

インテル® Visual Fortran コンパイラー 18.0 for Windows* リリースノート (インテル® Parallel Studio XE 2018)

このドキュメントでは、新機能、変更された機能、注意事項、および製品ドキュメントに記述されていない既知の問題について説明します。

パッケージに含まれるライセンスと本リリースノートの「著作権と商標について」をお読みください。本リリースのインテル® Visual Fortran コンパイラー 18.0 についての詳細は、次のリンクを参照してください。

- [動作環境](#)
- [使用方法](#)
- [ドキュメント](#)
- [インテルが提供するデバッグ・ソリューション](#)
- [日本語のサポート](#)
- [サンプル](#)
- [再配布可能なライブラリー](#)
- [テクニカルサポート](#)
- [互換性](#)
- [新規および変更されたコンパイラー機能](#)
- [新規および変更されたコンパイラー・オプション](#)
- [Visual Studio* 統合の変更点](#)
- [終了予定のサポート](#)
- [終了したサポート](#)
- [既知の問題](#)
- [Microsoft* Visual Studio* 2013/2015/2017 に関する注意事項](#)
- [Fortran 2008 および Fortran 2015 機能の概要](#)
- [著作権と商標について](#)

変更履歴

このセクションでは製品アップデートにおける重要な変更内容を説明します。

Update 1 (インテル® Fortran コンパイラー 18.0.1)

- 日本語版を含む最初のアップデート
- Microsoft* Visual Studio* 2017 統合の安定性が向上
- 報告された問題を修正

インテル® Visual Fortran コンパイラー 17.0 以降 (インテル® Visual Fortran コンパイラー 18.0.0 での変更)

- [新しいオプション /assume contiguous_pointer と /assume contiguous_assumed_shape](#)
- [新しいオプション /Qopt-zmm-usage](#)
- [/check contiguous による CONTIGUOUS ポインターへの連続していないポインター代入の診断](#)
- [Control-Flow Enforcement Technology \(CET\) サポート](#)
- [SVML を強制する新しいオプション /Qimf-use-svml](#)
- [SVML 呼び出しをコンパイル時に割り当て](#)
- [Microsoft* Visual Studio* 2015 Shell とライブラリーを追加](#)
- [Microsoft* Visual Studio* 2017 をサポート](#)
- [Microsoft* Windows Server* 2016 をサポート](#)
- [Microsoft* Visual Studio* 2012 のサポート終了](#)
- [すべての -o* オプションを -qo* オプションに変更](#)
- [/Qinit=\[keyword\] による初期化方法の拡張](#)
- [インストール・イメージからオフライン・ドキュメントを削除](#)
- [状況依存ヘルプ \(F1\) とヘルプメニュー項目のリンク先の変更](#)
- [インテル® Xeon Phi™ 製品ファミリーへのオフロードの新機能と変更された機能](#)
- [OpenMP* TR4 Version 5.0 Preview 1 の機能](#)
- [プロファイルに基づく最適化のハードウェア・ベースのイベント・サンプリング](#)
- [新規および変更されたコンパイラー・オプション](#)
- [Fortran 2008 の機能をサポート](#)
- [報告された問題を修正](#)

[先頭へ戻る](#)

製品の内容

インテル® Visual Fortran コンパイラー 18.0 for Windows* には、次のコンポーネントが含まれます。

- インテル® Visual Fortran コンパイラー 18.0。IA-32 およびインテル® 64 アーキテクチャー・システムで動作するアプリケーションをビルドします。
- Microsoft* Visual Studio* で Fortran アプリケーションをデバッグするための Fortran Expression Evaluator (FEE)
- Microsoft* 開発環境への統合
- Microsoft* Visual Studio* 2015 Shell とライブラリー (評価版ライセンスでは提供されません)
- サンプルプログラム
- オンラインおよびオンディスク・ドキュメント

[先頭へ戻る](#)

動作環境

アーキテクチャー名についての説明は、「[インテル® アーキテクチャー・プラットフォームの用語](#)」(英語)を参照してください。

- インテル® ストリーミング SIMD 拡張命令 2 (インテル® SSE2) 対応のインテル® 64 アーキテクチャー・ベースのプロセッサを搭載したコンピューター (第 2 世代以降のインテル® Core™ i3/i5/i7 プロセッサ、インテル® Xeon® プロセッサ E3/E5 ファミリー、または互換性のあるインテル以外のプロセッサ)
 - 機能を最大限に活用できるよう、マルチコアまたはマルチプロセッサ・システムの使用を推奨します。
- RAM 2GB (4GB 推奨)
- 4GB のディスク空き容量 (すべての機能およびすべてのアーキテクチャー)
- Microsoft® Windows® 7、Microsoft® Windows® 8、Microsoft® Windows® 8.1、Microsoft® Windows® 10、Microsoft® Windows Server® 2012 (R2)、Windows Server® 2016 (インテル® 64 のみ)
 - Microsoft® Windows® 8、Microsoft® Windows® 8.1 および Microsoft® Windows Server® 2012 では、製品は「デスクトップ」環境にインストールされません。「Windows® 8 UI」アプリケーションの開発はサポートされていません。
- IA-32 対応アプリケーション [\[4\]](#) またはインテル® 64 対応アプリケーションのビルドに、Microsoft® Visual Studio® 開発環境あるいはコマンドライン・ツールを使用する場合は、次のいずれか:
 - Microsoft® Visual Studio® 2017 Professional Edition 以上または Microsoft® Visual Studio® Community 2017 (「C++ によるデスクトップ開発」コンポーネントがインストールされていること) [\[7\]](#)
 - Microsoft® Visual Studio® 2015 Professional Edition 以上または Microsoft® Visual Studio® Community 2015 (「Visual C++ 2015 用の共通ツール」コンポーネントがインストールされていること) [\[6\]](#)
 - Microsoft® Visual Studio® 2013 Professional Edition 以上 または Microsoft® Visual Studio® Community 2013
 - Microsoft® Visual Studio® 2015 Shell (インテル® Visual Fortran コンパイラーの特定のライセンスに付属) ベースのインテル® Visual Fortran 開発環境 [\[1\]](#)[\[2\]](#)[\[3\]](#)
- IA-32 [\[5\]](#) アーキテクチャー・アプリケーションのビルドに、コマンドライン・ツールのみを使用する場合は、次のいずれか:
 - Microsoft® Visual C++® Express 2015 for Windows® Desktop
 - Microsoft® Visual C++® Express 2013 for Windows® Desktop
- インテル® 64 対応アプリケーションのビルドに、コマンドライン・ツールのみを使用する場合は、次のいずれか:
 - Microsoft® Visual C++® Express 2015 for Windows® Desktop
 - Microsoft® Visual C++® Express 2013 for Windows® Desktop
 - Microsoft® Windows® Software Development Kit for Windows® 8
- ドキュメントの参照用に Adobe® Reader® 7.0 以降

注:

1. Microsoft* Visual Studio* 2015 Shell ベースのインテル® Visual Fortran 開発環境は、インテル® Parallel Studio XE 2018 Composer Edition for Fortran Windows* のアカデミック・ライセンスと商用ライセンスに含まれています。評価版ライセンスや学生および教育関係者向けの無料ソフトウェア・プログラムで提供される「無料ツール」ライセンスには含まれません。この開発環境は、Fortran アプリケーションの編集、ビルド、デバッグに必要なものがすべて揃っています。ただし、次のような、Visual Studio* 製品の一部の機能は含まれていません。
 - リソースエディター (代用としてサードパーティー・ツールの ResEdit* (<http://www.resedit.net/> (英語)) を参照してください。)
 - Compaq* Visual Fortran プロジェクトの自動変換
 - Visual C++* や Visual Basic* などの Microsoft* の言語ツール
2. インテル® Visual Fortran コンパイラーを Microsoft* Visual Studio* 2015 Shell で使用するには、Microsoft* Windows® 10 ソフトウェア開発キット (SDK) をインストールする必要があります。[こちらの記事](#) (英語) の説明を参照してください。
3. Microsoft* Visual Studio* 2015 Shell ベースのシステムにインテル® Visual Fortran 開発環境をインストールすると、Microsoft* Visual Studio* Shell は実行に必要な Microsoft* コンポーネント (.NET Framework など) がシステムに含まれているかどうか確認し、不足しているコンポーネントを自動的にダウンロードしてインストールします。
4. インテル® Visual Fortran コンパイラーは、デフォルトで、インテル® SSE2 命令対応のプロセッサが必要な IA-32 アーキテクチャー向けのアプリケーションをビルドします。コンパイラー・オプションを使用して任意の IA-32 アーキテクチャー・ベースのプロセッサ上で動作するコードを生成できます。ただし、インテル® マス・カーネル・ライブラリー (インテル® MKL) を呼び出すアプリケーションでは、インテル® SSE2 命令に対応しているプロセッサが必要です。
5. アプリケーションは、上記の開発用と同じ Windows* バージョンで実行できます。また、Windows* 7 よりも前の非エンベデッドの Microsoft* Windows* 32 ビット・バージョンでも実行できますが、インテルではこれらの互換性テストを行っていません。開発アプリケーションは、古いバージョンの Windows* にはない Windows* API ルーチンを使用している可能性があります。アプリケーションの互換性テストをご自身の責任で行ってください。アプリケーションを実行するには、特定のランタイム DLL をターゲットシステムにコピーしなければならないことがあります。
6. インテル® Visual Fortran コンパイラーを Microsoft* Visual Studio* 2015 で使用するには、Visual Studio* から「Visual C++ 2015 用の共通ツール」コンポーネントをインストールする必要があります。[こちらの記事](#) (英語) の説明を参照してください。
7. インテル® Visual Fortran コンパイラーを Microsoft* Visual Studio* 2017 で使用するには、Visual Studio* から「C++ によるデスクトップ開発」コンポーネントをインストールする必要があります。[こちらの記事](#) (英語) の説明を参照してください。

[先頭へ戻る](#)

インテル® Visual Fortran コンパイラーの使用方法

コマンドラインおよび Microsoft* Visual Studio* からのインテル® Visual Fortran コンパイラーの使用方法は、「インテル® Parallel Studio XE 2018: インテル® C++ コンパイラー 18.0 for Windows* 入門」(<install-dir>/documentation_2018/ja/compiler_f/ps2018/getstart_wf.htm) を参照してください。

[先頭へ戻る](#)

ドキュメント

製品ドキュメントへのリンクは、<install-dir>/documentation_2018/ja/compiler_f/ps2018/get_started_lf.htm にあります。

インストール・イメージからオフライン・コア・ドキュメントを削除

インテル® Parallel Studio XE のインストール・イメージからオフライン・コア・ドキュメントが削除されました。インテル® Parallel Studio XE のコンポーネントのコア・ドキュメントは、[インテル® ソフトウェア・ドキュメント・ライブラリー](#) (英語) からオンラインで参照できます。また、[インテル® ソフトウェア開発製品レジストレーション・センター](#) から、日本語ドキュメントを含むオフライン・ドキュメントをダウンロードすることもできます: **Product List > Intel® Parallel Studio XE Documentation**。

インストールされるドキュメントの変更は、Microsoft* Visual Studio* 統合開発環境 (IDE) から利用できるドキュメントに影響します。詳細は、「[状況依存ヘルプ \(F1\) とヘルプメニュー項目のリンク先の変更](#)」を参照してください。

デベロッパー・ガイドおよびリファレンス、新機能とリリースノート、インストール・ガイド

すべてのツール・コンポーネントのデベロッパー・ガイドおよびリファレンス、新機能とリリースノート、インストール・ガイドは、[Intel® Parallel Studio XE Support > Documentation](#) (英語) から入手できます。

Windows* ベースのアプリケーションの作成についてのドキュメントは Web から入手可能

Windows* ベースのアプリケーションの作成についてのドキュメントは、インテル® ソフトウェア・ドキュメント・ライブラリーの Web サイトから入手できます。「[インテル® Visual Fortran を使用した Windows* ベースのアプリケーションの作成とビルド](#)」(英語) を参照してください。

Windows Server* 2012 の Microsoft* Internet Explorer* 10 でドキュメントが表示されない問題

Windows Server* 2012 では、Internet Explorer* 10 でヘルプまたはドキュメントを表示できない場合、Microsoft* Internet Explorer* のセキュリティ設定を変更すると表示されるようになります。[ツール] > [インターネット オプション] > [セキュリティ] を選択して、信頼済みサイトのリストに "about:internet" を追加します。必要に応じて、ドキュメントを参照した後に信頼済みサイトのリストから "about:internet" を削除します。

Microsoft* Windows® 10 日本語版の Microsoft Edge* でドキュメントが表示されない問題

Microsoft* Windows® 10 日本語版では、Microsoft Edge* でインテル® コンパイラー・ドキュメントを表示すると、左上の [目次]、[キーワード]、[検索] ボタンが動作しません。ボタンをクリックすると、空白ページが表示されます。

この問題は、将来のリリースで修正される予定です。回避策として、インテル® コンパイラー・ドキュメントの表示には Internet Explorer* を使用してください。デフォルトのブラウザを Internet Explorer* に設定するには、Google* で「Internet Explorer を通常使うブラウザにする」を検索してください。

複数のペインを含むドキュメントが Visual Studio* 内のブラウザで正しく表示されない

Visual Studio* 内のブラウザには複数のペインを含むドキュメントが正しく表示されない制限があります(左のペインに目次が表示されますが、右のペインにコンテンツが表示されません)。回避策: Visual Studio* の [ヘルプ] メニューから同じドキュメントにアクセスします。

[先頭へ戻る](#)

インテルが提供するデバッグ・ソリューション

インテルが提供するデバッグ・ソリューションは GNU* GDB ベースです。詳細は、「[インテル® Parallel Studio 2018 Composer Edition for Fortran - デバッグ・ソリューション・リリースノート](#)」(英語) を参照してください。

[先頭へ戻る](#)

日本語のサポート

日本語のサポートは、すべてのアップデートではなく、一部のアップデートで提供されます。

[先頭へ戻る](#)

サンプル

製品のサンプルは、「[インテル® ソフトウェア製品のサンプルとチュートリアル](#)」(英語) からダウンロードできます。

[先頭へ戻る](#)

再配布可能なライブラリー

詳細は、「[インテル® Parallel Studio XE の再配布ライブラリー](#)」(英語) を参照してください。

[先頭へ戻る](#)

テクニカルサポート

インストール時に製品の登録を行わなかった場合は、[インテル® ソフトウェア開発製品レジストレーション・センター](#)で登録してください。登録を行うことで、サポートサービス期間中 (通常は 1 年間)、製品アップデートと新しいバージョンの入手を含む無償テクニカルサポートが提供されます。

テクニカルサポート、製品のアップデート、ユーザーフォーラム、FAQ、ヒント、およびその他のサポート情報は、<http://www.intel.com/software/products/support/> (英語) を参照してください。

注: 販売代理店が製品のテクニカルサポートを提供している場合、インテルではなく販売代理店にお問い合わせください。

[先頭へ戻る](#)

互換性

一般に、インテル® Visual Fortran コンパイラーの以前のバージョン (8.0 以降) でコンパイルされたオブジェクト・コードおよびモジュールは、バージョン 18.0 でもそのまま使用できます。ただし、次の例外があります。

- バージョン 12.0 よりも前のコンパイラーでビルドされた CLASS キーワードを使用して多相変数を宣言しているソースは再コンパイルする必要があります。
- マルチファイルのプロシージャー間の最適化 (/Qipo) オプションを使用してビルドされたオブジェクトは再コンパイルする必要があります。
- バージョン 12.0 よりも前のコンパイラーでビルドされた REAL(16)、REAL*16、COMPLEX(16)、COMPLEX*32 データ型を使用しているオブジェクトは再コンパイルする必要があります。
- バージョン 10.0 よりも前のコンパイラーでインテル® 64 アーキテクチャー用にビルドされたモジュール変数を含むオブジェクトは再コンパイルする必要があります。Fortran 以外のソースからこれらの変数を参照する場合、不正な先頭の下線を削除するように外部名を変更する必要があります。

- バージョン 11.0 よりも前のコンパイラーでコンパイルされた、派生型宣言の外部で ATTRIBUTES ALIGN ディレクティブを指定したモジュールは再コンパイルする必要があります。この問題を検出すると、コンパイラーはメッセージを表示します。
- 派生型宣言の内部で ATTRIBUTES ALIGN ディレクティブを指定したモジュールは 13.0.1 以前のコンパイラーでは使用できません。
- Fortran 2008 のサブモジュール機能を実装するため、バイナリー .mod ファイルの内部フォーマットが大幅に変更されました。このため、バージョン 16.0 の Fortran コンパイラーで作成されたモジュールファイルは、バージョン 15.0 以前の Fortran コンパイラーで使用することはできません。
- インテル® Xeon Phi™ x100 製品ファミリー向けにコンパイル/ビルドされたオブジェクト/ライブラリーは、インテル® Xeon Phi™ x200 製品ファミリー向けにコンパイル/ビルドされたオブジェクト/ライブラリーと互換性がありません。

REAL(16) および COMPLEX(16) データ型のスタック・アライメントの変更

以前のリリースでは、REAL(16) または COMPLEX(16) (REAL*16 または COMPLEX*32) 項目が値で渡される場合、スタックアドレスは 4 バイトでアラインされます。パフォーマンスを向上するため、バージョン 12.0 以降のコンパイラーは、これらの項目を 16 バイトでアラインし、引数が 16 バイト境界でアラインされていると仮定します。

この変更は、主にコンパイラーにより生成される REAL(16) 値の計算を行うライブラリー・ルーチン (組込み関数を含む) の呼び出しに影響します。以前のバージョンでコンパイルしたコードをバージョン 13 のライブラリーとリンクする場合、またはアプリケーションをインテルのランタイム・ライブラリーの共有バージョンにリンクする場合、正しくない結果が返される可能性があります。

この問題を回避するには、REAL(16) および COMPLEX(16) データ型を使用しているすべての Fortran ソースを再コンパイルしてください。

インテルによる OpenMP* スタティック・ライブラリーの提供を終了

インテルによる OpenMP* スタティック・ライブラリー libiomp5mt.lib の提供が終了し、/Qopenmp-link:static コマンドライン・オプションがサポートされなくなりました。libiomp5mt.lib に対するすべての参照を、DLL インポート・ライブラリー libiomp5md.lib に変更してください。この変更に伴い、OpenMP* を使用するアプリケーションを、インテル® コンパイラーが存在しないシステムに配布する場合、インテル® コンパイラーの再配布可能コードのインストールが必要になります。詳細は、「[インテル® Parallel Studio XE の再配布ライブラリー](#)」(英語) を参照してください。

Fortran Expression Evaluator

Fortran Expression Evaluator (FEE) は、インテル® Visual Fortran コンパイラーとともにインストールされる Microsoft* Visual Studio* のプラグインです。Fortran コードを処理できるように、Microsoft* Visual Studio* IDE の標準デバッガーを拡張します。その他は標準デバッガーと同じです。

[先頭へ戻る](#)

新規および変更されたコンパイラー機能

インテル® Fortran コンパイラーは、Fortran 2008 標準のすべての機能をサポートします。インテル® Fortran コンパイラー 18.0 で追加された Fortran 2008 機能は、以下のとおりです。必要に応じて、[Fortran 2008 標準](#) (PDF、英語) および [Fortran 2015 標準の草案](#) (PDF、英語) を参照してください。

Fortran 2008 の機能

- ISO_FORTRAN_ENV の COMPILER_OPTIONS と COMPILER_VERSION
- 三角組込み関数と双曲線組込み関数への COMPLEX 引数
- FINDLOC 組込み関数
- MAXLOC 組込み関数および MINLOC 組込み関数の BACK オプション引数
- PROCEDURE リストにおける複数の型バインド・プロシージャ
- ポインター仮引数への非ポインターデータ項目の引き渡し
- 割り当て可能な左辺 (LHS) による多相代入
- 再帰型と前方参照の割付けコンポーネント
- DATA 文の制限の排除

インテル® Xeon Phi™ 製品ファミリーへのオフロードの新機能と変更された機能

インテルは、ユーザーの課題に最適なソリューションを提供するため、インテル製品の市場を継続的に評価しています。この評価プロセスの一環として、インテルはインテル® Xeon Phi™ コプロセッサ 7200 (開発コード名 Knights Landing) を市場に投入しないことを決定しました。

- インテル® Xeon Phi™ 7200 プロセッサの迅速な採用状況を考慮し、インテルは一般市場に Knights Landing (開発コード名) コプロセッサを展開しないことを決めました。
- インテル® Xeon Phi™ プロセッサは、引き続きインテルのソリューション・ポートフォリオの主要要素として、ユーザーに最も魅力的で競争力のあるソリューションを提供します。

インテル® Fortran コンパイラーは、Windows* 上ではインテル® Xeon Phi™ x200 製品ファミリーベースのシステムへのオフロードをサポートしません。通信にクラスター・ファブリックを利用するクラスター構成の Linux* 上ではサポートします。

インテル® Xeon Phi™ x100 製品ファミリー (開発コード名 Knights Corner) のサポートは、このリリースで終了しました。

プロファイルに基づく最適化のハードウェア・ベースのイベント・サンプリング

プロファイルに基づく最適化 (PGO) のハードウェア・ベースのイベント・サンプリングは、インテル® コンパイラーとインテル® VTune™ Amplifier を利用して PGO の多くの利点を得ることができる、新しい低オーバーヘッドのモデルです。インテル® VTune™ Amplifier がサポートさ

れているシステムでデータを収集することができます。詳細は、各製品のデベロッパー・ガイドを参照してください。

OpenMP* 機能

[OpenMP* Technical Report 4 : Version 5.0 Preview 1](#) (英語) 仕様のタスク・リダクション言語機能がサポートされました。

TASKGROUP に TASK_REDUCTION 節が追加されました。

TASK に IN_REDUCTION 節が追加されました。

TASKLOOP に REDUCTION 節と IN_REDUCTION 節が追加されました。

OpenMP* SIMD ディレクティブの新しいキーワード `monotonic`、`overlap`、および `lastprivate`:

```
!$omp ordered simd overlap(overlap_index)
```

```
!$omp ordered simd monotonic([var:step]s)
```

```
!$omp simd lastprivate(conditional:[vars])
```

詳細は、コンパイラー・ドキュメントまたは上記の OpenMP* 仕様へのリンクを参照してください。

新しいディレクティブと追加されたディレクティブ

[先頭へ戻る](#)

新規および変更されたコンパイラー・オプション

詳細は、コンパイラーのドキュメントを参照してください。

- [/Qinit:\[keyword\]](#)
- [-o* オプションを -qo* オプションに変更](#)
- [/Qprof-gen-sampling](#)、[/Qprof-use-sampling](#)
- [/Qoffload-arch:\[arch\]](#)
- [/assume:contiguous_pointer](#) オプションと [/assume:contiguous_assumed_shape](#) オプション
- [/check:contiguous](#)
- [/Qimf-use-svml](#)
- [/Qimf-force-dynamic-target](#)
- [/Qcf-potection](#)

廃止予定のコンパイラー・オプションのリストは、ドキュメントのコンパイラー・オプションのセクションを参照してください。

/Qinit=[keyword] オプションの拡張

スカラーと配列の初期化で、これまでの ZERO や SNAN に加えて、HUGE と MINUS_HUGE {表現可能な正/負の最大整数または実数値}、TINY と MINUS_Tiny {表現可能な正/負の最小整数または実数値}、INFINITY と MINUS_INFINITY {正/負の IEEE 無限大} もサポートされるようになりました。

すべての -o* オプションを -qo* オプションに変更

以前のバージョンで非推奨とされていたすべての -o* オプションが、このバージョンでは -qo* オプションに置き換えられました。ただし、Linux* と macOS* で出力ファイルを変更する -o オプションは変更されません。

Windows* では、/Qoffload-option コンパイラー・オプションでターゲット・コンパイラーに渡されるコンパイラー・オプションに影響します。

古い -o* オプションを使用すると、次のようなコンパイラー診断が出力されます。

```
$ ifort -openmp example.f90
ifort: コマンドライン・エラー: オプション '-openmp' はサポートされていません。オプション '-qopenmp' を使用してください。
```

影響のあるオプション:

- [no-]openmp
- openmp-lib=<arg>
- openmp-link=<arg>
- [no-]openmp-offload
- [no-]openmp-simd
- openmp-stubs
- openmp-threadprivate=<arg>
- openmp-report[=<level>]
- openmp-task=<arg>
- opt-args-in-regs=<arg>
- [no-]opt-assume-safe-padding
- opt-block-factor=<arg>
- [no-]opt-calloc
- [no-]opt-class-analysis
- [no-]opt-dynamic-align
- [no-]opt-gather-scatter-unroll
- [no-]opt-jump-tables=<arg>
- opt-malloc-options=<arg>
- [no-]opt-matmul
- [no-]opt-mem-layout-trans=<arg>
- [no-]opt-multi-version-aggressive
- [no-]opt-prefetch[=<val>]
- opt-prefetch-distance=<arg>

-opt-ra-region-strategy[=<arg>]
-[no-]opt-report-embed
-opt-report-file=<arg>
-opt-report-filter=<arg>
-opt-report-format=<arg>
-opt-report-phase=<arg>
-opt-report-routine=<arg>
-opt-report-help
-opt-report[=<arg>]
-opt-report-per-object
-opt-streaming-cache-evict=<arg>
-opt-streaming-stores=<arg>
-[no-]opt-subscript-in-range
-opt-threads-per-core=<arg>

/Qoffload-arch=[arch] のデフォルトを変更

このリリースでは、**/Qoffload-arch=[arch]** コンパイラー・オプションのデフォルトが、インテル® Xeon Phi™ x200 製品ファミリーをターゲットとする **mic-avx512** に変更されました。

新しいコンパイラー・オプション /Qprof-gen-sampling、/Qprof-use-sampling

これらの新しいプロファイルに基づく最適化オプションは、インテル® コンパイラーとインテル® VTune™ Amplifier を利用してハードウェア・ベースのイベント・サンプリングを行います。

新しいオプション /assume contiguous_pointer と /assume contiguous_assumed_shape

これらの新しいオプションは、すべての形状引継ぎ配列やポインターにユニットストライドがあることをアサートします。

新しいオプション /check:contiguous

このオプションは、CONTIGUOUS ポインターへの連続していないポインター代入の診断に役立ちます。/check:all オプションには、このチェックが含まれます。

SVML を強制する新しいオプション /Qimf-use-svml

新しいオプションは、スカラー数学演算に LIBM が使用されている場合、SVML の使用を強制します。これにより、ベクトルコードとスカラーコードの計算結果がビット単位で同じになることが保証されます。この機能を利用して、コンパイラーは /fp:precise FP モデルで数学関数をベクトル化し、ベクトル化されたコードはスカラーコードと同じ結果を生成します。

Control-Flow Enforcement Technology (CET) サポート

Control-flow Enforcement Technology (CET) は、ROP (Return-Oriented Programming) や COP/JOP (Call/Jump-Oriented Programming) などの脆弱性を悪用する特定の攻撃からプログラムを防御します。詳細は、[プレビュー・ドキュメント](#) (英語) を参照してください。

新しいコンパイラー・オプション /Qcf-potection[:keyword] により、コンパイラーの CET サポートを有効にできます。

SVML 呼び出しをコンパイル時に割り当て

SVML 関数に対するコンパイラーのデフォルトの動作が変更され、CPU 固有の SVML エントリーの呼び出しが実行されます。新しいオプション /Qimf-force-dynamic-target を指定すると、以前の動作に戻して動的に SVML をディスパッチできます。

新しいオプション /Qopt-zmm-usage

新しいオプション /Qopt-zmm-usage:low|high を使用して、コンパイラーにより生成される zmm コードをチューニングできます。引数値 low は、Skylake Server (開発コード名) マイクロアーキテクチャー・ベースのターゲット上のエンタープライズ・アプリケーションなどで、インテル® アドバンスト・ベクトル・エクステンション 2 (インテル® AVX2) ISA からインテル® アドバンスト・ベクトル・エクステンション 512 (インテル® AVX-512) ISA へのスムーズな移行を可能にします。ZMM 命令向けのチューニングには、#pragma omp simd simdlen() などの明示的なベクトル構文を使用することを推奨します。引数値 high は、幅広いベクトル操作を利用して命令ごとの計算量を高める、ベクトル演算が主体の HPC コードなどのアプリケーションに適しています。デフォルト値は、Skylake Server (開発コード名) マイクロアーキテクチャーをターゲットとする場合は low で、コンパイルターゲットを組み合わせる場合は high です。

[先頭へ戻る](#)

Visual Studio* 統合の変更点

Microsoft* Visual Studio* 2017 をサポート

インテル® Visual Fortran コンパイラーの統合サポートに Microsoft* Visual Studio* 2017 が追加されました。詳細は、「[動作環境](#)」を参照してください。

Microsoft* Visual Studio* 2015 Shell をサポート

インテル® Visual Fortran コンパイラーの統合サポートに Microsoft* Visual Studio* 2015 Shell が追加されました。詳細は、「[動作環境](#)」を参照してください。

状況依存ヘルプ (F1) とヘルプメニュー項目のリンク先の変更

F1 キーで表示される状況依存ヘルプは利用できません。また、新しい [ヘルプ] メニュー項目は、入門ドキュメントにジャンプします。オフライン・コア・ドキュメントのダウンロードに関する情報は、「[インストール・イメージからオフライン・コア・ドキュメントを削除](#)」を参照してください。

Microsoft* Visual Studio* のオンラインヘルプ形式の変更

オンラインヘルプ形式がブラウザーベースになりました。Microsoft* Visual Studio* の [ヘルプ] メニューからインテルのドキュメントを参照する場合、または F1 キー、ダイアログボックスにあるヘルプボタン、その他の GUI で状況依存ヘルプを参照する場合、デフォルトのブラウザーに対応するヘルプトピックが表示されます。デフォルトのブラウザーによっては、次のような既知の問題があります。

- [ヘルプ設定の設定] が [ブラウザーで起動] に設定されている場合、[ツール] > [オプション] > [F# ツール] または [ツール] > [オプション] > [IntelliTrace] で F1 キーを押すと、ブラウザーが 2 つ開きます。
- Chrome*: 検索またはキーワードからトピックを表示すると、目次が同期しません。[トピックを同期] も動作しません。
- Firefox*: 目次が表示されないことがあります。検索の大文字と小文字は区別されます。
- Safari*: Windows* の反応が遅くなります。

[先頭へ戻る](#)

終了予定のサポート

[先頭へ戻る](#)

終了したサポート

IA-32 ホスト・インストールのサポートを終了

IA-32 ホストへのインストールのサポートを終了しました。

Microsoft* Visual Studio* 2012 のサポートを終了

Microsoft* Visual Studio* 2012 のサポートを終了しました。

インテル® Xeon Phi™ x100 製品ファミリー (開発コード名 Knights Corner) のサポート終了

インテル® Xeon Phi™ x100 製品ファミリー (開発コード名 Knights Corner) は、2017 年 1 月に終息したため、インテル® Parallel Studio XE 2017 でのみサポートされます。インテル® Xeon Phi™ x100 製品ファミリーに対するインテル® Parallel Studio XE 2017 のサポートは、製品終息から 3 年後の 2020 年 1 月に終了します。有効なサポートサービスをお持ちの方にサポートが提供されます。

[先頭へ戻る](#)

既知の問題

日本語ファイル名に関するコマンドライン診断表示の問題

コンパイル診断で日本語が含まれているファイル名は、ネイティブのインテル® 64 対応アプリケーション用コンパイラーを使用して、Windows* コマンドラインでコンパイルした場合に正しく表示されません。Visual Studio* を使用する場合やインテル® 64 対応アプリケーション用クロスコンパイラーまたは IA-32 対応アプリケーション用コンパイラーを使用する場合は、この問題は発生しません。

Fortran を含む言語が混在したプログラムがデバッグできない

Visual Studio* 2012 以降で、.NET マネージ・コード・アプリケーションから呼び出される Fortran コードのデバッグを有効にするには、

[ツール] > [オプション] > [デバッグ] > [全般] で [Managed C++ 互換モード] チェックボックスをオフにします。

マネージ・コード・アプリケーションの場合は、プロジェクト・プロパティにある [デバッグ] > [アンマネージ コード デバッグを有効にする] も確認します。

PARAMETER 定数と /debug-parameters および /debug コンパイラー・オプションに関連した内部コンパイラー・エラー

PARAMETER 定数と /debug-parameters および /debug コンパイラー・オプションを一緒に使用すると、内部コンパイラー・エラーが発生することがあります。

以下に、内部エラーが発生する使用例を示します。

```
module sample_mod
  implicit none
  integer, parameter :: isize=32

  contains

  subroutine example1
    call example2(isize)
  end subroutine example1

  subroutine example2(jsize)
    integer, intent(in) :: jsize
    write(*,*) jsize
  end subroutine example2
end module sample_mod
```

Visual Studio* の Debug 構成のビルドでは、/debug-parameter オプションと /debug オプションが一緒に指定されることが多く、ビルド中にいくつかの内部エラーが発生することがあります。

この問題は、次のいずれか方法で回避できます。

1. Visual Studio* では、[プロパティ] > [Fortran] > [Debugging (デバッグ)] > [Information for PARAMETER Constants (PARAMETER 定数の情報)] を [None (なし)] に設定して、/debug-parameters を無効にします。
2. Visual Studio* では、任意の /debug-parameters 設定 (上記の 1 の手順を参照) と一緒に使用する場合、[プロパティ] > [Fortran] > [Debugging (デバッグ)] > [Debug Information Format (デバッグ情報の形式)] を [Line Numbers Only (行番号のみ)] に設定して、/debug:minimal 設定を使用します。
3. コンパイラー・コマンドラインでは、/debug:full を使用する場合、コマンドラインまたはアプリケーションのビルドスクリプトからすべての /debug-parameters を削除します。
4. コンパイラー・コマンドラインでは、任意の /debug-parameters 設定と一緒に指定する場合、コマンドラインまたはアプリケーションのビルドスクリプトで /debug:minimal を使用します。

警告 #31001: pdb にアクセスするための dll (mspdb110.dll など) が指定されたパスに存在しません。

Microsoft* Visual Studio* Express を使用して、IA-32 向けのコンパイルで /debug ([Fortran] > [General (全般)] > [Debug Information Format (デバッグ情報の形式)]) を有効にすると、次の警告が出力されることがあります。

警告 #31001: pdb にアクセスするための dll (mspdb110.dll など) が指定されたパスに存在しません。通常は設定エラーです。コンパイルは /Zi の代わりに /Z7 を使用して継続されますが、プログラムをリンクするときに同様のエラーが発生する可能性があります。

64 ビットの Windows* アプリケーション (つまり、インテル® Visual Fortran コンパイラー) は 64 ビットの dll を使用する必要がありますが ([https://msdn.microsoft.com/ja-jp/library/windows/desktop/aa384231\(v=vs.85\).aspx](https://msdn.microsoft.com/ja-jp/library/windows/desktop/aa384231(v=vs.85).aspx))、Microsoft* Visual Studio* Express は 32 ビット・バージョンの mspdb*.dll のみ提供しており、64 ビット・バージョンがないため、この警告が表示されます。この警告が表示されても、実行ファイルは正常にビルドされます。ただし、デバッガーで使用されるデバッグ情報はオブジェクト・ファイル (.obj) に埋め込まれます。/Z7 が有効な場合、コンパイラーは .pdb ファイルを生成しません。

OpenMP* 4.5 の OMP THREADPRIVATE と共通ブロック名の特定の使用方法が診断されない

OpenMP* 4.5 の規則では、共通ブロック名を指定する THREADPRIVATE ディレクティブが 1 つのプログラムユニットにある場合、同じ名前を指定する COMMON 文を含むすべてのプログラムユニットで、最後の該当する COMMON 文の後に THREADPRIVATE ディレクティブがなければなりません。インテル® Visual Fortran コンパイラーでは、この使用方法が適切に診断されません。

例えば、次のプログラムは OpenMP* 4.5 仕様に準拠していませんが、ifort は OMP THREADPRIVATE 文に続く 2 つの COMMON 文に対してエラーメッセージを出力しません。

```
PROGRAM ex1
  COMMON /common_blk1/x
  !$OMP THREADPRIVATE(/common_blk1/)

  COMMON /common_blk1/y
  COMMON /common_blk1/z

END PROGRAM
```

[先頭へ戻る](#)

Microsoft* Visual Studio* 2013/2015/2017 に関する注意事項

Microsoft* Visual Studio* 2010 ではいくつかの変更がありました。そのほとんどは、メインプログラムが C/C++ の言語が混在したアプリケーションのビルドに影響するものです。これらの変更は、Visual Studio* 2013/2015/2017 にも適用されます。

インテル® Fortran ランタイム・ライブラリーを参照するための Microsoft* Visual C++* の設定

以前のリリースでは、インテル® Visual Fortran コンパイラーの LIB フォルダを C/C++ プロジェクトで利用できるようにするために [ツール] > [オプション] > [プロジェクトおよびソリューション] > [Visual C++ ディレクトリ] で設定を行っていました。Visual Studio* 2010 以降では、この方法が変更されています。

1. Visual Studio* で C++ プロジェクトを含むソリューションを開き、[表示] > [プロパティ マネージャー] を選択します。[表示] メニューの直下に [プロパティ マネージャー] が見つからない場合は、[表示] > [その他のウィンドウ] の下にあります。[プロパティ マネージャー] ダイアログボックスが表示されます。これは、[プロパティ] ウィンドウや [プロパティ ページ] とは関係ありません。
2. プロパティ ツリーの Debug | Win32 の横にある三角または + 記号をクリックしてこのフォルダを展開します。
3. Microsoft.Cpp.Win32.user をダブルクリックします。
4. [VC++ ディレクトリ] を選択します。
5. [ライブラリ ディレクトリ] の右側のフィールドをクリックします。
6. ドロップダウンから <編集...> を選択します。
7. [新しい行] ボタンをクリックするか、Ctrl+Insert キーを押します。
8. 表示された新しいフィールドに、次のように入力します。

```
$(IFORT_COMPILER18)\compiler\lib\ia32
```

9. [OK] をクリックします。もう一度 [OK] をクリックして、[プロパティ ページ] も閉じます。
10. Visual Studio* のメニューから [ファイル] > [すべてを保存] を選択します。

インテル® 64 (x64) 構成でビルドする場合は、次の手順を実行してください。

1. [プロパティ マネージャー] を開いて、Debug | x64 フォルダを展開します。
2. Microsoft.Cpp.x64.user をダブルクリックします。
3. [VC++ ディレクトリ] を選択します。
4. [ライブラリ ディレクトリ] の右側のフィールドをクリックします。
5. ドロップダウンから <編集...> を選択します。
6. [新しい行] ボタンをクリックするか、Ctrl+Insert キーを押します。
7. 表示された新しいフィールドに、次のように入力します。

```
$(IFORT_COMPILER18)\compiler\lib\intel64
```

8. [OK] をクリックします。もう一度 [OK] をクリックして、[プロパティ ページ] も閉じます。
9. Visual Studio* のメニューから [ファイル] > [すべてを保存] を選択します。

[ソリューション エクスプローラー] タブをクリックするか、Ctrl+Alt+L キーを押して [ソリューション エクスプローラー] を表示します。

Debug | x64 フォルダに Microsoft.Cpp.x64.user プロパティ・ページが見つからない場合は、フォルダを右クリックして [新しいプロジェクト プロパティ シートの追加] を選択します。そして、MsBuild 4.0 プロパティ・ページの場所を参照します。Windows* XP では、通常以下の場所にあります。

```
C:\Documents and Settings\\Local Settings\Application Data\Microsoft\MSBuild\v4.0
```

Windows* 7 および Windows* 8 では、通常以下の場所にあります。

```
C:\Users\\AppData\Local\Microsoft\MSBuild\v4.0
```

これらのパスを表示するためには、隠しファイルと隠しフォルダの表示を有効にする必要があります。

Microsoft.Cpp.x64.user.props を選択して [開く] をクリックします。後は、上記の手順に従ってください。

プロジェクトの依存関係の調整

以前のバージョンの Visual Studio* から依存関係が設定されているプロジェクトを変換する場合、既存のプロジェクトの依存関係は Visual Studio* 2013/2015/2017 によって参照に変換されます。C/C++ プロジェクトで Fortran プロジェクトを参照している場合、C/C++ プロジェクトのビルドで MSB4075 エラーが発生することがあります。この問題を解決するには、次の操作を行います。

1. C/C++ プロジェクトを右クリックして、[参照] を選択します。
2. 参照リストに Fortran プロジェクトがある場合は、プロジェクトを選択してから [参照の削除] をクリックします。参照リストにあるすべての Fortran プロジェクトに対してこの操作を行います。[OK] をクリックします。
3. ほかの C/C++ プロジェクトでも上記の手順を実行します。

これにより、プロジェクトの依存関係が更新されます。

1. C/C++ プロジェクトを右クリックして、[プロジェクトの依存関係] を選択します。(Visual Studio* 2013 では、[ビルド依存関係] > [プロジェクト依存関係] を選択します。)
2. このプロジェクトと依存関係のあるプロジェクトのチェックボックスをすべてオンにします。
3. [OK] をクリックします。
4. 依存関係のあるほかの C/C++ プロジェクトでも上記の手順を実行します。

以前のバージョンの Visual Studio* とは異なり、Visual Studio* 2012 は依存関係のあるプロジェクトの出力ライブラリーを自動でリンクしません。そのため、親プロジェクトのプロパティ・ページで [Linker (リンカー)] > [Additional Directories (追加のライブラリー・ディレクトリー)] からこれらのライブラリーを明示的に追加する必要があります。必要に応じて、Visual Studio* のマクロである \$(ConfigurationName) と \$(PlatformName) を使用してパスを指定することができます。次に例を示します。

```
..\FLIB\$(ConfigurationName)\FLIB.lib
```

\$(ConfigurationName) は Release または Debug に置換されます。同様に、\$(PlatformName) は Win32 または x64 に置換されます。

Microsoft* Visual Studio* IDE からインテル® Parallel Studio XE のドキュメントを開くと「この種類のリンク (ファイル) を開くアプリがインストールされていません」エラーになる問題

.htm* ファイルの既定のアプリにブラウザーが設定されていない場合、Microsoft* Visual Studio* IDE で [ヘルプ] > [Intel Compilers and Libraries (インテル(R) コンパイラーおよびライブラリー)] からドキュメントを開くと、「この種類のリンク (ファイル) を開くアプリがインストールされていません」エラーが表示されます。

この問題を解決するには、[コントロール パネル] > [表示方法:] を [大きいアイコン] または [小さいアイコン] に設定し、[既定のプログラム] > [ファイルの種類またはプロトコルのプログラムへの関連付け] または [既定のプログラムの設定] で .htm ファイルを任意のブラウザーに関連付けます。または、[設定] > [システム] > [既定のアプリ] > [既定のアプリの選択] で .htm ファイルの既定のアプリとして任意のブラウザーを選択します。

インテル® C++ コンパイラーの Visual Studio* 2017 統合の問題

Microsoft* Visual Studio* 2017 との統合ではいくつかの問題が見つかっています。これらは間欠的な問題で、すべてのシステムで発生するわけではありません。最新の Visual Studio* 2017 バージョン **15.3.3** では、統合に関する問題は報告されていません。詳細は、<https://software.intel.com/en-us/articles/intel-software-development-tools-integration-to-vs2017-issue> (英語) を参照してください。

Microsoft* Visual Studio* 2012 で正しくないインテル® Fortran コンパイラー・バージョンが表示される

同じシステムにインテル® Parallel Studio XE 2017 とインテル® Parallel Studio XE 2018 がインストールされている場合、Microsoft* Visual Studio* 2012 で正しくないインテル® Fortran コンパイラー・バージョンが表示されます。詳細は、<https://software.intel.com/en-us/articles/wrong-intel-fortran-compiler-version-displayed-in-microsoft-visual-studio-2012> (英語) を参照してください。

32 ビットの Co-Array Fortran が 64 ビットの Microsoft* Windows* で動作しない

32 ビットの Co-Array Fortran は、必要なユーティリティー "mpiexec.exe" と "smpd.exe" が正しく動作しないため、64 ビットの Microsoft* Windows® 10 または Microsoft* Windows Server* 2012 R2 では動作しません。これは、互換性の問題です。詳細は、<https://software.intel.com/en-us/articles/coarray-fortran-doesnt-work-on-microsoft-windows-10> (英語) を参照してください。

[先頭へ戻る](#)

Fortran 2008 および Fortran 2015 機能の概要

インテル® Fortran コンパイラーは、Fortran 2008 標準のすべての機能をサポートします。また、Fortran 2015 標準の草案のいくつかの機能もサポートします。その他の機能は将来のリリースでサポートされる予定です。このバージョンでサポートされる Fortran 2015 標準の草案の機能は、次のとおりです。

- 「Technical Specification 29113 Further Interoperability with C」のすべての機能。次の機能を含みます。
 - 型引き継ぎ (TYPE(*))
 - ランク引き継ぎ (DIMENSION(..))
 - 互換性のある仮引数の制約の緩和
 - Fortran で使用される C コード操作「C 記述子」を定義する C インクルード・ファイル ISO_Fortran_binding.H

[先頭へ戻る](#)

著作権と商標について

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® ストリーミング SIMD 拡張命令 3、インテル® ストリーミング SIMD 拡張命令 3 補足命令などの最適化が該当します。インテルは、他社製マイクロプロセッサに関して、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依存の最適化は、インテル® マイクロプロセッサ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。

注意事項の改訂 #20110804

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も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、インテル製品の欠陥や故障によって人身事故が発生するような用途向けに使用することを前提としたものではありません。

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。この情報だけに基いて設計を最終的なものとししないでください。

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

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、<http://www.intel.com/design/literature.htm> (英語) を参照してください。

インテル・プロセッサ・ナンバーはパフォーマンスの指標ではありません。プロセッサ・ナンバーは同一プロセッサ・ファミリー内の製品の機能を区別します。異なるプロセッサ・ファミリー間の機能の区別には用いません。詳細については、http://www.intel.co.jp/jp/products/processor_number/ を参照してください。

インテル® Visual Fortran コンパイラーは、インテルのソフトウェア使用許諾契約書 (EULA) の下で提供されます。

詳細は、製品に含まれるライセンスを確認してください。

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 の商標です。

Microsoft および Windows は、米国 Microsoft Corporation の、米国およびその他の国における登録商標または商標です。

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

[先頭へ戻る](#)