

LAN上でのサービス
ベースシステム構築の一方式

何 千山* 橋高大造** 荻野 正* 田中英彦*
(* 東京大学工学部 ** 三菱電機)

4F-6

1. はじめに

分散サービスを有効に管理し、ユーザに便利なユーティリティを分散環境で提供することを目的としたシステムが、サービスベースシステム(SBS)である。サービスベースシステムの特徴は、ユーザが分散性を意識せずに、サービスを自由に組み合わせて利用することができ、システム管理者が他のノードと独立にサービスを拡張ができることである。SBSは、網の形態に制限はなく、LAN上にも広域網にも適用できるシステムであり、本稿では、LAN環境に適したSBSのモデルとその実験システムについて検討する。

2. LAN環境でのSBSの特徴

LANの場合、通信の高速性や高信頼性などの特徴を生かすために、それまでの通常の通信回線を使った分散システムとは全く異なった構成を取る場合が多い。LANが非常に普及してきた今、サービスベースシステムでも、LANに適した構成を考える必要がある。

まず、LANの特徴をいくつかあげ、それがLAN上のサービスベースシステムの構成にどのような影響を及ぼすかを考える。

(1) 通信の高速性と高信頼性

LANに於いては、高速大量のデータ転送をすることが可能になっているため、通信量のある程度増やしても、性能への影響は少ない。LAN上のSBSでは、サービスの要求/応答が頻繁に起こるような集中管理モデルを仮定してもよい。

(2) システム管理者

通常の網では、ノードごとにシステム管理者が異なっているのが普通であるが、LANに於いては、複数のノードが同じ組織に属して同一の管理者のもとで管理されている場合が多い。その同じ組織内では、プリンタやディスクといった計算機資源は共有されているのが普通である。複数のノードが同じ管理者によって管理されている場合は、集中して管理する方法が効率的である。この同じ管理者の下にあるグループをクラスタと呼ぶことにする。

(3) LAN間の相互接続

現在の計算機網では、gatewayによってLANとLANを接続して広域網を構成するのが普通である。LANのSBSがうまく構築できれば、広域網を用いた巨大なSBSを構築できる。

次に、SBSの位置付けを考えてみる。LAN上でのシステム・ソフトウェアとしては、プロトコル、OS(ネットワークOS、分散OS)、応用的ソフトウェアなどがある。SBSは、最初からシ

ステムを開発するのではなく、以上のようなシステムソフトウェアの環境の上に構築されるシステムである。すなわち、SBSは、上位層のプロトコルの機能(例えば応用層)だけ定義し、下位層については、ネットワークを構築する設計者に任せる。なお、SBSはネットワークOS、分散OSの機能(remote copy, remote login等)に基づいて構築される。SBSでは、サービスを自由に組み合わせることができ、各ノードでは、独立にサービスを拡張できるので、SBSの上で応用的ソフトウェアを容易に構築できる。

3. システムの構成

LAN環境での特徴を生かすために、クラスタ単位の集中管理方法を検討する。LANはクラスタから構成される。図1に示すように、クラスタには、ユーザが直接使えるノードとして、高性能のワークステーション(WS)と、簡単な処理機能しか持たないパーソナルコンピュータ(PC)が存在する。また、クラスタには、そのクラスタ内で共有されるサーバ(ファイルサーバ、プリンタサーバなど)が存在する。クラスタ内のサービス管理及び外部のクラスタとのサービス要求を行なうために、クラスタ・サービスサーバ(CSS)を導入する。CSSは、クラスタ内から利用できるすべてのサービスについての記述を集中して持ち管理するため、他の各ノードが持つべきサービス記述の量をかなり減少させることができる。従って、CSSでは、クラスタ内の大部分のサービスの要求、応答を扱うため、高速、大規模のデータベース機能が必要である。

クラスタ内からクラスタ外へのサービスの要求/応答、そしてクラスタ外からクラスタ内へのサービスの要求/応答は、すべてCSS

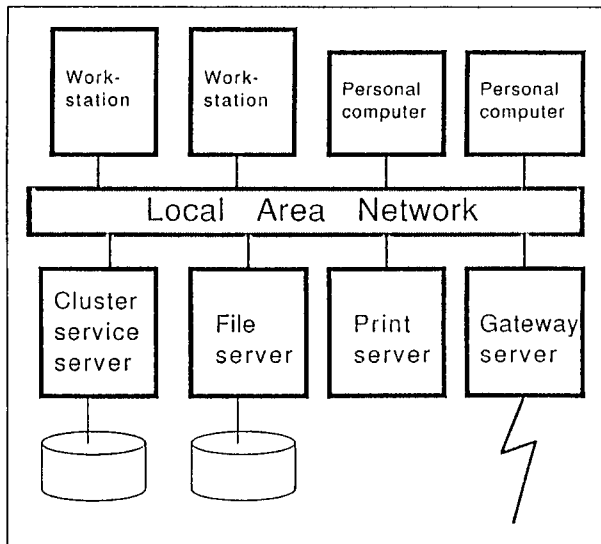


図1 クラスタの構成

A Model of Service Base System
for Local Area Networks

Qianshan HE, Taizou KITAKA, Tadashi OGINO,
and Hidehiko TANAKA. University of Tokyo.

を介して行なわれるので、クラスタ外のサービスを新たに取り入れる時など、クラスタ外のサービスの使用変更は、CSSのビューの変更のみで行なうことができる。また、クラスタ内のノードの追加、削除、変更といったノードの再構成も、クラスタ内で独立に処理できる。このように、CSSを導入することにより、クラスタ内とクラスタ外が分離され、サービスの管理が非常にやりやすくなる。

各クラスタのサーバ（プリンタ、ファイル・サーバなど）の提供するサービスはまとめられてCSSで記述され、WS、PCなどのノードに提供される。WSは、自ノード内のサービスについての記述を持ち、CSSからのサービス要求に従ってサービスを提供することができる。また、WSはCSSで記述したサービスを利用することもできる。PCは処理能力が低いので、自分が他のノードにサービスを提供することはなく、自分でできる簡単な処理以外は、すべてCSSにサービスの要求を行なうことにしている。

従って、PCには、簡単なサービス・インタプリタがあればよい。一方、WSには、サービスを提供するために、インタプリタのほかに、サービスの処理系と記述管理部がなければならない。各WS、PCでのSBSが提供する機能は、ユーザに提供する機能と応用プログラムに提供する機能に分けられることができる。ユーザに提供する機能は、インタプリタによって実現され、ユーザがインタプリタでサービスの要求等を行うことができる。応用プログラムに提供する機能は、各ノードにあるSBSライブラリによって実現される。SBSライブラリには、サービスの要求、応答用のルーチン群が用意されており、応用プログラムからこれらのルーチン呼び出すことによって、CSSにサービスを要求したり、サービスを組み合わせたりすることができる。

4. クラスタ・サービス・サーバにおけるサービスの管理

CSSでは、サービスに関する情報を次の3つのレベルに分けて記述、管理する：

- (1) 内部ビュー：クラスタ内の各ノードが提供するサービスに関するビューである。
- (2) 外部ビュー：クラスタ内の各ノード或いは他のクラスタのCSSに見せるビューである。
- (3) 概念ビュー：クラスタ内のサービスと他のクラスタのサービスを統合したビューである。

CSSは、サービスに関する情報を格納する関係データベースを中心として、図2のような構成になる。サービス要求モジュールは、クラスタ内、クラスタ外からのサービス要求を受け取り、データベースを検索し、必要なサービスに関する情報をメッセージ・デリバリー・モジュールに渡す。メッセージ・デリバリー・モジュールはこれらのサービスに関する情報を分析し、サービスの存在するノード或いはクラスタへさらに要求を出し、返ってくる応答を、最初にサービスの要求を出したノード或いはクラスタへ送る。ネットワーク・モジュールはノード間の通信を管理する。

サービスを各ノードに提供する前に、予めCSSでサービスに関する情報を記述しておかなければならない。サービスに関する情報

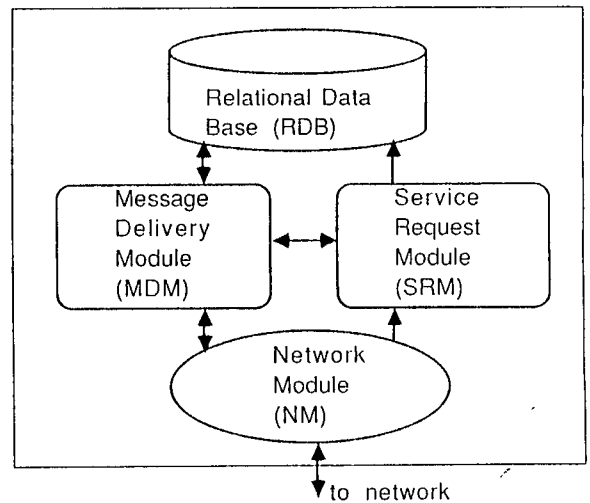


図2 クラスタ・サービス・サーバの構成

は複数の属性からなり、属性と属性間の関係を表す為に、既にtreeの表現形式を提案した[1]。サービスの記述方法としては、まず、サービスに関する情報をtreeで表し、さらに正規化された関係データベースのリレーションとして蓄積される。現在の実験システムでは、正規化されたリレーションとして、以下の六つを用意している：

```
name ( e_name, c_name, i_e_name, cluster, node, fd )
dictionary ( e_name, text )
directory ( e_name, type, acc_path )
arg_inf ( e_name, argnum )
argument1 ( e_name, io_inf, datatype )
argument2 ( e_name, io_inf, datatype )
```

その他に、ネットワーク情報を管理するリレーションが二つある。

5. 実験システム

以上のモデルを詳しく検討するために、我々は実験システムを構築した。実験システムは、昨年に完成した東京大学工学部LANを利用し、3台のSUN3、3台のVAXからなっている。これらの計算機は二つのクラスタに分けられ、各クラスタに一台のノードをCSSとする。実験システムの上で、サービスを利用する例題により、このモデルの有効性を示した。

6. おわりに

本稿では、LANに適したSBSの構成方式について検討した。LANの環境では、LANの特徴を利用して、柔軟性の高いSBSを構築することができる。SBSは、他のLANシステム・ソフトウェアと比べてユーザに便利なユーティリティを提供することを目的とし、将来のLANの応用システムの開発にとって非常に有効であろう。今後の課題としては、システムの効率についての評価を行い、サービスに関する情報を容易にデータベース化するためのインタフェースを検討する。

参考文献：

- [1] 何千山他、「サービスベースシステムにおける分散資源及び資源の仕様記述」、第35回情報処理全国大会、3U-7、1987.10