データベースシステム 111－6
（1997. 1. 21）

通信量を考慮したデータウェアハウスの更新反映処理

粕川 雄也　　中西 通雄　　橋本 昭洋
大阪大学基礎工学部情報工学科
{kasukawa, naka, hasimoto}@ics.es.osaka-u.ac.jp

複数の分散された自律的なデータベースを統合する方法としてデータウェアハウス法がある。データウェアハウスは一種の実体化ビューとみなすことができるため，データウェアハウスの更新は実体化ビューの手法を利用できる。しかし，既存の実体化ビューの管理手法では，更新処理に伴う通信量や記憶量などのデータウェアハウスで考慮すべき問題にはあまり注目されていなかった。本稿では，実体化ビューに加えて，付加的に中間データを保存することによって，通信量を減らすデータウェアハウスの更新法を提案する。本提案法を用いれば，データウェアハウス本体に記憶しておくべき中間データ量を抑えつつ，通信量を減少させることができる。

Updating a data warehouse with low traffic cost

Takeya KASUKAWA　　Michio NAKANISHI　　Akihiro HASHIMOTO
Department of Information and Computer Sciences, Osaka University

Data warehousing is a technique for integrating distributed and autonomous databases. As a data warehouse can be regarded as a set of materialized views, it is expected that previous works on materialized views in central databases may be applied to update data warehouses. However, little work have considered traffic cost or space cost when a data warehouse is updated. In this paper, we show that a little amount of additional stored data in a data warehouse can reduce the traffic cost for updating the warehouse.

1 はじめに

複数のデータベース（要素データベース）が分散配置され，それぞれが独自に運用，管理されているような環境において，複数の要素データベースにまたがる検索の要求に応えるため，要素データベースを統合する研究が数多くなされてきた。その方法の一つに，データウェアハウス (data warehouse，以下ウェアハウス) を用いる方法がある [1]。ウェアハウスにはさまざまな定義があるが，本稿においては次の定義を用いる。ウェアハウスとは，各要素データベースから興味のあるデータのみを抽出して統合し，それを検索要求を受けるデータベース上に保存，管理する。ウェアハウスを利用することで，検索時に各要素データベースと通信する必要がなくなるため，高速な情報検索が可能となる。しかし，各要素データベースとウェアハウスが独立して運用されるため，要素データベースの更新をウェアハウスへ反映させるような機構が必要となる。この更新反映法として，ウェアハウスを実体化ビュー (materialized view) として考え，既存の解決法を適用する方法が提案されている [1–3]。本稿では各要素データベースおよびウェアハウスとして関係データベースモデル考える。また，実体化ビューのことを導出関係 (derived relation) と呼ぶ。

導出関係の更新反映法を利用した，ウェアハウスの更新手法には主に次のようなものがある [2]。

方法 1: ウェアハウスを管理するデータベースでは，導出関係だけを保存し，要素データベースが管理する基底関係を重複して持たない。更新反映処理に必要なデータは問い合わせ文を発行することにより得る。

方法 2: ウェアハウスを管理するデータベースでは，導出関係に加え，要素データベースの基底関係のコピーを保存し，基底関係と導出関係の両者の更新反映処理をする。

方法 2 を用いれば，基底関係の更新内容と導出関係だけで更新反映処理ができるので，転送すべき

データは基底関係の更新内容だけでよい [3]。しかし，要素データベースが科学データを扱ったデータベースである場合のように，データベースが巨大な場合には実現が困難となる可能性がある。また方法1では，ウェアハウスの更新に必要なデータを要素データベースから送信する必要があるので，転送すべきデータの量が問題となる。つまりウェアハウスの更新反映法では，通信量とウェアハウス側で記憶する基底関係の量がお互いにトレード・オフの関係にある。

本稿では，ウェアハウスを「比較演算付き conjunctive query で定義された導出関係」とみなす。比較演算付き conjunctive query で定義されたウェアハウスの更新のうち，削除についてはウェアハウスに現れる属性にすべての基底関係のキー属性を含めれば，削除された組のキー値を通信するだけでウェアハウスを更新することができる。しかし，挿入については，挿入された組だけからでは不十分であり，さらに別の情報を通信する必要がある。そこで，条件の一部を満たす組の各キー値をウェアハウス側に持つことで，記憶量を抑えつつ通信量を減らすような更新反映法を提案する。本提案法を，ウェアハウスが基底関係の自然結合であるような例で評価し，更新反映処理に伴なう通信量が方法1に比べ減少することを示す。また付加情報を保存するために必要な記憶領域は要素データベースの組の大きさに依存しないことから，方法2に比べ十分少ないことも示す。

2 準備

2.1 問い合わせ言語

本稿で用いる問い合わせ言語のクラスとして比較演算付き conjunctive query を考える [4]。比較演算付き conjunctive query Q は射影属性の集合 $A^Q = \{a_1, \cdots, a_p\}$，選択条件の集合 $C^Q = \{c_1, \cdots, c_q\}$，関係集合 $R^Q = \{R_1, \cdots, R_n\}$ の3つ組 (A^Q, C^Q, R^Q) で定義することができる。A^Q は問い合わせ結果に現れる属性の集合である。C^Q は Q の問い合わせ結果となる組が満たさなければならない条件で，各要素は $z_1 \theta z_2$ という形の原子式である。なお z_1，z_2 は属性または定数，θ は $=, <, \leq$ などの比較演算子である。なお，z_1 と z_2 がともに定数になることはないものとする。R^Q は A^Q，C^Q に現れる属性を含む関係の集合である。以上の定義で与えられた問い合わせ Q を関係代数式で示すと

X

X_1	X_2
a	あ
b	い
c	う

Y

Y_1	Y_2	Y_3
e	あ	i
f	う	k
g	え	j
h	お	k

Z

Z_1	Z_2	Z_3
i	4	イ
j	9	ロ
k	13	ハ

図 1: 関係データベースの例

次のようになる。

$$\pi_{A^Q}\sigma_{c_1 \wedge \ldots \wedge c_q}(R_1 \times \cdots \times R_n)$$

A^Q，C^Q，R^Q のそれぞれの右肩の Q は問い合わせ文が明らかな場合は省略する。

C^Q の要素 c_k を $c_k = z_{k1} \theta z_{k2}$ とする。z_{k1}，z_{k2} の一方が関係 R_i の属性であり，かつ，残りが定数の場合，または両者が関係 R_i の属性である場合，この c_k を関係 R_i の局所比較といい，関係 R_i のすべての局所比較の集合を関係 R_i の局所選択 $LS^Q_{R_i}$ と表す。さらに，すべての関係の局所選択の和集合を $LS^Q = \bigcup_{R_i \in R^Q} LS^Q_{R_i}$ とする。

また，z_{k1}，z_{k2} の一方が関係 R_i の属性であり，かつ，残りが関係 R_j の属性であるとき，この c_k を関係 R_i，R_j の大域比較といい，関係 R_i, R_j のすべての大域比較の集合を関係 R_i，R_j の大域選択 $GS^Q_{R_i,R_j}$ と表す。ただし $GS^Q_{R_i,R_j} = GS^Q_{R_j,R_i}$ とする。さらに，すべての関係の大域選択の和集合を $GS^Q = \bigcup_{R_i,R_j \in R^Q} GS_{R_i,R_j}$ とする。

関係 R_i の組を t_i とする。$LS^Q_{R_i}(t_i)$ は，$LS^Q_{R_i}$ に含まれる原子式の論理積に t_i の値を代入したものとする。つまり $LS^Q_{R_i}(t_i)$ は t_i の値を原子式に代入したときに，すべての原子式が真になれば真の値をとり，一つでも偽の原子式があれば偽の値をとる関数である。同様に，関係 R_i，R_j の組をそれぞれ t_i，t_j としたとき，$GS^Q_{R_i,R_j}(t_i, t_j)$ は $GS^Q_{R_i,R_j}$ に含まれる原子式の論理積に t_i，t_j の値を代入したものとする。

例 1 図1は関係データベースの例である。下線のついた属性がキー属性である。比較演算付き conjunctive query Q として次のものを考える。

$$\begin{aligned} A &= \{X_1, Y_1, Z_1, Z_3\} \\ C &= \{X_2 = Y_2, Y_3 = Z_1, Z_2 < 10\} \\ R &= \{X, Y, Z\} \end{aligned} \qquad (1)$$

このとき，

$GS_{X,Y} = \{X_2 = Y_2\}$

$GS_{Y,Z} = \{Y_3 = Z_1\}$

$GS_{X,Z} = \phi$
$LS_X = LS_Y = \phi$
$LS_Z = \{Z_2 < 10\}$
である。 □

2.2 関連モデル

文献 [5] の記法にならって，一般の比較演算付き conjunctive query Q に対する関連とキー関連を定義する。まず，関係 R_i と R_j の関連を次のように定義する。

定義 1 比較演算付き conjunctive query Q に対して，$\{\langle t_i, t_j\rangle | t_i \in R_i,\ t_j \in R_j,\ GS^Q_{R_i,R_j}(t_i,t_j) \wedge LS^Q_{R_i}(t_i) \wedge LS^Q_{R_j}(t_j) = \text{真}\}$ で表される集合を Q に対する関係 R_i，R_j の関連といい，$rel^Q(R_i,R_j)$ と表す。 □

関連 $rel^Q(R_i,R_j)$ は，$GS^Q_{R_i,R_j}(t_i,t_j)$，$LS^Q_{R_i}(t_i)$，$LS^Q_{R_j}(t_j)$ がすべて真であるような関係 R_i，R_j の組 t_i, t_j の系列 $\langle t_i, t_j\rangle$ の集合である。

関係 $R_1, \cdots, R_n$ の関連は次のように定義する。

定義 2 比較演算付き conjunctive query Q に対して，$\{\langle t_1, \cdots, t_n\rangle | \forall R_i, R_j\ \langle t_i, t_j\rangle \in rel^Q(R_i,R_j)\}$ で表される集合を Q に対する関係 $R_1, \cdots, R_n$ の関連といい，rel^Q と表す。 □

次に関連の射影を定義する。

定義 3 関連を rel^Q，ある属性の集合を A とする。関連の要素 $r \in rel^Q$ から属性集合 A の値を取り出したものを，r の A に関する射影といい，すべての $r \in rel^Q$ の A に関する射影の集合を関連 rel^Q の A に関する射影という。 □

関連の派生形としてキー関連を考える。関連の要素は組の系列であるのに対し，キー関連の要素はキー属性値の系列である。以下では，関係 R_i のキー属性を $K(R_i)$ とし，組 t_i のキー属性値を $k(t_i)$ と表す。なお議論の簡単化のため，キー属性は単一の属性であるとする。

定義 4 比較演算付き conjunctive query Q に対して，$\{\langle k(t_i), k(t_j)\rangle | t_i \in R_i, t_j \in R_j,\ GS^Q_{R_i,R_j}(t_i,t_j) \wedge LS^Q_{R_i}(t_i) \wedge LS^Q_{R_j}(t_j) = \text{真}\}$ で表される集合を，Q に対する関係 R_i，R_j のキー関連といい，$krel^Q(R_i,R_j)$ と表す。 □

定義 5 比較演算付き conjunctive query Q に対して，$\{\langle k_1, \cdots, k_n\rangle | \forall R_i, R_j\ \langle k_i, k_j\rangle \in krel^Q(R_i,R_j)\}$ で表される集合を Q に対する関係 $R_1, \cdots, R_n$ のキー関連といい，$krel^Q$ と表す。 □

問い合わせ Q，関連 rel^Q，キー関連 $krel^Q$ について次の補題が成立する。

補題 1 比較演算付き conjunctive query を Q とし，Q の計算結果を E_Q とする。また Q に対して作られた関連 rel^Q の A^Q に関する射影を $(rel^Q)^d$ とする。$(rel^Q)^d$ の要素を関係モデルの組とみなすと，E_Q と $(rel^Q)^d$ は等しい。 □

補題 2 比較演算付き conjunctive query を Q とし，Q の関連を rel^Q，キー関連を $krel^Q$ とする。このとき rel^Q の $\{K(R_i) | R_i \in R^Q\}$ に関する射影は $krel^Q$ に等しい。 □

補題 3 R_i，R_j を関係，R_j' を R_j と同じ属性を持つ関係とする。このとき
$rel^Q(R_i, R_j \cup R_j') = rel^Q(R_i,R_j) \cup rel^Q(R_i,R_j')$
が成立する。またキー関連についても同様の式が成立する。 □

補題 4 比較演算付き conjunctive query Q_1，Q_2 が $C^{Q_1} = C^{Q_2}$ であり，また，R_i, R_j がともに R^{Q_1}，R^{Q_2} の要素のとき
$rel^{Q_1}(R_i,R_j) = rel^{Q_2}(R_i,R_j)$ が成立する。またキー関連についても同様である。 □

補題 5 比較演算付き conjunctive query Q とする。$R_i, R_j \in R^Q$ が $GS^Q_{R_i,R_j} = \phi$ とする。もし任意の $k \in [0, m-1]$ について $GS^Q_{R_{l_k},R_{l_{k+1}}} \neq \phi$ となる系列 $\langle R_{l_0}, \cdots, R_{l_m}\rangle$ $(i = l_0, j = l_m)$ が存在すれば，$rel^Q(R_i,R_j) \supseteq \{\langle t_0, t_n\rangle | \forall k \in [0, m-1] \langle t_k, t_{k+1}\rangle \in rel^Q(R_{l_k}, R_{l_{k+1}})\}$ が成り立つ。またキー関連についても同様である。 □

例 2 例 1において

$$rel^Q(X,Y) = \left\{ \begin{array}{l} \langle (a, \text{あ}), (e, \text{あ}, i)\rangle, \\ \langle (c, \text{う}), (f, \text{う}, k)\rangle \end{array} \right\}$$

$$rel^Q(Y,Z) = \left\{ \begin{array}{l} \langle (e, \text{あ}, i), (i, 4, \text{イ})\rangle, \\ \langle (g, \text{え}, j), (j, 9, \text{ロ})\rangle \end{array} \right\}$$

$$rel^Q(X,Z) = \left\{ \begin{array}{l} \langle (a, \text{あ}), (i, 4, \text{イ})\rangle, \\ \langle (b, \text{い}), (i, 4, \text{イ})\rangle, \\ \langle (c, \text{う}), (i, 4, \text{イ})\rangle, \\ \langle (a, \text{あ}), (j, 9, \text{ロ})\rangle, \\ \langle (b, \text{い}), (j, 9, \text{ロ})\rangle, \\ \langle (c, \text{う}), (j, 9, \text{ロ})\rangle \end{array} \right\}$$

$rel^Q = \{\langle (a, \text{あ}), (e, \text{あ}, i), (i, 4, \text{イ})\rangle\}$

である。また，

$krel^Q(X,Y) = \{\langle a,e\rangle, \langle c,f\rangle\}$

$krel^Q(Y,Z) = \{\langle e,i\rangle, \langle g,j\rangle\}$

$krel^Q(Z,X) = \left\{ \begin{array}{l} \langle a,i\rangle, \langle b,i\rangle, \langle c,i\rangle, \\ \langle a,j\rangle, \langle b,j\rangle, \langle c,j\rangle \end{array} \right\}$

$krel^Q = \{\langle a,e\rangle, \langle e,i\rangle\}$

である。 □

2.3 前提および記法

ウェアハウスを，複数の要素データベースに分散した基底関係に対する比較演算付き conjunctive query W を定義とする導出関係とする。ただし，すべての R_i に対して $GS^W(R_i,R_j) = \phi$ となるような R_j は存在しないとする。議論の簡単化のため，R^W に含まれる基底関係 はそれぞれ別の要素データベースで管理されているものとし，基底関係 R_i を管理するデータベースを DB_i と表す。また，ウェアハウス W を管理するデータベースを DB^W と表す。

DB_i や DB^W には，次のような性質があるとする。

- すべてのデータベースでは関係データベースモデルを使用し，問い合わせ式は 比較演算付き conjunctive query のみを受け付ける。
- 関係 R_i のキー属性 $K(R_i)$ は単一の属性である。またすべての基底関係の属性名は異なる。
- 任意の DB_i と DB^W の間にはエラーフリーな通信路が存在する。
- 任意の DB_i では，自分が管理する基底関係への問い合わせのみを処理できる。他のデータベースが管理する基底関係の内容が必要となる問い合わせを処理するには，問い合わせを発行する側が問い合わせの一部として，他の要素データベースの内容を送らなければならない。
- 任意の DB_i では，自分が管理する基底関係について，指定されたある時点から現在までの間にどのような更新がされたか (更新前の基底関係，更新後の基底関係，挿入，削除された組) を知ることができる。

基底関係の更新は挿入および削除のみを考える。変更操作は，変更前の組を削除し，変更後の組を挿入したものと考える。基底関係 R_i に挿入された組は関係 I_{R_i}，削除された組は関係 D_{R_i} として問い合わせ時に参照できるとする。なお，I_{R_i}，D_{R_i} の属性などの定義は R_i と同じである。

ウェアハウス W の射影属性 A^W には全基底関係のキー属性が含まれているものとする。つまり $\{K(R_i)|R_i \in R^Q\} \subseteq A^W$ である。

3 提案手法

本章では，ウェアハウス更新反映法を，前回更新反映処理した時点から，基底関係 R_1 で挿入のみ発生した場合，基底関係 R_1 で削除のみ発生した場合，すべての基底関係で挿入と削除の両方が発生した場合の3つに分けて説明する。なお，ウェアハウスの更新処理中にすべてのデータベースで別の更新は発生しない (もしくは，更新前のデータベースにアクセスできるような機構が存在する) と仮定する。

3.1 単一の基底関係の挿入時

導出関係が2つ以上の基底関係に対して定義されているとき，基底関係に挿入された組と更新前の導出関係の内容から，更新後の導出関係を求めることは一般にできない [6,7]。従って，基底関係への挿入操作をウェアハウスに反映させるためには，要素データベースへの問い合わせが必要である。要素データベースに組が挿入されたとき，それを比較演算付き conjunctive query で定義されたウェアハウスに反映させる場合，以下の定理が成立する [2,8-11]。

定理 1 比較演算付き conjunctive query で定義されたウェアハウス W について，基底関係 R_1 に組 I_{R_1} が挿入されたとき，W に挿入すべき組は $(A^W, C^W, \{I_{R_1}, R_2, \cdots, R_n\})$ という問い合わせ I_W の結果と等しい。 □

I_W を計算するには，いくつかの部分問い合わせを要素データベースに発行し，返された結果を統合する必要がある。もしその部分問い合わせ結果があらかじめ計算されていれば，データ通信量を少なくすることができるので，どのようなデータをあらかじめ計算しておけばよいかを考える。

補題 1より，I_W を求めるには I_W に対する関連 rel^{Iw} を求めればよい。また定義 2，補題 5より，関連 rel^{Iw} を求めるには，関連 $rel^{Iw}(R_i,R_j)$ $(i \neq 1, j \neq 1)$ および関連 $rel^{Iw}(I_{R_1},R_i)$ $(i \neq 1)$ を求めればよい (ただし，$R_i, R_j \in R^W$)。

ここで，$GS^W_{R_i,R_j} \neq \phi$ である2つの関係 R_i，R_j の関連 $rel^W(R_i,R_j)$ が実体化され DB^W で管理されており，要素データベースへ問い合わせを発行せずに求めることができるとする。補題 4より，$rel^W(R_i,R_j) = rel^{Iw}(R_i,R_j)$ $(i \neq 1, j \neq 1)$ であるので，I_W を求めるためには，関連 $rel^{Iw}(I_{R_1},R_i)$ $(i \neq 1)$ のみを DB_i への問い合わせにより求めれ

ばよい。また，関連 $rel^W(R_i, R_j)$ を管理するためには，この関連自体の更新も必要になるが，定理1から，$rel^W(R_1, R_i)$ $(i \neq 1)$ に I_W の計算時に求まる関連 $rel^{I_W}(I_{R_1}, R_i)$ を追加すればよいことが導かれる。

しかし，関連 $rel^W(R_i, R_j)$ の実体を保存するには，W と比べてより大きな記憶領域が必要となる。そこで，関連ではなくキー関連 $krel^W(R_i, R_j)$ を実体化することを考える。$krel^W$ を求める方法は，rel^W を求める方法と同様である。しかし，I_W を求めるには rel^W が必要であるので，$krel^W$ から rel^W を求めなければならない。補題2やキー値の性質から，rel^W の要素と $krel^W$ の要素の間には一対一対応があるので，$krel^W$ の要素から rel^W の要素を求めるには，$krel^W$ の要素に含まれるキー値から対応する組の値 (ただし実際に必要なのは A^W に含まれる属性の値のみ) を求めればよい。この値は要素データベースから求めなければならないが，処理に必要なデータ通信量の大きさは，I_W の計算結果の大きさとほぼ同じである。また，場合によってはこの組の値は DB^W にある W の中から抽出できる場合があるので，実際の通信量はさらに少なくなる可能性がある。

例 3 例1において 問い合わせ式 (1) を W の定義とする。W の組は $(a, e, i, イ)$ のみである。今，関係 X に組 $(d, え)$ が挿入されたとする。このとき，$krel^W(I_X, Y) = \{\langle d, g\rangle\}$ であることが問い合わせにより求まる。従って，$krel^{I_W}$ は $\langle d, g, j\rangle$ である。$Z_1 = j$ である関係 Z の組の属性 Z_3 の値は (ロ) であることが問い合わせにより求まるので，W に挿入すべき組は $(d, g, j, ロ)$ である。 □

3.2 単一の基底関係の削除時

削除の更新反映は，count 法などの，問い合わせにより行う方法が提案されている [5,8,10]。しかし，射影属性にすべての基底関係のキー属性が含まれている場合には，$K(D_{R_i})$ の値を含むような W の組を削除すればよく，それ以外の要素データベースの内容を参照する必要はない [2,5,7,9]。

3.3 複数の基底関係の更新時

更新を反映処理した後に任意の関係 R_i で I_{R_i} の挿入と D_{R_i} の削除が発生した場合を考える。ただし，直前の反映処理時点から現時点までの間に，R_i に組 t が挿入された後に削除された場合は，t は I_{R_i} にも D_{R_i} にも含まれないものとする。

挿入と削除の両方が発生したときのウェアハウスの更新処理は，まず (1) 基底関係から削除された組を用いてウェアハウスから削除されるべき組を求め，(2) 基底関係に挿入された組からウェアハウスに挿入されるべき組を求めればよい。(1) については，単一の基底関係の削除時の方法を複数回適用して求めることができるので，(2) について説明する。

複数の基底関係への挿入をウェアハウスに反映させる方法については，次の定理が知られている [2,3]。なお，任意の基底関係 R_i について，更新前の R_i から D^{R_i} を削除したものを R_i^- とし，R_i^- に I_{R_i} を挿入したものを R_i^+ と表記する。

定理 2 $R_1, \cdots, R_n$ に対して，ある順序が決まり，R_i の順番を $order(R_i) \in [1, n]$ とする。次に $j \in [1, n]$ について $F(R_i, j)$ を次のように定義する。

$$F(R_i, j) = \begin{cases} R_i^- & order(R_i) < j \text{のとき} \\ I_{R_i} & order(R_i) = j \text{のとき} \\ R_i^+ & order(R_i) > j \text{のとき} \end{cases}$$

また，$I(j)$ を次の問い合わせの結果とする。

$A = A^W$, $C = C^W$,

$R = \{F(R_1, j), \cdots, F(R_n, j)\}$

このとき，比較演算付き conjunctive query で定義されたウェアハウス W について，複数の基底関係に組 I_{R_i} が挿入された時，W に挿入される組 I_W は，

$$\bigcup_{j \in [1,n]} I(j)$$

である。 □

従って，すべての $I(j)$ について 3.1節の手法を用いればよい。さらに，$krel^{I(j)}$ は，任意の2関係を $R_i, R_j \in R^W$ とすると，補題3，補題4を利用すれば，次の2関係のキー関連から求めることができる。(なお，$order(R_i) < order(R_j)$ とする)

$krel^W(R_i^-, R_j^-)$

$krel^{I(i)}(I_{R_i}, R_j^+)$

$krel^{I(j)}(R_i^-, I_{R_j})$

さらに，キー関連 $krel^W(R_i, R_j)$ の更新もこれらのキー関連から求めることができる。

4 アルゴリズム

複数の基底関係で挿入および削除が発生した場合の提案法を関係モデル上で実現したアルゴリズムを示す。前章において DB^W で実体化された $krel^W(R_i, R_j)$ は，アルゴリズム中では M_{R_i,R_j} という導出関係で表わす。

[M_{R_i,R_j} の定義] 任意の 2 関係 R_i, R_j の $GS^W_{R_i,R_j}$ について，$GS^W_{R_i,R_j} \neq \phi$ ならば，M_{R_i,R_j} を次の定義で作成する。

$A = \{K(R_i), K(R_j)\}$,
$C = GS^W_{R_i,R_j} \cup LS^W_{R_i} \cup LS^W_{R_j}$,
$R = \{R_i, R_j\}$

[削除に対する処理]

(1) 任意の DB_i に次の問い合わせ (A, C, R) を発行する。

$A = \{K(R_i)\}$, $C = LS^W_{R_i}$, $R = \{D_{R_i}\}$

この問い合わせ結果を $E_1(R_i)$ とする。もしすべての $E_1(R_i)$ が空であれば，ここで W の削除に対する処理を終了する。

(2) 任意の $R_i \in R^W$ について，W から属性 $K(R_i)$ の値が $E_1(R_i)$ に含まれる組を削除する。

(3) 任意の $R_i, R_j \in R^W$ について，M_{R_i,R_j} から属性 $K(R_i)$ の値が $E_1(R_i)$ に含まれる組および属性 $K(R_j)$ の値が $E_1(R_j)$ に含まれる組を削除する。

[挿入に対する処理]

(1) 任意の DB_i に次の問い合わせ (A, C, R) を発行する。

$A = \mathcal{A}(C^W) \cap R_i$, $C = LS^W_{R_i}$, $R = (I_{R_i})$

この問い合わせ結果を $E_1(R_i)$ とする。なお，$\mathcal{A}(C^W)$ は C^W に現れる属性の集合である。もしすべての $E_1(R_i)$ が空であれば，ここで W の挿入に対する処理を終了する。

(2) $GS_{R_i,R_j} = \phi$ でないような $R_i, R_j \in R^W$ について，次の (2.1), (2.2) を行う (ただし，$order(R_i) < order(R_j)$ とする)。

(2.1) DB_j に次の問い合わせ (A, C, R) を発行する。

$A = \{K(R_i), K(R_j)\}$,
$C = GS^W_{R_i,R_j} \cup LS^W_{R_j}$,
$R = \{E_1(R_i), R_j^+\}$

この問い合わせ結果を $E_{2.1}(R_i, R_j)$ とする。

(2.2) DB_i に次の問い合わせ (A, C, R) を発行する。

$A = \{K(R_i), K(R_j)\}$,
$C = GS^W_{R_i,R_j} \cup LS^W_{R_i}$,
$R = \{R_i^-, E_1(R_i)\}$

この問い合わせ結果を $E_{2.2}(R_i, R_j)$ とする。

(3) 任意の $j \in [1, n]$ について，次の問い合わせ (A, C, R) を求める。

$A = \{K(R_i) | R_i \in R^W\}$, $C = C^W$,
$R = \{R(R_1, j), \cdots, R(R_n, j)\}$

これは M_{R_i,R_j}, $E_{2.1}(R_i, R_j)$, $E_{2.1}(R_i, R_j)$ から求めることができる。この問い合わせ結果を $E_3(j)$ とする。

(4) 任意の $R_i \in R^W$ について，$E_3(1) \cup \cdots \cup E_3(n)$ からキー属性 $K(R_i)$ の値を射影したものを $E_4(R_i)$ とする。

$A = \{K(R_i)\}$, $C = \phi$,
$R = \{E_3(1) \cup \cdots \cup E_3(n)\}$

(5) DB_i に次の問い合わせ (A, C, R) を発行する。

$A = A^W \cap R_i$,
$C = \{R_i.K(R_i) = E_4(R_i).K(R_i)\}$,
$R = \{R_i, E_4(R_i)\}$

この問い合わせ結果を，$E_5(R_i)$ とする。

(6) 任意の $j \in [1, n]$ について，$E_5(R_i)$ $(R_i \in R^W)$ と $E_3(j)$ を自然結合させる。この計算結果を $E_6(j)$ とする。

(7) すべての $j \in [1, n]$ について，$E_6(j)$ を W に挿入する。

(8) 任意の $R_i, R_j \in R^W$ について，M_{R_i,R_j} に $E_{2.1}(R_i, R_j)$, $E_{2.2}(R_i, R_j)$ を挿入する。

5 評価

ウェアハウスの更新反映処理を問い合わせのみで行う方法と，本稿の提案手法を用いる方法の 2 種類について評価する。評価には分子生物学データのデータベースをもとにした次の例を用いる。

例 4 次の 4 つの基底関係がある。

$H(id, place, data)$, $G(id, place, seq)$,
$P(id, seq, name)$, $M(id, name, data)$

各関係の id がキー属性である。

これらに対して，次のウェアハウスを考える。

$$A^W = \left\{ \begin{array}{l} H.id, H.data, G.id, \\ P.id, M.id, M.data \end{array} \right\},$$

$$C^W = \left\{ \begin{array}{l} H.place = G.place, \\ G.seq \sim P.seq, \\ P.name = M.name \end{array} \right\},$$

$R^W = \{H, G, P, M\}$

ここで，$G.seq \sim P.seq$ は，$G.seq$ の中に $P.seq$ が含まれていれば真，含まれていなければ偽となる比較演算とする。

各属性 a の大きさ $S(a)$[bytes] は次のとおりとする。

関係	属性	大きさ
H	id	8
	$place$	30
	$data$	100
G	id	8
	$place$	30
	seq	10000

関係	属性	大きさ
P	id	8
	seq	1000
	$name$	30
M	id	8
	$name$	30
	$data$	100

関係 R に含まれる組の数 $T(R)$ は次のとおりとする。

関係	組の数
H	1000
G	100000
P	2000
M	100

関係 R_i の組1つに対して，GS_{R_i,R_j} を満たす関係 R_j の組数の期待値を J_{R_i,R_j} で表す。$J_{G,H}=1$，$J_{P,G}=10$，$J_{M,P}=10$ である。また，$J_{H,G}=100$，$J_{G,P}=\frac{1}{10}$，$J_{P,M}=\frac{1}{2}$ とする。 □

評価は，(1) 問い合わせの際に要素データベースと DB^W の間でやりとりされるデータの総量 (通信量) および，(2)W，M_{R_i,R_j}，全基底関係のそれぞれを保存するために必要な量 (記憶量) について行う。なお，通信量として，要素データベースから送られるデータの量と，問い合わせに必要となる他要素データベースの内容の量だけを考え，問い合わせ文自体の量や通信時のヘッダなどは考えないこととする。

5.1 通信量

複数の基底関係で挿入が発生した場合の通信量を例4を用いて評価する。関係 H，G，M，P に挿入された組をそれぞれ ΔH，ΔG，ΔM，ΔP とする。また，ΔH，ΔG，ΔP，ΔM の組の数をそれぞれ，$T(\Delta H)$，$T(\Delta G)$，$T(\Delta P)$，$T(\Delta M)$ とする。

5.1.1 問い合わせのみで求める場合

定理2より，ウェアハウスの更新には次の4つの式を計算する必要がある。

1. $(A,C,R)=(A^W,C^W,\{H,G,P,\Delta M\})$
2. $(A,C,R)=(A^W,C^W,\{H,G,\Delta P,M\})$
3. $(A,C,R)=(A^W,C^W,\{H,\Delta G,P,M\})$
4. $(A,C,R)=(A^W,C^W,\{\Delta H,G,P,M\})$

1つめの問い合わせを計算する手順は，既存の最適化法を用いると，次のようになる。

1. ΔM を求める。
2. P を管理するデータベースに次の問い合わせを発行する。
   $A=\{P.id,P.seq,\Delta M.id\}$,
   $C=\{P.name=\Delta M.name\}$,
   $R=\{P,\sigma_{id,name}(\Delta M)\}$
   この結果を $\Delta P\cdot M$ とする。
3. G を管理するデータベースに次の問い合わせを発行する。
   $A=\{G.id,G.place,\Delta P\cdot M.id\}$,
   $C=\{G.seq\sim\Delta P\cdot M.seq\}$,
   $R=\{G,\sigma_{\Delta P.id,seq}(\Delta P\cdot M)\}$
   この結果を $\Delta G\cdot P\cdot M$ とする。
4. H を管理するデータベースに次の問い合わせを発行する。
   $A=\{H.id,H.data,\Delta G\cdot P\cdot M.id\}$,
   $C=\{H.place=G.place\}$,
   $R=\{H,\sigma_{\Delta G.id,place}(\Delta G\cdot P\cdot M)\}$
   この結果を $\Delta H\cdot G\cdot P\cdot M$ とする。
5. ΔM，$\Delta P\cdot M$，$\Delta G\cdot P\cdot M$，$\Delta H\cdot G\cdot P\cdot M$ を各キー値により自然結合し，属性集合 A^W の値を射影する。

これらの手順における通信量の期待値は，

- ΔM の計算：
  $T(\Delta M)\{S(M.id)+S(M.name)+S(M.data)\}$
- $\Delta P\cdot M$ の計算：
  $T(\Delta M)\{S(M.id)+S(M.name)\}$
  $+J_{M,P}T(\Delta M)\{S(P.id)+S(P.seq)+S(M.id)\}$
- $\Delta G\cdot P\cdot M$ の計算：
  $J_{M,P}T(\Delta M)\{S(P.seq)+S(P.id)\}$
  $+J_{M,P}J_{P,G}T(\Delta M)\{S(G.id)+S(G.place)+S(P.id)\}$
- $\Delta H\cdot G\cdot P\cdot M$ の計算：
  $J_{M,P}J_{P,G}T(\Delta M)\{S(G.place)+S(G.id)\}$
  $+J_{M,P}J_{P,G}J_{G,H}T(\Delta M)\{S(H.id)+S(H.data)+S(G.id)\}$

の合計であり，計算すると $40,416T(\Delta M)$ である。同様に，2つめ，3つめ，4つめの式の計算に伴なう通信量の期待値を計算すると，それぞれ $4,142T(\Delta P)$，$20,214.2T(\Delta G)$，$2,003,996T(\Delta H)$ となる。従って，問い合わせのみでウェアハウスに挿入する組を求めるのに必要な通信量はこれらの合計となる。

5.1.2 提案手法を用いる場合

提案手法では，前章のアルゴリズム中の次の処理で伴なうデータ転送が発生する。

- $E_1(H)$，$E_1(G)$，$E_1(P)$，$E_1(M)$ の計算
- $E_{2.1}(H,G)$，$E_{2.2}(H,G)$ の計算

- $E_{2.1}(G,P)$, $E_{2.2}(G,P)$ の計算
- $E_{2.1}(P,M)$, $E_{2.2}(P,M)$ の計算
- $E_5(M)$ の計算
- $E_5(H)$ の計算

これらの計算に必要な通信量を計算すると，$2,372T(\Delta H) + 20,223.4T(\Delta G) + 3,470T(\Delta P) + 11,952T(\Delta M)$ となり，問い合わせのみの場合に比べて M への挿入については $\frac{1}{4}$ に，H への挿入については $\frac{1}{1000}$ に減少する。

5.2 記憶量

次に，ウェアハウス，導出関係 M_{R_i,R_j}，全基底関係のそれぞれが必要とする記憶量を求めて比較する。

- ウェアハウス
  ウェアハウスに存在する組の数は，$J_{M,P} \cdot J_{P,G} \cdot J_{G,H} \cdot T(M)$ である。また，ウェアハウスの組1つあたりの大きさは，A^W に含まれる属性の大きさの合計である。ウェアハウスの保存に必要な記憶量はこれらを掛けたものになり，計算すると 2,320,000 バイトとなる。
- 導出関係 M_{R_i,R_j}
  導出関係 M_{R_i,R_j} は $M_{H,G}$, $M_{G,P}$, $M_{P,M}$ の3つであり，組の数はそれぞれ
  $J_{M,P} \cdot J_{P,G} \cdot J_{G,H} \cdot T(M)$,
  $J_{M,P} \cdot J_{P,G} \cdot T(M)$, $J_{M,P} \cdot T(M)$ である。また，組1つあたりの大きさはそれぞれ
  $S(H.id) + S(G.id)$, $S(G.id) + S(P.id)$,
  $S(P.id) + S(M.id)$ である。従って，導出関係 M_{R_i,R_j} の保存に必要な記憶量の合計を計算すると，336,000 バイトとなる。
- 全基底関係
  基底関係 H, G, P, M の組の数および組1つあたりの大きさは，例4に示されており，それを計算すると，およそ1ギガバイトとなる。

以上から中間関係の保存に必要な記憶量はウェアハウスや全基底関係が必要とする記憶量に比べて小さいといえる。

6 まとめ

本稿で提案した手法は，既存の問い合わせによる方法と比べて，任意の比較演算付き conjunctive query で通信量が減少するとは限らない。例えば，GS_{R_i,R_j} を満たす組がほぼすべての基底関係中の組であるような場合では，中間関係の組数が非常に多くなる。このような場合に対しては，本稿での提案手法と既存の手法を組合せることで解決できると思われる。

参考文献

[1] Widom, J.: Research Problems in Data Warehousing, *Proceedings of 4th CIKM* (1995).

[2] Zhuge, Y., Garcia-Molina, H., Hammer, J. and Widom, J.: View Maintenance in a Warehousing Environment, *Proceedings of ACM SIGMOD*, pp. 316–327 (1995).

[3] Quass, D., Gupta, A., Mumick, I. S. and Widom, J.: Making Views Self-Maintainable for Data Warehousing (1995). http://www-db.stanford.edu/pub/quass/1995/sm.ps.

[4] Klug, A.: On Conjunctive Queries Containing Inequalities, *Journal of ACM*, Vol. 35, No. 1, pp. 146–160 (1988).

[5] 木實新一, 古川哲也, 上林弥彦: 導出データを持つデータベースにおける更新処理, 信学技報, DE91-37, pp. 69–78 (1991).

[6] Blakeley, J. A., Coburn, N. and Larson, P.-A.: Updating derived relations: detecting irrelevant and autonomously computable updates, *ACM TODS*, Vol. 14, No. 3, pp. 369–400 (1989).

[7] Gupta, A., Jagadish, H. and Mumick, I. S.: Data Integration using Self-Maintainable Views, Technical Report Technical Memorandom 113880-941101-32, AT&T Bell Laboratories (1994).

[8] Blakely, J. A., Larson, P.-A. and Wm., T. F.: Efficiently updating materialized views, *Proceedings of ACM SIGMOD*, pp. 61–71 (1986).

[9] Ceri, S. and Widom, J.: Deriving production rules for incremental view maintenance, *Proceedings of 17th VLDB*, San Mateo, CA, USA, Morgan Kaufmann, pp. 577–589 (1991).

[10] Gupta, A., Mumick, I. S. and Subrahamanian, V.: Maintaining views incrementally, *Proceedings of ACM SIGMOD*, pp. 157–166 (1993).

[11] Gupta, A. and Mumick, I. S.: Maintenance of Materialized Views: Problems, Techniques, and Applications, *IEEE Data Engineering Bulletin*, Vol. 18, No. 2, pp. 3–18 (1995).