インテル® VTune™ プロファイラーを使用して インテル® GPU 向けにアプリケーションを最適化

この記事は、インテル® デベロッパー・ゾーンに公開されている「Optimize Applications for Intel® GPUs with Intel® VTune™ Profiler」の日本語参考訳です。

この記事は、インテル® GPU (計算、メディア・クライアント、およびサーバー・アプリケーション向け) をターゲットとするアプリケーションのチューニングと最適化に関するガイドを提供します。インテル® VTune™ プロファイラーの GPU プロファリング機能に注目します。

目次

- GPU 計算リソースの利点
- インテル® VTune™ プロファイラーの GPU プロファイルと最適化ワークフロー
- GPU プロファイル用にシステムを準備
- パフォーマンスのサマリーを取得
- CPU 上でプロファイルおよび最適化
- CPU 上で最適化
- GPU 上でプロファイルおよび最適化
- GPU 上で最適化
- OpenMP* オフロードを使用する GPU 依存のアプリケーションをプロファイル
- OpenCL* アプリケーションのプロファイル
- DPC ++アプリケーションのプロファイル
- まとめ

GPU 計算リソースの利点

グラフィックス処理ユニット (GPU) は、強力なハードウェア・コンポーネントであり、CPU を使用するアプリケーションやシステムのパフォーマンスを大幅に向上できます。本来 GPU は 3D グラフィックスのレンダリングをアクセラレートするように設計されていますが、次の 2 つの利点により、さまざまな分野におけるハイパフォーマンスな計算タスクの汎用的な選択肢となっています。

- GPU はプログラマブルであり、ゲーム、クリエイティブ・プロダクション、AI など 3D グラフィックスのレンダリング以外のアプリケーションを高速化できます。
- GPU は、大きなデータブロックを使用して並列処理を実行し、マシンラーニングなどの分野に強力な計算能力を提供します。

インテル® グラフィックスのパフォーマンスは、GPU 機能の利点を活用して CPU の役目を削ぐことなく、GPU の機能を「パフォーマンス・アクセラレーター」として使用します。つまり、GPU にオフロードするか、CPU で実行を続行するかを選択することで、システムのパフォーマンスに大きな影響を及ぼす可能性があります。

GPU にオフロードすることを決定したら、インテル® VTune™ プロファイラーを使用してパフォーマンスと効率を調査します。このツールは、アプリケーションがシステムで利用可能なハードウェア・リソースからどれくらい利益を得られるか判断するのに役立つ、汎用的なパフォーマンス解析ツールです。

注:
GPU は、グラフィックスをレンダリングしたり (ビデオまたはゲーム・アプリケーションで)、多様な用途で大量の計算を実行できるデバイスですが、この記事では、計算集約型 GPU アプリケーションのプロファイルに注目して説明します。

ツールの選択

- インテル® グラフィックス上の GPU 計算およびメディア解析向けインテル® VTune™ プロファイラー
- ゲーム・アプリケーションのグラフィック・プロファイル向けインテル® グラフィックス・パフォーマンス・アナライザー (インテル® GPA) (英語)

GPU 計算プロファイルについて

- 『oneAPI GPU 最適化ガイド』の最適化ツールの説明
- インテル® Advisor のオフロードのモデル化機能 (英語)
- インテル® Advisor の GPU ルーフライン解析機能 (英語)

グラフィックスのプロファイルについて

- インテル® グラフィックス・パフォーマンス・アナライザー・クックブックの「インテル® プロセッサー・グラフィックス向けのパフォーマンス最適化」(英語) レシピ
- 同クックブックの「第 11 世代インテル® プロセッサー・グラフィックス向けの 3D レンダリング・アプリケーションの開発と最適化」(英語) レシピ

インテル® VTune™ プロファイラーの GPU プロファイルと最適化ワークフロー

1. GPU プロファイル用にシステムを準備します。
2. パフォーマンスのサマリーを取得します。
3. CPU 上でプロファイル、解析および最適化を行います。GPU オフロード解析を実行します。解析結果と**データ転送メトリック**を使用して以下を特定します。
    - アプリケーションが CPU 依存であるか GPU 依存であるか。アプリケーションが CPU 依存である場合、解析のガイドに従って GPU オフロードを改善して、GPU 依存にします。
    - GPU へのオフロードが最適であるか。
    - パフォーマンスが重要であり、最適化が必要な GPU カーネル。
4. GPU 上でプロファイル、解析および最適化を行います。GPU 計算/メディア・ホットスポット解析を実行します。
    - 最も時間を消費する GPU カーネルの特徴付けを行います。命令タイプごとに動的命令数の内訳を取得するか、GPU ハードウェア・メトリックをベースに GPU 利用率の効率を確認します (**Characterization (特性化)** モード)。

- パフォーマンスが重要な基本ブロックのデータ、および GPU カーネルのメモリーアクセスで発生した問題に関するデータを特定します (**Source Analysis (ソース解析)** モード)。

GPU プロファイル用にシステムを準備

インテル® VTune™ プロファイラーを使用して GPU 解析を行うため、関連するアクセス許可がありドライバーがシステムにインストールされていることを確認します。

関連資料

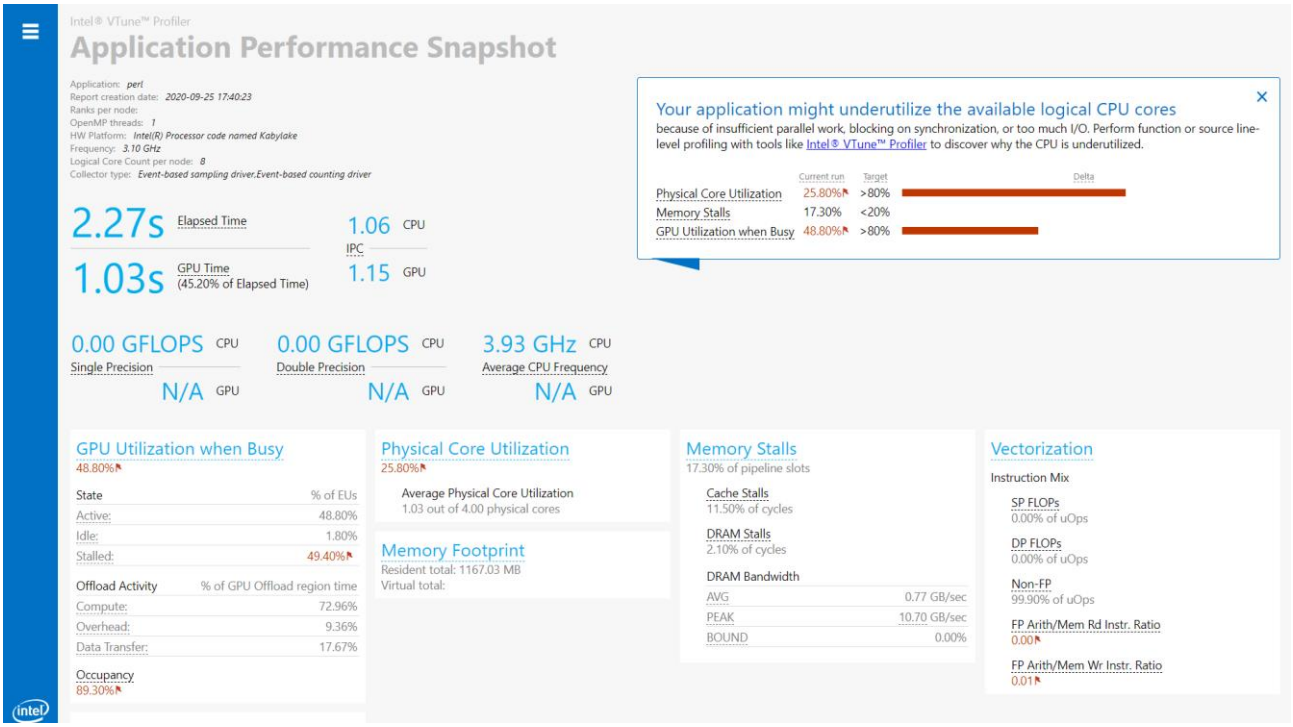
- GPU 解析向けにシステムを設定する方法の手順
- Linux* ディストリビューション用に汎用 GPU (GPGPU) 機能を有効にするインテル® ソフトウェアのインストール、展開、および更新 (英語) に関する情報

パフォーマンスのサマリーを取得

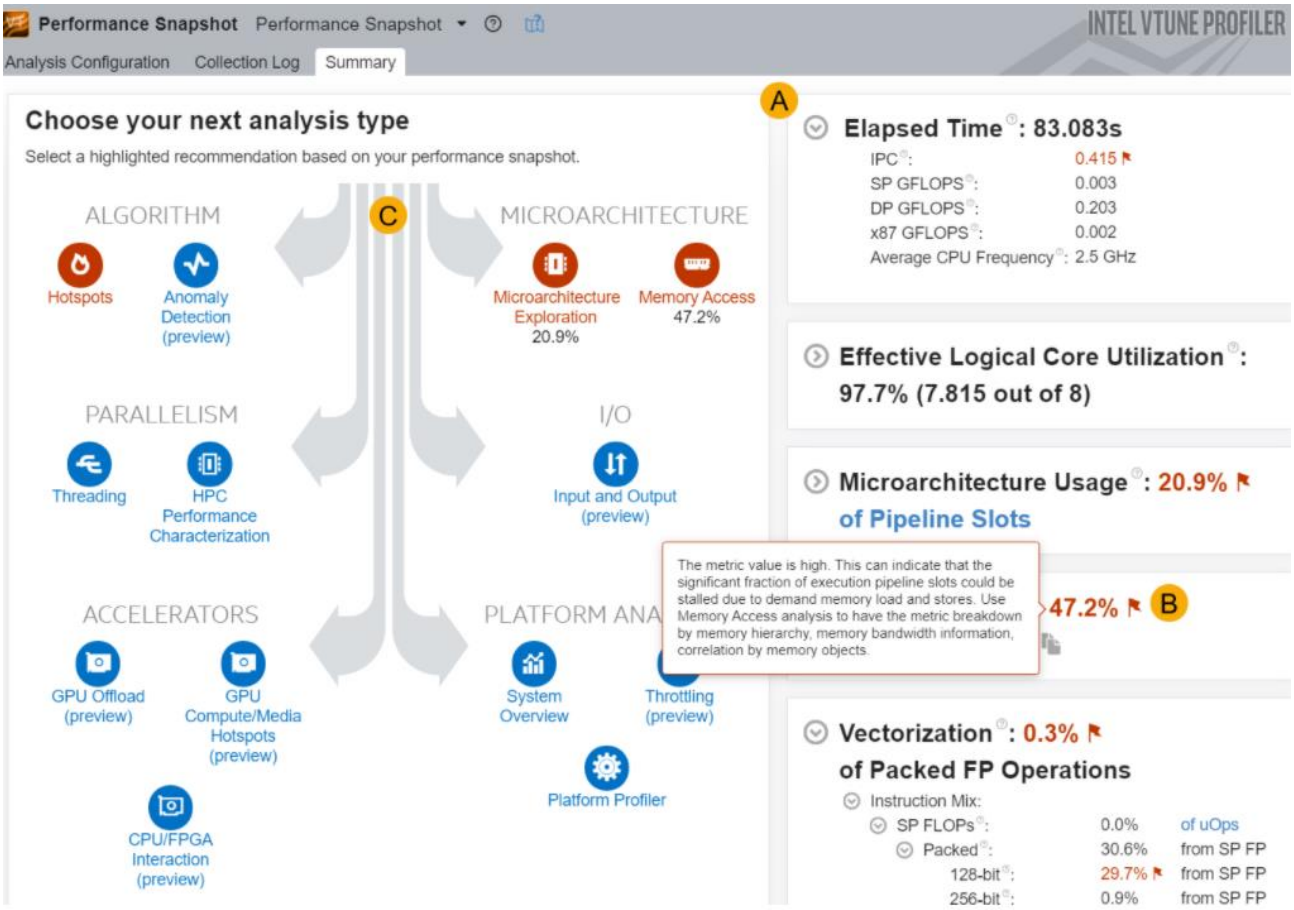
インテル® VTune™ プロファイラーのアプリケーション・パフォーマンス・スナップショット (Linux*) (英語) やパフォーマンス・スナップショットでアプリケーションに関連するパフォーマンスの問題のサマリーを取得します。

ツールの選択

- Linux* システムで MPI アプリケーションを最適化する場合は、**アプリケーション・パフォーマンス・スナップショット (Linux* システムのみ)** を選択します。軽量 GUI 使用するかコマンドラインから実行します。
- その他のアプリケーションでは、**パフォーマンス・スナップショット**を実行します。インテル® VTune™ プロファイラーの解析ツリーからこの解析を実行します。



アプリケーション・パフォーマンス・スナップショットの結果のサマリー (Linux* のみ)



インテル® VTune™ プロファイラーのパフォーマンス・スナップショットの結果のサマリー

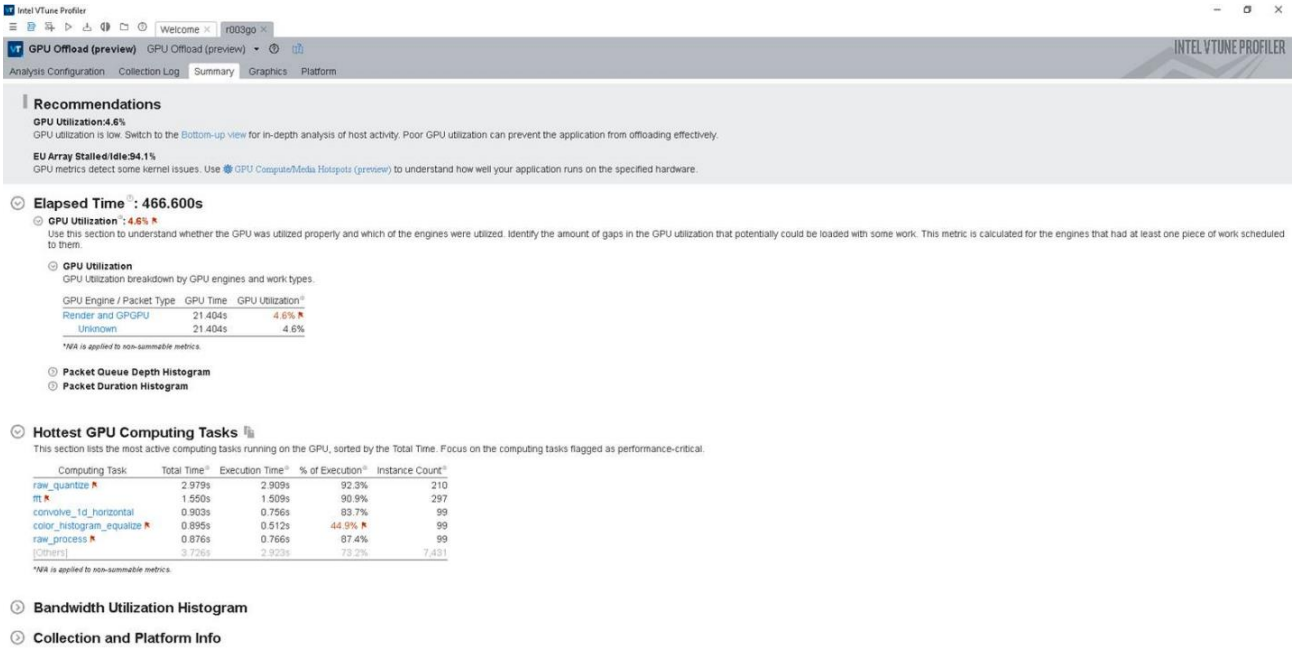
関連資料

- アプリケーション・パフォーマンス・スナップショット入門 (Linux* 版)
- インテル® VTune™ プロファイラーのパフォーマンス・スナップショット

CPU 上でプロファイルおよび最適化

インテル® VTune™ プロファイラーで GPU オフロード解析を実行して、アプリケーションが CPU 依存であるか GPU 依存であるかを確認します。また、アプリケーションがコードを GPU に効率良くオフロードしているか確認できます。解析結果とデータ転送メトリックを使用して以下を特定します。

- アプリケーションが CPU 依存であるか GPU 依存であるか。アプリケーションが CPU 依存である場合、解析のガイドに従って GPU オフロードを改善して、GPU 依存にします。
- GPU へのオフロードが最適であるか。
- パフォーマンスが重要であり、最適化が必要な GPU カーネル。



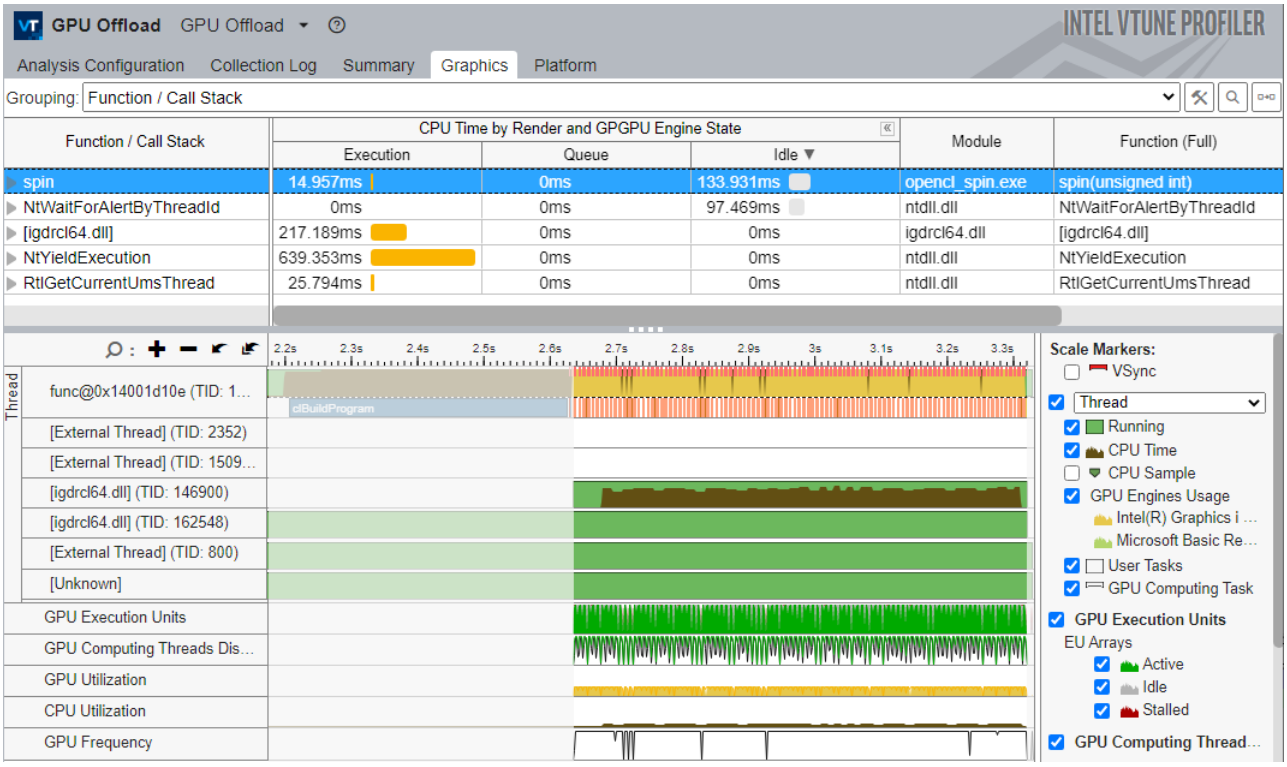
インテル® VTune™ プロファイラーの GPU オフロード解析の結果のサマリー

注:
システムに複数の GPU が搭載されている場合、GPU オフロード解析を設定する際に **[Target GPU (ターゲット GPU)]** オプションを使用してプロファイルする GPU を選択します。

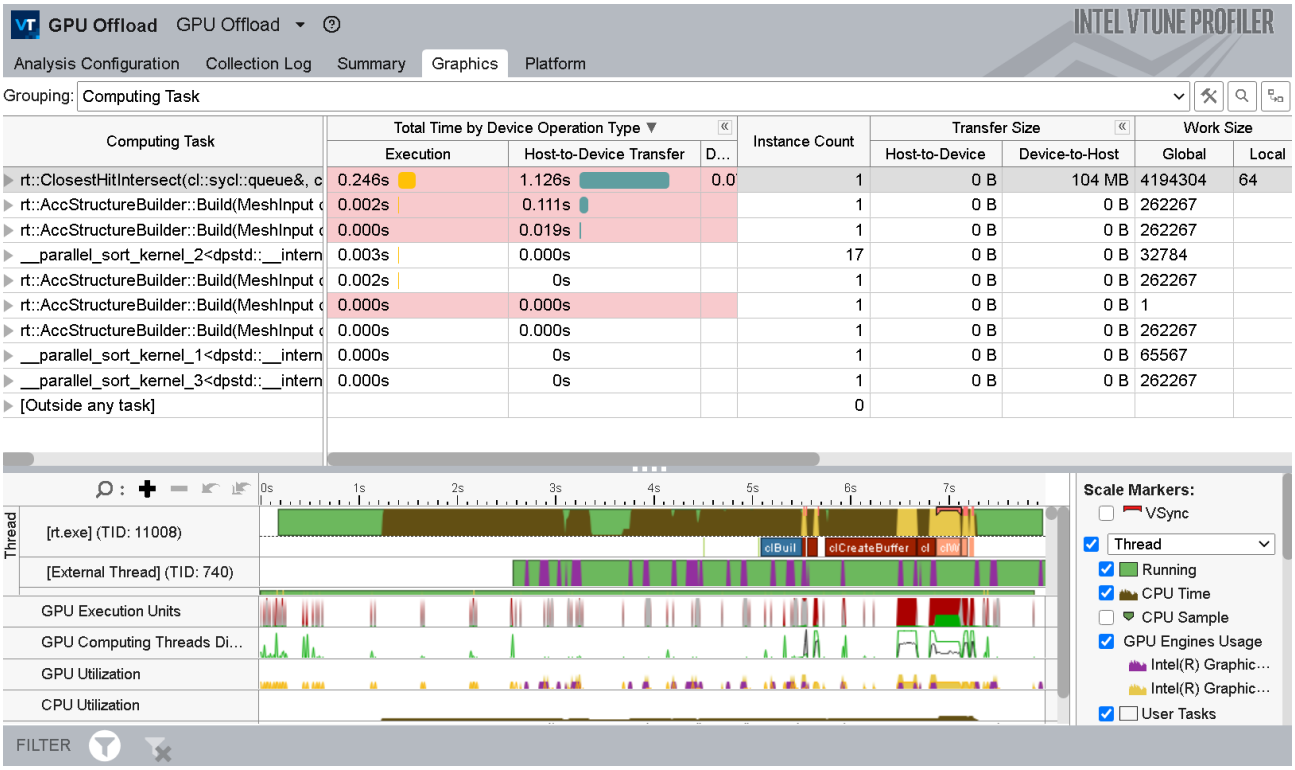
CPU 上で最適化

GPU オフロード解析の結果からデータ転送メトリックを使用して、ホストとデバイス間の転送操作にかかった時間を確認します。

- 解析のサマリーには実行時間とデータ転送時間が示されます。実行時間が短い場合、アプリケーションのオフロード構造が最適でないことを示しています。
- 選択したグループ化レベルに応じて、**[Graphics (グラフィックス)]** タブに収集されたデータが表示されます。
    - **Function/Call Stack (関数/コールスタック)]** グループ化レベルには、GPU がアイドル状態のときに CPU で実行された最もアクティブな関数が表示されます。
    - **[Computing Task (計算タスク)]** グループ化レベルでは、カーネルごとの計算タスクの合計処理時間の内訳と、データ転送サイズが表示されます。

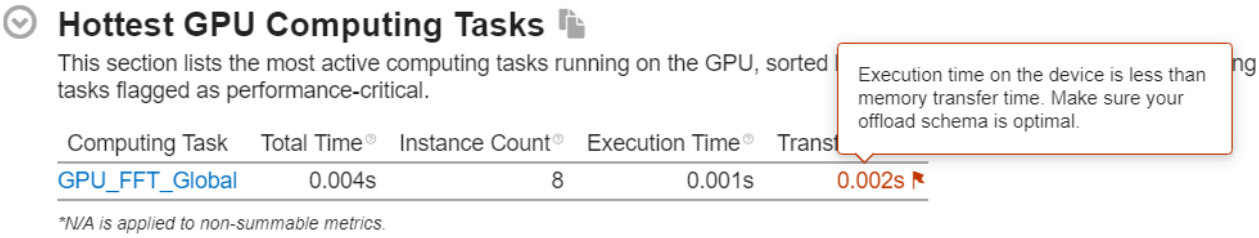


GPU オフロード解析の関数/コールスタックのグループ化



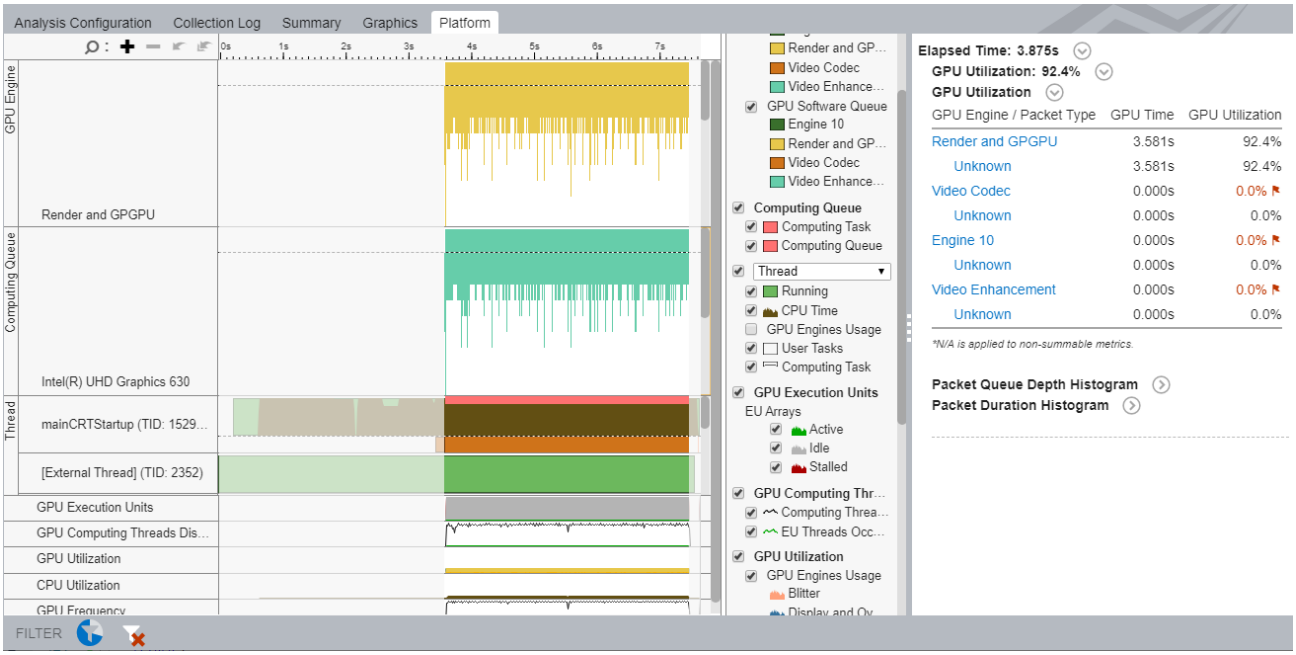
GPU オフロード解析の計算タスクのグループ化

解析が提供するガイドは、パフォーマンスを制限する不要なメモリー転送を特定するのに役立ちます。アプリケーションのパフォーマンスに重要な GPU カーネルを特定して解析を完了します。



GPU オフロード解析で表示される最もホットな計算タスクの内訳

オフロードステージに注目するには、**GPU Utilization (GPU 利用率)** メトリックを参照します。データを時間でフィルターすると、**[Platform (プラットフォーム)]** タブのコンテキストの概要にこのメトリックが表示されます。



[Platform (プラットフォーム)] タブの GPU 利用率メトリック情報

- このメトリックが高いかどうか確認してください。
- 同様に、EU がほとんどの領域でアクティブ/アイドル状態であるかを確認します。アイドル状態の EU は、小さなカーネルをスケジュールする際にオーバーヘッドが生じる可能性があります。これは、カーネル・インスタンスの期間をカーネルのコマンドバッファーの期間と比較することで確認できます。
- カーネルのすべての呼び出しに影響するスケジュールのオーバーヘッドを軽減するため、それぞれのカーネル・インスタンスが処理するデータサイズを増やすことを検討してください。

関連資料

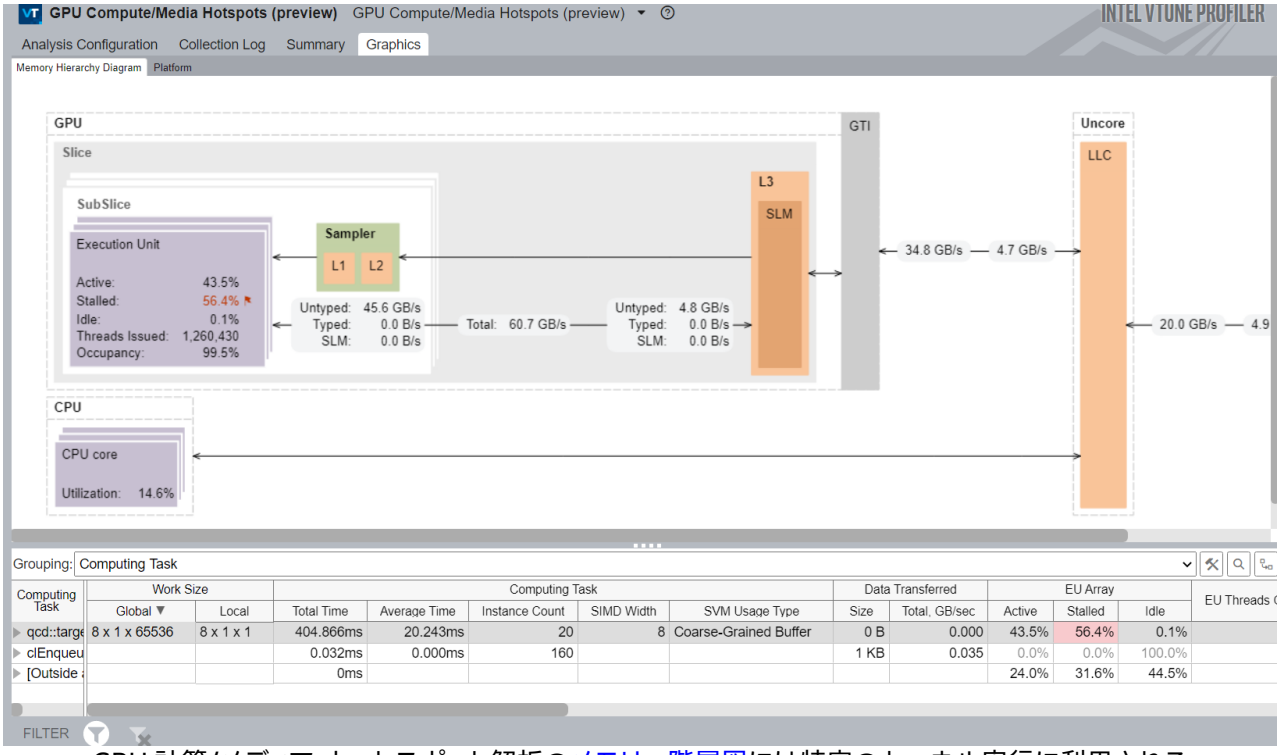
- GPU オフロード解析
- oneAPI GPU 最適化ガイド

GPU 上でプロファイルおよび最適化

次の問いに当てはまる場合、GPU 上の解析を検討してください。

- CPU 上でアプリケーションは十分に最適化されていますか?
- CPU - GPU 間の相互作用は妥当なものですか?
- GPU は長時間ビジー状態ですか?
- アプリケーションは GPU 依存ですか?

CPU での作業が終了したら、GPU 計算/メディア・ホットスポット解析を実行します。GPU オフロード解析で特定したパフォーマンス上重要なカーネルを調査します。利用率の高い GPU カーネルは、必ずしも効率良く使用されるとは限りません。GPU 計算/メディア・ホットスポット解析は、GPU の使用を特性化し、非効率である原因を特定するのに役立ちます。



GPU 計算/メディア･ホットスポット解析のメモリー階層図には特定のカーネル実行に利用される GPU のメモリー･アーキテクチャーが示される

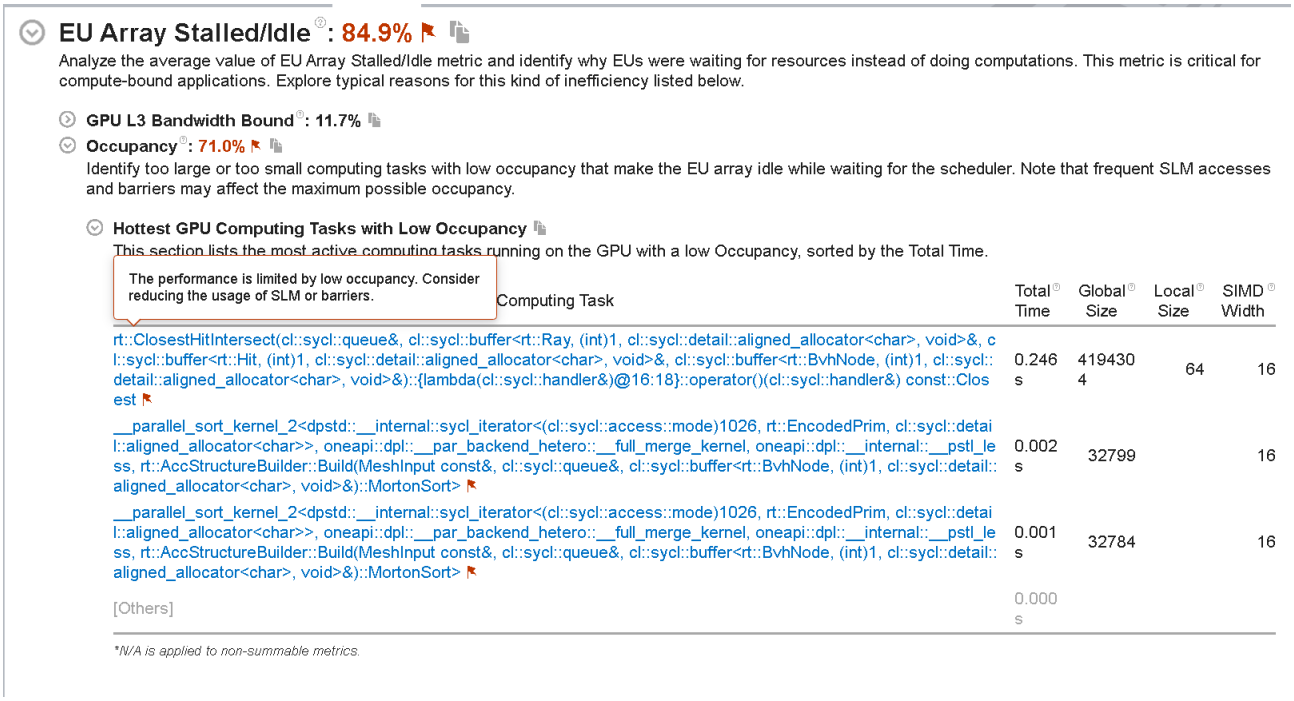
- 最も時間を消費する GPU カーネルの特徴付けを行います。命令タイプごとに動的命令数の内訳を取得するか、GPU ハードウェア･メトリックをベースに GPU 利用率の効率を確認します (**Characterization (特性化)** モード)。
- パフォーマンスが重要な基本ブロックのデータ、および GPU カーネルのメモリーアクセスで発生した問題に関するデータを特定します (**Source Analysis (ソース解析)** モード)。
- メモリー階層を調査して、GTI、キャッシュ、および GPU 間の重要なデータ･トラフィックを特定します。EU アレイのストールによりパフォーマンスの問題が生じることがあります。

関連資料

- コマンドラインから GPU 解析を設定
- 利用可能なインテル® プロセッサー･シリーズとプロセッサー･グラフィックス (英語) の関係を理解する

GPU 上で最適化

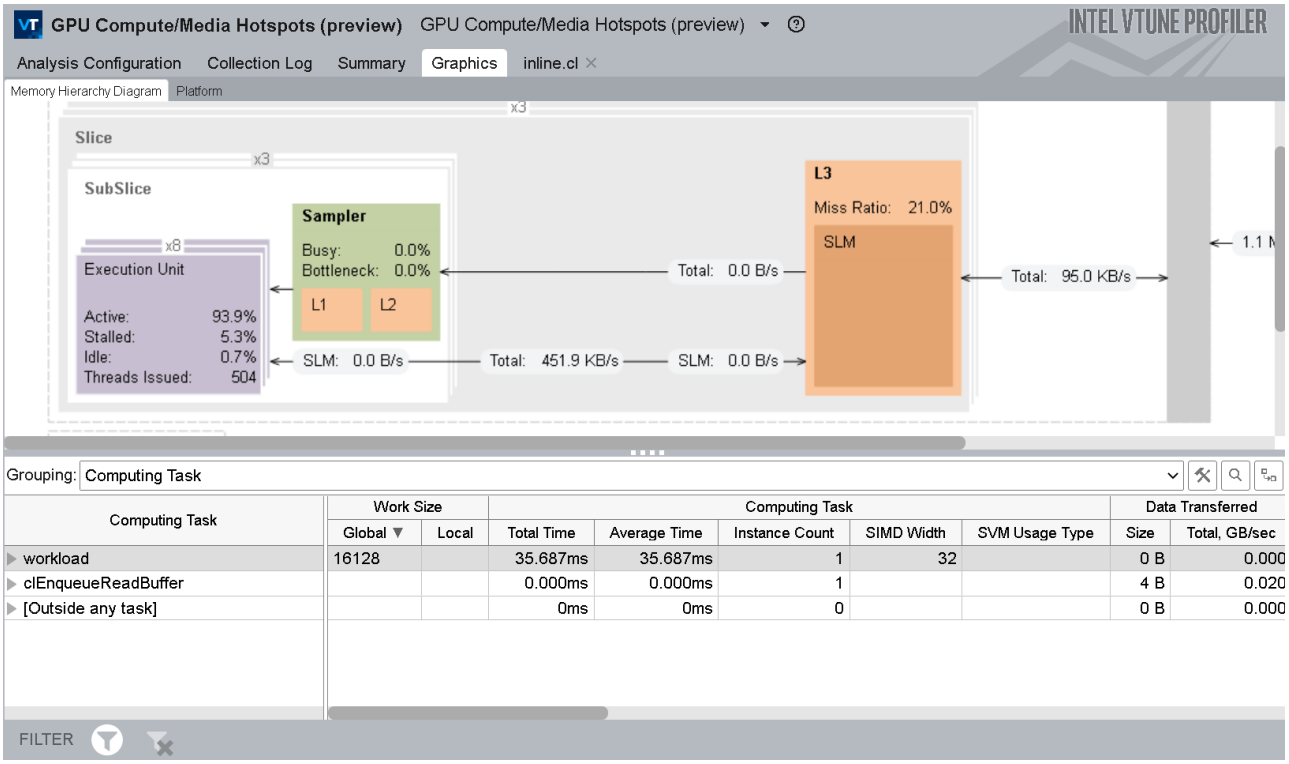
GPU にオフロードされたコード領域のパフォーマンスを最適化するには、いくつかの方法があります。最初に、GPU 計算/メディア･ホットスポット解析の **[Summary (サマリー)]** ウィンドウで、潜在的なボトルネックを理解してナビゲートします。ここでは、計算タスクに関連するメトリックも確認できます。



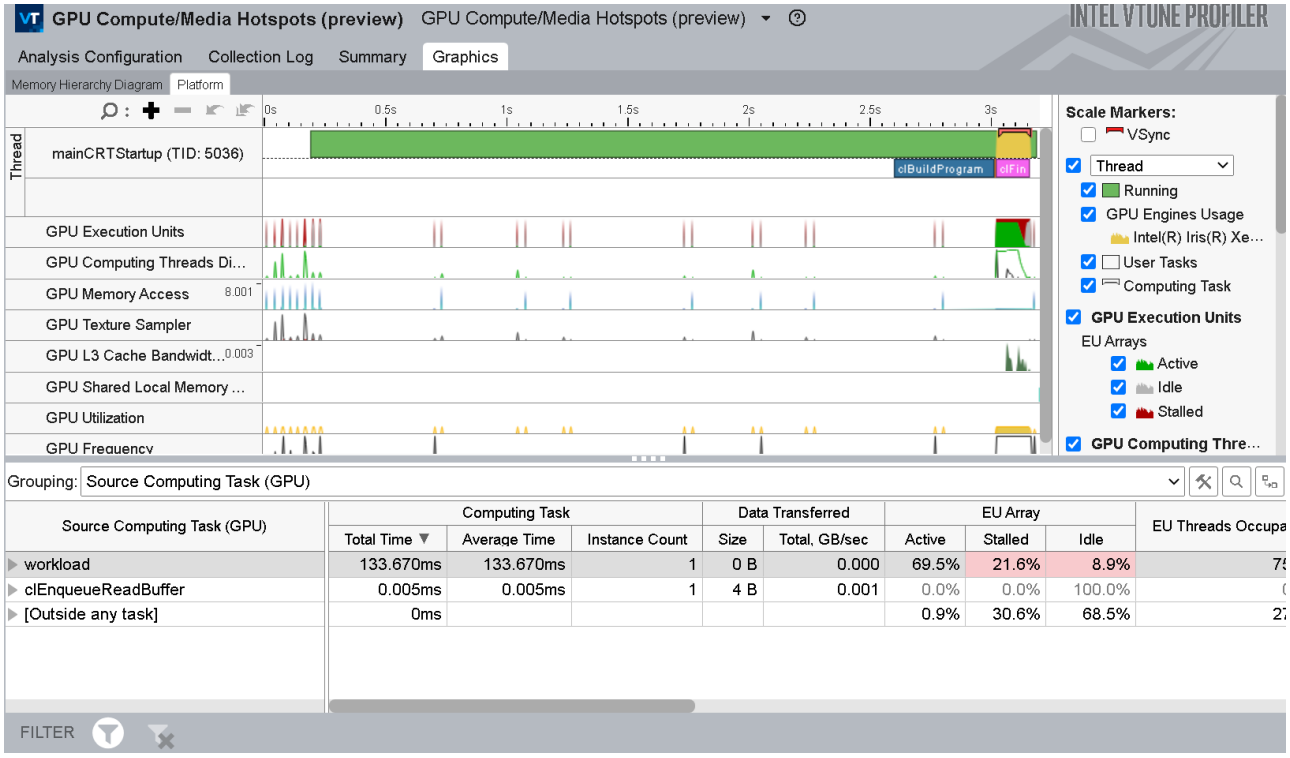
GPU 計算/メディア･ホットスポット解析の [Summary(サマリー)] タブ

次に **[Graphics (グラフィックス)]** タブに切り替えて、次のビューを確認します。

- **[Computing Task (計算タスク)]** グループ化レベルには、カーネルごとの計算タスクの合計処理時間の内訳と、データ転送サイズが示されます。
- **[Source Computing Task (ソース計算タスク)]** グループ化レベルには、プログラムが記述されたソースレベルのカーネルに関する同様の情報が表示されます。

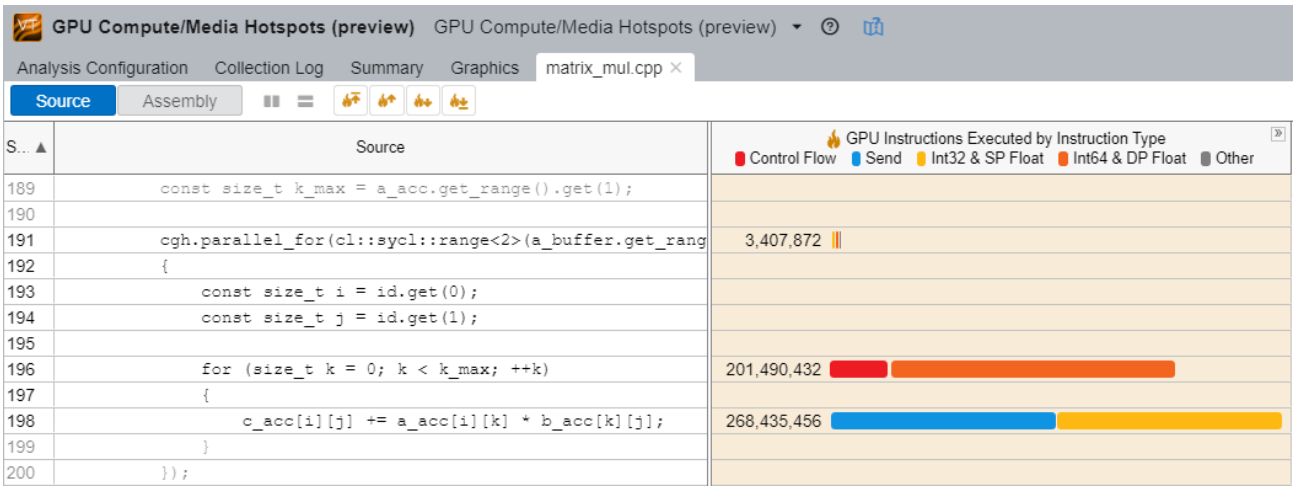


GPU ホットスポット解析の計算タスクのグループ化



GPU ホットスポット解析のソース計算タスクのグループ化

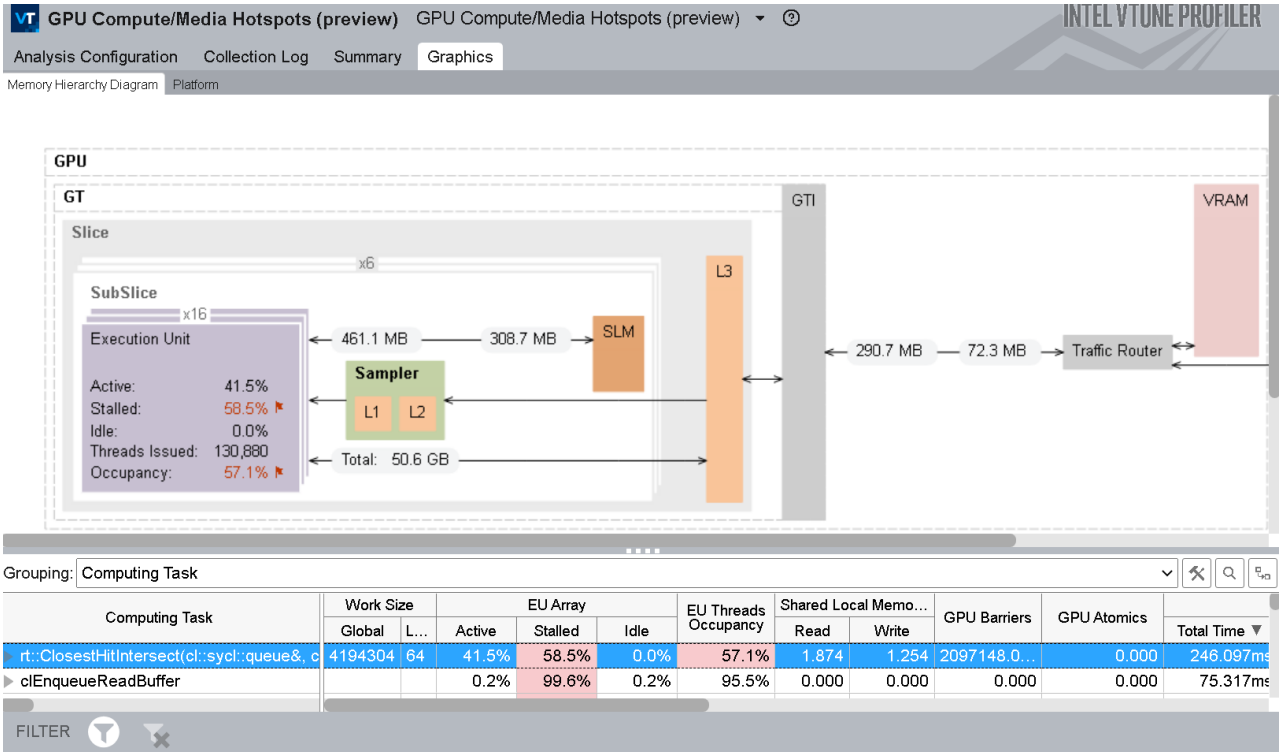
GPU 計算/メディア・ホットスポット解析では、**Dynamic Instruction Count (動的命令カウント)** 解析によってコードを調査し、各タイプの命令数を確認します。



動的命令カウント解析

この例では、int64 命令が int32 命令よりも 4 倍遅くなっています。調査中の場所で実際に int64 命令が必要であるか確認してください。コードを簡単に書き直すことで、実行される命令数を減らしてパフォーマンスを向上できる可能性があります。

この解析は、**Compute Basic (基本計算)** プリセットを使用して行うこともできます。このプリセットでは、**Memory Hierarchy diagram (メモリー階層図)** (前のセクションで紹介) に L3 トラフィックに関する追加情報を示します。



GPU ホットスポット解析のメモリー階層図

計算タスクを GPU アーキテクチャーに割り当てるため、グループ化されたメトリックとメモリー階層図に示される情報を調査します。

- リストでフラグが示される EU のストールと占有率に注目します。
- GPU バリアを多用している場所、または大量のローカルメモリー要求がある場所を確認します。
- 図から、データセットの初期サイズに対し GPU ユニット間で転送されるデータの比率が高いか確認します。
- 図の「Show Data As (データの表示形式)」機能を使用して、サイズ、帯域幅、GPU ユニット間の最大帯域幅のパーセンテージなどの情報を取得します。

関連資料

- GPU 計算/メディア・ホットスポット解析
- GPU メトリックのリファレンス

OpenMP* オフロードを使用する GPU 依存のアプリケーションをプロファイル

インテルの OpenMP* を使用してインテル® GPU にオフロードする場合、**HPC Characterization (HPC 特性)** 解析を実行します。ここでは、GPU 利用率のメトリックを解析して GPU 使用を調査し、ワークが GPU にどのようにオフロードされているか理解します。GPU がビジー状態である場合、さらに EU のアクティビティーを調査して、EU がストールまたはアイドル状態ではないか確認します。

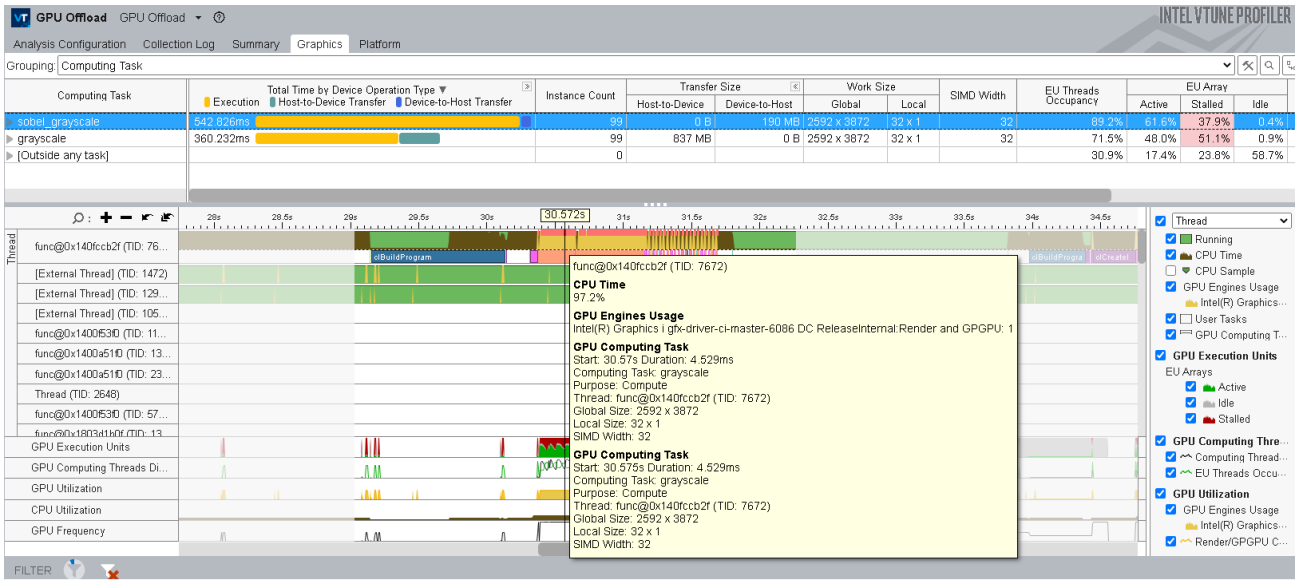
関連資料

- HPC 特性ビューの GPU 利用率メトリック
- HPC 特性解析

OpenCL* アプリケーションのプロファイル

GPU オフロード解析を使用して、OpenCL* アプリケーションのプロファイルを開始します。

- **Transfer (転送)** および **Work (ワーク)** のサイズを確認して、GPU にスケジュールされたカーネルの効率を調査します。
- タイムラインで OpenCL* 呼び出しに割り当てられた CPU と GPU の使用状況を監視し、対応するパフォーマンス・メトリックを確認します。



インテル® VTune™ プロファイラーで OpenCL* アプリケーションをプロファイル

OpenCL* アプリケーションのパフォーマンス解析には、GPU 計算/メディア・ホットスポット解析の一部の機能を利用することができます。この解析を **Characterization (特性化)** モードで実行し、**Full Compute (完全な計算)** プリセットオプションを選択します。これは、**Overview (概要)** および **Compute Basic (基本計算)** イベントグループを使用して、EU ストールの原因を理解するのに役立つデータを収集します。([Graphics (グラフィックス)] ウィンドウの [Timeline (タイムライン)] ペインにある) **GPU Computing Threads Dispatch (GPU 計算スレッドのディスパッチ)** メトリックが、OpenCL* アプリケーションが GPU で多くのワークを実行することを示す場合、**[Trace GPU Programming API (GPU プログラミング API のトレース)]** を有効にして解析を再実行します。

インテル® VTune™ プロファイラーは、プリセットされたしきい値でトリガーされるパフォーマンスの問題を含む、GPU で実行される OpenCL* カーネルをハイライト表示します。

Analysis Configuration | Collection Log | Summary | Graphics | Platform | Bottom-up

Grouping: Computing Task / Instance

Computing Task / Instance	Work Size		Computing Task					EU Array			L3 Shader Bandwidth,
	Global	Local	Total Time ▼	Average...	Instanc...	SIMD ...	SV...	Active	Stalled	Idle	
Accelerator_Intersect_R	1048576	64	53.398s	0.080s	669	16		85.6%	14.3%	0.1%	90.368
AdvancePaths_MK_DL_	1048576	64	10.280s	0.015s	669	16		69.4%	30.5%	0.1%	71.860
AdvancePaths_MK_SPL	1048576	64	10.243s	0.015s	669	16		30.1%	69.7%	0.2%	63.976
AdvancePaths_MK_GEN	1048576	64	9.236s	0.014s	669	16		35.2%	64.7%	0.1%	65.490
AdvancePaths_MK_GEN	1048576	64	8.226s	0.012s	669	16		24.8%	75.0%	0.2%	33.103
AdvancePaths_MK_HIT_	1048576	64	7.804s	0.012s	669	16		32.5%	67.3%	0.1%	47.085
AdvancePaths_MK_RT_I	1048576	64	7.160s	0.011s	669	16		43.8%	55.2%	1.0%	113.391
AdvancePaths_MK_DL_	1048576	64	5.164s	0.008s	669	16		39.3%	60.5%	0.2%	58.673
AdvancePaths_MK_NEX	1048576	64	4.812s	0.007s	669	32		17.2%	82.5%	0.3%	31.891
AdvancePaths_MK_HIT_	1048576	64	2.991s	0.004s	669	32		11.8%	87.9%	0.3%	28.215
AdvancePaths_MK_RT_I	1048576	64	2.614s	0.004s	669	32		16.9%	82.8%	0.3%	28.784

GPU 計算/メディア・ホットスポットを使用して OpenCL* カーネルを解析

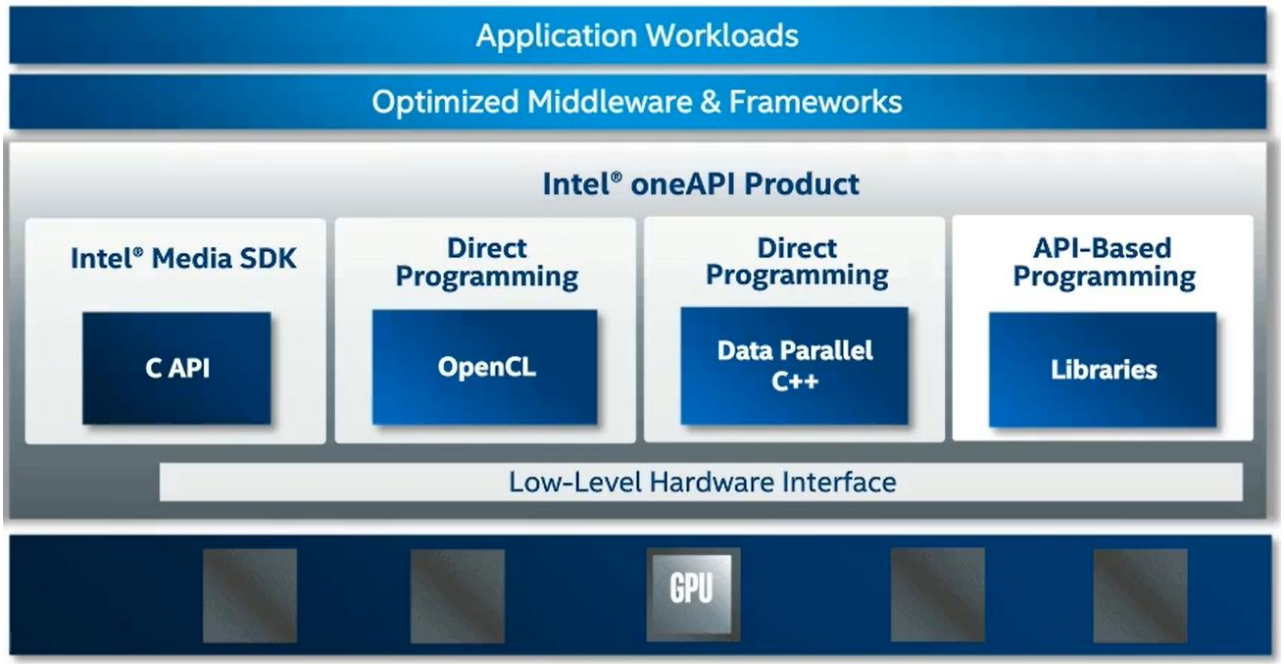
- 計算タスクの目的ごとに、**Compute (計算)** (カーネルの場合)、**Transfer (転送)** (ホストと GPU 間でデータを転送するルーチン)、**Synchronization (同期)** でデータを並べ替えることができます。
- 合計時間が最も長いカーネルに注目します。これには、平均時間値が長いカーネルと、時間値が短く頻繁に呼び出されるカーネルが含まれます。

関連資料

- OpenCL* コードを使用する GPU In-kernel プロファイルの概要 (英語)
- OpenCL* カーネル実行の調査 (英語)
- GPU OpenCL* アプリケーション解析 (英語)
- OpenCL* カーネル解析メトリックのリファレンス (英語)

DPC ++アプリケーションのプロファイル

データ並列 C++ (DPC++) は、インテル® oneAPI 仕様の中核を成します。DPC++ は、ISO C++ と Khronos Group の SYCL* 標準をコミュニティーの拡張機能と組み合わせて、インテル® CPU と GPU の並列プログラミングを独自に適用した堅牢な高水準言語です。CPU ホストと GPU 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 (USM) を実装しています (訳者注: SYCL* 2020 では USM は標準機能となりました)。



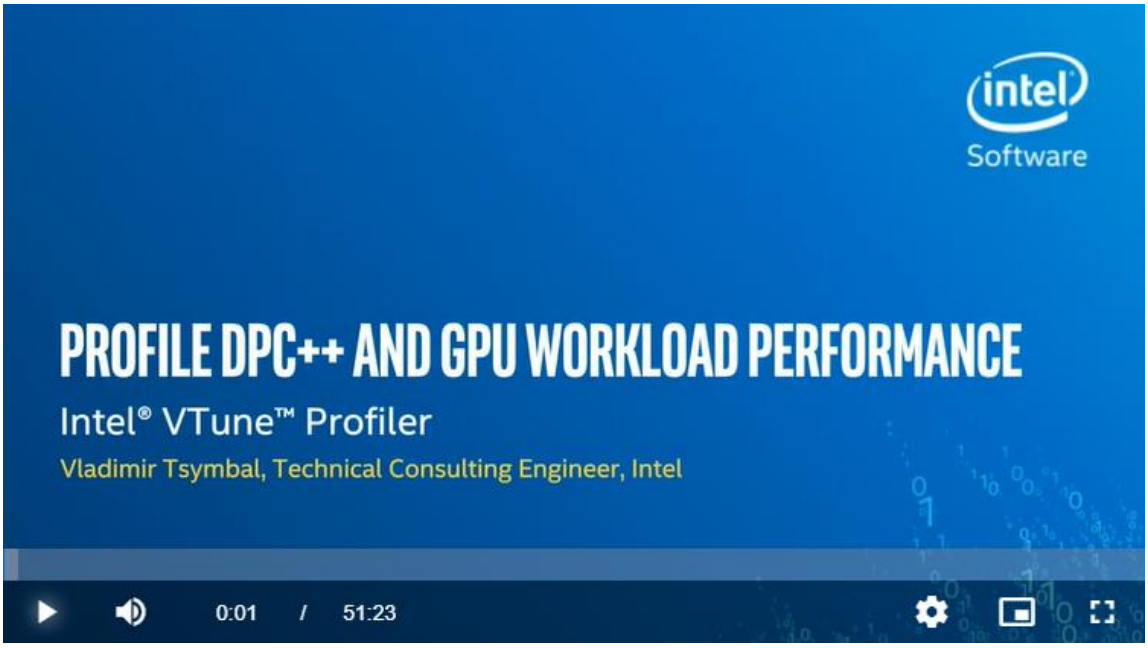
インテル® VTune™ プロファイラーを使用して DPC++ アプリケーションの GPU 解析を行うには、インテル® oneAPI ベース・ツールキットで提供される**インテル® C++ コンパイラー**が必要です。

CUDA* で記述された GPU アプリケーションは、インテル® DPC++ 互換性ツールを使用して DPC++ に移行できます。

注:
DPC++ アプリケーションでは、選択するバックエンド (レベルゼロ、OpenCL* など) によってプロファイルの方法が異なることがあります。プロファイル手法は、OpenCL* アプリケーションのプロファイルに似ています。

関連資料

- インテル® C++ コンパイラー (英語)
- インテル® oneAPI プログラミング・ガイド (英語)
- DPC++ 言語と API リファレンス (英語)
- CUDA コードを DPC++ へ移行するための『インテル® DPC++ 互換性ツール・ユーザーガイド』(英語)
- 既存の CUDA* コードを DPC++ へ移行するウェビナー (英語)
- 「GPU で実行する DPC++ アプリケーションのプロファイル」レシピ



DPC++ と GPU ワークロードのパフォーマンスをプロファイルするウェビナー (英語)

まとめ

計算集約型のアプリケーションをインテル® GPU にオフロードする場合、GPU 向けにこれらのアプリケーションを最適化して、デバイスの並列処理の能力を十分に活用することで最大の利点を得られます。

インテル® VTune™ プロファイラーやその他のインテル® ソフトウェア解析ツールには、この目的のために設計されたプロファイル機能が備わっているため、GPU の可能性を極限まで活用できます。

インテル® VTune™ プロファイラーのダウンロード

製品および性能に関する情報

[1] 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。