ＳＲ２００１ ＯＳの開発コンセプト

2Ｈ－1　西門　隆,* 岩嵜　正明,* 藤田　不二男,** 山本　徹,*** 柴宮　実 **
* (株)日立製作所 システム開発研究所、**同 ソフトウェア開発本部、
***日立ソフトウェアエンジニアリング(株)

１．はじめに

大規模科学技術計算では、これまで主にベクトル型スーパコンピュータが使われてきた。しかし、メモリを共用するベクトル型スーパコンピュータでは処理能力の向上に限界があり、より複雑な計算には対応しきれなくなってきた。

そこで、ローカルにメモリを持つプロセッサ(ノード)を高速な通信路で結び、各ノード上のプログラムがデータ交換をしながら並列に各ローカルメモリ上のデータを処理する分散メモリ型のアプローチが取られるようになってきた。ＳＲ２００１は、この分散メモリ型の科学技術計算用の並列計算機である。

本稿では、ＳＲ２００１について簡単に説明すると共に、それを制御するＯＳの開発コンセプトと概要について述べる。

２．ＳＲ２００１の概要

ＳＲ２００１は図１に示す通り、ＲＩＳＣ型プロセッサを持つノードを２次元のクロスバ方式高速通信路で繋いだ構成を持つ。システムは機能分割されており、ディスクや外部ネットワーク装置等の周辺装置を持つＩ／Ｏノードと、周辺装置を持たず主に計算を担当する計算ノードにより構成する。Ｉ／Ｏノードの１つ(ＳＩＯＵ)にはコンソールが接続され、システム全体の管理を担当する。

各次元方向のクロスバ通信路は、各ノードのプロセッサとは独立したデータ交換装置(Exchanger)で繋がれており、任意のノード間は距離に関係なく、最大でも１つのExchangerを経由するだけで高速なデータ転送が行える。ノード間通信には以下の２パタンがあり、それぞれのパタンに対して、１対１通信だけでなく、複数ノードへのブロードキャスト通信が行える。

（１）バッファ型通信
　受信側が用意したリングバッファにデータを転送する通信方式

（２）直接メモリ転送型通信
　受信側の格納場所を指定してデータを転送する通信方式

さらに、ノード間の高速同期のためにバリヤ同期機能を持つ他、システムメイテナンスやシステムデバッグ[1]用にクロスバとは別の通信路を持つ。

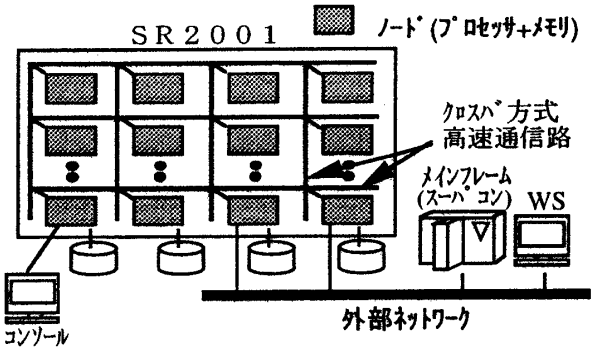

図１．ＳＲ２００１のシステム構成

３．ＯＳの開発コンセプト

科学技術計算用並列計算機では以下の２つが要求される。

（１）スケーラビリティ

並列計算機は、単に高い処理性能を実現するだけでなく、より高度な計算にも容易に対応可能とするため、要求に応じてノードや周辺装置を追加することで、プログラムを変更することなく、性能・機能が向上可能であることが必要である。すなわち、ユーザプログラム自身は、システムの構成を意識しなくても、ＯＳ側でシステムの増強を認識して、自動的に性能が向上することが必要である。

（２）マシン特性の最大限の活用

物理計算などでは、まだ解けていないグランドチャレンジ問題を解くことが並列機のメインの使用目的である。従って、(１)のＯＳ機能によって自動的にスケーラブルに性能が向上できるだけでなく、必要であれば部分的にユーザがマシンを意識してマシンに依存したチューニングを施し、そのマシンが持つ最大性能を引き出すことが可能であることが必要である。

そこで、ＳＲ２００１では、これらの２点を考慮したＯＳの開発を目指した。

４．ＳＲ２００１　ＯＳの概要

前章で述べた開発コンセプトに基づき、ＳＲ２００１では以下の特徴を持つＯＳを開発した。

（１）シングルＵＮＩＸ+システムイメージの提供

ユーザプログラムにシステム構成を意識させずスケーラブルに機能・性能の向上を可能とするため、システム全体を１つのＵＮＩＸシステムのように

Design Concept of SR2001 OS
Takashi NISHIKADO*, Masaaki IWASAKI*,
Fujio FUJITA*, Toru YAMAMOTO**, Minoru SHIBAMIYA*
*Hitachi, Ltd., **Hitachi Software Engineering Co., Ltd.

+ UNIX は、X/Open Company Limitedがライセンスしている米国ならびに他の国における登録商用です。

見せる機能を提供する。この機能は、図２に示す通り、機能別のＯＳサーバが連携して１つのＯＳを構成するマイクロカーネル方式により実現する。すなわち、メモリ管理やプロセス間通信機能等の基本機能だけを持つマイクロカーネルを各ノードに登載し、各ノードの機能に応じてさらにファイルサーバやネットワークサーバを動作させる構成をとる。

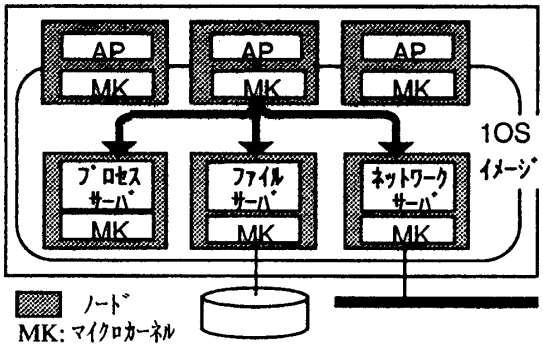

図２．ＳＲ２００１　ＯＳの構成

例えば、周辺装置を持たない計算ノード上のプログラムからのＩ／Ｏ要求は、マイクロカーネルを通して、対応するサーバにメッセージ通信の形で送られ処理される。各サーバへの要求の振り分けは、ノードの構成とは独立した概念の、システム内でユニークに管理されているファイル名、プロセスid等使って行うため、ノード追加に対応してＯＳ側で振り分け先を自動的に増やすことで機能・性能を向上させることが可能である。

並列機を制御する方式としては、本方式以外にも各ノードにＵＮＩＸを登載してシステム全体をクラスタとして管理するクラスタ制御方式があるが、本方式は以下の点で並列機制御に適している。

(a) 各ノード間でファイルシステムを互いにマウントし合う必要がなく、ノード数を増しても管理が複雑にならない。
(b) 各ノードに外部ネットワークアドレスを割り当てる必要がなく、各サイトに割り当てられた数少ないネットワークアドレスの使用を最低限にすることが可能。
(c) 計算ノードでのＯＳによるメモリ占有量が少ないため、ユーザは多くの実メモリを使用可能。

（２）用途別によるノード割り当て機能

並列計算機上で動作するプログラムには、複数ノード使って計算をするプログラムもあれば、１つのノード上で動作してユーザとの会話処理を行うプログラムもある。後者のタイプのプログラムは、１つのノードを他と共用してTSS的に実行しても問題ないが、典型的な並列プログラムでは、各ノードのプログラムが同期して動くことを前提としているため、一部のノードを共用してノード間の処理時間がばらつくと、プログラム間で待ちが発生するという問題が生じる。そこで本システムでは、図３に示す通り、用途別にノードを分割し、その分割単位に設定されたノード共用・非共用の属性に従い、各ノードをプログラムに割り当てる機能を提供する。

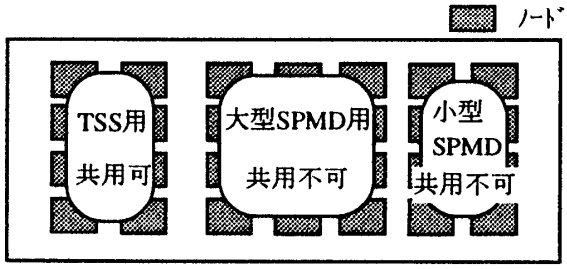

図３．用途別ノード割り当て機能

（３）低オーバヘッドノード間通信機能[2]

従来ＵＮＩＸのＴＣＰ／ＩＰ等を使った通信では、ＯＳ内でのプロトコル処理のため、送受信時に数百マイクロ秒のオーバヘッドが発生する。そこで図４に示す直接メモリ転送型通信を使用し、受信側が予め指定したユーザ空間領域に送信側が直接データを転送する機能を提供する。これにより、送受信時のＯＳのプロトコル処理をなくし、必要最小限の起動オーバヘッドでノード間通信を行うことを可能としている。

また、通信状況等のモニタリング機能を提供し[3]、マシンに依存したチューニングを可能としている。

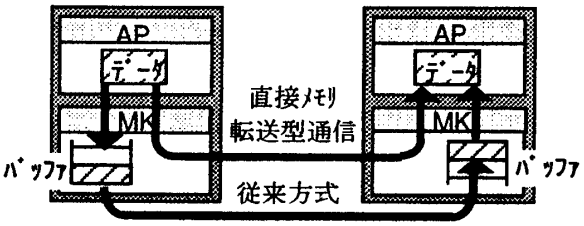

図４．低オーバヘッドノード間通信

５．終わりに

科学技術計算用並列計算機ＳＲ２００１用ＯＳの開発コンセプト及び概要について述べた。本ＯＳは、マイクロカーネル方式を用いてＯＳ側でスケーラブル性を実現するとともに、ユーザがマシン特性を最大限活用したプログラミングをすることも可能にした。

参考文献

[1] 山本、他：SR2001 におけるカーネルデバッガ　情報処理学会第50回全国大会(1995)
[2] 薗田、他：SR2001 における高速プロセッサ間通信機能　情報処理学会第50回全国大会(1995)
[3] 吉松、他：SR2001 における並列トレーサ／モニタ機能 情報処理学会第50回全国大会(1995)