

## 第Ⅶ章 クライアントサーバシステム

## 第Ⅶ章 クライアントサーバシステム

### 学習目標

1. クライアントサーバモデルの基本型（水平および垂直）を理解させること。また、このモデルのコンピュータ環境としての利点と留意点を挙げられるようにすること。
2. 代表的なサーバ機能について、提供されるサービスの概要およびクライアントからのサービス要求→サーバでの要求の処理および結果の返答について一連の流れを図で示せるようにすること。
3. クライアントサーバモデルを実現するために備えるべき環境や機能を概略的に説明できるようにすること。また、クライアントやサーバにどのようなソフトウェア要素が必要となるか大まかに捉えることができるようにすること。
4. 例えば、RPCあるいは同等な機能を使った簡単なプログラムを解読できるようにすること。また、多くのクライアントサーバシステム構築事例に目を通し、このモデルの優れた点を理解させること。

### 全体概要

近年のコンピュータ技術の革新により、汎用コンピュータを中核とした大規模な情報システム作りの終焉を告げようとしている。また、従来の情報システム部門の提供するシステムに飽きたらない企業のユーザ部門が自らコンピュータシステムを業務のやり方に最適な形態で実現しつつある。

これを可能としたのは80年代の分散処理システムに始まり、90年代初期に始まり現在でもますます進展を続けるクライアントサーバコンピューティングの概念が確立されたからである。既に何らかの形でコンピュータに係わりある人を含めれば、従来型の情報システム部門の情報処理技術者の数に比べ、いわゆるコンピュータのエンドユーザ人口の方が圧倒的に多くなってきている。

従来汎用コンピュータで行ってきた業務処理システムのダウンサイジングにますます拍車がかかり、システムのおもりは汎用コンピュータからネットワーク環境下での膨大なパソコン群に対して比重がどんどん移るであろう。高度な専門教育を受けた情報処理技術者でなくともシステムの企画や開発さらには維持が行える環境が整ってきているからである。

しかしながら、情報システムの実績という点ではまだまだ若年であり、今後企業、行政、教育機関などさらに情報化が進むためには、クライアントサーバコンピューティングの市場で、高機能化しながらも一段と使いやすくなるソフトウェアが次々と開発されること、また、この技術を利用する者のコンピュータリテラシーの向上、クライアントサーバに関連するシステムのより深い理解が求められる。

これからの情報システムの優良は、いかに安価に、適正な性能でかつ信頼性の高いクライアントサーバシステムを構築できるかにかかっている。このことから、本章についての深い関心が求められる。

内容のあらまし

節 項	内 容
1. オープンシステム化時代とクライアントサーバシステム	
2. クライアントサーバ処理	
(1) クライアントサーバモデル	クライアントサーバモデルの基本型
(2) ハードウェアとサービス	サービスの実行と実行されるハードウェアとの物理的な関係
(3) クライアントサーバと分散処理	クライアントサーバモデルと分散処理システムとの相違
(4) クライアントサーバシステムの構造	クライアントサーバモデルの構造
(5) 各種サーバ	水平型、垂直型
(6) クライアントサーバによるアプリケーション開発	ファイルサーバ、CPUサーバ
(7) ソフトウェアインタフェース	データベースサーバ、通信サーバ
(8) クライアントサーバシステムの特徴	プリントサーバ、メールサーバ
3. クライアントサーバシステムのための機能	クライアントとサーバの役割分割
(1) 基本環境	Xウィンドウ、SQL、インタフェースの標準化、低価格なシステム構築
(2) サーバ機能	クライアントサーバモデルの利点
4. クライアントサーバシステムのソフトウェア構造	クライアントサーバシステム構築上の留意点
(1) 基本環境	FTP、NFS、RPC、CICS
(2) サーバ機能	コミュニケーション
5. 事例	プリントサーバ、ファイルサーバ
(1) ユーザインタフェースにおけるクライアントサーバシステムの事例	データベースサーバ
(2) オンラインランザクションにおけるクライアントサーバシステムの事例	ユーザインタフェースサーバ
(3) 商用のクライアントサーバ型構成事例	コンピューティングサーバ
6. 主要用語	クライアントサーバシステムのソフトウェア構造
	プロセス間通信、ソケットインタフェース
	アドレスのバインド処理
	クライアントサーバ モデル

## 1. オープンシステム化時代とクライアントサーバシステム

80年代の末頃からオープンシステム化の時代が幕を開け、コンピュータや種々の情報機器がメーカーや機種の違いに依存せず接続されるようになった。折しもクライアントサーバコンピューティングの概念が登場し90年代を代表するコンピュータ処理形態といわれるようになったが、これはオープンシステム化とクライアントサーバコンピューティングとが表裏一体の関係にあることを示している。

例えば、従来のオフィスコンピュータがクライアントサーバによる分散システムへ移行すればオープンシステム化をもたらすであろうし、オープン技術がなければクライアントサーバシステム構築も容易ではなからう。

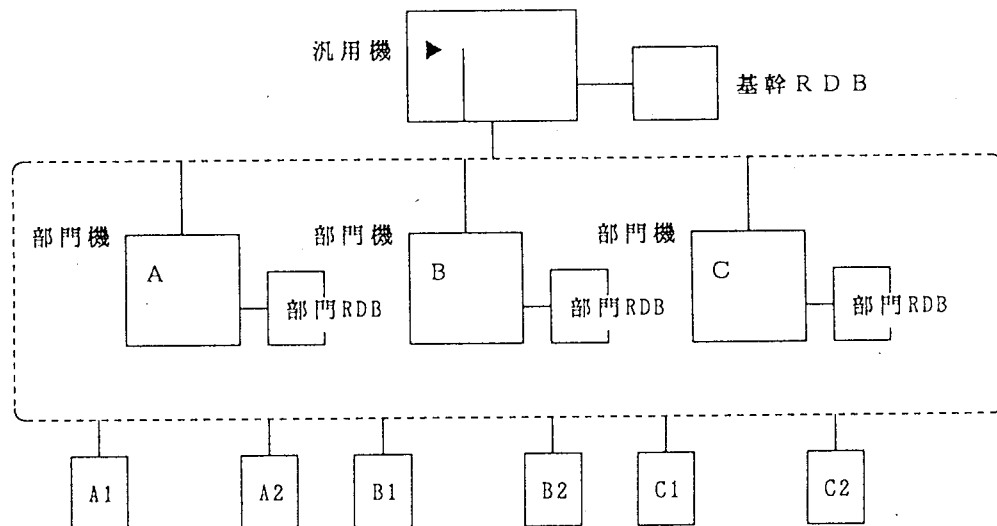
形態的には、WSやPCなど高度な情報処理機能を持つマシンをネットワークで結び、通信によって分散処理を行うモデルである。

クライアントサーバでは、データベースシステムや印刷機能がサーバとなり、ネットワークで結ばれたクライアントとしてのWSやPCから共有の資源としてアクセスができる。サーバマシンは必ずしもWS、PCでなく大型のマシンの場合もあるが、従来のようなホストマシンとしてではなく、“サービス者”としての立場をとるところに特徴がある。

クライアントサーバでは、クライアントがサーバからデータを手元へダウンロードし、画面の生成、データのユーザによる加工に代表されるアプリケーションの実行を行うため、応答が速い。アプリケーションは通常エンドユーザが行うことでEUCが推進される。また、クライアントサーバでは同一メーカーの製品だけでシステムを構成する必要はなく、ユーザに最適なコスト、機能の最も要求に合うものが選定できる。

クライアントサーバシステムとして有効に機能する例としては、リレーショナルデータベースシステムとUNIX、あるいはWindowsとが結合したシステムである。UNIX-RDBMSでの分散型データベース構築がその典型である。

図表Ⅶ-1 典型的なクライアントサーバシステム



基幹RDBと業務部門毎のRDBとが論理的に連携。物理的に自由なデータ配置が可能。例えば、部門AのクライアントA1から部門Aだけでなく部門BやCのデータを検索したり、更新したりすることができる。各部門のクライアントパソコンはミドルウェア（パソコンからサーバのRDBに接続するようなソフト）を用いて市販の表計算ソフト、データベースアプリケーションからサーバ機のRDBを無意識にメニュー操作だけでアクセスし、統計計算、グラフ表示、文書化を簡単にできる。

エンドユーザは、パソコンで自由に基幹あるいは部門データベースを手元にダウンロードし、部門間あるいは全社レベルで情報の共有化を図ることができ、OA業務の効率化、創造的な仕事を可能とできる。

クライアントサーバモデルは、異機種混在でシステムを構成できるため、通信プロトコルの標準化やネットワーク対応基本ソフトが必要となる。プロトコルに関しては、ほとんどのケースでUNIXのTCP/IPが用いられ、汎用機やオフィスコンピュータでもTCP/IPネットワークに対応できている。

例えば、FTP（ファイル転送プログラム）は、ターミナルエミュレーション機能を提供しており、安価なパソコンがUNIX端末として利用できる。また、LAN対応のOSともいえるネットウェアやWindows-NTがLAN上でのパソコンサーバ、時にはUNIXマシンをサーバやクライアントを実現しており、ネットワーク基盤が強化されてきている。

現在注目されているクライアントサーバの適用例は、パソコンだけでクライアントサーバデータベースを構築していることである。これは、クライアントサーバの利点を生かすアプリケーションがどんどん開発されてきていることによるものである。ネットワーク管理ツールにより、分散システムを統合的に一元管理する機能が提供されることで可能となった。また、売上管理、予算管理、給与管理、在庫管理、生産管理、人事管理、顧客管理など企業の基幹OA業務をユーザ自らが安価で簡単に使用できる業務ソフトパッケージが続々と売り出されてきたこともクライアントサーバ化へ拍車を掛けている。

この他、ソフトウェア面でみると、グループでの共同作業を支援するグループウェアはクライアントサーバ環境下で、異種マシンやOS間でつながるといことで普及の速度がめざましく上がっている。また、OAツールの代表として表計算プログラム、ワープロ、電子メール、スケジュール管理ソフトは統合オフィスシステムの実現を可能とし、BPRを可能とし、ビジネスの方法論を根底から変える威力として注目されている。

今後、クライアントサーバモデルによるシステム化は色々な局面で応用されるであろう。基幹情報システムのダウンサイジング、科学技術計算処理系やCAE系、部門コンピューティング、小規模な基幹システムに顕著となりうる。それぞれの作業領域でのデータや情報あるいはノウハウの共有化と、作業員間の時間空間を超えた連携を可能とし、利用者に喜ばれるシステムへとますます発展していくものと期待できる。

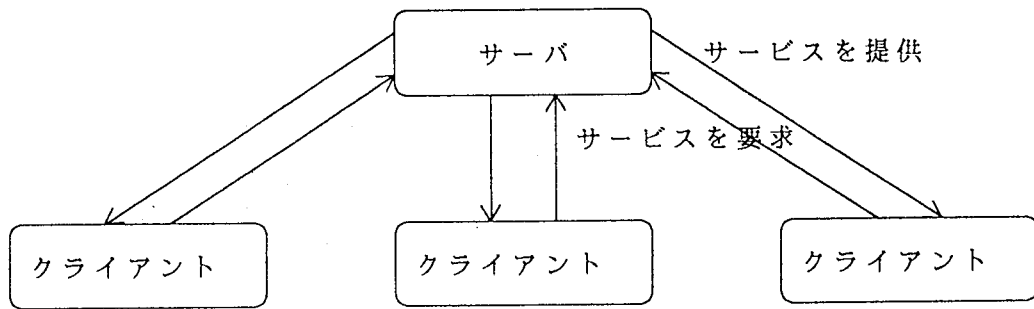
## 2. クライアントサーバ処理

### (1) クライアントサーバモデル

ネットワーク環境下で各ユーザが種々の計算処理が行われる。この計算処理においては各種の処理要求が実行されるが、その要求がユーザの手元のコンピュータで処理されるべきか、特定の専用コンピュータ上で処理される方が好ましいかは利用者にとって処理性能や経済性に大きく影響する問題である。

クライアントサーバモデル（図表Ⅶ-2）は、ネットワーク上のどこで要求が出されその処理がどこでなされるかという観点で、実行効率がよくしかも経済的なコンピューティング環境を実現するシステムとして90年代初期に登場した。このモデルでは要求を出す側をクライアント、それを処理する側をサーバと呼ぶ。このモデルでは、複数のユーザ（クライアント）がサーバを共有することになる。

図表VII-2 クライアントサーバモデル



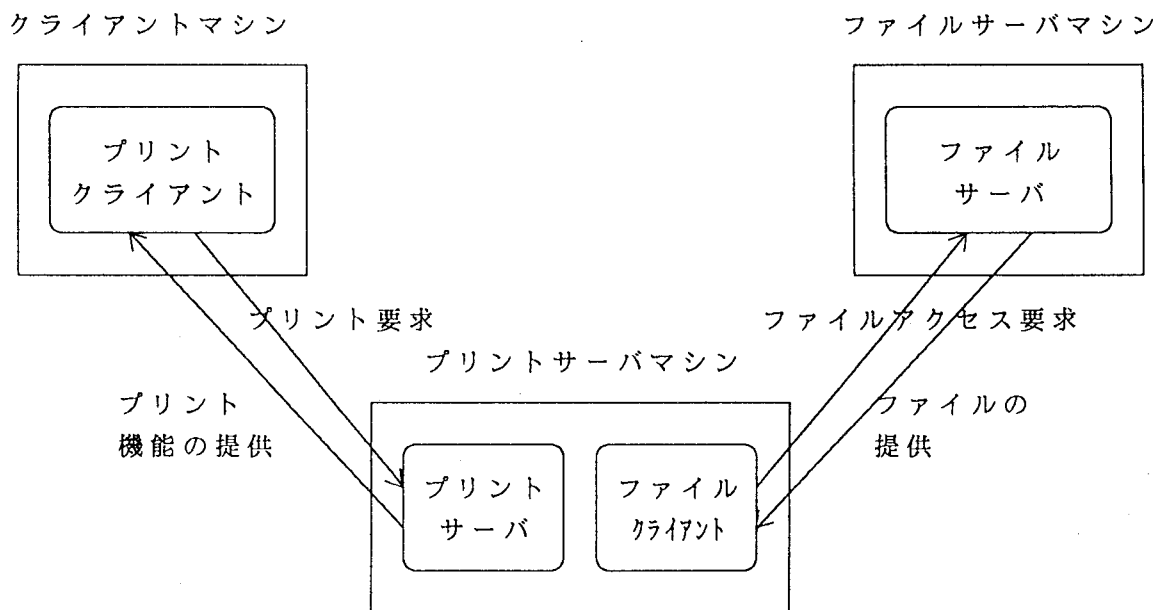
クライアント側の仕事としては、ユーザとのインタフェース、入力データの収集や検証、データの照会といった“フロントエンド”処理が代表的である。一方、サーバ側の仕事としては、共有する周辺装置の管理、共有するデータベースの検索など“バックエンド”処理が挙げられる。

(2) ハードウェアとサービス

大量なプリント、特殊な印刷などのプリント、大容量ファイルへのアクセスなどそのサービスを提供するサーバが存在するコンピュータをサーバマシン、サービスを要求するクライアントが存在するコンピュータをクライアントマシンという。

このサーバやクライアントはコンピュータ環境下ではプログラムでありオペレーティングシステムから見ればプロセスとして捉えられる。サーバマシン、クライアントマシンについて留意すべき点は、サーバマシンにはサーバとしてのプロセス、クライアントマシンにはクライアントとしてのプロセスしか搭載されていないというわけではないことである。図表VII-3に見られるように、プリントサーバマシンにはファイルサービスを要求するファイルクライアントが存在する。

図表VII-3 クライアントサーバモデル



(3) クライアントサーバと分散処理

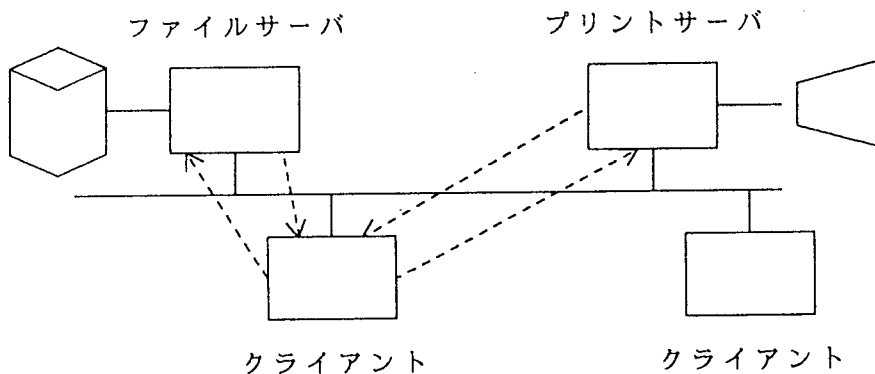
クライアントサーバモデルは、必ずしも分散処理システム形態をとるとは限らない。1台のマシン上にあるサービスのクライアントとサーバが同居する場合もある。例えば、Xウィンドウシステムは、1台のUNIXワークステーション上に両方存在することができる。

また、密結合のマルチプロセッサシステムでは別々のCPUにクライアントとサーバを割り振ることができる。

(4) クライアントサーバシステムの構造

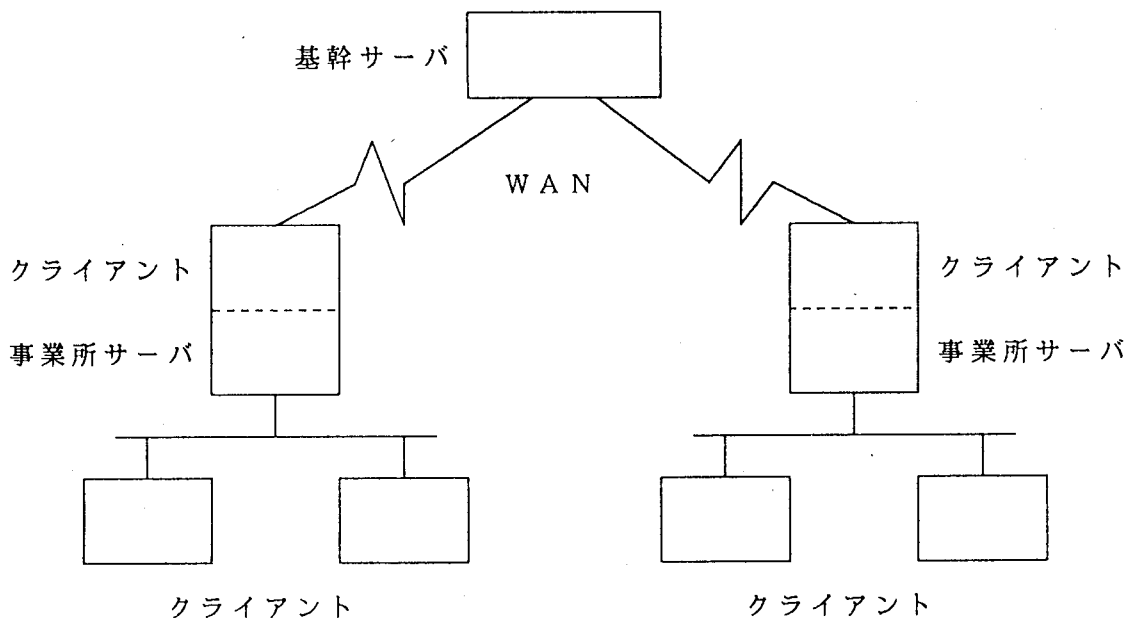
単純な構造として水平型のクライアントサーバシステムが構築できる(図表VII-4)。ネットワークを介してサービスの要求と実行がなされる。クライアント間に階層性はない。

図表VII-4 水平型クライアントサーバシステムの例



より複雑な構造として垂直型のクライアントサーバシステムが構築される(図表VII-5)。ある企業の本店に販売管理データベースがあり、その情報を各支店のクライアントがWAN(公衆回線あるいは専用回線)を介して利用する様な例に代表される。

図表VII-5 垂直型クライアントサーバシステムの例



## (5) 各種サーバ

しばしば構築される事例として下記が挙げられる。

- ① ファイルサーバ            —  ハードディスクなどの補助記憶装置を提供する。
- ② CPUサーバ               —  科学技術系のような大規模な計算能力を提供する。
- ③ データベースサーバ      —  データベースの検索や管理機能を提供する。
- ④ 通信サーバ               —  他のネットワークへのアクセスを行う。
- ⑤ プリントサーバ          —  印刷機能を提供する。
- ⑥ メールサーバ            —  電子メールの集配信機能を提供する。

## (6) クライアントサーバによるアプリケーション開発

クライアントサーバシステムにおけるアプリケーション開発においては、どの仕事をどこで実行するかということで、クライアントとサーバの役割分担を正しく認識することが重要である。

## (7) ソフトウェアインタフェース

クライアント側のアプリケーションからサーバ側のサービスシステムを実行する上で、XウィンドウやSQLがGUI、データベースアクセスサービスのインタフェースとして重要な役割を果たしている。このインタフェースの標準化はマルチベンダ化の促進、アプリケーション開発生産性の向上に役立っている。

また、クライアントサーバ型のパソコンネットワーク環境では、NetwareやWindows NTなどの普及により、より高度で低価格な安定したエンドユーザコンピューティングを享受できるようになってきた。これによるメリットは大きく、各種の流通ソフトウェアを安価に購入し信頼性の高いシステム構築ができる点で評価がますます上がっている。

## (8) クライアントサーバシステムの特徴

クライアントサーバモデルでは、アプリケーションの分割や階層化によって他のコンピューティングモデルに比べ多くの利益がもたらされる。ユーザの視点では、操作の容易性、情報へのアクセス容易性が、開発者の視点では、開発のスピード、メンテナンスの容易性が高く評価される。トータル的にも、システムの構築や維持を経済的に進められる。

## ① 利点

- ・ アプリケーションの分割によるモジュール化が行われるため、システム開発の迅速化、保守性の向上が図られる。
- ・ データベースサーバによるデータの一元管理から情報の一貫性が保たれ、GUIにより簡単な操作で情報の取り出しが可能となる。
- ・ クライアントサーバモデルを組織全体（企業、行政、学校など）に採用することで、必要な場所で必要な情報を、データの所在場所に依存せずにアクセスでき、情報を使っての意思決定促進につながる。
- ・ アプリケーションを分割し小型化できるため、従来のような汎用コンピュータでなくとも小型のマシンの分散設置でシステムを組むことができるためハードウェアコストが下がる。負荷に応じた最適なコンピュータを利用でき、コンピュータ資源の有効活用が可能となる。
- ・ 新しいコンピュータ技術の統合や新しいコンピュータシステムへの移行が容易。例えば、クライアントの追加や削除、高性能サーバへの移行が円滑に進められる。したがって、システム性能を段階的にまた容易に拡張できる。

## ② 留意点

- ・ ソフトウェアの保守   —  サーバ側のプログラム変更がクライアントまたは他のサーバプログラムに影響する場合がある。



- ・ 障害の原因究明 — 多数のベンダによる多数のコンポーネントでシステムが構成されるため、障害発生時のエラー原因を特定することが容易でない。要因の切り分けや障害の修復に対策が必要である。
- ・ 性能の低下 — 大型データの転送はネットワークの負荷を高め、性能低下につながる。また、クライアントがサーバを共有するために、サービス要求・提供インタフェースをとるために、いくつかのオーバーヘッドが生じる。
- ・ データの保全性 — データベースの整合性を保持する配慮が必要
- ・ セキュリティ — 情報の共有を重視するため、データの機密保護に関する十分な対策が必要となる。
- ・ システム管理 — ソフトウェアの更新、システムのバックアップに対する正しい管理が必要

### 3. クライアントサーバシステムのための機能

#### (1) 基本環境

##### ① FTP (File Transfer Program)

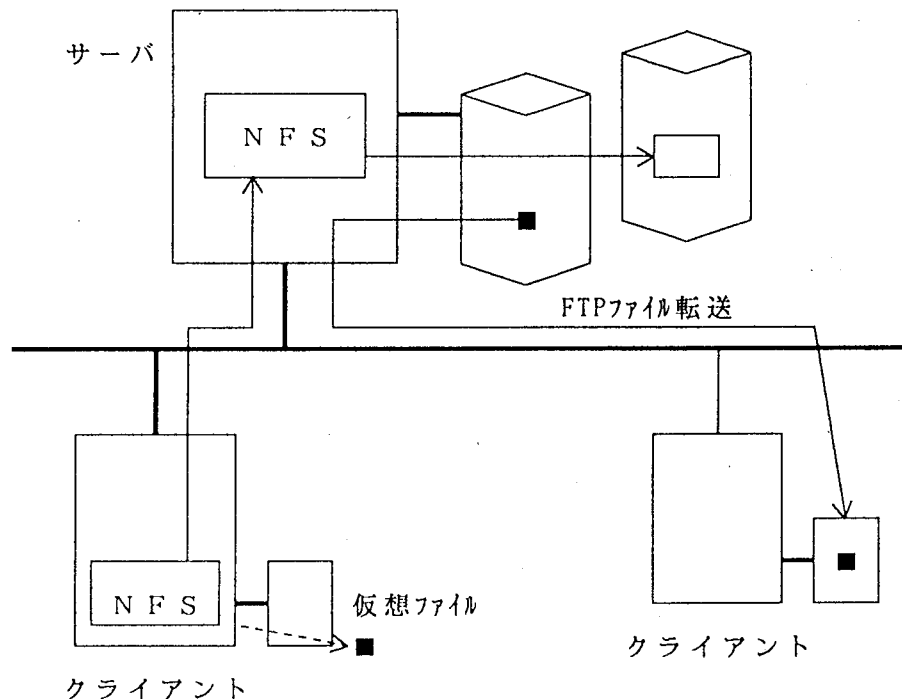
ネットワーク上の異なるコンピュータ間でファイルを参照したり転送したりする機能。

##### ② NFS (Network File System)

あるコンピュータからネットワーク上の他のコンピュータ上に存在するファイルをその物理的な配置を意識せずに、あたかも自身に存在するファイルであるかのように読み書きを可能とする機能。

例えば、ファイルサービス機能はFTPやNFS機能を用いて図表VII-6のように実行される。サーバの持つ大容量ファイルをNFSを使いローカルなファイルとしてアクセスする。サーバのファイルをFTPを使いクライアントのディスク装置に転送する。

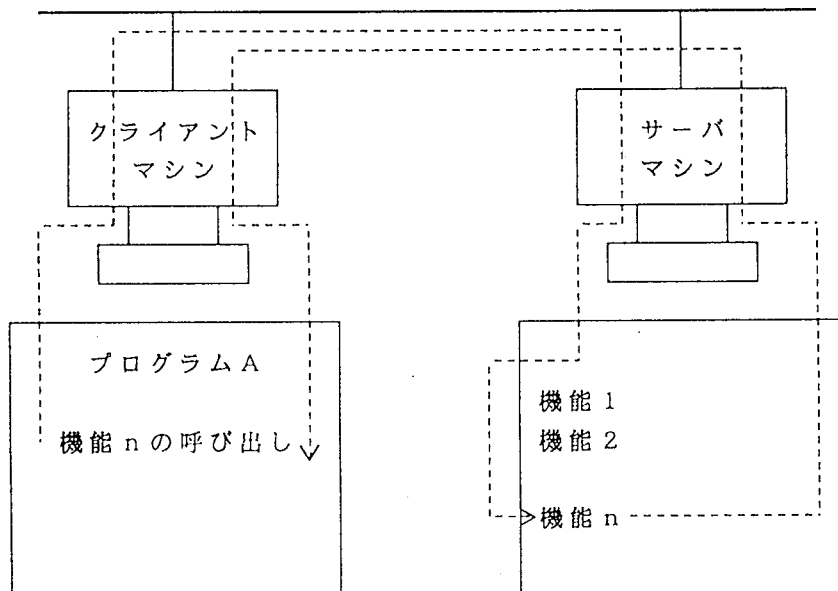
図表VII-6 ファイルサーバの例



## ③ RPC (Remote Procedure Call)

ネットワーク上の他のコンピュータ上に存在する処理手続きを呼び出す機能。サーバ側にサービスを提供するプロシージャ（関数）群として用意される通信プロトコルである。クライアント側はそれらの関数があたかも自分のものであるかのような呼出方法でサーバの機能を利用することができる（図表VII-7参照）。

図表VII-7 RPC



## ④ CICS (Customer Information Control System)

もとは、IBMの汎用コンピュータ上でオンラインシステムの構築を支援するシステムであり、これを応用してネットワーク上でファイル転送やリモートプロシージャコールを可能としている。

CICSにおいてクライアントサーバシステムを実現する機能として、

## ア. FS (Function Sipping)

あるクライアントから他のマシン上にあるデータを物理的な位置を意識せずにアクセス機能を提供するものである。例えば自マシンでのREAD/WRITE要求がデータの存在する他マシンに伝えられてそこで実行され結果が戻される。

## イ. TR (Transaction Routing)

あるクライアントからのトランザクション処理要求が必ずしも自マシンで処理されず、その処理を行うプログラムが存在する他マシンで行われる。要求したマシン側には対応する処理プログラムの存在するマシンIDが登録されており、ネットワークを意識せずにトランザクション処理が行われる。

## ウ. DPL (Distributed Program Link)

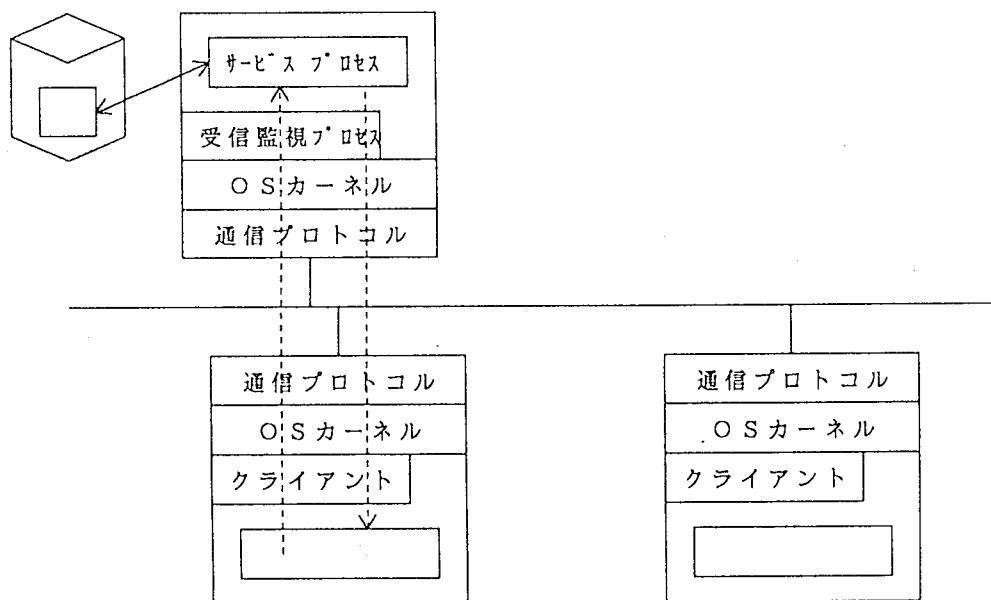
あるクライアントから他のマシン上にあるプログラムを呼び出したい場合、ネットワークを意識せずにその呼び出しができる。異なるマシン間で共通のデータ領域を設ければ、互いに自由にデータの交換が可能となる。

(2) サーバ機能

サーバは、ネットワーク上のいかなるクライアントからのサービス要求をいかなる時に出されても確実にそれを受け取り、然るべきサービス処理を行わなければならない。一方、クライアント側はサーバに処理の依頼を出し、その結果がサーバから送られてくるのを待つだけでよい。

サーバには、クライアント側からの処理要求を常時監視し、依頼を受け付けるプロセスが常駐している必要がある。

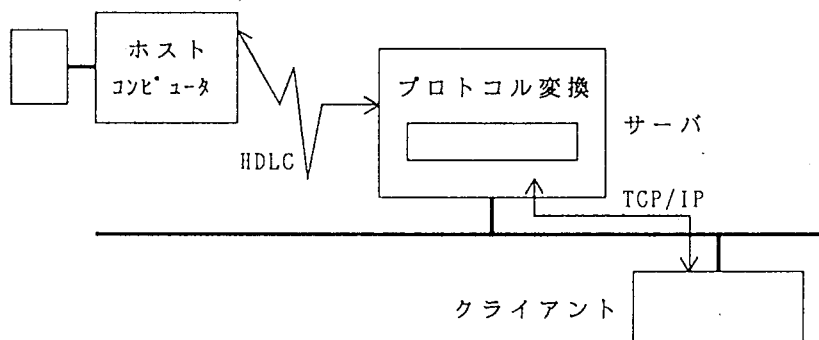
図表VII-8 サーバの機能



① コミュニケーションサーバ

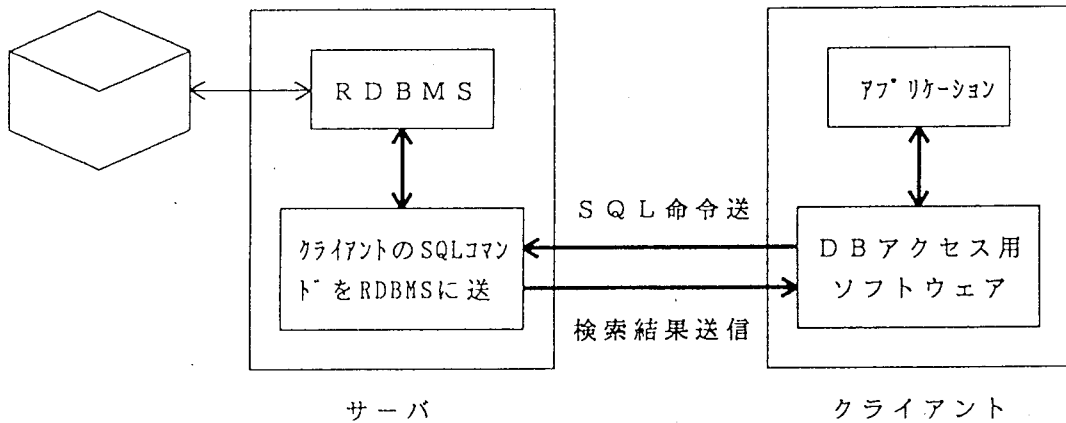
通信ソフトウェアを使い、例えば、TCP/IPと汎用コンピュータメーカ独自の通信プロトコル（例えばIBMのSDLCあるいは国内メーカのHDLCなど）を翻訳し、クライアントのアクセス可能な範囲を拡大することができる（図表VII-9）。

図表VII-9 コミュニケーションサーバの例



② データベースサーバ

図表Ⅶ-10 SQL インタフェースを備えたRDBMS利用例



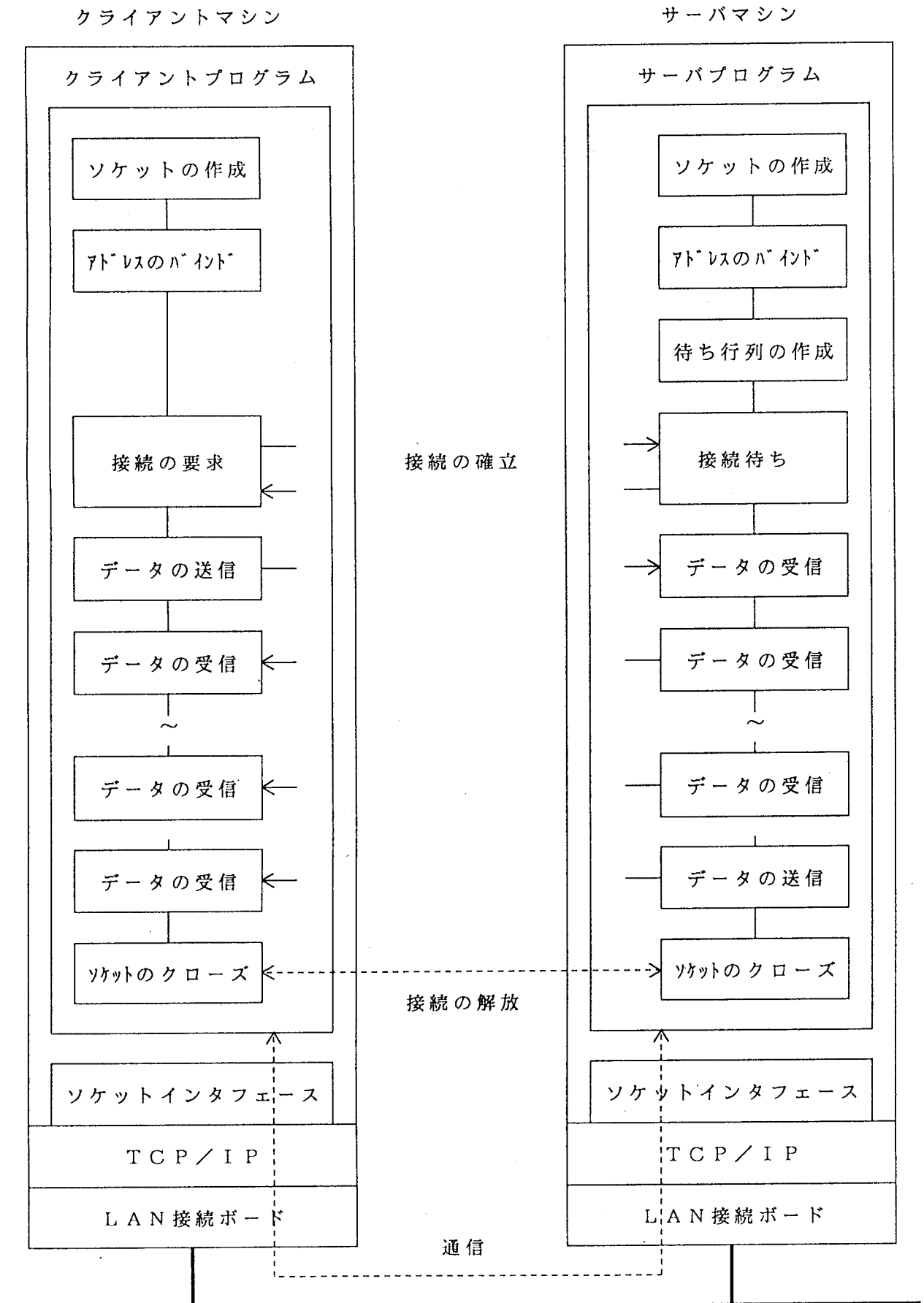
このような製品例として、クライアント側では、SequeLink アプリケーションが日本語 SequeLink クライアントを使い、サーバ側で、日本語 SequeLink サーバを用いて [ORACLE、informix、Sybase] などの DBMS を選択できる。

4. クライアントサーバシステムのソフトウェア構造

図表Ⅶ-11は、クライアントとサーバのそれぞれのプログラムが二つのプロセス間通信によって進められるフローを示している。各々のマシンは、図にあるようなソフトウェア構造をとっている。通信を司るのは、TCP/IPとソケットインタフェースである。両プログラム間での通信には、ソケットの作成と、ソケットへのネットワーク上のアドレスの結び付け（バインドという）が必要となる。

サーバは、複数のクライアントからの同時依頼を受け付けられるような仕掛けが必要で、クライアントは依頼拒否に対する再依頼処理が必要となる。

図表VII-11 ソケットを使ったプロセス間通信フロー

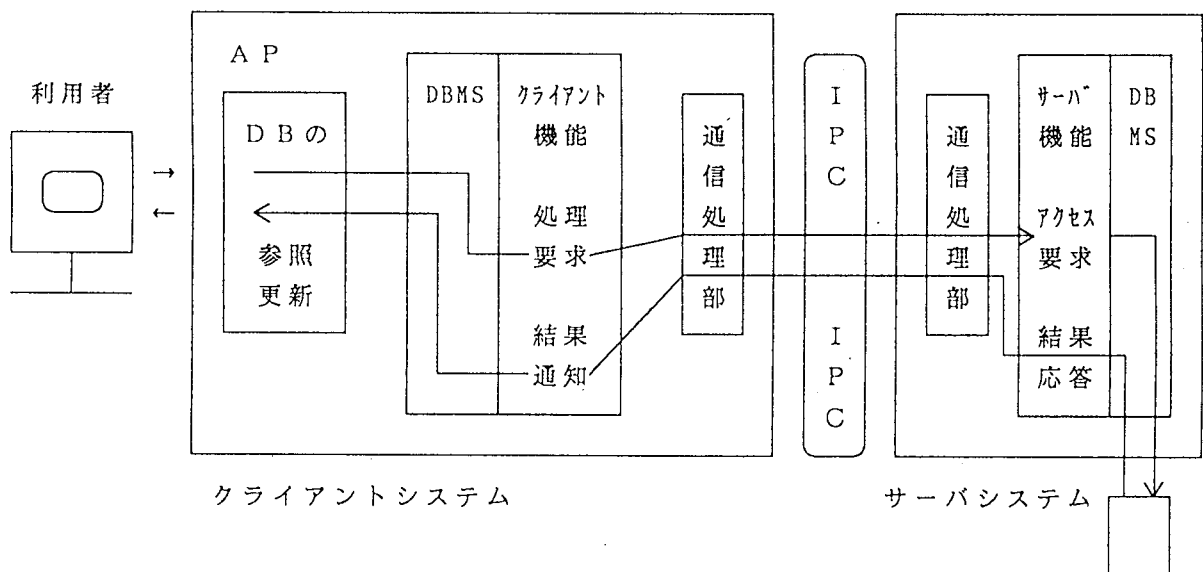


5. 事例

単純で一般的なクライアントサーバ事例を図表VII-12に示す。クライアントのAP（アプリケーションプログラム）からDBMSに対してDBのアクセス要求が出ると、クライアント側のDBMSはデータベースの所在を調べ、サーバに存在する場合はクライアント機能を用いてサーバのデータベースアクセス処理の要求を送る。下図では、クライアントとサーバの間でIPC（インターナルプロセスコミュニケーション）によるプロセス間通信が行われる。

サーバ側ではサーバ機能がクライアントからのアクセス要求を受理し、DBMSに対して実データベースへの実際のアクセスを実行する。アクセスが終了するとその結果が逆の経路でクライアント側に返される。

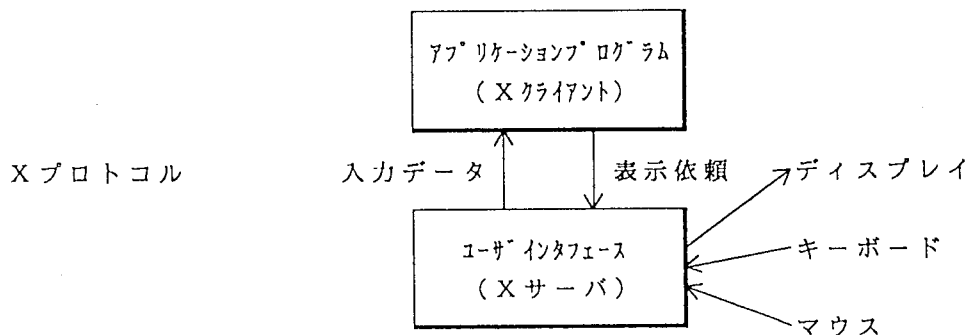
図表VII-12



(1) ユーザインタフェースにおけるクライアントサーバシステムの事例

Xウィンドウシステムはクライアントサーバモデルを採用した実現構造を持っている。

図表VII-13 Xウィンドウシステムのクライアントサーバモデル



Xサーバは、クライアントからの依頼によりウィンドウの生成、文字や図形の描画を行うとともに、ディスプレイやキーボード、マウスの入出力管理を行い、入力データについてはクライアントに渡す。

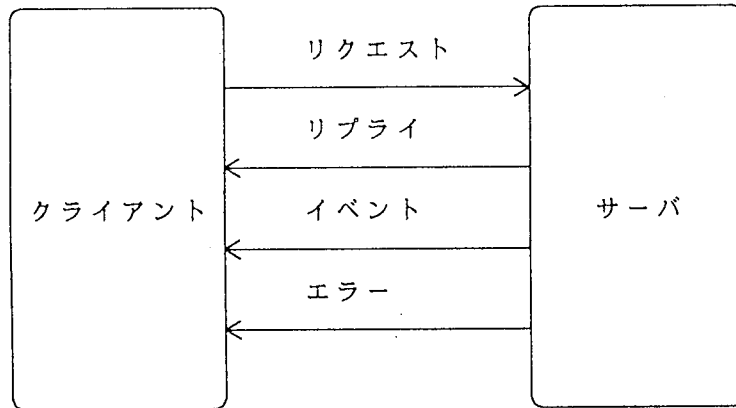
Xクライアントは、逆にサーバに対してディスプレイ上へのウィンドウの生成や文字、図形の描画依頼を出す。また、サーバからキーボードやマウス操作による入力情報を得て、それに対応する業務処理を実行する。

Xウィンドウシステムでは、ハードウェアに依存した処理をサーバで行うことにより、Xクライアントのハードウェア依存性を低くしているため移植性の高いプログラムを作成できネットワーク化し易い利点がある。

① Xプロトコル

Xウィンドウシステムでは、XクライアントとXサーバ間の通信に「Xプロトコル」というメッセージフォーマットを規定している。クライアントはこの形式でXサーバへの依頼を記述する（図表VII-14）。

図表VII-14 Xプロトコル

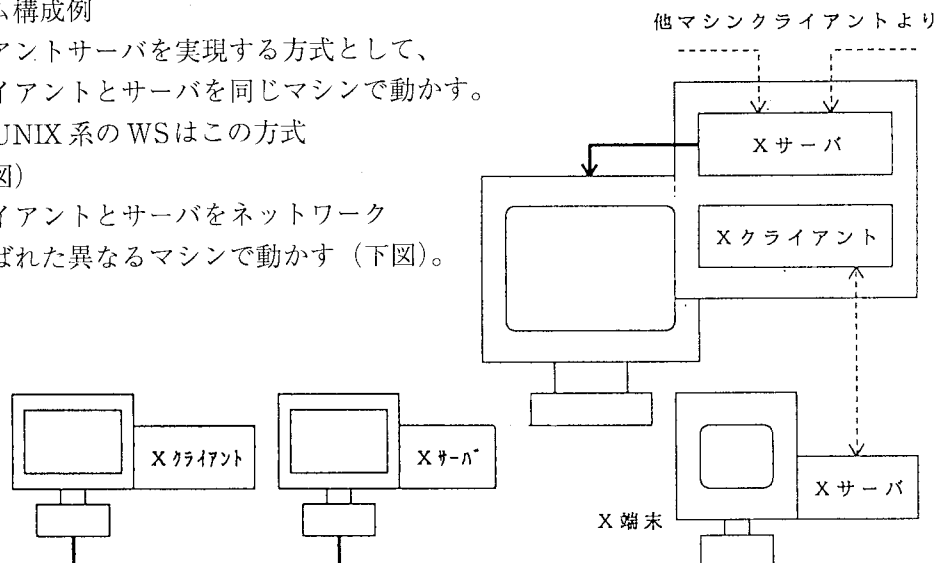


- ・ リクエスト  
ウィンドウの生成・消滅、ウィンドウへの文字や図形の描画や再描画などXサーバに対して処理要求を出す。
- ・ リプライ  
Xクライアントからのリクエストに対してXサーバが返すメッセージ  
例えば、ウィンドウの大きさやその位置が情報として返される。
- ・ イベント  
キーボード入力、マウスの移動やクリックなどによりXサーバ側に発生した事象をXクライアントに通知する。
- ・ エラー  
Xサーバ側にエラーが発生した場合、メッセージがXクライアントに返される。

② システム構成例

クライアントサーバを実現する方式として、

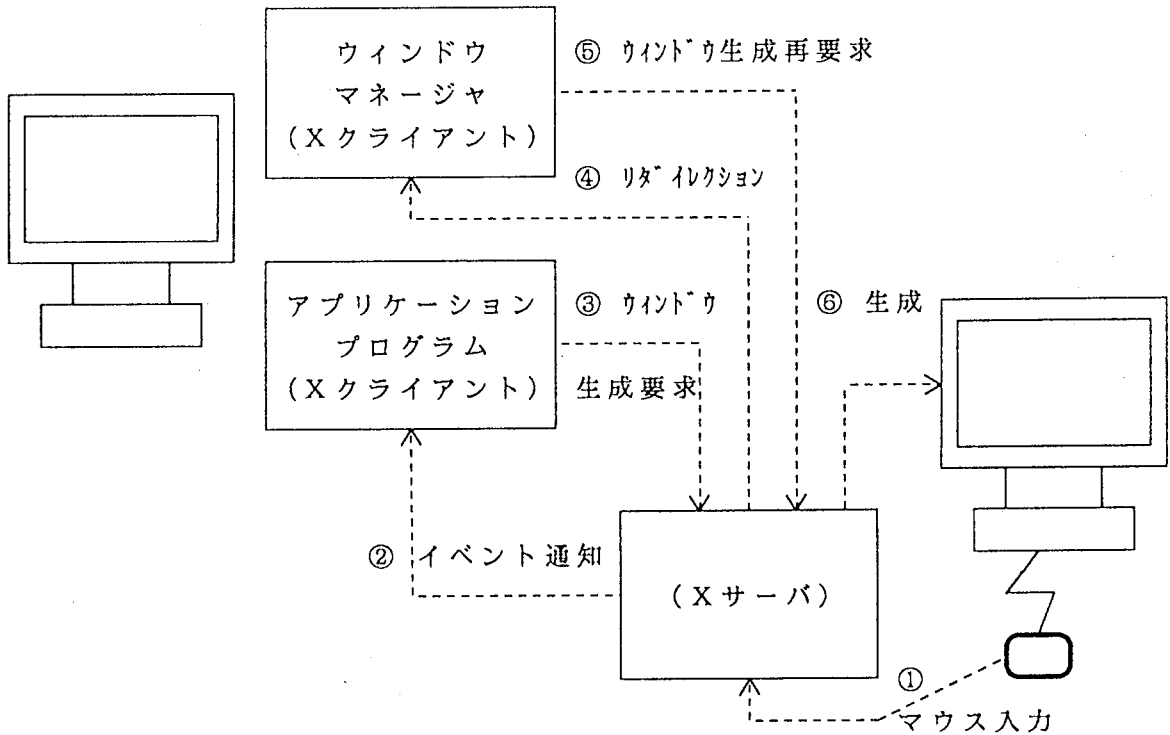
- ・ クライアントとサーバを同じマシンで動かす。  
通常UNIX系のWSはこの方式  
(右図)
- ・ クライアントとサーバをネットワークで結ばれた異なるマシンで動かす (下図)。



③ 処理例

マウスを用いて画面生成操作を行うと、その情報をXサーバが受け取り、マウスクリックイベントをアプリケーションプログラム（Xクライアント）に通知する。Xクライアントは、Xサーバに画面生成リクエストを発行する。Xサーバは、それをウィンドウマネージャ（Xクライアント）にリダイレクションする。ウィンドウマネージャは、規定を付加してXサーバにリクエストを再発行する。最後に、Xサーバはそのリクエストを基にウィンドウを生成する。

図表Ⅶ-15 処理例



(2) オンラインランザクションにおけるクライアントサーバシステムの事例

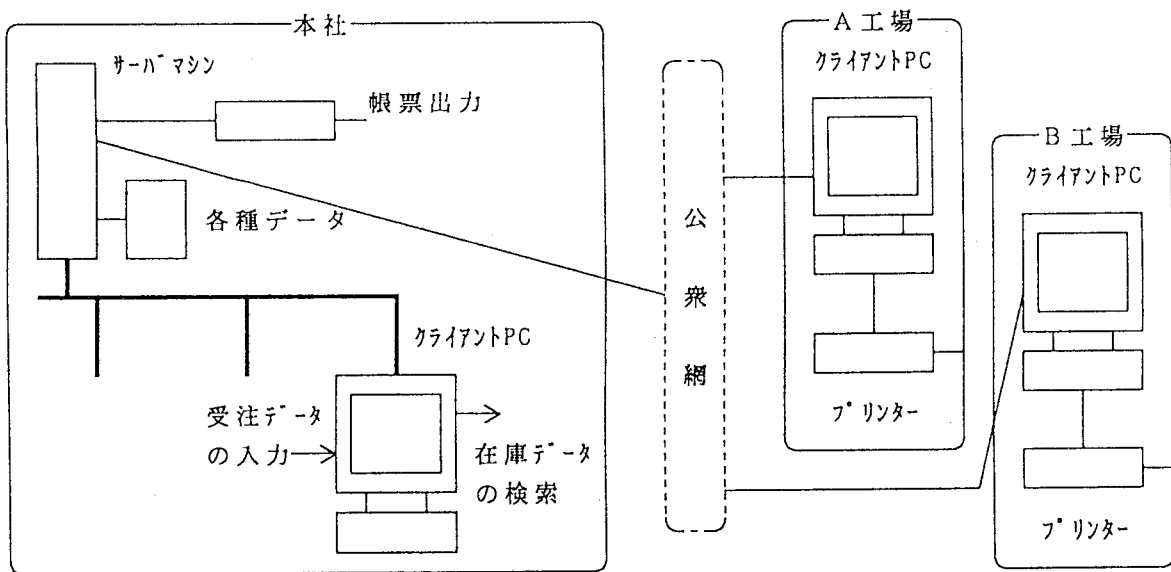
受発注処理、在庫問い合わせ、残高照会、座席予約などに代表されるオンラインランザクションシステムでは、各所に設置された端末からネットワークを介して中央のデータベースの読み込み、書き込み、更新などをリアルタイムに実行する。

例えば、受注配送業務をクライアントサーバモデルで構築することを考える。顧客からの受注に始まり在庫引き当てが行われ、配送指示・出荷指示を出すものとしよう。

システムとしては、本社にパソコンサーバをおき、クライアントパソコンを本社と工場に設置し、パソコンベースのクライアントサーバとする。顧客からの受注、工場在庫引き当て、車の配送割当がネットワークで結ばれオンライン化される。受注、在庫などのデータを手元のパソコンで管理し、帳票としてプリント出力ができるシステムである。



図表VII-16 クライアントサーバ型受注配送システム



当システムでは、クライアントPCから受注情報の他、得意先、在庫など受注処理に必要なマスターデータの入力、また帳票出力の指示ができる。このデータが構内回線や公衆回線を介してサーバに登録される。サーバはデータを一括管理し、何処のPCからも同じデータを画面で検索したり、配送、出荷のための帳票出力をオンラインでできる。

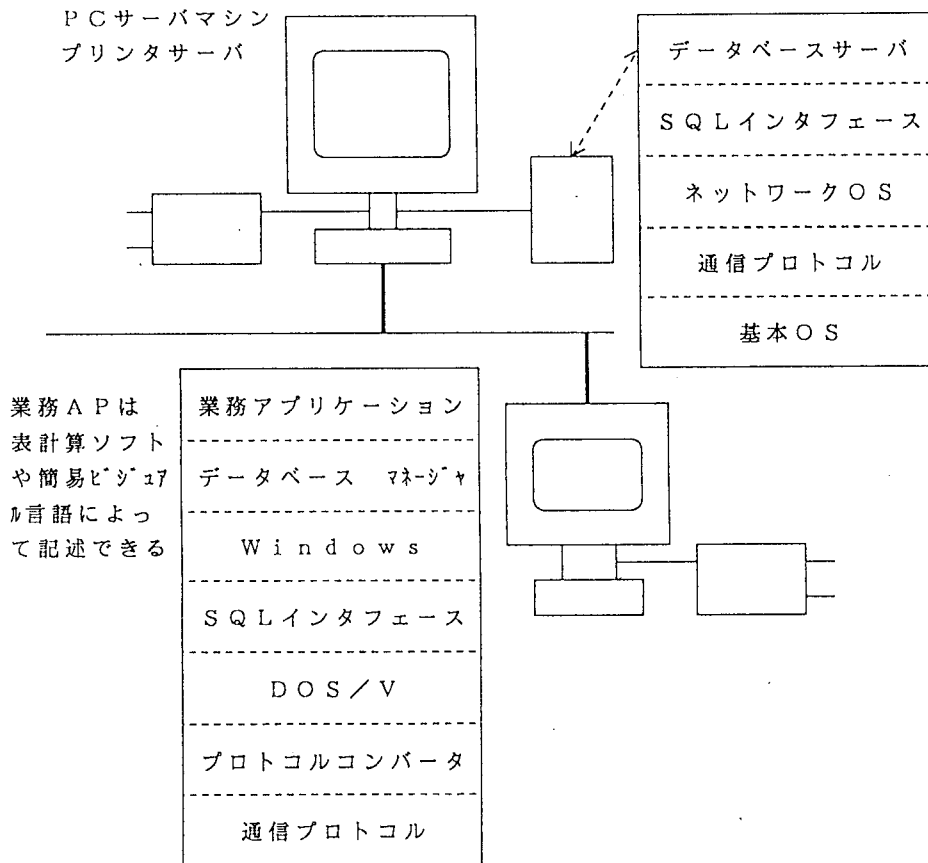
クライアントサーバシステムでは、従来のホストマシンで行ってきた入出力処理、業務処理などのアプリケーションプログラムおよびデータベース処理を、別々のマシンで行うという点で大きく異なっている。

図表VII-17にあるように、クライアントマシンでは、端末からのデータ取得やチェックおよび端末への出力などの入出力処理ならびに業務処理を行う。業務処理にはデータベース処理が存在するが、それはサーバに対してアクセス処理を依頼し、その結果を取得するだけである。

サーバマシンでは、クライアントからの依頼をもとに、データベースを読んだり、書いたり、あるいは更新を行う。

業務アプリケーションはホストコンピュータ時代にはCOBOLなどで書いていたが、最近は表計算ソフト、簡易ビジュアルライザによって簡単にできるようになった。

図表Ⅶ-17 クライアントサーバ型のシステム



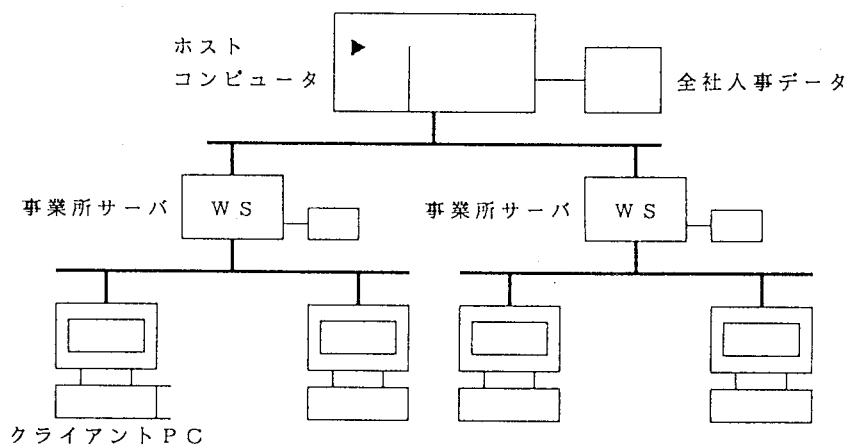
① 人事情報システム

ホストコンピュータで構築されていた人事情報システムの使い勝手に限界が来たことを想定する。市販されている人事情報システムパッケージの作りが非常に柔軟でカスタマイズが容易にできる。このため、このパッケージを導入し、

- ・ 人事にからむ、昇格、出向、異動と辞令の作成
  - ・ 人事シミュレーションによる人事企画
- 等の業務の生産性向上を図ることとした。

人事データの更新はホストコンピュータに残し、最新データはサーバマシンにロードしておく（図表Ⅶ-18）。

図表Ⅶ-18 人事システムの構成例

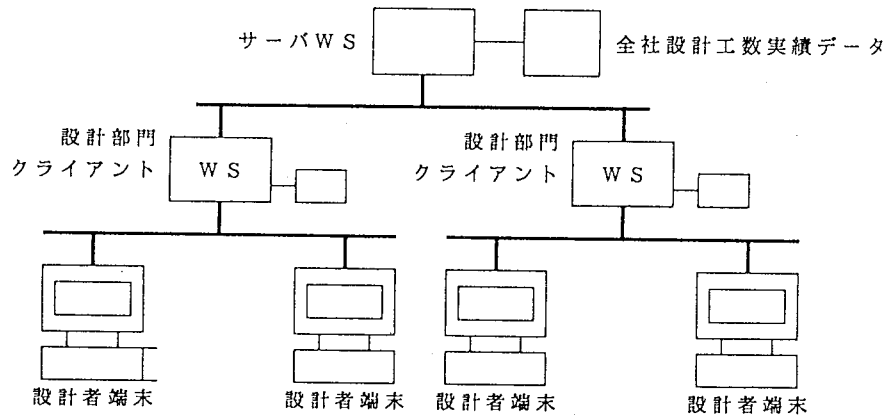


② 設計工数予測システム

自動車メーカーにおける設計工数実績情報を各設計部門に公開。スプレッドシートを用いて工数の分析、新車設計時の設計工数、必要設計要員の推計、設計者の担当業務割当などの計画・立案を可能としている(図表VII-19)。

各設計部門では、それぞれの設計に関わる生産性の向上を図れるような仕掛を用意。企画情報、市場情報、製品カタログ、部品カタログ、設計図面、作業進捗情報はすべてサーバデータ化して、コンカレントエンジニアリングをある程度可能としている。

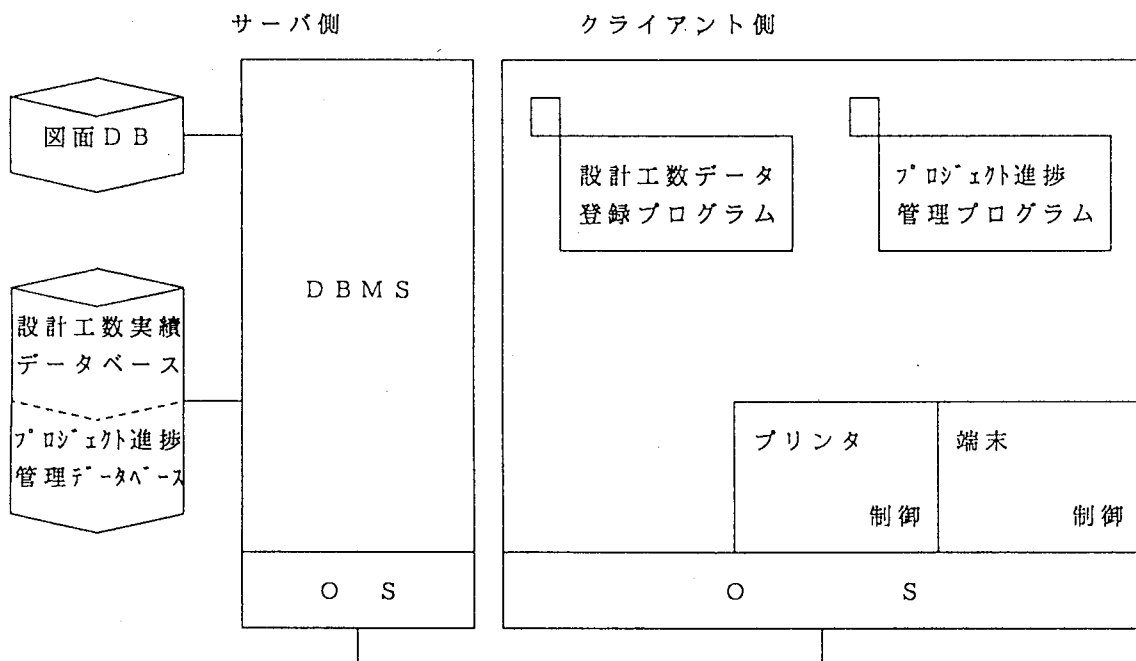
図表VII-19 設計工数予測システム



③ ソフトウェア構成

上記設計工数予測システムのソフトウェア構成は図表VII-20の通り。サーバマシンは、設計工数実績データベース、各種図面データベース、プロジェクト進捗管理データベース、クライアントマシンからのデータアクセス要求を実際に行うデータベース管理システムからなる。クライアントマシンは各種データベースのアクセス、新しい設計工数データの登録、進捗データベースの参照や入力、更新を行うプログラムと、他に端末制御やプリンタ制御などから構成される。

図表VII-20 ソフトウェア構成



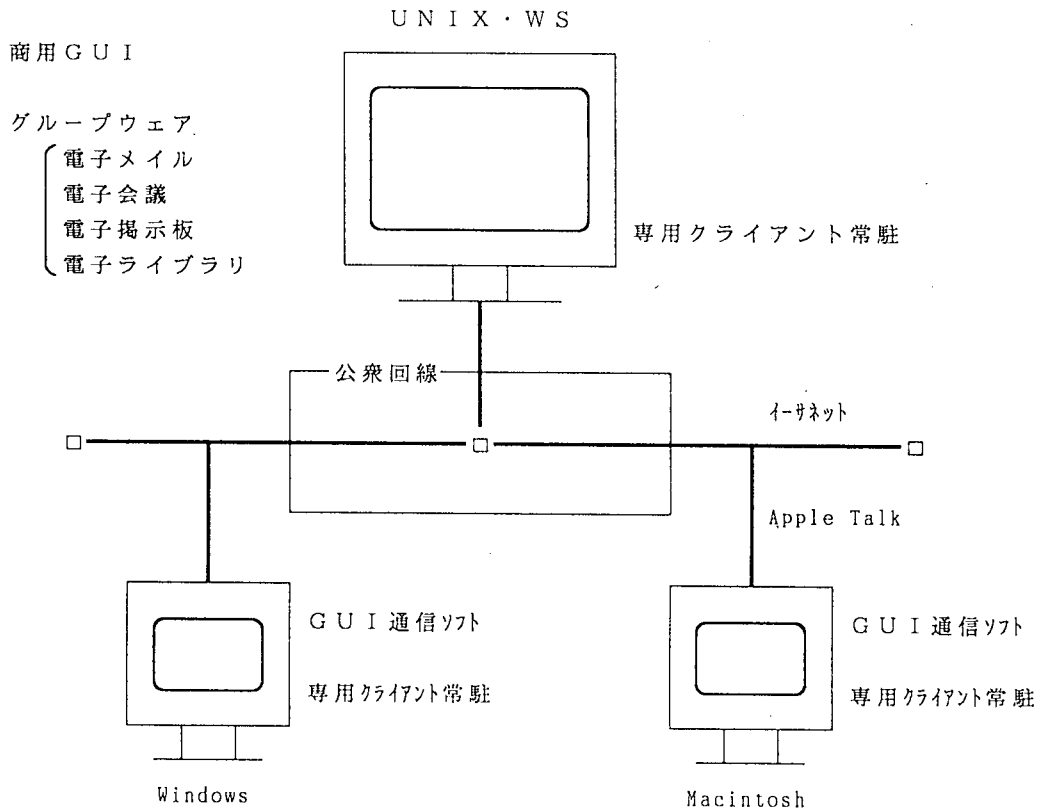
(3) 商用のクライアントサーバ型構成事例

① クライアントサーバ型グループウェア

UNIX ワークステーションをサーバとしてネットワーク上でグループウェアを実現している例 (図表Ⅶ-21)。また、ネットワーク上でのGUI環境を利用することも可能としている。

クライアントには通信プロトコルとしてTCP/IPもサポートしており、また、FTPエミュレーションも可能であるため、高速なファイル転送が実現されている。

図表Ⅶ-21 商用クライアントサーバシステムの事例

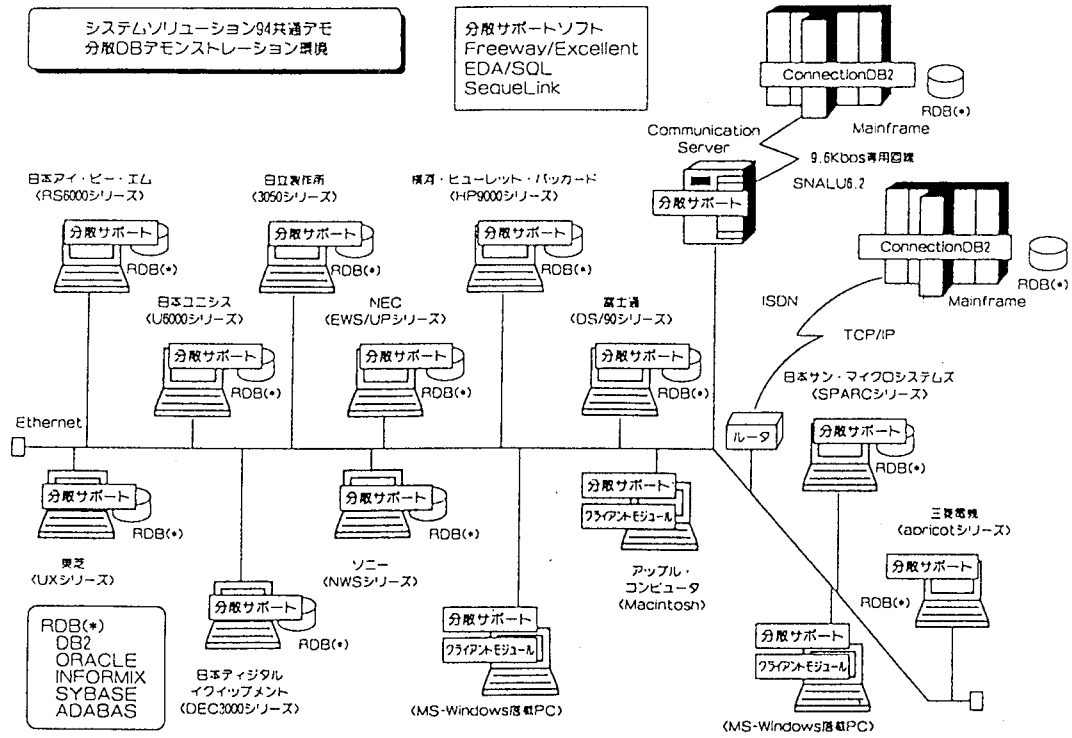


② シングルイメージの分散データベース

(社)日本情報システムユーザー協会(略称:JUAS)が94年のビジネスショーにおいて、種々のプラットフォームとデータベースソフトウェアを用いてマルチベンダ環境のクライアントサーバシステムを公開した。

図表Ⅶ-22は、メインフレーム、WSメーカー、パソコンメーカー、システムベンダなどオープンシステムビジネスを推進する企業が参加し、完全に異機種環境でありながら1つの標準による分散データベースとオンライントランザクション・プロセッシングの実演イメージである。“システム・ソリューション'94共通デモ”と称しクライアントサーバ型システムでの分散データベース、オンライントランザクション処理、分散ネットワーク、分散管理が実演された。

図1 シングルイメージ分散データベース



(社) 日本情報システムユーザー協会

OFFICE C/S

## 6. 主要用語

クライアント	サーバ	クライアントサーバモデル
コンピューティングサーバ	クライアント/サーバシステム	
プラットフォーム	サーバマシン	データベースサーバ
ファイルサーバ	プリントサーバ	ユーザインタフェースサーバ
コミュニケーションサーバ	FTP	NFS
RPC	ソケット	TCP/IP
ソケットインタフェース	CICSのFS	CICSのTR
CICSのDPL	バインド	Xウィンドウシステム
ネットワーク透過	Xクライアント	Xサーバ
Xプロトコル	排他制御	