

전자상거래 활성화를 위한 웹 웨어하우징의 유지 전략

이현창^U 김경창
홍익대학교 전산학과
{hcllee, kckim}@cs.hongik.ac.kr

Strategy for Maintenance of Web Warehousing to activate an Electronic Commerce

Hyun-Chang Lee^U Kyung-Chang Kim
Dept. of Computer Science, Hongik University

요 약

전자상거래는 비즈니스의 목적을 추구하기 위해서 거래 당사자들이 인터넷 등의 전자적인 매체를 이용하여 상품이나 서비스, 거래에 수반되는 경제활동을 일컫는다. 이러한 전자상거래는 기존의 상거래 형태가 고객의 필요와 요구에 부응해 고객과의 개별화된 일대일 관계를 유지하는 개별 고객 점유 형태로 탈바꿈하였다.

이러한 패러다임의 변화속에서 고객 중심의 패러다임을 추구하는 최적의 환경을 제공할 수 있도록 인터넷이 그 역할을 담당하고 있다. 이를 위해서 기술적으로 개별 마케팅이 이루어질 수 있는 분석이 필요하며, 이를 제공하는 환경이 바로 웹 웨어하우징이다. 그러나 전자상거래 플랫폼에서 웹 상에서 나타나는 데이터가 HTML형태이므로 실행시에 필요한 형태로 재구성하여야한다. 그러나 기존 연구에서는 복잡한 방법과 웹 웨어하우스에 소스 결과를 바로바로 적용할 수 없었다. 이에 본 연구에서는 웹 웨어하우스에 저장된 데이터를 정확하고 효율적으로 유지 관리할 수 있는 알고리즘을 제시하며 성능을 평가를 보인다.

1. 서론

비즈니스의 목적을 추구하기 위하여 거래 당사자들이 인터넷과 같은 네트워크 환경에서 전자적인 매체를 이용하여 상품이나 서비스의 거래 뿐 아니라 거래에 수반되거나 보조하는 제반 경제활동을 전자 상거래라고 한다. 이제까지 소비자에 의한 기업의 상품구매 수단으로 국한되어 온 전자상거래는 점차 기업 내 뿐 아니라 기업간 정보 교환에까지 그 범위가 확장됨에 따라 미래 기업의 궁극적인 목표인 가상기업을 실현하는 필수적인 수단으로 이해되어 가고 있다.

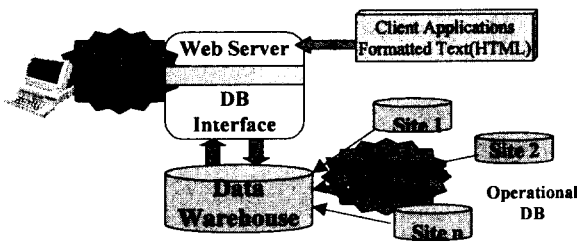
특히 기존의 상거래가 표준화된 고객집단으로부터 단일한 요구를 얻어 단일 품종으로 생산해 판매했던 대량생산과 시장점유의 패러다임이었다면 전자상거래에서 구현되는 패러다임은 개별 고객의 필요와 요구에 부응해 고객과의 개별화된 일대일 관계를 유지하는 개별화되며 고객점유의 패러다임이라고 할 수 있다[7].

이와 같은 패러다임의 변화 속에서도 인터넷은 기존의 매스 마케팅을 뛰어넘어 고객중심의 패러다임을 구현하는 개별 마케팅이 실현될 수 있는 최적의 환경을 제공하고 있다. 이러한 개별화된 관계형성의 가능성은 또한 전자상거래의 의미를 확장해 주고 있다. 이를 위해서는 기술적으로 개별 마케팅이 이루어질 수 있는 분석이 필요

하며, 이러한 환경을 제공할 수 있도록 해주는 것이 웹 웨어하우징(web warehousing)이다. 이를 <그림 1>에 도시하였다.

웹 웨어하우징은 주로 데이터 웨어하우스로 부터 정보를 알리기 위한 목적으로 웹기술을 사용하는 데이터 웨어하우징이다[2]. 일반적으로 웹 웨어하우징은 웹 브라우저를 사용하여 접근할 수 있어야하며, 인터넷이나 인트라넷을 통해 연결되며 데이터베이스로부터 동적으로 내용을 생성시킬 수 있는 기능을 제공해야한다.

이를 위해서 웹 데이터 웨어하우스의 형태는 크게 가상(virtual)의 형태와 실체화된(materialized) 형태로 존재한다[6,8]. 가상의 웹 웨어하우스는 흔히 CGI 스크립트에 의해 동적으로 사용자 요구를 계산하는 반면에, 실체화된 웹 웨어하우스는 미리 계산되어서 정적 HTML 페이지로 저장된다. 전자상거래 플랫폼에서 웹 상에서 나타나는 데이터는 HTML(Hyper Text Markup Language) 형태로 서버에 저장된 것이 아니라 데이터베이스에 있는 내용을 실행 시에 필요한 형태로 재구성하는 것이므로 웨어하우스 데이터베이스에 저장된 데이터를 상품관리, 주문관리, 배달관리, 회원관리 등에 적용하여 개발하면 웹과 전자상거래의 데이터베이스가 직접적으로 연동되어 관리할 수 있게 되므로 효율적이다. 이에 본 논문에서는 웹 웨어하우스에 저장된 데이터를 정확하고



<그림 1> 웹 웨어하우스의 구조

로 수행을 위해서 보내며, 둘째는 U_{i-1} 번째, 즉 S_{2-1} 번째 사이트와 U_{i+1} 번째인 S_3 번째에 소스에 대해 효율적으로 유지 관리할 수 있는 알고리즘을 제시하여 신속한 질의 응답을 이룰 수 있도록 한다.

본 논문에서 제2장은 관련연구를 살펴봄, 3장에서는 논문에서 제시하고있는 뷰 유지 알고리즘에 대해 설명한다. 4장에서는 다른 알고리즘들과 비교하며, 5장에서 결론 및 향후연구 방향으로 구성하였다.

2. 관련연구

기존에 연구되었던 전자상거래 계층별 분류체계는 인터넷 페이지를 포함한 애플리케이션 계층과 상거래를 위한 기반 서비스 층인 미들웨어 계층, 그리고 물리적인 네트워크 계층으로 구분할 수 있다.

미들웨어는 애플리케이션과 네트워크 계층사이에서 서비스를 연결해 주는 소프트웨어라 할 수 있으며, 주로 데이터베이스 연결 기술과 메시징 기술, 디렉토리 서비스 등을 들 수 있다. 미들웨어란 애플리케이션이나 데이터베이스, 운영체제, 네트워크의 트랜스포트 계층 사이에 위치하는 소프트웨어로, 데이터베이스와 애플리케이션들이 여러 개의 서버에 존재하게 되는 분산처리 환경에서 고객이 원하는 정보를 얻기 위해 네트워크 상에 분산되어 있는 운영체제, 데이터베이스 및 애플리케이션들을 연결해 주는 소프트웨어를 말한다[1].

미들웨어는 분산되어 있는 애플리케이션과 애플리케이션, 애플리케이션과 데이터베이스, 데이터베이스와 데이터베이스를 연결함으로써, 분산 애플리케이션 환경을 하나의 로컬 애플리케이션 환경처럼 운용할 수 있도록 해준다. 특히 데이터베이스와의 연결을 이용한 기술을 살펴보면 전자상거래 플랫폼에서 웹 상에서 나타나는 데이터는 HTML(Hyper Text Markup Language) 형태로 서버에 저장된 것이 아니라 데이터베이스에 있는 내용을 실행 시에 필요한 형태로 재구성하는 것이다. 이와같은 구성은 질의 응답 시간을 증가시키는 문제점이 발생한다.

이를 해결하기 위한 방법으로 점진적 방법을 들 수 있다. 이들 알고리즘으로는 단일 소스 환경의 보상알고리즘[9]과 재생성된 질의 결과를 모두 받을 때 까지 뷰에 반영 할 수 없는 Strobe[10] 알고리즘과 복잡한 유지 방법인 SWEEP[11] 알고리즘을 들 수 있다.

3. 웹 웨어하우스의 유지 전략

본 장에서는 웹 웨어하우스의 점진적 유지를 위한 효율적인 알고리즘에 대해서 살펴보기 전에 소스에서 발생하는 갱신은 튜플 삽입 혹은 삭제 같은 하나의 튜플 액션으로 구성되었다고 가정한다. 첫 번째 할 일은 소스와 웨어하우스에서 발생하는 사건(event)들에 대해 살펴본다.

3.1 소스/웨어하우스의 사건들

소스와 웨어하우스의 사건들은 원자성을 가지며 같은 사이트에서 수행되는 일련의 연산에 해당한다. 하나의 사건 내에서는 상기 기술된 순서대로 수행이 이루어지며, 서로 다른 사건들 사이의 순서는 무관하게 수행될 수 있다고 가정한다.

1) 소스에서의 사건

- S_req_i : 발생한 갱신 정보 U_i 가 뷰 정보와 관련성이 존재하는지 확인하기 위해 웨어하우스로 정보 송신
- S_exe_i : 발생한 갱신정보 U_i 를 수행한후 결과를 전송한다.

2) 웨어하우스에서의 사건

- W_chk_i : 소스로부터 갱신 정보(S_req_i)를 받아서 SALUS에 등록후 U_i 가 뷰의 정의에서 나타난 테이블을 갱신하는지 검사한다.
- W_eva_i : SALUS에 등록된 갱신정보 U_i 보다 먼저 존재하는 갱신 $U_j(j < i)$ 중에서 i 번째 릴레이션에 대한 연산이 끝났는지를 평가하여 끝났으면 수행을 위해 소스로 보낸다.
- W_ans_i : 소스에서 보내온 응답 테이블 A_i 를 받아서 이를 기반으로 뷰를 갱신한다.

3.2 웹 웨어하우스 유지 예제

본 절에서는 세 개의 다른 소스에 저장되어있는 테이블들에 삽입 삭제 연산을 예로 들어 알고리즘을 살펴본다. $S1/r_1 : \underline{W} \ \underline{X} \ \underline{S2/r_2 : X} \ \underline{Y} \ \underline{S/3r_3 : Y} \ \underline{Z}$
 $\quad \quad \quad 1 \ 2 \quad \quad \quad 2 \ 3$

$$SALUS = \emptyset$$

뷰의 정의 $V = \prod_{w,v} (r_1 \bowtie r_2)$ 라 가정하고 웨어하우스의 실제 뷰(MV)는 $MV = \{\}$ 으로 초기화되었으며, SALUS는 비어있는 상태로 $SALUS = []$ 이다. SALUS에 들어가는 갱신 정보는 <갱신 ID, 소스/테이블 이름, 갱신될 값, 연산 종류> 형식으로 가정한다. 갱신 ID는 소스로부터 받은 갱신 정보에 부여되며, 순차적으로 증가되는 값이다. 테이블 이름은 소스에서 연산이 일어날 테이블 대상이며, 갱신될 값은 삽입과 삭제 연산에 따라서 삽입될 값 혹은 삭제될 값이다. 연산 종류에는 삽입 연산과 삭제 연산이 존재하며, 수정 연산은 삭제와 삽입 연산으로 수행한다.

다음 예제는 소스에서 2개의 갱신이 U_1, U_2 순서로 발생되며, 웨어하우스에도 순서대로 도착한다고 가정한다. 소스에서 발생한 각 갱신 정보는 다음과 같다. $U_1 = insert<r_2, (2,4)>$, $U_2 = insert<r_1, (3,2)>$. $SALUS = [<U_1, S_2/r_2, (2,4), insert>, <U_2, S_1/r_1, (3,2), insert>]$ 이다.

웨어하우스는 U_1 에 대한 평가를 위해서 3가지 형태로 분리하여 소스로 보내진다. 하나는 갱신이 발생된 사이트로 보내지며, 나머지 연산은 갱신이 발생된 사이트를 중심으로 좌측과 우측으로 조인 결과를 얻기 위해 같은 연산을 수행하기 위해서 각기 보내진다. 이들 연산중 S_2 에서는 삽입연산이 수행되며, S_1 에서는 r_{10} [2,4]를 수행하며, S_3 에서는 r_{10} [2,4]를 수행하여 각각 그 결과를 웨어하우스로 보낸다. 웨어하우스에서는 조인 결과를 받아서 더 이상 평가를 수행할 사이트가 존재하지 않으면 뷰에 반영하여 최신의 상태를 유지한다. U_2 에 대한 연산도 같은 과정을 거쳐서 수행하도록 한다.

4. 성능 평가

본 논문의 웹 웨어하우징 방법은 기본적으로 SALUS를 이용한 점진적 접근 방법을 사용하였다. 특히 성능 평가의 공정성을 위해서 본 논문에서는 [10,11]과 같은 평가 기준 환경에서 연구가 이루어졌다. 이결과 본 논문에서 제시하는 알고리즘은 웹 웨어하우징을 유지하기 위한 복잡성을 크게 감소시켰으며, <표 1>과 같은 결과를 얻었다.

<표 1>

위 실험결과를 통하여 알 수 있듯이 SALUS를 이용한 결과가 메시지 처리시 비용이 절반으로 줄어들음을 알 수 있었다. 이같은 결과는 갱신이 발생한 사이트를 중심으로 좌측과 우측으로 나누어서 병렬로 수행후 웨어하우스에서 최종적으로 결과를 서로 조인함으로써 가능하였다. 이에 필요한 과정 및 중간 과정은 [11]을 참조하였다.

5. 결론

전자상거래는 비즈니스의 목적을 추구하기 위하여 거래 당사자들이 인터넷 등의 전자적인 매체를 이용하여 상품이나 서비스, 거래에 수반되는 경제활동을 전자상거래라고 한다. 이러한 전자상거래는 기존의 상거래가 표준화된 고객집단으로부터 단일 요구를 얻어 단일 품종으로 생산하던 시장 점유 형태이었다면 오늘날은 고객의 필요와 요구에 부응해 고객과의 개별화된 일대일 관계를 유지하는 개별 고객 점유 형태의 시장 형태라고 말할 수 있다.

이러한 패러다임의 변화속에서 고객 중심의 패러다임을 추구하는 최적의 환경을 제공할 수 있도록 인터넷이 그

역할을 담당하고 있다. 이를 위해서 기술적으로 개별 마케팅이 이루어질 수 있는 분석이 필요하며, 이를 제공하는 환경이 바로 웹 웨어하우스이다. 그러나 전자상거래 플랫폼에서 웹 상에서 나타나는 데이터가 HTML 형태로 데이터베이스내에 저장되어 있지 않으므로 실행시에 필요한 형태로 재구성하여야 한다. 이때 웹 웨어하우스에 저장된 데이터를 정확하고 효율적으로 유지 관리할 수 있는 알고리즘이 필요하다.

이에 본 논문에서는 웹웨어하우스를 유지하고 관리하기 위해 단순하면서도 적은 비용으로 유지 관리가 가능한 알고리즘을 제시하였다.

6. 참고 문헌

- [1]. Caroline Long, Middleware : Overview, Datapro, 1444 Technologies to Watch., Feb, 1995
- [2]. R.D.Hackathorn, "Web Farming for the Data Warehouse", 1999.
- [3]. GartnerGroup, "Key Electronic Commerce Trends", 1994.
- [4]. Nii, "A Framework for Electronic Commerce in the NII", 1996.
- [5]. 김명철, "웹을 위한 데이터베이스 통로의 분류체계", 1996.3 <http://grigg.chungnam.ac.kr/~uniweb/documents/3rdWWW/text.html>
- [6]. Alexandros Iabrinidis and Nick Roussopoulos, "WebView Materialization". ACM SIGMOD, May 2000
- [7]. Broadvision, Building Dynamic Web Applications with Broadvision ONE-TO-ONETM, White Paper, Apr., 1997. <http://www.broadvision.com/>
- [8]. Alexandros Iabrinidis and Nick Roussopoulos, "On the Materilization of WebViews". WebDB, 1999.
- [9]. Y.Zhyge,H.Garcia-Molina,J.Hammer, and J.Widom. View Maintenance in a Warehousing Environment, In SIGMOD, pages 316-327,May 1995.
- [10]. Y.Zhugе,H.Garcia-Molina, and Janet L.Wiener. The Stroe Algorithms for Multi-Source Warehouse Consistency. In Proceedings of the International Conference on Parallel and Distributed Information Systems, December 1996.
- [11].D.Agrawal,A. Abbadi,A.Singh,T.Yurek. Efficient View Maintenance at Data Warehouses. ACM SIGMOD pages 417-427, 1997.