インテルのサンプルを使用して DPC++ を開始する

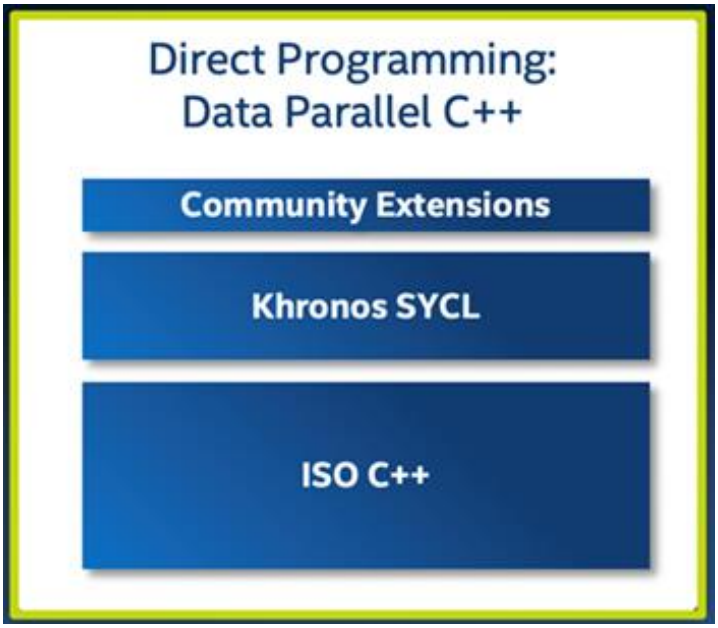
この記事は、インテル® デベロッパー・ゾーンで公開されている「Explore DPC++ with Samples from Intel*」の日本語参考訳です。

バージョン：2021.1
更新日：12 / 04 / 2020

DPC++ の概要

データ並列 C++（DPC++）アプリケーションは、並列処理を行う C++ プログラムです。DPC++ は、データ並列プログラミングとヘテロジニアス・コンピューティング向けに設計されています。DPC++ は、CPU、GPU、FPGA、およびアクセラレーターに一貫したプログラミング言語（C++）と API を提供します。各アーキテクチャーは、個別にまたはまとめてプログラムして使用できます。これにより、開発者は一度習得するだけで、異なるアクセラレーター用のプログラムを作成することができます。各アクセラレーターでは、最高のパフォーマンスを発揮するためアルゴリズムの適切な定式化とチューニングが必要ですが、言語とプログラミング・モデルは、ターゲットデバイスに関係なく一貫しています。

DPC++ は、Khronos Group の SYCL* をベースにしており、データ並列処理とヘテロジニアス・プログラミングをサポートします。また、インテルでは、顧客のコードに価値を提供することを目的に SYCL* の機能を拡張し、標準化団体と連携して採用に向けた取り組みを行っています。例えば、DPC++ 言語には、ホストとアクセラレーター間のメモリー使用を容易にする統合共有メモリーの実装が含まれます。これらの機能は、SYCL* 言語の将来のバージョンに組み込まれる予定です。SYCL* の詳細は、SYCL* 仕様 1.2.1（英語）を参照してください。



DPC++ と仕様の詳細は、oneAPI.com（英語）サイトにあります。

このガイドの目的は、開発者が oneAPI プログラミング・モデルを使用してプログラミングし、最適なアプリケーション・パフォーマンスを達成するため適切なアーキテクチャーをターゲットにして最適化するのを支援することです。

サンプル・プロジェクトのビルドと実行

以下は、インテル® oneAPI ベース・ツールキット導入ガイドのコマンドラインと IDE の説明へのリンクです。

- コマンドラインを使用したサンプル・プロジェクトのビルドと実行：
  - Linux*（英語）
  - Windows*（英語）
- IDE を使用したサンプル・プロジェクトのビルドと実行：
  - Linux*（Eclipse* または Visual Studio* コード）（英語）
  - Windows（Visual Studio*）（英語）
- GitHub*（後述の各サンプル のセクションに対応する Git リポジトリーへのリンクがあります）

サンプル 1：単純なデバイスオフロード構造

サンプル 1 は、データ並列プログラムの「Hello, World!」とも言えるベクトル加算を使用します。このプログラムは、オフロードデバイスをターゲットにする方法を示し、DPC++ アプリケーションの基本構造を提供します。サンプル 1 は、メモリーを管理する方法の例として、バッファーと統合共有メモリー（USM）を使用した 2 つの異なるソースファイルを提供します。

ベクトル加算は、GPU と FPGA の両方のデバイスセレクターを提供します。

このサンプルでは、DPC++ の基本要素（機能）を使用して、1D 配列を使用した単純な計算をアクセラレーターにオフロードする方法を学びます。次の基本機能を使用します。

- 1 次元配列データ。
- デバイス・セレクター・キュー、バッファー、アクセサー、およびカーネル。
- バッファーとアクセサーまたは USM を使用したメモリー管理。

サンプルコードの詳細は、ベクトル加算サンプルコード（英語）を参照してください。

次のサンプルを使用します。

- CLI または IDE サンプル名：**vector-add**
- ベクトル加算サンプルの Git リポジトリー（英語）

サンプル 2：基本的な DPC++ 機能の定義

このサンプルでは、2 次元ステンシルを使用して 2 次元の等方性媒体中を伝播する波をシミュレーションし、次の DPC++ の基本機能をステップごとに説明します。

- DPC++ キュー（デバイスセレクターと例外ハンドラーを含む）。
- DPC++ バッファーとアクセサー。

- カーネル定義内での関数呼び出しとアクセサー引数のポインター渡し。カーネル内の関数呼び出しは、シングル・タイム・ステップの計算処理（グローバル ID 変数で指定された格子点の更新）を実行します。

サンプルコードの詳細は、等方性媒体中の 2 次元有限差分波動伝搬（ISO2DFD）サンプルコード（英語）を参照してください。

次のサンプルを使用します。

- CLI または IDE サンプル名：**iso2dfd_dpcpp**
- ISO2DFD サンプルの Git リポジトリー（英語）

サンプル 3：複雑なアプリケーションの最適化

このサンプルコードでは、前述のサンプルで紹介した DPC++ の概念を拡張し、複雑なステンシル計算を 3D で解く方法を説明します。グリッドサイズを 2D から 3D に変更することで、非効率なデータ・アクセス・パターン、低 B / F 比、低占有率に関連する一般的な GPGPU（デバイス）プログラミングの問題を発見できます。このサンプルコードでは、DPC++ の機能を使用してこれらの根本的な問題に対処し、パフォーマンスを最適化する方法を学べます。以下が含まれます。

- DPC++ ローカルバッファーとアクセサー（各 DPC++ ワークグループがアクセスおよび管理するローカル・メモリー・バッファーを宣言）。
- 共有ローカルメモリー（SLM）を最適化するコード。
- DPC++ カーネル（parallel_for 関数と nd-range<3> オブジェクト）：
    - DPC++ キュー（カスタム・デバイス・セレクターと例外ハンドラーを含む）。

次のサンプルを使用します。

- CLI または IDE サンプル名：**iso3dfd_dpcpp**
- ISO3DFD サンプルの Git リポジトリー（英語）

サンプル 4：同期する

サンプル 2 と 3（規則的なグリッド上のステンシルカーネル）と比較すると、このサンプルは、多数の移動粒子と固定グリッドのセルとの相互作用により複雑さが増しています。このサンプルでは、同期（アトミック操作）などの新しい DPC++ 機能を説明します。

次の DPC++ ツールを使用して計算をアクセラレーターにオフロードする方法を学べます。

- DPC++ キュー（デバイスセレクターと例外ハンドラーを含む）。
- DPC++ バッファーとアクセサー（ホストとデバイス間のデータ通信）。
- DPC++ カーネル（parallel_for 関数と range<1> オブジェクト）。
- 同期のための DPC++ アトミック操作。
- API ベースのプログラミング：oneMKL を使用した乱数生成。

サンプルコードの詳細は、粒子拡散サンプルコード（英語）を参照してください。

次のサンプルを使用します。

- CLI または IDE サンプル名：**particle-diffusion**
- 粒子拡散サンプルの Git リポジトリー（英語）

次のステップ

チュートリアル

次のチュートリアルをお試しください。

- ベクトル加算サンプルを使用した DPC++ の基礎
- マンデルブロ・サンプルを使用した USM（DPC++ プログラミングの基本機能）

オフロードするコードの決定

インテル® Advisor を使用して、アクセラレーターにオフロードすることでメリットが得られるコード領域を判断できます。オフロード・アドバイザー機能では、標準のプロファイル機能に加えて、パフォーマンス予測データを収集できます。ターゲットデバイスにオフロードすることで、CPU ベースのアプリケーションのパフォーマンスを高速化するコードを判断します。「インテル® Advisor 導入ガイド」は、以下の作業に役立ちます。

- ルーフライン解析を使用して CPU コードや GPU コードのメモリー使用と計算処理を最適化。
- ベクトル並列処理を増やし効率を向上。
- 複数のスレッド設計のモデル化、チューニング、テスト。
- ヘテロジニアス・アルゴリズムを使用したデータフローや依存関係計算の作成と解析。

CUDA* コードから DPC++ コードへの変換

インテル® DPC++ 互換性ツールと呼ばれる移行エンジンを使用して、CUDA* コードを標準ベースの DPC++ コードに移行できます。導入ガイド（英語）とユーザーガイド（英語）は、既存の CUDA* アプリケーションの移行を支援し、移行プロセスの一般的なワークフローをカバーしています。このツールを使用して、複数のソースファイルとヘッダーファイルで構成されたプログラムを移行できます。また、このツールは以下を提供します。

- カーネルと API 呼び出し用のワンタイムの移行ポート。
- インテル® oneAPI DPC++ / C++ コンパイラーでコンパイル可能な、出力の生成に使用されるインラインのコメントガイド。
- 操作を効率化するコマンドライン・ツールと IDE プラグイン。

関連情報

インテル® oneAPI ツールキットのトレーニング・サイト（英語）では、DPC++ とサポートツールに関する詳細な情報、幅広いチュートリアル、ビデオ、ウェビナーを利用できます。

ドキュメント	説明
インテル® oneAPI プログラミング・ガイド(英語)	oneAPI と DPC++、プログラミング・モデル、プログラミング・インターフェイス、DPC++ ランタイム、API、ソフトウェア開発プロセスに関する情報が得られます。
導入ガイド(英語)	導入ガイドには、より詳細な情報が含まれています。
チュートリアル(英語)	チュートリアルでは、各機能が詳しく説明されています。

法務上の注意書き

インテルのテクノロジーを使用するには、対応したハードウェア、ソフトウェア、またはサービスの有効化が必要となる場合があります。

絶対的なセキュリティーを提供できる製品またはコンポーネントはありません。

実際の費用と結果は異なる場合があります。

本資料は、(明示されているか否かにかかわらず、また禁反言によるとよらずにかかわらず)いかなる知的財産権のライセンスも許諾するものではありません。

本資料で説明されている製品には、エラッタと呼ばれる設計上の不具合が含まれている可能性があり、公表されている仕様とは異なる動作をする場合があります。現在確認済みのエラッタについては、インテルまでお問い合わせください。

インテルは、明示されているか否かにかかわらず、いかなる保証もいたしません。ここにいう保証には、商品適格性、特定目的への適合性、および非侵害性の黙示の保証、ならびに履行の過程、取引の過程、または取引での使用から生じるあらゆる保証を含みますが、これらに限定されるわけではありません。

製品とパフォーマンス情報

[1] 実際の性能は利用法、構成、その他の要因によって異なります。詳細は、www.Intel.com/PerformanceIndex(英語)を参照してください。