この資料の対象：ソフトウェア開発者

アプリケーションのパフォーマンス最適化に注目する開発者

- パフォーマンスのエキスパートである必要はなりません
- しかし、アプリケーション自体のことは熟知していることが前提です

インテル® Xeon® E5 v4 ファミリーのプロセッサーを対象とする開発者

インテル® VTune™ Amplifier XE パフォーマンス・アナライザーを使用する開発者

- ここで紹介するパフォーマンスに関する情報は、他のツール（PTU など）にも適用できますが、ここではインテル® VTune™ Amplifier XE に注目します

© 2017, Intel Corporation. 転載を禁じます。* 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。

このスライドの使い方

一度スライドを読み通して、データ収集時に再度見てください

パフォーマンス解析は、何度かの繰り返しで達成されることを忘れないでください

ソフトウェアの最適化は、以下を行ってから始めてください：

- 任意のコンパイラー最適化オプションを適用（/O2、/QxAVX2 など）
- 適切なワークロードを選択
- 基準となるパフォーマンスの測定

© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® Xeon® プロセッサー E5 v4 ファミリー向けのソフトウェア最適化にインテル® VTune™ Amplifier XE を使用する

ソフトウェア & サービスグループ

Ver.1.0

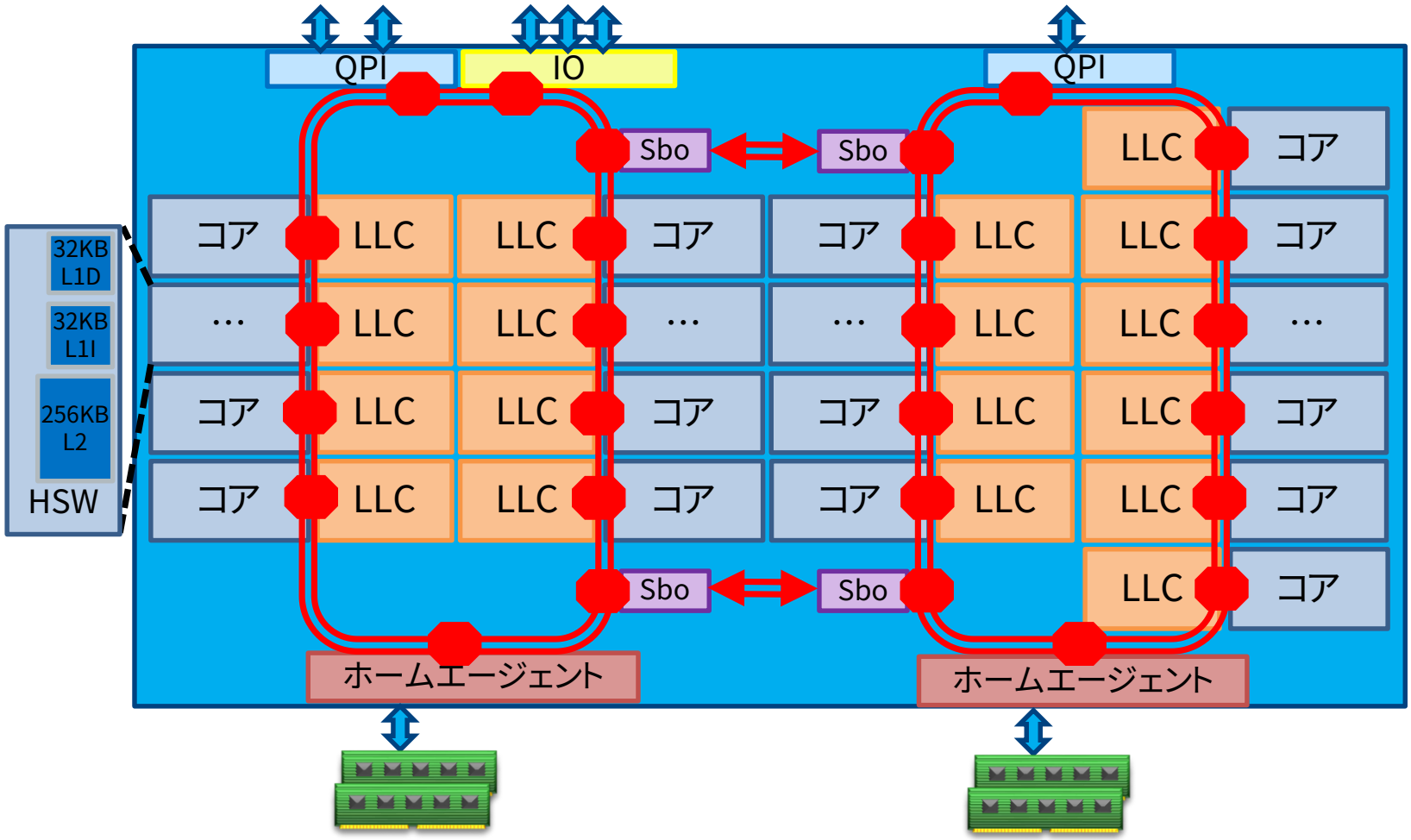
© 2017, Intel Corporation. 無断での引用、転載を禁じます。* その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

内容

- インテル® Xeon® プロセッサー E5 v4 ファミリー
- インテル® VTune™ Amplifier XE
- ソフトウェア最適化サイクル
  - ホットスポット
  - 効率を判定する方法
  - ボトルネックを特定
  - 一般的な効率を損ねる原因:
- その他のチューニングの推奨事項

© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® Xeon® プロセッサー E5 v4 ファミリー



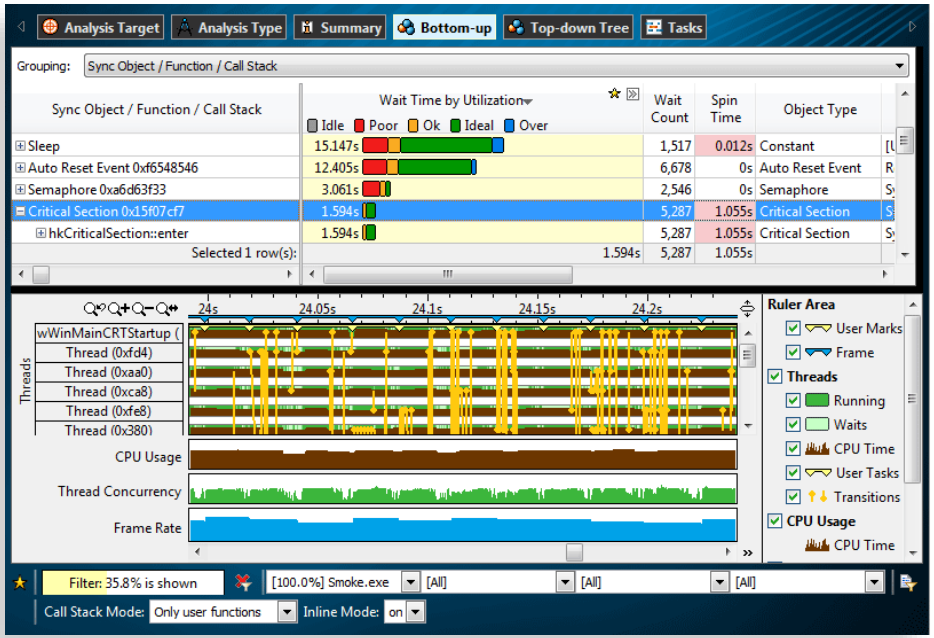
構成			
コア周波数	～ 3.7GHz	QPI スピード	～ 9.6GT/秒
コア数	～ 22 個	DDR4 周波数	～ 2133 MHz
L3 キャッシュサイズ	～ 55MB	ソケット数	～ 2 個

© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® VTune™ Amplifier XE

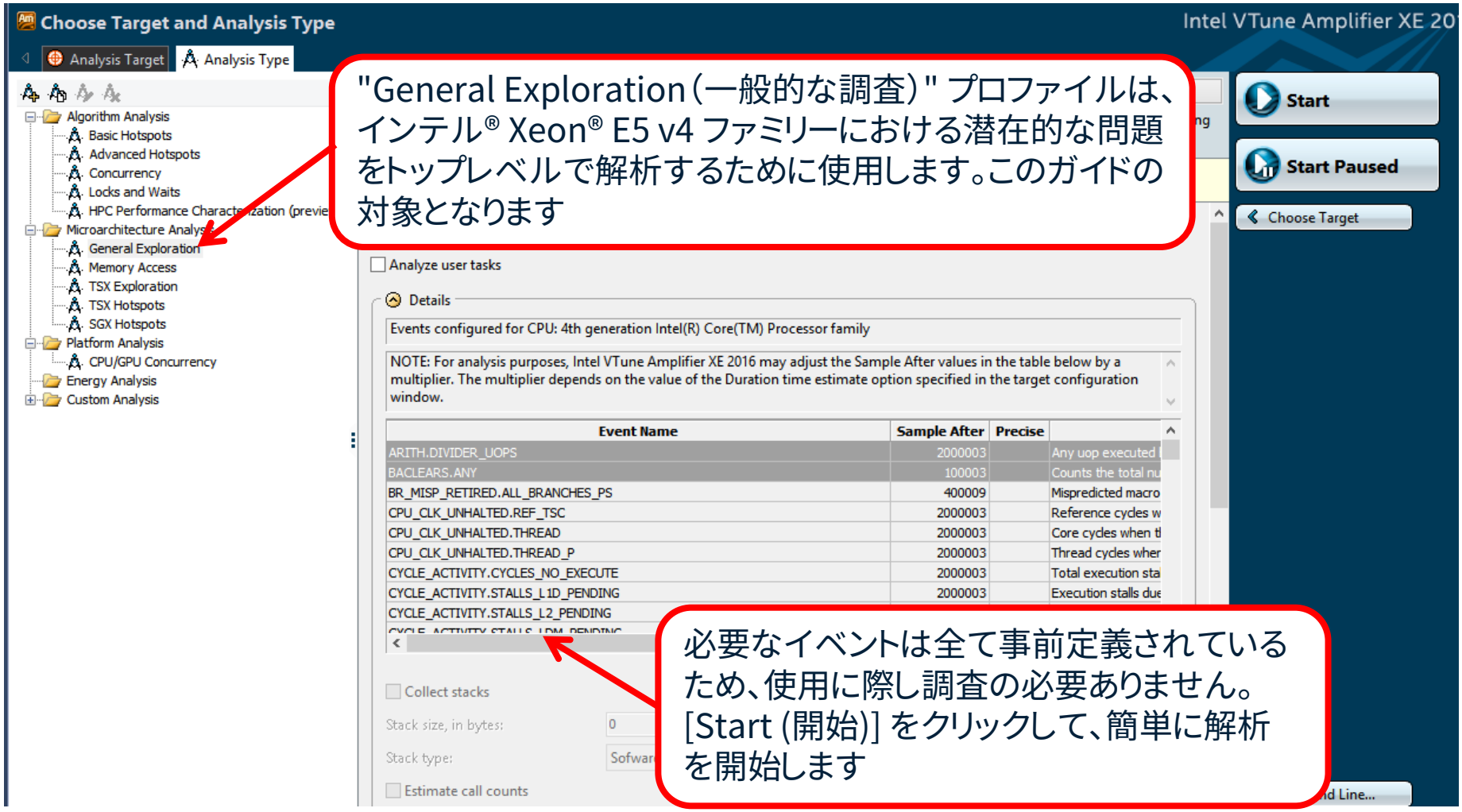
インテル® VTune™ Amplifier XE の機能：

- 複数の収集タイプ
  - ホットスポット（統計呼び出しツリー）
  - スレッドの並行性
  - ロックと待機の分析
  - イベント・ベース・サンプリング
- 全ての解析タイプを
  タイムライン表示へ統合
- ソースとアセンブリー表示
- C/C++、Fortran、Java、アセンブリー、.NET との互換性
- Visual Studio* への統合、コマンドラインでの利用、もしくは Windows® と Linux* 向けスタンドアロン・インターフェイスを利用



© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® VTune™Amplifier XE の事前定義プロファイル



© 2017, Intel Corporation. 転載を禁じます。* 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。

拡張 General Exploration 表示

General Exploration 表示が選択され（デフォルト）、General Exploration プロファイルを実行すると、拡張表示が有効になります

収集されたすべてのデータは、計算済みの測定基準によって階層形式（次のスライドを参照）で表示されます

General Exploration | General Exploration Viewpoint (change) | 2016

Analysis Target | Analysis Type | Collection Log | Summary | Bottom-up | Event Count

Grouping: Function / Call Stack

Function / Call Stack	Clockticks	Instructions Retired	CPI Rate	Front-End Bound	Bad Speculation	Back-End Bound	Retiring	Module	Function (Full)	Source File	Start Address
grid_intersect	14,948,022,422	12,672,019,008	1.180	7.0%	3.9%	74.8%	14.3%	3_tachyon_omp.exe	grid_intersect	grid.cpp	0x40de70
sphere_intersect	9,618,014,427	9,026,013,539	1.066	5.7%	4.5%	73.1%	16.6%	3_tachyon_omp.exe	sphere_intersect	sphere.cpp	0x40a9e0
grid_bounds_intersect	1,064,001,596	690,001,035	1.542	8.0%	4.5%	75.0%	12.5%	3_tachyon_omp.exe	grid_bounds_intersect	grid.cpp	0x40eeb0
func@0x1002df7f	602,000,903	420,000,630	1.433	22.1%	36.3%	0.0%	48.9%	libiomp5md.dll	func@0x1002df7f		0x1002df7f
pos2grid	246,000,369	160,000,240	1.538	7.7%	0.0%	84.6%	23.2%	3_tachyon_omp.exe	pos2grid	grid.cpp	0x40f140
shader	212,000,318	190,000,285	1.116	4.5%	0.0%	91.0%	4.5%	3_tachyon_omp.exe	shader(struct ray *)	shade.cpp	0x408de0
tri_intersect	198,000,297	192,000,288	1.031	4.8%	4.8%	76.0%	14.4%	3_tachyon_omp.exe	tri_intersect	triangle.cpp	0x40ad40
__kmp_x86_pause	184,000,276	82,000,123	2.244	0.0%	0.0%	17.4%	100.0%	libiomp5md.dll	__kmp_x86_pause		0x100925a0
Raypnt	172,000,258	256,000,384	0.672	16.6%	16.6%	44.8%	22.1%	3_tachyon_omp.exe	Raypnt(struct ray *,d...	vector.cpp	0x408570
Selected 1 row(s):	14,948,022,422	12,672,019,008	1.180	7.0%	3.9%	74.8%	14.3%				

Thread: OMP Master Thread #... / OMP Worker Thread ... / OMP Worker Thread ... / OMP Worker Thread ... / Thread (TID: 7860) / Thread (TID: 10804) / Thread (TID: 11216) / Thread (TID: 11200) / Hardware Events

Ruler Area: Region Insta... / OpenMP Ba... / Thread / Running / Hardware Event ... / INST_RETIR / Hardware Events / Hardware Event Count / INST_RETIR

FILTER 100.0% | Process: Any Process | Thread: Any Thread | Module: Any Module | User functions + 1 | Show inline functions | Functions only

© 2017, Intel Corporation. 転載を禁じます。* 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。

拡張 General Exploration 表示

General Exploration　General Exploration viewpoint (change)

Analysis Target | Analysis Type | Collection Log | Summary | Bo

Grouping: Function / Call Stack

Function / Call Stack	Clockticks	Instructions Retired	CPI Rate	Front-End Bound	Bad Speculation	Back-End Bound	Retiring
grid_intersect	14,948,022,422	12,672,019,008	1.180	7.0%	3.9%	74.8%	4.3%
sphere_intersect	9,618,014,427	9,026,013,539	1.066	5.7%	4.5%	73.1%	6.6%
grid_bounds_intersect	1,064,001,596	690,001,035	1.542	8.0%	4.5%	75.0%	12.5%
func@0x1002df7f	602,000,903	420,000,630	1.433	22.1%	[illegible]	[illegible]	48.9%
pos2grid	246,000,369	160,000,240	1.538	7.7%	[illegible]	[illegible]	23.2%
shader	212,000,318	190,000,285	1.116	4.5%	[illegible]	[illegible]	4.5%
tri_intersect	198,000,297	192,000,288	1.031	4.8%	[illegible]	[illegible]	14.4%
__kmp_x86_pause	184,000,276	82,000,123	2.244	0.0%	[illegible]	[illegible]	100.0%
Raypnt	172,000,258	256,000,384	0.672	16.6%	[illegible]	[illegible]	22.1%
[illegible]	168,000,252	64,000,096	2.625	28.3%	[illegible]	[illegible]	56.5%
Selected 1 row(s):	14,948,022,422	12,672,019,008	1.180	7.0%	4.5%	4.8%	14.3%

Bad Speculation	
Branch Mispredict	Machine Clears
3.9%	0.0%
4.5%	0.0%
4.5%	0.0%
0.0%	36.3%
0.0%	0.0%
0.0%	0.0%
0.0%	4.8%
0.0%	0.0%
0.0%	16.6%
0.0%	5.7%
3.9%	0.0%

階層データの表示は、各コアのパイプラインで実行スロットがどのように利用されているか示します

パイプラインの利用率のカテゴリーに関連する内訳を見るためカラムを拡張します：リタイアメント、投機の問題、バックエンドもしくはフロントエンドのパイプライン・スロット

© 2017, Intel Corporation. 転載を禁じます。* 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。

拡張 General Exploration 表示

General Exploration General Exploration viewpoint (change)

Analysis Target | Analysis Type | Collection Log | Summary | Bottom-up | Event Count | Platform

Grouping: Function / Call Stack

Function / Call Stack	Clockticks	Instructions Retired	CPI Rate	Front-End Bound	Bad Speculation	Back-End Bound	Retiring
grid_intersect	14,948,022,422	12,672,019,008	1.180	7.0%	3.9%	74.8%	14.3%
sphere_intersect	9,618,014,427	9,026,013,539	1.066	5.7%	4.5%	73.1%	16.6%
grid_bounds_intersect	1,064,001,596	690,001,035	1.542	8.0%	4.5%	75.0%	12.5%
func@0x1002df7f	602,000,903	420,000,630	1.433	22.1%	36.3%	0.0%	48.9%
pos2grid	246,000,369	160,000,240	1.538	7.7%	0.0%	84.6%	23.2%
shader	212,000,318	190,000,285	1.116	4.5%	0.0%	91.0%	4.5%
tri_intersect	198,000,297	192,000,288	1.031	4.8%	4.8%	76.0%	14.4%
__kmp_x86_pause	184,000,276	82,000,123	2.244	0			
Ra			.672	16			
			.180	7			

パイプライン利用率の各カテゴリーに表示される事前計算された測定基準は、利用者の解析時間を軽減します

ホットスポットがピンク色のセルで強調されている場合、その値は VTune Amplifier XE が持つしきい値を超えており、調査の必要があることを意味します

© 2017, Intel Corporation. 転載を禁じます。* 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。

パフォーマンス計測の複雑性

第 5 世代インテル® Core™ プロセッサー・ファミリーとインテル® Xeon® プロセッサー E5 v4 ファミリーには、パフォーマンス計測に大きな影響を与える 2 つの機能があります：

- インテル® ハイパースレッディング・テクノロジー
- インテル® ターボ・ブースト・テクノロジー 2.0

これらの機能が有効にされている場合、パフォーマンス・データを計測して解釈するのが難しくなります

- 大部分のイベントはスレッドごとにカウントされますが、いくつかのイベントはコアごとにカウントされます
  - 特定のイベントについては、インテル® VTune™ Amplifier XE のヘルプをご覧ください

一部の専門家は、これらの機能を無効にしてパフォーマンスを解析し、最適化が終了した後に再度有効にする手法を取ります

© 2017, Intel Corporation. 転載を禁じます。* 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。

"ハードウェア上のソフトウェア" のチューニング手順

各ホットスポットで

- 効率を判断
  - もし十分でなければ：
    - 主なボトルネックを特定
    - 非効率であるアーキテクチャー上の理由を見つけます
    - 問題を最適化します

繰り返します!

© 2017, Intel Corporation. 転載を禁じます。* 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。

"ハードウェア上のソフトウェア" のチューニング手順

各ホットスポットで

- 効率を判断
  - もし十分でなければ:
    - 主なボトルネックを特定
    - 非効率であるアーキテクチャー上の理由を見つけます
    - 問題を最適化します

繰り返します!

© 2017, Intel Corporation. 転載を禁じます。* 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。

ホットスポットの特定

ホットスポットとは：
アプリケーションが多くの時間を費やしているコード中の場所

なぜ重要なのか：
そこに最適化の労力を費やすべきです！

- アプリケーションの実行時間の 2% にしか満たない関数を改善する必要があるか？

どのように検出するか：
インテル® VTune™ Amplifier XE の、"Basic Hotspots" か "Advanced Hotspots" 解析タイプを利用

- 通常ホットスポットは、CPU_UNHALTED.THREAD イベント（別名 "clockticks"）で定義されます

© 2017, Intel Corporation. 転載を禁じます。* 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。

"ハードウェア上のソフトウェア" のチューニング手順

各ホットスポットで

- 効率を判断
  - もし十分でなければ：
    - 主なボトルネックを特定
    - 非効率であるアーキテクチャー上の理由を見つけます
    - 問題を最適化します

繰り返します!

© 2017, Intel Corporation. 転載を禁じます。* 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。

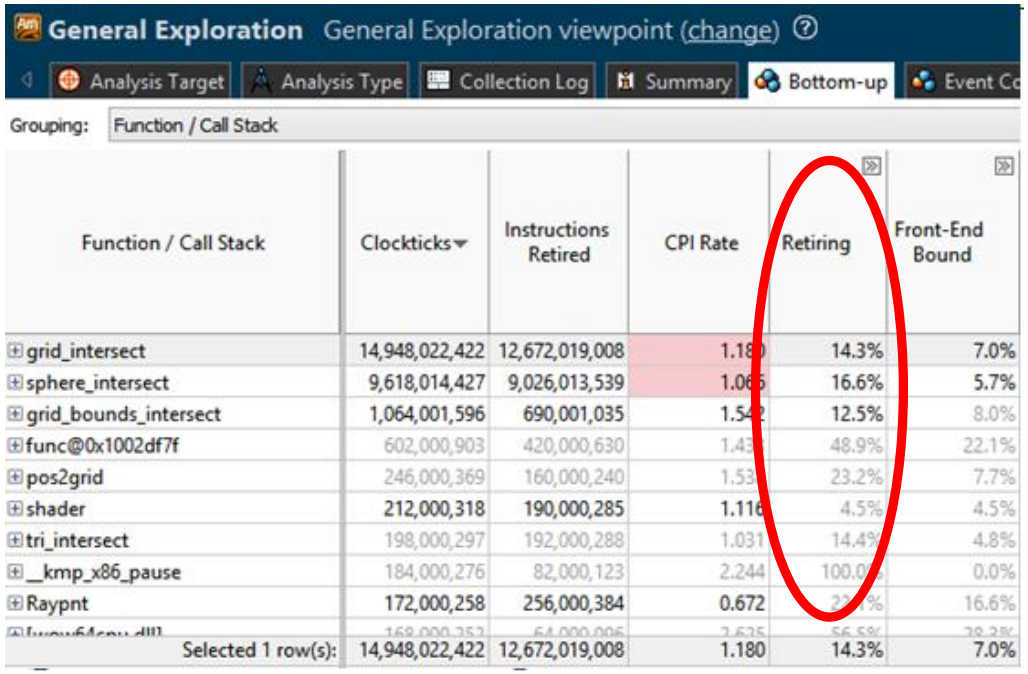
効率法 1: リタイアしたパイプライン・スロットの比率 (%)

なぜ重要なのか：アプリケーションがプロセッサーを使用する効率を理解するのに役立ちます

どのように検出するか：”General Exploration” プロファイルの測定基準：*Retiring* (リタイアした比率) *に注目*

何を行うか：

- 特定のホットスポットの場合：
- 75% 以上のパイプライン・スロットがリタイアした場合 (.75 以上)、3 番目の効率法に進み、ベクトル化によってパフォーマンスを向上できるか確認するには、コードスタディー 1 を参照



Function / Call Stack	Clockticks	Instructions Retired	CPI Rate	Retiring	Front-End Bound
grid_intersect	14,948,022,422	12,672,019,008	1.180	14.3%	7.0%
sphere_intersect	9,618,014,427	9,026,013,539	1.065	16.6%	5.7%
grid_bounds_intersect	1,064,001,596	690,001,035	1.542	12.5%	8.0%
func@0x1002df7f	602,000,903	420,000,630	1.43	48.9%	22.1%
pos2grid	246,000,369	160,000,240	1.53	23.2%	7.7%
shader	212,000,318	190,000,285	1.116	4.5%	4.5%
tri_intersect	198,000,297	192,000,288	1.031	14.4%	4.8%
__kmp_x86_pause	184,000,276	82,000,123	2.244	100.0%	0.0%
Raypnt	172,000,258	256,000,384	0.672	2[illegible]%	16.6%
[illegible]	[illegible]	[illegible]	2.625	56.5%	28.2%
Selected 1 row(s):	14,948,022,422	12,672,019,008	1.180	14.3%	7.0%

- それ以外は、次のスライドに進みます

© 2017, Intel Corporation. 転載を禁じます。* 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。

効率法 1: リタイアしたパイプライン・スロットの比率(%)

何を行うか:パイプライン・スロットのリタイア比率が 75% 以下のホットスポットは、効率を判断する際にアプリケーションのタイプを考慮してください。ホットスポットが期待される範囲以下である場合、それは非効率であると言えます

	各カテゴリーのパイプライン・スロットの期待される範囲(うまく最適化されたホットスポット向け):		
カテゴリー	クライアントおよびデスクトップ向けアプリケーション	サーバー、データベースおよび分散型アプリケーション	ハイパフォーマンス・コンピューティング(HPC)アプリケーション
リタイア	**20-50%**	**10-30%**	**30-70%**

詳しい情報は以下をご覧ください:http://software.intel.com/en-us/articles/how-to-tune-applications-using-a-top-down-characterization-of-microarchitectural-issues

© 2017, Intel Corporation. 転載を禁じます。* 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。

効率法 2: 命令あたりのサイクル数(CPI)の変化

なぜ重要なのか:2 つのデータセットを比較する際に役立つ、もう一つの効率の尺度

- アプリケーションのワークロードの 1 部を実行する命令の平均クロックを表示します

どのように検出するか:”General Exploration” プロファイルの測定基準: *CPI Rate*

何を行うか:

- CPI は、アプリケーションとプラットフォームに依存し大きく変化します
- コードサイズが変わらないなら、最適化では CPI を減らすことに注目すべきです

General Exploration General Exploration viewpoint (change)

Analysis Target | Analysis Type | Collection Log | Summary | Bottom-up | Event Co

Grouping: Function / Call Stack

Function / Call Stack	Clockticks	Instructions Retired	CPI Rate	Retiring	Front-End Bound
grid_intersect	14,948,022,422	12,672,019,008	1.180	14.3%	7.0%
sphere_intersect	9,618,014,427	9,026,013,539	1.066	16.6%	5.7%
grid_bounds_intersect	1,064,001,596	690,001,035	1.542	12.5%	8.0%
func@0x1002df7f	602,000,903	420,000,630	1.433	48.9%	22.1%
pos2grid	246,000,369	160,000,240	1.538	23.2%	7.7%
shader	212,000,318	190,000,285	1.11[illegible]	4.5%	4.5%
tri_intersect	198,000,297	192,000,288	1.03[illegible]	14.4%	4.8%
__kmp_x86_pause	184,000,276	82,000,123	2.2[illegible]4	100.0%	0.0%
Raypnt	172,000,258	256,000,384	[illegible].672	22.1%	16.6%
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Selected 1 row(s):	14,948,022,422	12,672,019,008	1.180	14.3%	7.0%

© 2017, Intel Corporation. 転載を禁じます。* 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。

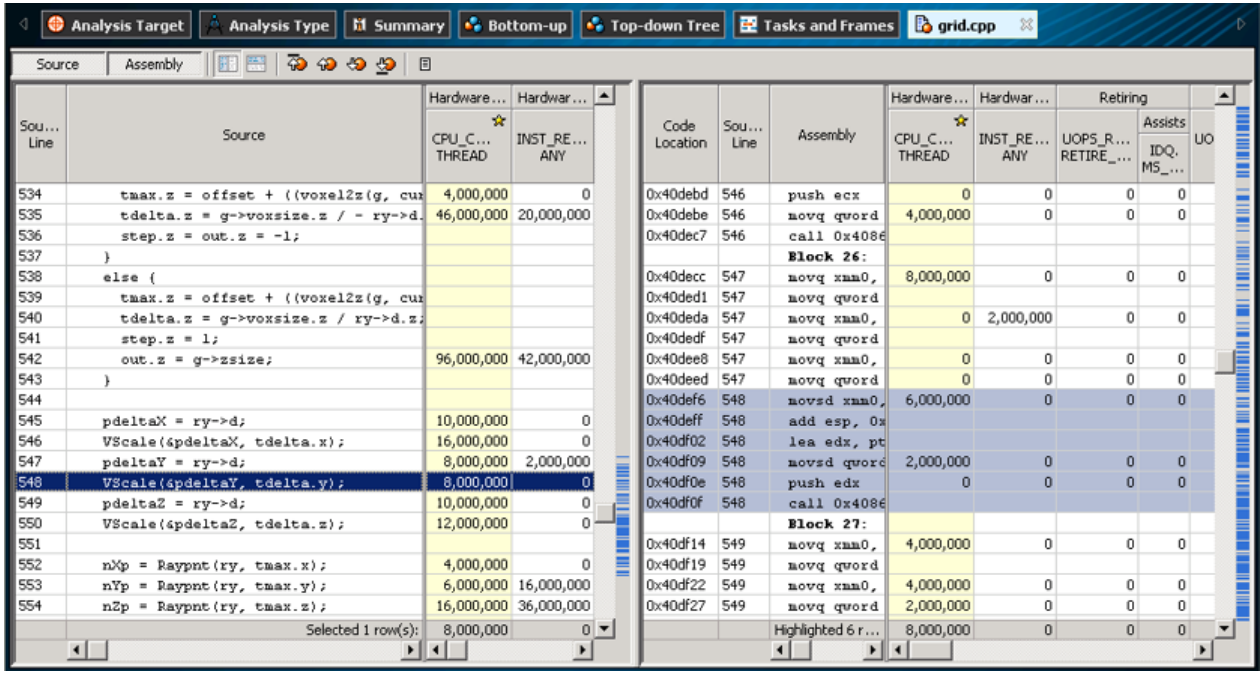
効率法 3: コードを調査

なぜ重要なのか：効率法の 1 と 2 では、実行される命令の時間を測定しました。非効率である他の原因は、非常に多くの命令を実行することです

どのように検出するか：インテル® VTune™ Amplifier XE のソースと逆アセンブリー表示を利用します

何を行うか：

- 最新の命令を利用しないとコードサイズが大きくなります
- 潜在的な問題に関しては、次のスライドを参照



© 2017, Intel Corporation. 転載を禁じます。* 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。

効率法 3、コードスタディー 1：旧式の浮動小数点もしくは整数コードをインテル® アドバンスト・ベクトル拡張（AVX と AVX2）に変換します

なぜ重要なのか：SIMD 命令を使用することで、浮動小数点演算のパフォーマンスを大幅に向上できます。既存の FP もしくは整数 SSE コードを AVX 命令に変換すると、広いベクトルデータ（256 ビットまで）のサポート、NDS 操作を可能にする 3 もしくは 4 オペランドの構文、そして電力節約などいくつかの利点があります

どのように検出するか：既存の SSE 命令（xmm レジスターを検索）、MMX 命令（mm レジスターを検索）、もしくは x87 浮動小数点命令のアセンブリー・コードを調査します。パック操作を行っていない（faddp、fmul、や addss などのスカラー SSE 命令）コードを特定します

何を行うか：

- インテル® コンパイラーの /QxCORE-AVX2 (Windows*) や -xCORE-AVX2 (Linux*) オプションを使用
- GCC: -march=core-avx2
- AVX への最適化に関しては、Intel® 64 and IA-32 Architectures Optimization Reference Manual の、第 11 章をご覧ください

© 2017, Intel Corporation. 転載を禁じます。* 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。

効率法 3、コードスタディー 2：Fused Multiply Add (FMA) 命令の利点を活用する

なぜ重要なのか：インテル・マイクロプロセッサー (コード名: Haswell) の FMA 命令は、FP 乗算命令とレイテンシーは同じです。新たな 2 つの FMA ユニットは、2倍のピーク FLOPs/サイクルを可能にします

どのように検出するか：r=(x*y)+z、r=(x*y)-z、r=-(x*y)+z、もしくは r=-(x*y)-z 形式の操作を行うソースコードを探します。該当するアセンブリーコードを見て、FMA 命令が使用されているか調べます。FMA 命令は、**VFM** (VFMADD132PD や VFMSUBADD132PD のように) もしくは **VFNM** (VFNMADD132PD のように) という名前で始まります

何を行うか：

- FMA を生成するため、コンパイラーのオプションを使用します (精度のゆるい浮動小数点モデルを利用した方がいいでしょう)
- インテル® コンパイラー：-fma もしくは /Qfma と CORE-AVX2 以上の引数を持つ -x か /Qx 、または -march (Linux*) か /arch (Windows*) オプション
- GCC: -xfma もしくは -march=core-avx2

© 2017, Intel Corporation. 転載を禁じます。* 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。

"ハードウェア上のソフトウェア" のチューニング手順

各ホットスポットで

- 効率を判断
  - もし十分でなければ:
    - 主なボトルネックを特定
    - 非効率であるアーキテクチャー上の理由を見つけます
    - 問題を最適化します

繰り返します!

© 2017, Intel Corporation. 転載を禁じます。* 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。

主なボトルネックを特定

コードが非効率であるかどうかを調べるため、方法 1 もしくは 2 を適用している場合、最初に主要なボトルネックを見つけます

“General Exploration” で実装されるトップダウンの階層は、アプリケーションの利用率を CPU コアの 4 つのカテゴリーに分類します：

- フロントエンド依存
- バックエンド依存
- 投機の問題
- Retiring（リタイアした比率）

主要なボトルネックは、パイプラインの断片化が高く、
最初に調査しなければいけません

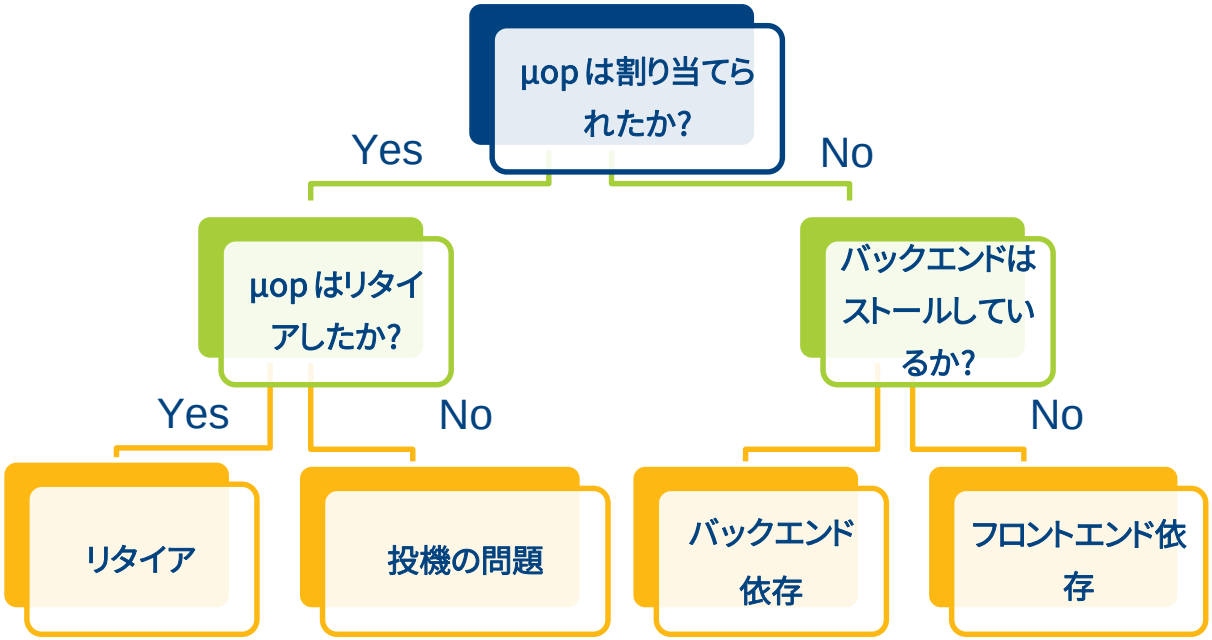
© 2017, Intel Corporation. 転載を禁じます。* その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

問題の分類

パイプライン・スロットは、抽象的なコンセプトであり、1 つのマイクロオペレーションを操作するために必要なハードウェアのリソースを表します

このプロセッサーでは、各コアごとに各サイクルで 4 つのパイプライン・スロットがあります

アプリケーションやホットスポットで利用可能な各パイプライン・スロットで何が起こっているかに応じて、パフォーマンスは分類されます



© 2017, Intel Corporation. 転載を禁じます。* 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。

ボトルネックの調査

どのカテゴリーが主要なボトルネックであるかを判断し、次にそのカテゴリーを以下の表と比較します

- コードのホットスポットの比率(%)が範囲外であれば、そのカテゴリーから問題を調査します

	各カテゴリーのパイプライン・スロットの期待される範囲(うまく最適化されたホットスポット向け):		
カテゴリー	クライアントおよびデスクトップ向けアプリケーション	サーバー、データベースおよび分散型アプリケーション	ハイパフォーマンス・コンピューティング(HPC)アプリケーション
リタイア ↑	20-50%	10-30%	30-70%
バックエンド依存 ↓	20-40%	20-60%	20-40%
フロントエンド依存 ↓	5-10%	10-25%	5-10%
投機の問題 ↓	5-10%	5-10%	1-5%

要点: ↑ 高い値が良い
↓ 低い値が良い

© 2017, Intel Corporation. 転載を禁じます。* 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。

"ハードウェア上のソフトウェア" のチューニング手順

各ホットスポットで

- 効率を判断
  - もし十分でなければ：
    - 主なボトルネックを特定
    - 非効率であるアーキテクチャー上の理由を見つけます
    - 問題を最適化します

繰り返します!

© 2017, Intel Corporation. 転載を禁じます。* 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。

非効率であるアーキテクチャー上の理由を見つけます：問題のスライド

問題はカテゴリー別にリストされ、カテゴリーごとに説明されます

潜在的な問題ごとに、いくつかの重要な情報が示されます

なぜ重要なのか： なぜ潜在的な問題について考えなければいけないのか。

どのように検出するか： インテル® VTune™ Amplifire XE のどのプロファイルと測定基準を使用するか。表示項目がハイライトされている場合、その項目を調査する必要があります

何を行うか： **問題を最適化する**のに役立ちます。調査を続行、もしくは最適化すべき提案を提示します

イベント名と測定基準の式は別途示されます。VTune Amplifier XE のロジックに組み込まれているため、このスライドには含まれません。開発者は、事前定義されたプロファイルと測定基準のみを使用して、問題があるかどうかを指摘されるだけです

© 2017, Intel Corporation. 転載を禁じます。* 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。

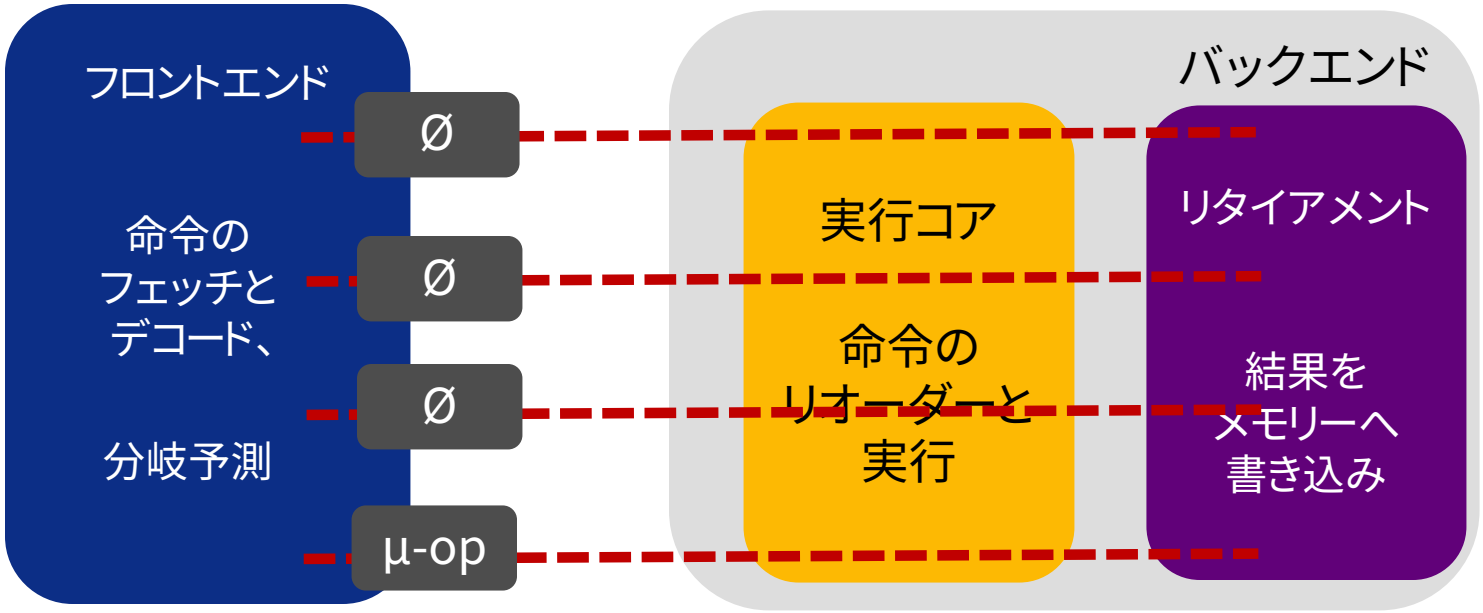
フロントエンド依存のカテゴリーをチューニング

パイプラインのフロントエンド

- 命令をフェッチ
- 命令をマイクロオペレーションにデコード
- サイクルごとに最大 4 マイクロオペレーションをバックエンドへ供給

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

フロントエンド依存の概要



フロントエンドが、4つのパイプライン・スロット全てに
マイクロオペレーションを供給できない時に発生:
一般に、コードのフェッチや命令デコードの遅延が原因

© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® VTune™ Amplifier XE におけるフロントエンド階層のブレークダウン

Retiring	Front-End Bound	Bad Speculation	Back-End Bound
14.3%	7.0%	3.9%	74.8%
16.6%	5.7%	4.5%	73.1%
12.5%	8.0%	4.5%	75.0%
48.9%	22.1%	36.3%	0.0%
23.2%	7.7%	0.0%	84.6%
[illegible]	[illegible]	[illegible]	91.0%
[illegible]	[illegible]	[illegible]	[illegible]

Front-End Bound	
Front-End Latency	Front-End Bandwidth
5.6%	1.4%
4.7%	1.0%
7.1%	0.9%
6.3%	15.8%
15.4%	0.0%
0.	
0.	

Front-End Bound								
Front-End Latency						Front-End Band...		
ICac... Miss...	ITLB Ove...	Bran... Rest...	DSB Swit...	Len... Cha.. Prefi...	MS Swit...	Fron... Band.. DSB	Fron... Band.. MITE	Fron... Band.. LSD
0.000	0.000	0.083	0.008	0.000	0.000	0.109	0.031	0.000
0.000	0.000	0.020	0.008	0.000	0.000	0.036	0.008	0.000
0.000	0.000	0.179	0.000	0.000	0.000	0.000	0.107	0.000
0.000	0.003	0.000	0.000	0.000	0.000	0.568	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.618	0.000

フロントエンド依存のカラムを展開して、マイクロオペレーションが供給されなかった "Front-End Latency" と 4 マイクロオペレーション以下が供給された "Front-End Bandwidth" に分類されるフロントエンド依存の比率を確認できます

フロントエンド依存が主要なボトルネックである場合、フロントエンド・レイテンシーに注目します。このカテゴリー下のハイライトされた問題を解決します

© 2017, Intel Corporation. 転載を禁じます。* 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。

フロントエンドのレイテンシー

なぜ重要なのか： フロントエンド・レイテンシーは、バックエンドで実行するマイクロオペレーションが不足する原因となります (命令スタベーション)

どのように検出するか： フロントエンド・レイテンシーのサブカテゴリ―測定基準: *ITLB Overhead*、*ICache Misses*、*Length-Changing Prefixes*

何を行うか： ホットスポットでこれらの測定基準がハイライトされている場合、より良いコード配置と生成テクニックを試みます：

- コンパイラーのプロファイルに基づく最適化 (PGO) を試してください
- リンカーの順序付けテクニックを使用します (Microsoft リンカーでは /ORDER、GCC ではリンカースクリプト)
- コードサイズを減らすには、/O1 や /Os オプションを使用します
- 動的に生成されたコードは、ホットなコードを同じ場所に配置し、間接呼び出しを回避するようにしましょう

© 2017, Intel Corporation. 転載を禁じます。* 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。

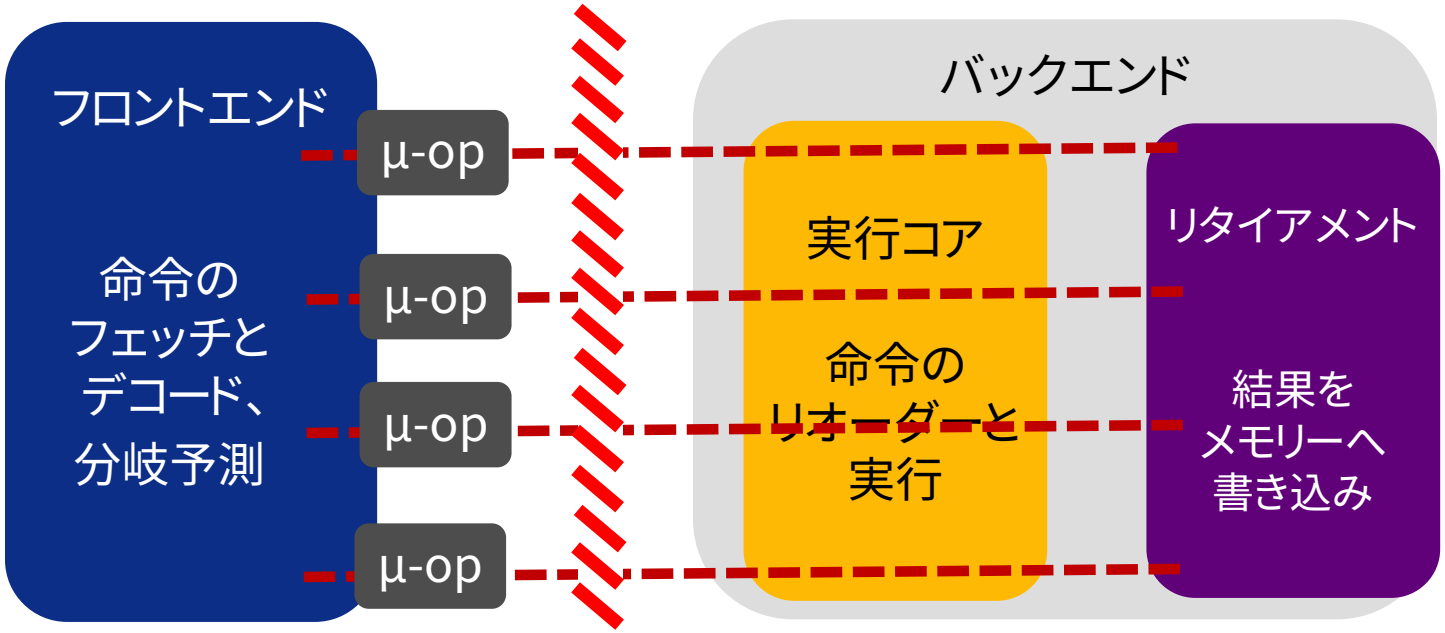
バックエンド依存のカテゴリーをチューニング

パイプラインのバックエンド

- フロントエンドからのマイクロオペレーションを受け取る
- 実行ユニット内の実行をスケジュールするため、必要に応じて命令をリオーダー(入れ替え) する
- メモリーから必要なオペランドを取り込む
- 操作を実行
- 結果をメモリーへ書き込み

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

バックエンド依存の概要



バックエンドが 4 つのパイプライン・スロット全てに
マイクロオペレーションを受け入れることができない時に発生:
通常、内部構造がすべてマイクロオペレーションで埋まっていて
データ待ちの状態

© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® VTune™ Amplifier XE における バックエンド階層のブレークダウン

Retiring	Front-End Bound	Bad Speculation	Back-End Bound
14.3%	7.0%	3.9%	74.8%
16.6%	5.7%	4.5%	73.1%
12.5%	8.0%	4.5%	75.0%
48.9%	22.1%	36.3%	0.0%
23.2%	7.7%	0.0%	84.6%
4.5%	4.5%	0.0%	91.0%

Back-End Bound	
Memory Bound	Core Bound
40.8%	34.0%
42.5%	30.7%
48.2%	26.8%
0.0%	0.0%
0.0%	84.6%
18.2%	72.8%
0.0%	76.0%
0.0%	17.4%
22.4%	22.4%
0.0%	0.5%
40.8%	34.0%

バックエンド依存のカラムを展開して、多くのメモリー操作が実行中であるためバックエンドが新しいマイクロオペレーションを受け入れることができない "Memory Bound" と実行ポートの飽和問題 "Core Bound" に分類されるバックエンドの全ての比率を確認します

© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® VTune™ Amplifier XE における バックエンド階層のブレークダウン

"Memory Bound" のカテゴリーを展開して、メモリー階層の各レベルに関連する問題を確認します

Back-End Bou	
Memory Bound	Core Bound
40.8%	34.0%
42.5%	30.7%
48.2%	26.8%
0.0%	0.0%
0.0%	84.6%
18.2%	72.8%
0.0%	76.0%
0.0%	17.4%
22.4%	22.4%
0.0%	0.5%
40.8%	34.0%

Back-End Bound			
Memory Bound			
L1 Bound	L2 Bound	L3 Bound	DRAM Bound
0.159	0.083	0.031	0.000
0.104	0.020	0.052	0.000
0.287	0.000	0.048	0.000
0.498	0.000	0.000	0.000

Back-End Bound														
Memory Bound														
L1 Bound						L2 Bou..	L3 Bound					DRAM Bound		Store Bound
DTLB Ove...	Loa... Bloc...	Lock Late...	Split Loads	4K Alia...	FB Full		Con... Acc...	Data Shar...	L3 Late...	L3 Band..	SQ Full	Memory Bandwidth	Memory La... LLC Miss	
0.094	0.000	0.000	0.000	0.005	0.000	0.063	0.000	0.000	0.432	0.356	0.000	0.000	0.000	0.000
0.084	0.000	0.000	0.000	0.012	0.000	0.071	0.000	0.000	0.547	0.369	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.342	0.000	0.000	0.000	0.000	0.000
0.003	0.000	0.000	0.000	0.020	0.000	0.058	0.000	0.000	0.596	0.404	0.000	0.000	0.000	0.000
0.000	0.000	0.000		0.000		0.000	0.000	0.000	0.411	0.247	0.000	0.000	0.000	0.000

© 2017, Intel Corporation. 転載を禁じます。* 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。

キャッシュミス

なぜ重要なのか： キャッシュミスは、アプリケーションの CPI を高めます。レベル 2 もしくは 3 ミスによる長いレイテンシーのデータに注目します

どのように検出するか： “メモリー依存” のサブカテゴリー、測定基準：
L3 Latency、LLC Miss

何を行うか： ホットスポットでどちらの測定基準もハイライトされている場合、ミスを減らすことを考慮します：

- データストレージを減らすためアルゴリズムを変更
- キャッシュに収まるようにブロック化してデータをアクセス
- 共有の問題を調査 (アクセス競合を参照)
- ベクトル化のためデータをアライメント (コンパイラーにそれを知らせます)
- キャッシュライン入れ換え解析の概要に関しては、”インテル® 64 および IA-32 アーキテクチャー最適化リファレンス・マニュアル” の B.3.4.2 節をご覧ください
- ストリーミング・ストアを利用
- ソフトウェア・プリフェッチ命令を利用

© 2017, Intel Corporation. 転載を禁じます。* 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。

リモート・メモリー・アクセス

なぜ重要なのか： NUMA アーキテクチャーでは、メモリーの位置によってレイテンシーが異なります。リモートメモリーの読み込みコストは高くなります

どのように検出するか： ”General Exploration” プロファイルの測定基準：*Remote DRAM*

何を行うか： ホットスポットで測定基準がハイライトされている場合、NUMA アフィニティーを改善します：

- スレッドの移動の問題である場合、スレッドをコアに固定 (アフィニティーやピニング) することを試します
- メモリーを使用するスレッドが最初にメモリーを参照 (タッチする、割り当てではない) することを確実にします
- OpenMP では アフィニティー環境変数を使用
- アプリケーションが NUMA を意識したオプションをサポートしていれば使用 (SQL サーバーの softnuma など)
- NUMA 効率の良いスレッド・スケジューラーを使用 (インテル® スレッディング・ビルディング・ブロックやインテル® Cilk™ Plus など)

© 2017, Intel Corporation. 転載を禁じます。* 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。

NUMA の問題の特定

次の階層で組織化されたメトリックに注目:

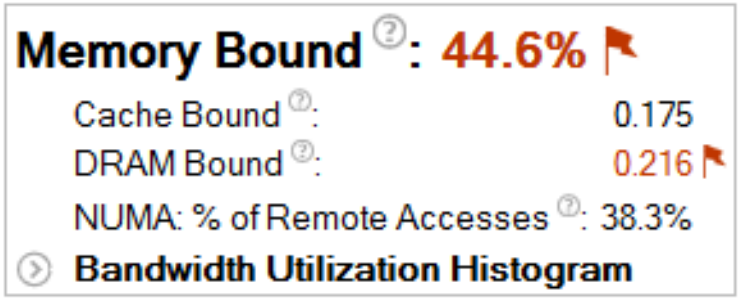
Memory Bound > DRAM Bound > Local DRAM

Memory Bound > DRAM Bound > Remote DRAM

Memory Bound > DRAM Bound > Remote Cache

LLC Miss Count > Local DRAM Access Count、
LLC Miss Count > Remote DRAM Access Count、
LLC Miss Count > Remote Cache Access Count

Average Latency (cycles)



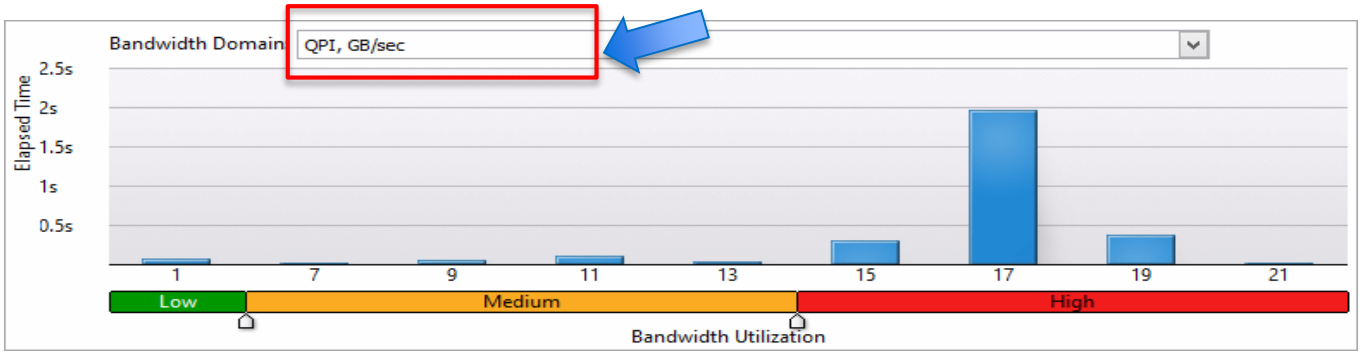
計画: Apache Pass NUMA メトリックの拡張

© 2017, Intel Corporation. 転載を禁じます。* 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。

NUMA の問題の特定

2016 以降

インテル® QuickPath インターコネクト帯域幅ドメインと帯域幅利用率のヒストグラム:



Grouping: Bandwidth Domain / Bandwidth Utilization Type / Function / Call Stack

Bandwidth Domain / Bandwidth Utilization Type / Function / Call Stack	CPU Time	Memory Bound	Loads	Stores	LLC Miss Count
DRAM, GB/sec	2.873s	0.818	1,764,005,292	846,012,690	37,202,232
DRAM Read, GB/sec	2.873s	0.818	1,764,005,292	846,012,690	37,202,232
DRAM Write, GB/sec	2.873s	0.818	1,764,005,292	846,012,690	37,202,232
QPI, GB/sec	2.873s	0.818	1,764,005,292	846,012,690	37,202,232
High	2.570s	1.000	1,326,003,978	742,811,142	33,001,980
main	1.996s	0.699	1,068,003,204	608,409,126	33,001,980
__intel_ssse3_rep_memcpy	0.566s	0.000	252,000,756	128,401,926	0
func@0x66ee8	0.006s	0.000	0	0	0
[vmlinux]	0.001s	0.000	6,000,018	0	0
func@0x1a2df4	0.001s	0.000	0	0	0

© 2017, Intel Corporation. 転載を禁じます。* 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。

Contested Accesses (競合するアクセス)

なぜ重要なのか： コア間にまたがって（L2 レベルで）共有されるデータの変更は、データアクセスのレイテンシーを増加させます

どのように検出するか： ”メモリー依存” のサブカテゴリー、測定基準：Contested Accesses (競合するアクセス)

何を行うか： ホットスポットでこの測定基準がハイライトされた場合、ソース表示から HITM を生成しているソースコードの行を見つけます

- HITM を生成した命令の次の命令にタグ付けされる MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HITM_PS イベントを探します
- コードを詳しく調べて、真の共有か偽りの共有化を判断します。適切な対策を施します：
  - 真の共有に対しては、共有の要求を減らします
  - 偽りの共有には、キャッシュライン境界にパディングを挿入します

© 2017, Intel Corporation. 転載を禁じます。* 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。

データ共有

なぜ重要なのか： (L2 レベルで) コア間のクリーンなデータ共有 (読み込み共有) は、一貫性を維持するため少なくとも最初の参照でペナルティーが発生します

どのように検出するか： ”メモリー依存” のサブカテゴリー、測定基準：Data Sharing (データ共有)

何を行うか： ホットスポットでこの測定基準がハイライトされた場合、ソース表示から HIT を生成しているソースコードの行を見つけます:

- HITM を生成した命令の次の命令にタグ付けされる MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HIT_PS イベントを探します
- コードを詳しく調べて、真の共有か偽りの共有化を判断します。適切な対策を施します：
  - 真の共有に対しては、共有の要求を減らします
  - 偽りの共有には、キャッシュライン境界にパディングを挿入します

© 2017, Intel Corporation. 転載を禁じます。* 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。

その他のデータアクセスの問題：ストア・フォワードが行われないことによるロードのブロック

なぜ重要なのか： ストアの結果がパイプラインを介してフォワードできない場合、依存するロードがブロックされることがあります

どのように検出するか： ”メモリー依存” のサブカテゴリー、測定基準：Loads Blocked by Store Forwarding (ストアフォワードでブロックされたロード)

何を行うか： ホットスポットで測定基準がハイライトされている場合、以下を調査します：

ソースを参照し、LD_BLOCKS_STORE_FORWARD イベントを探します。通常このイベントは、ブロックされたロードの次の命令にタグ付けされます。ロード命令の位置を確認し、フォワードできないストア命令を探します。通常 10 -15 命令以内にあります。もっとも一般的なケースは、同じアドレスに対するロードよりも小さな断片をストアする場合です。後続のロードと同じもしくは大きなサイズのデータをストアすることで問題を解決します

© 2017, Intel Corporation. 転載を禁じます。* 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。

その他のデータアクセスの問題:4K エイリアス

なぜ重要なのか: エイリアスの競合により、ロードを再発行しなければいけません

どのように検出するか: ”メモリー依存” のサブカテゴリー、測定基準:4K Aliasing

何を行うか: ホットスポットでこの測定基準がハイライトされている場合、ソースコード・レベルで測定基準を識別して調査します:

ロードのアライメントを変更することで、この問題を解決します。入力と出力バッファー間のオフセットを変更することで(可能であれば)、データを 32 バイトにアライメントするか、32 バイトにアライメントされていないメモリーへのアクセスに 16 バイトのメモリーアクセスを使用します

© 2017, Intel Corporation. 転載を禁じます。* 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。

その他のデータアクセスの問題：DTLB ミス

なぜ重要なのか： 最初のレベルの DTLB ロードミス（STLB にヒット）は、レイテンシーのペナルティーがかかります。第 2 レベルのミスではページウォークが必要となり、アプリケーションのパフォーマンスに影響します

どのように検出するか： ”メモリー依存” のサブカテゴリー、測定基準：DTLB Overhead

何を行うか： ホットスポットでこの測定基準がハイライトされている場合、ソースコード・レベルで測定基準を識別して調査します:

この問題を解決するには、より優れたメモリー割り当て機能を利用したり、プロファイル・ガイド最適化により、ターゲットデータの局所性を TLB のサイズに合わせるか、仮想化システムでは拡張ページテーブル（EPT) を使用するか、ラージページを使用するか、（データベースやサーバー・アプリケーションのみで）、データの局所性を高めます

© 2017, Intel Corporation. 転載を禁じます。* 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。

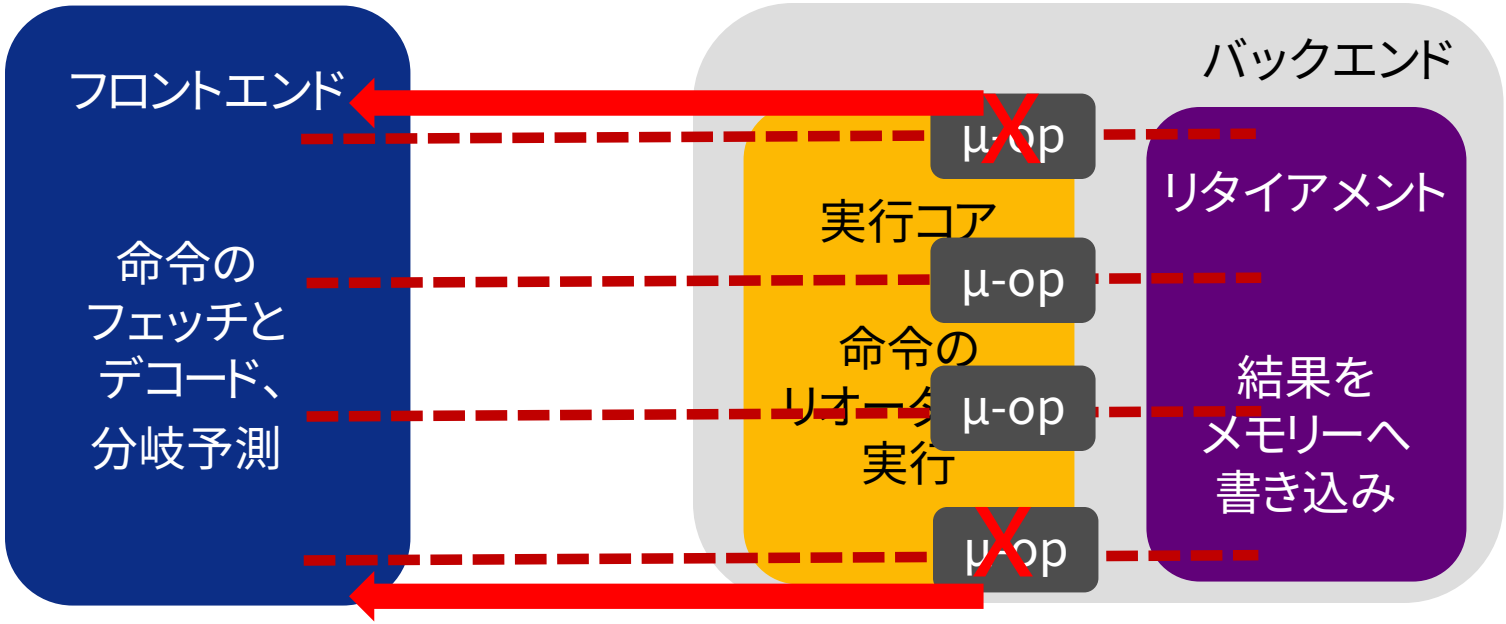
投機の問題のカテゴリーをチューニング

投機が行われる場合：

- マイクロ・オペレーションは、それがリタイアするかどうかにかかわらず実行が許可されます
- これは、アウト・オブ・オーダー・パイプラインで高い命令レベルの並列性をもたらします

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

投機の問題の概要



マイクロオペレーションがバックエンドから削除され、
リタイアしない場合に発生

© 2017, Intel Corporation. 転載を禁じます。* 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。

分岐予測ミス

なぜ重要なのか： 誤って予測された分岐は、無駄な処理や命令スタベーション（新しい命令がフェッチされるまで待機するため）により、パイプラインの効率を低下させます

どのように検出するか： ”General Exploration” プロファイル、測定基準：Branch Mispredict

何を行うか： ホットスポットでどちらの測定基準もハイライトされている場合、予測ミスを減らすことを考慮します:

- コンパイラー・オプションやプロファイル・ガイド最適化によってコード生成を改選します
- 分岐文において、もっとも分岐するターゲットが主な命令フローとなるように手動で調整します

© 2017, Intel Corporation. 転載を禁じます。* 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。

マシンクリア

なぜ重要なのか：マシンクリアは、パイプラインをフラッシュし、ストアバッファーを空にするため、大幅なレイテンシーのペナルティーとなります

どのように検出するか： ”General Exploration” プロファイル、測定基準：Machine Clears

何を行うか： ホットスポットでこの測定基準がハイライトされている場合、幾つかのイベントを使用して原因を特定します:

- MACHINE_CLEARS.MEMORY_ORDERING イベントが多い場合、ソースコード・レベルで調査します。これは、4K エリアス競合やロック競合によって引き起こされます
- MACHINE_CLEARS.SMC イベントが多い場合、自己修正コードによる原因です。これは避けるべきです

© 2017, Intel Corporation. 転載を禁じます。* 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。

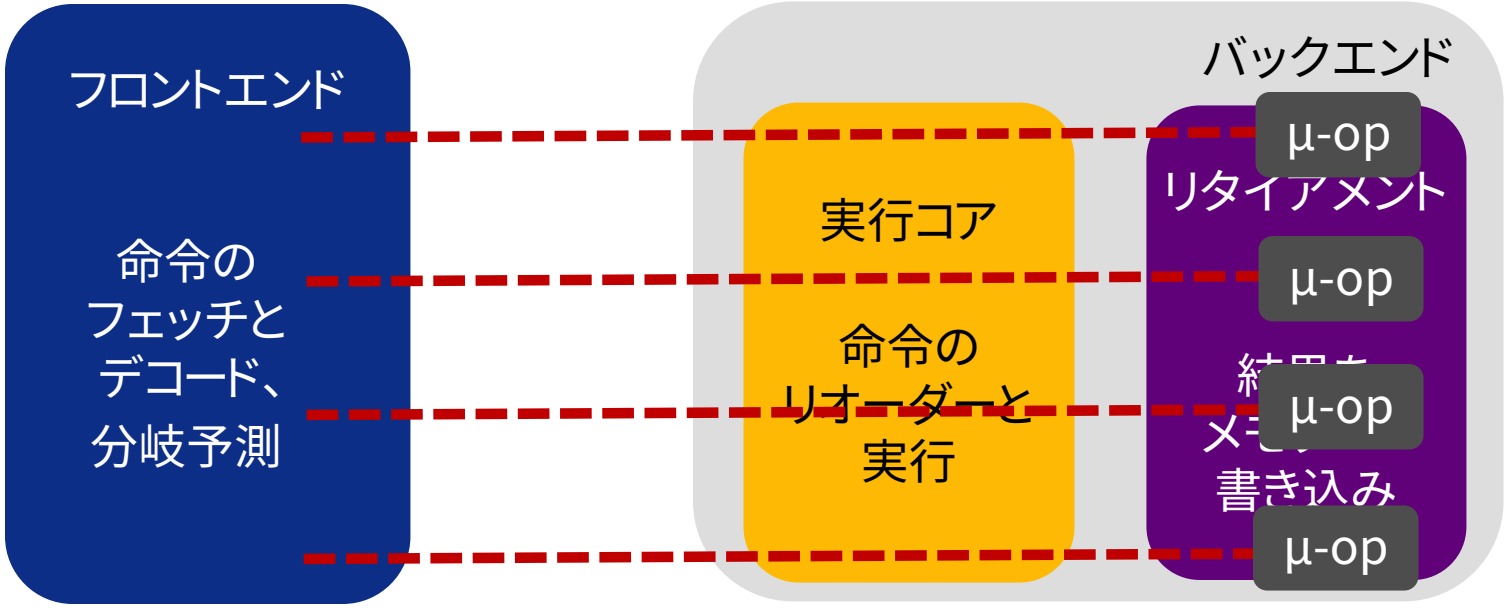
リタイア・カテゴリーのチューニング

リタイアメントは：

- マイクロオペレーション実行の完了を示します
- マイクロオペレーションが命令を構成する最後のオペレーションであれば、命令実行の完了を意味します
- 命令の実行結果は、アーキテクチャー状態（キャッシュ、メモリーなど）にコミットされます

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

リタイアの概要



パイプライン・スロットがマイクロオペレーションで満たされた時に発生します。これは望ましいカテゴリーです。いくつかの問題もあります

© 2017, Intel Corporation. 転載を禁じます。* 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。

インテル® VTune™ Amplifier XE におけるリタイア階層のブレークダウン

Retiring	Front-End Bound	Bad Speculation	Back-End Bound
14.3%	7.0%	3.9%	74.8%
16.6%	5.7%	4.5%	73.1%
12.5%	8.0%	4.5%	75.0%
48.9%	22.1%	36.3%	0.0%
23.2%	7.7%	0.0%	84.6%
4.5%	4.5%	0.0%	91.0%

最良のケースである "General Retirement" とマイクロコード・シーケンサーから生成されたマイクロオペレーションのリタイアである "Microcode Sequencer" で区分されるリタイアスロットの比率を見るには、リタイアを展開表示します

Retiring		
General Retirement		Microcode Sequencer
FP Arithmetic	Other	
0.140	0.860	0.000
0.192	0.808	0.000
0.000	1.000	0.000
0.000	1.000	0.000

パフォーマンスの問題を解決することで、ほとんどの場合「General Retirement」に分類される µops は増加します

© 2017, Intel Corporation. 転載を禁じます。* 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。

FP 算術演算

なぜ重要なのか： 非効率な浮動小数点演算は高いコストにつながります

どのように検出するか： *“General Exploration”*プロファイル、測定基準：*FP Arithmetic*、*FP x87*、*FP Scalar*、*FP Vector*

何を行うか： FP x87 または FP スカラー測定基準が顕著な場合、ベクトル化の効率を高めてください:

- インテル® コンパイラーの /QxCORE-AVX2 (Windows®) や -xCORE-AVX2 (Linux*) オプションを使用
- GCC: -march=core-avx2
- AVX への最適化に関しては、インテル® 64 および IA-32 アーキテクチャー最適化リファレンス・マニュアル の、第 11 章をご覧ください。

General Retirement			
FP Arithmetic			Other
FP x87	FP Scalar	FP Vector	
0.000	0.140	0.000	0.860
0.000	0.192	0.000	0.808
0.000	0.000	0.000	1.000

© 2017, Intel Corporation. 転載を禁じます。* 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。

マイクロコード・アシスト

なぜ重要なのか： マイクロコード・シーケンサーからのアシストには、長いレイテンシーのペナルティーが伴います

どのように検出するか： ”General Exploration” プロファイル、測定基準：Microcode Sequencer

何を行うか： ホットスポットでこの測定基準がハイライトされた場合、他のアシストイベントを使用して原因を特定するため再度サンプリングを行います:

- もし、FP_ASSISTS.ANY / INST_RETIRED.ANY がかなり多いようであれば、浮動小数点演算のデノーマルを調べてください。SSE/AVX 命令を使用している場合、FTZ と（もしくは）DAZ を有効にするか、演算をスケールアップもしくはダウンすることで解決できます
- ((OTHER_ASSISTS.AVX_TO_SSE_NP*75) / CPU_CLK_UNHALTED.THREAD) もしくは、((OTHER_ASSISTS.SSE_TO_AVX_NP*75) / CPU_CLK_UNHALTED.THREAD) が、.1 より大きければ SSE と AVX コード間の移行を減らします。詳細は http://software.intel.com/en-us/articles/avoiding-avx-sse-transition-penalties を参照してください

© 2017, Intel Corporation. 転載を禁じます。* 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。

"ハードウェア上のソフトウェア" のチューニング手順

各ホットスポットで

- 効率を判断
  - もし十分でなければ：
    - 主なボトルネックを特定
    - 非効率であるアーキテクチャー上の理由を見つけます
    - 問題を最適化します

繰り返します！

目立ったホットスポットが無くなるまで繰り返すと、
リタイアのパイプライン・スロットは期待どうりとなるでしょう

© 2017, Intel Corporation. 転載を禁じます。* 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。

その他のトピック: メモリー帯域幅

なぜ重要なのか： 帯域幅のボトルネックは、キャッシュミスが発生した時にレイテンシーを増加させます

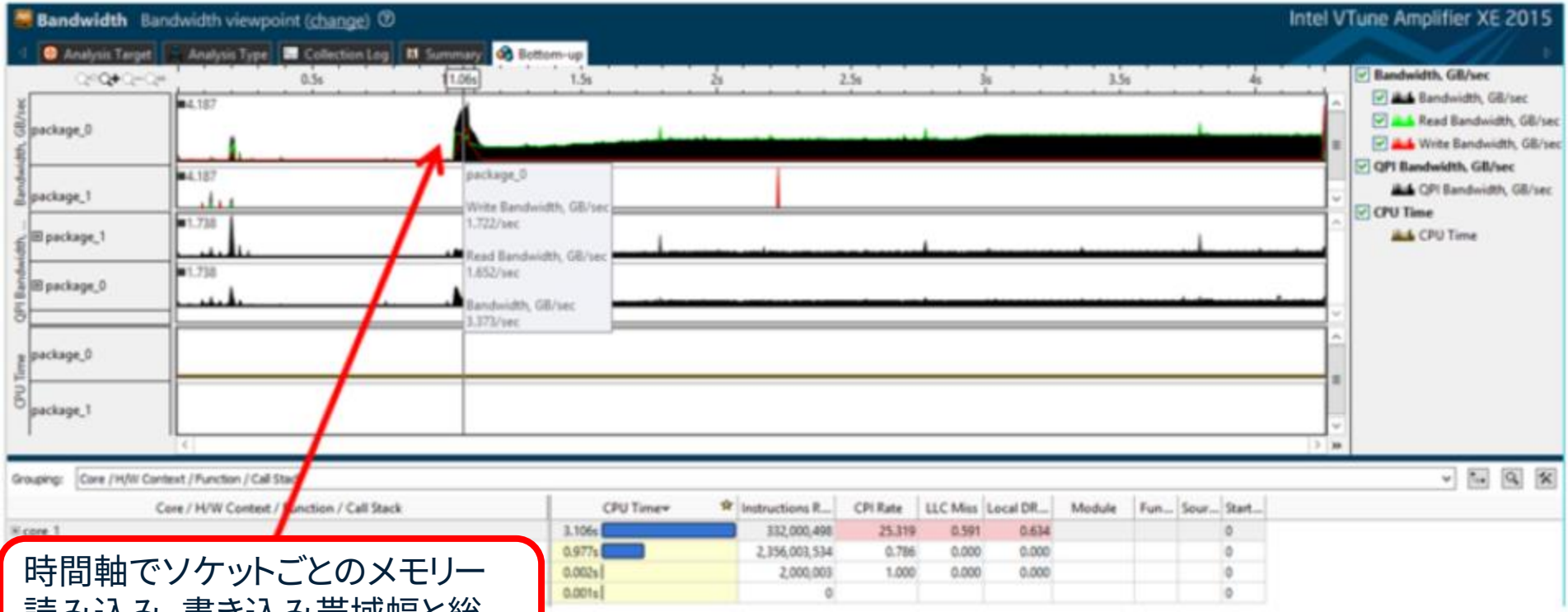
どのように検出するか： 帯域幅解析

何を行うか：

- プラットフォームのソケットごとにメモリーの理論的な最大帯域幅を GB/秒で計算します: (<MT/秒> * 8 バイト/クロック * <チャンネル数>) / 1000
- アプリケーションを実行してで帯域幅を計測します。ソケットごとの総帯域幅が 75% 以上である場合、アプリケーションには（高い）ロード・レイテンシーがある可能性があります
- 適切であれば、システムチューニングの調整を行います (メモリー DIMM のアップグレードやバランス調整、ハードウェア・プリフェッチャーの無効化)
- 可能であれば帯域幅の利用を軽減します：非効率な SW プリフェッチの削除、データ領域や共有を減らすアルゴリズムの変更、データ更新の軽減、ソケットに分散するメモリーアクセス・バランスの調整などが考えられます

© 2017, Intel Corporation. 転載を禁じます。* 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。

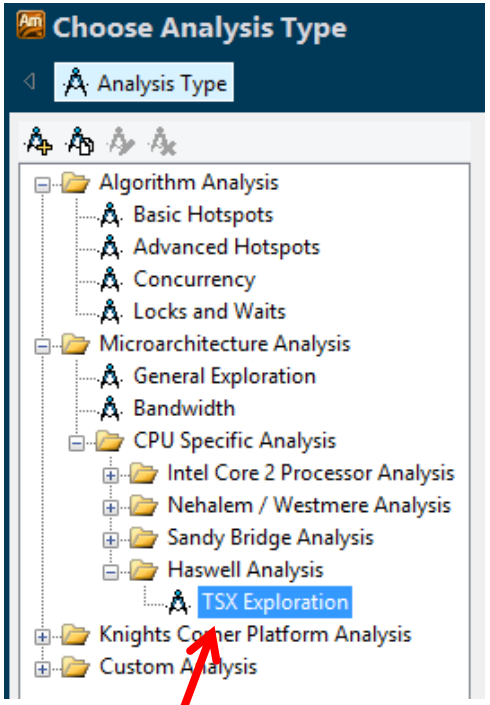
その他のトピック: メモリー帯域幅

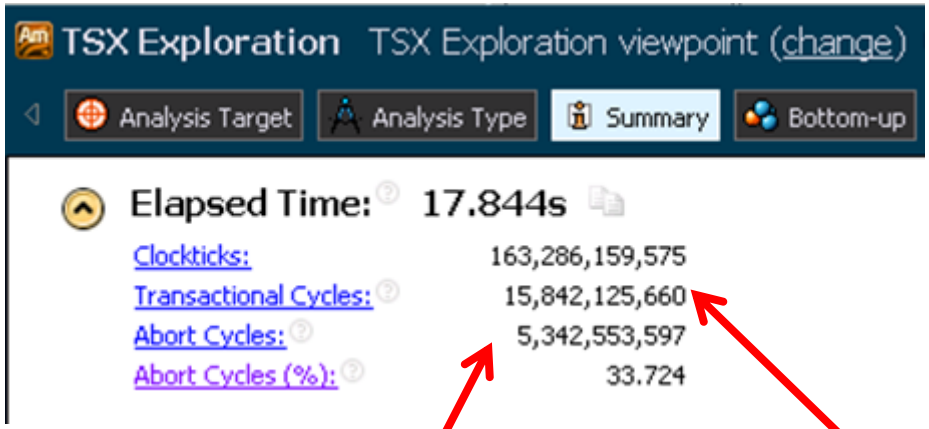


時間軸でソケットごとのメモリー読み込み、書き込み帯域幅と総帯域を観察します

© 2017, Intel Corporation. 転載を禁じます。* 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。

その他のトピック: TSX Exploration (TSX の調査)





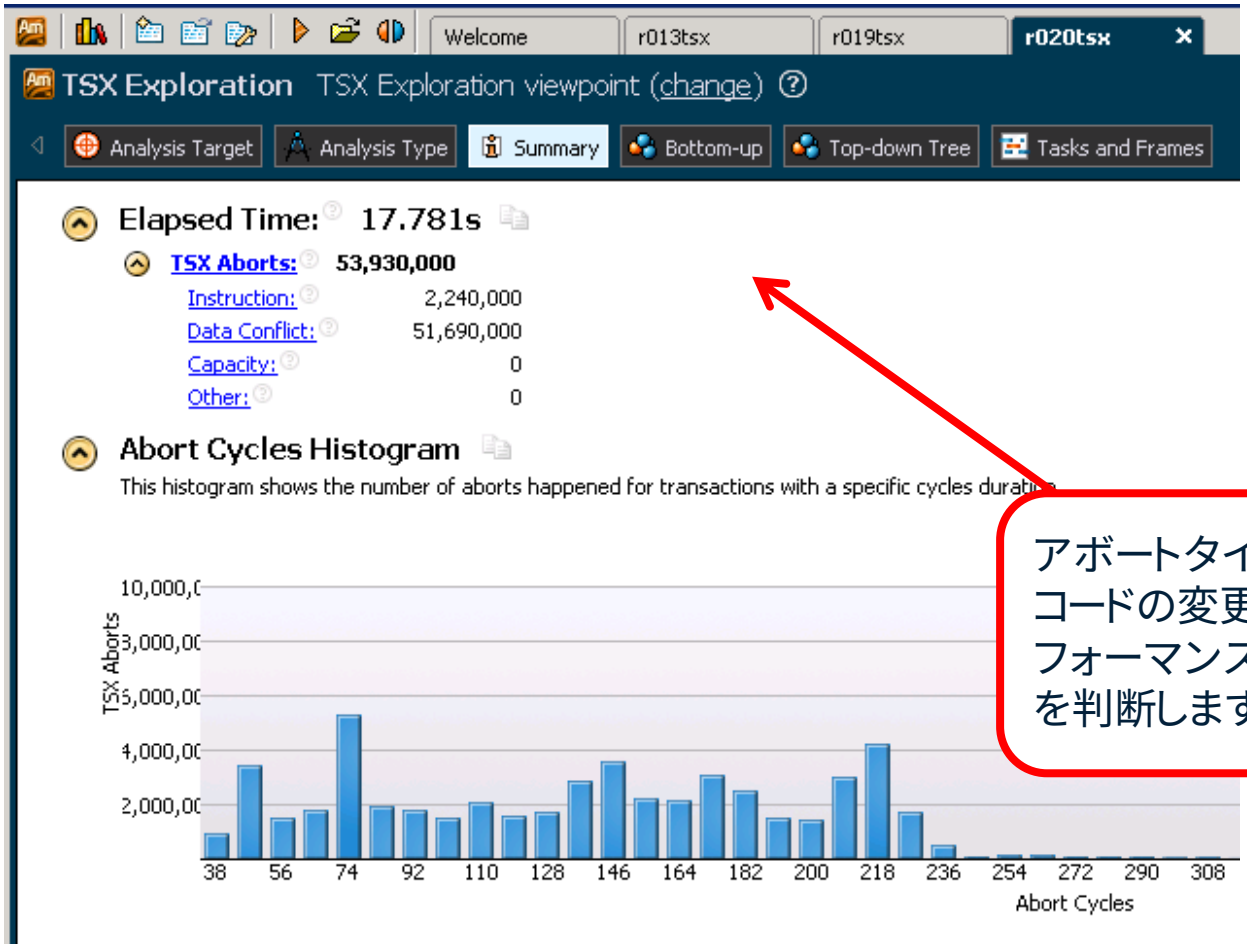
TSX 部分で費やされているサイクル数

アボートされている TSX 部分で費やされているサイクル数

TSX Exploration 解析タイプを選択します

© 2017, Intel Corporation. 転載を禁じます。* 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。

その他のトピック: TSX Exploration (TSX の調査)



アボートタイプをブレークダウンして、コードの変更がインテル® TSX のパフォーマンスを改善できるかどうかを判断します

© 2017, Intel Corporation. 転載を禁じます。* 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。

その他のトピック: Multiplexing Reliability (信頼性の多重化)

General Exploration General Exploration viewpoint (change)

Collection Log | Analysis Target | Analysis Type | Summary | Bottom-up | PMU Events | Platform

Grouping: Function / Call Stack

Function / Call Stack	Clocktic...	Instructions Retired	CPI Rate	Filled Pipeline Slots		Unfilled Pipeline Slots (Stalls)	
				Retiring	Bad Speculation	Back-End Bound	Front-End Bound
grid_intersect	14,076,021,114	12,468,018,702	1.129	0.210	0.076	0.650	0.063
sphere_intersect	9,306,013,959	9,206,013,809	1.011	0.282	0.038	0.615	0.065
grid_bounds_intersect	1,098,001,647	690,001,035	1.591	0.123	0.020	0.781	0.075
func@0x1002e3d5	922,001,383	700,001,050	1.317	0.000	0.000	1.000	0.000
__kmp_x86_pause	354,000,531	212,000,318	1.670	0.000	0.000	1.000	0.000
tri_intersect	222,000,333	152,000,228	1.461	0.405	0.000	0.561	0.101
pos2grid	212,000,318	186,000,279	1.140	0.248	0.000	0.717	0.035
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Selected 1 row(s):	14,076,021,114	12,468,018,702	1.129	0.210	0.076	0.650	0.063

グリッドの各行の測定基準に対する信頼性を確かめるには、MUX Reliability を使用します

© 2017, Intel Corporation. 転載を禁じます。* 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。

ありがとうございます！

より詳しい情報について：

インテル® VTune™ Amplifier XE 向けビデオ、フォーラムおよびリソース
http://www.isus.jp/intel-vtune-amplifier-xe/

インテル® 64 アーキテクチャーおよび IA-32 アーキテクチャー・ソフトウェア開発マニュアル:
http://www.intel.com/products/processor/manuals/index.htm

他のマイクロ・アーキテクチャー向けのインテル® VTune™ Amplifier XE のチューニング・ガイド：
http://www.isus.jp/products/vtune/processor-specific-performance-analysis-papers/

統合グラフィック・コントローラーの最適化：
www.intel.co.jp/jp/software/products

© 2017, Intel Corporation. 転載を禁じます。* 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。

法務的な免責事項

本資料に掲載されている情報は、インテル製品の概要説明を目的としたものです。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を許諾するものではありません。製品に付属の売買契約書『Intel's Terms and Conditions of Sale』に規定されている場合を除き、インテルはいかなる責任を負うものではなく、またインテル製品の販売や使用に関する明示または黙示の保証 (特定目的への適合性、商品適格性、あらゆる特許権、著作権、その他知的財産権の非侵害性への保証を含む) に関してもいかなる責任も負いません。

インテルによる書面での合意がない限り、インテル製品は、その欠陥や故障によって人身事故が発生するようなアプリケーションでの使用を想定した設計は行われていません。

インテル製品は、予告なく仕様や説明が変更されることがあります。機能または命令の一覧で「留保」または「未定義」と記されているものがありますが、その「機能が存在しない」あるいは「性質が留保付である」という状態を設計の前提にしないでください。これらの項目は、インテルが将来のために留保しているものです。インテルが将来これらの項目を定義したことにより、衝突が生じたり互換性が失われたりしても、インテルは一切責任を負いません。この情報は予告なく変更されることがあります。この情報だけに基づいて設計を最終的なものとしないでください。

本書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。

最新の仕様をご希望の場合や製品をご注文の場合は、お近くのインテルの営業所または販売代理店にお問い合わせください。

本資料で紹介されている資料番号付きのドキュメントや、インテルのその他の資料を入手するには、1-800-548-4725 (アメリカ合衆国) までご連絡いただくか、インテルの Web サイトを参照してください。

1 インテル® ハイパースレッディング・テクノロジー (インテル® HT テクノロジー) には、インテル® HT テクノロジー対応のインテル® プロセッサー、チップセット、BIOS、およびオペレーティング・システム (OS) を搭載したコンピューター・システムが必要です。パフォーマンスは、ご使用のハードウェアおよびソフトウェアによって異なる場合があります。プロセッサーによってサポートされる HT テクノロジーの詳細を含む情報は、こちらをご覧ください。

インテル® バーチャリゼーション・テクノロジーには、インテル® プロセッサー、BIOS、仮想マシンモニター (VMM) が有効なコンピューター・システムが必要です。機能性、性能もしくはその他のバーチャライゼーション・テクノロジーの特長は、使用するハードウェアやソフトウェアの構成によって異なり、BIOS の更新が必要になる可能性があります。ご利用になる OS によっては、ソフトウェア・アプリケーションとの互換性がない場合があります。詳細については、アプリケーションの開発元にお問い合わせください。

インテル・アーキテクチャー上の 64 ビット・コンピューティングには、インテル® 64 アーキテクチャー向けのプロセッサー、チップセット、BIOS、オペレーティング・システム、デバイスドライバー、アプリケーションとコンピューター・システムが必要です。パフォーマンスは、使用するハードウェアおよびソフトウェアによって異なる場合があります。詳細については、システムベンダーにお問い合わせください。

インテル® ターボ・ブースト・テクノロジーを利用するには、同テクノロジーに対応したプロセッサーを搭載したシステムが必要です。インテル® ターボ・ブースト・テクノロジーのパフォーマンスは、ハードウェア、ソフトウェア、およびシステム全体の構成によって異なります。利用するシステムがインテル® ターボ・ブースト・テクノロジーをサポートしているかどうかは、PC の製造元にお問い合わせください。詳細は、http://www.intel.com/technology/turboboost を参照ください。

Intel、インテル、Intel ロゴ、Look Inside、Look Inside ロゴは、アメリカ合衆国および / またはその他の国における Intel Corporation の商標または登録商標です。

*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。

© 2014 Intel Corporation.

最適化に関する注意事項

インテル® コンパイラーでは、インテル® マイクロプロセッサーに限定されない最適化に関して、他社製マイクロプロセッサー用に同等の最適化を行えないことがあります。これには、インテル® ストリーミング SIMD 拡張命令 2 (インテル® SSE2)、インテル® ストリーミング SIMD 拡張命令 3 (インテル® SSE3)、ストリーミング SIMD 拡張命令 3 補足命令 (SSSE3) 命令セットに関連する最適化およびその他の最適化が含まれます。インテルは、他社製マイクロプロセッサーに関して、いかなる最適化の利用、機能、または効果も保証いたしません。本製品のマイクロプロセッサー依存の最適化は、インテル® マイクロプロセッサーでの使用を前提としています。インテル® マイクロアーキテクチャーに限定されない最適化のなかにも、インテル® マイクロプロセッサー用のものがあります。この注意事項で言及した命令セットの詳細については、該当する製品のユーザー・リファレンス・ガイドを参照してください。

改訂 #20110804

© 2017, Intel Corporation. 転載を禁じます。* 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。

intel®