

## 解 説



## 日本におけるオペレーティングシステム研究の動向

## 4.2 耐故障性を持った商用の並列システム —SURE SYSTEM 2000 の OS “SXO”†—

黒 羽 法 男† 吉 田 浩† 村 松 洋†

### 1. はじめに

SXO は、通信制御/通信処理向けの並列システム SURE SYSTEM 2000 用に新たに開発された商用分散 OS である。高い通信負荷を処理しつつ、システムの連続運転と、ネットワークの成長に対応できるスケーラビリティを実現するところに設計の目標がある。特に耐故障性と高負荷通信処理のスケーラビリティを実現するために、複数のマイクロプロセッサによる NORMA\* (NO Remote Memory Access) 型並列構成としている。

SXO の 1 つの柱である連続運転性に関しては、プロセスペア方式による OS 機能の切替えを基盤に、ハードウェアとソフトウェアの耐故障化や活性保守など、システムの運用全般にわたる運転の継続を図った。

本稿では、耐故障化やスケーラビリティの基本となる「並列化」、すなわち、多数の通信トランザクションの並列処理を実現する商用の分散 OS の設計において直面する問題と解決した方法、および今後の課題に関する研究の紹介を行う。

### 2. 商用分散 OS の課題

分散 OS の設計には、定石化しつつある手法がある。すなわち、マイクロカーネルを採用し、メッセージパッシングを行うというものである。多くの研究や試作も、これに沿っているが、主な対象とされているのは、科学技術計算やデータベース検索といった内部並列性を多量に含んだアプリ

ケーションであり、図-1 に示す想定のもとに検討されていることが多い。これに対して、高負荷のトランザクション処理などの処理を主な用途とする商用システムには、以下のような特徴がある。

#### (a) 種類の多様性と並列化の制約

たとえば通信処理では、単なるゲートウェイ処理から、ファイル転送や電文の蓄積交換まで多様なアプリケーションがある。処理自体の分割は困難で、負荷も動的に変動する。

#### (b) 実用的な並列度の上限

高負荷トランザクション系処理で実際に遭遇する負荷は 10 から 100 プロセッサ相当であり、この領域の性能と運用性が重要である。

このような条件では、並列処理での多プロセッサの考慮をマイクロカーネルに隠蔽し、他のソフトウェアは、マイクロカーネルが提供する单一システムイメージに乗るだけというアプローチは、十分な性能やスケーラビリティが得られないことが多い。すなわち、ビジネス分野の商用並列システムの設計では、以下の課題の解決が必要である

(1) ディスク、LAN/WAN の高負荷アクセスに耐える入出力アーキテクチャ (ソフトウェア込み)

(2) システムの目的を果たす上で重要な、上位の OS 機能や、ミドルウェアの並列処理化 (特に LAN/WAN の通信アクセス法、トランザクション処理モニタ (TP モニタ) やデータベース管理システム (DBMS) といった高負荷のもの)

(3) 並列動作するアプリケーションやシステム運用に対する多数のプロセッサの見せ方

(4) OS 自身の生産性、およびミドルウェアやアプリケーションの生産性や移植の容易さ

この課題にこたえるために SXO で採用したア

† Fault Tolerant Parallel Operating System—SXO for SURE SYSTEM 2000—by Norio KUROBANE, Hiroshi YOSHIDA and Hiroshi MURAMATSU (Fujitsu Limited).

† 富士通(株)沼津工場

\* 各プロセッサが固有メモリを持ち、他プロセッサのメモリにはアクセスしない形態。

	研究・試作	商用
適用分野	科学技術計算系アプリケーション	高負荷ビジネス系アプリケーション
想定CPU数	1000個規模	10~100個規模
アプリケーションの性質	・処理時間短縮が目標 ・科学技術計算やデータベース検索処理など一様分割可能な性質を持つ問題を選択して適用	・処理件数向上が目標 ・通信処理やトランザクション処理など分割不可能なものや負荷が動的に変動するものも多い
並列化の対象	・演算の並列化	・入出力処理の並列化
ソフトウェアで解決する主要領域	・いわゆるOSのカーネル部の構成法	・OS上位機能やミドルウェアの構成法 ・アプリケーションの移植性や生産性

図-1 商用分散OSの課題

アーキテクチャの要点は、次のようにまとめられる。

### (1) アクセスの分散・並列化と管理系の集中化

課題(1)～(4)に対応する。3.2に具体的な対策を述べる。

### (2) 分散・並列アクセスに対する負荷の配分

課題(2), (3)に対応する。3.3に具体的な対策を述べる。

### (3) 多プロセッサ上での並列動作をするソフトウェアの生産性を上げるためにカーネルやミドルウェアの対処

課題(4)に対応する。3.4に具体的な対策を述べる。

## 3. SXOのアーキテクチャの特長

### 3.1 OSの構造の概要

SXOが動作するSURE SYSTEM 2000のハードウェアは、NORMAアーキテクチャを基本としながらも、共用を強く意識した、以下の特徴を持つ。

#### (1) どのPMからも全入出力装置にアクセスできるI/Oアーキテクチャ

#### (2) プロセッサ切換えのためのPM間共用の引継ぎ情報格納用不揮発メモリ

これらの特徴を持ったSXOの構造を、図-2に示す。マイクロカーネルは、PMごとに存在し、メッセージ、ディスパッチャ、割込み処理、入出力、仮想記憶、異常処理、PMの相互監視などを実行。上位のOS機能は、個別のアドレス空間を持つサーバとして実現される。カーネル間、カーネルとサーバ間では、相手PMの位置に係わらず、統一したインターフェースでメッセージパッシ

ングを行う。

SXOでは、ファイル、コンソール機能、プログラム管理などの十数個のOS基本機能のサーバに加えて、通信アクセス法などの上位のOS機能や、TPモニタ、DBMSなどのミドルウェアの一部もサーバの形態で実現している。

### 3.2 分散アクセスアーキテクチャ

#### (1) アクセス系の分散・並列化

SXOでは、高トラフィックの入出力処理に対応するために、いわゆるアクセス系のOS機能は、各プロセッサに分散して実現している。たとえば、ディスクアクセスのドライバは、サーバではなく、各マイクロカーネルの中に置き、メッセージパッシングのオーバヘッドをなくすと共に、全PMからディスクに対し並行してアクセス要求を発行できるようになっている。

また、通信アクセス法は、処理の複雑さから、カーネルではなくサーバとして実現されているが、各PMに1つずつ配置され、並列に動作する。

#### (2) 管理系の集中化

システム運用面から見た单一システムビューとしては、どのPMからも同じファイルやディレクトリが見えるといったように、ファイルなどの各種資源が一元的に管理されているビューを提供することが重要である。品質確保や機能追加におけるOS作成者の生産性も考慮すると、機能単位に一元管理を行うことが单一システムビューを実現しやすい。通信やトランザクションの定常処理では、この種の管理機能の動作頻度が低いことから、各資源を管理するサーバの現用系をシステム内のどれか1つのPMに置き、システム全体の資源を集中管理する方式が、想定している100プ

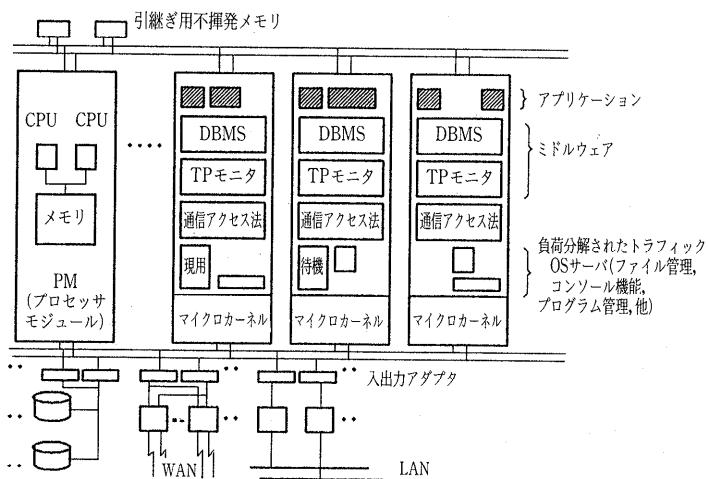


図-2 SURE SYSTEM 2000 のハードウェアおよびソフトウェア構造

ロセッサ規模では実用上優れていると判断した。

### 3.3 分散アクセス上の負荷配分

分散アクセスアーキテクチャが意味を持つためには、この上に、実際のアクセスのトラフィックを配分するしくみが重要となる。

#### (1) 通信アクセス

LAN/WAN のアダプタは、全 PM からアクセスできるので、配下の複数の回線の負荷を、各 PM に分配することが必要となる。この配分割御は、システムの負荷状況を見ながら動的に行うこととも考えられるが、SXO では、現実的な解決策として、回線数が均等になるよう自動化する方式と、運用設計に基づく負荷配分をネットワーク定義ファイルで指示する方式の 2 つを開発した。定義ファイルを用いても、SXO の特長である定義体活性保守機能によって、運用中でも動的に変更が行えるので、実用上は、拡張性が損なわれることはない。

今後の課題としては、ATM (Asynchronous Transfer Mode) LAN のような、単位プロセッサ能力を越える大きな通信負荷の分割方法がある。

#### (2) 通信アプリケーション

並列に動く通信アクセスやディスクアクセスの上で、通信トランザクションのアプリケーションを並列に実行するためのしくみとして、SXO は、専用の TP モニタを装備している。これは、同じアプリケーションプログラムを全 PM に起動し、

回線からのメッセージを振り分ける。アプリケーションの設計手法は、従来、オンライン系のプログラミングでよく行われていたのと同じく、マルチタスキングによる多重走行を基本としている。

### 3.4 分散 OS 上のソフトウェア開発の生産性

#### (1) マイクロカーネルと上位の機能分担メッセージパッシングの例

マイクロカーネルのメッセージ機構は、システムの基本性能に大きく影響するため、できるだけ軽く小さく作りたい。一方、サーバの生産性の観点からは、高級なサービスが欲しい。たとえば、メッセージが紛失しないことや、発行順に到着することの保証、サーバの処理能力を越える量のメッセージが待ちキューに滞留することを防ぐ FIFO 制御などである。このトレードオフを考慮して、SXO のマイクロカーネルでは、

①メッセージの順序性は保証しない

②重装備の FIFO 制御は行わず、大量のメッセージが滞留したサーバは異常終了させる

③宛先サーバの異常時でも再送によるメッセージの到着は保証するが、冗送もありうるというところまで行い、メッセージの逆転や冗送の対処は、送信データに通番を含めるなどの方法で、サーバ側で対処するという分担とした。

こうして、十分高速なメッセージパッシングを実現するとともに、各サーバの生産性も、従来の集中型 OS と同程度とすることができた。ただし、一般的なメッセージパッシングシステムにお

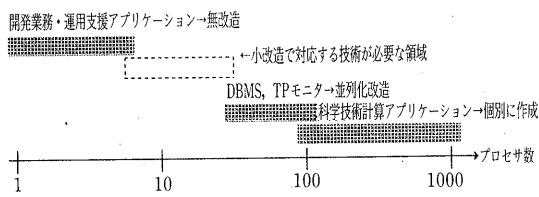


図-3 上位ソフトウェアの並列化対応

いては、特に、順序制御やフロー制御に関しては、より定式化した考察の余地があると考えられる。

(2) 並列化への対応とソフトウェアの生産性  
ビジネス系の商用の並列システムでは、マイクロカーネルとサーバによるOS自体の並列化に加え、プログラマに負担をかけずに、種々のアプリケーションを多数のプロセッサ上で並列に実行できるようにすることが必要となる。注目すべきことは、科学技術計算分野とは異なって、高負荷トランザクションアプリケーションの大半は、TPモニタによって通信や実行の制御を受け、DBMSに依頼して、ディスクアクセスを行うことである。

100プロセッサ級の並列度に対応するためには、これらのミドルウェアに、並列化を考慮した改造を加えることが重要である。前項で述べたTPモニタに加えて、DBMSでは、データベースを複数のディスクに物理的に分割して受持ちのPMを決めておき、トランザクション要求を解析して、DBMS内部で通信しながら、アクセスの対象となる分割単位を管理するPMに処理を配分する制御を行う。このような改造によって、アプリケーションに対しては、逆に、並列化を意識させずに済む。

一方、開発業務や運用支援アプリケーションの大半は、最近のプロセッサ能力の著しい向上から見て、1プロセッサないし数個の対称型マルチプロセッサで十分であり、並列処理をほとんど必要としない。このような非並列負荷に対しては、マイクロカーネルとOSサーバにより、集中型システムと変わらない見え方で標準的なOSのアプリケーションインターフェースが提供される单一システムビューを実現した。この範囲ならば、アプリケーションは無改造で動作し、性能も十分である。

図-3に示すように、通信処理やトランザクション処理の範囲では、この二段階の並列度を考慮したこと、実用上は十分であった。一方、この中間のプロセッサ数が2~30個までの並列度に、アプリケーションができるだけ改造せずに対応する技術は、Sxoの適用分野では特に登場していないものの、並列処理一般の課題としては、未解決といえる。今後、並列処理の適用分野の拡大を図る場合には、解決が必要な課題の1つであろう。

#### 4. 分散アクセスとOS設計

耐故障性を持った分散OSを新規に設計するならば、Sxoのように、NORMAアーキテクチャに共用I/Oを持つハードウェアの上で、OSの分散アクセスアーキテクチャを採用することが、性能、スケーラビリティ、耐故障性の面で最良と考える。

しかし、現実に行われているもう1つのアプローチとして、UNIXなど既存の流通OSを改造して移植する場合を考えると、分散アクセスアーキテクチャの採用は、必ずしも容易ではない。一枚岩の構造を前提とした既存カーネルでは、たとえば、ファイルシステムは、プロセス間でファイルのバッファやファイル情報の共有などを行っている。ここに、分散アクセスアーキテクチャを導入するには、ファイルシステムを構造面から作り直すことが必要になり、OSの改造の生産性や、既存・流通品のファイルシステムに対する互換性の点では、著しく不利となる。

この場合は、PM間でメモリ共用が可能なNUMA(Non Uniform Memory Access)アーキテクチャに共用I/Oを加えたハードウェアを採用し、ファイルシステムは従来の構造に準じたものとする方式が、Sxoの経験から見て、有効と考えられる。ただし、メモリ共用に起因する耐故障性の問題とスケーラビリティ限界が生じ、Sxoのアーキテクチャでは100プロセッサ規模のスケーラビリティを目指せるのに対し、10プロセッサ規模を想定して流通OSを動かす解として意味があるといえよう。

## 5. 分散 OS と耐故障性

### 5.1 耐故障性範囲と部品化の選択

耐故障性を持つ商用 OS としては、ハードウェア故障に対しても、ソフトウェア障害に対しても、システムの連続運転を保証する必要がある。

一般的に、耐故障性の実現においては、独立度の高い部品へのシステムの分割と、各部品の冗長組込みが基本となる。SXOにおいては、ソフトウェア障害に対しても運用を継続するという観点から、OS のプログラムを部品化の対象とした。この部品化の作業において、分散 OS という基本構造は大変よく適していることが確認された。

### 5.2 ソフトウェア部品の制御手法

品質管理がよくなされたプログラムにおいては、障害の再現性は低く、たとえば、システムの再立ち上げを行うことで、現象は消えることが広く知られている。これは、ソフトウェアの障害が処理内容よりは、環境やタイミングに依存することに起因するものと考えている。

ソフトウェア部品の冗長化の最も単純な方式として、稼働側のサーバから待機側へのメモリデータの複写が考えられる。しかし、環境も同時に複写されるため、障害原因も待機側のメモリ内に伝播してしまうので SXO では採用しなかった。

そこで、SXO では、エッセンス引継ぎ方式とよぶクリーンなデータのみを抽出して受け渡す、すなわち、環境を共用しない方法を開発した。具体的には、要求の処理に失敗したサーバは、マイクロカーネルや自らの異常検出論理により閉塞され、待機状態にある同種のサーバが処理を受け継ぐ。この過程において、マイクロカーネルは処理要求のメッセージを新たなサーバへ再送するなどの耐故障性機構の重要な役割を果たす。機構や効果は、参考文献 3), 4), 9) に詳しい。

### 5.3 エッセンス引継ぎの評価

エッセンス引継ぎは、定常オーバヘッドが低いことに加え、ハードウェアの故障や活性保守に対しても、また、ソフトウェアの活性保守にも活用される。詳細は文献 1) ~10) を参照されたい。

## 6. おわりに

以上のように、SXO は、「分散メモリ非共用(NORMA)アーキテクチャ上の分散OS」を基本

## 情 報

構造とすることによって、スケーラビリティの高い商用の耐故障・連続運転システムを実現している。

このような商用分散 OS の設計の鍵は、

- ①入出力アクセス性能や入出力処理の並列化
- ②信頼性/連続運転性
- ③ソフトウェアの移植性向上や改造量の低減
- ④負荷分散・負荷調整

といった要件のバランスのとり方にある。SXO では、①②にはほぼ最適解を見い出したのに対して、③④に関しては、プロセッサ数や運用形態に応じて、利用できる機能や設計・運用方法の複数の選択肢を用意することが実用上重要という知見を得たことの意義が大きい。加えて、③は、NUMA アーキテクチャによる共用メモリの高性能な実現といったハードウェア面の解決が、また、④は、ミドルウェアによるソフトウェア面の解決が、今後、必要となってくるものと考えている。

## 参 考 文 献

- 1) 河部本: SURE SYSTEM 2000 新通信プロセッサのアーキテクチャ、情報処理学会、計算機アーキテクチャ研究報告, 86-1 (1991).
- 2) Kabemoto, A. and Yoshida, H.: The Architecture of the Sure System 2000 Communications Processor, IEEE Micro, Vol. 11, No. 4, pp. 28-31, 73-78 (1991).
- 3) 伊達他: 連続運転システム SURE SYSTEM 2000 の OS SXO, 情報処理学会第 42 回全国大会論文集, 7 K-2~8 (1991).
- 4) 村松他: システムを止めずに保守・運用が可能な OS を開発, 日経エレクトロニクス, No. 520, pp. 209-223 (1991).
- 5) Muramatsu, H. et al.: Operating System SXO for Continuous Operation, Proc. 12 th IFIP Congress 1992, Vol. 1, pp. 615-621 (1992).
- 6) 春日他: SURE SYSTEM 2000 によるネットワーク構築, FUJITSU, Vol. 44, No. 3, pp. 234-242 (1993).
- 7) 大島: 連続運転オペレーティングシステムにおけるロードモジュール管理方式, 情報処理学会第 44 回全国大会論文集, 7 G-3 (1992).
- 8) Yoshida, H. et al.: Fault Tolerance Assurance Methodology of the SXO Operating System for Continuous Operation, IEICE Trans. Inf. Syst., Vol. E 75-D, No. 6, pp. 797-803 (1992).
- 9) 木塚: サーバ型空間の耐故障化方式, 情報処理学会第 43 回全国大会論文集, 2 L-5 (1991).
- 10) 山本他: SXO の通信処理アプリケーション制御, 情報処理学会第 43 回全国大会論文集, 3 C-1~2 (1991).

(平成 6 年 7 月 1 日受付)



黒羽 法男 (正会員)

1952年生。1977年東京大学工学部電気工学科卒業。同年富士通(株)に入社し、現在に至る。汎用システムMシリーズ・スーパコンピュータ VPシリーズのオペレーティングシステムの開発、超高信頼並列システム SURE SYSTEM 2000 のオペレーティングシステム SXO の企画・開発に従事。



村松 洋 (正会員)

1947年生。1969年東京大学理学部物理学科卒業。1971年同大学院修士課程修了。1972年同大学院博士課程中退、富士通(株)入社。現在、同社グローバルサーバ事業本部主席部長、兼富士通研究所並列ソフト研究部長。オペレーティングシステム、言語処理系の開発・企画、システム性能評価に従事。共著「ソフトウェア構造」(オーム社)、ACM会員。



吉田 浩 (正会員)

1955年生。1978年東京大学工学部電子工学科卒業。1980年同大学院情報工学修士課程修了。同年富士通(株)に入社し、現在に至る。MシリーズオペレーティングシステムおよびSXOの開発、並列システムの企画に従事。IEEE会員。

