

分散名前管理システムの構成法

6Q-5

の比較と検討

古宇田 フミ子 田中 英彦 元岡 達

(東京 大学 工学部)

1 はじめに

分散処理系の名前管理方式を考察している。OSIの参照モデルの上位層からローカルエリアの範囲で、ローカルエリアに存在するプロセスの名前管理を扱い、アドレスについては、考察外としている。

計算機資源の呼称はその管理方式により独自の名前付けが行なわれている。計算機を相互結合して資源の共有を図ろうとすると名前付けの差異は利用の隘路になりかねない。異なる名前体系に渡って処理可能な動的名前管理方式が必要になるが、これについて構成案を示したので、本文でこの検討を行なう。

2 分散名前管理の構成法

名前管理方式は、名前の捉え方と、分散処理系とを考慮して構成した。

ここでの名前の捉え方は「研25」にも述べたが、対象を一意に識別するための識別名と対象の属性の一つを表わす意味付け名との二面から捉え、異なる意味付け名を組み合わせで記述名とした。

対象を一意に識別する範囲を限定し、対象を示す名前の集合としてnaming domain(以下、n. d. と略す。)を導入し、名前管理の基本とした。n. d. の利用者を明確にし、使用目的に応じて、名前体系を変えることができるようにした。

一方、分散処理系では、情報の伝達遅延や不着の可能性などから、処理に関してはホストの自律性(site autonomy)、データの冗長性や保護機能の充実が必要となる。

これらを考慮して、a) ①ホスト内の全プロセスを一意に識別するためのn. d. (Pnd)と②網で情報交換するために共通に識別するためのn. d. (Cnd)の2つのn. d. を基本とし、利用者の要求に応じてFnd、Vnd、Undなどの共通n. d. や個別n. d. を設け[全30]、b) n. d. の要素間には「同じ対象を指す」という関係があるので、この関係を、識別名から対象へはPndの識別名経由、他の識別名へはPndまたはCnd経由として写像関係を表1のように作った。

3 名前管理の処理方式

名前管理は動的名前付け処理と静的名前管理の二種類の処理が必要になる。動的な面では、名前のn. d. への登録(register)とその名前の対象への結合(bind)、名前の更新(これには、資格の変

更、名前から対象への結合が変わる場合や名前の付け替えがある。)や対象が無くなることによる名前の削除の管理がある。静的な面では定常の名前変換、例えば利用者の識別名を他の利用者の識別名に変えることや、記述名を識別名にすること、またこれらの名前から対象に行くことがある。動的名前処理は利用者と係わりが深いので利用者にも見えるが、静的なものは見えない。

名前と対象が登録、結合済の状態(静的)の名前変換の様子を図1に示す。名前管理では利用者の名前を網を通して伝達できるよう通信系の名前に変換し、これを更に他の名前体系に変えている。

動的名前処理の一例として、名前結合アルゴリズムの概略を述べる。名前の対象への結合はプロセス間通信とのからみで両方向とする。

①利用者Rは相手プロセスの識別名(この利用者が識別する予定の名前。)、記述名、ケイパビリティ、使用n. d. を名前管理プロセス(nap1)に示し、結合の要求を出す。②nap1は識別名の一意性を調べn. d. に登録する。利用者のCndの識別名が無ければ登録する。相手の記述名の登録の有無を調べる。③相手プロセスの存在ホストの名前管理プロセス(nap2)にこの旨を知らせる。④nap2はRの記述名とCnd識別名を登録し、提出された記述名を基に相手の存在の有無を調べる。あれば結合要求を知らせる。⑤nap2は結合可能ならば相手のn. d. にRの識別名を登録し、n. d. の要素間の写像を作る。結果をnap1に返す。⑥nap1は相手のCnd識別名を登録し、Rのn. d. などの間の写像を完成させ、Rに結果を知らせる。(図1)

表1 写像g

写像の名前	行き先	写像のタイプ
dm1	descriptive name -> pnd	<g2>
	pnd -> descriptive name	<g1>
dm2	descriptive name -> cnd	<g2>
	cnd -> descriptive name	<g1>
nm	snd -> cnd	<g1>
	cnd -> snd	<g1>
psul	pnd -> und	<g2>
und内	und -> pnd	<g1>
culr	cnd -> und	<g2>
und内	und -> cnd	<g1>

4 考察

提案方式には次のような特徴がある。

- (1) 相対名前付け方式を採用した。利用者には各々に応じた名前付けが可能となり、異なる名前体系に渡る名前管理が容易になった。
- (2) 名前管理は分散向きの構成である。「同じ対象を表わす」名前間の変換をCndの識別名、すなわち網での識別名を媒介として行なっているので変換回数は高々4回で済んでいる。
- (3) 相対名前付けでは名前の変換をするので「名前が正しい対象を指す。」ことの信頼性が問題となり得るが、提案方式では対策を幾つか設けた。

具体的には、名前の結合を両方向にしたこと、動的な名前処理のアルゴリズムではホスト間の通信を一往復で済むようにしたこと、更新処理には、保護機能(A-listとC-listによる制御)を働かせていること、名前の登録と結合の状態を分けたことなどである。

- (4) 動的な名前付けに対し、網の不信頼性に対する保護機能を設けた。

名前の登録や結合を行なうことは、利用者で使用許可を与えていることと同等であり、名前形式を変換することはアクセス可能とすることであるから、結果的に保護機能が強化されている。対象と名前の結合関係が変わる時や新たに生成する時などはこの保護機能が威力を発揮する。更新中は、分散系であることから伝達遅延による情報不一致が問題となり得るが、n, d. を用い名前に資格チェックを入れていることや名前変換を行なうために写像があることなどから①対象にアクセスする直前ではA-listによるチェックを受ける。②それ以前の名前変換時に写像が切れる。③名前自身に更新中のフラグがあり確認できること、から不正アクセスに対する安全性は高い。

- (5) 拡張性。提案方式の現在のモデルは一つの網について考えているので、Cndは1つであるが、異なる網体系がある場合はそれ用のCndを増やすことで容易に拡張ができる。

5 比較

用語の定義はOSIで用いられている名前とアドレス[084]に近い。ここでの考察は、そのようなOSIの名前とアドレスの実装方式に関わるものである。例えば、OSIでは名前の一意性の範囲についての記述はないが、提案方式ではn, d. を用いることでこれを明確にし、かつ異なるn, d. を利用することで、同じ名前で別の対象を指すことや、別名の管理を容易にした。

絶対名前付けと相対名前付け。一例として、ゼロックスのClearinghouse [C83] (以下CHと略す。)

は一様性を重視し絶対名前付けを採っている。絶対名前付けは集中方式向きの構成であるが、CHでは名前を3回層に構成し、網にも適用できるように工夫している。これは提案方式におけるCnd一つだけの構成に対応している。Cndの識別名は、網の情報を持ち、名前形式の中に管理構造が入っている。n, d. を複数設ける方式では名前には管理構造のないn, d. が作れるので、利用者に名前の自由形式を提供することができる。

6 おわりに

分散処理系向きの名前管理の一方式について、異なる名前体系に適用できる相対名前付けに関して簡単な比較と考察を行った。

分散系で、しかも異なる名前体系を実際に用いて実験するのは困難なので、名前関係やその構造の基本的部分についてPascalを用いてアルゴリズムの正当性や効率、オーバーヘッドなどをシミュレート中である。プログラムの大きさは4500行程度である。

参考文献

- [研25] マルチメディア通信と分散処理研究会 25-5 (1985.5.16)
- [全30] 情報処理学会第30回全国大会 7P-2 (1985.3.15)
- [084] 今本善信 名前とアドレスアドホック会議出席報告 SC21-J-84-22(1)
- [C83] Oppen, D.C., Dalal, Y.K. : The Clearinghouse Information Systems Vol.1, No.3 July 1983 pp230-253

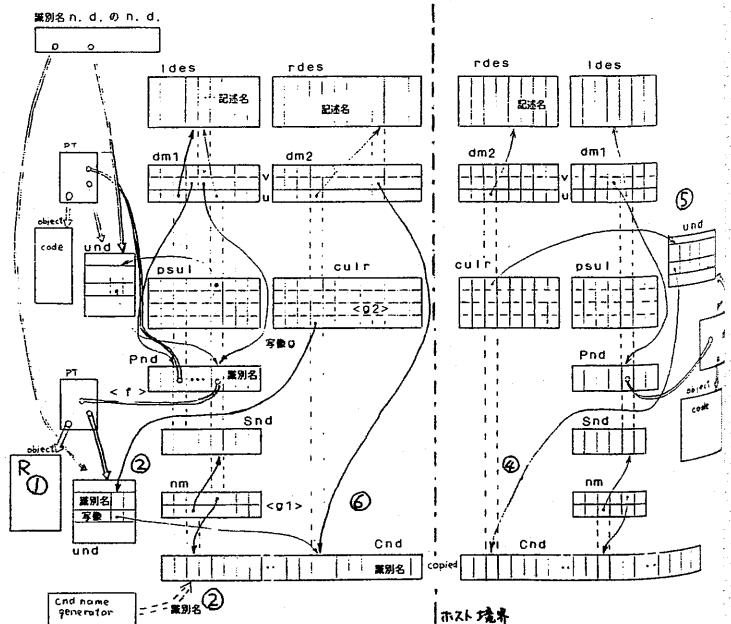


図 1 名前の概念的捉え方