

分散データベースシステムにおける通信と処理のモデル

田畑孝一 (京都大学情報処理教育センター)

1. はじめに

データベースの構造を示すために、種々のデータモデルすなわち階層型、ネットワーク型、リレーショナル型などのモデルや、あるいはデータベース管理システムのためのANSI/X3/SPARCの三層スキーマモデルなどがある。同様に分散データベースシステムのためのモデルが必要であり、ここではそのための一つのモデルを提案する。

一般に分散データベースでは複数のデータモジュール間で大量のデータを互いに移動し処理を行うので、このモデルではデータの移動(すなわち通信)とその処理の過程に注目しそれをレイヤ構造としてとらえる。このとらえ方は分散データベースをコンピュータネットワークアーキテクチャの応用階層における機能の一つとしてとらえるのとは視点を異にしている。このモデルはまたデータベースマシンにおけるデータの移動と処理の過程も表現している。

2. 分散データベースシステムにおける通信処理の基本概念

コンピュータネットワーク技術において、その基本的な機能は異なりホスト(あるいはターミナル)内に存在する一対のプロセス間における情報交換の機能(プロセス間通信)である(図1)。

この機能によってプロセスは互いにメッセージを交換する。送り出されたメッセージはそのまま相手に届けられるか、あるいは両端のプロセスが既知の定形的な変形が途中で加えられて届けられる。定形的な変形とは例えば文字コード変換などである。

このように、コンピュータネットワーク技術における通信の基本概念は一対の通信実体間におけるメッセージ通信であり、この概念は必ずしも分散データベースにむいたものとはいえない。分散データベースでは単なる「一対の通信実体間における通信」ではなく、「いくつものデータ実体間に行われる通信と処理」である。すなわち、単にデータの移動ではなく、与えられた条件を満たすようにデータ間で処理を行い、その結果得られたものを提供する。

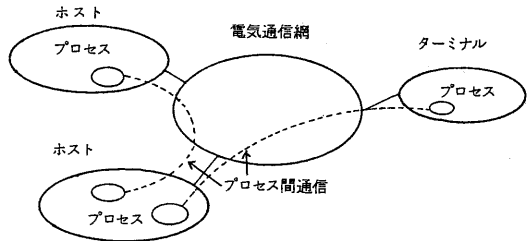


図1 コンピュータネットワークにおける通信

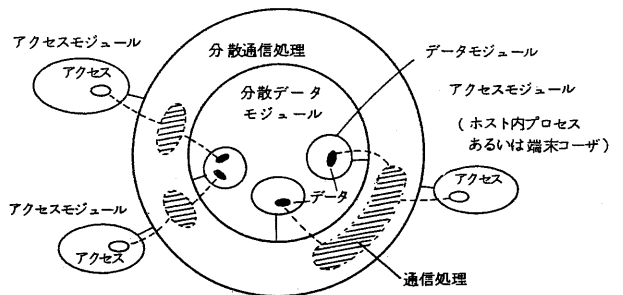


図2 分散データベースシステムにおける通信処理

分散データベースシステムにおける通信処理の基本概念を次のように考える(図2)。分散データベースシステムは、アクセスモジュール(ポスト内プロセスあるいは端末ユーザ)からの要求に応じて、複数データモジュールの中のデータ相互を通信・処理し、アクセスモジュールにデータを提供する(あるいはデータモジュール中のデータの更新を行う)。この場合、通信と処理は別々の概念というより、むしろ通信しながら処理を行うというように、より高次の概念と考える。大量のデータを対象として処理するためにはデータを移動させながら処理を行うという考え方が有効と考えられるからである。

図1のコンピュータネットワークの基本概念には広域ネットワークからローカル/インハウスネットワークまで含まれるのと同様、図2の分散データベースシステムの基本概念には広域のものか

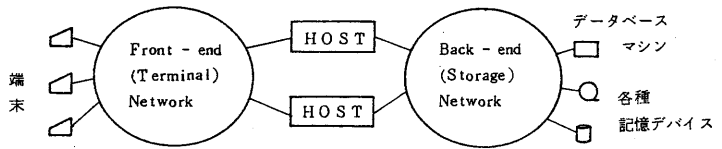


図3 Front-End Network と Back-End Network

ら、Back-End Storage Network [2] (図3) やデータベースマシンまで含まれる。

コンピュータネットワーク技術は図3のFront-End Terminal Networkの観点で発展してきたものといえる。一方、Back-End Storage Networkは今後コンピュータにおける通信の重要課題となる。Back-End Storage Networkの技術的基盤は主としてローカルネットワーク技術である。したがって、バス結合や線状結合、ループ/リング結合などの技術を中心としたが、マイクロウェーブ、衛星通信などによる遠距離リンク結合も含まれる。Back-End Storage Networkの概念は、しかし通信と処理をいまだ別の機能として取扱っているようである。それに対して、ここの分散データベースシステムの場合には通信しながら処理を行うという、より高次の機能を取り入れるものとする。

3. 分散データベースシステムにおける通信処理の参照モデル

図2に示した分散データベースシステムにおける通信処理の基本概念を、より詳細なレイヤ構造で表わすことを考える。

レイヤ構造の表現の基本的な考え方を図4に示す。システムは階層的な構造、すなわちいくつかのレイヤから成る構造をもつと考える。あるレイヤはレイヤ内の通信処理ファシリティを利用してそのレイヤ内における種々の機能を實現する。そしてこれらのレイヤ内機能と直下のレイヤから提供されるサービス機能を用いて、直上のレイヤにサービス機能を提供する。図4の(n)-レイヤにおける(n)-機能は(n)-通信処理ファシリティを利用して實現

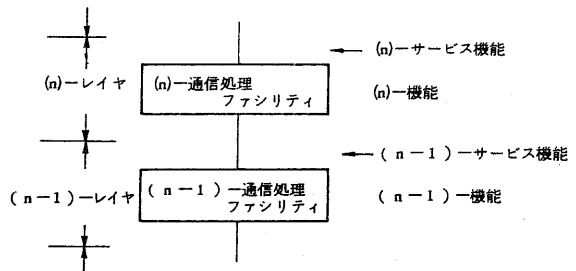


図4 レイヤ構造

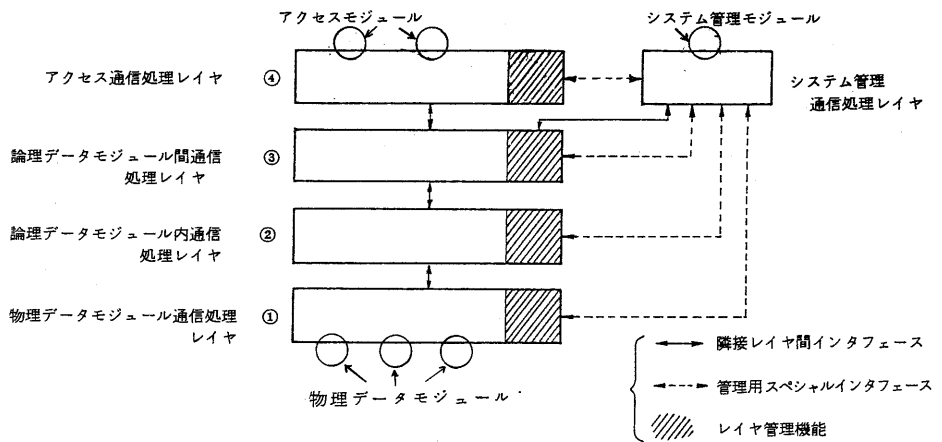


図5 分散データベースシステムにおける通信処理の参照モデル

される。(n)-サービス機能は(n)-機能と(n-1)-サービス機能を用いて実現される。

以上のようなレイヤ構造の考え方をみると、分散データベースシステムにおける通信処理は図5のようなモデルで表わすことができる。システムは4つのレイヤ、すなわち、下から順に、物理データモジュール通信処理レイヤ、論理データモジュール内通信処理レイヤ、論理データモジュール間通信処理レイヤ、およびアクセス通信処理レイヤから成っている。ここではこのモデルを参照モデルと呼ぶことにする。

この参照モデルにおいて、レイヤ④、すなわちアクセス通信処理レイヤではあるアクセスモジュール（言換えればあるホスト内のプロセッサあるいはある端末ユーザ）が発生したアクセス要求について、そのアクセスモジュールとその要求を実際に処理するホスト（即ちプロセッサ）の間における通信処理が行われる。このレイヤはいわば図3に示す Front-End Terminal Network の機能に相当する。そして、アクセス要求を実際に処理するホスト内のプロセッサはレイヤ③のサービス機能を用いて、そのアクセス要求の解決にあたる。

レイヤ③は論理データモジュール間通信処理レイヤである。ここに論理データモジュールとは論理的に一つのまとまったデータモジュールを指している。これに対し、物理データモジュールは一つの物理記憶媒体を指している。一つの論理データモジュールは必ずしも一つの物理データモジュール内に格納されているとは限らず、いくつもの物理データモジュールに分散して格納されていることもある。

レイヤ③はレイヤ④から依頼されたアクセスがいかなる論理データモジュールに関係するかを判断し、またこれらの論理データモジュールで内部処理を受け取ったデータを相互にどのように関連（すなわち通信処理）させるかを判断し、それを実行する。各々の論理データモジュール内での処理はレイヤ②のサービス機能を利用して行う。いかなる論理データモジュールが関係するかを判断するための参照情報（ディレクトリ）は特別の論理データモジュールに格納してあると考へてよい。図3の Back-End Storage Network の機能はレイヤ③の二つの例に相当する。

レイヤ③, すなわち 論理データモジュール内通信処理レイヤでは, レイヤ②から依頼された論理データモジュール内処理要求がいかなる物理データモジュールに関係するかを判断し, またそれらの物理データモジュールで処理を受け取ったデータを相互にどのように関連(すなわち通信処理)させるかを判断し, 実行する。各々の物理データモジュールでのデータ処理はレイヤ①のカーヒス機能を利用する。

レイヤ④, すなわち 物理データモジュール通信処理レイヤでは, レイヤ②から依頼された物理データモジュールについての処理を行う。処理には複数の物理記憶媒体が関与することもありうる。

図5に示すように, 各レイヤはそれぞれ自分自身を管理する レイヤ管理機能をもつ。レイヤ管理機能にはそのレイヤの初期化・エラー回復機能, データディクショナリ/ディレクトリ情報管理機能, 通信処理量の最適化機能, データベースの完全性(integrity)を保つための制御機能, さらに安全性(security)を保つための機能などが含まれる。

レイヤ④相当の位置に, 図5に示すように, システム管理通信処理レイヤが存在する。システム管理モジュールは一般のアクセスモジュールとは別格であり, システム全体の管理制御をこのレイヤを通じて行う。このレイヤはレイヤ①~④のレイヤ管理機能部と直接する特別なインタフェースをもち, 各レイヤの初期化・エラー回復をはじめとする管理にあたる。

4. 参照モデルにおけるアクセス処理の例

この参照モデルにおけるアクセスの処理の様子を例で示す。今, あるアクセスモジュールでユーザ応用を処理中に分散データベースアクセス $f(x, y, z)$ を行う必要性が生じたとする。ここに, x, y, z はこの分散データベース内の論理データで, f はそれらに対するオペレーション(参照のみならず更新でもよい)を示す。このアクセス要求はレイヤ④における通信処理機能により, それを実際に処理するホストのプロセッサに伝えられる。そして, そのプロセスではレイヤ③のカーヒス機能を用いてアクセス $f(x, y, z)$ を処理する(図6, 図7)。

レイヤ③, すなわち 論理データモジュール間通信処理レイヤでは, まずデータ x, y, z がどの論理データモジュールに属するものか判断される。ここでは,

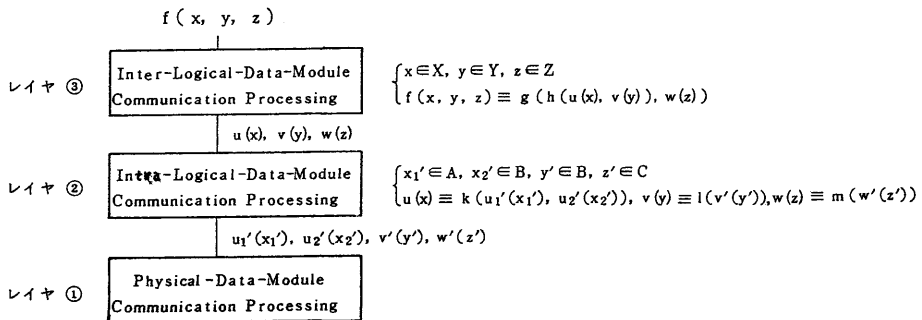


図6 参照モデルにおけるアクセス $f(x, y, z)$ の処理

x, y, z はそれぞれ論理データモジュール X, Y, Z に属しているものとして
 いる。そして、 $f(x, y, z)$ を、レイヤ②のサービス機能を利用して得ることが
 できる $u(x), v(y), w(z)$ に対する論理データモジュール間オペレーション
 $g(h(u(x), v(y)), w(z))$ の形に分析して処理する。

レイヤ②, すなわち論理データモジュール内通信処理レイヤでは、 $u(x), v(y)$
 あるいは $w(z)$ をそれぞれ X, Y あるいは Z の論理データモジュール内のオペレ
 ーションと理解して処理を行う。論理データモジュール A および
 B 内にそれぞれ x_1', x_2' として存在し、 y, z はそれぞれ B, C 内に y', z' とし
 て存在するとする。 $u(x)$ はレイヤ①のサービス機能を利用して得ることができ
 る $u_1'(x_1'), u_2'(x_2')$ に対する論理データモジュール X 内のオペレーション
 $k(u_1'(x_1'), u_2'(x_2'))$ の形に分析して処理する。 $v(y), w(z)$ もそれぞれ論理デ
 ータモジュール Y, Z 内のオペレーション $l(v(y'))$ 、 $m(w(z'))$ で処理される。

レイヤ①, すなわち物理データモジュール通信処理レイヤでは、物理データモ
 ジュール A に対応して x_1', x_2' 、 B に対応して y', z' 、 C に対応して z' の物理デー
 タ処理を行い、 $u_1'(x_1'), u_2'(x_2'), v(y'), w(z')$ の要求に答える。

5. 分散データベースシステムの構造と参照モデルとの対応

この参照モデルが分散データベースシステムの構造とどのように対応するかを
 みるために、一つの例を示す(図7)。

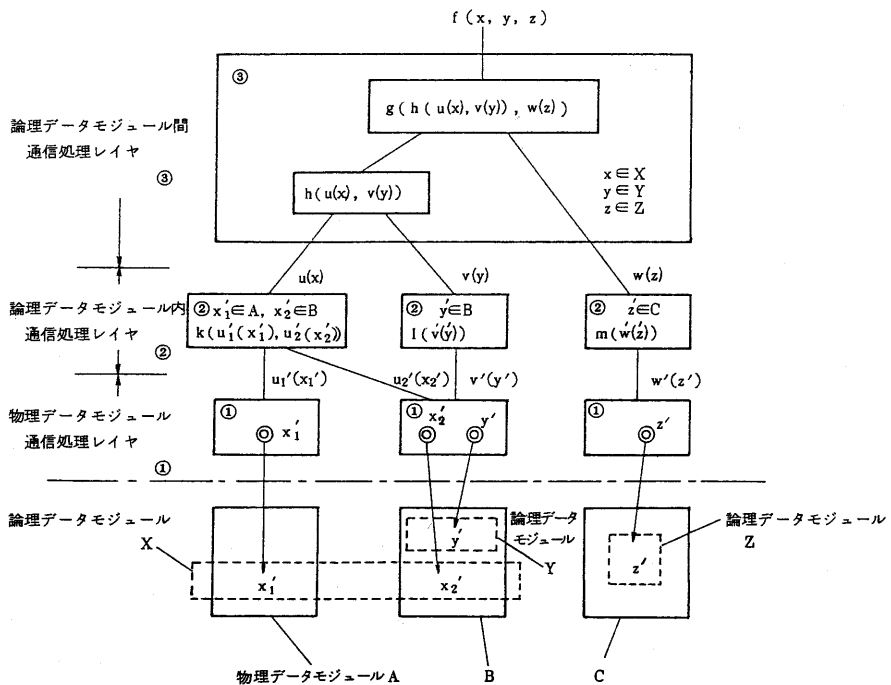


図7 参照モデルにおける $f(x, y, z)$ の処理の詳細

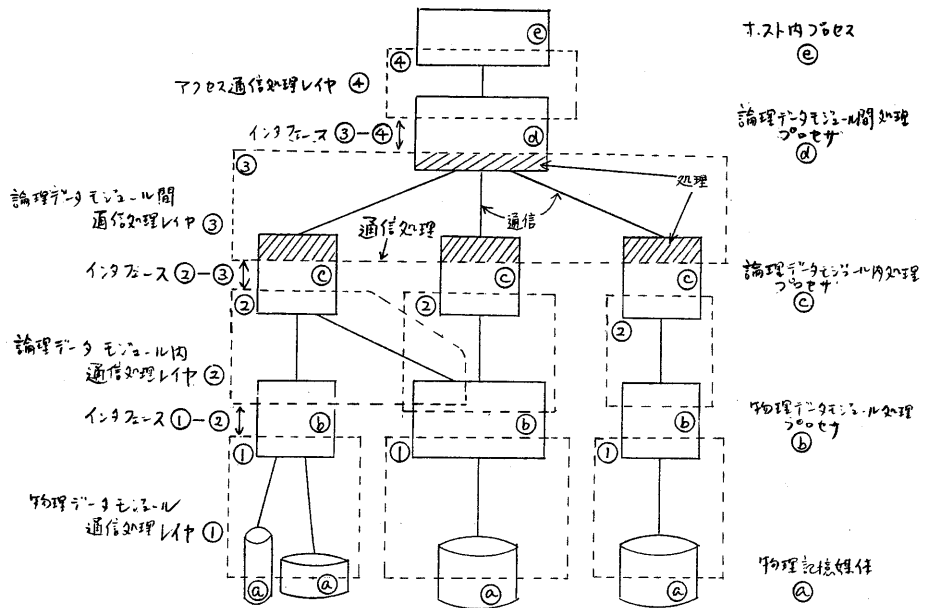


図8 分散データベースシステムの構造と参照モデルの対応

図8の分散データベースシステムには、いくつかの物理記憶媒体 (a) があり、これらのそれぞれを管理・制御する物理データモジュール処理プロセッサ (b) がある。一つの物理データモジュール処理プロセッサが複数の物理記憶媒体の管理・制御にあたる場合もある。一つの論理データモジュールに対応して一つの論理データモジュール内処理プロセッサ (c) がある。 (c) で扱う論理データを物理データとして物理記憶媒体に格納するには (b) の働きを要する。論理データモジュール内の論理データが複数の物理データモジュールに分散して格納されることもあるので、一つの (c) が複数の (b) と関係する場合もある。いくつかの論理データモジュールをまたがるアクセス要求を処理するために論理データモジュール間処理プロセッサ (d) がある。そして、アクセス要求を出すアクセスモジュールとしてホスト内のプロセス (e) がある。図8においてこれら (a) ~ (e) は実線で示されている。

参照モデルの通信処理レイヤは図8において点線で示されている。すなわち、(a) ~ (b) の間に物理データモジュール通信レイヤ (1)、(b) ~ (c) の間に論理データモジュール内通信処理レイヤ (2)、(c) ~ (d) の間に論理データモジュール間通信処理レイヤ (3)、そして (d) ~ (e) の間にアクセス通信処理レイヤ (4) がある。例えば、通信処理レイヤ (3) では斜線で示した4つの部分がこれらを紹介する通信線を介し、互いに協同処理を行うことにより、このレイヤで受けもつべき通信処理を遂行する。通信線はトリー状に示してあるがこれは単に例であって、これら4つの部分が網状その他の通信線形態で結ばれていてもよい。メッセージ・データの交換の様式についても、これら4つの部分の相互におよび 1:1 の対向通信、あるいは 1:n の放送型通信など種々ありうる。これらの形態・様式はこのレイヤにおける通信処理プロセスの一環とみなされる。

表1 参照モデルと既存の概念/システムの対応

レイヤ番号	通信処理レイヤ	概念/システムの例
④	アクセス	Front-End Terminal Network
③	論理データモジュール間	Back-End Storage Network
③	論理データモジュール内	EDC (ハード), DIRECT (ソフトウェア)
①	物理データモジュール	Staging, RAP (回転記憶+論理セル), EDC (回転記憶+論理セル), DBC (構造記憶+大容量), DIRECT (ソフトウェア)

6. 考察

ここで提案した分散データベースシステムの参照モデルは、物理データモジュール、論理データモジュールの地理的存在位置に依存していない。すなわち遠距離通信によるものも対象としている。

データベースマシンは大量のデータを複数の物理データモジュール内に分割格納しており、これから分割格納されたデータの（アクセス要求内容に対応した）総合処理を行うことにより、データベースアクセスに応答する。したがってデータベースマシンについてもここで示した分散データベースシステムの参照モデルで表現することかできる。

表1は広域分散データベースシステムからデータベースマシンにいたるシステム構成について、既に提案されている概念や製作されたシステムの例を参照モデルのレイヤとの対応で示したものがあつた。レイヤ④のアクセス通信処理レイヤは図3のFront-End-Terminal-Networkの概念に対応し、いわゆるコンピュータネットワーク技術によってこのレイヤの技術は熟成の域に達している。レイヤ③の論理データモジュール間通信処理レイヤに対応する概念例として図3のBack-End-Storage-Networkの概念がある。2.で述べたように今後検討されるべき重要な課題である。

レイヤ③は、論理データモジュール内における通信処理を対象としているが、それら論理データモジュールの存在位置が互いに遠い場合、近い場合いづれにあつても、通信処理量（通信量と処理量の2つを決めた総合的指標）を適当な通信処理時間（応答時間）のもとで最小化する技法の出現が待たれている。それのためにどのような処理機構が望ましいかを考えるにあつては、通信と処理を別個のものとして考えず、それらを有機的に統合した通信処理の概念でアプローチする必要がある。Bernstein [3] の半結合 (semi-join) の技法、Hewner と Yao [4] の Query Processing の技法（各論理データモジュールにおけるデータの出現割合をもとにモジュールごとの処理とモジュール間の処理順序をスケジューリング統合通信量や応答時間の最小化をはかるといった）はこのレイヤにおける通信処理技法として位置づけられる。

データベースマシンにおける通信処理技法については、現在までのところこのデータベースマシンはレイヤ①の物理データモジュール通信処理の段階を対象と

してあり、レイヤ③の論理データモジュール内通信処理の段階が研究の対象となり始めたといふところにある。EDC [5] のバンドの概念は複数の物理データモジュール（磁気バブル）をまたがる形で一つの論理データを格納するためのものであり（図7における論理データモジュールXに相当する）、論理データは、スリットに分割して格納される。EDCではバンドごとのリレーションが表わされ、そのリレーションを構成しているタプルのいくつかが一つの磁気バブルに格納される。Restriction, Projection オペレーションは各物理データモジュール通信処理として、RAFと同様な回転記憶（磁気バブル）と論理セル方式によって並列に遂行されている。Join オペレーションは論理データモジュール内通信処理として複数の物理データモジュールにわたる処理が行われる。あるリレーションのタプルに関する情報が複数の物理モジュールに放送型で送られ、それぞれにおいてJoinすべき別のリレーションのタプルが探され（Restrictionとして）、その結果がこれら物理データモジュールから集められる。

データベースマシンにおけるレイヤ③の論理データモジュール間通信処理については、いわゆるオフラインオートメーションシステムにおける分散データベースのためのマシンの間接と相俟って、そのための技法が発展すると思われる。

以上、分散データベースシステムを通信処理の観点から整理し、一つのモデルを提案した。分散データベースはその構成要素データベースのデータモデル/スキーマを全体として統合しなされるが、その観点からの論理モデル、すなわち統合データベースモデル/スキーマモデルとして、滝沢・浜中 [6] の「4層ステップ構造」がある。このような論理モデルをここで提案した通信処理モデルの2つを統合することによってさらに進めたモデルを構成することが期待できよう。

7. あわりに

ここで提案した分散データベースシステムの通信処理モデルは、日本情報処理開発協会の滝沢 誠凡との共同研究の過程で得られたものであり、凡に感謝申し上げます。またこの研究は一部文部省科学研究費により行われつつありますが、その研究が一つにおいて指導賜わります京都大学情報処理教育センター長大野 豊教授に感謝申し上げます。

参考文献

- [1] 猪瀬, 苗村, 田畑, 浅野, "コンピュータネットワーク技術", 情報処理学会 (1980)
- [2] "Back-End Storage Network", Computer, IEEE Computer Society (Feb., 1980)
- [3] P. A. Bernstein and N. Goodman, "Full Reducers for Relational Queries Using Multi-Attribute Semi-Joins", Proc. of the IEEE Computer Networking Symposium, Gaithersburg, Maryland (Dec., 1979)
- [4] A. R. Hewner and S. B. Yao, "Query Processing in Distributed Database Systems", IEEE Transaction on Software Engineering, Vol. SE-5, No. 3 (1979)
- [5] 植村, 前川, "データベースマシン", 情報処理学会 (1980)
- [6] M. Takizawa and E. Hamanaka, "The Four-Scheme Structure Concept on the Gross Architecture of Distributed Database and Heterogeneity Problems," Journal of Information Processing, IPSJ, Vol. 2, No. 3 (1979)