과(학사)

1998년 숭실대학교 대학원 컴퓨터학과  
(석사)

1998년~현재 숭실대학교 컴퓨터학과  
대학원 박사 수료

관심분야 : 메타 검색, 데이터베이스 시스템 튜닝 및 성능평가,  
데이터베이스 워크로드

### 이 상 호

e-mail : shlee@computing.ssu.ac.kr

1984년 서울대학교 컴퓨터공학과(학사)

1986년 미국 노스웨스턴 대학교 전산학과  
(석사)

1989년 미국 노스웨스턴 대학교 전산학과  
(박사)

1990~1992년 한국전자통신연구원 선임연구원

1999~2000년 미국 George mason 대학교 교환 교수

1992~현재 숭실대학교 컴퓨터학부 교수

관심분야 : 인터넷 데이터베이스, 데이터베이스 튜닝 및 성능평가,  
웹 기술