関係DBMSビューサポートサブシステムのLUPによる構成法について

増永良文（東北大学　電気通信研究所）

1. はじめに

現代のデータベース管理システム（DBMS）体系はANSI/X3/SPARCレポート［TSIC78］に報告されている様に，応用プログラム（利用者）の物理的かつ論理的データ独立性を保証する為に，云わゆる内部，概念，外部スキーマの三層構成をとる傾向にある。　この体系中では利用者視野（user's　view，これを本稿ではビューと呼ぶ）は外部スキーマによって定義される。　しかしながら現在のデータベースシステムでは商用システムであれ研究用システムであれ外部スキーマをサポートしているシステムはほとんどなく，従って利用者の論理的データ独立性はほとんど実現されていない［KIM79］。外部スキーマがサポートされる為には外部スキーマと概念スキーマ間のインタフェースが確立されていなければならないが，現状ではこのインタフェース設計に困難さが存在しているということである。　解決しなければならない主要な問題点として次の二点が考えられる：

（1）少くとも一人の利用者視野をサポートする為に必要な外部スキーマと概念スキーマ間のインタフェースの構成問題，

（2）マルチプルビューがお互いに干渉されることなくサポートされる為の干渉現象の制御問題。

本稿はこれら問題解決の第一ステップとして（1）の問題を外部スキーマ，概念スキーマ共に関係モデルで記述されていて，関係DBMSが一個のビューをサポートするとした場合のインタフェース構成（本稿ではビューサポートサブシステムの構成と呼ぶ）を提示している。　本構成はビュー定義木上を動作するLUP（Local　view　Update　Processor）の動作を記述するという形で提示され，LUP動作は著者が既に主張しているビュー及びビュー更新の"意味"に従って規定される［MASU79］。

2. ビューとビュー更新問題

2.1　ビューとビュー定義木

データの関係モデルでは実体の属性と実体相互間の関連は数学的関係として表現される。　［CODD70］で定義された通り，n個の属性ドメイン$dom(A_1)$，$dom(A_2)$，・・・，$dom(A_n)$（ここに$dom(A_i)$は属性A_iのドメインを表わす）上の関係$R(A_1, A_2, \cdots, A_n)$は直積$dom(A_1) \times dom(A_2) \times \cdots \times dom(A_n)$の（これを本稿では簡略に$dom(R)$と表わす）有限部分集合である。　基本関係とはディスク等の物理的記憶媒体上に実現されている関係のことを云い，それら全体が関係データベースの概念スキーマを構成しているとする。　ビューはそれら基本関係に一連の関係代数演算やAVARAGE等の計算関数（但し本稿では議論を簡明にする為にこれら計算関数は考察の対象としない）を適用して得られる仮想的関係のことをいう。　元来関係代数演算としては4つの通常の集合演算（直積，和集合，

共通集合，差集合演算）と4つの関係代数独自の演算（射影，結合，商，制限演算）からなっている。　しかしながらこれら8つの演算は極小集合ではなく，本稿では一つの極小集合をなす次の5つの演算（直積 $\otimes$，和集合 $\cup$，差集合 $-$，射影 $R[A]$，制限 $R[A\theta B]$）をビューを定義する為に使える演算とする。この体系のもとでは，例えば関係 R のドメイン A 上と関係 S のドメイン B 上での θ-結合は $R[A\theta B]S \equiv (R\otimes S)[A\theta B]$ と定義され，R の A 上の S の B 上による商は $R[A\div B]S \equiv R[\bar{A}]-((R[\bar{A}]\otimes S[B])-R)[\bar{A}]$，ここに $\bar{A}$ は A の R の属性集合に関する補集合，と定義される。　一般に任意のビューが上述5つの演算を使って定義出来るが，その定義を木状に表現したものをビュー定義木と云うことにする。　ビューはその木の根になり，それを定義している基本関係が葉となっている。（ビュー定義木の概念は既に［OSHA79］にみられる。）　例えば二つの基本関係 ED（EMP，DEPT）と DM（DEPT，MGR）の自然結合ビュー EDM は $EDM \equiv ((ED\otimes DM)[DEPT^1 = DEPT^2])[EMP^\frown DEPT^2{}^\frown MGR]$ と定義され，その定義木は図−1に示されている。（ここに DEPT の右肩についている数字1と2は各々その DEPT は ED，DM の属性であることを明示する為に用いている。）

2.2　ビュー更新問題と意味論

ビューからのデータ読出しは常にサポート出来るが，ビューを更新（書込，削除，書換）しようとすると必ずしもその要求が実現されるとは限らない。　これはビューが一般に基本関係から適当な関係代数演算を用いて導出された仮想的な関係であり，それへの更新要求は丁度所望の更新結果を実現する様な基本関係の更新が存在して始めて実現されるからである。　この見地からビュー更新問題は次の二つの問題点を有する：

（1）ビューの更新可能性とは一体何か，

（2）ビュー更新実現の為の具体的手法の解明。

これら二つの問題は複雑に絡むが，本節では（1）の問題考察にはビュー及びビュー更新の意味を十分に考慮する必要のあることを特に指摘し，次章以降では逆に（2）の問題点を中心に考察を深め，（1）の問題解決も目指す。

さて，基本関係 ED，DM の意味を形式的に M_{ED}，M_{DM} と表わす。　これらは各々 $dom(ED)$，$dom(DM)$ 上の述語であり，具体的には $dom(ED)$ の元であるペアー $(e1, d1)$ が $M_{ED}(e1, d1)=T$（真）であるのはその時点で“従業員 e1 が売場 d1 で働いている”時及びその時のみであり，従って $(e1, d1)$ がその時点での ED のインスタンスの元である時及びその時のみである。　ED と DM の自然結合ビュー EDM の意味は ED と DM の意味を使って次の様に定義される：

$$\text{(E1)}\quad (\forall t\in dom(EDM))(M_{EDM}(t)\equiv M_{ED}(t[EMP^\frown DEPT])\wedge M_{DM}(t[DEPT^\frown MGR]))$$

今，ED と DM のある時点におけるインスタンスは図−2に示される通りとする。　ビュー edm（edm なる小文字列はビュー EDM のインスタンスを強調する時に使用する）からタップル $(e1, d1, m1)$ を削除しようとする。これは（E1）式より“$M_{EDM}(e1, d1, m1)$ が偽であるのは（i）$M_{ED}(e1, d1)$ が偽であって $M_{DM}(d1, m1)$ は真であるか，（ii）$M_{ED}(e1, d1)$ は真であって $M_{DM}(d1, m1)$ が偽であるか，あるいは（iii）M_{ED}

$(e1, d1)$も$M_{DM}(d1, m1)$も共に偽であるのか，のいづれかでなければならない”ことが判る。大事なことはedmからの一タップル$(e1, d1, m1)$の削除命令が発せられたとすれば，その命令の根底には(i)，(ii)，(iii)いづれかの事象が発生していたという事を意味的考察は主張している事であり，従ってedmからのこの場合のタップル削除命令としては(a) EMP値が$e1$でDEPT値が$d1$なる全てのタップルを削除する命令か，(b) DEPT値が$d1$でMGR値が$m1$なる全てのタップルを削除する命令か，(c) EMP値が$e1$でDEPT値が$d1$であるタップルかDEPT値が$d1$でMGR値が$m1$であるタップルを全て削除する命令，のいづれかしかありえないことが判るということである。この考え方が次章以降，更新変換の基本的考え方となっている。

3．LUPによるビューサポートサブシステムの一構成法

3．1　理論的基礎

［誘引された構造的保全制約］　直積ビューが満さねばならない保全制約で，それが二つの関係の直積であるという定義から構造的に課せられるものである。つまり二つの関係を$R(A1, A2, \cdots, An)$と$S(B1, B2, \cdots, Bp)$とする。この時直積ビュー$R \otimes S$は次の保全制約を満さなければならない。これを本稿では以下ISIC（Induced Structural Integrity Constraint）と呼ぶことにする：

(E-2)　$(\forall t, t' \in dom(R \otimes S))((t, t' \in R \otimes S)\ implies\ ((t[A1 \wedge A2 \wedge \cdots \wedge An] \wedge t'[B1 \wedge B2 \wedge Bp]) \in R \otimes S))$.

尚，以下の説明の都合上，次に述べる記号法を導入する。D，Iで各々削除，挿入文を表わす。Rを一般に関係とする時，$res(D, R)$でRにDが施こされた結果得られるであろう所望の結果関係を表わす。（$res(I, R)$も同様に定義する。）$diff(D, R) = R - res(D, R)$と定義する。（同様，$diff(I, R) = res(I, R) - R$。）ISICの主張していることは次の通りである：“直積ビュー$R \otimes S$に対する削除文DがR及び（又は）Sに対する削除文に云わゆる削除的副効果（deletion side-effect）なく変換される為の構造的必要かつ十分条件は$(\forall t, t' \in dom(R \otimes S))((t, t' \in res(D, R \otimes S))\ implies\ ((t[A1 \wedge A2 \wedge \cdots \wedge An] \wedge t'[B1 \wedge B2 \wedge \cdots \wedge Bp]) \in res(D, R \otimes S)))$が成立することである”。$R \otimes S$に対する挿入文Iについても全く同様なことが成立する。

［更新文変形則］　Vをビュー定義木中の根ではない直積ビューとし，Vの一レベル下のビューはあるA，B，θに対して$V[A \theta B]$であったとする。DをVへの削除文とする。この時Dを次の二つの条件を満すD′に変形する規則を削除文変形則という：(1) $diff(D', V) \supsetneq diff(D, V)$，(2) $diff(D', V) \cap V[A \theta B] = diff(D, V)$。尚，この規則の運用についてはLUP動作の記述を行なう際述べる。

次に同様Vをビュー定義木中の根ではない直積ビューとする。IをVへの挿

入文とする。 この時、Iをdiff(I', V)⊇diff(I, V)なる条件を満す他の挿入文I'に変形する規則を挿入文変形則と云うことにする。 削除の場合と同様この規則の運用についてはLUP動作の記述の節で述べる。
〔補間則〕 射影ビューV[A]に対する挿入文を一レベル上のビューVへの挿入文に変形する際、Vの内包的、外延的性質を使用してVに新たに挿入されるタップルのA以外の属性値を出来る限り補間しておくという規則である。 これは次節以降述べるLUPの仕事である。 例えば自然結合ビューedmに一タップル(e5, d3, m4)を挿入する文は一レベル上のビューe[d=d]mへの一タップル(e5, d3, d3, m4)を挿入する文に変形される。 尚この問題は関係モデルの云わゆるnull値問題と深く絡む。

3.2 LUP

LUPとは関係DBMSのビューサポートサブシステムをインプリメントしようとする為のvehicleである。 それはビュー定義木上を動くプロセッサーで、最初根に位置し、そこに発せられた更新文が一レベル上のノード(一つの場合と二つの場合がある)の関係への更新文に変換可能であるか否かを検証し、もしそれが可能なら変換された更新文を当該ノードに出力し、出力先のノードに移動する。(もし二つのノードに出力が出た場合、LUPは二つに複製されて、当該ノードに移動する。) 変換不可能なことが判明すればその旨利用者にアナウンスし変換を止める。 LUPは上述の動作を繰返す。 もしビュー定義木上のLUPが全ていずれかの葉に到達し得たならば、ビュー更新の為の仮計算が開始される。 この仮計算は(i)基本関係の更新が保全制約に触れることなく実行されるか、と(ii)ビュー更新により云わゆる付加的副効果(additive side-effect)がどこかで発生しないか、を検証する為である。 仮計算が支障なく実行終了すればその通り基本関係(従ってビュー)を更新する。 尚、仮計算時のLUPの同期については3.4節でもう一度述べる。

3.3 LUPの動作

LUPの動作をそれが駐在しているノードのビューの種類、更新が削除であるのか挿入であるのかに大別して記述する。 これらの動作は第二章で提示したビュー及びビュー更新の意味論的立場にその理論的基礎を置き記述されている。 次の記号法を導入する: ビュー定義木の各ノードには重複しない非負整数が割当てられる。 一般にノードnは一つの先祖anc(n)と高々二つの子孫$des_L(n)$と$des_R(n)$(一つの場合はdes(n))を持つ。 根の先祖はφ(空)である。 rel(n)でノードnの関係を表わすことにする。

[削除文Dの変換動作]

<u>動作D-1-1</u> (rel(n)が直積ビューでanc(n)=φの時): ISICがres(D, rel(n))に対して成立するか否か検証する。 成立すれば$des_L(n)$と(又は)$des_R(n)$にdiff(D, rel(n))[α]、ここにαはrel($des_L(n)$)の全属性集合、と(又は)diff(D, rel(n))[β]、ここにβはrel($des_R(n)$)の全属性集合、を削除する文D_{Lout}と(又は)D_{ROOT}を出力し、出力先のノードにLUPは移動する。 成立しなければそこで変換不可能を利用者にアナウンスし変換を止める。

<u>動作D-1-2</u> (rel(n)が直積ビューでanc(n)≠φの時): ISICがres(D, rel(n))に対して成立するか否か検証する。 成立すれ

ばLUPは動作D－1－1と同様に動作する。成立しないならrel（anc（n））がrel（n）の制限ビューであるか否かをみる。もしそうなら削除文変形則の適用を開始し，res（D′, rel（n））に対しISICが成立しかつ変形が極小である，つまりD′以外に $diff(D', rel(n)) \supsetneq diff(D'', rel(n)) \supsetneq diff(D, rel(n))$ でres（D″, rel（n））に対しISICが成立するD″が存在しない，D′を見つける。もし見つかればそのD′に対して動作D－1－1と同様にLUPは動作する。それ以外は全て変換不可能とアナウンスし変換を止める。

動作D－2（rel（n）が和集合ビューの時）：$rel(n) = rel(des_L(n)) \cup rel(des_R(n))$ であり，rel（n）の形式的意味記述は $(\forall t \in dom(rel(n)))(M_{rel(n)}(t) \equiv M_{rel(des_L(n))}(t) \vee M_{rel(des_R(n))}(t))$ である。従ってrel（n）からタップルtを削除するためには，$rel(des_L(n))$ と $rel(des_R(n))$ から共にtを削除しなければならないことが判る。（これが意味論的LUP動作規定の一例である。）従ってLUPはdiff（D, rel（n））を $rel(des_L(n))$ と $rel(des_R(n))$ から削除するべく D_{Lout}，D_{Rout} を当該ノードに出力し，複製されて移動する。

動作D－3（rel（n）が差ビューの時）：意味論的に $rel(n)$ $(= rel(des_L(n)) - rel(des_R(n)))$ からタップルtを削除するには（i）$rel(des_L(n))$ からtを削除するか，（ii）$rel(des_R(n))$ にタップルtを挿入するか，（iii）（i）と（ii）を共に実行するか，のいずれかの曖昧性が存在することが判る。従って（a）会話方式を導入し，利用者に上のいずれの意味でrel（n）への削除文Dが発せられたのかをきき，その結果に基づいてLUPの動作を決定する本動作か，（b）変換にここで意味論的曖昧性が生じてしまったとしてこれ以降の変換を中止する，この動作を動作D－3′とする，のいずれかの動作が考えられる。

動作D－4（rel（n）が射影ビューの時）：$rel(n) = rel(des(n))[A]$ で，rel（n）の形式的意味記述は $(\forall t \in dom(rel(n)))(M_{rel(n)}(t) \equiv (\exists u \in dom(rel(des(n))))(u[A] = t \wedge M_{rel(des(n))}(t)))$ であるから，rel（n）からタップルtを削除することはrel（des（n））から $u[A] = t$ となる全てのタップルuを削除することに等しい。この様にLUPは動作する。

動作D－5（rel（n）が制限ビューの時）：$rel(n) = rel(des(n))[A\theta B]$ の時，LUPはrel（n）に対する削除文Dをrel（des（n））に対し，Dの指定すると同じ条件を満すrel（des（n））のタップルtで $t[A]\,\theta\,t[B]$ なるものを削除しようとする文D′に変換する。この変換は［STON75］の質問変形（query modification）の手法で常に可能である。

［挿入文Iの変換動作］

動作I－1－1（rel（n）が直積ビューで $anc(n) = \phi$ の時）：ISICがres（I, rel（n））に対して成立するか否か検証する。成立するなら動作D－1－1の場合と同様 $des_L(n)$ と（又は）$des_R(n)$ にdiff（I, res（n））［α］と（又は）diff（I, res（n））［β］

を挿入しようとする文 D_{Lout} と D_{ROUT} を出力し，出力先に移動する。もし成立しなければLUPは利用者に変換不可能をアナウンスし変換を中止する。

動作I－1－2（rel（n）が直積ビューでanc（n）≠φの時）：ISICがres（I，rel（n））に対して成立するか否か検証する。もし成立していれば動作I－1－1と同様にLUPは動作する。成立しなければres（I'，rel（n））に対しISICが成立する様Iを最小変形しI'を求める。この変形は常に可能である。このI'についてLUPは動作I－1－1と同様動作する。

動作I－2（rel（n）が和集合ビューの時）：動作D－3で発生したと全く同様な意味での意味論的曖昧性が生じる。従ってLUPは（a）利用者にrel（n）へのタップルtの挿入は（i）rel（des_L（n））へのtの挿入を意味しているのか，（ii）rel（des_R（n））への挿入を意味しているのか，（iii）（i）と（ii）を共に意味しているのか問い合せて動作を決定するのか，（b）意味論的曖昧さの為に変換不可能とするのかのいずれかの動作をする。この場合動作I－2'と云うことにする。

動作I－3（rel（n）が差集合ビューの時）：意味論的考察からLUPはIをdiff（I，rel（n））をrel（des_L（n））に挿入する挿入文 I_{Lout} とそれをrel（des_R（n））から削除する削除文 D_{Rout} に変換し出力し複製されて移動する。

動作I－4（rel（n）が射影ビューの時）：rel（n）（＝rel（des（n））[A]）に挿入しようとする各タップルtについて，A以外の属性値についてはnullとしたタップルuを作り，それをrel（des（n））に挿入する様にLUPはIを変換する。尚変換された挿入文に3．1節で述べた補間則を適用して値を補うかどうかはノードdes（n）に駐在するLUPの仕事となる。この問題は云わゆる関係モデルのnull値問題と深くかかわるが本稿ではこれ以上議論しない。

動作I－5（rel（n）が制限ビューの時）：LUPはrel（n）に対する挿入文Iをrel（des（n））に対し，Iの指定すると同じ条件を満すrel（des（n））のタップルtでt[A]θt[B]なるもののみを挿入しようとする文I'に変換する。この変換は動作D－5の場合と同様，質問変形の手法を用いて常に可能である。

3．4 更新実行制御

[実行制御] 前節でLUPの動作を述べた。もしある時間の後に全ての変換が成功しビュー定義木上の全てのLUPが葉以外のノード上には駐在しなくなったならLUPはビュー更新の為の仮計算を開始する。LUP軌跡という言葉でLUPが駐在したノードの張るビュー定義木の部分木を云うことにする。LUPによる（仮）更新動作の同期をとる為にLUP軌跡の分岐点には道標（milestone）をおく。LUPはLUP軌跡の葉から根の向きに更新を実行してゆき道標のある時点で相手方からのLUPがそこ迄降りて来るのを待つ。この同期がとれた時点で二つのLUPは一つに合併され，以下同様な実行パターンが繰り返される。尚注意しなければならないのは削除文の実行に関しては動作D－3でタップルの挿入文が誘起されない限り，云わゆる削除的副効果は動作で規定された変換の性質上起り得ず，基本関係のみ仮実行の後，直ちに実実行に入

ってよい。挿入文についての実行や削除文でも動作D−3で挿入文への変換が生じた場合，云わゆる付加的副効果が生じるかもしれないので，次に述べる制御が必要で，かつ仮実行を行なわなければならない。
[付加的副効果の制御] LUPが動作I−1−2を採った事により生起するかもしれないので，LUP軌跡のノードにこの動作が採られた場合星印（＊）を付随させることにする。更新の仮実行はこの星印のついたノードより下に来た場合，付加的副効果が発生していないかどうか検証しながら行なわれる。ビューに所望の挿入が副効果なく行なわれる事が判明すれば実実行に移る。尚，挿入の場合でもLUP軌跡に星印が一つも付随されなかった場合，削除の場合と同様仮実行は不要で直ちに実実行に入る。

4. 例題

ビューは図−1で定義されている通りのEDM，そのインスタンスは図−2に示されている通りとする。edmビューに対して削除文D_0が発せられたとする。

D_0："edmからd1とm1をDEPT^{2}値とMGR値に持つ全てのタップルを削除せよ。"

LUPは動作D−4によりD_0をe[d=d]mへの削除文D_1に変換ノード1に移動する。

D_1："e[d=d]mからd1とm1をDEPT^{2}値とMGR値に持つ全てのタップルを削除せよ。"

次いで動作D−5によりD_1はeddmへの削除文D_2に変換されLUPはノード2に移動する。

D_2："eddmからd1とm1をDEPT^{2}値とMGR値に持ちかつDEPT^{1}値とDEPT^{2}値が等しい全てのタップルを削除せよ。"

eddmは直積ビューなので，ノード2のLUPはISICがres(D_2, eddm)に対して成立しているか否か検証する。この場合成立していない。このノードは根ではないので削除文変形則が適用可能かどうか検証する。この場合それが可能でその結果D_2はD_3に変形される。

D_3："eddmからd1とm1をDEPT^{2}値とMGR値に持つ全てのタップルを削除せよ。"

従ってノード2でのLUPは動作D−1−2により，D_3をノード4の基本関係dmへの削除文D_4に変換する。実実行が直ちに行なわれ所望の削除がビューedmで行なわれる。

D_4："dmからd1とm1をDEPT^{2}値とMGR値に持つ全てのタップルを削除せよ。"

尚，この例題の場合のLUP軌跡を図−3に示す。

5. おわりに

LUPを用いた関係DBMSのビューサポートサブシステムの一構成法を示した。LUPを導入した利点は次の通りである。

(1) ビュー定義木上を動くLUPは局所的な動作(この数は有限)のみを行なえばよい。
(2) LUP動作の積み重ねとしてビュー更新変換過程を均一,統一的に記述出来る。
(3) LUPの挙動は丁度我々がビューサポートサブシステムをインプリメントする場合の一手順を示している。

[謝辞] 本研究を遂行する機会を与えて下さった本学野口正一教授に感謝する。又これ迄本研究に関連して御討論下さった皆様に感謝する。尚,本研究は昭和54年度文部省科学研究費補助金,課題番号468008の援助のもとで行なわれたことを付記する。

[文献]

[CODD70] Codd, E.F., "A relational model of data for large shared data banks," CACM 13, 6, 1970, p. 377-387 (1970)
[KIM79] Kim, W., "Relational database systems," Computing Surveys 11, 3, p. 185-211 (1979)
[STON75] Stonebraker, M., "Implementation of integrity constraints and views by query modification," Proc. 1975 SIGMOD Conf., ACM, N.Y., p. 65-78 (1975)
[TSIC78] Tsichritzis, D.C. and A. Klug (ed.), "The ANSI/X3/SPARK DBMS framework, report of the study group on database management systems," Inform. Systems. 3, p. 173-191 (1978)
[MASU79] 増永,"関係データベースでの利用者視野を通してのデータ処理について,"情報処理学会DBMS研究会資料,14-3 (1979年7月)

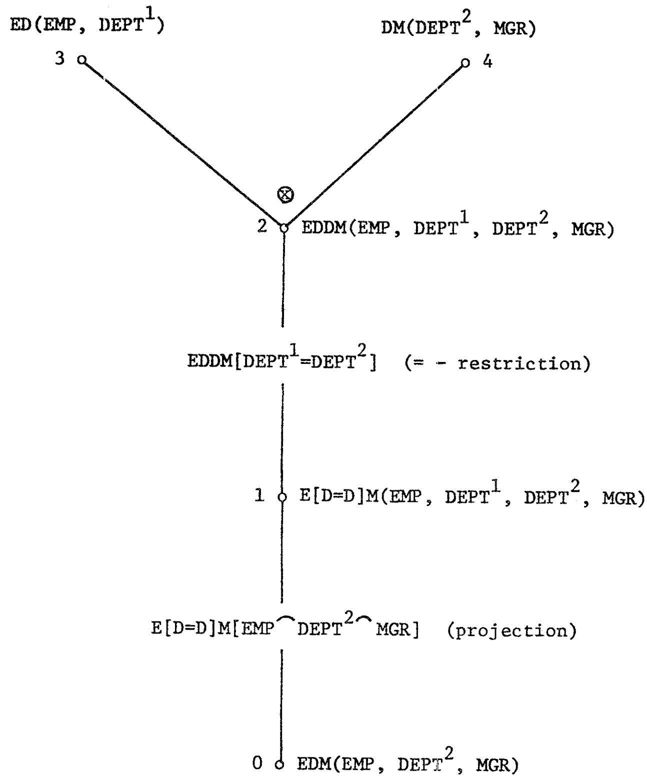

図-1. ビュ-EDMのビュ-定義木

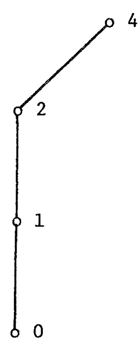

図-3. LUP軌跡(ビュ-edmに削除文D_0の場合)

ed:

ED	
EMP	DEPT^1
e1	d1
e2	d1
e3	d2
e4	d3

dm:

DM	
DEPT^2	MGR
d1	m1
d1	m2
d2	m3

eddm:

EDDM				
EMP	DEPT^1	DEPT^2	MGR	
e1	d1	d1	m1	*
e1	d1	d1	m2	*
e1	d1	d2	m3	
e2	d1	d1	m1	*
e2	d1	d1	m2	*
e2	d1	d2	m3	
e3	d2	d1	m1	
e3	d2	d1	m2	
e3	d2	d2	m3	*
e4	d3	d1	m1	
e4	d3	d1	m2	
e4	d3	d2	m3	

e[d=d]m: This consists of five tuples of eddm marked *.

edm:

EDM		
EMP	DEPT^2	MGR
e1	d1	m1
e1	d1	m2
e2	d1	m1
e2	d1	m2
e3	d2	m3

図-2. ED, DM, EDDM, E[D=D]M 及び EDM の インスタンス.