

# スーパーコンピュータ「京」の オペレーティングシステム

## Operating System for the K computer

● 師尾 潤      ● 山田雅彦      ● 加藤丈治

---

### あらまし

富士通では、スーパーコンピュータ「京」が世界一の性能を実現するため、オペレーティングシステム(OS)の開発に当たって以下の三つの施策の機能改善に取り組んだ。

まず一つ目は、独自CPUおよびインターコネクトのハードウェア性能を限界まで引き出すために、ハードウェア拡張機能をアプリケーションから直接制御する仕組みを備えた。二つ目は、スケーラビリティの改善として、複数のノード間でジョブの実行時間、システムの実行時間を合わせることでシステムの外乱による並列プログラムの同期待ち時間の増加を抑える同期スケジューリング機能を導入した。三つ目は、メモリアクセス性能の向上とメモリ使用効率の改善をねらい、複数のページサイズを利用できるマルチページサイズを実現した。

本稿では、今回開発したOSの性能向上のための機能、およびユーザビリティ、堅ろう性についても紹介する。

### Abstract

For the K computer to achieve the world's highest performance, Fujitsu has worked on the following three performance improvements in the development of the operating system (OS). First, to bring out the maximum hardware performance of our original CPU and interconnect, we provided a mechanism of controlling hardware extensions directly from applications. As the second improvement, we have introduced the synchronization scheduling function that minimizes the synchronization wait time of parallel programs resulting from system interruptions by coordinating job runtime and system runtime between multiple nodes. Third, multiple page size support that allows use of more than one page size has been achieved for improved memory access performance and memory utilization efficiency. This paper also describes the performance improvement functions, usability and robustness of the OS developed.

---

## まえがき

スーパーコンピュータは、様々な科学技術分野の大規模なシミュレーションの計算で使われている。例えば、天気予報では、予測範囲を市町村レベル、更には駅レベルへと分割し、範囲ごとに温度や風向、風速などを計算して精度の高い予測が可能となる。航空機の空力シミュレーションでは、模型や実機を作ることなく、航空機の動きを推定することが可能となる。創薬では、膨大な種類や組合せから治療薬の候補となる物質を抽出し、治療効果のある物質を絞り込むことができる。

このような大規模なシミュレーションを実施するには、使用するメモリ、計算ノード数、ファイルサイズなども膨大となるため、スーパーコンピュータのハードウェア性能を限界まで引き出すオペレーティングシステム（OS）が不可欠である。

本稿では、大規模シミュレーションなどを実現するスーパーコンピュータ「京」<sup>(注)</sup>のOSについて述べる。

## 「京」のソフトウェア構成

一般にスーパーコンピュータでは、

- (1) 膨大な計算を分割
- (2) 多数の計算機（計算ノード）を同時並行に協

(注) 理化学研究所が2010年7月に決定したスーパーコンピュータの愛称。

調動作

(3) 計算ノード間ではMPI（Message Passing Interface）と呼ばれる標準の通信ライブラリを利用して高速に計算結果を転送することにより高速に計算を実行する。

OSはアプリケーションの起動やIO要求を処理し、また時刻管理などのシステムの処理を行う。「京」のOSでは、OSのカーネルやライブラリを改良することにより、ハードウェア性能、ソフトウェア性能を引き出したものを作り上げた。「京」のOSは、開発当時のシステムの10倍の性能である10 PFLOPSを目標に開発を実施した。

「京」を構成する基本ソフトウェアは、OSと基本ミドルウェア（運用管理ソフトウェア、言語システム）からなる（図-1）。OSはLinuxカーネルのアーキテクチャ依存部分を実装するとともに「京」のハードウェアを利用するためのドライバを追加して、上位のミドルウェアから利用可能としている。

「京」に搭載するOSは、表-1に示す「ユーザビリティ向上」「性能向上」「堅ろう性向上」を目標として開発に取り組んだ。次章以降でそれぞれの特長を実現させる機能を紹介する。

## ユーザビリティ向上（ユーザ資産有効利用）

既存のスーパーコンピュータの中には、特殊なOSを用いているものもあるが、それでは、ユーザ

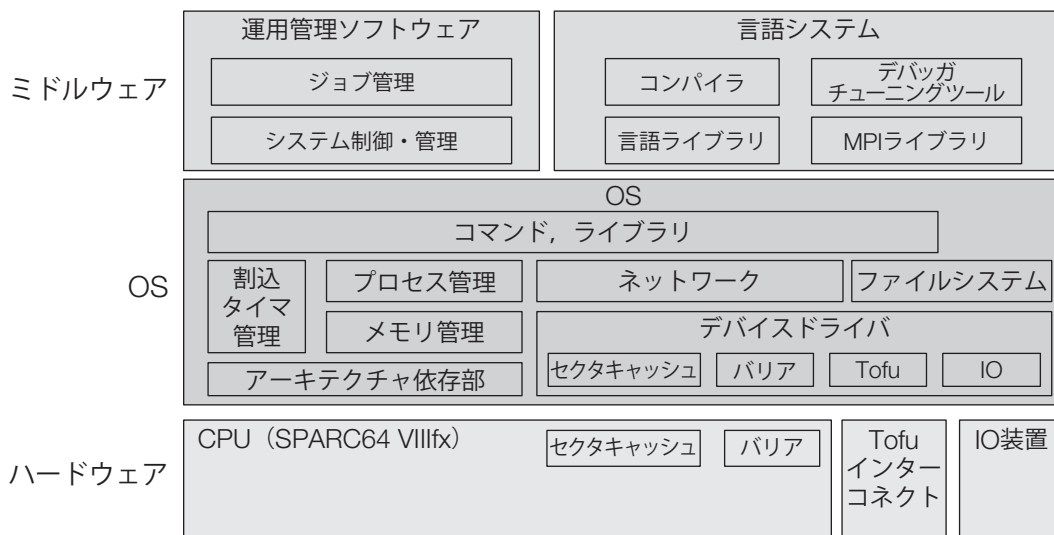


図-1 「京」のソフトウェア構成

表-1 「京」のOSの特長

|           | 特長              | 実現方法                                                            |
|-----------|-----------------|-----------------------------------------------------------------|
| ユーザビリティ向上 | ユーザ資産有効利用       | Linux<br>POSIX API                                              |
| 性能向上      | ハード拡張直接制御       | SIMD命令サポート<br>ハードウェアバリア<br>セクタキャッシュ<br>ハード制御レジスタのデバイス<br>ドライバ拡張 |
|           | スケラビリティ向上       | 同期スケジューリング機構<br>rsadc                                           |
|           | メモリ性能・効率改善      | ラージページ機構                                                        |
| 堅ろう性向上    | セキュリティ<br>RAS強化 | ユーザアクセス保護<br>故障時の縮退運用                                           |

がこれまでに開発してきたソフトウェアを活用できない。「京」のOSは、ユーザの資産を継続利用できるようにOSとしてLinuxを採用した。

「京」のOSは、一般的なOSと同様に、プロセス管理、メモリ管理、IO管理（デバイスドライバ）、ファイルシステム、ネットワークの機能を備える。Linuxは、スーパーコンピュータのランキングのTOP500のうち、91%以上のシステムで使われており、ユーザアプリケーションの移植性を考慮して採用した。ユーザはUnix系のアプリケーションプログラムインタフェース（API）であるPOSIX APIを「京」で使うことができる。

さらに、「京」のOSでは拡張されたハードウェアを利用するためのAPI（ライブラリ、コマンド）を拡張しており、コンパイラやMPIライブラリの機能を通じてハードウェアの機能を容易に利用することができる。

これらの工夫により、「京」のOSでは、一般的なオープンソースソフトウェア（アプリケーションプログラムやツールなど）をソースプログラムを変更することなく再コンパイルで移植可能である。

## 性能向上

### ● ハードウェア拡張直接制御

SPARC64 VIIIfxは、SPARC64 VIIをベースに高速化のためにレジスタ数拡張、SIMD命令拡張、コア間ハードウェアバリア機構、セクタキャッシュを搭載したCPUである。「京」のOSではこれらのハードウェア拡張機能をアプリケーションから直接制御可能とするデバイスドライバを追加した仕組みを備えた。ドライバがバリア同期やセクタ

キャッシュを制御するレジスタをアプリケーションのメモリ空間にマップすることで、システムコールのオーバヘッドを伴わずに高速に制御できる。

セクタキャッシュ機構は、CPUに搭載されているキャッシュを仮想的な二つの領域に分割し、アプリケーションから繰り返し使用するデータをキャッシュに残りやすくする機構である。

具体的には、セクタ0には使用頻度が低いデータをキャッシュし、セクタ1には、頻繁に使用するデータをキャッシュするようにコンパイラがメモリアクセス命令を生成する。これにより、頻繁に利用するデータがキャッシュから追い出されてしまうことを避けることができ、アプリケーションの実行性能を上げることができる。頻繁に利用するデータの量は、アプリケーション内の関数によって異なる場合があるため、ユーザアプリケーションから利用するセクタキャッシュの割合をプログラムの実行中に変更可能としてキャッシュの利用効率を最適化可能とした。

セクタキャッシュの操作に伴うデバイスドライバ呼出し時のシステムコール発行に伴うオーバヘッドが発生すると、キャッシュ制御による性能改善効果を十分に得られないため、「京」ではセクタキャッシュを制御するレジスタをユーザアプリケーションから直接操作可能とするようにドライバを設計、実装した。これによりセクタキャッシュの制御を数μ秒から数n秒に短縮した。

### ● メモリアクセス性能向上

「京」のOSでは、メモリアクセス性能とメモリ使用効率の双方を両立させるための機能として、ラージページ機構を搭載している。

通常、64ビットSPARCのLinuxに搭載された仮想メモリ管理機構は、プログラムが使用するメモリを8 Kバイトページ単位で管理している。

ハードウェアによって行われるアドレス変換処理を高速化するためにキャッシュ（TLB）を有しているが、そのエントリ数は有限である8 Kバイトのページサイズでは数Mバイト以上のメモリを利用する場合にTLB内で処理できず、OSのアドレス変換処理によるオーバヘッドが発生する。

一つのエントリで管理するページサイズを大きくすることにより、TLBを利用して数Gバイト以上のメモリを高速にアクセスできる。ただし、データ領域の全てを単一のサイズのラージページで管理すると、ラージページ中の未使用メモリ領域により、メモリの使用効率が悪化することがある。そこで、「京」のOSでは、ページサイズを4 Mバイト/32 Mバイト/256 Mバイトから選択的に利用できるようにしている。

例えば、大域変数などの静的データ領域に比べ、スタックをより多く使用するアプリケーションを実行する場合は、スタックに大きなページを割り当て、データ領域に小さなページを割り当てるようにすることで、メモリの使用効率とTLBミス発生回数の軽減による性能向上の双方を両立させることができる（図-2）。

## ● スケーラビリティ向上

複数のノードにまたがる並列ジョブの性能向上（スケーラビリティ向上）を目的として、同期スケジューラ、統計情報採取機構を開発した。

### 【同期スケジューラ】

「京」のシステム上で動作する単一の計算ジョブは、複数ノード上に展開され、一定の計算処理が完了するたびにインターコネクトで接続されたノード間で互いに同期・通信を行う並列プログラ

ムである。

一方、システムの運用のためにはOSや運用ソフトウェアのデーモン（システムの管理のためにバックグラウンドで動作するプログラム）が動作する必要があるが、これらのデーモンはジョブとは関係なく非同期に動作する。

このデーモンの動作は、並列プログラムの動作の外乱（システムノイズ）となり、数万ノードの計算では外乱がノード数に比例して増大して大きな性能低下につながる場合がある。この問題を解決する方法として下記の二つの方法がある。

- (1) 複数CPUコアで構成されるノードの特定のコアをOS専用としてデーモンを動作させて各ノードでジョブへの外乱をなくす方式（空間的分割）
- (2) デーモンを計算ノード間で同期させて実行することで、並列プログラムの同期・通信への外乱をノード数に依存せずに一定に抑える方式（同期スケジューラ方式）（時間的分割）

(1)の方式では「京」のCPUが持つ8個のコアの内、1個をシステムに割り当て、ジョブに7個のコアを割り当てた場合、実行効率が87.5%（7/8）に制限されてしまうため、「京」では(2)の同期スケジューラ方式を開発した。

「京」の同期スケジューラではOSはTofuインターコネクトのバリア機能を利用してノード間で同期を取る。同期スケジューラではLinuxのプロセススケジューラの機能を拡張し、例えば100 ms間隔で同期を取り、その100 msの内1 msでデーモンを動作させる。並列プログラムは99%の時間で全てのノードで同期して外乱を受けずに動作することができ、ノード数に比例した性能の向上が図れる（図-3）。

### 【統計情報採取機能】

「京」では同期スケジューラによりデーモンによ

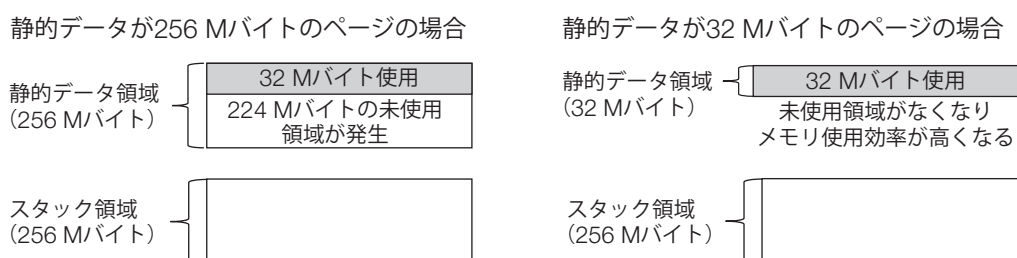


図-2 複数ページの効果(32 Mバイトの静的データを使用するプロセスの例)

るジョブへの影響を低減するとともに、デーモンの処理を最適化することで外乱自体を低減している。ここでは一つの事例として統計情報採取機能の改善を説明する。

計算ノードで、ユーザアプリケーションが動作するときに、システムで定期的に動作するデーモンプロセスがユーザアプリケーションの外乱（システムノイズ）になり、ユーザアプリケーションの実行性能の揺れ（ユーザアプリケーションの実行タイミングによって実行性能に差が出る）の原因となり得る。

このため、計算ノードでシステムノイズを削減するためには、システムで定期的に動作するデーモンプロセスなどを極力少なくすることが必要である。

システムが正常に動作していることを監視したり、性能上のボトルネックを分析したりするためにOSはCPUやメモリの利用状況などの統計情報を採取する。一般にOSの統計情報は、定期的にシステム処理を実行するための機能であるcron経由で、sadcコマンドを起動して、カーネルデータの読出しと統計情報ファイルへの書込みを行う。小規模のシステムでは、この方法でも大きなシステ

ムノイズにならないが、「京」は巨大なシステムであるのでこの統計情報採取処理もシステムノイズの要因となる。

「京」では計算ノードの性能統計情報をIOノードで収集するリモートsadc機能（rsadc）を開発した（図-4）。「京」ではラックごとにIOノードにIO装置を集約することで低コスト化と高信頼化を図っており、このIOノードに多数の計算ノードの性能統計情報のIO処理をオフロードすることで外乱の低減を実現している。

計算ノードのksadcデーモンは各ノードの性能統計情報をメモリ上に保持し、IO処理はIOノード上のrsadcにより行う。rsadcは各計算ノードの性能統計情報をTofuインターコネクトのリモートDMA（Direct Memory Access）機能により計算ノード上のデーモンの処理なしにIOノードのメモリに転送して、ディスクに保存することができる。

また、「京」では多数のラックで構成されるシステムにおいて、課金やシステム管理に必要なログファイルを、計算ノードへのアクセスは行わずに多数のラックのIOノードからログファイルを一つの制御ノードに収集する機能を開発し、並列ジョブへの外乱なしに、管理者がシステム全体の状況

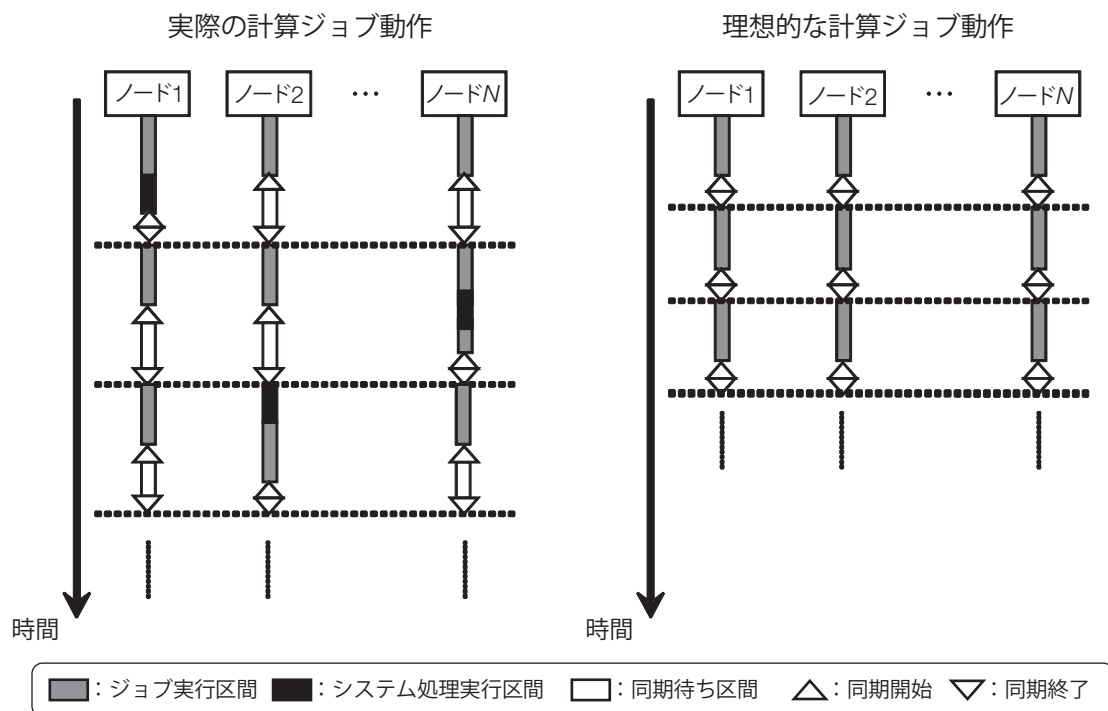


図-3 同期スケジューラの動き

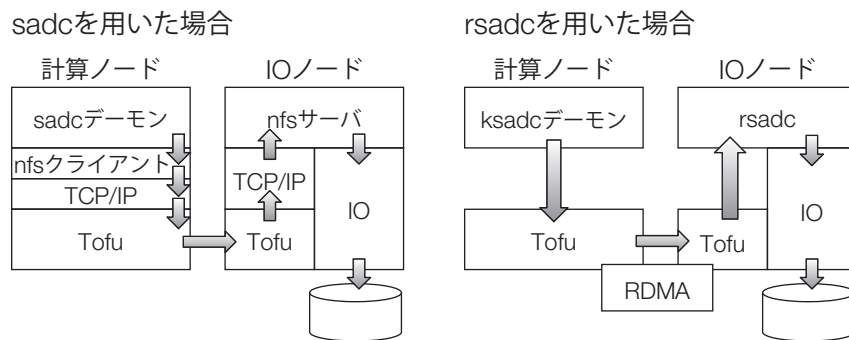


図-4 rsadcの模式図

を把握可能としている。

### 堅ろう性向上

「京」は多数のハードウェア部品から構成され、またいろいろなユーザから長時間の計算を行うことが求められるシステムであるため高い堅ろう性が必要になる。例えば、ハードウェアの故障に対しては、一つの部品の故障があっても運用を継続したり、故障箇所を早期に検出して交換したりするRAS (Reliability, Availability, Serviceability) 機能が重要である。また利用者のグループ内でファイルを共有する、あるいは、ほかのグループには参照させないセキュリティが求められる。

「京」では、ハードウェアの冗長化とLinuxが備えているマルチパスドライバなどのRAS機能やセキュリティ機能、および富士通の従来のスーパーコンピュータやサーバで強化してきたRAS機能をLinuxに追加することで高い堅ろう性を実現した。

以下ではRAS機能の強化とセキュリティ機能の例を説明する。

#### ● RAS強化

「京」のOSでは、従来の富士通のサーバの技術を継承して間欠的なメモリエラーのリカバリや、ハードウェア故障箇所の特定・通知機能を強化している。

$\alpha$  線などによって起こるメモリエラーに対し、「京」のOSではハードウェアのメモリパトロール機能と連携してデータ訂正を行うことで高い信頼性を実現している。

ハードウェアはOSが指定した間隔でメモリ読出し (パトロール) を行う。ECC (Error Correcting Code) の機構によりメモリデータの1ビットエラー

を検出した場合に、OSはメモリへのデータの書き込みを行うことでECCを含めたエラー訂正を行う。これを定期的に行うことにより、訂正可能な1ビットエラーが訂正不可能な2ビットエラーに進展することを防止して信頼性を向上させている。

OSは、故障ページを検出すると、後続のメモリ割当要求発生時に、そのページを割当てないようにマーキングを施すことでページの縮退を行う。

メモリエラーを検出した場合にOSはそのメモリの搭載スロットを特定して管理者に通知を行うことで、ノードの停止～部品の交換作業時間を短縮している。メモリだけでなく、CPU、IO装置 (PCIチャネル、GigabitEthernetアダプタ、FibreChannelアダプタなど) について検出したエラー事象から詳細な故障被疑箇所を特定できるようなRAS機能を強化した。

#### ● セキュリティ機能

「京」のシステムは、多数のユーザが同時に利用する。このため、セキュリティ機能は必須である。しかし、セキュリティ機能が計算性能に影響を与えては本末転倒である。

「京」のセキュリティ機能は、UNIX系OSで標準的に利用されているユーザ管理機能とファイルシステム機能を組み合わせて実現している。

OSは以下の一般的なユーザ管理による認証に関するセキュリティ機能をサポートする。

- ・パスワードの暗号化
- ・ファイルおよびディレクトリのアクセス権の変更
- ・ファイルおよびディレクトリの所有者の変更
- ・ファイルおよびディレクトリのグループ所有権の変更

ファイルおよびディレクトリは「読み込み」、「書

込み」,「実行」を自分, グループ, その他といった分け方でアクセスを制御するCAPP(Controlled Access Protection Profile) によるセキュリティ管理を行う。

### む す び

本稿では,「京」のOSの特長について述べた。

世界に類を見ない今回の大規模システムのOSの開発は, 性能の追求とユーザビリティの両立を目指したチャレンジングな取組みであった。「京」によって実施される大規模シミュレーションが様々な科学分野や産業の発展へとつながる基盤として, OSの開発に継続して取り組んでいく。

### 著者紹介



**師尾 潤** (もろお じゅん)

次世代テクニカルコンピューティング  
開発本部ソフトウェア開発統括部 所属  
現在, 次世代HPCのソフトウェア開発  
業務に従事。



**加藤丈治** (かとう たけはる)

次世代テクニカルコンピューティング  
開発本部ソフトウェア開発統括部 所属  
現在, 次世代HPCのソフトウェア開発  
業務に従事。



**山田雅彦** (やまだ まさひこ)

次世代テクニカルコンピューティング  
開発本部ソフトウェア開発統括部 所属  
現在, 次世代HPCのソフトウェア開発  
業務に従事。