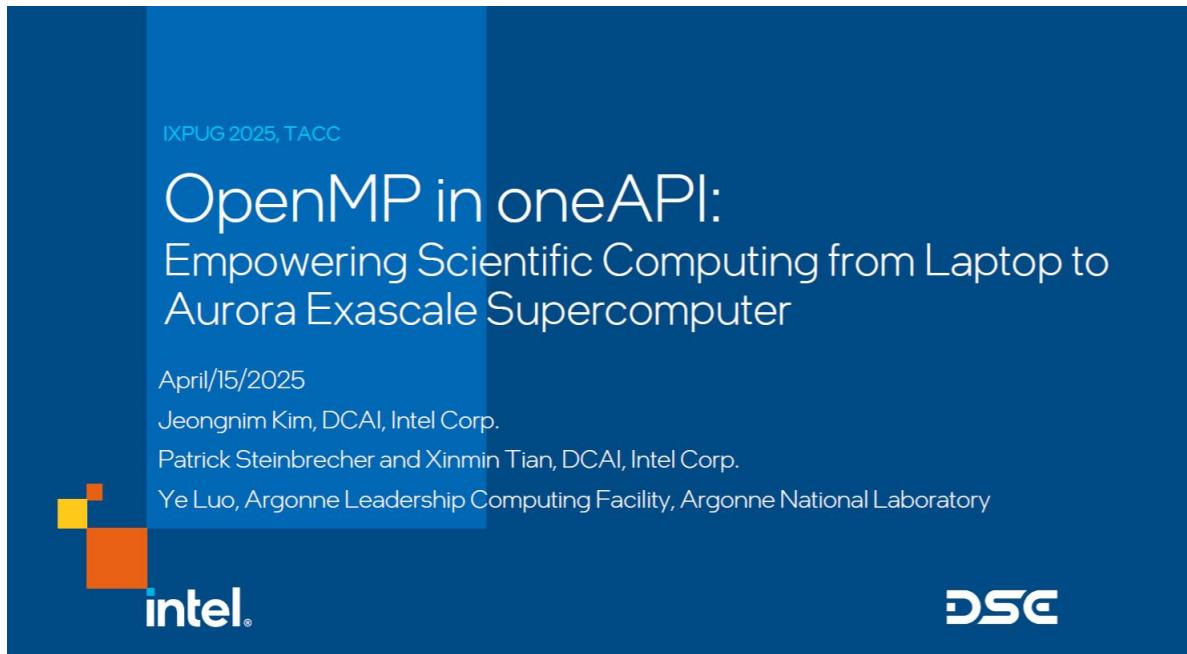


# OpenMP と科学計算の将来

この記事は、インテルのウェブサイトで公開されている「[IXPUG Take-Out: OpenMP\\* and the Future of Scientific Computing](#)」の日本語参考訳です。原文は更新される可能性があります。原文と翻訳文の内容が異なる場合は原文を優先してください。

Intel Extreme Performance Users Group (IXPUG) の 2025 年年次カンファレンス (英語) において、アルゴンヌ国立研究所のアルゴンヌ・リーダーシップ・コンピューティング・ファシリティーの Ye Luo 氏とインテルコーポレーションの Jeongnim Kim が「[OpenMP in oneAPI: Empowering Scientific Computing on Intel Platforms, From Laptop to Aurora Exascale Supercomputer \(oneAPI の OpenMP: ラップトップから Aurora エクスケール・スーパーコンピューターまで、インテルのプラットフォームで科学計算を強化する\)](#)」(英語) と題した講演を行い、OpenMP 並列プログラミング・モデルが既存のワークロードだけでなく、あらゆる階層型ヘテロジニアス・プラットフォームにおける高速オフロード・コンピューティングの将来においても重要な役割を担うことを説明しました。インテル® Core™ プロセッサー、インテル® Xeon® プロセッサー、その他の CPU はそれぞれが得意とする処理を実行でき、専用のアクセラレーターや GPU は高度に並列化可能な計算処理や信号処理タスクを支援します。そして、OpenMP を使用することで、さまざまな並列計算ストリームの調整を効率良く行うことができます。

この記事では、カンファレンスで発表された要点を紹介します。ビデオで詳細を確認されることをお勧めします。



⇒ テキサス先端計算センター (TACC) で開催された Intel Extreme Performance Users Group (IXPUG) の 2025 年年次カンファレンスの講演ビデオ「[OpenMP in oneAPI: Empowering Scientific Computing on Intel Platforms, From Laptop to Aurora Exascale Supercomputer \(oneAPI の OpenMP: ラップトップから Aurora エクスケール・スーパーコンピューターまで、インテルのプラットフォームで科学計算を強化する\)](#)」(英語)

⇒ [講演資料 \(英語\)](#)

# OpenMP 並列プログラミング・モデル

OpenMP は、ラップトップからスーパーコンピューターまで、さまざまなプラットフォーム向けの並列アプリケーションを開発する柔軟なインターフェイスをプログラマーに提供する、移植性と拡張性に優れたモデルとして設計されています。

主な機能は、以下のとおりです。

- ディレクティブベースの並列処理。C や Fortran 開発者には非常に自然に感じられるものです。
- 共有メモリーモデル。共有メモリードメインの唯一の制限は、オペレーティング・システムのサポート状況です。
- タスクとループの並列処理。既存のコードに並列処理を簡単に追加できます。
- デバイスオフロード。高速化とカスタムルーチン実行を実現します。
- アーキテクチャーとコンパイラー間の移植性。OpenMP のオープン・スタンダード・アプローチにより実現します。

ベクトルスカラー積を計算し、結果をベクトルに加算する典型的で単純な並列オフロードルーチンは、次のようになります。

```
void axpy(const float *a, const float *b, float *c, int n)
{
#pragma omp target teams distribute parallel for \
    map(to:a[0:n],b[0:n]) map(from:c[0:n])
    for (int id=0; id < n; id++)
    {
        c[id] = a[id] + b[id];
    }
}

// ...

float a[64]={1}, B[64]={2}, c[64];
axpy(a,b,c,64);
```

OpenMP は、1997 年 (Fortran) と 2002 年 (C/C++) の誕生以来、大きく進化し、2015 年には OpenMP 4.5 で target オフロード構文が追加されました。

講演の 7 分 25 秒の時点から、Yeongmin Kim が近年追加された OpenMP 仕様の主要な機能について説明しています (図 1)。

# OpenMP Specifications Highlights

## [Version 5, a Major Leap Forward](#)

- Memory management enhancements
  - *declare mapper* for custom data mapping
  - *allocate* to control memory allocations
- Tasking improvements
  - *taskloop* simplify parallel loop execution
  - *depend* clause enhancement
- Device offloading
  - Improved *target* for heterogenous computing
  - *require* to specify device capabilities
- Loop transformations
  - *loop* for flexible loop parallelization
  - *order(concurrent)* for safe execution
- Error handling
- Tools and debugging

## [Version 6.0 for Easier Programming](#)

- Enhanced device features
  - *workdistribute* directive divides the execution of array notation
  - improved support for unified memory models and asynchronous data transfers.
  - *target enter/exit* data for finer control over data movement.
- Improved loop transformations
  - Use of loop fusion, reversal and interchange.
- Greater user control of storage resources and memory spaces
- Support for parallelization of the latest C, C++ & Fortran language standards
- Interoperability with accelerator programming models, e.g. SYCL/DPC++

図 1: 近年追加された OpenMP 仕様

これらの追加機能の詳細については、OpenMP ウェブページの以下のプレスリリースを参照してください。

- ⇒ [OpenMP 5.0 is a Major Leap Forward \(OpenMP 5.0 は大きな飛躍\) \(英語\)](#)
- ⇒ [OpenMP® ARB Releases OpenMP 6.0 for Easier Programming \(OpenMP ARB がプログラミングを容易にする OpenMP 6.0 をリリース\) \(英語\)](#)

インテル® oneAPI DPC++/C++ コンパイラーとインテル® Fortran コンパイラーは、最新の OpenMP 機能のサポートにおいて最先端を走り続けています。これらのコンパイラーの開発チームは、OpenMP 標準と LLVM 互換のオープンソース・コンパイラーへの実装に積極的に貢献しています。

インテル® コンパイラーの OpenMP サポートの最新情報については、以下の記事を参照してください。

- ⇒ [インテル® ソフトウェア開発ツール最新情報](#)
- ⇒ [インテル® コンパイラーを利用した高度な OpenMP デバイスオフロード \(コンパイラー 2025.x における OpenMP 実装のサポートに関する説明\)](#)

## 強化された OpenMP コンポーネンティ

インテル® コンパイラー特有の機能として、さまざまな分散コンピューティングおよび GPU アクセラレーター・オフロード・フレームワークにまたがる科学計算や階層型ヘテロジニアス・マルチノード計算のコンポーネンティーに非常に重要な追加機能があります。

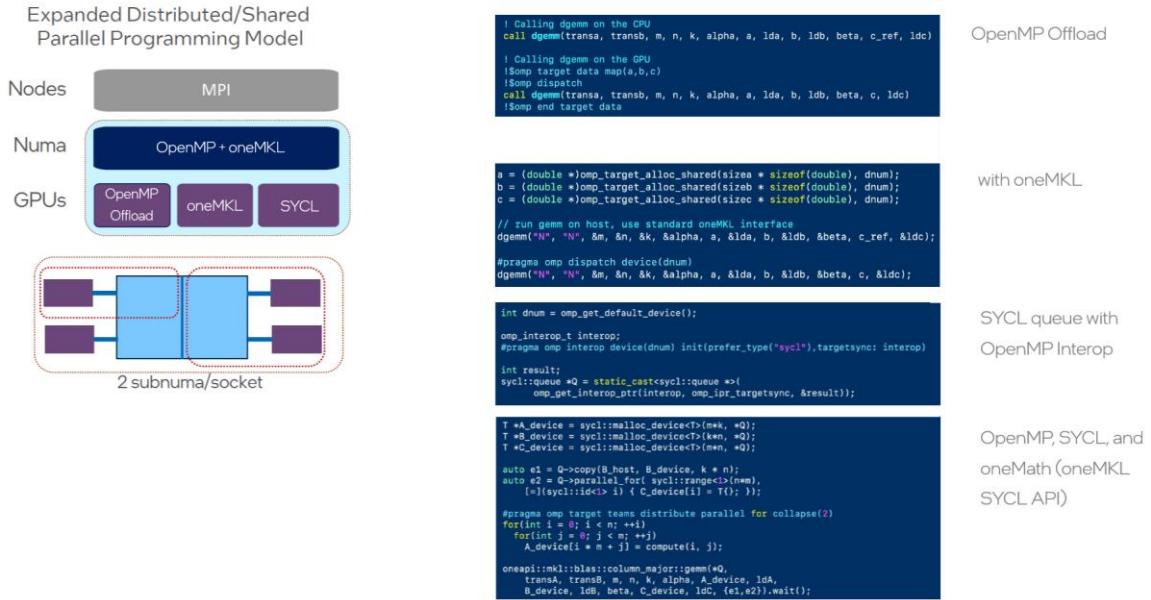


図 2: OpenMP マルチ API コンポーザビリティー

複数のソケット、複数のオフロード・ソフトウェア・プログラミング・モデル、そして異なる NUMA ドメインからなる環境を考えてみましょう。FPGA、カスタム・アクセラレーター、チップレット、GPU の人気が高まり、競合が激化するにつれて、メモリー・アーキテクチャーはさらに複雑になっています。

Jeongnim Kim は、講演の 11 分の時点から、インテル® oneAPI ツールキットにおける強化された OpenMP のコンポーザビリティーが、このような高度な計算アーキテクチャーにどのように役立つかについて説明しています。

図 2 は、oneMKL C API を使用した OpenMP オフロード呼び出しと、SYCL キューによる oneMath SYCL API 呼び出しを完全に連携させた gemm コードの例を示しています。このコードは、このレベルのコンポーザビリティーの中核となる原則を示しています。このレベルでは、パフォーマンス・ライブラリー（インテル® oneAPI マス・カーネル・ライブラリー、インテル® oneAPI コレクティブ・コミュニケーション・ライブラリーなど）に加え、SYCL ユーザーカーネル、MPI ノード、さらには Python 呼び出しも共存し、連携できます。

このアプローチは、最小限の変更で SYCL ライブラリーを OpenMP や MPI と併用できるようにします。これは、Kokkos (英語) などのマルチプラットフォーム科学計算フレームワークの OpenMP バックエンドや、C/C++ バインディングを使用する各種プログラミング・モデルに統合されています。もちろん、ほかの多層型ハイパフォーマンス・コンピューティング・スタックと同様に、複雑さが増すと、パフォーマンス・チューニングの要件も増加します。

GAMESS (英語)、VASP (英語)、BerkeleyGW (英語)、ThunderSVM (英語) など、このアプローチを採用しているアプリケーションや科学ワークコードは複数あります。

## QMCPACK OpenMP 移植性ケーススタディー

電子構造計算に特化した最新の高性能オープンソース量子モンテカルロ (QMC) シミュレーションである QMCPACK (英語) は、前述の方法で、MPI、OpenMP、SYCL、および oneMKL を組み合わせて強化された OpenMP コンポーザビリティーを活用し、ノードと GPU 間で計算リソースを効率良く管理しています。

講演の 19 分の時点から、Ye Luo 氏は QMCPACK における oneAPI による OpenMP 移植性のメリットについてさらに詳しく説明しています。oneAPI が提供する柔軟性により、シミュレーション・ワークLOADを Aurora 上で実行できるだけでなく、アルゴンヌ国立研究所の [アルゴンヌ・リーダーシップ・コンピューティング・ファシリティー \(ALCF\)](#) (英語) にあるほかのスーパーコンピューターから比較的容易にコードを移行できることを指摘しています。

oneAPI により強化された OpenMP コンポーネンツを活用することで、QMCPACK は米国エネルギー省 (DOE) が資金提供する [エクサスケール・コンピューティング・プロジェクト \(ECP\)](#) (英語) のすべてのプラットフォーム間で移植可能になります。

Ye Luo 氏は、パフォーマンスが一貫しており、計算ユニット間のデータ移動を注意深く考慮することで、機能的な移植性だけでなく、SYCL と OpenMP の混合環境をターゲットとする計算カーネルのパフォーマンス移植性も維持することが非常に容易であると強調しています。これにより、QMCPACK はチームの科学研究ニーズに応じて計算能力にアクセスし、その能力を最大限に活用することができます。

## OpenMP による並列コンピューティングのさらなる向上

OpenMP (英語) は、高速コンピューティング向けの堅牢なマルチベンダー、マルチプラットフォーム、マルチベース言語プログラミング・フレームワークであり続けています。OpenMP のアプローチはオープンで標準ベースであるため、ベンダーは基盤となるハードウェア向けにパフォーマンスが最適化された独自の実装を提供できます。

アプリケーション開発者は、これらの高レベルの OpenMP プログラミングの概念と手法を実装するだけで、GPU ハードウェアの潜在能力を最大限に発揮する、移植性に優れたハイパフォーマンスのコードを作成できます。さらに、強化された OpenMP コンポーネンツにより、OpenMP プログラミング・フレームワークはそのままに、より大きなコードベースを統合および移行して、SYCL ベースのヘテロジニアス・オフロードをサポートし、共存できるようになります。

インテルは、このような取り組みの最前線に立ち、自社の OpenMP ランタイム・ライブラリー、[インテル® Fortran コンパイラ](#)ー、および[インテル® oneAPI DPC++/C++ コンパイラ](#)ーにこれらの機能を統合できることを嬉しく思っています。インテルはオープン・スタンダードの精神を尊重し、OpenCL、SYCL、OpenMP などのアクセラレーテッド・コンピューティング・フレームワークの多くのオープンソース・イニシアチブと、LLVM や中間 SPIR-V 抽象化レイヤーなどのコンパイラ・イニシアチブに貢献しています。

### コンパイラのダウンロード

[インテル® Fortran コンパイラ](#)ーと[インテル® oneAPI DPC++/C++ コンパイラ](#)ーは、[こちらから](#)ダウンロードできます。

[インテル® HPC ツールキット](#)や[インテル® oneAPI ベース・ツールキット](#)の一部として入手することもできます。これらのツールキットには、基本ツール、ライブラリー、解析ツール、デバッグツール、コード移行ツールが含まれています。

LLVM コンパイラ・プロジェクトへの貢献は、[GitHub](#) (英語) を参照してください。

## 関連情報

- Fortran と OpenMP でヘテロジニアス・プログラミングの課題を解決
- The Parallel Universe
- OpenMP ターゲットオフロード [1:06:07] (英語)
- OpenMP サポート: インテル® Fortran コンパイラー・デベロッパー・ガイドおよびリファレンス (英語)
- OpenMP サポート: インテル® oneAPI DPC++/C++ コンパイラー・デベロッパー・ガイドおよびリファレンス (英語)
- OpenMP 仕様 (英語)

---

## 製品および性能に関する情報

<sup>1</sup> 性能は、使用状況、構成、その他の要因によって異なります。詳細については、<http://www.intel.com/PerformanceIndex/> (英語) を参照してください。