

**解 説**

**1. 分散処理アーキテクチャ†**



田 中 英 彦††

**1. ま え が き**

今日のコンピュータシステムは、その多くが単独で存在せず、通信接続装置を経由してなんらかの形で他のコンピュータシステムや端末と接続されている。工場内の制御用コンピュータ、銀行など金融業で使われているコンピュータ、オフィス内のコンピュータ、研究所で使われているコンピュータなど、ほとんどの応用分野において今や分散処理は当たり前のことになってきている。分散処理であるとわざわざことわるまでもなく、情報処理システムは分散化されてきているのが現状であろう。しかし、それらのシステムにおける分散処理の目的は LAN におけるファイルサーバ、プリントサーバ、ゲートウェイサーバなどの共有や、広域網でのデータ収集・分配、工場内のデータ収集・制御、遠隔データベースへのアクセスなどがほとんどで単純な利用形態であり、分散処理の機能レベルはそう高くない。また、網接続の規模も単一組織内に閉じたものが多い。しかし、分散の奥は広く、複数のコンピュータ資源を利用者からの要求に応じて組み合わせたり、個別網を接続して大規模網とすることによって利用情報の範囲を著しく拡大したり、さまざまなメディアを組み合わせた情報通信システムと情報処理システムとの融合利用を計ることなどは、一部始められているとはいえ、まだまだ今後の作業である。分散処理を取り巻く環境としては、コンピュータ、特に強力なパーソナルコンピュータの低価格化、LAN 技術の発達、OSI の参照モデルや諸プロトコルの標準化などが進展し、今や分散処理は新たな局面を迎えたといえよう。情報処理の中心は、もはや単独のメインフレームコンピュータよりも、むしろ、強力なマイクロプロセッサによるワークステーションと、いくつかのメインフレーム、及びこれらを結合する通信装置による分散

システムに移りつつある。すなわち、われわれは、ある目的を遂行するコンピュータシステムを設計する場合、コンピュータハードウェア、オペレーティングシステム、応用プログラムの設計のほかに、分散処理設計という新たな作業が必要である。

本文ではまず、分散処理を巡る動きを概観し、次に分散処理技術の概要を述べる。分散処理の集中処理と異なる大きなポイントは、システム全体の状態を時々刻々把握している視点の存在を仮定できないところであり、分散処理特有の制御技術を生んでいる。これらの概要に触れた後、次いで、これらの技術を組み合わせ、応用を実現するための基本となる分散処理アーキテクチャについて述べ、最後に今後の展開と課題について述べる。

ところで、分散処理システムといえば広義には、上述したネットワーク分散のほかに、単一のコンピュータを複数のプロセッサで実現するコンピュータ複合体も含まれるが、本特集ではそれを含まない。しかし、最近はこれらの中に位置するシステムも現れているし、また基本となる技術には共通するものが多く、実質上は含まれていると考えられよう。

**2. 分散処理システムの動き**

**2.1 分散処理を巡る技術の歩み**

分散処理技術の源は、単一の計算機システムを時分割で使用するタイムシェアリングシステムにまで逆のぼることができよう。データ通信の発達にともなって、計算機のすぐ近くでそれを使うことから、遠隔の端末から計算機を共同利用できるようになった。これはその後、1970 年代初頭のコンピュータネットワークの発達につながり、複数の計算機それぞれの中で動いているプログラム間の通信概念、及び仮想端末の考え方を生んだ。初期のコンピュータネットワークは、メール、リモートログイン、及びファイル転送などが主要な機能で、凝った使い方は研究はされたが実用されるには至っていない。地理的に離れた複数の計算機

† Distributed Processing Architecture by Hidehiko TANAKA (Department of Electrical Engineering, The University of Tokyo).

†† 東京大学工学部電気工学科

を通信回線で結び合わせ、特定の目的に応じたシステムを実現するという、いわゆる分散処理システムはこのコンピュータネットワークの発達とともに多くのシステムが作られ、負荷分散、機能分散といった区分けが行われるようになった。たとえば、端末に少し情報処理機能を持たせて、端末表示のフォーマティングを行ったり、端末からの入力データを少し貯め込み誤りが無いことを確認したり、適当な形式に編集してから中央に送出するなどはこの時代の代表的な使用形態である。コンピュータネットワークの発達はネットワークアーキテクチャという概念を生み、1970年代後半には多くのネットワークアーキテクチャが存在するところとなり、それらの間の接続を容易なものとするためにその標準化が開始された。いわゆる開放型システム間相互接続(OSI)の技術である。現在、基本参照モデルのほか、第5層までのサービスとプロトコルが国際標準になっており、残りの2層(プレゼンテーション層と応用層)の主要サービスとプロトコルも国際標準案の状態にある。コンピュータは通信回線などによって相互接続されることが普通になってくると、OS内に占める通信機能の重要性が増し、データベース機能とともにその役割を拡大してきている。Network OS(NOS)という用語は、そうした傾向を示す言葉である。機器構成から見ると、これらの機能拡大は、専用装置を生むきっかけとなり、通信制御プロセッサ、データベースマシンなど、CPUから機能を分散させる機能分散型プロセッサが作られるようになった。

一方、1980年代に入ると、地理的に近い領域内のコンピュータなどのシステムを高速の通信路で結ぶLAN技術が発達するところとなり、コンピュータシステムのイメージを変え始めている。すなわち、パーソナルコンピュータを数多くLANに接続し、LAN上にはファイルサーバ、プリントサーバ、通信サーバなどいくつかの専用装置を接続して各パーソナルコンピュータから共同利用するという形態である。すなわち、従来のコンピュータを機能によって分割しLANによって総合システムにまとめ上げるという考え方であり、コンピュータとは集中した機器群であるという従来のイメージを大きく変えつつある。特に、最近のVLSIの発達は32ビットマイクロプロセッサや大容量のメモリLSIなどによる強力なパーソナルコンピュータやワークステーションを可能とし、この傾向に拍車をかけている。今やマイクロプロセッサによる情報処理は、大型汎用機のそれよりもコスト性能比が著しく高

く、また処理性能の絶対値もかなり高いレベルに至っている。しかし、パーソナルコンピュータ単独でかなりのことができるとはいえ、そのようなスタンドアロン分散だけではやはり限界があり、大型機の有効な分野があることも事実であろう。大規模データベース、大型計算処理などがそうであるほか、通信機能によって単独の世界からほかの多くのシステムとの情報交換を可能とすることは、情報処理の著しい機能向上を意味する。マイクロインフレームリンクの発達、いくつかの大学における強力なワークステーションを用いた新しいコンピュータファシリティの試みなどはこの線上にある。分散OSは、この時代のOSを代表する言葉であろう。

## 2.2 分散処理の目的の変遷

分散処理において直接、分散の対象になるものは、業務ないしは特定の機能、処理負荷、データ、ユーザアクセスなどであり、これらの分散により管理や制御、また、運用保守などが分散されることになる。分散処理の目的として考えられるものにはさまざまなものがあるが、その時代の技術的背景によって少しずつ強調のされ方に差がある。

まず通信コストの低減がある。データの発生場所で誤りを正し、また生データに加工を加えて遠隔コンピュータで必要とする形にして送ることによりデータ量が著しく減少することを利用するものである。次にシステムの信頼性向上があげられる。一般に集中システムでは中央のコンピュータが故障すればすべてのサービスが中断するが、分散システムでは一部の障害に止まるのでシステムがフェイルソフトになる。また、代替システムが存在するのが普通なのでその意味でも信頼性が向上しうる。次に、企業内システムのように、その企業組織形態に合わせた分散処理システムを構築することが容易で、それによって利用や管理の親和性が向上する利点があげられる。1970年代の後半でよくいわれた水平分散や垂直分散はその辺りの話である。対等システム間は水平分散という形を取るのに対し、システム間で従属関係がある場合はそれに合わせた垂直分散となる。これと少し似たポイントとして、利用者に対するサービス性の向上がある。分散システムでは手許近くのコンピュータが端末との対話を直接制御するので、集中システムよりも一般に応答性がよくなる。さらに、手許での処理をユーザの要求に合わせて細かくチューニングすることははるかにやりやすくなる。集中システムでは1カ所に集まるソフトウェア量

が膨大となりその保守が難しくなることは容易に想像できよう。大規模ソフトウェアから、小規模多種ソフトウェアへの動きであり、また多様性の増加という今後の社会の動きにマッチした自然な流れであるとみることでもできよう。

次に、システムの拡張性向上というポイントがある。一般にシステムは、その規模や機能が年とともに拡張するものであり、要求に合わせて少しずつ種々の構成要素を強化できることが望ましい。分散システムではそれが行いやすく、システムの成長がスマートである。以上のほか、システムのコスト性能比が重要なことは当然であろう。あらゆる項目をこれらに帰着させてシステムを設計するのが普通であるから、これを今さら持ち出すことは無意味かも知れない。しかし、分散処理の場合は、マイクロプロセッサやメモリなどの VLSI 技術さらに LAN 技術によるハードウェアコストの低下が著しく、直接的にシステムのコスト低下につながりやすいという要因がある。もはや Gosh の法則が成立しなくなって久しいのである。

しかし、これらの利点は単に分散処理システムを構成しさえすれば得られるわけではない。このような効果を上げるためにさまざまな工夫が必要である。通信コスト低減にはそのような処理分割の工夫が、信頼性向上には代替えが可能となるような構成の工夫が、組織体との整合性向上には各要素システムの機能明確化と他要素システム間のインタフェースの明確化の工夫が、利用者に対するサービス性向上には各所におけるシステムチューニングの努力が、そして拡張性を維持するにはもともとのシステム間インタフェースの陳腐化に強い汎用のものとしておく工夫が、経済性のためにはニーズに合った負荷分散/機能分散の工夫が、それぞれに必要なのである。要するに分散処理システムとは今や、システムの設計をする上での自由度が一つ増えたに過ぎない。システム全体としての複雑さは増しているが、インタフェースを明確にすることによって構成要素のモジュール性を高め、仮想化や階層構造の技術を用いることによって、その複雑さから逃れる技法が肝要である。それによって利用者には、分散性の意識、重複データの存在、時おり生じるシステム障害、データの無矛盾性維持などのめんどろな事柄を考えないで済ませる“透明性”を提供する必要があると言えよう。

### 3. 分散処理技術

#### 3.1 基本技術

分散処理の基本技術として次に述べる6つを考える。これらは集中処理においても必要なものであるが、分散処理において特にその特徴が現れる。

##### (1) プロセス間通信

物理的に離れているプロセス間の通信なので共有メモリは使えず、メッセージパッシングを用いることになる。通信プリミティブとしては SEND/RECEIVE があり、さらに同期式と非同期式に分けられる。同期式とは SEND を出した後、その返答が戻るまで待つものをいい、待たずに先に進むのが非同期式である。プログラムとの親和性から、手続き呼び出しの一種(遠隔手続き呼び出し)としてプロセス間通信を実装するものが多く、これは同期式である。

##### (2) 名前とアドレス

アクセスしたいものを名前で指定して、その物理的位置を示すアドレスとのマッピングを行う必要がある。一般に名前は、同じ対象に対して複数の名前が存在することを許す必要があるし、また逆に、一つの名前でもその使われ方に応じて異なる対象を指すことがある。さらに、名前の登録や削除は時々刻々行われるし、分散システムのようにその規模がオープンエンドなシステムでは、システム全体で用いられる名前の数は膨大になる。したがって、全名前を1カ所で管理することには限界があり、全体を適当な範囲に区切ってそれぞれの中で対応表を管理するなどの工夫が要る。また、分散トランザクションなどの応用ではシステム全体にわたってユニーク性の保証された識別名を必要とすることがあり、限られたビット数で拡張性を残しつつその生成を行う手法が必要である。

##### (3) 同時性制御

複数の利用者が共用のデータベースにアクセスする場合、それぞれの処理の順序付けをうまく行わないと矛盾が生じることがある。すなわち、データの更新をともしなう場合で、このような矛盾が生じないように同期をうまく取る手法が同時性制御である。これには、共有資源に対する使用権の確保を系統立てて行う二相ロック方式や、各アクセスや資源それ自体に、仮想的な時刻(タイムスタンプ)を付けその大小関係で矛盾を防ぐタイムスタンプ方式などがある。

##### (4) コミットメント制御

分散処理では、いくつかの部分的な処理を網内で並

行的に進め、それらすべての完了をもって一つの仕事の区切りとすることがある。このようなとき、すべてが無事に完了すればよし、もし一部が不成功であれば残りのすべてを取り消して、all-or-nothingの制御を行わねばならない。分散データベースで複数のコピーを同時更新するような場合がそうで、これをコミットメント制御という。これはまた、複雑になりがちな処理を簡潔に見せる働きがあり、処理のモジュール化に役立つ。

#### (5) 障害回復手法

複数プロセスが相互にインタラクトし合っている状況でなんらかの障害が生じたとき、それまでの処理を適当に取り消して適切な位置から再開するための手法が必要である。これに重要な概念は、アトミックアクションである。また、処理の取り消しをする場合、プロセスが通信し合っていることから、一つの処理の取り消しが他の処理の取り消しを生み、さらにそれがまたほかの取り消しを生むという形で、バックアウトが波及することがある。このようなドミノ効果の影響を軽減する工夫も必要である。

#### (6) 安全性

遠隔コンピュータからのログイン(遠隔ログイン)を可能とするには、そのユーザの認証が必要である。認証は一般に集中化するのが望ましいが、分散処理では各システムの自律性を尊重する必要もあり、これらの適切なバランスを取らねばならない。また、分散システムではケーバリティやパスワードなどをシステム間で転送する必要も生じるので、今後の分散システムに暗号化の機能は重要になるであろう。

#### (7) デッドロック検出/回避

共有資源の割当てを行うと一般にデッドロックが問題となる。これは集中システムでも同じであるが、分散システムでは資源の要求一割当ての関係が広くシステム間にまたがり、その検出が一般に難しくなる。したがって、Transaction-Wait-For グラフを階層的に作って、そのループ検出を容易にするとか、待ちにタイムアウトを導入するとか、資源に適当な全順序を付けるなどの工夫が必要になる。

### 3.2 通信技術

次に、システムを結びつけるための通信技術を以下の4つに分けて概観する。

#### (1) データ伝送・交換

OSIの7層モデルのうち下位4層に関する技術である。すでに標準化がなされているものが多く、特に

下位1,2層では広域網・LANそれぞれに対し多くの標準がある。これらの多様性を吸収し、エンドエンド間でのすっきりとしたインタフェースを提供するのがネットワーク層及びトランスポート層であり、これらの標準化もほぼ山を越したところである。

#### (2) ネットワークアーキテクチャ

OSI基本参照モデルの標準化は完了し、上位3層のサービスとプロトコルの標準化もほぼ目途が付いている。今後の重要な仕事は、応用層の内部構造及びローカルシステムとの関係を明確化すること、ならびにコネクションレス型転送の標準化、OSI管理の標準化である。応用層内には多くのプロトコルに対応して多くのサービス要素があり、応用プログラムからはそれらを組み合わせて使う必要がある。したがってそれに対応して、各サービス要素間の関連を付けたり、ローカルシステムとのインタフェースを取る構造が必要であるが、いまだその辺りは固まっていない。また、従来のOSIは通信機能のみを問題とし、ローカルOSや応用プログラムとの関係があまり陽に扱われてこなかったが、分散処理という立場からみると、通信内部の問題よりもそれとのインタフェースが問題であろう。

#### (3) 応用プロトコル

現在、応用プロトコルとして開発されているものは、ファイル転送アクセス及び管理、仮想端末、ジョブ転送と操作、メッセージ向きテキスト交換システムである。その他、遠隔データアクセス、トランザクションなどが検討され始めている。

#### (4) 諸メディア

テレテックス、ビデオテックス、ファクシミリ、電話など、いわゆるテレマティーク系の諸システムがあるが、オフィスなどの仕事は、音声・画像・文書・データを駆使して行うものであるから、情報処理とこれらとの融合も今後は必要であろう。

### 3.3 システム技術

要素技術を組み合わせ、分散処理システムを実現するためのシステム技術を以下にまとめる。

#### (1) 言語と利用者インタフェース

分散処理システムの制御システムや応用それ自体を記述する言語としては、通常の汎用言語を用いて各システム内のプログラムを作成し、手続き呼び出しや入出力文を用いて網経由で結合するのが通例であるが、分散した応用全体を記述する言語の検討もある。データ型とその値とを対にして送るとか、各システム内プログラム記述と、それらの間の結合関係記述を分離す

るなどの工夫がある。また異種コンピュータシステムを結合した分散システムでは、遠隔ログインによってさまざまなシステムが使えるが、そのままではコマンド言語の差異に直接ユーザが対処する必要があり使い難い。このような状況に対処するために1970年代中頃、網の利用インタフェースを変換する装置(NAM, REX, RITAなど)が考えられたこともあったが、今後はOSインタフェースの標準化がその進むべき道であろう。

### (2) サービス

コンピュータシステムを利用者ごとのワークステーションと共用サービス資源(サーバ)とで構成する場合、そのサーバとして用いられているものには、ファイル、プリンタ、プロセス、端末、メール、時刻、ブート、ゲートウェイなどの諸サーバがある。ファイル、プリンタは最も良く使われているもので、プロセスサーバは要求に応じて所定のプロセスを生成するもの、端末サーバは、複数プロセッサのプールがあるようなシステムで動的に各端末に対してプロセッサを割り付ける役目を持つ。メールサーバはシステム内のメールサービスを行い、時刻サーバは時刻情報を生成するもの、ブートサーバは、初期立ち上げや障害からの回復を司どり、ゲートウェイサーバはそのローカルなシステムと外界とを継ぐ接点で、一般には広域網プロトコルとローカルプロトコルとの変換を行うものである。

### (3) 分散データベース

データベースは共用資源として最も重要なものであり、分散データベースを実現する技術は重要である。これには、異種データベースの統合技術、問合せの最適化と変換技術、データディクショナリ・ディレクトリ(DD/D)技術、制御技術、及び分散データベース自体の設計技術などがある。異種性とはデータモデルやアクセス言語などの差異がそうで、共通のモデルなどを設定してそれとの変換を行うことになる。問合せの最適化は、複数のデータベースへのアクセスを組み合わせることでその問合せに答える場合の最適化で、伝送量や処理の少ない問合せ分割を求める手法である。DD/Dは分散データベースの中心をなすメタデータであって、単一データベースにおける通常のDD/D情報のほかに異種性情報、分散情報などを含む。これらを網内でどう配置し使うかが問題となる。制御技術とは重複データに対する更新処理があったり、複数の問合せが入り組んでも内容のコンシステンシを保つとか、障害回復、安全性など、基本技術のところ述べて

た手法のことである。また、分散データベースの設計とは、トップダウン的にデータベースの分割と配置を定めたり、逆にボトムアップ的に既存データベースがあったときの統合法を定めるなどの作業である。

上述技術は一般論であるが、現実的には遠隔データベースアクセスプロトコルを標準化し、各要素データベースの自律性を認めつつ分散データベース機能を順次高めてゆくのが実際のところであろう。

### (4) 分散 OS

従来のOSは単一の集中システムを対象としていたのに対し、ワークステーションと諸サーバの組み合わせでシステムを構成するような場合を対象としたOSである。複数の要素が組み合わさって始めてシステムとして完全なものとなることを想定してはいるが、利用者にはできるだけそのような分散は見せない。分散OSはいまだ開発の初期にある。今後、多くの利用やソフトウェアの積み重ねで磨かれるべき重要な技術である。

### (5) 管理

分散処理は多くの管理すべき対象を持っている。通信系、プロセッサ、ディスクなどの諸ハードウェア資源の管理、その上で走っている分散した応用の管理、さらに、どのようなプログラムやデータがどこにあるかといったソフトウェア・データ資源の管理、利用者のアカウントやUID、パスワードなどの管理などである。これらの管理下で利用者により動的に共用資源を割り付けて応用を走らせるが、前述の通りデッドロック対策が必要である。

## 4. 分散処理アーキテクチャ

本章では分散処理の一般的な構造について考察する。

### 4.1 システム要素

分散処理システムを構成する要素としては次の6つに分けることができよう。

- ① ハードウェア
- ② 制御プログラム
- ③ インタフェースシステム
- ④ データ
- ⑤ 応用プログラム
- ⑥ 人間

①には、プロセッサ、ディスク、入出力装置、LAN・広域網・ゲートウェイ・諸メディアなどの通信系が含まれる。②は各装置内の制御プログラム群で、OS、

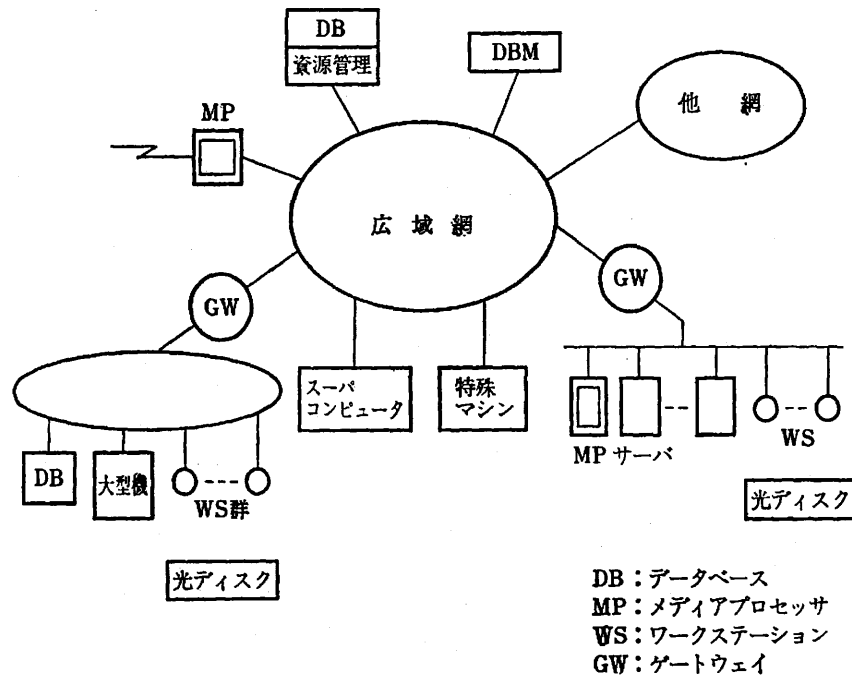


図-1 分散処理システムの物理アーキテクチャ

DBMS, OSI 実現プログラムなどがこれに入る。③は諸言語プロセッサ、ディスプレイ及び対話システムがそうで、今後はここに人工知能の成果を生かした音声認識、自然言語処理などの機能が入ることになる。④はデータベースに貯えられる種々のデータで、DD/D などのメタデータも含むほか、今後はデータの意味記述が強化され、いわゆる知識として多量の情報が蓄積されてゆくことになる。⑤は物理的には④と同じくデータベースに貯えられるものではあるが、今後は再利用が進む結果、プログラムベースとしてその機能内容、外部仕様などの形式化が進展するものと思われる。⑥はシステムの利用者であるが、システムの一構成要素としてとられ、人間が持っている常識、本やマニュアルから得て持っているシステム資源に対する知識などを考慮したシステム設計には不可欠の要素である。

4.2 物理アーキテクチャ

今後の分散処理システムの一般的な物理アーキテクチャとしては図-1 のような形が考えられる。ユーザ

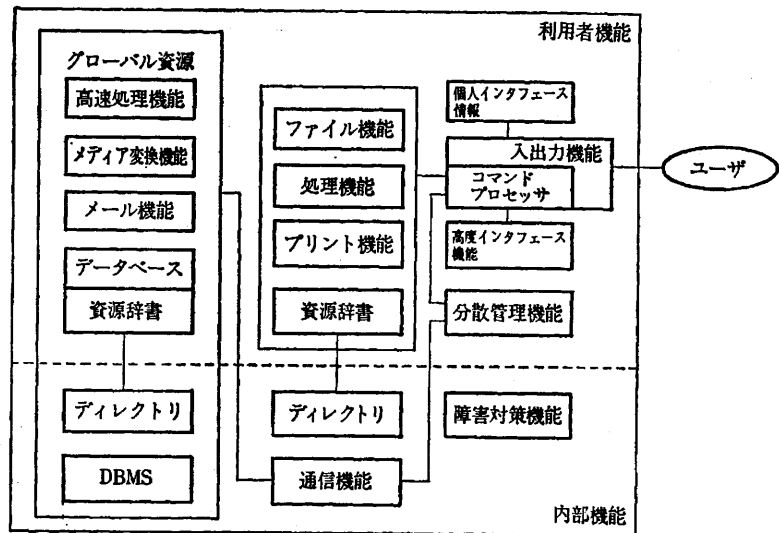


図-2 分散処理システムの論理アーキテクチャ-利用者のビュー-

はワークステーションを通じてシステムを使うが、種々のサーバ群とともに LAN によってローカルシステムを形成する。サーバには前述のようなもののほか、通常の処理機能を提供する汎用機などのプロセッサ群も含まれる。多くの LAN は ISDN のような広域通信系を通して結ばれ、広域網もまたそれ自体ほかの広域網（たとえば他社の網、外国網、他業種網など）と結ばれている。さらに、直接広域網に結ばれている要素装置もあり、種々のデータベースやスーパーコンピ

ュータ、特殊プロセッサなどが共用される。また電話、ビデオテキスト、ファクシミリなどの諸メディアと情報処理の世界との橋渡しをするメディアプロセッサも重要な資源であろう。情報は、これら通信系を通してオンラインで授受されるものほかに、光ディスクや印刷物を通してオフラインで転送されるものも存在する。

4.3 論理的アーキテクチャ  
物理アーキテクチャの上

さまざまなソフトウェアをのせ論理アーキテクチャを形成するが、まず、利用者の観点からシステムの機能に着目したビューを図-2に示す。図中、破線から上が利用者から見える機能で、下はシステム内で重要な機能ではあるが、利用者からは明示的には見えないものである。また、この図では広域網などを通してアクセスする資源をグローバル資源として、ローカル資源と少し区別している。理想的には利用者に対し全く一様に見せたいのであるが、各機能相互間の関係を明示する意味からこのように描いた。この図は、OAなど非定型的な処理を対象とした一般的なシステムを想定しているため、工場内の制御システムや金融システムなど定型的な応用の場合は不要なものを除いて処理の効率化を計ることができる。

次に、このような機能を持ったシステムを実現するための制御システムからみたビューを図-3に示す。ワークステーションなど個人対応のシステム制御が右半分、それがOSIのサービス及びプロトコル機能を通して外部システムとつながっている。外部システム自体、インタフェース機能を除いて個人システムと同種の構造をしており、制御システムから見て識別できるその複数システムがOSIによって結合されている。たとえば、一つめの外部システムは、ユーザが所属するLAN上システム、二つめは広域網システム、三つめは他所のLANシステムまたはほかの広域網システムがこれに相当する。図では通信機能はすべてOSIとしているが、当然非OSIの通信系も存在しその場合はそれとの変換機能が入る。また、複数の資源

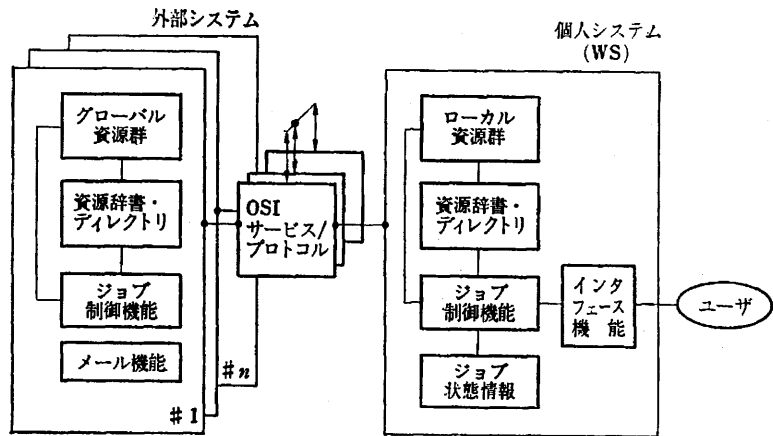


図-3 分散処理システムの論理アーキテクチャシステムマネージャのビュー

を組み合わせた複雑な処理それ自身を資源として登録すれば、その利用者からはその複雑さが見えなくなるので資源の仮想化が可能となりシステムのモジュール化が促進される。

5. むすび

マイクロプロセッサやLANの発達、OSIの標準化など分散処理に必要な要素技術はそろってきた。分散処理の目的はさまざまであるが、今後はユーザに対するサービス性を軸に大きく発展してゆくことであろう。今や分散処理はなにも特殊な技術でなく、通常の技術として定着しつつあるため、一見古い完成された技術のように見えるが現実はそうではない。広く浸透してゆくことによって従来の計算機に対するイメージを大きく変えうる影響力を持っている。スタンドアロンのパソコンも分散処理の一種であるが、単なる孤立システムではなく、なんらかの媒体を通してほかと情報授受したり、必要とあらば通信接続ができるというところで他と密接な関係を保っており、それを可能にすることが重要である。

今後の課題としては、分散データベース、情報資源辞書システム、分散OS、OSIの機能標準などの開発があげられるが、分散処理システムは相互接続によってその含む機能は複雑になる一方である。したがって、システムの複雑さを終端し、利用者には複雑さを見せない形での情報資源辞書システムの開発が今後の重要課題であろう。(昭和61年12月3日受付)