

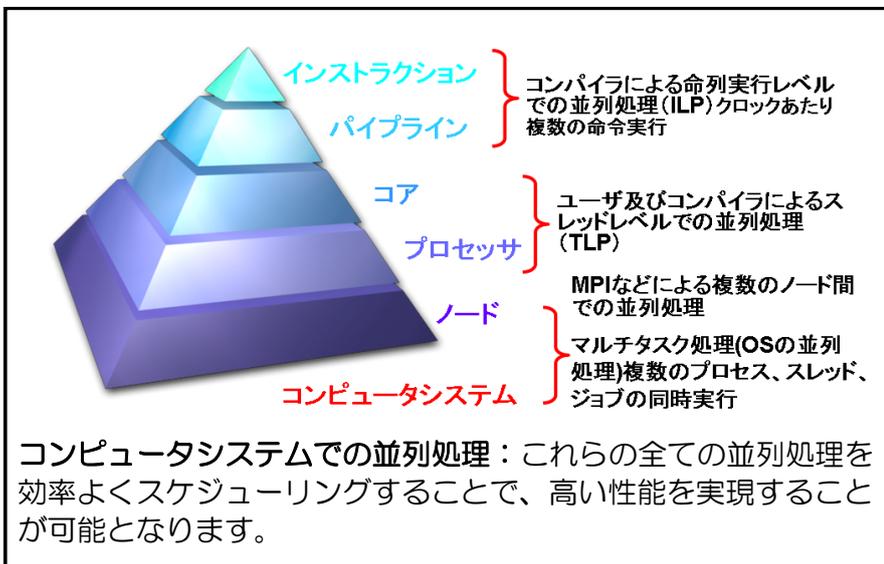


High Performance and Productivity  
 スケーラブルシステムズ株式会社



## OpenMPプログラミングトレーニング

現在のコンピュータシステムは、並列処理を基本として、そのシステムが構築されています。一般には並列処理とは、コンピュータで複数の処理を同時に進行させる事により、処理の高速化とその処理効率の向上をはかるために利用されています。コンピュータシステムには、非常に多くの機能が実装されており、ミクロのレベルからマクロのレベルまで、ユーザが利用可能な資源は、複数のコンポーネントから構成され多重化されています。それらを可能な限り同時に利用することで、高速化とより大規模な処理が可能となります。



このコンピュータの並列処理については、ユーザが全く意識することなく、ユーザが求める処理を分散・協調させた処理もコンピュータでは行われています。現在のコンピュータ・プラットフォームの頭脳となるプロセッサは、トランジスタ数が "ムーアの法則" に従い指数関数的に増加することで性能向上が図られ、命令レベルの並列性 (ILP: Instruction Level Parallelism) を高めるためのさまざまな高速化技術 (Out-of-Order 実行や命令の投機的実行、SIMD命令) などが開発され利用されています。これらの命令レベルの並列性の利用は、マイクロプロセッサ

自身が制御したり、また、コンパイラがその並列性を最大限に活用するための最適化を行うことで、ユーザがあまり意識することなくユーザのコードの高速実行に寄与しています。

また、デュアルコアとマルチコアのプロセッサ上では同時に複数のプロセスの実行が可能であり、オペレーティングシステムは、カーネルの実行、ドライバ、ライブラリ、アプリケーションのすべての処理プロセスを最適にスケジュールすることで、大幅な処理性能の向上を図ることを可能とします。更に、アプリケーションを再コンパイルしたり、新しいライブラリをリンクしたりすることで、その性能を大幅に向上させることが可能となります。アプリケーションの実行時に複数のスレッドを生成して、今まで逐次的に実行されていた処理を各スレッドが同時・並列に実行し、より短時間で処理を終了することが可能となるためです。

今後は、より多くのコアが一つのプロセッサ上に実装され、また、デュアルコアとマルチコアが、一般的なプロセッサとなることは間違いありません。デュアルコアとマルチコアの持つ大きな可能性とその能力を最大限に発揮するためのキーとなる、マルチスレッドでのアプリケーション開発のためのプログラミングへの関心は高まっています。

先に述べたようにデュアルコアとマルチコア上でのアプリケーションの高速化には、アプリケーションの実行時に、複数のスレッドが並列に処理を行うことが必要になります。ここで問題となるのはアプリケーションプログラムに対して、並列処理を適用する為の特別な作業やそのための開発工数が必要になるかということです。実際には、マルチスレッド化や並列化といった作業にはそれほどの時間を必要とするものではありません。マルチスレッド対応の開発ツールがあれば、これらの並列化は容易に行うことが可能です。プログラムの開発者やプログラマーは、プログラムの本質的なロジックを記述することに専念し、並列化については、既に高度に最適化・並列化されたライブラリを利用したり、並列化コンパイラの支援によって、プログラムのマルチスレッド化を図ることが現在では可能になっています。

その一つの方法として、OpenMPによるマルチスレッドプログラミングがあります。OpenMPはユーザがプログラムの並列化を指示する構文をプログラム中に記述することで、マルチスレッド並列プログラムを開発する枠組みを提供します。

プログラム開発者や研究者がプログラムを作るのは、そのプログラムの並列化を行う為ではありません。ある処理、解析を目的にプログラムを書き、そのプログラムをプラットフォームで効率良く、高速に実行できることを目的としています。これらのコンパイルツールは、開発者が本来のプログラムの開発目的である、これらのアルゴリズムの実装やロジックの検証のための作業に専念することを可能とし、並列化という必要ではありますが本質的ではない手間のかかる作業を開発者の代わりに担うものです。

スケラブルシステムズ株式会社では、マルチスレッドプログラミングのための技術情報の提供、プログラミングのトレーニング、利用技術に関するコンサルテーションなど幅広い活動を行います。

## 並列プログラミング (OpenMP) に関するトレーニング

### Part 1 並列プログラミングについて

- 並列プログラミングについての概要説明
- スレッドとプロセスの違いについて
- OpenMPと他のAPI (特にMPI)との違いについて
- 並列化アプリケーションの開発に際してのアプローチ

### Part 2 OpenMPの概要とAPIのご紹介

- 並列実行領域 (Parallel Regions) 構文
- ワークシェアリング (Worksharing) 構文
- データ環境 (Data Environment) 構文
- 同期 (Synchronization) 構文
- 実行時間関数/環境変数 (Runtime functions/environment variables)

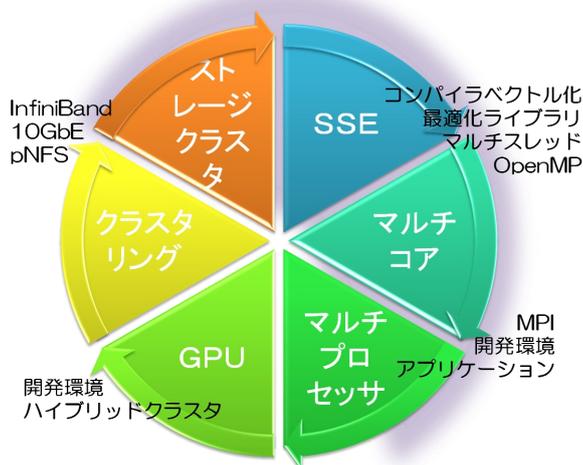
### Part 3 OpenMPプログラミング

- プログラミングでの注意事項の説明
- OpenMPでの最適化事例 (SPEC OMP)

### Part 4 OpenMPプログラムサンプルの紹介とデモ

### Part 5 OpenMPプログラミングに関するトピックスの紹介

詳細については、弊社にお問合わせください。



Scalable Systems Co., Ltd.

スケラブルシステムズ株式会社

〒102-0083 東京都千代田区麹町 3-5-2 BUREX麹町11F

Www.sstc.co.jp