

インテル® C++ コンパイラー・クラシック・リリースノート

本書は、英文「[Intel® C++ Compiler Classic Release Notes](#)」(英語)の日本語参考訳です。

2022年12月9日

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。

このリリースノートでは、インテル® C++ コンパイラー・クラシックのみ取り上げます。

インテル® C++ コンパイラー・クラシック 2021.8 (インテル® oneAPI 2023.0 リリース) の主な機能

- 問題の修正
- 機能やセキュリティ・アップデートなど、サードパーティー・コンポーネントの最新バージョンが含まれました。
- Xcode* 14.0.1 以降では、'extern template' とマークされたクラスと '_attribute_(internal_linkage)' とマークされたメンバー関数との相互作用により、リンク時にそのようなメンバー関数への未定義の参照エラーが表示されることがあります。エラーを回避するには、各クラス・テンプレートの明示的なインスタンスを追加します。以下に例を示します。

```
$ cat head.cpp
```

```
#include <sstream>
struct my_sb : public std::basic_stringbuf<char> {
};
#ifdef WORKAROUND
template class std::basic_stringbuf<char>;
#endif
int main() {
    my_sb sb;
}
```

```
$ icc head.cpp -wd10441
Undefined symbols for architecture x86_64:
```

```
"_ZNSt3115basic_stringbufIcNS_11char_traitsIcEENS_9allocatorIcEEE7seekp
osENS_4fposI11_mbstate_tEEj", referenced from:
    ___ZTV5my_sb in iccRZoeUh.o
```

```
ld
```

```
: symbol(s) not found for architecture x86_64
```

```
$ icc head.cpp -wd10441 -DWORKAROUND
$
```

- インテル® C++ コンパイラー・クラシック 2021.8.0 には、最新の機能とセキュリティーのアップデートが含まれていません。2023 年 3 月にリリース予定のインテル® C++ コンパイラー・クラシックには、追加の機能やセキュリティー・アップデートが含まれる予定です。最新バージョンが入手可能になり次第、アップデートしてください。

インテル® C++ コンパイラー・クラシック 2021.7.1 (インテル® oneAPI 2022.3.1 リリース) の主な機能

インテル® C++ コンパイラー・クラシックが 2021.7.1 に更新され、機能やセキュリティー・アップデートが含まれました。最新バージョンが入手可能になり次第、アップデートしてください。

インテル® C++ コンパイラー・クラシック 2021.7 (インテル® oneAPI 2022.3 リリース) の主な機能

- 問題の修正
- インテル® C++ コンパイラー・クラシックは古い機能 (非推奨) であるため、呼び出しごとに追加の診断メッセージが出力されます。この診断は、コンパイル中の出力に影響を与える可能性があります。例えば、コンパイラーを使用して前処理情報を生成すると (`icpc -E`)、追加の非推奨の診断が行われ、前処理済みの出力に影響します。
- この出力は、`-diag-disable=10441` (Linux*/macOS*) または `/Qdiag-disable:10441` (Windows*) オプションを指定して無効にできます。このオプションは、コマンドライン、設定ファイル、またはオプション設定環境変数に追加できます。
- インテル® C++ コンパイラー・クラシック (macOS* 版) は、インストールされている Xcode* に応じて、必要なヘッダー/ライブラリーを決定します。コンパイラーを実行し、'xcrun' 情報が初期化されていない場合、次のエラー (または同様のエラー) が発生します。

```
icc: error #10401: error running 'xcrun --sdk macosx --show-sdk-version'
```

コンパイラーを再実行すると、'xcrun' 情報がキャッシュされるため、このエラーは発生しなくなります。

インテル® C++ コンパイラー・クラシック 2021.6 (インテル® oneAPI 2022.2 リリース) の主な機能

- 問題の修正
- コンパイラーが Unicode の脆弱性についてソースコードのチェックを行うかどうか判断する、新しいコンパイラー・オプション `wcheck-unicode-security` がサポートされました。

Linux* の構文: `-W[no-]check-unicode-security`

macOS* の構文 (クラシック C/C++ のみ): `-W[no-]check-unicode-security`

Windows* の構文: `/W[no-]check-unicode-security`

インテル® C++ コンパイラー・クラシック 2021.5 (インテル® oneAPI 2022.1 リリース) の主な機能

- 問題の修正
- Xcode* 13.1 (macOS* 12.0.1) および Visual Studio* 2019 16.11.6 をサポート。Microsoft* Visual Studio* 2022 プレビューリリース 17.0 RC3 (プレビュー 7) を制限付きでサポート。Visual Studio* 2022 は、次のリリース (インテル® oneAPI 2022.2) でフルサポートされる予定です。
- /Qstd=c++20 (Windows*) または -std=c++20 (Linux*/macOS*) オプションで新しい C++20 の機能をサポート。詳細は、「[ドキュメント](#)」セクションを参照してください。

インテル® C++ コンパイラー・クラシック 2021.4 の主な機能

- 問題の修正
- Visual Studio* 2019 16.10.3 および Visual Studio* 2017 15.9.34 をサポート
- Xcode* 12.5 をサポート

インテル® C++ コンパイラー・クラシック 2021.3 の主な機能

- 問題の修正

インテル® C++ コンパイラー・クラシック 2021.2.1 の主な機能

- 問題の修正
- Microsoft* Visual Studio* 2019 16.9.3 をサポート。このリリースで VS 2019 v16.9.3 で発生していた数学組込み関数呼び出しのリンク問題が修正され、インテル® C++ コンパイラー・クラシックが VS 2019 16.9.3 で正しく動作するようになりました。VS 2019 16.9.3 を使用している C++ ユーザーは、このパッチをインストールしてリンク問題を修正できます。

重要:

- この 2021.2.1 はパッチリリースです。インテル® Fortran を使用している場合は、このパッチリリースをインストールしないでください。システムにインストールされている既存のインテル® Fortran コンパイラー・クラシックおよびベータ版インテル® Fortran コンパイラー 2021.x に問題が発生します。
- 2021.2.1 リリースは Windows* で C++ のみを使用しているユーザー向けに、インテル® C++ コンパイラー・クラシックの問題のみ修正します。DPC++ および ICX ユーザー向けの 2021.2.0 以降の新しい修正や機能は含まれていないため、DPC++ および ICX ユーザーはこのリリースをインストールする必要はありません。
- 2021.2.1 リリースは [オンライン](#) (英語) でのみ提供されます。このリリースは、インテル® ソフトウェア開発製品レジストレーション・センターでは提供されず、NuGET リポジトリから取得できません。

インテル® C++ コンパイラー・クラシック 2021.2 の主な機能

- 以下のコンパイラー・オプションにより、開発コード名 Alder Lake および開発コード名 Sapphire Rapids の機能をサポート。
 - -mavxvnni

- -mclldemote
 - -mhreset
 - -mptwrite
 - -mserialize
 - -mwaitpkg
 - -march=alderlake, -xalderlake
 - -march=sapphirerapids, -xsapphirerapids
- 報告された問題を修正

インテル® C++ コンパイラー・クラシック 2021.1.2 パッチリリースの新機能

- この 2021.1.2 はパッチリリースです。フルコンパイラーは含まれず、既存のコンパイラーの問題のみ修正します。既存のインテル® oneAPI HPC ツールキット 2021.1.1 の上にインストールします。
- このパッチリリースは、`--version` コンパイラー・オプションを使用すると正しくないバージョン文字列が表示される問題を修正します。初期リリース 2021.1.1 に含まれているコンパイラーのほかの修正は含まれていません。
- パッチリリースをインストールするとき、ユーザーは使用するすべてのコンパイラー (インテル® Fortran コンパイラー、インテル® oneAPI DPC++/C++ コンパイラー、インテル® C++ コンパイラー・クラシック) の最新のパッチをインストールします。
- OpenCL* アプリケーション向けインテル® CPU ランタイムも再インストールする必要があります。Windows* 向けの OpenCL* アプリケーション向けインテル® CPU ランタイムは、[こちら](#) (英語) からダウンロードできます。Linux* の場合、パッケージは APT および YUM リポジトリ経由で配布されます。[Linux* パッケージ・マネージャーを利用してインテル® oneAPI ツールキットをインストールする](#) (英語) の手順に従ってリポジトリをセットアップし、"intel-oneapi-runtime-ocl" パッケージをインストールしてください。

インテル® C++ コンパイラー・クラシック 2021.1 の主な機能

- oneAPI 環境モジュールスクリプトに 2 つの新しいスクリプト `icc` および `icc32` が含まれました。これらのスクリプトはコンパイラー・パッケージの一部としてインストールされ、インテル® C++ コンパイラー・クラシック (ICC) を使用するための環境を設定します。これらの新しい modulefile スクリプトは Tcl 環境モジュール実装のバージョン 3.x/4.x および modulefile の Lmod 実装と互換性があります。
- `/Qstd=c++20` (Windows*) または `-std=c++20` (Linux*/macOS*) オプションで新しい C++20 の機能をサポート。
- OpenMP* 5.0 の新機能を追加
 - SIMD ディレクティブの IF 節
 - SIMD ディレクティブの NONTEMPORAL 節
- 報告された問題を修正

インテル® C++ コンパイラー・クラシックのインストール方法

インストール手順は、すべてのインテル® ソフトウェア開発製品にドキュメントの一部として含まれています。最新バージョンのインテル® oneAPI HPC ツールキットのインストール・ガイドはオンラインでも提供されています。[インストール・ガイド](#) (英語) のインテル® oneAPI ページを確認してください。

インテル® C++ コンパイラー・クラシックの使用法

インテル® C++ コンパイラー・クラシックの使用法の詳細:

- [インテル® oneAPI ツールキット \(Linux* 版\) 導入ガイド](#) (英語)
- [インテル® oneAPI ツールキット \(Windows* 版\) 導入ガイド](#) (英語)
- [インテル® oneAPI ツールキット \(macOS* 版\) 導入ガイド](#) (英語)

ドキュメント

製品のドキュメントはオンラインでも提供されています。

- [インテル® C++ コンパイラー・クラシック・デベロッパー・ガイドおよびリファレンス](#) (英語)
- [インテル® C++ コンパイラーでサポートされる C++20 の機能](#) (英語)

終了予定のサポート

macOS* 10 のサポートは終了予定です。macOS* 10 がサポートされる最後のリリースはインテル® oneAPI 2022.1 です。

Visual Studio* 2017 のサポートはインテル® oneAPI 2022.1 で終了しました。

Microsoft* Visual Studio* 2017 とインテル® コンパイラーの統合は古い機能 (非推奨) で、将来のリリースで削除される予定です。Visual Studio* との統合のサポートは、Visual Studio* の最新のメジャーバージョンおよび 1 つ前のメジャーバージョンでのみ利用できます。

サポートしていない機能およびサポートを終了した機能

インテル® Xeon Phi™ プロセッサ x200 (開発コード名 Knights Landing、略称 KNL)、インテル® Xeon Phi™ プロセッサ (開発コード名 Knights Mill、略称 KNM)、および Yocto 組込みターゲットのサポートは終了しました。

インテル® Xeon Phi™ 製品では、インテル® Parallel Studio XE 2020 以前のコンパイラー、ライブラリー、ツール、またはインテル® oneAPI ベース・ツールキットおよびインテル® oneAPI HPC ツールキットのバージョン 2021.2/2021.1 のコンパイラーを使用してください。

ループ・プロファイラー機能は将来のリリースで削除される予定です。

ループ・プロファイラーに関連する次のコンパイラー・オプションは古いオプション (非推奨) で、将来のリリースで削除される予定です。

- `profile-loops=keyword`
- `profile-loops-report=value`
- `profile-functions`

- `guide-profile`

インテル® Cilk™ Plus のサポートは終了しました。

ICC -qnextgen コンパイラー・オプションは削除されました。

`qnextgen` (Linux*) および `/qnextgen` (Windows*) コンパイラー・オプションは削除されました。
`icc -qnextgen` または `icc /qnextgen` の代わりに `icx` を使用してください。

インテル® C++ コンパイラー (`icx`) は、インテル® oneAPI ベース・ツールキットのインテル® oneAPI DPC++/C++ コンパイラーとして利用できます。

OpenMP* オフロードは、インテル® oneAPI HPC ツールキットのインテル® oneAPI DPC++/C++ コンパイラーとして利用できます。

インテル® C++ コンパイラー (`icx`) の新機能および OpenMP* オフロードについては、「[インテル® oneAPI DPC++/C++ コンパイラー・リリースノート](#)」を参照してください。

既知の問題

- OpenMP* 5.0 標準に準拠するように、スケジュール種別が `dynamic` または `guided` の場合のワークシェアリング・ループ構文の OpenMP* デフォルト・ループ・スケジュール修飾子を `nonmonotonic` に変更しました。`monotonic` 動作を想定しているユーザーコードは、この変更により正しく動作しなくなる可能性があります。以前のコードの動作を維持するには、`schedule` 節に `monotonic` スケジュール修飾子を追加してください。
- MSVC は 80 ビットの浮動小数点型として `long double` をサポートしたことがないため、Windows* で `-Qlong-double` オプションを使用するとエラーが発生する可能性が高くなります。Microsoft* の数学ライブラリー、書式付き入力と出力は、80 ビット `long double` をサポートしていません。また、Microsoft* C++ 標準ライブラリー (`libcpmt.lib`、`libcpmt.d.lib`) は、標準 `double` 拡張数学関数名と競合する、`frexp1`、`expl`、`log1`、`sin1`、`cos1`、`atan1` などのシンボルを定義します。これらのシンボルは、Microsoft* ライブラリーでは倍精度 (FP64) で実装されていて、インテルの数学ライブラリー (`libmmt.lib`) では拡張倍精度 (FP80) で実装されています。このため、`libcpmt.lib` が最初にリンクされている場合、FP80 数学関数の正しい動作は保証されません。逆に、`libmmt.lib` が最初にリンクされている場合、一部の Microsoft* C++ 標準ライブラリー関数が誤動作する可能性があります。`frexp1` などのシンボルは標準 C++ 関数と同じモジュールで定義されているため、`-Qlong-double` オプションを使用してコンパイルされた単純な C++ プログラムでリンクエラーが発生することが分かっています。このエラーは、`libmmt.lib` の前に `libcpmt.lib` をリンクすることで回避できます。
- **macOS* Big Sur 11 でのエラー**
-`!System` などのライブラリーが見つからないというエラーが発生することがあります。ICC が Xcode* やシステム・ライブラリーを見つけられない問題が発生した場合は、次の手順を実行してください。
Xcode* をインストールした後、次のコマンドを実行します。

```
sudo xcode-select -s /Applications/Xcode.app/Contents/Developer
```


次のコマンドを実行してライセンスを認証します (まだ認証していない場合)。

```
sudo xcodebuild -license
```
- **Xcode* 10 および 11 での「no rule to process file」エラー**
macOS* の Xcode* 10 および 11 には新しいビルドシステムが含まれていて、デフォルトとして構成されています。現在、インテル® コンパイラーはこのビルドシステムをサポートしていません。Xcode* でインテル® C++ コンパイラー・プロジェクトをビルドすると、「no rule to process file」エラーが表示

されます。Xcode* でプロジェクトをビルドするには、プロジェクトの設定を [Legacy Build System] に変更してください。

詳細は、[この記事](#) (英語) を参照してください。

終了予定のサポート

インテル® C++ コンパイラー・クラシック (ICC) は古い機能 (非推奨) で、2023 年後半のリリースで削除される予定です。Windows* および Linux* の継続的なサポート、新しい言語のサポート、新しい言語機能、最適化のサポートが提供される、LLVM ベースのインテル® oneAPI DPC++/C++ コンパイラー (icx) へ移行することを推奨します。2021.7 以降、macOS* のサポートはインテル® プロセッサーを搭載している Mac* コンピューターのみを対象としていることに注意してください。icx の詳細は、『[インテル® oneAPI DPC++/C++ コンパイラー・デベロッパー・ガイドおよびリファレンス](#)』(英語) および『[icc ユーザー向け dpcpp/icx 移行ガイド](#)』を参照してください。

法務上の注意書き

インテルのテクノロジーを使用するには、対応したハードウェア、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。

絶対的なセキュリティを提供できる製品またはコンポーネントはありません。

実際の費用と結果は異なる場合があります。

© Intel Corporation. Intel、インテル、Intel ロゴ、その他のインテルの名称やロゴは、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。

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。

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。

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。

製品および性能に関する情報

¹ 性能は、使用状況、構成、その他の要因によって異なります。詳細については、<http://www.intel.com/PerformanceIndex/> (英語) を参照してください。