超並列計算機 SR2201 における分散プロセス管理機能の実装

1F−3

副島健一*　薗田浩二*　佐藤雅英*　志村光弘**
*(株) 日立製作所　**(株) 日立ソフトウェアエンジニアリング

1 はじめに

従来の超並列計算機の多くは、少数ジョブによる科学技術計算をその主な用途としていた。ところが最近では、超並列計算機を少数ジョブによる科学技術計算だけではなく、多数ジョブで共用したり多数ユーザが各々対話的に使用したりしたいという要求がでてきた。そこで日立の超並列計算機 SR2201 では、これらの用途に使用する場合に効率よくプロセス管理を行なうための機構について検討し実装した。本稿ではその概要を報告する。

2 機能要件

多数ジョブや多数ユーザの対話処理を行なうためのプロセス管理機構の機能要件は以下の通りである。

(1) 低システムコール処理負荷
システムコールの発生頻度が多くなるため、システムコール全般の分散処理が必要である。特に多数ユーザによる対話処理では、端末 I/O の処理の分散化も必要である。

(2) 低プロセス管理負荷
ジョブやユーザ数が増えると、システムコールの発生頻度だけでなく管理するプロセス数も増加する。そのため、プロセス管理のオーバーヘッドを最小限に押える必要がある。

(3) 単一システムイメージ
多数ユーザの対話処理では、各ユーザがファイルの共用や相互参照、共通サービスの利用等を行なうため、管理の容易性を図るために、単一システムイメージの提供が必要である。

3 従来方式とその問題点

従来の超並列計算機用 OS の主な処理方式を図 1(a) ～ (d) に示すとともに、先述の要件について検討する。

(a) クラスタ方式：各ノード毎に別の OS を動作させる方式であり、スケーラビリティは高いが、システムイメージがノード毎に存在するという欠点がある。

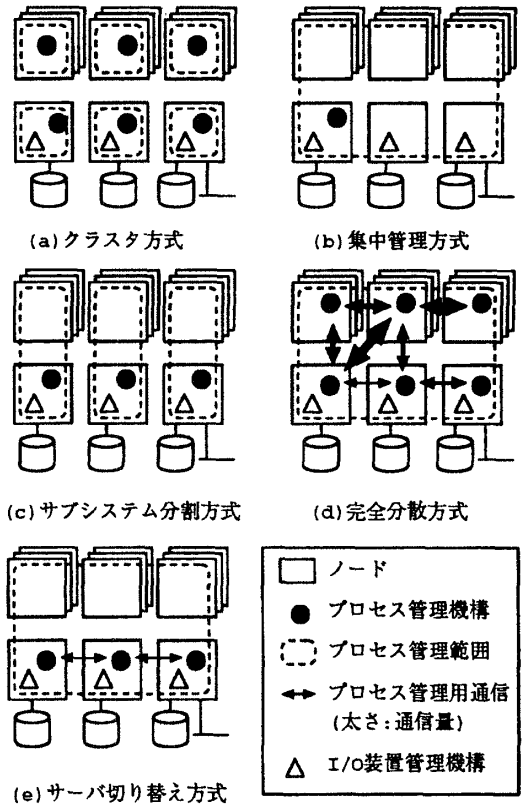

図 1: プロセス管理方式

(b) 集中管理方式：システム中の全プロセスを一箇所で集中管理する方式であり、少数ジョブからなる科学技術計算には最適であると言えるが、システムコール発生頻度が増えると一箇所に負荷が集中する可能性がある。

(c) サブシステム分割方式：システムを複数のサブシステムに分けサブシステム毎に別のシステムイメージを提供する方式であり、サブシステム分割の単位を調整することである程度の負荷分散は可能であるが、システムイメージが複数になるという欠点がある。

Implementation of Distributed Process Management Facility on SR2201
Ken'ichi Soejima
Systems Development Laboratory, Hitachi, Ltd.
1099 Ohzenji, Asao, Kawasaki, 215, Japan

(d) 完全分散方式： 複数のノード上で動作するプロセス管理機構が連携してプロセス中の全プロセスを管理する方式であり、システムコール処理負荷の分散と単一システムイメージの提供を両立できるが、プロセス管理情報のデータの一貫性制御によるプロセス管理負荷が増え、障害時に回復が困難であるという欠点がある。

SR2201 用 OS では、これまで少数科学技術計算ジョブ運用を前提として、I/O 処理や一部の頻出するシステムコールを分散処理しながら、図 1 (b)、(c) の方式を採用してきた。今回さらに多数ジョブや多数ユーザによる対話利用を可能とするため、2 節で述べた要件 (1) ～ (3) を満たすプロセス管理方式として図 1(e) に挙げるサーバ切り替え方式を検討し実装した。

4 分散プロセス管理の実現方式

SR2201 用 OS では、マイクロカーネル方式を採用し、プロセス管理をサーバの形で実現している。以下、図 1(e) に従ったプロセスサーバの分散方式について述べる。

4.1 システムコール処理の分散化方式

プロセスサーバを I/O デバイスが接続された全 I/O ノード上に動作させ、システム全体のプロセスをこれらのサーバで分割して管理する。各ノードでは、システムコール処理の窓口となるシステムコールエミュレータが動作し、各ノード上のプロセスから発行されたシステムコールは、内容に応じて分散処理する。すなわち、ローカル処理可能なシステムコールは同エミュレータで処理し、I/O 関連のシステムコールは対応する I/O デバイスを持つ I/O サーバに、プロセス関連のシステムコールはそのプロセスに対応するプロセスサーバに振り分けて処理する。

4.2 プロセスサーバ間連携処理の削減方式

プロセス管理を複数のプロセスサーバに分散化すると、システムコール処理時にプロセスサーバ間に跨る処理がより多く発生し、性能が低下する。そこで、以下の方式により、大部分の処理を各プロセスサーバ内に閉じさせ、分散化によるサーバ間連携処理を大幅に削減した。

(1) 管理サーバの設定・変更の限定
各プロセスの管理サーバの設定・変更は、基本的にユーザログイン時を契機として行い、ユーザセッション単位での負荷分散に限定した。これにより、(2) の管理サーバの継承に従ってプロセス間の関係を各プロセスサーバ内に閉じる形にできる他、ノード障害時に影響範囲が限定でき、管理が容易となる。また、管理サーバ設定・変更の契機をログイン時に限定することにより、オープンファイル等の環境の継承を最小限にし、設定・変更処理のオーバーヘッドを少なくした。

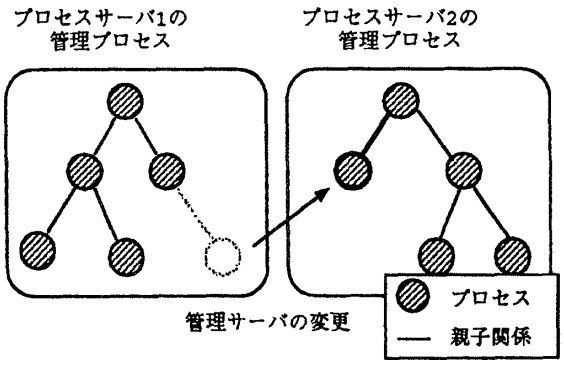

図 2: 管理サーバ変更の例

(2) 管理サーバの継承及びプロセス関係のローカル化
各プロセスの管理プロセスサーバは、プロセスの親子間で引き継ぐ。また、ログイン時に管理サーバを設定・変更した際には、それまでのプロセスの親子関係、グループ関係を断ち切り、各プロセスサーバの第 1 プロセスの子供として、図 2 に示す通り、各プロセスサーバ毎に独立のプロセス木を構成する。

4.3 端末処理の分散化方式

プロセス管理を分散化しても、ネットワークの疑似端末が別の I/O ノードで管理されていると、端末 I/O の度にノード渡りのサーバ間通信が発生し、十分な端末 I/O 性能を出すことができない。そこで、管理プロセスサーバの設定・変更時に、疑似端末処理の I/O サーバも移動先プロセスサーバと同一ノードで動作する I/O サーバに変更することで、端末処理の負荷分散を実現した。

5 まとめ

少数科学技術計算ジョブ運用だけでなく、多数ユーザによる対話用途に超並列計算機 SR2201 を適用するため、管理サーバ切り替え方式による分散プロセス管理方式を実装した。本方式により、超並列計算機システム全体を単一システムイメージに保ちながら、システム規模に応じたスケーラブルなユーザ数の対話利用を可能とした。

参 考 文 献

1) J. ボイキン他 著, 岩元 訳:“Mach オペレーティングシステム”, トッパン,1994.

2) S.J.Leffler 他 著, 中村他 訳 “4.3BSD の設計と実装”, 丸善株式会社,1992.