第 1 回 クラスタゼミ

ゼミ担当者　：　斉藤宏樹，釘井睦和，勝崎俊樹
指導院生　　：　谷村勇輔，児玉憲造，片浦哲平，下坂久司
開催日　　：　2002 年 4 月 18 日

ゼミ内容: クラスタとは何か理解し，クラスタについての基本的な知識をつける．また Linux を利用した並列クラスタシステム，Beowulf クラスタについて学ぶ．さらに，本研究室にあるクラスタについて性能や仕組みを把握する．

1　はじめに

クラスタ（cluster）とは，英語で「ブドウの房または星団のような集合」を意味する．
クラスタをコンピュータシステムに当てはめた PC クラスタとは，一般的に使用されているパーソナルコンピュータ（PC）をネットワーク結合して構築した並列計算機の 1 種である．つまり，PC がネットワークによってぶどうの房のように群れをなしているのである．この PC クラスタが最近非常に注目されている理由として，次のようなことが挙げられる．

- PC の性能が飛躍的に向上してきた点
- オープンソースやフリーウェアなソフトウェアの利用によりコストパフォーマンスが非常に良いという点

2　クラスタの定義

Pfister によるとクラスタの定義は「クラスタは，単一で稼動するコンピュータの集まりで，一つの計算資源として使用可能な並列もしくは分散システムである」とある．

また，市場調査会社である Aberdeen Group の定義によれば，クラスタ・システムとは

- 複数ノードで構成されたコンピュータ・システム
- 全体として
  - 単一システムとして機能すること
  - 高可用性を有すること
  - クラスタ全体に対するシステム管理機能を有すること
  - クラスタ・ファイルシステムを有すること
  - スケーラブルなプラットフォームをサポートすること
  - 柔軟なシステム構成を構築できること

これらの定義の中で，クラスタが単一で稼動するコンピュータの集まりであるということは重要なことである．つまり，クラスタを構成する各ノードはコンピュータの最小構成である CPU，メモリ，および OS などを必ず有しているということである．

一般的にＰＣクラスタという表現は次の３種類のシステムのいずれかを意味する．

- 並列演算クラスタ（HPC:High Performance Computing）
  主に科学技術計算で利用される並列プログラミング・アプリケーションを使用するためのクラスタ．スーパーコンピュータと同等の処理能力を低コストで実現する．Beowulf クラスタは典型的な実例．
- 高可用性クラスタ（HA：High Availability）
  ミッション・クリティカルなアプリケーションを実行するためのクラスタ．クラスタは冗長化構成を持ち障害発生時には切替える。フォールト・トレラント・コンピュータの適用分野でシステムをより低コストに実現できる．
- 負荷分散クラスタ（LB：Load Balancing）
  単一のアプリケーションを負荷分散して実行するためのクラスタ．ネットワーク・サービス・アプリケーションの実行に適している．

3　クラスタの必要性

本研究室では多くの人が，最適化問題を解くための手法やその計算モデルについて研究している．最適化問題を解くには膨大な計算を行うのだが，1 台では CPU 資源の不足やメモリ不足で解けない，結果を得るのに多くの時間を要するような問題がクラスタを用いた並列化によって可能になる．また，同レベルのパフォーマンスを持つスーパーコンピュータに比べてコストが安いということも挙げられる．これらの観点から見てもクラスタを用いることが必要になってくる．

4 Beowulfシステムの概要

4.1 Beowulfとは

Beowulfとは，ネットワーク技術によって相互接続されたPCクラスタである．UNIX系オペレーティングシステム上で，MPIなどのメッセージライブラリを使用して並列計算を行う．

最近まで，大衆市場PCで使用されるマイクロプロセッサの性能と，高価な科学ワークステーションで使用されるマイクロプロセッサの性能の差は極めて大きかった．しかし近年，この二つのクラスのマイクロプロセッサの能力差が劇的に収束し，今日ではそのようなギャップは無くなってしまった．この流れを利用して，NASAとカリフォルニア工科大学のチームが共同で安価なPCを開発し，大量につなげることによりスーパーコンピュータ並みの計算速度を得ることに成功した．この時の，複数のPCをネットワークで接続し，低コストで科学技術計算用コンピュータ（スーパーコンピュータ）を実現しようというNASAのプロジェクト名がBeowulfであった．

Beowulfは複数のLinuxコンピュータをクラスタ化し，並列仮想スーパーコンピュータを形成する技術をいう場合が多い．何故ならBeowulfのオペレーティングシステムには，ソースコードが付加コストなしで広く入手可能なLinuxを用いる場合が多いからである．

4.2 Beowulfシステム

Beowulfシステムは，一つ以上のスレーブノードと一つのマスターノードがあり，それらをイーサネットなどのネットワークで一つに接続したシステムである．Beowulfシステムを構築するのに用いる部品は，Linuxが動かせる任意のPCや，標準イーサネットアダプタとスイッチなどの一般的なハードウェア部品である．Beowulfは，特注ハードウェアを全く使わないで容易に構築することができる．

また，Beowulfシステムで用いるソフトウェアは，Linuxオペレーティングシステムや PVM（Parallel Virtual Machine），MPI（Message Passing Interface）などの，オープンソースなソフトウェアである．Beowulfは単一ホストからログインするクラスタである．大規模なクラスタやネットワークの負荷が高いアプリケーションを使用する場合に，ネットワークに Gigabit Ethernet や Myrinet が使われる例がある他，CPU に Alpha を用いることもあるが，ショップで手に入るパーツを使ってコストパフォーマンスを上げることが，Beowulfシステムの特徴の一つである．

Beowulfシステムの性能を十分に引き出すには，並列タスクの粒度を中規模から大規模にし，通信がそれほど多く生じないような特徴を持った，適切なアルゴリズムを使用しなければならない．何故ならBeowulfでは，コストを低く抑えることが目的の一つであるから，多い通信量に対応させることは困難である．幸いにも，多くの大規模問題がこのような要求条件を満たしている．

4.3 Beowulfシステムの利点

Beowulfシステムには，以下の利点がある．

- 低コストで高性能が得られる．
  Beowulfシステムの思想は，安価で誰にでも実現可能な高性能計算であり，低価格なPCやソフトウェアを用いて構築することが可能である．

- 技術動向に速やかに対応できること．
  Beowulfシステムは独自のアーキテクチャーがあるわけではなく，大衆市場システムを作る複数のベンダから得られる部品を使っているため，容易に最新技術をBeowulfシステムに取り込むことができる．

- 拡張性がある．
  Beowulfシステムは，拡張性がある．システムサイズは，わずか一個の低価格ハブで接続される少数台ノードから，何百台ものプロセッサで複雑なトポロジを組み込んだシステムまで，広範囲なものが可能である．

4.4 Beowulfシステムの欠点

Beowulfシステムには以下の欠点がある．

- 通信性能が低い．
  専用の高速スイッチで利用している Gigabit Ethernet や Myrinet などを使用すると，Beowulfシステムの構築が高価になるため基本的に使用しない．そのため通信性能が低くなる．

- 並列プログラムの作成が困難
  逐次に実行していくプログラムでは，通信量を考慮する必要はないが，並列に実行する場合は，通信量が多いプログラムになると処理速度が低下する．そのため，アルゴリズムを変更しなければならない場合，作成するのが困難な場合がある．

5 Beowulfシステムの構造

5.1 ハードウェア

Beowulf システム用のノードは通常，商用大衆市場から得られる優れた価格性能比を持つパーソナルコンピュータである．これは，必ずしも絶対的に最低価格のシステムを意味するのではなく，むしろ手元にある問題に対して能力やコストを最適にバランスを取らせることを意味する．Beowulfシステムに使われるハードウェアとしては，以下のものが挙げられる．

- プロセッサ

  パーソナルコンピュータに利用され，一般大衆に低コストで販売されているマイクロプロセッサには，3つの主要なファミリがある．MS-DOS，Windows の PC に利用される Intel x86 ファミリ，MacOS の PC に利用される IBM/Motorola PowerPC ファミリ，NT や Digital Unix に利用される DEC Alpha ファミリである．

  Beowulf システムで利用される OS，Linux ではこれら 3 つのファミリをそれぞれ，その目的によって使い分ける．例えば，Intel プロセッサは Beowulf クラス計算に主に使われ，Alpha は浮動小数点演算重点型計算が必要な部分に，という具合である．ただ，パーソナルコンピュータはほとんど Intel x86 ファミリなので，たいていの場合は Intel x86 ファミリが用いられる．三木研で用いられる Beowulf システムも Intel x86 ファミリの CPU を使っている．

- マザーボード

  マザーボードは，チップをただ装着するだけの便利な基盤という意味だけでない．独自の複雑な論理回路を含み，計算機システムの性能，柔軟性，有用性などに大きな効果を発揮する．これによって，ノードに組み込める最大メモリ量，メモリやコントローラなどに対するインターフェースポートなどを選択できる．また，ISA バスと PCI バスの両方を高性能装置に対してサポートしているのが一般的である．

- メモリ

  システムの性能と有用性は，プロセッサと同程度にメモリに依存している．メインメモリとして，SIMM や DIMM としてパッケージ化された DRAM 部品で構成される．メインメモリを選択する上で考慮する必要があるのは，速度，容量，バーストモード，オンチップ誤り修正などである．

- ハードディスクドライブ

  システム唯一の不揮発性記憶装置は，マザーボード/BIOS パラメータのための若干の EPROM を除くと，ほとんどがハードディスクドライブによって提供される 2 次記憶である．商用システムが SCSI ディスクと SCSI プロトコルを使用するのに対し，一般消費者用システムにはスループットの対価格費が優れた EIDE 標準を使用する．ハードディスクは，不揮発性記憶装置を提供するだけでなく，見かけ上のメモリ容量を拡張する手段としても使用される．

- フロッピーディスクドライブ

  フロッピーディスクドライブは，転送媒体としては意味を持たない．しかし，コストは安いし，初期システムインストレーションやクラッシュ回復のために重要であるので，各ノードに 1 台設置しておく必要がある．

- サポート装置

  （a）外部ローカルエリアネットワーク

  Beowulf は，多くの場合幅広いユーザを抱え，アクセスを許すようなインフラストラクチャ内で使用される．このような環境への接続は，ローカルエリアネットワークに接続された 1 個以上のネットワークインタフェースカード（NIC）を通して行われる．したがって，Beowulf システム内の複数ノードが外部アクセスを可能とするためのカードと IP アドレスを持つ必要がある．このとき，NIC のタイプは，LAN 環境と互換性のあるものを選ぶ．このとき，外部とローカルエリア内とをつなぐ入り口をマスターと呼ぶ．マスターは内部と外部とを結ぶために 2 つの NIC を持ち，外部に対する唯一の接触点となっている．なぜこのような構造になっているかというと，外部と子ノードとの接触を避けることで，クラスタ内を独立空間とし，セキュリティーを高めてやるためである．このようなことが可能なのは，外部と子ノードが直接接触する必要がないからである．その様子を Fig1 に示した．

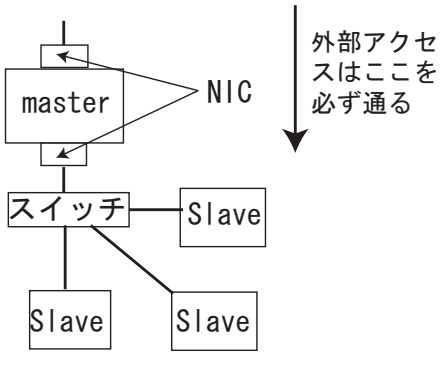

Fig. 1　master と slave の関係

  （b）*CD-ROM* ドライブ

  CD-ROM は大規模ソフトウェアパッケージ配布用の主要な媒体となっている．例えば，Linux は CD-ROM として複数のソースから入手可能である．Beowulf システム全体に対しては，1 台だけドライブがあれば十分である．

（c）モニター／キーボード／ビデオカード

必ずしも必要ではないが，大部分の Beowulf システムでは，直接のユーザインタフェースとして，モニタやキーボードを備えている．このノードは，システム管理，故障診断，統計情報提示などのために主に使用される．

5.2 ネットワーク

内部システムエリアネットワークは，持続的性能の達成や，コストへの貢献という双方の観点から，Beowulf クラスタの中で，ノードそれ自体に次いで最も重要なサブシステムである．Beowulf は，ローカルエリアネットワークとして開発され，市販されている Ethernet と TCP/IP プロトコルを使用することが非常に多い．他に必要となるものはスイッチである．

- Ethernet

  現在の Beowulf では，100Mbps の Fast Ethernet で結合されていることが多い．より高いバンド幅を求めるならば，Myrinet や Gigabit Ethernet を利用できる．

  三木研究室では，Fast Ethernet と Myrinet を利用している．

  また，Ethernet，Myrinet，TCP/IP の関係は Fig. 2 に示した．

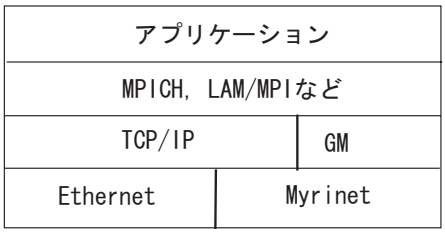

Fig. 2　Ethanet と TCP/IP の関係

- スイッチングハブ

  スイッチングハブは，ツイストペア線を用いてパケットをノードから受け取る．これらの信号はすべての接続されたノードにブロードキャストされるわけではなく，メッセージパケットの宛先アドレスフィールドが解釈され，パケットが目標ノードにのみ送られる

5.3 Beowulf クラスタの動作

Beowulf クラスタの動作は，以下のような流れで行われる．Beowulf クラスタの構造は Fig. 3 のようになっている．

master でコンパイルが行われ，実行ファイル a.out が作られる．

各 slave に a.out をコピーしなければ，並列処理はできない．そのため RSH（Remote Shell）を使う．これにより，パスワードなしでコマンドの実行やファイルコピー，ログインが可能となる．

master，slave 同時にプログラムの実行を行い，並列処理を行う．

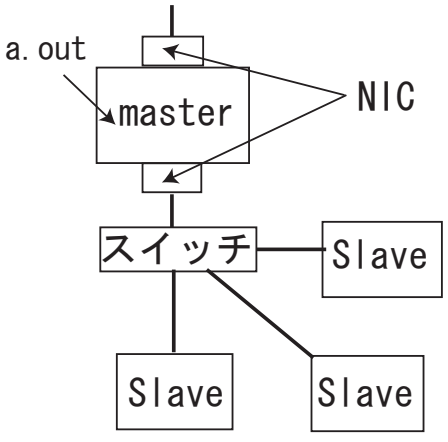

Fig. 3　Beowulf クラスタの動作

5.4 HPC クラスタを Beowulf クラスタで実現するための構成

HPC（High Performance Computing）クラスタを Beowulf クラスタを使って実現するためには，以下のような構成が必要となる．

Table 1　HPC クラスタを Beowulf クラスタで実現するための構成

ハードウェア	PC，イーサネット
OS	Linux
設定	rsh
通信ライブラリ	MPI

- rsh

  リモートシステム上で指定したコマンドを実行するためのコマンド. リモートホストへ接続し，指定されたコマンドをバッチ実行する。rsh は自分の標準入力をリモートコマンドにコピーし、リモートコマンドの出力を自分の出力にコピーする。このように、リモートコマンドを実行しても標準入出力が結合されるため、パイプ機能を利用して、各コマンドを独立したシステムで同時に実行させることができる。

- MPI
  MPI(Message Passing Interface) は，MPI Forum によって標準化された，並列アプリケーションのメッセージパッシングライブラリのインタフェース規約である．MPI の特長を次に示す．
  - プロセスを単位とした並列プログラムを記述できる．
  - メッセージパッシングを高速に実行するため，複数の通信モードがある．
  - プロセスグループと通信コンテクストを統合したコミュニケータを指定して通信を行う．
  - 集団通信のための関数を豊富に提供している．
  - プロセスを仮想的に格子状，又は網状に配置する仮想トポロジをサポートしている．

5.5 分散ファイルシステムサービス

Beowulf クラスタでは，ほとんどの場合 NFS（Network File System）プロトコルを使って，分散ファイルシステムサービスを提供する．NFS を利用することにより，遠隔ホストにあるファイルシステムをローカルにマウントすることが可能となる．これによって遠隔ホストとローカル上でファイルを共有することが可能となる．

NFS はクライアント/サーバモデルを使用しており，共有するディレクトリをサーバがエクスポートし，クライアントはそのディレクトリをマウントして中のファイルにアクセスできるようにする．

これについては Fig. 4 を例にとって説明する． Fig4 では，NFS サーバが/home をエクスポートし，Client1〜3 がエクスポートされた/home にマウントしている．エクスポートされた/home は NFS のパーティションであり，ハードは NFS サーバのものである．Client 達から見える/home の内容は NFS サーバの/home になり，/home へのファイルの書き込み，/home への読み込みはすべて NFS サーバの/home を対象とすることになる．

6 本研究室にある主なクラスタの性能

- Cambria system
  CPU：Pentium 800MHz × 256
  Memory：128M × 256（計 32GB）
  Network：FastEthanet
  OS：Debian GNU/Linux 2.2
- Gregor system
  CPU：Pentium 1GHz × 64 × 2
  Memory：512M × 64（計 32GB）
  Network：Myrinet 2000
  OS：Kondara HPC

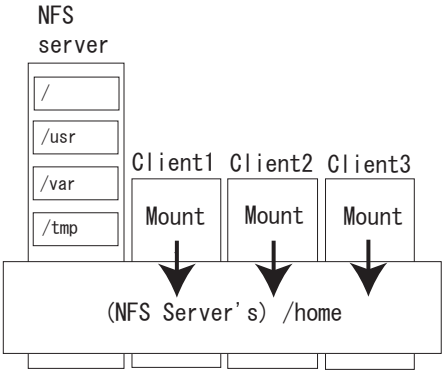

Fig. 4 NFS プロトコルの流れ

7 参考文献

- PC クラスタ構築法
  トーマス・L・スターリング他
  産業図書株式会社
  2001 年