공간 데이터 웨어하우스 구축기에서 추출된 데이터의 효율적인 적재를 위한 테이블 단위의 데이터 관리 기법

김형선°* 유병섭* 박순영* 이재동** 배해영*
인하대학교 컴퓨터정보공학과
{sunnymsg°, subi, sunny}@dblab.inha.ac.kr and letsdoit@dankook.ac.kr and hybae@inha.ac.kr

Data Management Method of Table Unit for Efficient Load in a Spatial Data Warehouse Builder

Hyungsun Kim°* Byeongseob You* Soonyoung Park* Jaedong Lee** Haeyoung Bae*
*Dept. of Computer Science & Information Engineering, Inha University
**Division of Information and Computer Science, Dankook University

요 약

공간 데이터 웨어하우스 구축기는 운영 데이터베이스의 데이터를 추출하여, 공간 데이터 웨어하우스 서버에 적재하는 과정을 효율적으로 관리하는 시스템이다. 구축기는 적재로 인한 서버의 부하를 줄이기 위하여 적재할 데이터를 임시 저장하는데, 기존 기법은 적재할 데이터를 하나의 저장 공간에 관리한다. 따라서 서버가 특정 차원 테이블에 대한 실시간 질의처리를 위해 특정 차원 테이블의 즉시 적재를 요청할 경우, 구축기는 이를 위해 임시 저장한 모든 데이터를 검색하므로 처리비용이 증가한다. 또한, 하나의 저장 공간에 적재할 데이터를 유지하여 서버에 데이터 적재 시, 저장을 위해 혼합된 데이터를 분석하는 비용이 증가한다.

본 논문에서는 공간 데이터 웨어하우스 구축기에서 추출된 데이터의 효율적인 적재를 위한 테이블 단위의 데이터 관리 기법을 제안한다. 제안 기법은 운영 데이터베이스로부터 추출한 데이터를 서버에 적재할 차원 테이블 단위로 구축기에서 각각 다른 저장 공간에 관리한다. 따라서 테이블 단위의 데이터 관리로 실시간 질의처리를 위한 특정 차원 테이블의 즉시 적재 비용이 감소하며, 테이블 단위의 병렬전송이 가능하여 전송비용이 감소한다. 또한, 서버로 전송된 데이터는 테이블 단위의 벌크 삽입이 가능하여 적재시간이 감소한다.

1. 서론

공간 데이터 웨어하우스(Spatial Data Warehouse)는 여러 운영 데이터베이스에서 추출된 공간 및 비공간 데이터를 주제별로 통합하여 의사결정을 지원하는 시스템이다[1, 2]. 공간 데이터 웨어하우스는 크게 서버(Server)와 구축기(Builder)로 이루어진다. 서버는 하나의 사실 테이블과 다수의 차원 테이블로 구성된 스타스키마를 이용하여, 관리자와 사용자들을 위한 다양한 의사결정 도구와 OLAP 연산 등을 제공하며, 구축기는 운영 데이터베이스의 데이터에 대해 추출과 변환 및 적재를 제공한다. 구축기가 추출한 소스 데이터는 구축기 임시 저장소에 임시 저장하고 주기적으로 서버에 적재한다[3]. 그러나 기존 기법은 적재할 데이터를 두 곳의 임시 저장 공간에서 관리한다. 따라서 서버가 특정 차원 테이블에 대한 실시간 질의처리를 위해 특정 차원 테이블의 즉시 적재를 요청할 경우, 구축기는 이를 위해 임시 저장한 모든 데이터를 검색하므로 처리비용이 증가한다[4]. 또한, 하나의 저장 공간에 적재할 데이터를 유지하여 서버에 데이터 적재 시, 저장을 위해 혼합된 데이터를 분석하는 비용이 증가한다[5].

본 논문에서는[1] 공간 데이터 웨어하우스의 구축기에서 추출된 데이터의 빠른 적재를 위한 테이블 단위의 데이터 관리 기법을 제안한다. 이는 운영 데이터베이스로부터 추출한 데이터를 공간 데이터 웨어하우스 구축기의 임시 저장소에 테이블 단위로 저장한다. 따라서 테이블 단위의 데이터 관리로 실시간 질의처리를 위한 특정 차원 테이블의 즉시 적재 비용이 감소하며, 테이블 단위의 병렬전송이 가능하여 전송비용이 감소한다. 또한, 서버로 전송된 데이터는 테이블 단위의 벌크 삽입이 가능하여 적재시간이 감소한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대해 설명하고, 3장에서는 제안하는 테이블 단위의 데이터 관리 기법에 대해 기술한다. 4장에서 성능 평가를 한 후, 5장에서 결론 및 향후 연구에 대하여 기술한다.

2. 관련 연구

본 장에서는 기존 데이터 웨어하우스에서 사용한 적재 기법과 본 논문에서 사용하는 데이터 벌크 삽입 기법에 대하여 설명한다.

2.1 기존 데이터 웨어하우스의 데이터 적재 기법

기존 데이터 웨어하우스 구축기에서는 중간 데이터 저장소(Intermediary Data Stores)와 스테이징 영역(Staging Area)을 두어 데이터 웨어하우스 서버에 적재될 데이터를 관리한다[6]. 추출된 데이터는 중간 데이터 저장소에 임시 파일로 관리하고, 추출된 데이터를 데이터 웨어하우스로 전송하기 전에 스테이징 영역에서 추출된 데이터를 정렬한다. 또한, 누락된 데이터 혹은 서로 다른 형식으로 이루어진 데이터들을 일치되도록 정제한 후, 하나의 임시 파일로 합병하고 서버로 전송한다.

1) 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성 지원사업의 연구결과로 수행되었음.

서버로 전송된 파일은 적재를 요청한 각 작업에서 자신에게 필요한 데이터를 찾기 위해 분석한다. 따라서 적재 데이터를 하나의 파일로 관리하므로 특정 차원 테이블의 즉시 적재 요청 시, 이를 처리하기 위한 비용이 증가하며, 서버로의 전송시간이 증가한다. 또한, 서버에서는 각 작업에서 필요한 데이터를 찾기 위한 분석 비용이 증가하며, 적재시간이 증가하고 사용자 응답시간이 증가한다.

2.2 데이터 벌크 삽입 기법

대량의 데이터 저장 시, 각각의 데이터마다 분석하여 저장하는 것은 데이터 처리 시간을 증가시킨다. 이를 해결하기 위하여 기존에 파티션된 B-트리를 이용한 벌크 삽입과 같은 데이터 벌크 삽입 기법이 연구 되었다[7].

데이터 벌크 삽입 기법은 관련된 대량의 데이터를 하나의 공간에 일시적으로 관리한다. 한번에 대량의 데이터 저장이 가능하고 데이터의 분석시간과 처리시간에 있어서 효율적이다. 데이터 저장 시, 별도의 분석 없이 대량의 데이터 저장이 가능하다. 따라서 데이터의 빠른 적재가 가능하다.

3. 구축기에서 효율적인 적재를 위한 테이블 단위의 데이터 관리 기법

본 장에서는 공간 데이터 웨어하우스 구축기에서 테이블 단위의 데이터 관리 기법을 제안한다. 먼저 공간 데이터 웨어하우스의 구성을 설명하고, 구축기의 임시 저장소에서 테이블 단위의 데이터 관리를 살펴본다. 마지막으로 공간 데이터 웨어하우스 서버의 테이블 단위의 적재를 설명한다.

3.1 공간 데이터 웨어하우스 구성

공간 데이터 웨어하우스는 그림 1과 같이 운영 데이터베이스와 공간 데이터 웨어하우스 구축기와 공간 데이터 웨어하우스 서버로 구성된다.

운영 데이터베이스에는 이질적인 데이터베이스 관리 시스템이 존재하며, 서버의 의사결정 지원에 기반이 되는 수년간 축적된 데이터가 저장되어 있다.

공간 데이터 웨어하우스 구축기는 이질적인 운영 데이터베이스에서 데이터를 추출하고 이를 변환 및 적재하는 ETL(Extract, Transform, Load)과정을 수행한다. 이때, 적재를 위한 데이터는 임시 저장소에 관리하고 주기적으로 서버에 적재하여 서버의 부하를 감소시킨다. 구축기의 임시 저장소는 추출한 데이터를 서버로 전송하기 전까지 임시 저장한다.

공간 데이터 웨어하우스 서버는 OLAP 연산 등을 제공하여 데이터 마이닝 도구 등을 지원하는 시스템으로 데이터의 다양한 분석을 통해 사용자의 의사결정 지원을 지원한다. 공간 데이터 웨어하우스 서버는 미들웨어 구조로 구성되어 있으며, 데이터의 효율적인 관리를 위해 인하대학교 데이터베이스 연구실에서 개발한 공간 데이터베이스 시스템인 Geo Millennium Server(GMS)를 사용한다[8].

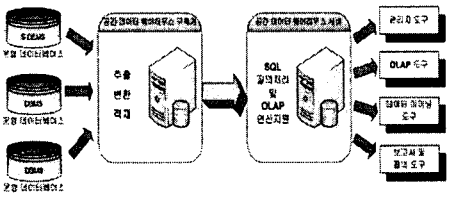

그림 1 공간 데이터 웨어하우스 구성

3.2 구축기의 임시 저장소에서 테이블 단위의 데이터 관리

구축기에서 테이블 단위의 데이터 관리를 위한 임시 저장소의 구조는 그림 2와 같다.

임시 저장소는 데이터 관리기와 테이블 단위의 저장소로 구성된다. 데이터 관리기는 임시 저장할 적재 데이터의 저장 위치를 계산하고 저장을 수행하는 컴포넌트이며, 테이블 단위의 저장소는 적재 데이터를 실제로 저장하는 공간이다. 운영 데이터베이스로부터 추출된 데이터는 ETL 관리자를 통하여 적재를 위한 데이터로 변경된다. 적재를 위한 데이터는 임시 저장소의 데이터 관리기로 임시 저장을 요청한다. 요청을 받은 데이터 관리기는 해당 데이터를 분석하여 저장할 차원 테이블을 찾고 데이터를 임시 저장한다. 이때, 임시 저장하는 데이터는 각 차원 테이블마다 벌크 삽입이 가능한 구조로 구성되며, 각 차원 테이블은 독립적으로 저장된다. 따라서 각 차원 테이블 마다 독립적으로 저장하므로 저장 관리 비용이 감소하며, 서버의 실시간 질의 처리를 위한 특정 차원 테이블의 즉시 적재 요청 시, 분석비용 없이 해당 차원 테이블을 빠르게 적재할 수 있다. 또한, 차원 테이블 단위의 관리는 적재 시, 데이터의 병렬 전송을 가능하게 하며, 차원 테이블 단위의 벌크 삽입으로 적재시간을 감소시킨다.

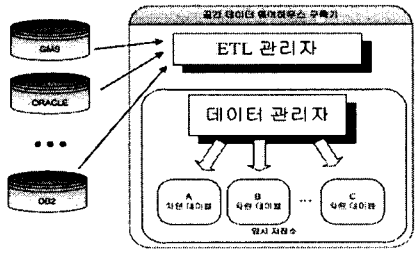

그림 2 테이블 단위의 데이터 관리를 위한 임시 저장소 구조

3.3 공간 데이터 웨어하우스 서버에서 데이블 단위의 적재

공간 데이터 웨어하우스 서버로의 적재는 두 경우가 존재한다. 첫 번째는 사용자의 실시간 분석요청으로 해당 차원 테이블들의 즉시 적재를 요청하는 경우이고, 두 번째는 메타데이터에 저장된 적재 주기에 따라 적재를 요청하는 경우이다. 첫 번째의 경우는 해당 차원 테이블만을 빠르게 적재하며, 두 번째의 경우는 구축기에 존재하는 모든 데이터를 적재한다.

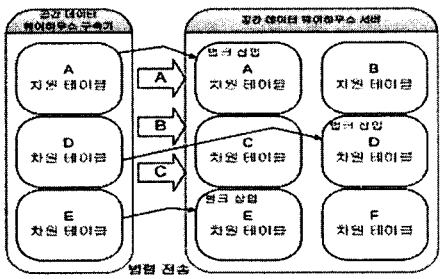

그림 3 서버에서 테이블 단위의 적재 과정

공간 데이터 웨어하우스 서버에서 테이블 단위의 적재과정은 위 그림 3과 같다. 먼저, 즉시 적재 또는 적재 주기에 의한 적재로 인해 서버는 구축기에게 차원 테이블의 적재를 요청한다.

요청을 받은 구축기는 해당 차원 테이블들을 서버로 테이블 단위의 병렬전송을 수행한다. 서버는 병렬로 전송된 차원 테이블들을 받고 각 차원 테이블 별로 데이터 삽입을 수행한다. 이 때, 저장할 데이터가 차원 테이블 단위로 구성되어 있으므로, 벌크 삽입 기법을 이용하여 데이터를 삽입한다.

따라서 즉시 요청으로 필요한 차원 테이블들만 적재가 가능하며, 이는 실시간 분석요청에 대한 응답시간을 감소시킨다. 또한, 구축기로부터의 데이터 병렬전송은 적재 데이터의 전송시간을 감소시키며, 전송 받은 데이터의 벌크 삽입은 데이터의 저장시간을 감소시킨다. 이는 전체 차원을 요구할 경우에도 테이블 단위의 병렬전송과 벌크 삽입이 가능하므로, 기존 기법보다 성능이 향상된다.

4. 성능 평가

본 장에서는 앞서 기술한 관련 연구의 기존 데이터 웨어하우스에서 이용한 데이터 벌크 삽입기법과 본 논문에서 제안한 테이블 단위의 데이터 관리기법을 이용한 성능 평가를 하였다.

성능 평가를 하기 위한 시스템 환경은 공간 데이터 웨어하우스 서버와 구축기, 운영 데이터베이스 관리 시스템으로 이루어졌으며, 각각은 동일한 사양의 시스템이다. 각 시스템의 하드웨어 사양은 표 1과 같다.

표 1 시스템 속성

시스템 하드웨어 사양	
CPU	Pentium4 3.0Ghz
Memory	1 Gigabyte
HDD	160 Gigabyte
OS	Windows 2000 Professional

성능 평가에 사용된 테스트셋은 다음과 같다.

서버는 20개의 차원 테이블이 존재하며, 적재 데이터는 운영 데이터베이스로부터 가져온다. 구축기에서 서버에 저장할 각 차원 테이블별 데이터 개수는 약 1,000개가 되도록 한다.

성능 평가는 테스트셋을 기반으로 1~20개의 차원 테이블을 서버에서 적재를 요청하는 것에 따라, 적재시간의 변화를 측정한다. 이때, 적재시간이란 적재를 요청한 시점으로부터 적재가 완료되는 시점까지의 시간을 말한다.

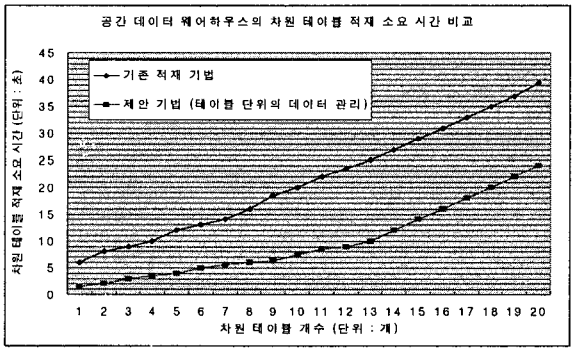

그림 4 공간 데이터 웨어하우스 차원 테이블 적재 소요 시간 비교

그림 4를 보면, 제안 기법이 기존 기법보다 성능이 우수함을 볼 수 있다. 기존 기법의 경우, 구축기에서 하나의 파일에 모든 차원 테이블 데이터를 관리하므로, 특정 차원 테이블 요청 시, 해당 테이블과 일치하는 데이터를 찾기 위해, 검색 비용이 증가한다. 또한, 직렬 저장을 사용하므로, 차원 테이블이 증가할수록 적재시간이 크게 증가한다. 제안 기법의 경우, 구축기에서 차원 테이블마다 독립적으로 데이터를 관리하므로, 특정 차원 테이블 요청 시, 해당 테이블에 일치하는 데이터를 쉽고 빠르게 찾을 수 있으므로, 데이터 검색 비용이 감소한다. 또한, 병렬전송과 병렬저장을 하므로, 차원 테이블이 증가하여도, 적재시간이 크게 증가하지 않는다. 따라서 제안 기법이 기존 기법보다 약 45%의 성능 향상이 이루어지는 것을 볼 수 있다.

5. 결론 및 향후 연구

본 논문에서는 다량의 데이터가 주기적으로 적재되는 공간 데이터 웨어하우스의 차원 테이블 생성을 위해 요청되는 데이터를 테이블 단위로 저장 관리하는 기법을 제안하였다.

본 논문에서 제안한 기법은 테이블 단위의 데이터 관리기법으로, 이는 이질적인 운영 데이터베이스에 수년 간 축적된 데이터를 공간 데이터 웨어하우스 구축기의 ETL 과정을 통해 추출하고, 이를 변환하여, 구축기의 임시 저장소 데이터 관리자가 서버로 적재할 추출 데이터를 테이블 단위로 임시 저장된 뒤, 사용자가 지정한 주기적인 시간 혹은 실시간 질의처리 요청에 따라 임시 저장소에 저장한 데이터를 벌크 적재한다. 구축기가 테이블 단위의 데이터를 공간 데이터 웨어하우스 서버로 벌크 적재 시, 테이블 단위의 병렬전송이 가능하며 이로 인해 구축기와 서버의 병목현상도 감소하다. 또한, 서버는 전송된 데이터가 별도의 분석 없이 각 차원 테이블로 벌크 적재를 할 수 있으므로, 서버에 전송된 데이터의 분석 비용이 감소하여, 사용자의 질의 처리 시간을 응답 시간을 감소시킨다.

향후 연구로는 공간 데이터 웨어하우스 구축기에 서버로 적재될 데이터의 데이터 복구에 대한 연구와 공간 데이터 웨어하우스 서버 저장소에 기존에 생성되었던 차원 테이블의 데이터 대한 보다 효율적인 벌크 적재 기법을 연구할 필요가 있다.

참고문헌

[1] Lafond, P., "Designing and Building the Distributed Geospatial Data Warehouse Architecture", Proceedings of The Twelfth Annual Symposium on Geographic Information Systems, Toronto, 1998.

[2] M. Stonebraker., et. al., "The SEQUOIA 2000 Project," Proc. of 3rd Symposium of Spatial database '93, pp. 397-412, 1993.

[3] S. Chaudhuri and U. Dayal, "An Overview of Data Warehousing and OLAP Technology", ACM SIGMOD Record, Vol. 26, pp. 65-74, 1997.

[4] Peter Griffiths, "Slowly Changing Dimmensions : A Data Warehouse Ongoing Challenge", DM Review, 2001.

[5] R. Kimball, Data Warehouse ToolKit, John Wiley, 1996.

[6] Paulraj Ponniah, Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals, John Wiley, 2002.

[7] Goetz Graefe, "Patitioned B-trees - a user's Guide", BTW 2003', pp. 668-671, 2003.

[8] 박상근, 박순영, 정원일, 김명근, 배해영, "GMS: 공간 데이터베이스 관리 시스템", 개방형GIS학회, 2003.