データベース・システム 35－2
（1983. 5. 19）

複数計算機による共用データベースの一制御方式

弘末清悟　（富士通株式会社）

1 はじめに

ここで述べる複数計算機システムによるデータベースの共用制御方式の目的は，以下の二点に集約できる。

高信頼性

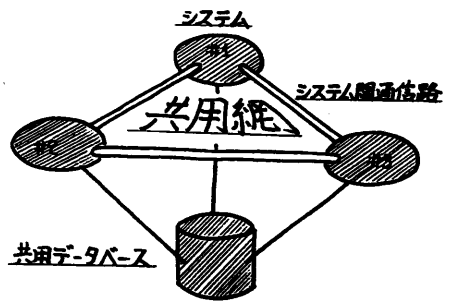
複数システムの一つがダウンしても，他の生存システムにより，データベース・サービスをフォールバック継続できる。

高処理能力

データベースを扱うトランザクションの処理を複数のシステムに分散することで，大量のトラヒックに対処できる。

上記の目的を実現するために，共用データベースを処理するシステム群の構成は図Aに示すような＜疎結合システム＞の形態をとり，データベースを直接的に共用する。

この制御方式にとって重要な概念は，図Aにも示されている＜共用網＞である。＜共用網＞とは，データベースを共用するシステムをノードとし，システム間をつなぐ通信路をパスとするネットワークのことである。本稿で述べる制御方式の特徴は，この＜共用網＞をあたかも一つのマルチプロセッサ・システムのように見せる点にある。その意味で，本制御方式を＜共用網制御方式＞と呼ぶことにする。



図A　共用データベース制御のシステム構成

2 共用データベース制御上の問題点

共用データベース制御上の主要な問題は，以下の二点にある。

・システム間排他制御

・復旧制御（ログデータの在り方等）

2.1 システム間排他制御の問題

＜共用網制御方式＞では，データベース・アクセスに関する排他制御をシステム間通信を利用して行う。

このシステム間通信方式の利点は，データベースの排他をＤＢＭＳの論理に応じた単位（グラニュー）で行えることである。即ち，単独システム上で実現されている排他制御をそのまま共用網へ延長できるわけである。

しかし，排他制御のためにシステム間通信を行うということは，それだけシステム・オーバヘッドを増大させることにもなる。

後述する＜多数決方式＞を中心とした体系は，このオーバヘッド問題を緩和する一つの答である。

2.2 復旧制御の問題

共用データベースを復旧するためのログデータを共用のログデータセットに取得するならば，そのデータセット自身の排他オーバヘッドの問題，あるいはログデータセットが共用網の処理能力上の隘路となってしまうといった問題などがあり，好ましくない。

＜共用網制御方式＞では，このような問題を避ける意味から，ログデータセットを各システムに分散して持たせ，ログデータを個別に取得させる。

しかし，ログデータを分散して取得させるということは，分散されたログデータ上において，データベースの更新順序を再現できる情報の設定が不可欠になることでもある。

このデータベース更新順序把握の問題は，後述する＜論理時刻＞の概念を導入することで解決される。

3 共用網制御方式の中核論理

前述したように，＜共用網制御方式＞の実現に関して中枢となる方式あるいは概念は，＜多数決方式＞と＜論理時刻＞である。

3.1 多数決方式による排他制御

＜多数決方式＞とは，図Bに示すように，共用網を構成するシステム群全体の過半数システムからのアクセス承認を受けるだけで，共用網としての承認とみなす方式である。

具体的には，図Bに示されているように，「データベース・グラニューの使用許可願（要求）」を共用網の中で順次回覧し――＜多数決回覧＞と呼ぶ――各システムから投じられた意見の多数決をとって共用網の判定とする方式である。

この＜多数決方式＞は，以下に述べる“規約”に従うことで成立する。

(1) 多数決回覧における投票内容とその規約

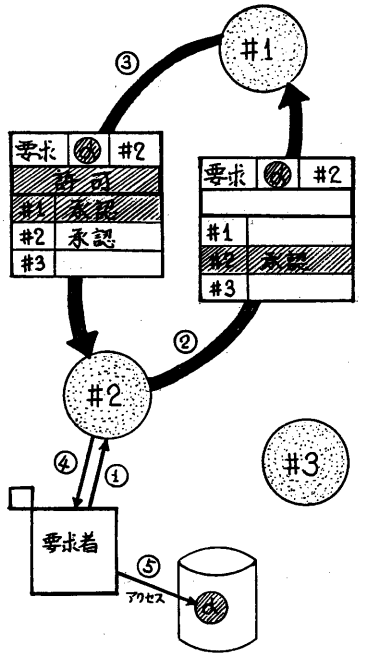
＜ADMIT ＞：承認

要求を認めることを示す。要求されたグラニューを“自身”の認識できる範囲では誰も使用していないと判断したシステムはこの＜ADMIT ＞を投ずる。

＜REFUSE＞：絶対拒否

要求を絶対的に拒否することを示す。要求されたグラニューを“自身”が使用中であるとき，そのシステムはこの＜REFUSE＞を投ずる。

＜REFUSE＞が投じられた時点で＜多数決回覧＞は中止され，後述の＜REJECT＞判定が下される。



① データベース使用要求（グラニューα）
② ＃2承認後の回覧（多数決は未成立）
③ ＃1承認による多数決成立（許可判定）
④ 共用網からの判定結果を要求者に通知

図B 多数決方式によるデータベース要求の排他制御

＜PASS＞：消極的拒否

要求を拒否するが，絶対的なものではないことを示す。要求されたグラニューは“自身”の認識できる範囲では誰かが使用中だが，その使用者が“自身”ではないとき，そのシステムはこの＜PASS＞を投ずる。

(2) 多数決回覧の結果判定

＜PERMIT＞：許可

要求が共用網から承認されたことを示す。この判定は，＜ADMIT ＞投票が過半数に達したとき下される。

＜REJECT＞：却下

要求が共用網から承認されなかったことを示す。この判定は，＜REFUSE＞が投じられたとき，あるいは，＜PASS＞投票が半数に達したとき下される。

(3) 投票後の状態管理

＜ADMIT ＞保留状態

多数決回覧に対し＜ADMIT ＞を投じたシステムがその判定結果を通知されていない状態をいう。

(4) 要求の衝突に対する対処

前述の保留状態を有するシステムは，その保留状態グラニューと同一のグラニューに対する使用許可要求を受けたとき，図Cに示す論理をとる。この論理は要求が衝突した場合，互譲によるデッドロックを防ぐための交通整理をすることに他ならない。即ち，要求の発生順序を後述の＜論理時刻＞によって判断し，早い方を優先する処置をとるわけである。

もちろん，＜論理時刻＞が＜現実世界＞の要求発生時刻に対応した順序関係を保っているとは限らない——逆転する場合もある。しかし，逆転現象が起るのは両者の＜現実世界＞における要求発生時刻が極めて近接していた場合に限られる。それ故，逆転現象は許容誤差範囲の問題でしかないといえる。

3.2 論理時刻の管理方式

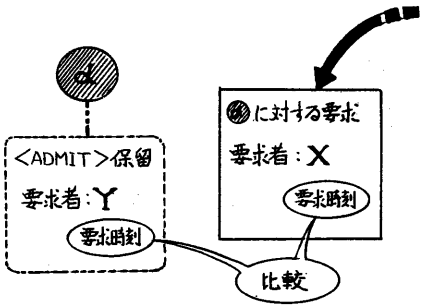
＜論理時刻＞とは，共用網における共用データベース処理の変化を表現するものであり，図Dに示す形式をとっている。

(1) 論理時刻の主要な役割り

・データベースの更新順序を表現する。即ち，図Eに示すように，分散取得されたログデータにその更新時刻として＜論理時刻＞を刻印することで，前述のログデータの分散問題に対処できる。

・データベース・グラニューの使用要求が衝突したとき，互譲によるデッドロックを回避する論理に利用される。

・トランザクション処理の進展具合を知るキー情報となる（図F参照）。



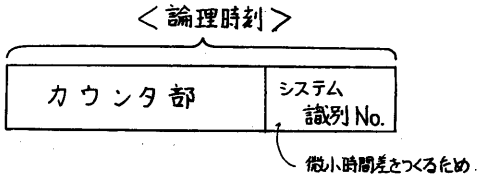
・Xが先発要求者の場合

➡ <PASS>を投じて回覧続行

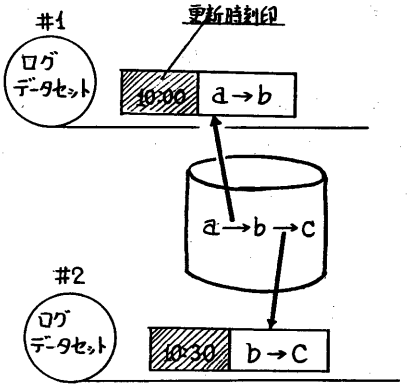
・Xが後発要求者の場合

➡ Yに対する結果が出るのを待つ

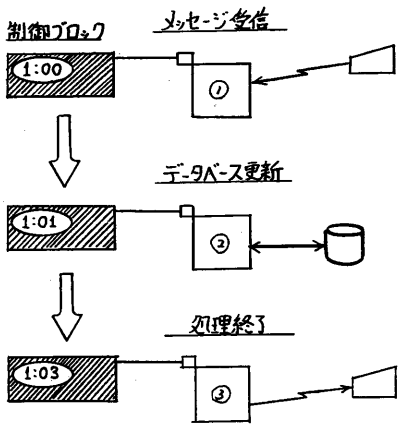
図C 要求衝突時の処理論理



図D 論理時刻の構成形式



図E 論理時刻による分散ログの正順化方式



図F　論理時刻によるトランザクション管理

(2)　論理時刻を表現する論理時計

＜論理時刻＞を表現する時計は，共用網に仮想的に存在する架空の＜共用網時計＞である。

このような架空の時計を具体化するために存在しているのが，以下に述べる＜システム時計＞と＜連絡時計＞である。図Gはこれらの関係を概念的に表したものである。

システム時計

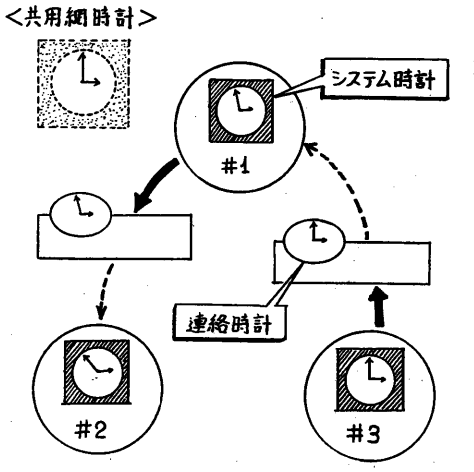
＜システム時計＞は，各システムに存在する論理的な時計（カウンタ）であり，後述の＜連絡時計＞と連動して架空の＜共用網時計＞に忠実に合せるように制御される。各システムの処理では，＜システム時計＞があたかも＜共用網時計＞であるかのごとく扱われる。

この＜システム時計＞は，共用データベースに関する処理（処理要求など）に呼応して時を刻み，次に述べる＜連絡時計＞によって補正される。

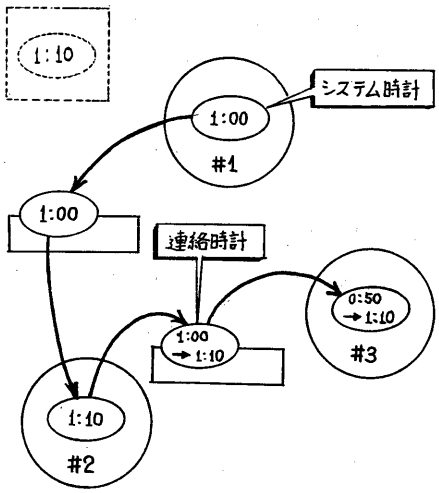
連絡時計

システム間通信の通信筒（パケット）ごとに組み込まれる論理的な時計である。

この＜連絡時計＞は，それが共用網を巡る過程で，各システムの＜システム時計＞の補正を行うと同時に



図G　共用網の中の各種論理時計の位置



図H　論理時計の相互補整方式

自身も必要な補正を受ける。その結果，共用網の中の各＜システム時計＞は互いに大きな狂いを生ずることなく，概ね架空の＜共用網時計＞に一致した時を刻むことができる。図Hは，この様子を示したものである。

4 システムのフォールバック制御

システムのフォールバック制御とは，システムダウンや通信路障害を原因としたシステム切離し，及びそのような現象に随伴するデータベースの最新性喪失問題の解決を実践する制御をいう。

共用網の許では，システムダウンや通信路障害はすべて，網を構成するシステム群が複数のグループに分裂したものと受け取ることができる（図Ｉ参照）。以降，これらの異常状態をまとめて＜分裂事象＞と呼ぶことにする。

4.1 分裂事象の検出

分裂事象の検出は，図Ｊに示すように，システム間の時間監視を引金にして開始される。

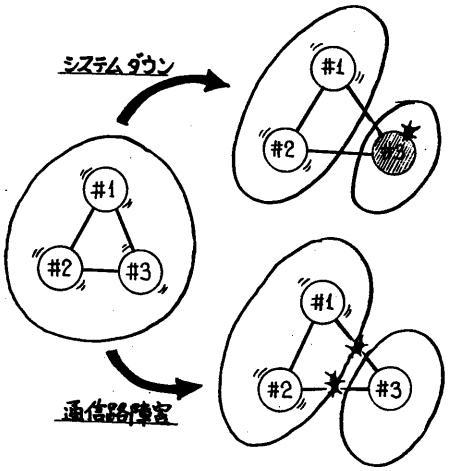
共用網制御方式の特徴は，分裂事象の検出を単独のシステムだけでは行わない点にある。即ち，図Ｊにあるように，相互に通信可能なシステム群（元の共用網構成システム群のサブグループ）が相互に同一の認識を持つことを確認しながら異常システム（群）を切り離す。

この分裂事象の検出処置自体が，相互通信可能なシステム群の把握をも兼ねていることは重要である。このことは，後述の共用データベース・サービスの継続を速やかに行うための伏線となっている。

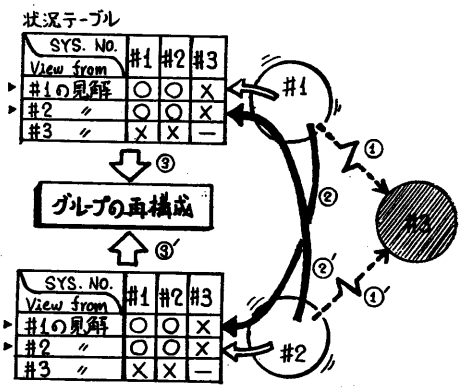
4.2 共用データベースの処理権

一般に，共用網の分裂事象を検出したからといってその検出グループだけが生存しているとは限らない。他にも生存グループが存在するかも知れない。

そこで注目されるのが，「どのグループに共用データベースの＜処理権＞を与えるか」といった問題である。無統制に＜処理権＞を与えたならば，データベースは破壊されてしまう。



図Ｉ　共用網の分裂事象



①／①' システム間診断（時間監視）
②／②' 各システムから見た状況の変化を通知
③／③' 相互認識が一致したことによる再構成

図Ｊ　分裂事象の検出と新グループの認識方法

共用網制御方式では，この＜処理権＞の扱いを，一般性を考慮して以下のように規約する。

過半数グループ優先規約

分裂後，元の共用網に対し過半数のシステム群を擁していたグループは，共用データベースの処理権を継続的に保持する。

少数グループの処理権放棄規約

分裂後，過半数を擁しきれなかったグループは共用データベースの処理権を無条件に放棄する。

以上の規約に従って，各グループは自らの振舞いを決定する。もし，過半数を擁するグループが存在しなかった場合は，操作員の権限でいずれか一つのグループに処理権が与えられる。

なお，以降，処理権を所有してデータベース・サービスを継続できるグループを＜継続グループ＞，処理権を持たないグループを＜離脱グループ＞と呼ぶことにする。

4.3 データベースの最新性喪失部分の閉塞

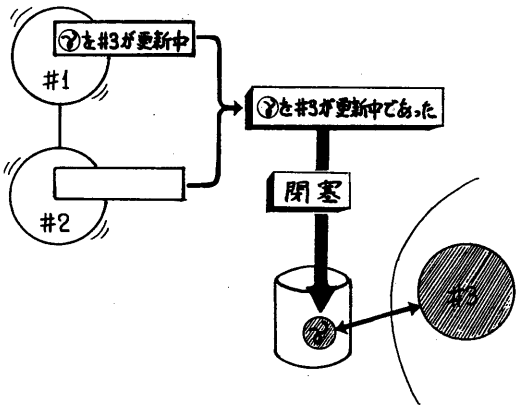
前述した方法によって＜継続グループ＞となったグループは，次に，分裂事象が誘発した共用データベースの最新性喪失部分を明らかにし，それを閉塞するか，あるいは復旧してしまうかのいずれかの処置をとる必要がある。＜継続グループ＞が実際にデータベース・サービスを継続できるのは，これらの処置が完了してからである。これらの処置が完了すれば，＜継続グループ＞は名実ともに新たな＜共用網＞となる。

＜共用網制御方式＞の主要な課題である「分裂事象発生後の速やかなデータベース・サービスの継続」を実現するために，次に述べる＜高速閉塞方式＞を採用する。

＜高速閉塞方式＞は，＜継続グループ＞が元の共用網に対し半数以上のシステム群を擁しているときに適用される。この条件を＜半数条件＞と呼ぶ。

この条件は，排他制御における＜多数決方式＞と密接に関係した条件である。即ち，最低半数のシステムが集まれば，そのうち少なくとも1システムは，あるデータベース・グラニューに対する最新要求を知っているわけだから，全体として，共用データベースの最新処理状況を推測することができる。

したがって，この＜半数条件＞を前提とした＜高速閉塞方式＞は，図Kに示すように，そのグループを構成するシステムが主記憶上に保持している共用データベース管理情報を参照して，＜離脱グループ＞のシステムが処理中だったデータベースのグラニューを閉塞するだけである。この処置は極めて速やかに完了する。



図K　データベースの高速閉塞処理

このように，＜高速閉塞方式＞はデータベース・グラニューを閉塞の単位とするので，十分に有効なデータベースのフォールバックがなされることになる。即ち，閉塞されるのは全データベース・グラニューのうち極めて微小な部分に過ぎないことになる。

なお，現実的な状況では，殆どの場合，この＜高速閉塞方式＞の適用が可能であると考える――半数以上の同時ダウンなど考えにくい。また，4.2 で述べたように＜継続グループ＞が過半数システムを擁していた場合には，見掛け上，データベース・サービスの滞りはない。

5 その他

システム間通信を前提とした＜共用網制御方式＞では，通信遅延あるいは通信文の脱送といった問題を無視できない。このような問題をフェイルセイフを基調として包括的に解決するために，時間監視が利用される。即ち，一定時間内にある要求の決着がつかなかった場合，要求者は一方的に取消し指示を発令する。この制御により，通信文の脱送などの問題が局所化され，問題を肥大化させないで済む。

6 最後に

ここで述べた＜共用網制御方式＞による共用データベースの制御は，所期の目的である高信頼性と高処理能力を目指す上で有力な方式の一つであると信ずる。

また，別の方向から，同様の目的を持ってアプローチされているものに分散データベースがある。

いずれにしても，分散志向は大きな“潮流”のように思われる。その意味からも，＜共用網制御方式＞の“良さ”を活かすべく，一層の工夫を凝らし，応用分野を拡げて行きたいと考えている。