

테이블 연관관계 도출을 통한 데이터베이스 워크로드 분석

김민수⁰

고려대학교 컴퓨터정보통신대학원
msbilly⁰@korea.ac.kr

Database Workload Analysis Based on Table Relationships

MINSU KIM⁰

Dept. of Computer Information Communication, Graduate School, Korea University

요 약

데이터베이스 시스템을 효율적으로 운영하기 위하여 데이터베이스 관리자는 시스템의 자원 사용과 응용 프로그램에 의한 워크로드의 특징을 알아야 한다. 워크로드 분석을 위해 테이블, 리소스, 튜닝 방법론 등 여러 연구가 진행되어 왔으나 워크로드를 형성하는 역할이 특정 테이블에만 집중되어 있는 현상에 대해서는 연구된 적이 없었다. 본 논문에서는 운영 시스템의 테이블 간의 연관 관계를 도출해 보고 연관 관계를 가지는 테이블들이 워크로드에 참여하는 유형과 횟수를 분석하는 워크로드 분석 도구를 제안하고 대형 CRM 분석 시스템에 적용하여 데이터베이스 시스템의 워크로드를 분석해 본다.

1. 서 론

데이터베이스 시스템을 효율적으로 운영하기 위하여 데이터베이스 시스템 관리자는 데이터베이스 시스템의 자원 사용과 응용 프로그램의 요구 사항, 워크로드 특징, 데이터베이스 시스템 정보를 필요로 한다[4]. 다양화 되고 복잡해 지는 응용 분야에 따라 데이터베이스 관리자도 워크로드의 유형에 맞는 시스템 관리 방법 및 튜닝 방법을 이용하여야 한다. 일반적으로 워크로드를 시스템에 부하를 가할 수 있는 요소 집합을 의미하며, 데이터베이스에서의 워크로드의 질의 집합을 의미한다. 질의 집합은 질의에 참여하는 테이블, 컬럼, 조인 관계, 제약 조건, 사용되는 테이블의 물리적 위치 등으로 구성되며 이중 테이블은 질의 구성 중 가장 중요한 위치에 있다. 이런 질의 연산 구성 및 스키마 정보, 데이터베이스 시스템 자원의 변경은 데이터베이스 시스템에 상이한 부하를 줄 수 있다. 예를 들어 특정 테이블에만 읽기/쓰기 트랜잭션이 많이 발생하는 워크로드와 다수의 테이블에 읽기/쓰기 트랜잭션이 골고루 나타나는 워크로드는 데이터베이스 튜닝 진행 시 상이한 방법의 진행이 필요하게 된다.

본 논문에서는 운영 중인 시스템으로부터 워크로드 형성의 중심이 되는 테이블 군을 찾고 워크로드의 유형을 확인할 수 있는 분석 도구를 구성 제시 한다. 본 논문의 구성은 다음과 같다. 2장에서는 워크로드에 대해서 진행되어 왔던 연구에 대해 설명하고, 3장에서는 워크로드 분석을 위해 테이블 단위에서 워크로드를 형성하는 중심이 되는 테이블 군을 찾을 수 있는 분석 도구를 설계한다. 4장에서는 실제 대형 CRM 분석 환경의

데이터베이스 환경에 설계된 분석 도구를 적용하여 데이터베이스 시스템의 워크로드를 분석하고 분석 결과를 설명한다. 5장에서는 논문의 결론 및 앞으로의 연구 방향을 제시 한다.

2. 관련 연구

데이터베이스 워크로드를 분석하기 위한 연구는 오래 전 부터 많은 분야에 걸쳐서 진행되어 왔다.

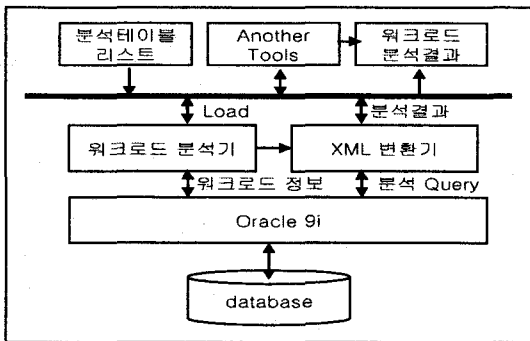
[1]은 DB2 환경 관계형 데이터베이스에서 SQL 구문의 복잡성과 구조성 분석으로 트랜잭션과 쿼리의 run-time 시의 워크로드 특징을 분석하는 도구인 REDWAR(Relational Database Workload Analyzer)를 개발하고, 워크로드의 대상이 되는 스키마 정보를 분석하였다. [2]는 전자 상거래 시스템에서 세 개의 응용 분야에 대한 워크로드 특징을 분석하고 QoS 요구 사항들을 정립하였고, Quarermaster 시스템을 제시 하였다. [3]은 데이터베이스 시스템에서 성능을 하락시키는 자원을 진단하는 연구를 수행하였다 [4]는 데이터베이스 튜닝에 도움을 주기 위해 질의 종류 분포, 질의 조건 분석, 사용된 시스템 자원 통계, 사용된 테이블 통계의 필요성과 활용 방안을 기술하고 인덱스 분석 도구를 구현하였다. 질의 분포는SELECT, INSERT, UPDATE, DELETE 구문의 개수 분포를 나타내고 사용 자원 통계는 질의 수행 중 사용된 데이터베이스 자원(예를 들어 CPU, 데이터 페이지, 락/래치 요청 등)이 사용된 횟수와 양을 나타낸다. 질의 조건 분석은 WHERE 구문에 나타난 조건 (LIKE, BETWEEN, IN 등)의 종류 및 개수를 나타낸다. 사용된 테이블 통계는 SELECT, FROM, ORDER BY, GROUP

BY 의 절에서 언급 되는 테이블과 컬럼의 목록과 개수를 나타낸다. [5]는 수행되는 자원 영역에 따라 워크 로드를 디스크 버퍼 관련 트랜잭션/질의 집합과 메모리(working storage) 관련 트랜잭션/질의 집합으로 세분화 하고 트랜잭션/질의 별로 목표 응답 시간을 설정하여 자원 할당 크기와 다중 메모리 단계(multiprogramming level)를 피드백 알고리즘에 의해 동적으로 조절하는 방법을 제안하였다.

[6] 과 [7] 은 웹 커뮤니티 간의 접속을 형태 및 횟수를 이용하여 웹 커뮤니티 차트를 생성하고 생성된 커뮤니티 차트를 분석 하는 방법을 제안 하였다.

3. 워크 로드 분석 도구 설계

본 연구는 국내 대형 통신사 CRM 분석 시스템을 대상으로 하여 진행 되었다. 이 대형 마트 시스템은 오라클 DBMS 로 구현이 되어 있고 1600 여 개의 테이블로 구성 되어 있다. 대형 마트 시스템 의 특성상 작업 분산과 부하 분산을 위한 모델링과 적절한 대용량 처리가 필수적으로 필요하며 특정 리소스 테이블에 작업 부하가 집중되는 것을 방지하기 위해 많은 물리적 설계를 고려한 상태이다. 본 연구에서는 DBMS 워크 로드 분석을 위하여 DBMS 시스템에 저장되어 있는 테이블 이용 이력 SQL 구문을 변환 하여 관리 테이블 목록과 비교하고, 비교된 결과를 토대로 테이블 간의 연결 집중도를 파악해 보고, 각 테이블의 이용률 및 어떤 종류의 DBMS 명령이 주로 사용되는지를 분석하여 DBMS 시스템의 워크 로드를 형성하는 원인 테이블을 도출해 보았다. SQL 수행 시 발생하는 JOIN 을 테이블 간의 관계로 정의 하고, 테이블 간의 관계 집약도를 연결 집중도로 정의 한다. 본 논문에서 구성한 워크 로드 분석 도구는 [그림1]과 같이 구성된다.



[그림 1]. 워크 로드 분석 도구 구성

워크 로드 분석기는 SQL_TEXT 의 쿼리 문장과 분석 테이블 리스트의 테이블 명을 비교 하여 분석 리스트의 테이블이 쿼리에 사용되었으면 해당 쿼리의 COMMAND_TYPE 값과 테이블 명, 사용 횟수를 결과 테이블에 입력 하게 된다. ADDRESS 와 HASH_VALUE 값이 동일한 쿼리에서 여러 개의 매칭된 테이블이 발생하는 것은 매칭된 테이블 들이 JOIN 관계로 묶여 있거나 Inline

View 로 싸여 있는 경우 이다. 이런 테이블간의 관계로 정의된 집합 군들 중 연결 집중도가 높은 테이블 이 쿼리 시에 출현하는 빈도까지 높으면 워크 로드 형성에 큰 영향을 주게 된다. 워크 로드 분석기에 의해 분석된 결과는 테이블 관계 다이어그램을 표현하기 위해 XML 형식으로 변형 되어 Tool 에 제공 하게 된다. 테이블 관계를 표현하는 Tool 은 별도 구현하지 않고 ER관계를 표현하는 Tool 을 이용 하였으며 테이블 간의 릴레이션 을 표현하는 방법인 relation-ship 표현을 테이블간의 연관 관계로 표현하고, 릴레이션의 개수를 연관 도의 크기로 표현하였다. DBMS 에서 획득하는 워크 로드를 분석하기 위해 ORACLE 의 V\$SQLTEXT 테이블을 이용한다. 본 연구에 이용된 V\$SQLTEXT 정보는 시스템 가동 후 충분한 시간이 경과한 특정 시점의 정보를 대상으로 하였기 때문에 데이터베이스 시스템에서 이용하는 SQL 구문은 충분한 반영 되었을 것이라는 전제 하에서 추출 되었다.

V\$SQLTEXT 테이블 은 DBMS 에서 실행된 쿼리 정보를 저장해 두는 테이블 로써 수행된 SQL 의 유형 과 SQL 전체 텍스트를 포함하고 있다. 테이블 간의 관계 분석을 위하여 ADDRESS 컬럼과 HASH_VALUE 값을 함께 이용 한다. V\$SQLTEXT 의 컬럼 구성은 [표 1]과 같다.

[표 1]. Oracle 시스템의 V\$SQLTEXT

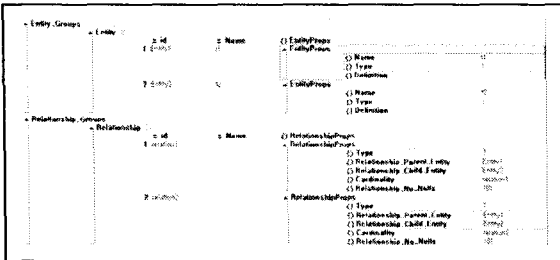
컬럼명	설명
ADDRESS	HASH_VALUE 와 함께 커서 사용 주체를 알려주는 Unique 값
HASH_VALUE	ADDRESS 와 함께 커서 사용 주체를 알려주는 Unique 값
COMMAND_TYPE	SQL Statement 값
PIECE	SQLText 구성 조각에 대한 번호
SQL_TEXT	이용된 SQL TEXT

COMMAND_TYPE 의 구성을 DDL 명령 군과 DML 명령 군으로 분류 하고 워크 로드를 구성하는 명령 군에 어떤 COMMAND 가 많이 사용되는 지를 분석한다. [표 2] 는 COMMAND_TYPE 의 구분 및 실행 결과에서 기술될 표기를 비교 정리한 표 이다. 연구 대상 시스템의 COMMAND_TYPE 중 테이블 기반 워크 로드로 분류하기 힘든 타입인 17:grant 와 50:Explain 은 워크 로드 분석에서 제외 하기로 한다

[표 2] command type 표기법

구분	Command_type	표기
DDL	75,15,11	Alter
	85	Truncate
	1	CTAS
	9	CIAS
DML	3	Select
	2	Insert
	7	Delete
	189	Merge
	6	Update

테이블 간의 relation 을 표현하기 위해 session 정보로 연결된 여러 개의 테이블을 XML 형식으로 변형하기 위해 XML 변환기를 사용하였으며 결과 XML 의 구성은 [그림 2] 와 같다.

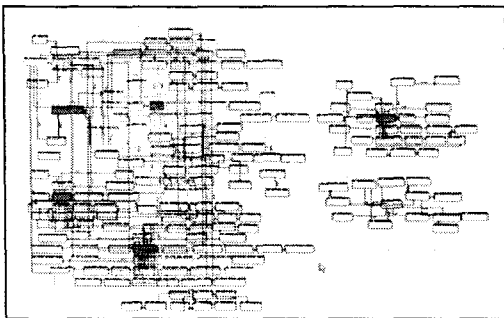


[그림 2] XML 변환 구조

XML entity 로는 각각의 테이블 명을 표현하는 " entity " 와 테이블 간의 관계를 표현하는 " relation " 이 사용된다. 각 entity 의 attribute 로는 ID, name 이 사용 되었으며 relation 의 sub entity 는 두 테이블 간의 관계를 표현하기 위하여 parent_entity 와 child_entity 를 포함하도록 하였다.

4. 연구 결과

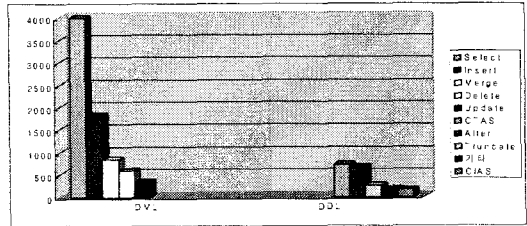
대용량 CRM 분석 시스템의 데이터베이스 COMMAND_TYPE 워크 로드 분석 결과 전체 1600 여 개 테이블 중 2% 미만인 50 여 개의 테이블이 전체 테이블 조인의 허브 역할을 하는 연결 집중의 중심 테이블 인 것이 확인 되었다. [그림3]은 쿼리 시 연결된 테이블 관계를 다이어 그램화 한 것이며 특정 테이블에 연결된 테이블의 개수가 많은 것을 볼 수 있다. 공통 코드 테이블의 경우는 100여 회 이상의 Join 횟수를 보였으며, Online 성 테이블 에서도 80여 회 의 연결 집중 도를 보인 테이블이 있었다. 또한 허브 역할을 하는 테이블들이 여러 개의 군집으로 형성되어 있는 것을 확인할 수도 있었다. 이는 업무 영역별로 허브 역할을 하는 테이블들이 존재 하는 것을 나타낸다.



[그림 3] 테이블 연결 집중도 분석

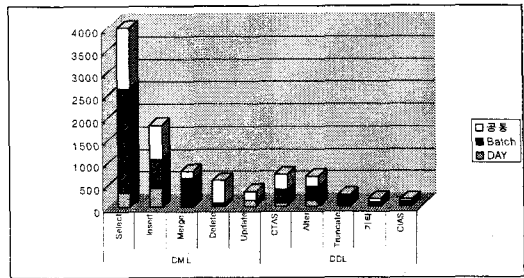
[그림4]는 워크 로드가 Select 에 의해서 주로 발생되었으며 Delete, Update, Merge (update + insert) 의 발생

횟수도 적지 않게 발생 하였음을 보여준다. 시스템의 운영 특성상 낮 시간 동안에는 사내 사용자들 에게 분석 통계 정보를 제공하고, 야간 시간에는 통계 자료 생성을 위해서 대용량 Batch 작업이 진행되고 있다.



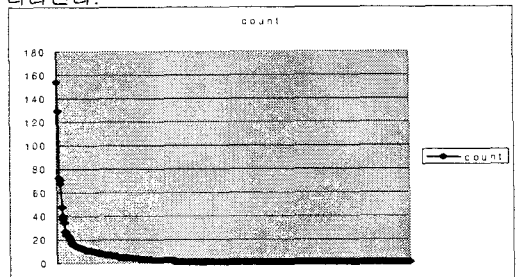
[그림 4] DML/DDL 워크 로드 분석

[그림5]에서는 실험 결과에 나타난 4000 여 회의 select 가 주간의 온라인 타임에 발생 된 것인지, 야간의 Batch 작업에 의한 것인지를 확인할 수 있다. select 구문이 배치 작업에 의해서 발생되었는지, 온라인 작업에 의해서 발생되었는지를 분리해 보았다.

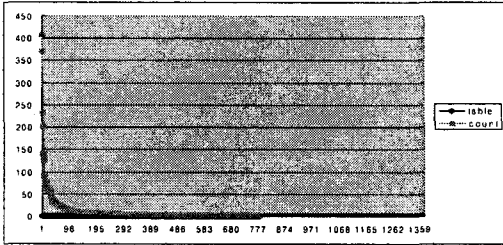


[그림 5] online/Batch 워크 로드 분석

DDL 은 CTAS 와 Alter 구문이 500 여 회 이상 진행된 것으로 나타났다. 하지만, [그림6,7] 에 나타 난 것처럼 DML 이나 DDL 이 발생하는 테이블의 분포는 특정 테이블에 지나치게 치중되어 있는 결과를 보였다. 이는 데이터베이스 워크 로드가 특정 테이블에 집중되어 나타나는 결과인 것이고 데이터베이스 관리자는 워크 로드가 집중되어 있는 테이블에 대한 튜닝 전략이나 관리에 더 많은 노력을 기울여야 한다는 것을 나타낸다.



[그림 6] DDL 발생 횟수 별 테이블 분포



[그림 7] DML 발생 횟수 별 테이블 분포

5. 결론 및 향후 연구 과제

본 논문에서는 조인에 참여하는 테이블간의 연관관계를 분석하여 데이터베이스 시스템 워크 로드의 원인을 밝혀 낼 수 있는 방안을 제시 하였다. 데이터베이스 시스템의 워크 로드 분석의 팩터로 사용할 수 있는 테이블의 쿼리에 참여하는 횟수 및 유형, 발생 횟수 별 분포 도 분석해 보았다. 본 논문은 DBMS 에 자동으로 생성되는 SQL 사용 이력과 데이터베이스 관리자가 필수적으로 관리해야 하는 테이블 명세서를 이용하여 데이터베이스 워크 로드를 분석할 수 있는 분석 도구를 제안하였으며 분석 도구를 통해 생성된 여러 유형의 워크 로드 원인을 밝혀 낼 수 있었다. 대상 시스템에서는 특정 테이블 이 쿼리에 참여 하는 횟수가 아주 높았으며 JOIN 에도 공통적으로 참여 하는 횟수가 높아 데이터베이스 워크 로드의 허브 역할을 하는 것으로 분석 되었다. DML 은 Select 가 주로 발생하였으며 이는 정기적인 Batch 작업에 의한 Select 가 주를 이루었다. 정기 Batch 작업의 SELECT 는 MERGE, INSERT, UPDATE, DELET 작업을 위한 준비 단계로 해석 될 수 있을 것이다. DDL 명령은 CTAS (Create AS Select) 구문과 Alter 구문이 많이 사용 된 것을 알 수 있다. DML 과 DDL 의 발생 횟수 별 테이블 분포는 전형적인 역함수 형태를 따르고 있었는데 전체 중 20% 이하의 테이블 이 전체 DML 과 DDL 명령의 80% 이상에 참여하는 결과를 보였다.

현재까지 데이터베이스 관리자는 전체 테이블 중 주요 관리 대상 테이블을 사이즈가 큰 테이블 또는 갱신 주기가 높은 테이블을 우선순위에 두고 관리하였으나, 본 논문의 워크 로드 분석기를 이용하면 워크 로드 발생 원인을 일으키는 테이블을 중심으로 데이터베이스 관리 활동을 펼 수 있을 것이다. 또한 워크 로드 분석을 위하여 물리적인 분산 정책에 본 논문의 결과를 이용할 수 있을 것이다.

본 논문은 동일한 작업이 반복적으로 발생하는 안정인 운영이 보장된 시스템을 대상으로 하여 진행 되었다. 쿼리의 유형이나 내용이 빈번하게 변하는 시스템에서는 V\$SQLTEXT 의 정보만으로 워크 로드를 분석하면 전체 워크 로드 중 일 부분만을 분석할 수 있다. 또한 V\$SQLTEXT 정보는 현재의 데이터베이스 시스템의 워크 로드를 분석할 수 있는 정보를 제공하지 않는다. V\$SQLTEXT 는

Queue 형식으로 일정 분량의 과거 쿼리만을 포함하고 있기 때문에 current 시점 트랜잭션 에 의한 워크 로드 분석에는 사용할 수가 없다는 단점이 있다.

향후 계획으로는 테이블 단위의 워크 로드 에서 발전시켜 컬럼 의 사용 형태까지 분석할 수 있는 워크 로드 분석기를 구현하고, 리소스의 이용 현황을 워크 로드 분석에 포함하여 현재 시점의 워크 로드에 대해서도 분석 가능하도록 연구 한다.

참 고 문 헌

- [1] P.S.Yu, M.S.Chen, H.U. Heriss and S.Lee, " One Workload Characterization of Relational Database Environments," IEEE Trasaction on Software Engineering, Vol.18, No.4, pp.347-355,1992
- [2] P.Martin. H.Y.Li, M.Zheng, K.Romanufa, and W.Powley. " Dynamic Reconfiguration Algorithm : Dynamically Tuning Multiple Buffer Pools" Proceeding of 11th DEXA Conference. Pp.92-101, London, 2000
- [3] D.G.Benoit, " Automated Diagnosis and Control of DBMS Resources," Ph.D Workshop on EDBT conference, 2000
- [4] S.Chandhuri and R. Narasayya, " AutoAdmin " What-if" Index Analysis utility," ACM SIGMOD conference, pp.367-378, 1998
- [5] K.P. Brown. M.Mehta. M.J. Carey, and M. Livny " Towards Automated Performance Tuning for Complex Work loads" , Proceeding of 20th VLDB Conference. pp.72-84. Santiago. 1994
- [6] Masashi Toyoda and Masaru Kisuregawa. " Creating a Web Community Chart for Navigating Related Communities." In Proc, Hypertext 2001, pp.103-112. 2001
- [7] Masashi Toyoda and Masaru Kitsuregawa. " Extracting Evolution of Web Communities from a Series of Web Archives."