

オフィス内分散処理用ネームサーバ

3Q-9

山下 東* 岩井 徳幸** 加藤 正道** 黒沢 浩一***
 *日立中部ソフトウェア(株) ** (株)日立製作所システム開発研究所
 ***日立ニュークリアエンジニアリング(株)

1. はじめに

現在、オフィス内LANの普及により、メール、文書キャビネット、分散ファイル、分散データベースなど、ワークステーション間で資源の共用や情報の交換などするための各種分散システムが開発されてきている。本稿では、これら分散システムの開発で必要となる、共用資源の名前管理を統一的行うネームサーバについて、開発方針を述べた後、その実現方式について報告する。

2. ネームサーバの開発方針

ネームサーバの開発において、以下の項目に重点を置き設計した。

- (1) O S Iディレクトリのサブセットとして、メール、分散データベース、分散プログラムライブラリなどのネームサーバとしても適用できるよう、オープン性を重視する。
- (2) システムの規模・形態に合わせて、ネームサーバを自由に配置できる。
- (3) ネームサーバ間の対話を実現して、ユーザに、すべてのネームサーバの情報を提供できる。

3. システムの構成(図1)

ネームサーバは、LAN上を論理的に区切られた空間(ドメイン)ごとに1つずつ配置され、それぞれのドメインごとに独立して運用できるようにする。これにより、要求の集中を防ぎ、システムの拡張性、信頼性を高める。また、他ドメインの情報は、ネームサーバ間の通信により実現する。

4. 実現方式

(1) 階層的名前付け

各ネームサーバの関係は全体としてツリー構造を形成する。共用資源の識別名は、そのツリー構造に従って付けられるので、各ネームサーバごとに資源名のユニーク管理を行えば、すべての資源はユニークに識別できる。

識別名の形式

〈絶対識別名〉 ::= /{〈ネームサーバ名〉/}...〈資源名〉

〈相対識別名〉 ::= {〈ネームサーバ名〉/}...〈資源名〉

または 〈資源名〉 など

(2) グローバルユーザID管理

ワークステーションごとのユーザ管理では、1人のユーザが複数のワークステーションを利用する場合、すべてのワークステーションに同じユーザIDを登録するとは限らない。また、異なるユーザが、別々のワークステーションで同じユーザIDを登録することも考えられる。したがって、リモート資源のアクセスには、ワークステーションごとのローカルユーザIDを利用することができない。

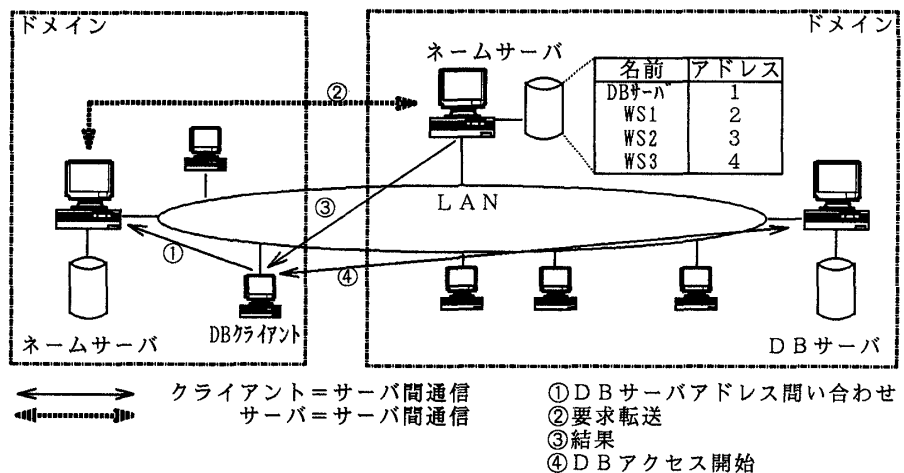


図1. ネームサーバシステム構成例とDBからの使用例

Name Server for Work Group Computing

Azuma YAMASHITA*, Noriyuki IWAI**, Masamichi KATO**, Koichi KUROSAWA***

*Hitachi Chubu Software, **System Development Lab., Hitachi,

***Hitachi Nuclear Engineering

これを解決するため、ネームサーバはネットワーク内で一意に識別できるグローバルユーザIDを管理し、ユーザは、自分のグローバルユーザIDと利用する各ワークステーションのローカルユーザIDのマッピング情報を登録する。これにより、ユーザは、ネームサーバを利用して自ワークステーションのローカルユーザIDを目的のワークステーションのローカルユーザIDに変換することができ、他のワークステーションの資源を利用できる。

(3) サーバ間通信と探索制御

クライアントは情報がどこにあるかを意識しないでネームサーバへ要求を出すので、要求されたネームサーバに情報があるとは限らない。したがって、要求を処理できるネームサーバへ要求の転送処理を行う必要がある。

OSIディレクトリではモデルの動作（要求をクライアントのサーバで実行できないときの動作）として、紹介(Referral)、連鎖(Chain)、多重紹介(Multicast)を定義している。しかし、連鎖の場合、1要求が複数ネームサーバを経由するため、同時に複数通信路を占有する。また、結果が大量データの場合、転送に時間がかかるなどという問題がある（図2. a）。そこで、図2. bのように、結果を得られたネームサーバから、直接、クライアントに結果データを送信するという探索方法を追加した。具体的には、以下の動作を基本として処理する。

クライアントとサーバまたは、サーバとサーバ間の通信において、実際にデータの送受信が行われる時だけ通信路を確立し、送受信が終了後、直ちに通信路を解放する。要求を他のサーバに転送したサーバは、以後、この要求に対する処理はしない。また、クライアントは、任意のサーバからの結果を待機することにより、結果の得られたサーバは直接クライアントに結果を返すことができる。結果を返すサーバが複数であった場合、クライアントは各サーバからの結果を繰り返し受信する。

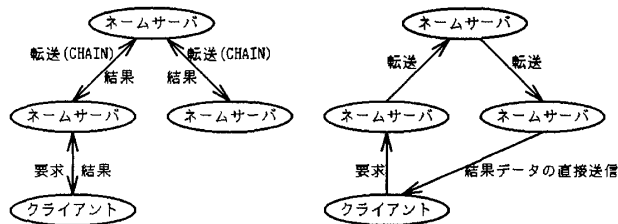


図2. a

図2. b

図2. 探索方法

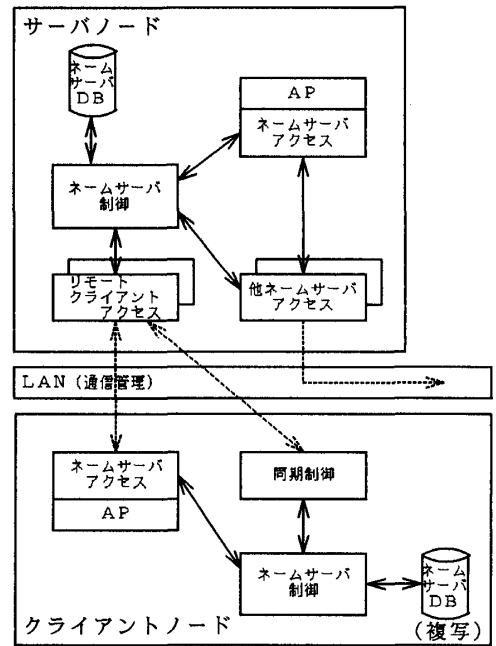


図3. 処理構成

(4) 性能向上と負荷分散

ネームサーバがリモートにある場合、ネームサーバへのアクセスは常に時間のかかるリモート通信となる。これに対処するため、クライアント側でカレントネームサーバのデータの複写を持ちアクセスできる方式とした。この方式は、ネームサーバへの要求集中を防ぐことができるうえ、ネームサーバが起動していない場合もカレントドメイン内を検索でき、システム全体の性能向上ができる。（図3）

5. おわりに

本稿は、主に、オフィス内の比較的小規模な分散処理システムにおけるネームサーバの処理方式について述べた。今後は、運用操作性や性能向上などの改良を図り、さらに、信頼性向上のための機能の検討を進めていく予定である。

参考文献

[1] JIS : “開放型システム間相互接続の基本参照モデル”, JIS X 5003, 1987.
 [2] “ローカルエリアネットワーク (LAN) に関する調査報告書”, 日本電子工業振興協会, 1988.
 [3] 岩井 他: “部門内処理向け分散RDB管理システム”, 情処40回全国大会, 1990.