大規模分散システムの集中運用管理における効率化技術の提案

敷田 幹文　井口 寧　丹 康雄　松澤 照男

北陸先端科学技術大学院大学 情報科学センター

近年，組織内ネットワークが急速に発展し，計算機システムが大規模化・分散化している．このようなシステムでは，分散配置された大量のクライアントとこれに対してサービスを行う大型サーバを持ち，運用管理は集中化している．そのため，従来の分散管理システムとは異なる管理方法が要求される．ここでは，北陸先端科学技術大学院大学の情報環境システムを例に，大量のクライアント機と大型サーバ機に起因する問題点を述べ，我々が実現した集中型運用管理のための効率化技術を紹介し，その有用性に関する議論を行う．

Efficient Management Techniques for Large-scale Distributed Systems

Mikifumi SHIKIDA　Yasushi INOGUCHI　Yasuo TAN　Teruo MATSUZAWA

Japan Advanced Institute of Science and Technology

Recently a number of machines on a network are rapidly increasing, and many sites have a large number of client machines and a lot of large-scale high performance servers. However it requires many cost to manage these large-scale distributed systems. In this paper, we propose some approaches and techniques to control these system efficiently, then discusses its effectiveness. We also implement these approaches in our university.

1 はじめに

近年，組織内ネットワークに接続される計算機の台数が急激に増えている．これは，計算機の高速化，低価格化だけでなく，個人向け計算機のオペレーティングシステムのネットワーク対応が進み，非専門家向けアプリケーションが増加したことにも関係している．

しかし，ユーザ層の広がりと計算機台数の増加は，ユーザと管理者の比を大きくし，また隔たりを広げることにもなった．さらに，大量のクライアント計算機に対してサービスを行うために，サーバ機やネットワークは大型化・高速化が進んでおり，これらの運用を管理する計算機・ネットワーク管理者の負担は年々増大している [5]．負担が増大する主な理由として，１台に対してはわずかな作業でも大量の機械に行うと膨大な作業時間を要し，場合によっては従来と同じ方法での作業が不可能なことがあげられる．即ち，これまで行われてきた管理作業を時間的により効率よく行うことが重要課題となっている．

我々の大学では，1990年の創立以来，次世代の情報環境を目指した計算機・ネットワークを設計し，実現してきた．そのために，教官及び学生はもちろん事務部門に至るまで，一人１台に近いワークステーションを配置し，これらを超高速のネットワークで接続している．また，各種サーバ類は一ヶ所に配置して全学に対してサービスを行っている．これらのクライアント，サーバ，及びネットワークは，集中管理することによって超大規模分散計算機システムを構成している [4]．しかし，我々のシステムは巨大であるため，これを少人数の教職員で運用管理するのは困難である．そのため，我々は効率的に管理するために各種の新しい試みを行っている．

本論文では，我々が設計し，運用管理している本学の情報環境システム (FRONTIER と呼ぶ) を例に，大規模分散システムを運用管理するための問題点として，大量のクライアント機と大型サーバ機に起因するものをあげ，これらに対して我々が実際に行って，管理作業の効率化に成功している要素技術について述べる．また，類似の製品・方法と比較することにより，本論文のアプローチの有用性を示す．

2 FRONTIER の概要

"FRONTIER" は北陸先端科学技術大学院大学の情報ネットワーク環境の総称であり，具体的には，個人用計算機，ファイルサーバ群，計算サーバ群，その他の各種サーバ群，および，これらを接続するネットワーク等からなる．

2.1 FRONTIER の規模

FRONTIER は最初の学生が入学した 1992 年 4 月に合わせて稼働を開始したが，それ以来，今日までのシステムの規模を表 1 に示す．表中のホスト数は NIS の hosts のエントリ数から DHCP 等を除いたものである．これは学内の全ての固定ホストを記述しているが，そのうち集中管理しているホスト数を括弧内に内数で示す．サブネット数は，バックボーンネットワーク等を含まない．また，ユーザ数は，非常勤職員・卒業生等を除いた．管理者数は，情報科学センターで運用管理を行っている教職員の数である．

表 1: FRONTIER の規模の年変化

年度	ホスト		サブネット	ユーザ	管理者
1992	340	(300)	20	190	1
1993	800	(650)	50	490	1
1994	1,000	(800)	60	680	2
1995	1,700	(1,000)	90	810	4
1996	2,600	(1,150)	100	880	4
1997	3,200	(1,220)	110	920	6
1998	5,000	(1,500)	120	1,090	6

2.2 FRONTIER の各種サーバ

FRONTIER では，一般ユーザへ提供する主要なサービスのほとんどを一ヶ所のサーバで対応している．管理するサーバの台数が少なくなるため，機械の大型化によって一台当たりの管理コストは増えるが，少人数で管理するのであれば総合的には管理コストが減る．

現在の FRONTIER で提供しているサービスの主要なものは，名前情報 (NIS, DNS)，電子メール，ネットニュース，WWW，Proxy 等である．また，ユーザのファイルもほとんどを集中管理している．これ以外に，プリンタ共有など各フロアに依存したサービスには，それぞれのフロアに分散配置したサーバを利用している．

2.3 FRONTIER のネットワーク

大規模に分散したクライアントに対して中央のサーバでほとんどのサービスを行うと，サーバとクライアントのネットワーク上の距離が広がる．そのため，小規模システムと同様のレスポンスを得るためには，組織内に超高速のネットワークを張り巡らせる必要性が生じる．

FRONTIER のネットワーク (FRONTNET) では，高速ネットワークを実現するために，レイヤー 3 ルーティングを行う高速 IP スイッチを 3 台導入し，これらから学内のほとんどのフロアに対してそれぞれ FDDI または 100BASE-TX のケーブルを敷設している．この 3 台のルータは 800Mbps の HIPPI インタフェースも備え，HIPPI スイッチによって互いに接続されている．また，各フロアでは，レイヤー 2 スイッチやハブを通して各クライアント機に接続している．この接続の概略を図 1 に示す．図中の右半分は旧世代のネットワークであり，現在は障害時に自動的に切り替わるバックアップ用に用いられている．

ファイルサーバ等のアクセス頻度の高いサーバ類は 3 台の IP スイッチに専用のポートで接続されている．即ち，学内のほとんどのクライアント機からは，2hop ないしは 3hop で各ファイルサーバにアクセス可能で，その経路は全てスイッチになっている．

3 大規模システム管理の問題点

前節で紹介した FRONTIER のように大規模な分散システムの運用管理を行うには，小規模なシステムとは異なる問題が発生する．この節では，これらの問題点について述べる．

3.1 クライアントの管理コストの増加

大規模分散システムではクライアント計算機が大量に導入される．これは台数が多いため，1 台あたりの管理コストの増減が全体のコストに大きな影響を与える．

FRONTIER では，1,000 台以上の UNIX ワークステーションと 500 台近い Macintosh を管理している．例えば，UNIX ワークステーションの約 9 割を占める，デスクトップ型 Sun ワークステーションでは，毎年新規導入される機械が 200-300 台程度ある．また，毎月ハードウェア修理を必要とするものが 5-10 台程度あり，これ以外のソフトウェア的トラブルも多い．そのため，これらを初期イ

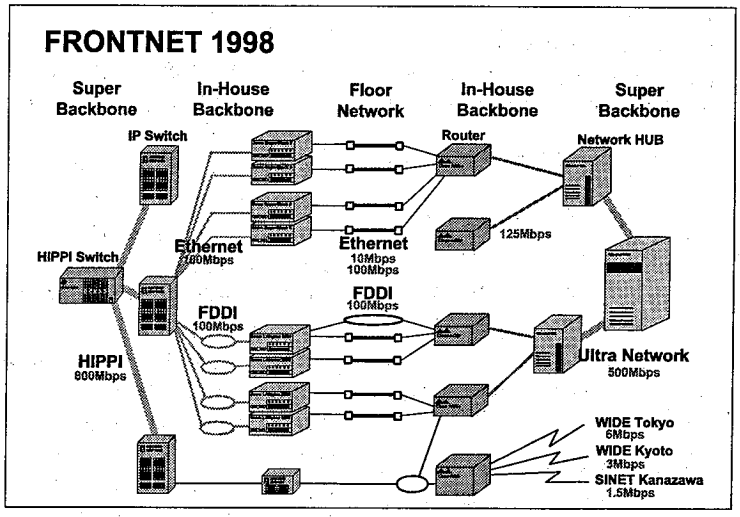

図 1: FRONTIER のネットワークの基本構造

ンストール，再インストールするための作業には極めて大きなコストを要する．

インストール作業は，単に必要なソフトウェアを内蔵ディスクに書き込むだけでなく，接続場所に合った各種設定を行うことを含む．このような作業は，故障時以外に機械の移動時にも必要になる．

3.2 サーバの大型化と依存度の増加

ファイルサーバ，電子メールサーバ，ネットニュースサーバ，WWW サーバ等，各種のサーバがあるが，いずれもクライアントの台数に見合った大型サーバが必要になる．アクセス量が少ない場合には無視できたシステムの弱点が，アクセス量が限度を越えたために重要な障害となることもある．

クライアント機の管理コストを下げるために，クライアントがアクセスするファイルの大部分をファイルサーバ上に配置することも考えられる．そのためには大容量の二次記憶装置が必要になり，ディスクドライブが多くなるために故障率も増加する．

また，大量のクライアント機に対して少数のサーバ機でサービスを行うので，1 台のサーバに障害が起きても多数のユーザに影響をおよぼす．

4 効率的管理方法

ここでは，前節で明らかにした問題を解決し，効率的な運用管理を行うために，大規模分散システムが備えるべき特徴を述べ，我々が実現したシステムについて述べる．

クライアント ハードウェア/ソフトウェア共に可能な限り統一を行い，初期設定や再設定のコストを可能な限り小さくする機構を備える．

サーバ 高信頼性・高可用性のための機構を備える．

4.1 クライアントワークステーションの管理

我々は，クライアント機のインストールコストの問題を解決するために，大部分を占める Sun ワークステーションの全自動インストール機構を実現した．この機構を “JaistStart” と呼ぶ．

この機構は，対象となる機械をディスクレスクライアントとしてインストールサーバからブートし，起動後のスクリプトで予め用意してあるディスクイメージを内蔵ディスクに書き込むものである．また，MAC アドレスやその他インストールに必要な情報は NIS 等に書かれており，イメージ書き込み後のスクリプトで各機械固有の設定も全て行い，最後に内蔵ディスクから起動して，そのままユーザが使用可能な状態にする．作業手順を図 2 に示す．また，この作業を行った後のクライアント機の振る舞いを図 3 に示す．

図 2 からもわかるように，この方法では，最初に ROM モニタ上でネットワークからの起動を指示するコマンドを打つ以外にクライアント機上での

1. クライアント機のIPアドレス，ホスト名，MACアドレスをNISのhostsとethersに書き込む
2. クライアント機のOSやアーキテクチャに応じたブートイメージの場所を，NISのbootparamsで指定する．
3. RARPサーバで，クライアント機のIPアドレスに対するブートファイルを指定する．
4. クライアント機のROMモニタで，ネットワークブートのコマンドを実行する．

図2: 全自動インストール機構の作業手順

1. ネットワークブート(RARP, TFTP)
2. /etc/rc等にて，内蔵ディスクが初期化されていなければ，ラベル書き込みと各パーティションの初期化を行う
3. 初期化されていれば，次の処理を行う
   (a) 内蔵ディスクの各パーティションを初期化する
   (b) 各パーティションのイメージをリストアする
   (c) ブートブロックの書き込み
   (d) ホスト名とIPアドレスからサブネットを調べる
   (e) そのフロアのプリンタ等を登録する
4. 内蔵ディスクからリブートする

図3: 全自動インストール機構の動作

操作を何も必要としない．これによって，多数の機械のインストールを行う場合には，開始と同時にその場を離れて次の機械に移ることが可能になるため，多数のクライアントをインストールするには大きな利点となる．また，現場での作業が極めて単純なため，場合によっては非管理者にこの作業を代行してもらうことも容易である．

個人用ワークステーションは，管理コストの削減やユーザからの要望によりほとんどが同一メーカになっており，OSのバージョンも統一している．しかし，今年度から異なるバージョンのOSや異なる機種もある程度の数量が導入されたため，現在は以下のOSと機種で稼働している．

- SunOS 4.1.4 (Sun SPARCstation)
- Solaris 2.6 (Sun SPARCstation, Ultra5 等)
- IRIX 6.3 (SGI O^2)

それぞれの機種やOSの違いにより，JaistStartの構造にもある程度の差がある．しかし，基本的な仕組みは変わらず，多くのUNIXワークステーションのインストールについて同様な機構の実現が可能であろう．

4.2 ファイルサーバの管理

クライアント機に対するNFSファイル共有サービスは，FRONTIERにおける最も重要なサービスの一つである．これは，クライアント機の管理コストを軽減するために，各ユーザの個人ファイルを収めるホームディレクトリや，共有アプリケーションの"/usr/local"などを，ほとんどファイルサーバ上に置いていることにも起因している．

現在のFRONTIERでは，各ユーザのホームディレクトリや共有アプリケーションのために3システムのファイルサーバで計600GBのディスクをサービスしている．また，ホームディレクトリ以外の目的として，700GBのディスクをサービスするシステムと，1TBの磁気テープによる大容量記憶装置のシステムがある．なお，これらのファイルサーバは毎年一部を機種更新している．

これらのファイルサーバの運用経験から，次のような問題点が明らかになった．

- ファイルサーバへの依存度が大きいために，障害が発生すると全学の多数のユーザに影響を及ぼす
- 負荷を分散し，各サーバや各インタフェースを均等に利用するのは難しい

これらの問題を解決するために，以下の3つの対処を行っている．

高可用性システム 個々のファイルサーバのディスクはRAID機構，ホットスペア機構を備え，活性挿抜が可能で，各種のインタフェースや電源等も二重化している．

また，特に影響が大きいアプリケーションファイルのサーバは，高可用性システムを採用している．これは通常2台のホストでサービスを行うが，一方の障害時には他方が2台分のサービスを行うことを可能にしている．ディスクやIPアドレスなども全て肩代わりするため，ユーザは数秒から数十秒待たされるだけで復旧したように思える．

この機能があると，サービスを中断することなくいつでもサーバを停止できるため，障害時のみでなくOSのパッチなどのメンテナンス時にも利用できる．FRONTIERでは24時間サービスを行っているため，この機能が管理スケジュールの自由度に大きく貢献している．

バックアップ 前述の安全設計にもかかわらずファイル内容にダメージが及んだ場合のことを考え，定期的にバックアップを行っている．大容量のデ

ィスクをバックアップするには大量の磁気テープを必要とするが，深夜に無人でバックアップを行うために，主要なファイルサーバのそれぞれには合計1TB以上となるオートチェンジャを備えている．

高可用性システムではハードウェアのトラブルを回避することができるが，ソフトウェアトラブルの場合には回避できないことも多い．FRONTIERの場合にはファイルサーバの負荷がとても高く，他のサイトでは発見されていなかったファイルシステムやデバイスドライバの不具合が露見したことも何度かあり，ユーザのファイルが破壊されたこともあった．そのような場合にはテープへのバックアップが必要となる．

負荷分散機構 ファイルサーバは複数のネットワークインタフェースを持っているものが多いが，最適なインタフェースはクライアントのネットワーク内の位置によって異なる．しかし，現在のNFSマウント機構では，最適なインタフェースを自動選択することはできない．そのため，我々は最適なインタフェースに関する情報をNISを利用して配布し，クライアントがこれを利用してマウントする機構を実現した．

この機能は次のような手順で実現されている．

前処理1 ネットワークトポロジーに基づいて，各サブネットから各サーバのどのインタフェースをアクセスすると最適となるかを決め，その結果をNISのマップに登録しておく．

前処理2 クライアントのautomountデーモンが参照するNISマップの各エントリで，サーバホスト名の第1候補を変数にしておく．
例: `home01 $FS1,fs1-f0:/home/home01`

クライアント クライアント機のブート時，automountデーモンを起動する前に，自分のサブネットアドレスを用いてNISの最適インタフェーステーブルから検索し，結果を環境に入れてautomountデーモンを起動する．
例: `automount -D FS1=fs1-f1`

なお，ネットワーク障害時には最適経路が変化するため，より正確に行うには最適インタフェーステーブルも動的に変更しなければいけない．しかし，一旦NFSマウントすると経路が変わってもアンマウントされないので，動的な変更を反映してマウントするインタフェースを変更するのは困難である．また，負荷分散するように自動制御することは難しいが，静的に記述することによって特定個所への集中を避けるような制御も可能になる．これらの理由により，現在は動的な自動変更を扱ってはいない．

5 評価

ここでは，我々のアプローチの有効性について，類似の製品・方法などとの比較を行い，本アプローチの利点と問題点を明らかにする．

5.1 クライアント管理

近年のオペレーティングシステムのインストールは，グラフィカルユーザインタフェースを用いて容易なインストールを可能にしているが，これは個人向けを考慮したものであり，対話的に設定を行う方法は大規模システムの管理に適していない．

Sun Microsystems社の現在のオペレーティングシステムには，Custom JumpStart[2]という自動インストール機構が備わっている．これは，OSのインストールの際に必要な情報を予めNISのマップなどに記述しておき，インストール後に行う作業もスクリプトで記述しておくと，完全に自動化できるという製品である．クライアント機上での作業がネットワークブートのコマンドのみでよいという点では我々の方法と全く同様である．しかし，これはあくまでもインストールメディアからのインストールであり，インストール後にパッチの適用やカスタマイズを行うので時間がかかる．カスタマイズを施した後のイメージを用いる我々の方法では，SPARCstation5にSunOS4をインストールする場合で十数分以内に完了する．また，Ultra5にSolaris 2.6のほとんどのパッケージを入れた場合でも35分程度であった．同じ構成のインストールを通常の方法で行った場合には1時間以上を要した．1台のインストールに必要な時間が2倍になると，大量のクライアントを一度に設置する場合には全体の作業計画に大きな影響を及ぼすことになる．

斎藤らのシステム[6]では，我々の方法に近い半自動初期設定の他に，定時自己診断を行っている．これは一定時間毎に自分自身を診断し，設定が正しくなければ修正する機能である．このシステムのクライアントは大学の演習室の端末であるため，厳密に画一化された環境を作ることは重要である．しかし，我々の場合は，ユーザは教官や大学院生がほとんどであり，各ユーザは自分専用の機械として自分の席に設置し，研究のためであれば統一環境とは異なる設定にすることも許されているため，このような自己診断・自動修正機能

を持たせることができない．そのため，我々は，インストール時の自動化と再起動時の自動診断のみを行っている．また，Zero Administration Kit for Windows95[1]も同様に，統一環境を強制する目的の製品である．

5.2 ファイル共有

広域のファイル共有機構として，カーネギメロン大学で開発されたAndrew File System (AFS)[3]等がある．米国の大学の計算機センターなどで大量のクライアント機と大勢のユーザを持つシステムではAFSを採用していることも多い．しかし，これは広域で低速のネットワークを想定したファイルシステムで，NFSに比べると性能が悪く，ファイルの書き込みは数倍の時間がかかる．大勢の計算機初級ユーザの演習用途であれば問題は少ないが，大学院生の研究室内での利用に適しているとは言えない．

このため，我々のアプローチでは，大規模ネットワークにそのためのファイルシステムを導入するのではなく，ファイルシステムは小規模の共有の場合と同じNFSで，ネットワークの方を高速・低レイテンシにすることによって，大規模であってもグループ内サーバにアクセスするのと同程度の性能を出せるようにした．

5.3 負荷分散機能

複数のサーバや複数のインタフェースで負荷を分散させる方法がいくつかある．例えば，DNSを利用したIPアドレスのローテーションや，ルータでMACアドレスを書き換える方法はよく知られている．

しかし，組織内部で使うNISではローテーションを頻繁に行うことに適していない．ローテーションを行っても，NFSマウントは継続時間が長いのでスムーズなローテーションにならない．また，ルータを用いると，そのルータの入り口は一ヶ所であるから，ファイルの書き込み時にはここがボトルネックになる．これらの方法は，短いコネクションが大量に発生し，書き込みよりも読み出しが圧倒的に多い，WWW等の場合に適していると言える．

また，経路情報を制御することにより，インタフェースの負荷分散を行う方法もある．この方法では，ネットワーク構成の変化で最適経路が変わっても自動的に変更される．我々の方法では自動変更されないが，そのように大きな構成変更が起こることは稀れである．また，静的に記述することによって，意図的に最適でない指定を行って負荷を分散させたり，クライアント側の特性によって優先順位をつけるなど，管理者側の方針を反映させやすいという利点がある．

6 おわりに

本論文では，大きな組織内に分散した，大規模分散計算機システムを少人数で効率的に管理するためのアプローチを提案し，特に，大量のクライアント機と大型サーバ機を効率的に管理するための要素技術を述べた．近年のクライアント機の増加やサーバ機の大型化は，分散システムの大規模化をさらに押し進めることが予想される．そのような組織では従来と同じ体制では管理作業に要する時間が増え過ぎて安定した運用に障害を来たすと思われるが，我々のアプローチを適用することにより，管理コストの大幅削減が見込まれる．

現在の大規模分散システムのサーバのほとんどはUNIXをオペレーティングシステムにしているが，Windows NTのサーバ等も組織規模の大型サーバとして対応してくることが予想される．今後は，そのような新たなプロトコルやサービスに対応した要素技術が必要である．

謝辞

本研究を進めるに当たって，初期のFRONTIERの設計者でもある篠田陽一博士には活発な議論や貴重なコメントを頂きました．また，日常の管理業務では情報科学センター技官の方々にご協力頂いております．ここに深く感謝致します．

参考文献

[1] Microsoft. *Zero Administration Kit for Windows95*, 1997. http://www.microsoft.com/japan/win95/ZAK/.

[2] Sun Microsystems, Inc. Solarisのインストール(上級編), 1997.

[3] Transarc Corporation. *AFS Systems Administrators Guide*. FS-D200-00.10.3.

[4] 敷田幹文, 篠田陽一. 分散システムマネジメント. 日本ソフトウェア科学会 チュートリアル, Sept. 1997.

[5] 斎藤 明紀他. 多人数教育計算機環境におけるシステム管理の省力化の一方法. 分散システム運用技術 研究報告, No. 6, Jul. 1997.

[6] 斎藤明紀. 大阪大学情報処理教育センターのシステム構築 5. *UNIX MAGAZINE*, pp. 75–82, Feb. 1997.