

▼Beowulf型コンピュータシステム

オープンかつフリーのソフトウェアと汎用ハードウェアを用いた計算機クラスタをそう呼ぶことがある。Beowulfは英雄叙事詩の英雄名であるが、1994年にNASAのCESDISで、最初に構築された計算機クラスタの名前だったことに因む。  
<http://beowulf.gsfc.nasa.gov/beowulf.html>

▼コモデティアーキテクチャ

容易に手に入る計算機(PC, WS)や、ネットワークなどの汎用の部品(コモデティ部品)を用いた計算機アーキテクチャ。

▼VIA (Virtual Interface Architecture)

1997年12月、Compaq, Intel, Microsoftが主体となり、100社以上の企業によって規格化された計算機クラスタ用の高速通信のための業界標準仕様。<http://www.viarch.org/>

▼レイテンシとスループットとバンド幅

バンド幅、スループットは、単位時間に送れる通信量、レイテンシ(通信遅延)はデータを送ってから受け取られるまでの時間で、同質の性能指標を表すわけではない。一般に、通信時間は、転送データ量によらない初期セットアップ時間とデータ量に比例したデータ転送時間とに分けられるが、レイテンシは主に初期セットアップ時間などに影響される。データ転送時間は、バンド幅によって決まる。小さいメッセージを多く通信する場合などは、バンド幅よりもむしろレイテンシ性能の影響が大きい。

▼AM (Active Messages) 通信, FM (Fast Messages) 通信, PM通信

1990年代の初頭にActive Messagesといわれる低通信遅延、メッセージ駆動型というべきメッセージ通信方式が、California大で開発された。1992年ころには、当時全盛であった、並列計算機nCUBEやCM5に研究実装され、ベンダの提供する従来型の通信方式にくらべ、同じハードウェアを使用しながら、1桁以上の通信高速化を可能にした。メッセージ転送におけるソフトウェアオーバーヘッド低減に着目した低通信遅延の通信方式開発のきっかけとなった。その後、AMをもとにしたFMがIllinois大で、さらに、PMがRWCP(日本)で開発された。

▼SMP (Symmetric MultiProcessor)

メインメモリへのアクセスがどのプロセッサからも対称(アクセスコストが均一)であるような、共有メモリ型中規模マルチプロセッサ。すべてのプロセッサとメモリモジュールが同一の接続機構(通常、共有バス)に接続されたシステムを指すことが多い。

▼MPP (Massively Parallel Processor)

比較的プロセッサ数の多い(厳密ではないが、数百台から数千台程度)並列計算機の総称。

▼SAN (System Area Network)

LANに比べ狭い範囲(近く)にある計算機を(計算機クラスタ)接続するための高信頼性、低通信遅延、高速のネットワークを指す。VIA, MyrinetなどもSANを強く意識している。

▼メッセージパッシングモデルと共有メモリモデル

並列プログラムを書く上での代表的な2つのプログラミングモデル。メッセージパッシングモデルは、メッセージ通信ライブラリなどを用いて、プロセッサ間のデータ交換を、相手プロセッサを指定してデータを送受信する方式。共有メモリモデルは、従来の逐次計算機のプログラムのように、メモリ(変数)へのリード/ライトで、間接的に共有メモリ(共有変数)を用いてプロセッサ間でデータを受け渡す方式。

▼メッセージ通信ライブラリ

複数のプロセッサを用いる並列処理プログラムで、プロセッサ間のデータ授受、同期などを、send/receiveといった、メッセージ受け渡し関数で行う方式が1989年ごろからいくつか開発され、並列プログラミングの1つの標準的なスタイルとして広まった。これらは、データの格納アドレス、データサイズ、送受信の相手プロセッサ識別番号などを引数として、データを送受信する関数で、ライブラリとして、CやFORTRANから使うことができる。従来の並列プログラムは、使用する並列計算機固有の命令やディレクティブ、関数などで書かれており移植性がなかった。これらのメッセージ通信ライブラリは、計算機内部の詳細をライブラリ実装に隠すことにより、並列計算機に依存しないユーザインタフェースを提供し、並列プログラムの移植性を飛躍的に高めた。

▼PVM (Parallel Virtual Machine)

1989年にOak Ridge国立研究所で開発されたメッセージ通信ライブラリの代表格。単なるデータ交換だけでなく、異機種計算機による並列処理を可能にするデータ表現の自動変換、動的プロセス生成時のマシンの自動選択など、仮想マシンとしての実装を含むソフトウェアライブラリで、フリーでソースが公開されている。いくつかの並列計算機ベンダも開発している。異機種計算機間で、プログラムの移植性があるばかりでなく、異機種計算機混在環境でもプログラムの相互動作ができる(Inter Operability)のが特徴。ただし、固有の並列計算機に特化した機能を用いた高速化はできない。<http://www.epm.ornl.gov/pvm>

▼MPI (Message Passing Interface)

1992年に並列計算機ベンダと学術組織など40の組織で構成されたMPI Forumによって規定された、メッセージ通信ライブラリの標準ユーザインタフェース。ライブラリの実装方式は定めず、この仕様をもとに、各計算機ベンダがそれぞれの計算機固有の方式を用いて、性能の高いライブラリを作成することを目的とする。現在、各ベンダの並列計算機で実装の異なるMPIライブラリが提供されている。LAN結合の計算機クラスタで使える汎用のフリーの実装ソフトとしては、MPICH(Argonne国立研究所)がある。<http://www.mcs.anl.gov/mpi>

▼HPF (High Performance Fortran)

従来のFORTRANのコメント行を使用して、並列化のためのディレクティブを挿入し、並列処理も可能にしたFORTRAN。並列プログラムの1つのユーザインタフェースになりうる。

▼OpenMP

1997年10月に提唱された、UnixやWindowsNTを用いたマルチプラットフォーム環境における共有メモリプログラミングのための応用プログラムインタフェース。DEC, IBM, SGI, Intel, HP, SUNなどの多くのベンダが参加し、パソコンからスパコンまでを視野においた、共有メモリプログラミングにおける業界標準インタフェースをめざしている。現在は、FORTRANで使用できるコンパイラディレクティブの標準セットを提供するのみだが、順次C, C++にも対応の予定。<http://www.openmp.org/>

▼SCI (Scalable Coherent Interface)

バンド幅とスケラビリティに限界のある従来型のバスに代わる、新しいマルチプロセッサシステム(計算機クラスタ)向けの、低通信遅延、高バンド幅、スケラブルなシステム間インタフェース。従来の、プロセッサ、メモリ、I/Oバス、ネットワークインタフェースなどの枠を越えた統括的な標準仕様(IEEE Standard-1596)である。

▼NAS並列ベンチマーク(NPB)

もともとはNASAの研究所で開発された、並列計算機の性能を評価するためのNAS(Numerical Aerodynamic Simulation)プログラム。従来のkernelベンチマーク(LINPACK, Livermore Kernel, NAS Kernelなど)は、主にベクトル型スーパーコンピュータ用であったが、NPBは、並列コンピュータ向けに開発された。5種のカーネルと3種の疑似応用の8つのプログラムからなり、問題サイズの大きさなどにより、3クラス(A, B, C)のレベルがある。<http://www.nas.nasa.gov>

▼スーパーリニアスピードアップ

通常、同一の応用処理を、n台のプロセッサで並列処理した実行時間は、1台による実行時間の、1/n以下には小さくならないはずで、理想的な場合でも最大n倍のリニアな高速化しか得られない。しかし、応用によっては、n倍以上の高速化ができる場合があり、このような場合をいう。たとえば、探索問題などで、並列処理によって、早い段階で多くの枝刈りができ、逐次処理よりも探索空間を小さくできると、n倍以上の高速化が可能である。

▼ERP (Enterprise Resource Planning)

企業の経営資源の効率的運用を経営戦略に役立てる仕組み。本会誌1998年7月号コラム参照。

▼トランスポートファイル

通常のファイルシステムでは複数の属性からなるレコードを単位としてデータを格納するが、属性を単位としてファイルをアクセスする応用では入出力効率が高くないため、属性ごとにデータを格納するトランスポートファイルが考案されている。各属性データにIDを付加することによりレコードを再構成することができる。

[本特集エディタ: 緑川博子]