

アプリケーション・パフォーマンス・スナップショット (APS) ユーザーガイド Linux* 版

この記事は、インテル® デベロッパー・ゾーンに掲載されている「[Application Performance Snapshot](#)」の日本語参考訳です。

アプリケーション・パフォーマンス・スナップショットを使用して、アプリケーションの利用可能なハードウェアの使用状況を素早く確認できます。

アプリケーション・パフォーマンス・スナップショットとは

アプリケーションの解析

収集データ量の制御

メトリック・リファレンス

詳細な MPI 解析

解析レポート

Function Summary for All Ranks (すべてのランクの関数サマリー)

MPI Time per Rank (ランクごとの MPI 時間)

Collective Operations Time per Rank (ランクごとの集合操作時間)

Message Sizes Summary for All Ranks (すべてのランクのメッセージ・サイズ・サマリー)

Data Transfers per Rank-to-Rank Communication (ランク間通信ごとのデータ転送量)

Data Transfers per Rank (ランクごとのデータ転送量)

Data Transfers per Function (関数ごとのデータ転送量)

Node-to-Node Data Transfers (ノード間のデータ転送量)

Communicator List (コミュニケーター・リスト)

特定のランクの詳細情報

フィルター機能

主要メトリックによるフィルター

行数によるフィルター

複数のフィルターの使用

MPI_Pcontrol による領域制御

インテル® Trace Collector 向け設定ファイルの作成

既知の問題と制限事項

著作権と商標について

はじめに

アプリケーション・パフォーマンス・スナップショットの使用を開始するには、「[アプリケーションの解析](#)」を参照してください。

メトリック・リファレンス

アプリケーション・パフォーマンス・スナップショットでサポートされるメトリックの詳細は、「[メトリック・リファレンス](#)」を参照してください。

詳細な MPI 解析

MPI 関数、メッセージ、通信の詳細な解析については、「[詳細な MPI 解析](#)」を参照してください。

- [アプリケーション・パフォーマンス・スナップショットとは](#)
- [アプリケーションの解析](#)
- [メトリック・リファレンス](#)
- [詳細な MPI 解析](#)
- [既知の問題と制限事項](#)
- [著作権と商標について](#)

関連項目

[アプリケーションの解析](#)
[メトリック・リファレンス](#)
[詳細な MPI 解析](#)

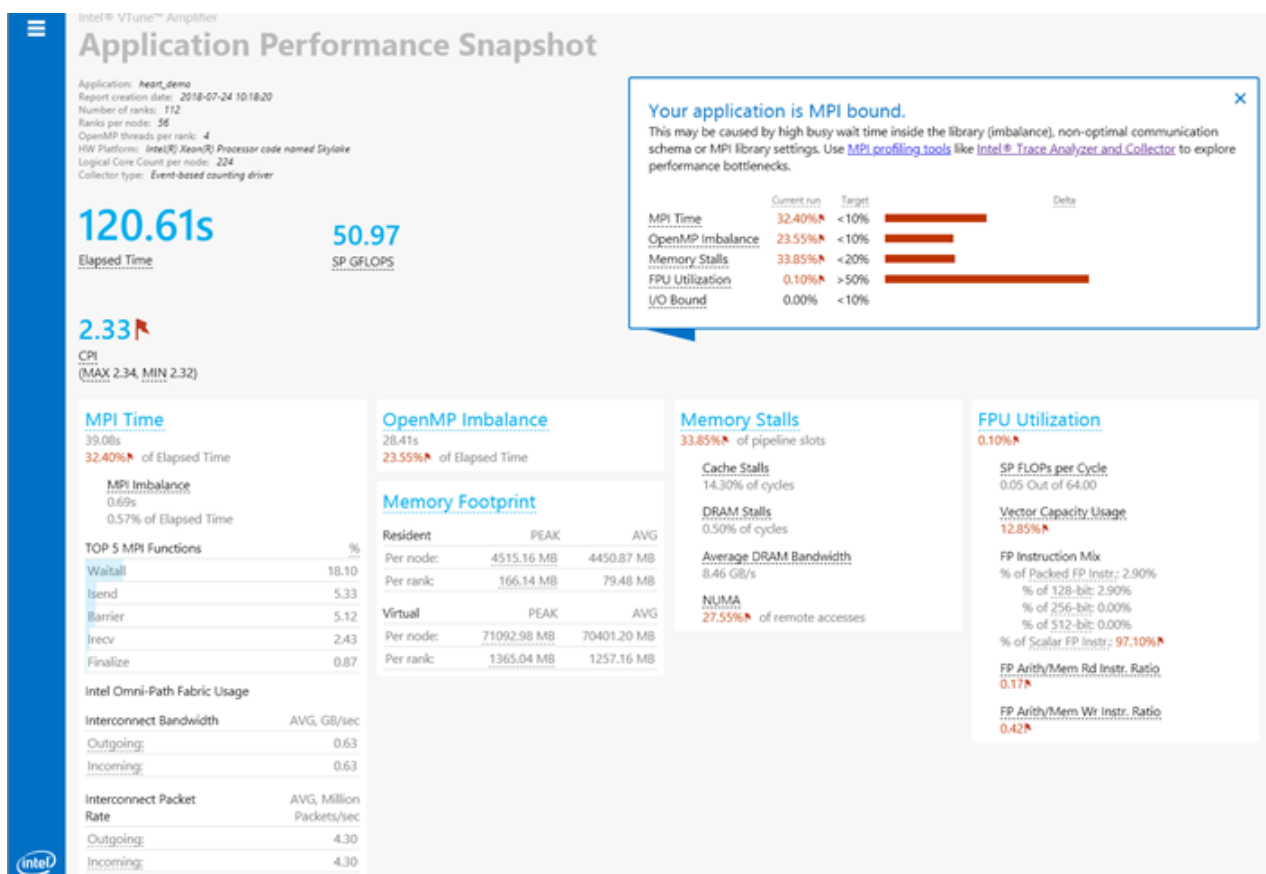
アプリケーション・パフォーマンス・スナップショットとは

利用可能なハードウェア (CPU、FPU、およびメモリー) を MPI アプリケーションがどのように活用しているか、アプリケーション・パフォーマンス・スナップショットを使用して簡単に参照できます。アプリケーション・パフォーマンス・スナップショットは、アプリケーションの CPU と FPU 使用、I/O とメモリー・フットプリント、メモリーアクセスのストール、および MPI と OpenMP* の利用率を解析します。解析後、インテル® プラットフォームを使用するシステム上でのパフォーマンス拡張の可能性が示されます。アプリケーション・パフォーマンス解析の最初のステップとしてこのツールを使用して、主な最適化分野の簡単なスナップショットを取得します。

アプリケーション・パフォーマンス・スナップショットは、インテル® デベロッパー・ゾーン (<https://software.intel.com/performance-snapshot> (英語)) から、無料でダウンロードできます。このツールは、インテル® Parallel Studio XE またはインテル® VTune™ Amplifier の一部としてインストールされます。

注

バージョン 2018 ベータ版以降では、Linux* 向けのアプリケーション・パフォーマンス・スナップショットに MPI パフォーマンス・スナップショットのほとんどの機能が含まれるようになり、MPI パフォーマンス・スナップショットは独立したツールとしては利用できなくなりました。



新機能

このユーザーガイドは、アプリケーション・パフォーマンス・スナップショット Linux* 版向けです。

以下は、これまでの製品リリースの変更履歴です。

Application Performance Snapshot 2019 Update 5

- Memory Stalls メトリックの DRAM Bandwidth 情報に Peak メトリックと Bound メトリックが追加されました。これらのメトリックは、メモリー帯域幅の使用に関する情報を提供します。これは、特にメモリー要件が異なる複数のフェーズで実行するアプリケーションにおいて役立ちます。

Application Performance Snapshot 2019 Update 4

- インテル® MPI ライブラリーが提供するコミュニケーターの内部 ID を収集する機能が追加されました。この機能は、Application Performance Snapshot 2019 Update4 以降およびインテル® MPI ライブラリー 2019 Update4 以降でサポートされます。

Application Performance Snapshot 2019 Update 3

- メッセージ量ごとの MPI アプリケーションの通信パターンをさらに適切に視覚化するため、HTML ベースのランク間通信の分布図を追加しました。

Application Performance Snapshot 2019 Update 2

- OpenMP* の機能を完全にサポートしました。
- ベクトル化効率のメトリックが改善されました。
- MPI インバランス時間は、収集のオーバーヘッドを最小化するためデフォルトの統計レベル 1 では計算されなくなりました。
- aps-report: 選択した MPI 関数セットの統計のみを表示するオプションが追加されました。
- MPI コレクターの一般的な最適化。

Application Performance Snapshot 2019 Update 1

- MPI インバランス収集が、純粋なアプリケーションのインバランスを測定可能なモードで拡張されました。このモードは、MPICH とバイナリー互換な MPI 実装に適用できます。必要に応じて、インバランス収集をオフにして収集のオーバーヘッドを最小化できます。
- MPI トレースにおけるオーバーヘッドの改善は、多数のランクがある状況では顕著に影響します。

Application Performance Snapshot 2019

- インテル® Omni-Path アーキテクチャー・インターコネクト帯域幅とパケットレートメトリックが追加され、MPI 通信のボトルネックを調査できるようになりました。
- MPI アプリケーションの通信パターンをさらに適切に視覚化するため、HTML ベースのランク間通信の分布図が追加されました。

Application Performance Snapshot 2018 Update 3 および 2019 Beta Update

- `aps-report` ユーティリティーに `--format` オプションが追加され、レポートをテキスト (*.txt) やカンマ区切り (*.csv) 形式で生成できます。CSV 形式は、レポート処理の自動化や Microsoft* Excel* などの表計算プログラムへのエクスポートに有用です。
- 集約通信時間カラムによりランク間のデータ転送量レポートが強化されました。
- 圧縮とデフォルトで設定された最小統計レベルで MPI トレースファイルのサイズが小さくなりました。`aps-report` ユーティリティーで生成されたレポートには、最小の統計レベルでは適用できないものがあります。詳細については、「[収集されるデータ量を制御](#)」を参照してください。
- `aps-report` ユーティリティーでレポート生成時間が大幅に短縮されました。

Application Performance Snapshot 2018 Update 2

- [コミュニケーター・リスト](#)がグラフに追加されました。
- [ランクごとの集合操作時間](#)グラフに、`-c`、`-m` オプションが追加されました。
- データ収集制御が追加されました。「[収集されるデータ量を制御](#)」を参照してください。
- インテル® Trace Analyzer & Collector との統合が追加されました。「[インテル® Trace Collector 向けの設定ファイルの作成](#)」を参照してください。

Application Performance Snapshot 2018 Update 1

- `MPI_Pcontrol` 領域数の制限がなくなりました。

Application Performance Snapshot 2018

- このツールは、`aps.sh` ではなく `aps` として起動されるようになりました。
- 結果ディレクトリーが `stat_*` から `aps_result_*` に変更されました。

Application Performance Snapshot 2018 Beta

- 最初のリリースです。

アプリケーションの解析

必要条件

(オプション) アプリケーション・パフォーマンス・スナップショットの実行中に高度なメトリックを収集するには、次のソフトウェアを使用します。

- 推奨コンパイラ: インテル® C/C++ または Fortran コンパイラ (その他のコンパイラも使用できますが、OpenMP* インバランスに関する情報はインテルの OpenMP* ライブラリーでのみ利用できます。)

ツールを実行する前に、環境を設定する必要があります。ターミナルで次のコマンドを実行します。

```
<install-dir>/apsvars.sh
```

ここで、<install-dir>はアプリケーション・パフォーマンス・スナップショットがインストールされている場所です。以下に例を示します。

```
$ source /opt/intel/performance_snapshots/apsvars.sh
```

ヒント

--help オプションを指定して `apsvars.sh` を起動すると、利用可能な設定オプションのリストが表示されます。

共有メモリー・アプリケーションの解析

1. 次のコマンドを実行します。

```
$ aps <my app> [<app parameters>]
```

ここで、<my app> はアプリケーションのパス、<app parameters> はアプリケーション・パラメータです。

アプリケーション・パフォーマンス・スナップショットは、アプリケーションを起動してデータ収集を実行します。

2. 解析が完了すると、コマンドウィンドウにレポートが表示されます。サポートされるブラウザで同じ情報を含む HTML レポートを開くこともできます。HTML レポートのパスは、コマンドウィンドウに含まれています。以下に例を示します。

```
$ firefox ./aps_report_01012017_1234.html &
```

3. レポートに表示されているデータの解析 HTML レポートのメトリックにホバーすると詳細情報が表示されます。
4. 結果の解析に基づいて、適切な次のステップを決定します。一般的な次のステップには、アプリケーションのチューニング、またはインテル® VTune™ Amplifier やインテル® Advisor などのパフォーマンス解析ツールによるさらなる調査が含まれます。

MPI アプリケーションの分析

1. 次のコマンドを実行して、MPI アプリケーションに関するデータを収集します。

```
$ <mpi launcher> <mpi parameters> aps <my app> [<app parameters>]
```

ここで、

- <mpi launcher> は、mpirun、srun、または aprun などの MPI ジョブランチャーです。
- <mpi parameters> は、MPI ランチャーのパラメーターです。

注

aps は最後の <mpi launcher> パラメーターでなければなりません。

- <my app> はアプリケーションのパスです。
- <app parameters> はアプリケーション・パラメーターです。

アプリケーション・パフォーマンス・スナップショットは、アプリケーションを起動してデータ収集を実行します。解析が完了すると、aps_result_<date> ディレクトリーが生成されます。

2. 次のコマンドを実行して、解析を完了します。

```
$ aps --report=aps_result_<date>
```

解析が完了すると、コマンドウィンドウにレポートが表示されます。Web ブラウザーで同じ情報を含む HTML レポートを開くこともできます。

3. レポートに表示されているデータの解析 HTML レポートのメトリックにホバーすると詳細情報が表示されます。
4. 結果の解析に基づいて、適切な次のステップを決定します。

一般的な次のステップには、mpitune ユーティリティーによる通信のチューニング、またはインテル® Trace Analyzer & Collector やインテル® VTune™ Amplifier などのパフォーマンス解析ツールによるさらなる調査が含まれます。

アプリケーション・パフォーマンス・スナップショットは、高度な MPI 解析機能も提供します。詳細は、「[詳細な MPI 解析](#)」を参照してください。

関連項目

[詳細な MPI 解析](#)

収集データ量の制御

アプリケーション・パフォーマンス・スナップショット (APS) は、収集データ量を制御するいくつかの方法を提供します。これにより、プロファイル・オーバーヘッドを軽減して、アプリケーションの関連するセクションに注目できます。

収集制御 API

デフォルトでは、APS はアプリケーションの実行全体の統計を収集します。一部のケースでは、特定のアプリケーション・フェーズで収集を有効/無効にすることが重要です。例えば、最も時間を費やしているセクションに注目したり、初期化フェーズやファイナライズ・フェーズで収集を無効にします。APS は、ソースコードからデータ収集を制御可能な API を提供します。

MPI アプリケーションでは、`MPI_Pcontrol()` API を使用します。`MPI_Pcontrol(0)` 呼び出しでデータ収集を一時停止し、`MPI_Pcontrol(1)` 呼び出しで再開できます。詳細は、「[MPI_Pcontrol による領域制御](#)」を参照してください。

MPI 以外のアプリケーションでは、ITT API を利用できます。ITT API を使用する前に、システムを設定する必要があります。設定手順は、<https://software.intel.com/en-us/vtune-amplifier-help-configuring-your-build-system> (英語) を参照してください。APS がスタンドアロン・パッケージとしてインストールされている場合、`<install_dir>` は `<install_dir>/internal/` と同じです。システムを設定したら、ITT API を使用できます。`__itt_pause()` でデータ収集を一時停止し、`__itt_resume()` で再開できます。

デフォルトでは、アプリケーションの起動時にプロファイルが有効になります。プロファイルなしでアプリケーションを起動するには、`-start-paused` オプションを使用します。プロファイルは、`MPI_Pcontrol(1)` または `__itt_resume()` の最初の呼び出しで自動的に開始されます。これは、初期化フェーズをスキップする場合に便利です。

MPI インバランスの収集

デフォルトでは、APS は MPI インバランス (アイドル時間) を収集して報告します。`APS_IMBALANCE_TYPE` 環境変数は、インバランスの計算方法に対する追加の制御を提供します。デフォルトのレベルは、`MPS_STAT_LEVEL` 環境変数の設定に応じて変わります。レベルを変更するには、`APS_IMBALANCE_TYPE` 環境変数を更新します。以下に例を示します。

```
export APS_IMBALANCE_TYPE=2
```


値	説明
APS_IMBALANCE_TYPE=0	MPS_STAT_LEVEL=1 の場合のデフォルト値。 インバランスの計算を無効にします。インバランスの計算を無効にすると、APS のオーバーヘッドが軽減されますが、アプリケーション・パフォーマンス解析の重要な統計である MPI インバランスに関する情報が提供されません。
APS_IMBALANCE_TYPE=1	MPS_STAT_LEVEL=2 以上の場合のデフォルト値。 インバランス (アイドル時間) は、インテル® MPI によってレポートされます。
APS_IMBALANCE_TYPE=2	インバランスは、集合操作の前に MPI_Barrier を呼び出して、呼び出しの時間を測定して計算されます。これにより、アプリケーションのインバランスに関するデータを提供できます。例えば、一部のランクがほかのランクよりも速くワークを完了する場合、MPI 集合操作を開始する前にほかのランクを待機する必要があります。この待機時間は、MPI_Barrier 呼び出しを使用して計算できます。

タイプでデータをフィルター処理

APS は、MPI 統計、OpenMP* 統計、またはハードウェア・カウンター統計のタイプ別に統計収集をフィルター処理できます。デフォルトでは、すべてのタイプのデータが収集されます。

データ収集タイプを指定するには、`-c (--collection-mode)` オプションを使用します。引数には、統計収集を有効にするタイプに応じて、`mpi`、`omp`、または `hwc` のカンマ区切りのリストを指定します。すべてのタイプの統計収集を有効にするには `all` を指定します (デフォルト)。

例えば、MPI アプリケーションでハードウェア・カウンター統計を無効にするには、次のように指定します。

```
mpirun -n 2 aps -c mpi,omp ./myapp
```

MPI 詳細レベルの設定

MPI アプリケーション向けに、APS は複数のレベルの統計収集を提供しています。収集されるデータ量が異なる 5 つの詳細レベルがあります。デフォルトはレベル 1 です。レベルを変更するには、`MPS_STAT_LEVEL` 環境変数を使用します。以下に例を示します。

```
export MPS_STAT_LEVEL=2
```

次の表に利用可能な詳細レベルの要約を示します。

レベル	収集される情報
1 (デフォルト)	MPI 関数とその時間
2	MPI 関数と送信データ量
3	MPI 関数、コミュニケーター、およびメッセージサイズ
4	MPI 関数、コミュニケーター、通信方向、および各方向のトラフィックの集計
5	MPI 関数、コミュニケーター、メッセージサイズ、および通信方向

アプリケーションが多くのコミュニケーターを使用する場合、レベル 5 では情報量が多すぎる可能性があります。その場合は、詳細レベルを下げることを検討してください。また、詳細レベル 1 - 4 では、そのレベルで提供される情報に応じて、一部の図を利用できません。

注

MPS_STAT_LEVEL 値は、APS_IMBALANCE_TYPE 環境変数のデフォルト値に影響します。詳細は、「[MPI インバランスの収集](#)」を参照してください。

コミュニケーターの内部 ID の収集

バージョン 2019 Update4 以降の APS およびインテル® MPI ライブラリーでは、実行間で同じ数のノードとノードごとのプロセスを維持する場合、APS でコミュニケーターの内部 ID を収集できます。この場合、実行間で内部 ID は変わりません。この関数を有効にするには、次の操作を行います。

- APS_COLLECT_COMM_IDS 環境変数を 1 に設定します。

```
export APS_COLLECT_COMM_IDS=1
```

- MPS_STAT_LEVEL を 3 以上に設定します。これらのレベルでのみ APS はコミュニケーターに関する情報を収集します。

```
export MPS_STAT_LEVEL=3
```

メトリック・リファレンス

このセクションでは、アプリケーション・パフォーマンス・スナップショットでサポートされるメトリックとその説明の一覧を示します。統計ファイルでメトリックのデータが利用可能な場合、コマンドラインと HTML レポートの解析サマリーに表示されます。一部のメトリックはプラットフォーム固有であり、一部はアプリケーションが MPI または OpenMP* を使用する場合にのみ利用できます。

Elapsed Time (経過時間)

指定したアプリケーションの実行時間 (秒)。

SP GFLOPS

1 秒間に計算された単精度ギガ浮動小数点操作数。すべての倍精度操作は 2 つの単精度操作に変換されます。SP GFLOPS メトリックは、第 3 世代インテル® Core™ プロセッサ、第 5 世代インテル® プロセッサ、および第 5 世代インテル® プロセッサでのみ利用できます。

CPI (リタイアした命令ごとのサイクル数) レート

実行された命令にかかった時間をサイクル数で測定した値。CPI =1 は、ハイパフォーマンス・コンピューティング (HPC) アプリケーションでは許容値と見なされますが、アプリケーション・ドメインごとに期待値は異なります。CPI 値は、レイテンシーの長いメモリー操作、浮動小数点操作、SIMD 操作、分岐予測ミスによりリタイアしない命令、またはフロントエンドの命令スタベーションがあると、大きくなる傾向にあります。

CPU Utilization (CPU 利用率)

このメトリックは、アプリケーションの並列効率を評価するのに役立ちます。アプリケーションによるシステム内のすべての論理 CPU コアの利用率を予測します。100% の利用率は、アプリケーションの実行中ずっと、すべての論理 CPU コアがビジーに保たれていることを意味します。このメトリックは、アプリケーションの有効なワークと並列ランタイムで費やされた時間を区別しないことに注意してください。

Serial Time (シリアル時間)

収集中にマスタースレッドの OpenMP* 領域外でアプリケーションが費やした時間。これは、アプリケーションの収集時間とスケーリングに直接影響します。値が大きい場合、コードの並列化やアルゴリズムのチューニングによって解決すべきパフォーマンスの問題を示している可能性があります。

MPI Time (MPI 時間)

MPI ライブラリー内で費やされた時間。15% を超える値については、MPI 通信効率を詳しく調査する必要があります。これは、ライブラリー内の長い待機時間、アクティブな通信、MPI ライブラリーの最適でない設定が原因の可能性があります。アプリケーションにロードバランスの問題があるかどうかを確認するには、MPI インバランス・メトリックを参照します。

MPI Imbalance (MPI インバランス)

プロセスがデータを待機中に、MPI ライブラリー呼び出しで費やされたプロセスごとの非生産的な待機時間の平均。

OpenMP* Imbalance (OpenMP* インバランス)

このメトリックは、ロードインバランスのためにアプリケーションが OpenMP* 同期バリアで浪費した経過時間の割合を示します。

Memory Stalls (メモリーストール)

このメトリックは、メモリー・サブシステムがパフォーマンスに与える影響を示します。要求ロード/ストア命令によりパイプラインがストールする可能性のあるスロットの度合いを測定します。このメトリック値は、要求ロード/ストア命令により実行パイプライン・スロットの大部分がストールする可能性を示すことができます。アプリケーションがキャッシュまたは DRAM 依存かどうか、および NUMA 効率を定義するには、第 2 レベルのメトリックを確認します。

Cache Stalls (キャッシュストール)

このメトリックは、マシンが L1、L2、および L3 キャッシュでストールした回数を示します。キャッシュヒットは DRAM ヒットよりもはるかに高速に処理されますが、パフォーマンスを大幅に低下させる可能性があります。このメトリックには、共有データに対する一貫性のペナルティーも含まれます。

DRAM Stalls (DRAM ストール)

このメトリックは、要求ロード/ストアが原因で CPU がメインメモリー (DRAM) でストールした回数を示します。

DRAM Bandwidth (DRAM 帯域幅)

このセクションのメトリックは、経過時間中のシステムによる高い DRAM 帯域幅使用率の範囲を示します。以下が含まれます。

- Average Bandwidth (平均帯域幅) - 経過時間中にシステムによって使用された平均メモリー帯域幅。
- Peak (ピーク) - 経過時間中にシステムによって使用された最大メモリー帯域幅。
- Bound (依存) - 経過時間のうち、メモリー帯域幅の使用率がプラットフォームの理論上の最大メモリー帯域幅の 70% のしきい値を超えていた部分。

一部のアプリケーションは、メモリー帯域幅を不均一に使用するフェーズで実行されます。例えば、初期化フェーズがあるアプリケーションは、最初により多くのメモリー帯域幅を使用する可能性があります。これらのメトリックを使用して、アプリケーションが実行中にどのようにメモリーを使用しているか特定します。

NUMA: % of Remote Accesses (NUMA: リモートアクセスの %)

NUMA (Non-Uniform Memory Architecture) マシンでは、ラスト・レベル・キャッシュをミスしたメモリー要求は、ローカルまたはリモートの DRAM によって処理されます。リモート DRAM へのメモリー要求には、ローカル DRAM へのメモリー要求よりもはるかに長いレイテンシーが必要です。頻繁にアクセスされるデータは、できるだけローカルに保持することを推奨します。このメトリックは、リモートアクセスの割合を示します。この値は小さいほうが良いです。

FPU Utilization (FPU 利用率)

このメトリックは、プログラムが浮動小数点ユニット (FPU) をどの程度利用しているかを示します。100% の値は FPU が完全にロードされていることを意味します。完全にロードされている FPU は、アプリケーション実行のすべてのサイクルで、ハードウェアの最大処理能力と FMA (Fused Multiply-Add) 命令 を利用してベクトル命令をリタイアします。

SP FLOPs per Cycle (サイクルごとの SP FLOPs)

1 クロックティックあたりの単精度 (SP) 浮動小数点操作 (FLOPs) 数。このメトリックは、ベクトルコードの生成と実行の効率を示します。サイクルごとの最大 FLOPs 数は、ハードウェア・プラットフォームにより異なります。FMA 対応 FPU では、サイクルごとの最大 FLOPs 数は、最大ベクトル処理能力の 2 倍です。すべての倍精度操作は 2 つの単精度操作に変換されます。

Vector Capacity Usage (ベクトル能力の使用)

このメトリックは、アプリケーション・コードのベクトル化と浮動小数点計算の関係を表します。100% の値は、すべての浮動小数点命令がベクトル処理能力を最大限に使用してベクトル化されていることを示します。

FP Instruction Mix (FP 命令ミックス)

% of Packed FP Instr (パックド FP 命令の %)

このメトリックは、すべてのパックド浮動小数点命令の割合を示します。また、128 ビット、256 ビット、および 512 ビット命令の割合が個別のメトリックとして示されます。

% of Scalar FP Instr (スカラー FP 命令の %)

このメトリックは、すべてのスカラー浮動小数点命令の割合を示します。

FP Arith/Mem Rd Instr. Ratio (FP 計算/メモリー読み取り命令の比率)

このメトリックは、浮動小数点計算命令とメモリー読み取り命令の比率を表します。0.5 未満の値は、ベクトル命令のデータアクセスがアライメントされていないため、ベクトル命令の実行パフォーマンスが低下する可能性があることを示しています。

FP Arith/Mem Wr Instr. Ratio (FP 計算/メモリー書き込み命令の比率)

このメトリックは、浮動小数点計算命令とメモリー書き込み命令の比率を表します。0.5 未満の値は、ベクトル命令のデータアクセスがアライメントされていないため、ベクトル命令の実行パフォーマンスが低下する可能性があることを示しています。

Back-end Stalls (バックエンド・ストール)

(第 2 世代インテル® Xeon Phi™ プロセッサ (開発コード名 Knights Landing) で利用可能)

スーパースカラー・プロセッサは、概念的に、フロントエンド (命令セットのフェッチとデコード) とバックエンド (要求された計算の実行) に分割できます。各サイクルで、フロントエンドは最大 2 つの操作を生成し、パイプライン・スロットに配置して、バックエンドに移動します。有用なワークを含むリタイアしたパイプライン・スロットの実際の数がこの最大値に等しくなることはまれです。これは、バックエンドが特定の操作を受け入れる準備ができていないことが原因の可能性があります (「バックエンド依存」の実行)。バックエンド依存の実行は、長いレイテンシー操作や、単一の実行ポートに多くの操作が集中するなどの実行リソースの競合により発生します。

L2 Hit Bound (L2 ヒット依存)

L1 をミスして L2 にヒットしたデータフェッチに費やされた CPU サイクルの割合。このメトリックには、共有データに対する一貫性のペナルティが含まれます。アクセス競合やデータ共有が問題の可能性であると示された場合は、まずそれらに対処します。そうでない場合は、L2 をミスするワークロードに適用するのと同じパフォーマンス・チューニングを検討します。

L2 Miss Bound (L2 ミス依存)

L2 ロードミスが処理されるのを待機するのに費やされた CPU サイクルの割合。可能な最適化として、ワーキングセットのデータサイズの縮小、データアクセスの局所性の改善、L2 に収まるチャンクでのデータのブロッキングと使用、ハードウェア・プリフェッチャーの活用が挙げられます。

SIMD Instructions per Cycle (サイクルごとの SIMD 命令)

(第 2 世代インテル® Xeon Phi™ プロセッサ (開発コード名 Knights Landing) で利用可能)

このメトリックは、プログラムが FPU をどの程度利用しているかを示します。FMA (Fused Multiply Add) は 2 命令としてカウントされます。

SIMD Instruction Mix (SIMD 命令ミックス)

% of Packed SIMD Instr (パッキング SIMD 命令の %)

このメトリックは、すべてのパッキング浮動小数点命令の割合を示します。

% of Scalar SIMD Instr (スカラー SIMD 命令の %)

このメトリックは、スカラー浮動小数点命令の割合を示します。

インテル® Omni-Path ファブリックのインターコネクト帯域幅とパケットレート

(インテル® Omni-Path ファブリック (インテル® OP ファブリック) が搭載され、インテル® VTune™ Amplifier ドライバーがインストールされた計算ノードで利用可能)

計算ノードごとの平均インターコネクト帯域幅とパケットレート (送信値と受信値)。値がインターコネクトの上限に近くなると、ネットワーク通信の待ち時間が長くなります。インターコネクト・メトリックは、[インテル® VTune™ Amplifier](#) のドライバーがインストールされている場合 (英語)、インテル® Omni-Path ファブリックで利用可能です。

詳細な MPI 解析

MPI アプリケーションのパフォーマンス概要では十分ではない場合、アプリケーション・パフォーマンス・スナップショットの高度な機能を使用して、アプリケーション・フローの理解を深めることができます。

MPI ランク、使用された関数、メッセージサイズ、その他のエンティティごとに解析結果を分類して、フィルター処理により結果の可読性を向上できます。特定のコードブロックを解析する `MPI_Pcontrol` 操作もサポートされています。

詳細は、次のセクションを参照してください。

- [解析レポート](#)
- [フィルター機能](#)
- [MPI_Pcontrol による領域制御](#)
- [インテル® Trace Collector 向け設定ファイルの作成](#)

関連項目

[解析レポート](#)

[フィルター機能](#)

[MPI_Pcontrol による領域制御](#)

[インテル® Trace Collector 向け設定ファイルの作成](#)

解析レポート

解析が終了するとデフォルトで表示されるサマリーページとは別に、アプリケーション・パフォーマンス・スナップショットはより詳細な MPI 解析機能を提供します。MPI アプリケーション実行に関する特定の情報を提供する各種レポートがあります。

各レポートには、`aps-report` ツールのコマンドライン・キーが割り当てられています。特定のレポートを表示するには、次のように対応するキーを指定して `aps-report` を実行します。

```
$ aps-report ./aps_result_<postfix> -f # prints the function summary
```

ほとんどのレポートでは、オプションを使用してフィルターを有効にできます。詳細は、「[フィルター機能](#)」を参照してください。利用可能なコマンドライン・キーとオプションの一覧は、`--help` オプションで確認できます。

利用可能なレポートの詳細な説明は、次のセクションを参照してください。

- [すべてのランクの関数サマリー](#)
- [ランクごとの MPI 時間](#)
- [ランクごとの集合操作時間](#)
- [すべてのランクのメッセージ・サイズ・サマリー](#)
- [ランク間通信ごとのデータ転送量](#)
- [ランクごとのデータ転送量](#)
- [関数ごとのデータ転送量](#)
- [ノード間のデータ転送量](#)
- [コミュニケーター・リスト](#)
- [特定のランクの詳細情報](#)

Function Summary for All Ranks (すべてのランクの関数サマリー)

キー: -f [--functions]

このレポートは、アプリケーションで使用されているすべての関数の情報を提供します。

例

```
| Function summary for all ranks
|-----
|
| Idle (%)      Function      Time(sec)      Time(%)      Idle(sec)
| Volume (MB)  Volume (%)     Calls
|-----
|
| 35.96        MPI_Barrier    92.40          36.20        91.79
|              0.00          0.00          7788
| 4.91         MPI_Waitall    91.70          35.93        12.53
|              0.00          0.00          11733612
| 0.00         MPI_Init       37.57          14.72        0.00
|              0.00          0.00          44
| 0.00         MPI_Isend      15.97          6.26         0.00
|              50469.13     97.36          18400506
| 0.00         MPI_Finalize   10.92          4.28         0.00
|              0.00          0.00          44
| 0.00         MPI_Irecv      6.57           2.58         0.00
|              0.00          0.00          18400506
| [filtered out 7 lines]
|=====
| TOTAL        255.24        100.00        104.39
| 40.90        51837.71     100.00        48543557
```

レポートの項目

列	
Function (関数)	関数名。
Time(sec) (時間 (秒))	すべてのランクでの関数実行の合計時間 (秒)。
Time(%) (時間 (%))	アプリケーションの実行全体に占める関数実行時間の割合。
Idle(sec) (アイドル (秒))	集合操作がアイドル状態であった時間 (秒)。
Idle(%) (アイドル (%))	集合操作がアイドル状態であった時間の割合 (%)。
Volume(MB) (容量 (MB))	すべてのランクで関数によって転送されたデータ量 (メガバイト)。
Volume(%) (容量 (%))	関数によって転送されたデータの割合。
Calls (呼び出し)	すべてのランクでのこの関数の呼び出し回数。
行	
TOTAL (合計)	それぞれの列の合計値。

MPI Time per Rank (ランクごとの MPI 時間)

キー: -t [--mpi_time_per_rank]

このレポートは、各ランクが MPI 操作に費やした時間を示します。

例

```
| MPI Time per Rank
|-----|
| Rank      LifeTime(sec)    MPITime(sec)    MPITime(%)
|-----|
| 0001      482964            396396          82.0756
| 0000      477183            387175          81.1378
|=====|
| TOTAL     960146            783571          81.6095
| AVG       480073            391785          81.6095
```

レポートの項目

列	
Rank (ランク)	ランク番号。
LifeTime(sec) (合計実行時間 (秒))	ランクの合計実行時間 (秒)。
MPITime(sec) (MPI 時間 (秒))	MPI 関数で費やした時間 (秒)。
MPITime(%) (MPI 時間 (%))	アプリケーションの合計実行時間に占める MPI 時間の割合。
行	
TOTAL (合計)	LifeTime(sec) 列の合計値、MPITime(sec) 列の合計値、および MPITime(%) 列の平均値。
AVG (平均)	それぞれの列の平均値。

Collective Operations Time per Rank (ランクごとの集合操作時間)

キー: -c [--collop_time_per_rank]

利用可能なオプション: -C [--communicators]、-M [--comm_id]、
-E [--internal_communicators]

このレポートは、各ランクが MPI 集合操作に費やした時間を示します。利用可能なオプションを使用してレポートを拡張できます。

例 1 (-c)

```
$ aps-report ./aps_result_<postfix> -c
```

```
| Collective Operations Time and Communicators for all ranks  
|-----  
| Rank      LifeTime(sec) CollOp Time(sec)  CollOp Time(%)  
|-----  
| 0003          3.48          2.32          66.73  
| 0002          3.48          2.21          63.38  
| 0001          3.48          2.08          59.72  
| 0000          4.53          2.52          55.67  
|-----  
| TOTAL          14.96          9.12          60.98  
| AVG            3.74          2.28          60.98  
|
```

例 2 (-c -C -M)

このレポートは、各ランクが特定のコミュニケーターで MPI 集合操作に費やした時間を示します。

```
$ aps-report ./aps_result_<postfix> -c -C -M 1
```

```
| Collective Operations Time and Communicators for all ranks
|-----
|-----
|
|      Function      Time(sec)      Time(%)      Volume(MB)
| Volume(%)      Calls      AvrMsgSize(B)
|-----
|
|      MPI_Allreduce      1.95      34.12      11087.89
| 100.00      130710      88949
|      MPI_Barrier      0.45      7.81      0.00
| 0.00      134990      0
| [filtered out 2 lines]
|=====
|-----
|-----
|
|      Function      Time(sec)      Time(%)      Volume(MB)
| Volume(%)      Calls      MsgSize(B)
|-----
|-----
|
|      MPI_Allreduce      1.95      34.12      11087.89
| 100.00      130710      ALL
|      Comm.: Id      Size
|                  1      4
|                  1.95      100.00      11087.89
| 100.00      130710      ALL
|-----
|-----
|
|      MPI_Barrier      0.45      7.81      0.00
| 100.00      134990      ALL
|      Comm.: Id      Size
|                  1      4
|                  0.45      100.00      0.00
| 100.00      134990      ALL
|-----
|-----
|
|      MPI_Gather      0.00      0.01      0.00
| 100.00      184      ALL
|      Comm.: Id      Size
|                  1      4
|                  0.00      100.00      0.00
| 100.00      184      ALL
|-----
|-----
|
|      MPI_Bcast      0.00      0.01      0.00
| 100.00      90      ALL
|      Comm.: Id      Size
|                  1      4
|                  0.00      100.00      0.00
| 100.00      90      ALL
|=====
|=====
```

例 3 (-c -C)

このレポートは、各ランクがすべてのコミュニケーターで MPI 集合操作に費やした時間を示します。

```
$ aps-report ./aps_result_<postfix> -c -C -E
```

Collective Operations Time and Communicators for all ranks

Volume (%)	Function	Calls	Time (sec) AvrMsgSize (B)	Time (%)	Volume (MB)
71.43	MPI_Allreduce	197658	3.44 73527	23.26	13859.89
0.00	MPI_Barrier	626018	2.64 0	17.87	0.00
14.29	MPI_Scatter	219396	2.31 13248	15.61	2771.99
14.29	MPI_Reduce	195384	1.64 14876	11.09	2771.97
[filtered out 2 lines]					
=====					
Volume (%)	Function	Calls	Time (sec) MsgSize (B)	Time (%)	Volume (MB)
100.00	MPI_Allreduce	197658	3.44 ALL	23.26	13859.89
80.00	Comm.: Id	130520	Size 4 1.62 ALL	47.12	11087.88
0.00		744	4 1.36 ALL	39.53	0.01
20.00		65260	2 0.46 ALL	13.23	2771.97
[filtered out 16 lines]					
100.00	MPI_Barrier	626018	2.64 ALL	17.87	0.00
100.00	Comm.: Id	744	Size 4 1.08 ALL	40.98	0.00
100.00		146848	2 0.59 ALL	22.47	0.00
100.00		73424	4 0.27 ALL	10.34	0.00
100.00		130792	2 0.25 ALL	9.38	0.00
100.00		65396	4 0.21 ALL	8.02	0.00
100.00		130792	4 0.18 ALL	6.69	0.00
100.00		65396	2 0.05 ALL	1.73	0.00
[filtered out 8 lines]					

レポートの項目

-c 列	
Rank (ランク)	ランク番号。
LifeTime(sec) (合計実行時間 (秒))	ランクの合計実行時間 (秒)。
CollOpTime(sec) (集合操作時間 (秒))	MPI 集合操作で費やした時間 (秒)。
CollOpTime(%) (集合操作時間 (%))	アプリケーションの合計実行時間に占める MPI 集合操作時間の割合。
-c -M および -c -C 列	
Function (関数)	関数名
Time(sec) (時間 (秒))	すべてのランクでの関数実行の合計時間 (秒)。
Time(%) (時間 (%))	アプリケーションの実行全体に占める関数実行時間の割合。
Idle(sec) (アイドル (秒))	集合操作がアイドル状態であった時間 (秒)。
Idle(%) (アイドル (%))	集合操作がアイドル状態であった時間の割合 (%)。
Volume(MB) (容量 (MB))	すべてのランクで関数によって転送されたデータ量 (メガバイト)。
Volume(%) (容量 (%))	関数によって転送されたデータの割合。
Calls (呼び出し)	すべてのランクでのこの関数の呼び出し回数。
AvrMsgSize(B) (平均メッセージサイズ (B))	メッセージの平均サイズ (バイト)。
行	
TOTAL (合計)	それぞれの列の合計値。
AVG (平均)	それぞれの列の平均値。

Message Sizes Summary for All Ranks (すべてのランクのメッセージ・サイズ・サマリー)

キー: -m [--message_sizes]

このレポートには、すべてのメッセージサイズのサマリーが含まれます。この情報は、特にインテル® MPI ライブラリーの内部しきい値のチューニングに役立ちます。

例

```
| Message Sizes summary for all ranks
|-----
| Message size (B)      Volume (MB)      Volume (%)      Transfers
| Time (sec)           Time (%)
|-----
11.36      0                0.00            0.00            1353910
           42.94
           4                0.80            0.00            209360
4.83      18.26
           4194304      8148.00         12.12           2037
1.59      6.01
           65536       7766.06         11.55           124257
1.14      4.29
           2097152      7954.00         11.83           3977
1.10      4.17
           131072       7772.12         11.56           62177
1.09      4.13
           262144       7784.25         11.58           31137
1.07      4.04
           1048576      7857.00         11.69           7857
1.06      4.01
           524288       7808.50         11.62           15617
1.05      3.95
           32768        6065.53         9.02            194097
0.68      2.58
           16384        3032.77         4.51            194097
0.40      1.51
| [filtered out 15 lines]
|=====
| TOTAL                67220.46        100.00           4394694
26.46      100.00
```

レポートの項目

列	
Message size (B) (メッセージサイズ (B))	メッセージサイズ (バイト)。
Volume (MB) (容量 (MB))	このサイズのメッセージのデータ転送量 (メガバイト)。
Volume (%) (容量 (%))	このサイズのメッセージのデータ転送量の割合。
Transfers (転送)	このサイズのメッセージの数。
Time (sec) (時間 (秒))	このサイズのメッセージの合計転送時間 (秒)。

行	
Time (%) (時間 (%))	このサイズのメッセージの転送時間の割合。
TOTAL (合計)	それぞれの列の合計値。

Data Transfers per Rank-to-Rank Communication (ランク間通信ごとのデータ転送量)

キー: -x [--transfers_per_communication]

このレポートは、各ランク間通信に関する情報を提供します。この情報は、ネットワーク使用のインバランスの検出に役立ちます。

例

```
| Data Transfers per Rank-to-Rank Communication
|-----|
| Rank --> Rank          Time (sec)      Time (%)      Volume (MB)
| Volume (%)      Transfers
|-----|
|-----|
| 0003 --> 0000          14.30          10.21          8835.73
| 7.33              370654
| 0002 --> 0003          14.15          10.10          8835.73
| 7.33              370622
| 0003 --> 0002          14.04          10.02          8142.73
| 6.75              336057
| 0003 --> 0001          14.02          10.01          7449.73
| 6.18              301492
| 0002 --> 0001          13.92           9.93          8142.73
| 6.75              336057
| 0002 --> 0000          13.89           9.91          7449.73
| 6.18              301524
| 0000 --> 0001          11.61           8.29          19577.18
| 16.24             847663
| 0001 --> 0000          11.39           8.13          17498.20
| 14.51             809130
| 0001 --> 0002           8.41           6.00          8835.72
| 7.33              370622
| 0001 --> 0003           8.14           5.81          7449.73
| 6.18              301492
| 0000 --> 0003           8.13           5.80          9528.71
| 7.90              337679
| 0000 --> 0002           8.10           5.78          8835.72
| 7.33              303114
|=====|
| TOTAL              140.11          100.00          120581.65
| 100.00             4986106
| AVG                11.68           8.33           10048.47
| 8.33              415508
```

レポートの項目

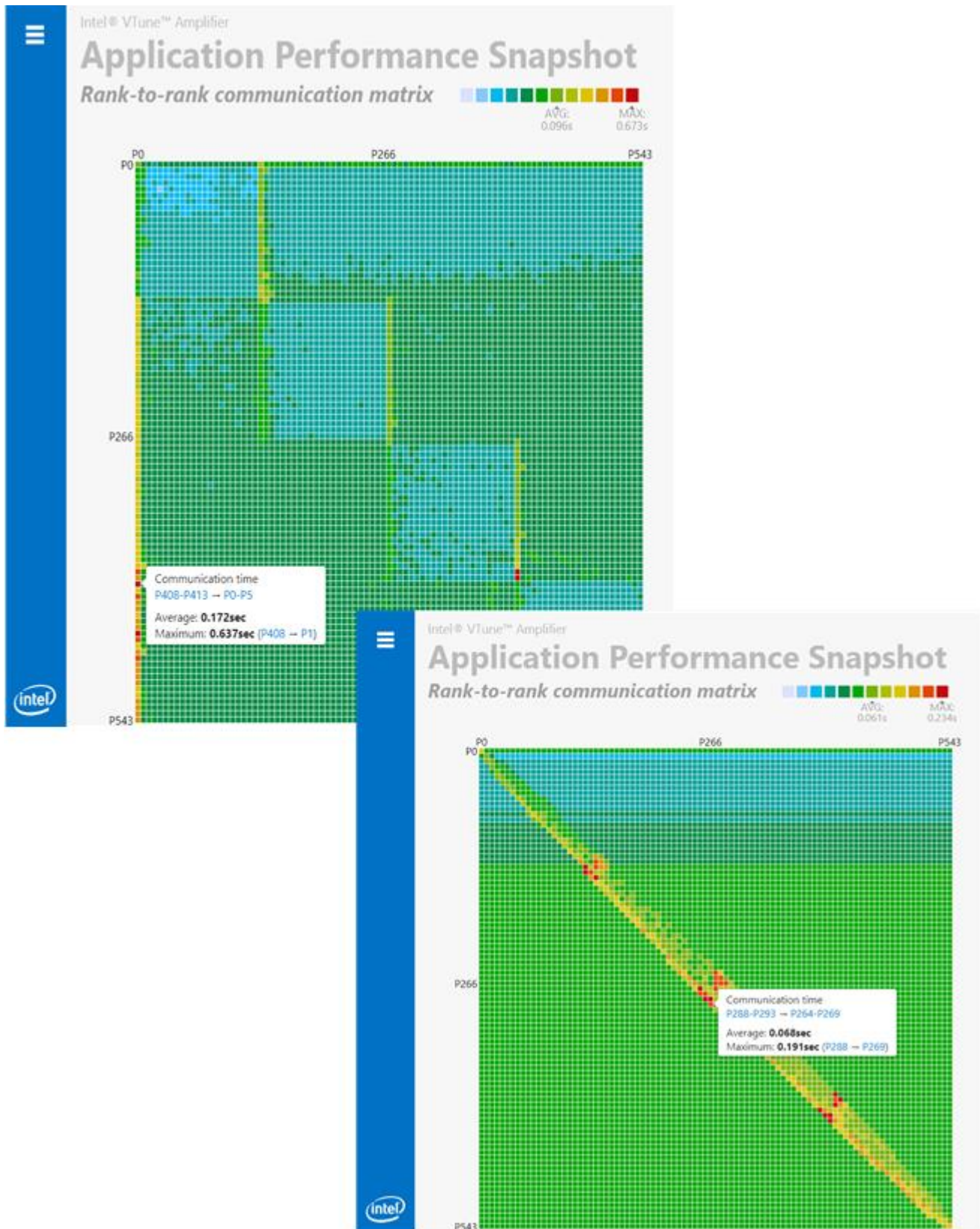
列	
Rank <-> Rank (ランク <-> ランク)	通信ランクの数。
Time (sec) (時間 (秒))	ランク間の通信時間。
Time (%) (時間 (%))	通信に費やした合計時間の割合。
Volume (MB) (容量 (MB))	通信で転送されたデータ量 (メガバイト)。
Volume (%) (容量 (%))	通信で転送されたデータ量の割合。
Transfers (転送)	通信内のメッセージの数。
行	
TOTAL (合計)	それぞれの列の合計値。
AVG (平均)	それぞれの列の平均値。値は次のように計算されます: $AVG = TOTAL / [(N*N + N) / 2]$ 。ここで、N はランクの数です。

グラフィック表現

ランク間通信のグラフィック表現を生成するには、`--format=html` を使用します。以下に例を示します。

```
aps-report -x --format=html <result name>
```

例



デフォルトでは、通信時間のダイアグラムが生成されます。通信容量のダイアグラムが必要な場合は、グラフィック・レポートの生成時に `-v` オプションを追加します。

Data Transfers per Rank (ランクごとのデータ転送量)

キー: `--r [--transfers_per_rank]`

このレポートは、各 MPI ランクのデータ転送量を示します。

例

```
| Data Transfers per Rank
|-----|
| Rank          Volume (MB)      Volume (%)    Transfers
|-----|
| 0001          22604.9         47.8261      1205911
| 0000          22604.9         47.8261      1206193
| 0002           12330          26.087       657791
| 0003           12330          26.087       657791
| 0004           6164.98         13.0435      328919
| 0005           6164.98         13.0435      328919
| 0006           6164.98         13.0435      328919
| 0007           6164.98         13.0435      328919
|=====|
| TOTAL          47264.9           100          2.52168e+06
| AVG            11816.2            25           630420
```

レポートの項目

列	
Rank (ランク)	ランク番号。
Volume (MB) (容量 (MB))	ランクが送信および受信したデータ量 (メガバイト)。
Volume (%) (容量 (%))	ランクが送信および受信したデータ量の割合。
Transfers (転送)	ランクが送信および受信したメッセージの数。
行	
TOTAL (合計)	Volume (MB) 列と Transfers 列はそれぞれ合計値の半分 (送信および受信された各バイトは 2 回カウントされているため)、Volume (%) 列の合計値。
AVG (平均)	それぞれの列の平均値。

Data Transfers per Function (関数ごとのデータ転送量)

キー: -u [--transfers_per_function]

このレポートは、各 MPI 関数 (すべてのランク) のデータ転送に関する情報を提供します。この情報は、データ転送に最も使用された関数を特定するのに役立ちます。

例

```
| Function summary for all ranks
|-----|
|          Function          Volume (MB)          Volume (%)          Transfers
|-----|-----|-----|-----|
|          Reduce           1385.98           66.6665           65128
|          Csend            692.997           33.3335           78073
|        Allreduce          0.000839233       4.03675e-05        132
|          Gather           0.000335693       1.6147e-05         44
|          Bcast            0.000312805       1.50461e-05        50
|          Send             3.8147e-06        1.83489e-07        1
|          Barrier          0                 0                 314
|=====|=====|=====|=====|
| TOTAL                    2078.98           100                143742
```

レポートの項目

列	
Function (関数)	関数名。
Volume (MB) (容量 (MB))	関数が送信および受信したデータ量 (メガバイト)。
Volume (%) (容量 (%))	関数が送信および受信したデータ量の割合。
Transfers (転送)	関数の呼び出し回数。
行	
TOTAL (合計)	それぞれの列の合計値。

Node-to-Node Data Transfers (ノード間のデータ転送量)

キー: -n [--node_to_node]

このレポートは、ノード間通信に関する情報を提供します。この情報は、ネットワーク使用のインバランスの検出に役立ちます。

例

```
| Volume summary for all nodes
|-----|
|           From                To                Volume (MB)
|-----|-----|
Node_name_1                Node_name_2                3
Node_name_2                Node_name_1                3
```

レポートの項目

列	
From (送信元)	送信ノード名。
To (送信先)	受信ノード名。
Volume (MB) (容量 (MB))	通信で転送されたデータの量 (メガバイト)。

Communicator List (コミュニケーター・リスト)

キー:

-l [--communicators_list]

-E [--internal_communicators]

これらのレポートは、アプリケーションで使用されているすべてのコミュニケーターの情報を提供します。レポートは、アプリケーション内の最も重要なコミュニケーターを特定できるように、時間でソートされています。

例 1: コミュニケーター・リスト

このレポートを生成するには、次のコマンドを実行します。

```
aps-report aps_result/--communicators_list
```

```
| Communicators used in the application
|-----
| Communicator Id  Communicator Size    Time(Rank Average) (sec)      Ranks
|-----
|           1           4           6.51 (1.63)           0,1,2,3
|-----
|           0           2           3.52 (1.76)           0,1
|-----
```

例 2: 内部 ID 付きのコミュニケーター・リスト

このレポートを生成するには、次のコマンドを実行します。

```
aps-report aps_result/--communicators_list --internal_communicators
```

```
| Communicators used in the application
|-----
| Communicator Id      Communicator Size    Time(Rank Average) (sec)      Ranks
|-----
| 4611686018431582208      4           2.44 (0.61)
0,1,2,3
|-----
| 4611686018431582656      4           1.80 (0.45)
0,1,2,3
|-----
| 4611686018429485456      2           1.77 (0.89)           0,1
|-----
| 4611686018431582608      4           1.40 (0.35)
0,1,2,3
```


4611686018429485552	2	1.24 (0.62)	0,1
4611686018431582704 0,1,2,3	4	0.86 (0.21)	
4611686018429485520	2	0.50 (0.25)	0,1
4611686018429485536	2	0.01 (0.00)	0,1
4611686018431582688 0,1,2,3	4	0.00 (0.00)	
4611686018429485472	2	0.00 (0.00)	0,1
4611686018429485488	2	0.00 (0.00)	0,1

レポートの項目

列	
Communicator List (コミュニケーター・リスト)	すべてのランクでの関数実行の合計時間 (秒)。
Communicator Size (コミュニケーター・サイズ)	アプリケーションの実行全体に占める関数実行時間の割合。
Time (Rank Average) (時間 (ランク平均))	コミュニケーターで MPI 集合操作で費やした時間とランク平均時間 (秒)。
Ranks (ランク)	ランク番号。

特定のランクの詳細情報

オプション: `-D` [`--details`]

このオプションを使用して、指定したレポートの詳細を表示します。次のような効果があります。

レポート名	説明
Message Sizes Summary (<code>-m -D</code>) (メッセージ・サイズ・サマリー)	関数別のメッセージサイズを表示します。
MPI Time per Rank (<code>-t -D</code>) (ランクごとの MPI 時間)	最小、平均、および最大時間値のプロセスの詳細を表示します。
Collective Operations Time per Rank (<code>-c -D</code>) (ランクごとの集合操作時間)	すべてのランクについて関数とメッセージサイズ別に情報を表示します。

詳細を表示するランクを指定するには、`-D` オプションの代わりに `-R` オプションを使用します。以下に例を示します。

```
$ aps-report ./aps_result_<postfix> -t -R 1
```

Time レポートでは、詳細にプロセスの実行時間とプロセスで使用された各関数の情報が含まれます。関数に関する情報は、次の方法で提供されます。

- 関数サマリー
- 各関数によって転送されたメッセージ詳細

これらの表では同じ列が使用されていますが、その意味はそれぞれの表で異なります。詳細は、以下の「レポートの項目」にある説明を参照してください。

例

```
#####
| RANK 3
| #####
| Life Time = 149.539 (sec)
| MPI Time = 140.662 (sec) 94.0639%
```

関数サマリー

```

|-----|
|-----|
| Volume (%)      Function      Calls      Time (sec)      Time (%)      Volume (MB)
|-----|-----|-----|-----|-----|-----|
| 31.2704         Alltoallv    109624     23.9206         15.9962         12330
|                    117939
| 5.21173         Alltoall     109624     23.8242         15.9317         2054.99
|                    19656
| 0               Barrier      11070      19.0241         12.7218         0
|                    0

```

5.21173	Allgather	17.8779	11.9553	2054.99
	109624	19656		
5.21173	Allgather	16	10.6996	2054.99
	109624	19656		
5.21172	Allreduce	5.72988	3.83169	2054.99
	100272	21490		
5.21173	Scatterv	5.53121	3.69883	2054.99
	109624	19656		
5.21173	Scatter	5.40639	3.61536	2054.99
	109624	19656		
5.21173	Gatherv	5.21395	3.48668	2054.99
	109624	19656		
5.21175	Gather	5.20341	3.47963	2055
	110665	19472		
0.977187	Reduce_scatter	3.92054	2.62174	385.307
	97624	4139		
15.6352	Send	3.33028	2.22703	6164.98
	328919	19654		
5.21173	Bcast	2.95886	1.97865	2055
	110288	19538		
5.21168	Reduce	2.72119	1.81972	2054.98
	97624	22072		
=====				
TOTAL				
100	1623830	140.662	94.0639	39430.2
		25462		

関数メッセージの詳細

Volume (%)	Function	Calls	Time (sec) MsgSize (B)	Time (%)	Volume (MB)
100	Alltoallv		23.9206	15.9962	12330
	109624		ALL		
11.6789		90	3.06902	2.05232	1440
			16777216		
11.6789		180	3.04682	2.03747	1440
			8388608		
11.6789		360	2.89341	1.93488	1440
			4194304		
11.6789		2880	2.83532	1.89604	1440
			524288		
11.6789		720	2.64094	1.76605	1440
			2097152		
11.6789		1440	2.53039	1.69212	1440
			1048576		
9.9757		4920	2.47286	1.65365	1230
			262144		
7.78591		30	2.04191	1.36546	960
			33554432		
[skipped 17 lines]					
=====					
100	Alltoall		23.8242	15.9317	2054.99
	109624		ALL		
11.6789		120	3.14809	2.1052	240
			2097152		
11.6789		60	2.96477	1.98261	240
			4194304		
11.6789		240	2.9347	1.9625	240
			1048576		
11.6789		3840	2.77462	1.85545	240
			65536		
11.6789		480	2.67651	1.78984	240
			524288		
11.6789		1920	2.52501	1.68853	240
			131072		

```

11.6789          960      2.50317          1.67392          240
                   262144
9.12411         6000     2.20381          1.47373          187.5
| [skipped 16 lines]
...

```

レポートの項目

関数サマリーの列	
Life Time (合計実行時間)	ランクの合計実行時間。
MPI Time / Coll Op Time (MPI 時間/集合操作時間)	MPI 呼び出しでランクが費やした時間と合計時間に占めるその割合。
Function (関数)	関数名。
Time(sec) (時間 (秒))	ランクでの関数実行の合計時間(秒)。
Time(%) (時間 (%))	アプリケーションの実行全体に占める関数実行時間の割合。
Volume(MB) (容量 (MB))	ランク内で関数によって転送されたデータ量(メガバイト)。
Volume(%) (容量 (%))	関数によって転送されたデータの割合。
Calls (呼び出し)	ランクでの関数の呼び出し回数。
AvrMsgSize(B) (平均メッセージサイズ (B))	ランク内で関数によって転送された平均メッセージサイズ(バイト)。
メッセージの詳細の列	
Function (関数)	関数名。
Time(sec) (時間 (秒))	このサイズのメッセージの転送時間(秒)。
Time(%) (時間 (%))	関数時間に占めるメッセージ転送時間の割合。
Volume(MB) (容量 (MB))	このサイズのメッセージのデータ転送量(メガバイト)。
Volume(%) (容量 (%))	関数のデータ転送に占めるメッセージのデータ転送の割合。
Calls (呼び出し)	このサイズのメッセージの転送数。
MsgSize(B) (メッセージサイズ (B))	関数によって転送されたメッセージのサイズ(バイト)。
行	
TOTAL (合計)	それぞれの列の合計値。

フィルター機能

アプリケーション・パフォーマンス・スナップショットは、数千の MPI ランクが並列に実行するアプリケーションを処理できます。そのようなアプリケーションでは、集計した結果レポートに数千行が含まれる場合があり、結果の読み取りが非常に困難です。

APS は、そのような結果の可読性を高めるフィルター機能を提供します

2 つの独立したフィルターがあり、すべてのレポートに適用できます。

- [主要メトリックによるフィルター](#)
- [行数によるフィルター](#)

注

フィルターを適用すると、スキップされた行数が表示されます。

フィルターはデフォルトで有効になり、次のキーの組み合わせを指定した場合と同じ動作になります。

```
-V 1 -T 1 -N 5
```

フィルターを完全に無効にするには、`-F` キーを使用するか、次のキーの組み合わせを使用します。

```
-V 0 -T 0 -N 0
```

詳細は、次のセクションを参照してください。

- [主要メトリックによるフィルター](#)
- [行数によるフィルター](#)
- [複数のフィルターの使用](#)

関連項目

[主要メトリックによるフィルター](#)

[行数によるフィルター](#)

[複数のフィルターの使用](#)

主要メトリックによるフィルター

すべてのレポートは、主要メトリックでソートされています。レポートの主要メトリックは次のとおりです。

- 時間 `--T (--time_threshold)` キーで制御
- 容量 `--V (--volume_threshold)` キーで制御

レポートが時間でソートされる場合、表のフィルター処理には時間のしきい値が適用されます。このレポートに容量しきい値は適用されません。逆も同様です。容量のしきい値は容量でソートされるレポートに適用され、時間でソートされるレポートには影響しません。

異なる主要メトリックでソートされる複数のレポートを表示する場合のみ両方のしきい値を使用できます。

どちらの主要メトリックもデフォルトのしきい値は 1% です。これは、次のコマンドライン・オプションの組み合わせと等価です: `-T 1 -V 1`。

ゼロ値 `-V 0` は容量によるフィルター処理を無効にし、`-T 0` は時間によるフィルター処理を無効にします。

しきい値の最大値は 100% です。

フィルターは、主要メトリックの割合がフィルターしきい値未満の行を非表示にします。例えば、容量でソートされる表では `Volume (%)` が 1% 未満の行は非表示になります。

注

デフォルトでは、行数によるフィルターが有効になります。このフィルターを無効にするには、`-N 0` キーを使用します。

例

ここで紹介するすべての例では、Message Size (メッセージサイズ) レポートを使用します。レポートは容量でソートされるため、フィルター処理には `-V` キーが使用されます。`-T` キーは効果がありません。

例 1

レポートのすべての行を表示するには、次のコマンドを実行します。

```
$ aps-report -V 0 -N 0 -m aps_result_<postfix>
```

出力:

Message Sizes summary for all ranks			

Message size (B)	Volume (MB)	Volume (%)	Transfers

262144	8340	11.6789	33360
524288	8340	11.6789	16680
1048576	8340	11.6789	8340
2097152	8280	11.5948	4140
4194304	8160	11.4268	2040
131072	8130	11.3848	65040

65536	7650	10.7126	122400
32768	5805	8.129	185760
16384	2966.25	4.15377	189840
8192	1500	2.10052	192000
8388608	1440	2.0165	180
16777216	960	1.34433	60
4096	750	1.05026	192000
2048	375	0.525129	192000
1024	187.5	0.262565	192000
512	93.75	0.131282	192000
256	46.875	0.0656411	192000
128	23.4375	0.0328206	192000
64	11.7188	0.0164103	192000
32	5.85938	0.00820514	192000
16	2.92975	0.00410266	192004
8	1.49738	0.00209684	196264
4	0.749222	0.00104917	196404
2	0.308998	0.000432704	162004
1	0.143055	0.000200326	150004
60	0.000114441	1.60257e-007	2
12	2.28882e-005	3.20513e-008	2
0	0	0	214062
=====			
TOTAL	71411	100	3466586

例 2

Message Size (メッセージサイズ) レポートにデフォルトの容量フィルターを適用するには、次のコマンドを実行します。

```
$ aps-report -N 0 -m aps_result_<postfix>
```

出力:

Message Sizes summary for all ranks			

Message size (B)	Volume (MB)	Volume (%)	Transfers

262144	8340	11.6789	33360
524288	8340	11.6789	16680
1048576	8340	11.6789	8340
2097152	8280	11.5948	4140
4194304	8160	11.4268	2040
131072	8130	11.3848	65040
65536	7650	10.7126	122400
32768	5805	8.129	185760
16384	2966.25	4.15377	189840
8192	1500	2.10052	192000
8388608	1440	2.0165	180
16777216	960	1.34433	60
4096	750	1.05026	192000
[skipped 15 lines]			

TOTAL	71411	100	3466586

この例では、Volume (%) 値が 1% 未満の行はすべてスキップされます。

例 3

Volume (%) 値が 10% 未満の行をすべてスキップするには、次のコマンドを実行します。

```
$ aps-report -V 10 -N 0 -m aps_result_<postfix>
```

出力:

```
| Message Sizes summary for all ranks
|-----|
| Message size (B)      Volume (MB)      Volume (%)      Transfers
|-----|
|      262144           8340           11.6789        33360
|      524288           8340           11.6789        16680
|     1048576           8340           11.6789         8340
|     2097152           8280           11.5948         4140
|     4194304           8160           11.4268         2040
|     131072           8130           11.3848        65040
|      65536            7650           10.7126       122400
| [skipped 21 lines]
|=====|
| TOTAL                71411           100            3466586
```

行数によるフィルター

このフィルターは、注目する行のみを表示できるようにします。通常、これらの行はレポートの上部、下部、中央にあります。フィルターキー `-N` は、それぞれの部分に表示される行数を定義します。これは表中の行に対するフィルターであるため、このフィルターは任意のメトリックでソートされたどの表にも適用できます。

デフォルト値は 5 であるため、表全体では 15 行 (上部 5 行、中央 5 行、下部 5 行) が表示されます。

ゼロ値 `-N 0` はフィルターを無効にします。フィルターの最大値は無制限です。

注

デフォルトでは、主要メトリックによるフィルターが有効になります。このフィルターを無効にするには、レポートのソートフィルターに応じて、`-T 0 -V 0` キーを一緒に使用するか、どちらか一方を使用します。

特定の行のみ表示する場合は、外部のテキスト・フィルター・ツールを使用してください。アプリケーション・パフォーマンス・スナップショットはそのような機能を提供していません。

例

ここで紹介するすべての例では、Message Size (メッセージサイズ) レポートを使用します。レポートは容量でソートされるため、フィルター処理には `-V` キーが使用されます。`-T` キーは効果がありません。

例 1

レポートのすべての行を表示するには、次のコマンドを実行します。

```
$ aps-report -V 0 -N 0 -m aps_result_<postfix>
```

出力は、「[主要メトリックによるフィルター](#)」の例 1 を参照してください。

例 2

表のサイズよりも大きな値を設定すると、すべての行が表示されます。例えば、表中のすべての行を表示するには、次のコマンドを実行します。

```
$ aps-report -V 0 -N 1000 -m aps_result_<postfix>
```

出力は、「[主要メトリックによるフィルター](#)」の例 1 を参照してください。

例 3

デフォルトのフィルター値 (5 行ずつ) を適用するには、次のコマンドを実行します。

```
$ aps-report -V 0 -m aps_result_<postfix>
```

出力:

```
| Message Sizes summary for all ranks
|-----|
| Message size (B)      Volume (MB)      Volume (%)      Transfers
|-----|-----|-----|-----|
|      262144           8340           11.6789        33360
|      524288           8340           11.6789        16680
|     1048576           8340           11.6789         8340
|     2097152           8280           11.5948         4140
|     4194304           8160           11.4268         2040
| [skipped 6 lines]
|     16777216           960            1.34433         60
|       4096             750            1.05026        192000
|       2048             375            0.525129       192000
|       1024            187.5          0.262565       192000
|        512            93.75          0.131282       192000
| [skipped 7 lines]
|          2            0.308998       0.000432704    162004
|          1            0.143055       0.000200326    150004
|         60            0.000114441    1.60257e-007     2
|         12            2.28882e-005    3.20513e-008     2
|          0             0              0              214062
|=====|
| TOTAL                71411          100            3466586
```

例 4

上部、中央、下部にそれぞれ 2 行ずつ (合計 6 行) 表示するには、次のコマンドを実行します。

```
$ aps-report -V 0 -N 2 -m aps_result_<postfix>
```

出力:

```
| Message Sizes summary for all ranks
|-----|
| Message size (B)      Volume (MB)      Volume (%)      Transfers
|-----|-----|-----|-----|
|      262144           8340           11.6789        33360
|      524288           8340           11.6789        16680
| [skipped 11 lines]
|       2048             375            0.525129       192000
|       1024            187.5          0.262565       192000
| [skipped 11 lines]
|          12            2.28882e-005    3.20513e-008     2
|          0             0              0              214062
|=====|
| TOTAL                71411          100            3466586
```

複数のフィルターの使用

デフォルトでは、前述の両方のフィルターが有効になります。時間と容量のデフォルトのしきい値は 1% で、行数フィルターのデフォルト値は 5 です。つまり、APS のデフォルトの動作は、`-V 1 -T 1 -N 5` キーを指定した場合と同じです。

注

主要メトリックによるフィルターが最初に適用され、残った行に行数によるフィルターが適用されます。

例

ここで紹介するすべての例では、28 行が含まれる Message Size (メッセージサイズ) レポートを使用します。

例 1

デフォルトのオプションを使用するには、次のコマンドを実行します。

```
$ aps-report -m aps_result_<postfix>
```

このコマンドラインは、次のコマンドラインと等価です。

```
$ aps-report -m -V 1 -T 1 -N 5 aps_result_<postfix>
```

出力:

```
| Message Sizes summary for all ranks
|-----|
| Message size (B)           Volume (MB)           Volume (%)           Transfers
|-----|-----|-----|-----|
|      262144                 8340                 11.6789             33360
|      524288                 8340                 11.6789             16680
|     1048576                 8340                 11.6789              8340
|     2097152                 8280                 11.5948              4140
|     4194304                 8160                 11.4268              2040
|     131072                 8130                 11.3848             65040
|      65536                 7650                 10.7126            122400
|      32768                  5805                  8.129              185760
|      16384                2966.25              4.15377            189840
|       8192                 1500                 2.10052            192000
|     8388608                 1440                 2.0165              180
|    16777216                 960                 1.34433              60
|       4096                 750                 1.05026            192000
| [skipped 15 lines]
|-----|-----|-----|-----|
| TOTAL                       71411                100                 3466586
```

容量フィルターは、下部にある Volume(%) が 1% 未満の行 (15 行) を非表示にします。残った 13 行は、表示する行数よりも少ないため、行数によるフィルターは適用されません。

例 2

容量が 1% 未満 (デフォルトのしきい値) の行を非表示にし、残った行の上部、中央、下部から 2 行ずつ表示するには、次のコマンドを実行します。

```
$ aps-report -N 2 -m aps_result_<postfix>
```

または

```
$ aps-report -V 1-N 2 -m aps_result_<postfix>
```

出力:

```
| Message Sizes summary for all ranks
|-----|
| Message size (B)           Volume (MB)           Volume (%)           Transfers
|-----|-----|-----|-----|
|           262144             8340                11.6789              33360
|           524288             8340                11.6789              16680
| [skipped 3 lines]
|           131072             8130                11.3848              65040
|           65536              7650                10.7126             122400
| [skipped 4 lines]
|           16777216           960                 1.34433              60
|           4096              750                 1.05026             192000
| [skipped 15 lines]
|=====|
| TOTAL                       71411                100                  3466586
```

MPI_Pcontrol による領域制御

アプリケーション・パフォーマンス・スナップショットは、MPI_Pcontrol() 操作による領域制御をサポートします。これにより、ソースコード内の特定のアプリケーション領域の統計収集を有効/無効にできます。領域制御は MPI メトリックと OpenMP* メトリックにのみ影響し、ほかのメトリックはアプリケーション全体で収集されます。

デフォルトでは、統計収集はアプリケーション全体に対して有効になります。特定の位置から収集を無効にするには、ソースコードに MPI_Pcontrol(0) 呼び出しを追加します。収集を再開するには、MPI_Pcontrol(1) 呼び出しを追加します。以下に例を示します。

```
...
MPI_Pcontrol(0);
MPI_Comm_size (MPI_COMM_WORLD, &size);
MPI_Comm_rank (MPI_COMM_WORLD, &rank);
MPI_Pcontrol(1);
...
```

MPI_Pcontrol() では、一意の番号 (5 以上) を使用して特定のコード領域をマークすることもできます。MPI_Pcontrol(<region>) 呼び出しで領域の開始をマークし、MPI_Pcontrol(-<region>) で領域の終了をマークできます。いくつかの非連続コードセクションを同じ領域番号でマークして、それらを同じ領域に含めることもできます。MPI_Pcontrol(0) と MPI_Pcontrol(1) は、すべての領域に対する統計収集をそれぞれ無効および有効にします。

次の例は、MPI_Pcontrol() でコード領域をマークする方法を示します。

```
if (rank == 0) {
    MPI_Pcontrol(5);
    for (i = 1; i < size; i++)
        MPI_Recv (&rank, 1, MPI_INT, i, 1, MPI_COMM_WORLD, &stat);
    MPI_Pcontrol(-5);
}
else {
    MPI_Pcontrol(6);
    MPI_Send (&rank, 1, MPI_INT, 0, 1, MPI_COMM_WORLD);
    MPI_Pcontrol(-6);
}
```

アプリケーションで領域がマークされている場合、アプリケーション全体と指定された領域に対して統計が収集されます。領域の統計データは、aps_result_<date>_<region> という名前の別のフォルダーに保存されます (例: aps_result_20171231_6)。

前述のとおり、0 から 4 は領域番号に使用できません (0 と 1 は特別な意味を持っており、2、3、および 4 は無視されます)。通常の領域は 5 以上の番号になります。この動作を変更し、1、2、3、および 4 を通常の領域にしたり、通常の領域の開始を 5 よりも大きな番号にすることができます。その場合、MPS_PCNTROL_REGION_BEGIN 環境変数を任意の値に設定します。

注

MPI 標準では、MPI_Pcontrol() は可変数の引数を持つことができます。アプリケーション・パフォーマンス・スナップショットでは、最初の引数のみが考慮されます。

インテル® Trace Collector 向け設定ファイルの作成

アプリケーション・パフォーマンス・スナップショット (APS) は、トレース・オーバーヘッドを軽減するため、インテル® Trace Collector 設定ファイル形式でインテル® Trace Analyzer & Collector との統合を提供します。

APS では、アプリケーションで最も使用される MPI 関数のみをトレースする設定ファイルを作成できます。時間、インバランス、容量でソートされた上位の関数を選択したり、選択する関数の数を変更することができます。

設定ファイルを作成するには、`-j (--itac_config)` オプションを指定して `aps-report` を実行します。オプションで `time`、`imbalance`、または `volume` 引数を使用して、それぞれの値でソートされた上位の関数を選択することもできます。選択される関数の数を制限するには、`-N` オプションを使用します。デフォルトでは、4 つの関数が選択されます。以下に例を示します。

```
$ aps-report ./aps_result_<postfix> -j time -N 5
```

上記のコマンドラインは、次のような設定ファイルを作成します。

```
ACTIVITY MPI OFF
SYMBOL MPI_Init ON
SYMBOL MPI_Recv ON
SYMBOL MPI_Send ON
SYMBOL MPI_Barrier ON
SYMBOL MPI_Bcast ON
ACTIVITY Application 0
```

既知の問題と制限事項

以下は、アプリケーション・パフォーマンス・スナップショットの既知の問題と制限事項のリストです。

- HTML サマリーレポートは、古いバージョンのブラウザでは正しく描画されない場合があります。正しく描画するには、ブラウザを最新バージョンにアップデートしてください。
- Windows* で共有ドライブから HTML レポートを表示する場合、Internet Explorer* 以外のブラウザを使用してください。
- サポートされていないコンパイラやコンパイラ・バージョンを使用すると、OpenMP* 関連のメトリックは利用できません。
- ディスク使用統計の収集に関する制限:
 - 統計を収集するため、アプリケーション・パフォーマンス・スナップショットは OS によって提供されるデータを使用します。I/O サブシステムからのデータの待機により発生する待機時間またはアイドル時間は、比較的低い精度で測定されます。そのため、実行時間の短いタスクでは、I/O サブシステムがアクティブに使用されていても、I/O 待機時間ゼロになることがあります。
 - プロセスの実行中にシステムがそのプロセスを別の CPU コアに再割り当てすると、データの読み取り/書き込みと I/O 待機時間はゼロになります。
- MPI サポートに関する制限:
 - MPI_THREAD_MULTIPLE モードはサポートされません。
 - アプリケーション・パフォーマンス・スナップショットは、インテル® Trace Collector と互換性がありません。両方のツールで同時に MPI アプリケーションを解析しないでください。
 - アプリケーション・パフォーマンス・スナップショットは、近隣の集合通信操作を完全にサポートしていません。実行時間のみ収集し、送信されたデータとその方向に関する情報は収集されません。
 - Disk Usage メトリックと Memory Footprint メトリックは、MPI_Pcontrol 領域統計では利用できません。

著作権と商標について

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。

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。

本資料には、開発中の製品、サービスおよびプロセスについての情報が含まれています。本資料に含まれる情報は予告なく変更されることがあります。最新の予測、スケジュール、仕様、ロードマップについては、インテルの担当者までお問い合わせください。

本資料で説明されている製品およびサービスには、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® テクノロジーの機能と利点はシステム構成によって異なり、対応するハードウェアやソフトウェア、またはサービスの有効化が必要となる場合があります。詳細については、OEM または販売店にお問い合わせいただくか、<http://www.intel.co.jp/> を参照してください。

Intel、インテル、Intel ロゴ、Intel Core、Xeon、Intel Xeon Phi、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 またはその子会社の商標です。

* 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。

© 2016-2019 Intel Corporation.

本ソフトウェアおよび関連ドキュメントはインテルの著作物であり、その使用には付随する明示的なライセンスが適用されます (「**ライセンス**」)。ライセンスに明記されている場合を除き、インテルから事前に書面による許可なしに、ソフトウェアまたは関連ドキュメントを使用、改変、複製、公開、配布、開示、転送してはなりません。

本ソフトウェアおよび関連ドキュメントは現状のまま提供され、ライセンスに明記されている場合を除き、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。

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。