

デモ18 ナノカーネル方式による異種 OS 共存技術「DARMA」の提案

新井利明*、関口知紀*、佐藤雅英*、井上太郎*、中村智明**、岩尾秀樹***

* (株)日立製作所システム開発研究所、** (株)日立製作所新事業推進本部、*** (株)日立製作所情報機器事業部

1. はじめに

独自のオペレーティングシステム(OS)を開発、保守するには多くの工数と期間を要する。そのため、多くの場合、広く普及している事実上の標準的(de facto standard)OS を利用してシステムを構築している。しかし、標準的 OS を利用する場合、障害時の対応や拡張性の点で、システム構築上の制約があった。

この制約を緩和する手段として、一つのマシン上に複数 OS の搭載を可能とするナノカーネル方式を提案する。標準 OS と独自 OS をマシン上に共存させ、OS 間の通信機能を付加することで、複数 OS 間の機能連携、機能補完が可能となり、システム全体の信頼性、拡張性を確保することができる。

2. 標準 OS 利用の問題点

標準的に普及している OS を使用することの利点は、豊富なソフトウェアが利用可能であることであるが、一方で、以下のような問題点も発生する。¹⁾

(1) 拡張性、機能追加、リアルタイム性

標準的 OS は汎用的に開発されているため、特定用途向けの機能が不十分であることが多い。その不足している機能が OS 構造に依存している場合にはその機能を個別に付加することは不可能である。たとえば、制御システムではリアルタイム性能確保が必須であるが、標準的 OS では応答性が不十分であることが多い。

(2) 信頼性

信頼性の確保も大きな問題である。標準的 OS およびそのアプリケーションソフトは一般にソースが公開されていない。そのため、障害が発生した場合、早急に対応することはほとんど不可能である。

(3) 従来との互換性

企業の情報システムでは、従来より蓄積された多く

のソフトウェア財産が存在する。これらのソフトウェアは長年にわたって使用されており、十分な信頼性を有している。そのため、標準的 OS 導入のためにこれらのソフトウェアを書き換えることは工数だけでなく、システムの信頼性を悪化させるという観点からも問題がある。

上記は、基幹システムや制御システムのような、長期間にわたる開発、保守、運用が必要なシステムを構築する際には致命的である。

3. ナノカーネル方式による解決

以上の問題点を解決し、標準的 OS を用いて、効率よく計算機システムを構築するため、ナノカーネル方式を提案する。ナノカーネル方式は、標準的 OS を使用しつつ、かつ、信頼性上あるいは機能上不可欠な部分には自由に独自の部品を追加することのできる構造、を確保するための基本機能である。これにより、一つのハードウェアプラットフォーム上に複数の OS を搭載させ、OS 間の関係、機能補完によってシステムの機能向上が可能となる。

基本構造と考え方を図1に示す。ナノカーネルは1マシン上に複数の OS を搭載するための、拡張され

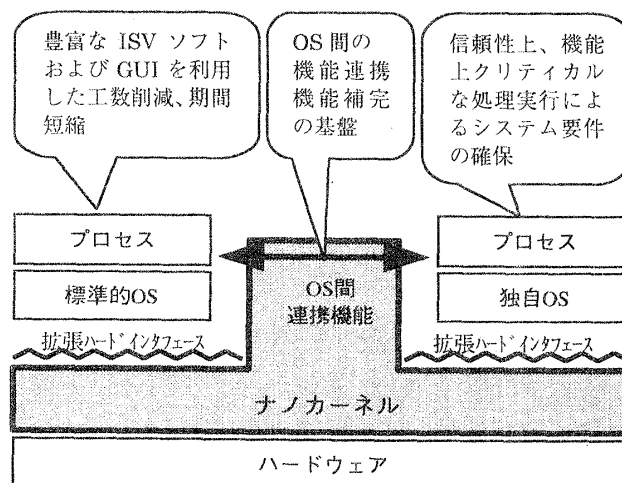


図1 ナノカーネルの基本概念

たハードウェアアーキテクチャを提供する。²⁾ ナノカーネルの機能は3つに分類できる。

(1) 資源管理機能

計算機資源を分割し、各 OS およびナノカーネル自体に割付ける機能である。メモリおよび I/O 機器は指定された OS に占有であるが、CPU、タイマは OS 間で共用する。

(2) OS 間通信機能

異なる OS 上のプロセス間での直接通信を可能とする機能であり、共用メモリ、メッセージパッシング、プロセス同期機能を提供する。これにより、OS 間の機能連携、機能補完が可能となる。

(3) 障害監視、回復機能

ナノカーネルの特徴機能の一つであり、OS に障害が発生した際に、その状況を他 OS から監視し、必要に応じて回復処理を起動できる。これにより、OS 障害の影響を最小限に限定でき、アプリケーションレベルでの引き継ぎも可能となる。

表1 ナノカーネル機能一覧

#	項目	機能概要	
1	資源管理機能	メモリ分割	物理メモリを分割し、各 OS、ナノカーネル、共用メモリに割付け
		I/O 装置分割	各 I/O 装置を各々 OS に占有割付け
		プロセッサ分割	プロセッサを時分割して OS 間で共用
		タイマ共用	各プロセッサに仮想タイマを提供
5	OS 間通信機能	共用メモリ	OS 間で共用するメモリを構築
6	OS 間通信機能	メッセージパッシング	異種 OS 上のプロセス間のメッセージパッシング
7		プロセス間同期	異種 OS 上のプロセス間の同期制御
8	障害監視、回復	障害発生を他 OS に通知 障害 OS 以外の OS の継続実行を保証 書意 OS み再起動	

以上のナノカーネル機能により、マシン上に標準 OS と独自 OS を共存させることが可能となり、標準 OS のみの場合の問題点を解決できる。

(1) 拡張性、機能強化、リアルタイム性

標準的 OS に不足している機能を、共存する独自 OS を構築して補うことが可能である。例えば、標準的 OS とリアルタイム OS を共存させ、ナノカーネルの異種 OS 間関係機能を用いることにより、リアルタイム性を要求する部分のみをリアルタイム OS が

実行し、その他の処理は標準的 OS の豊富なアプリケーションを利用する、というような機能分担が可能となる。

(2) 信頼性

標準的 OS に障害が発生した場合にも、独自高信頼 OS がその状況を把握し、情報収集及びオペレータへの通知を行うことにより、原因の追求、回避策の検討、復旧手段の確保を容易に行うことが出来る。

また、機器の制御等、中断を許されない処理のみを高信頼 OS 上で実行させることにより、工数削減と信頼性維持という要求を両立させることが可能となる。

(3) 従来との互換性

蓄積されたソフト資産を独自 OS で実行させることにより、資産を有効に活用することが出来る。新規機能の追加が必要な場合には、標準的 OS 上に構築することにより、従来機能は独自 OS で、新機能は標準的 OS で、と機能分担することも容易に実現できる。また、全機能をレガシ OS 上に搭載し、システムを運用しつつ徐々に機能を標準的 OS に移植する、といった段階的機能移植も可能である。

以上のように、ナノカーネル方式を利用することにより、オープン系システムの長所を維持したまま短所を補い、高信頼、高機能なシステムを構築することが可能である。我々は、ナノカーネル方式を応用した DARMA(Dependable Autonomous Realtime Management)ナノカーネルを PC/AT ハードウェア上に構築し、ナノカーネル機能を検証した。

参考文献

- 1) 「Windows 活用の実態と可能性」、日経デジタルエンジニアリング、日経BP社、1998年11月号(1998, No.10)
- 2) Robert P. Goldberg「Architectural Principles for Virtual Computer System」、Center for Research in Computing Technology, Harvard University、1971