

インテル® VTune™ Amplifier XE 2017

General Exploration (一般解析) がどのように動作するかを理解する

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「[Understanding How General Exploration Works in Intel® VTune™ Amplifier XE](#)」の日本語参考訳です。

目次

一般解析のメカニズム.....	1
一般解析の結果を解釈する.....	3
イベント・ベース・サンプリングの複雑性を理解する.....	4
参考資料.....	6

インテル® VTune™ Amplifier XE の General Exploration (一般解析) タイプは、アプリケーションやシステムのマイクロアーキテクチャーに関連するハードウェアのボトルネックを検出する際に使用します。一般解析は、ハードウェア・イベント・カウンターを使用して問題の検出と場所の特定を行い、利用者が見やすく対処しやすい形式でデータを表示します。この記事では、この解析で使用されるメカニズムを説明し、結果の解釈方法、およびこの解析を行う際の複雑性と問題を示していきます。

一般解析のメカニズム

一般解析プロファイルで収集され表示されるデータのほとんどは、CPU 上のパフォーマンス・モニタリング・ユニット (PMU) によって収集されるハードウェア・イベントをベースにしています。これらの PMU カウンターは、キャッシュミスや分岐予測ミスなどの各種イベントをカウントするためプログラム可能なハードウェア・レジスターです。PMU とイベントに関する詳細は、[Software Developers Manual \(英語\)](#) をご覧ください。インテル® VTune™ Amplifier XE は、イベン

トベース・サンプリング (EBS) と呼ばれる方法でイベントを収集します。EBS では、各 PMU は特定のイベントを SAV (Sample After Value) 間隔でカウントするようにプログラムされます。イベントのカウントが開始されると、カウンタはインクリメントされ、SAV 値に到達すると割り込みが発生し、カウントをメモリーに記録して収集したカウンタをリセットします (命令ポインタ、コールスタック、PID などいくつかのデータは保持されます)。例えば、PMU が L2 キャッシュミス を SAV 2000 でカウントするようにプログラムされると、2000 番目の L2 キャッシュミスに到達すると割り込みが発生し、データが記録 (収集) され、インテル® VTune™ Amplifier XE は、2000 個のキャッシュミス を割り込みハンドラーで取得された単一の IP に対応付けます。その後、PMU はリセットされ再び 2000 個のカウントを開始します。

2000 個のすべてのキャッシュミスが、単一の IP に関連付けられるのは、イベントベース・サンプリングと呼ばれるテクニックを使用するためです。実際に 2000 個のキャッシュミスが単一の命令で生じる可能性はありませんが、この方法で十分な数のサンプルを収集できれば、結果は統計的に実際の動作に相当すると考えることができます。シンボル情報とソースコードが参照可能である場合、命令ポインタ (IP) は特定のモジュール、関数およびアプリケーション内の発生頻度の高い行へ関連付けることができます。

選択されたマイクロアーキテクチャー上で収集可能な数百のイベントがありますが、プロセッサの世代ごとにイベントは変更される可能性があります。一般解析は、アーキテクチャーの専門知識がなくても、収集するイベントの複雑な選択を抽象化して、プラットフォーム間で一貫性のある分かりやすいメトリックをユーザーに提供します。これは、次のセクションで触れるトップダウン法 [1] を基に行われます。一般解析は、最近のアーキテクチャー上ではおよそ 60 個のイベントを収集できます。PMU カウンタの数は限られているため (通常論理コアごとに 4 個)、単一プロファイルですべてを収集するには多重化しなければいけません。これは、特定の時間 4 つのイベントを収集したら、次の時間軸では別の 4 つのイベントに入れ替えて収集を行い、プロファイル全体を収集するまでこの手順を繰り返します。イベントが収集されていない期間、特定のイベントの発生を予測する必要があります。この多重化の問題は複雑であり、記事の後半の複雑性のセクションで取り上げます。

一般解析が選択されると、インテル® VTune™ Amplifier XE はこれらすべてを自動的に処理します。

一般解析の結果を解釈する

一般解析の収集が完了すると、収集されたすべてのデータのサマリーが表示されます。また、経過時間、次のハードウェア・イベントをベースとする一般解析の主要メトリックなど共通のパフォーマンス・データも表示されます。メトリックは、いくつかのイベントを組み合わせた基準値です。例えば、クロックティック (CPU_CLK_UNHALTED.THREAD イベント) を完了した命令数 (INST_RETIRED.ANY イベント) で割ると、一般的に使用されるメトリックである命令ごとのサイクル数 (CPI) を取得できます。

インテル® VTune™ Amplifier XE のメトリックは、マイクロアーキテクチャーのボトルネックを特定するため階層的に構成されています。この階層構造と方法論は、トップダウン方式として知られています。これは、アーキテクチャーとマイクロアーキテクチャー・レベルの重要なボトルネックを素早く特定する簡単で正確な方法です。頻度の高いパフォーマンス上のボトルネックは階層構造で組織化され、そしてそれらのコストはマイクロアーキテクチャーに依存しないメトリックを使用して重み付けされます。その結果、階層構造はプロセッサの世代間で一貫性と前方互換性を持ち、新しいマイクロアーキテクチャーとモデル固有のイベントを理解するため伝統的に要求されていた習熟度を軽減します。この方法論の詳しい説明は、IEEE ドキュメント^[1]でご覧いただけます。また、「[マイクロアーキテクチャーのトップダウン解析法を使用してアプリケーションをチューニングする方法](#)」の記事も役に立ちます。また、[チューニング・ガイド](#)から、一般解析をベースにしたそれぞれのハードウェア・プラットフォーム向けのチューニング・ガイドを入手できます。解析を実行した後、アプリケーションのホットスポットに注目することは重要です。最も値の大きなカウンターは、CPU_CLK_UNHALTED.THREAD イベントです。次に、GUI に表示される階層構造の上位レベルでハイライトされているメトリックに注目します。ハイライト表示されたメトリックの値は、事前定義されたしきい値の範囲外にあり、パフォーマンス上悪影響があることを示しています。ハイライトされたメトリックを展開すると、上位に関連するサブメトリックが表示されます。再度ハイライトされたメトリックを探してパフォーマンスへの影響を特定し、ハイライトされたメトリックを展開表示します。ハイライトされたメトリックの階層構造の最下位のレベルに注目するところから開始します。各種メトリックの意味とパフォーマンスへの影響を理解するには、メトリックにマウスカーソルを移動するとドキュメントがポップアップ表示されます。チューニング・ガイドには、ハイライトされたメトリックに関連するパフォーマンス向上のための多くの推奨事項が記載されています。最下位のレベルにハイライト表示されるメトリックがない場合、ハイライトされる上位レベルに注目し、メトリックで示されるパフォーマンスの問題の原因を特定するようにします。コードを変更した後、一般解析を再度実行してパフォーマンスへの影響と、最適化されたコードでメトリックがどのように変化したかを確認します。インテル® VTune™ Amplifier

XE の GUI では、結果を並べて比較することができます。このパフォーマンス・チューニングの手順は、目標に達するまで、または開発者が時間を費やせる限り繰り返し継続されます。一般に、最初の問題を解決するとパフォーマンスに最も影響する改善が確認できますが、その後の変更では次第に収益逓減は減少していきます。

イベント・ベース・サンプリングの複雑性を理解する

標準ユーザーモードのパフォーマンス解析では、それほど一般的ではない EBS に関連するいくつかの複雑性があります。これらの問題がインテル® VTune™ Amplifier XE で解析した結果にどのように影響しているか理解することは重要です。

複雑性 1: インテル® ハイパースレッディング・テクノロジー

複雑性を招く原因の一つにインテル® ハイパースレッディング・テクノロジー (同時マルチスレッド化「SMT」の実装と呼ばれることもある) があります。SMT が有効化されたシステムでは、各物理コアは 2 つの論理プロセッサとして見える追加のハードウェアを持ちますが、ハードウェア・リソースのほとんどは共有されています。この場合の PMU とイベントの設計が複雑になることは容易に想像できます。SMT が有効化されていると、イベントがどちらの論理コアで発生したか区別するのが困難なことがあるため、いくつかのイベントは正確性を欠きます。

さらに、これらのイベントを使用するメトリックの計算では、ハードウェアに関していくつかの仮定を行うため複雑になります。例えば、現代のインテル CPU は、サイクルごとに 4 つの μ ops (マイクロオペレーション) をアロケートおよびリタイアできます。そのため、理想的な CPI は 0.25 となります。しかし、SMT が有効化されている場合、物理コアではサイクルごとに 4 つの μ ops の制限が適用されますが、各 SMT スレッドはアロケートとリタイアのリソースを共有します。つまり、SMT が有効化されたシステムでアプリケーションと複数のソフトウェア・スレッドが同時に実行されていると、メモリアクセスやポートの競合でわずかなストールが発生し、論理コアごと (SMT コア) の CPI は最良でも 0.5 となります。そして、ほとんどのアプリケーションでは、並列実行とシリアル実行が混在しています。これにより、SMT が有効化されているといくつかのメトリックにばらつきが生じ、この複雑性を考慮して解釈する必要があります。SMT の影響を受ける注意すべき主なメトリックは、リタイア、投機の問題、バックエンド依存およびフロントエンド依存のトップレベルのメトリック (カテゴリー) です。

SMT の影響は、1 つのコア上で両方の SMT スレッドが同時に実行される頻度と同じリソース (アロケーション・スロットなど) の競合の頻度に比例します。同時実行が多い場合、リタイ

ア、投機の問題、およびフロントエンド依存のメトリックが低く、そしてバックエンド依存のメトリックが高くなります。最悪のシナリオでは、これらのメトリックは ~1X となりますが、一般に変化はそれ以下となります。また、インテル® VTune™ Amplifier XE は、メトリックがパフォーマンスの問題を示すことを強調するため、メトリックのそれぞれに内部的なしきい値を持っており、多くの場合この分散はメトリックがハイライトされるかどうかに影響されません。SMT がパフォーマンスの結果に影響しているか意識することは大切であり、もし影響の可能性があれば、BIOS で SMT を無効にして解析を再度行ってください。

複雑性 2: イベントスキッド

EBS によるその他の複雑性は、PMU 割り込みハンドラーがイベントを発生させた命令ポイント (IP) を収集する際に生じるイベントスキッドです。ハードウェアの複雑性と割り込み処理における遅延により、指定されたイベントに割り当てられた割り込みハンドラーで IP を収集すると、実際にイベントが発生した位置から数命令後にスキッド (ずれる) します。メトリックをモジュールや関数レベルで評価する際には、多くの場合 IP はそのモジュールや関数内にあるため問題となりません。しかし、ソース行やアセンブリ命令レベルで参照すると、イベントカウントとメトリック値は、バイナリーでは 1 行あるいは 1 命令後方にずれて表示される可能性があります。行や命令レベルでメトリック値を見た場合におかしいと感じたら、数行前を見てスキッドしたメトリックに関連する実際の行や命令を見つける必要があります。イベントスキッドの特殊なケースは、イベントが分岐命令で生じた場合、イベントに関連する分岐先は離れた位置にあることです。メトリックが分岐先に関連付けられ、その原因が分岐元にある場合、この特殊ケースの影響に注意してください。いくつかのイベントは、“precise” モードで収集できます。これらのイベントは、通常イベント名の最後に _PS が追加されて示されます。

複雑性 3: イベントの多重化

一般解析タイプは、最近のプロセッサ上では 60 個を超えるイベントを収集します。そのため、前述したようにイベントの収集を多重化する必要があります。実行時間が短い、多重化されたすべてのイベントグループの多重化時間を通してサイクルを取得する際に一定の状態のタイプが繰り返されない場合、多重化によって収集されるデータに変化が生じる可能性があります。有効なデータを取得するため特定のアプリケーションを実行する時間には、多くのバリエーションがあります。しかし、一般解析では最大 15 グループが 10ms 間隔で、各グループ交互に取得されることを忘れないでください。さらに、インテル® VTune™ Amplifier XE には “**MUX Reliability (MUX 信頼性)**” メトリックがあります。これは、グリッドの列に与えられたメトリックのセットの多重

化の誤りに関する正確性を決定します。この値の範囲は 0 から 1 であり、一般的に値が 0.9 を超えた場合、その列のメトリックの信頼性は高いと言えます。データが、前述した兆候 (低い MUX 信頼性、短い実行期間) により多重化の影響を受けていると想定される場合、長時間実行するか GUI で multiple run (複数実行) を有効にして再度解析を行ってください。

複雑性 4: 割り込みハンドラーとマスク

インテル® VTune™ Amplifier XE で EBS を行う際にもう 1 つ注意すべき問題は、割り込みハンドラーと割り込みマスクがサンプリングに与える影響です。前述のように、サンプリングは割り込みと割り込みハンドラーを使用してカウントされます。解析されるプラットフォームやアプリケーションで、(他の割り込みハンドラーなどで) 長い時間割り込みがマスクされると、その間サンプルは収集されません。これはまた、割り込みハンドラー自身も EBS で完全にプロファイルできないことを意味します。割り込みがマスクされている間に複数の PMU 割り込みが発生すると、割り込みがアンマスクされたときに最後の割り込みのみがハンドルされます。この問題は、予測よりも多くのサンプルがカーネルのアイドルループ (割り込みは通常マスクおよびアンマスクされる) に起因する場合に、しばしば見受けられます。

これらの問題を意識して、これらの複雑性に影響される解析を認識できるようにします。

参考資料

[1] A. Yasin, "A Top-Down Method for Performance Analysis and Counters Architecture", Performance Analysis of Systems and Software (ISPASS), IEEE International Symposium on, 2014

コンパイラーの最適化に関する詳細は、[最適化に関する注意事項](#)を参照してください。

© 2017 Intel Corporation. 無断での引用、転載を禁じます。

Intel、インテル、Intel ロゴ、VTune は、アメリカ合衆国および / またはその他の国における Intel Corporation の商標です。

* その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。