

Compaq* Visual Fortran コンパイラからインテル® Visual Fortran コンパイラへのアプリケーションの移植

Steve Lionel – インテル® コンパイラ・サポート

インテル® Fortran コンパイラでは、バージョン 8.0 で、これまでにない大幅な変更が行われました。Linux* 版インテル Fortran コンパイラおよび Windows* 版インテル Fortran コンパイラのバージョン 8.0 は、インテルと Compaq* (現在は Hewlett-Packard*) がこれまで培ってきた Fortran 技術が見事に融合された Fortran コンパイラです。インテルのコード生成、最適化、および並列化技術を Compaq Visual Fortran の機能、拡張仕様、言語処理技術と組み合わせることで、IA-32 システムと Itanium® システムで互換性のある、比類のないランタイム・パフォーマンスを提供する業界最先端の Windows 用および Linux 用 Fortran コンパイラが誕生しました。

ほとんどの場合、ソースを変更することなく、新しいコンパイラでリビルドするだけで構いませんが、アプリケーションによっては、コードの変更とビルド方法の調整が必要な場合があります。この資料では、それらの違いについて説明します。詳細な情報については、コンパイラのリリースノートおよびドキュメンテーションを参照してください。

この資料は、Compaq Visual Fortran (以下 CVF と呼びます) からインテル Visual Fortran 8.0 への移植について説明しています。インテル Fortran 7.1 からの移植については、別の資料を参照してください。特定のプラットフォームのみに適用される項目については、該当する箇所を示します。

製品の互換性

インテル Visual Fortran はシステム上で CVF と共存することができ、必要な場合は以前の製品をそのまま使用しても構いません。その際、インテル Visual Fortran は Microsoft Visual Studio 6 と、CVF は Microsoft Visual Studio.NET と、それぞれ統合できない点に注意してください。2 つの Fortran 製品は別々のフォルダにインストールされ、それぞれ独自のレジストリ変数を使用します。

すべての Fortran ソースはインテル Visual Fortran で再コンパイルする必要があります。CVF でコンパイルされたオブジェクトとスタティック・ライブラリをインテル Visual Fortran で使用することはできません。しかし、CVF で作成したダイナミック・リンク・ライブラリ (DLL) をインテル Visual Fortran アプリケーションで使用することは可能です。CVF 用のサードパーティのライブラリはインテル Visual Fortran で動作しない可能性があります。詳細は、ライブラリの供給元にお問い合わせください。

サポートされていない機能

インテル Visual Fortran は、CVF でサポートされているすべての言語構文をサポートしていますが、CVF の一部の機能についてはサポートしていません。サポートしていない機能には、次のようなものがあります:

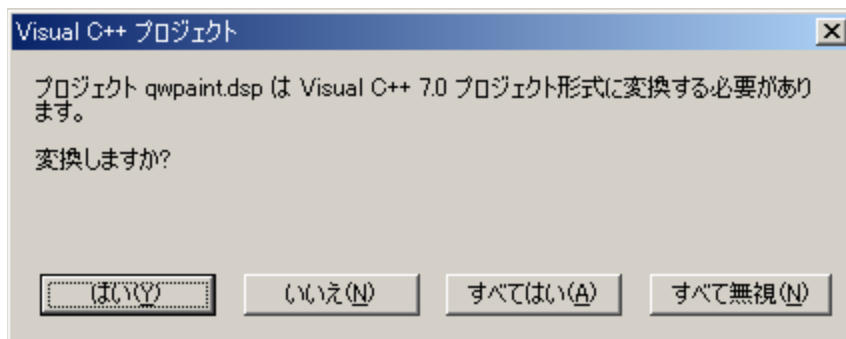
- Fortran FORMAT エディタ
- Compaq 拡張数値演算ライブラリ (インテル[®] マス・カーネル・ライブラリ、または IMSL や NAG のようなサードパーティのライブラリで代用可能です)
- Source Browser クロス・リファレンス・ツール
- COM Server Wizard (CVF Professional Edition)

CVF COM Server Wizard で作成されたアプリケーションをインテル Visual Fortran でビルドすることは可能です。CVF がシステムにインストールされていれば、CVF を使用してインターフェイスを変更することもできます。

CVF プロジェクトの移行

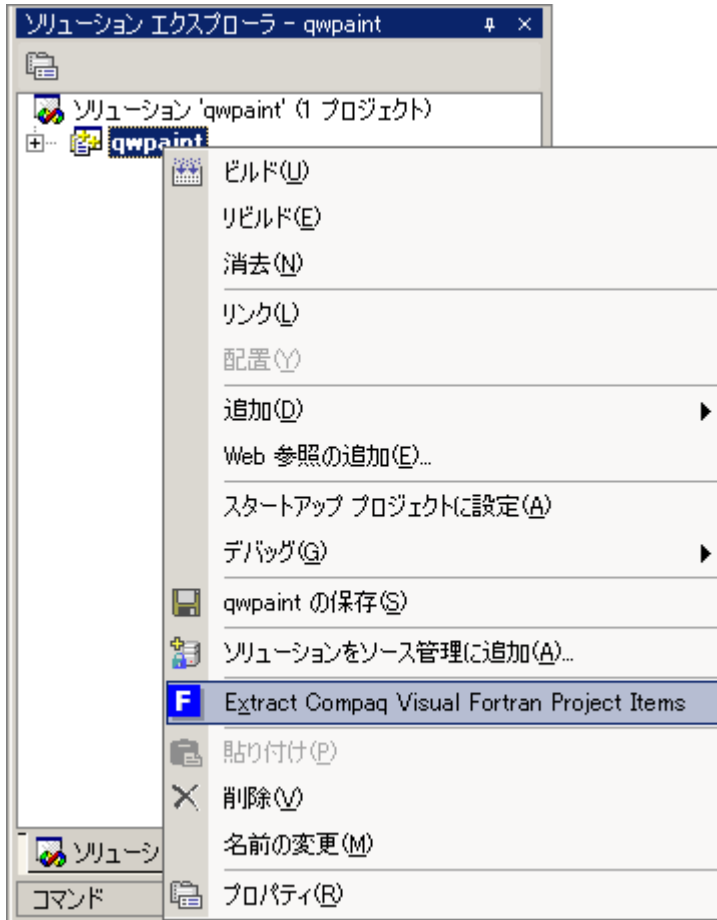
インテル Visual Fortran には、CVF から簡単に移行できるプロジェクト変換ウィザードが用意されています。変換は次の 2 つのステップで行われます:

1. CVF ワークスペースを Visual C++ .NET で開きます。次のようなメッセージが表示されます:

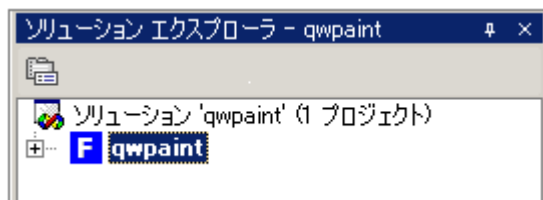


[すべてはい] をクリックして、プロジェクトを Visual C++ .NET プロジェクトの "ソリューション" (ワークスペースのようなもの) に変換します。

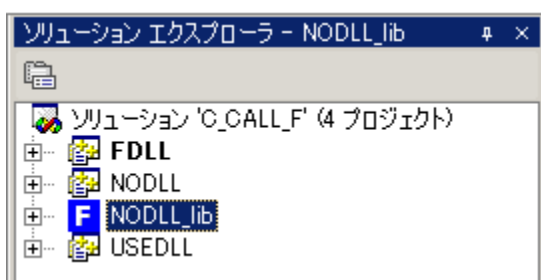
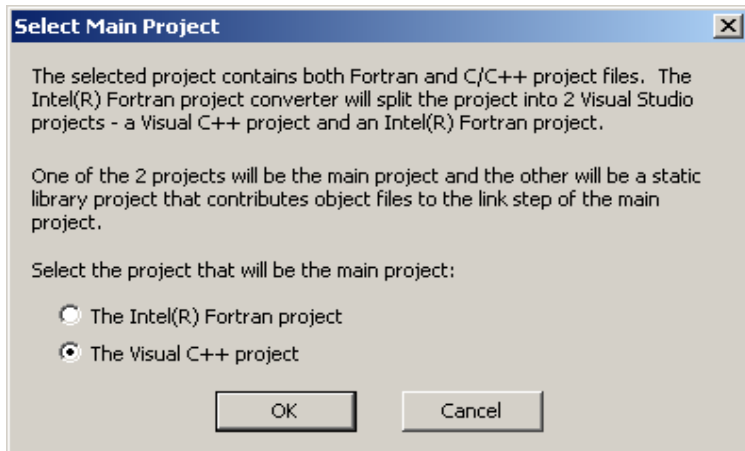
2. 画面の右にある [ソリューション エクスプローラ] タブに、プロジェクトが表示されます。各プロジェクトについて、プロジェクト名を右クリックして、[Extract Compaq Visual Fortran Project Items] を選択します:



プロジェクトが変換されます。



CVFプロジェクトが Fortran と C の両方のソースを含む場合は、ソリューションで 2 つの単一言語プロジェクトに変換し、どちらかをスタティック・ライブラリにビルドして、残りをそのライブラリとリンクする必要があります。これは、Visual Studio.NET では 1 つのプロジェクトで複数の言語を使用できない制限があるためです。プロジェクト変換ウィザードは、Fortran と C のどちらの言語をメイン (リンク可能な) プロジェクトにするかを確認し、必要な調整を行います。複数の言語を含むプロジェクトの変換例を次に示します:



この例では、CVF の NODLL プロジェクトは C++ の実行形式プロジェクト NODLL と、インテル Fortran スタティック・ライブラリ・プロジェクト NODLL.lib に分割されています。変換ウィザードは自動的に NODLL.lib を NODLL に依存させます。そのため、ライブラリが最初に作成されてから、C コードにリンクされます。

ここで注意すべき重要な点は、CVF プロジェクトを変換する場合、変換ウィザードはプロジェクトの設定をデフォルトから CVF 互換になるように変更するということです。このため、CVF では "Default" だったデフォルトの呼び出し規約が、変換ウィザードによって "CVF" に変更されます。呼び出し規約の詳細は、後のセクションを参照してください。

Visual C++.NET IDE

Microsoft Visual C++.NET 統合開発環境 (IDE) は、CVF の操作環境とさまざまな点で異なるため、同じ作業を行うことが多少困難な場合があります。ここでは、主な相違点についてのみ説明します。

プロジェクトとソリューション

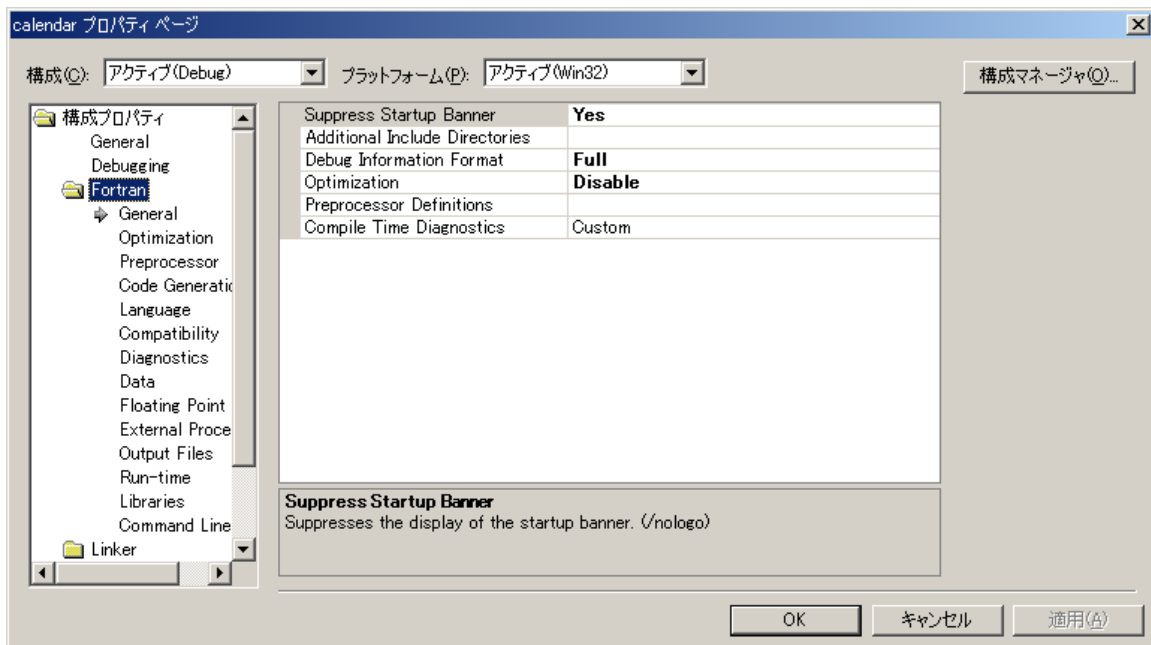
CVF (および Visual C++ 6.0) では、プロジェクトはワークスペースに含まれていました。ワークスペースはコンテナとほとんど同じで、ビルドプロセスには関与していませんでした。プロジェクトで作成したもの (ほとんどの場合、EXE、LIB、または DLL) には Fortran と C の両方のコードを含めることができました。1 つのプロジェクトが常に "アクティブな" プロジェクトとして指定されました。

Visual C++.NET では、プロジェクトを 1 つの言語としか関連付けられない点を除けば、プロジェクトは以前とほぼ同じです。例えば、Fortran プロジェクトに C ファイルを追加した場合、C ファイルは無視されます。ソリューションは、複数のプロジェクトを含む点ではワークスペースと同じですが、ソリューションをビルドして、指定された順 (ユーザが変更できます) で含まれているプロジェクトをすべてビルドできるという点でワークスペースとは異なります。

前述したように、Fortran と C の両方のソースを含むアプリケーションの場合は、Fortran コードは Fortran プロジェクトに、C コードは C (または C++) プロジェクトに含めなければなりません。プロジェクトは別々にビルドされ、必要な場合は、一緒にリンクします。実行形式と DLL プロジェクトの場合は特に問題ありませんが、複数言語のスタティック・ライブラリをビルドしている場合は、1 つのプロジェクトから別のライブラリにオブジェクトを追加する "カスタム・ビルド・ステップ" を行う必要があります。

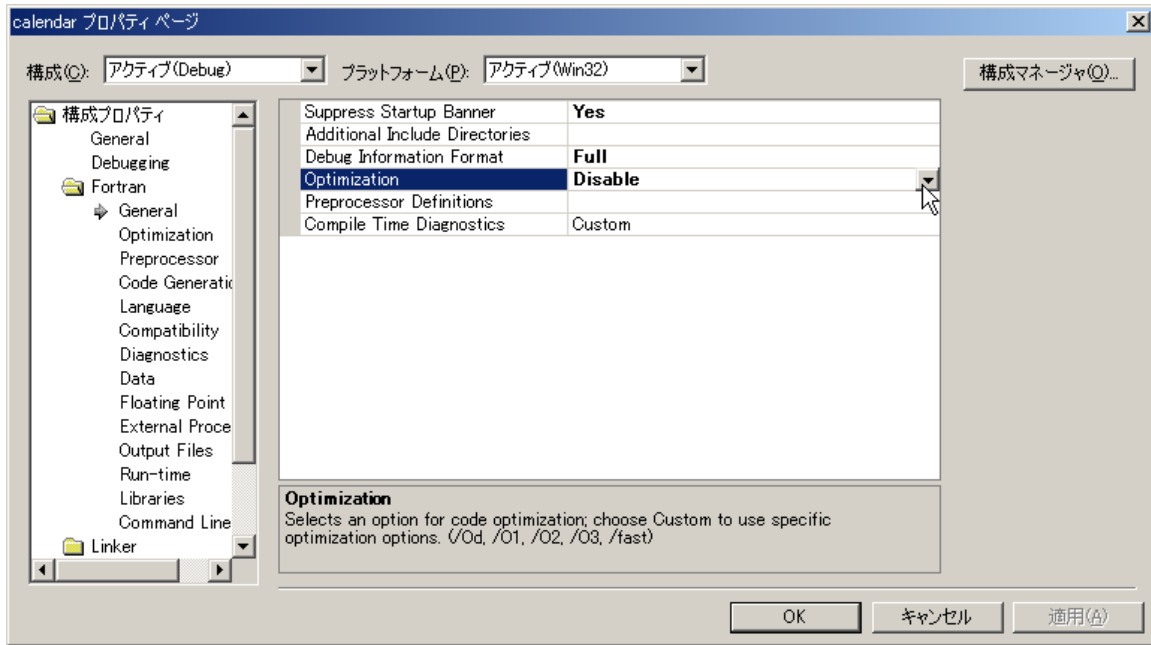
設定の変更

Visual C++.NET IDE で設定を変更する方法も、多少異なります。設定の変更は、タブ付きのダイアログではなく、ツリー形式の "プロパティ" ページで行います。プロジェクトのプロパティ・ページの例を次に示します:



この例では、Fortran の [General] プロパティ・ページが表示されています。デフォルトのプロパティ値が太字で表示され、選択しているプロパティの簡単な説明が画面の下部に表示されています。

プロパティの値を変更するには、値をクリックすると値の右端に表示される、矢印アイコンをクリックします。

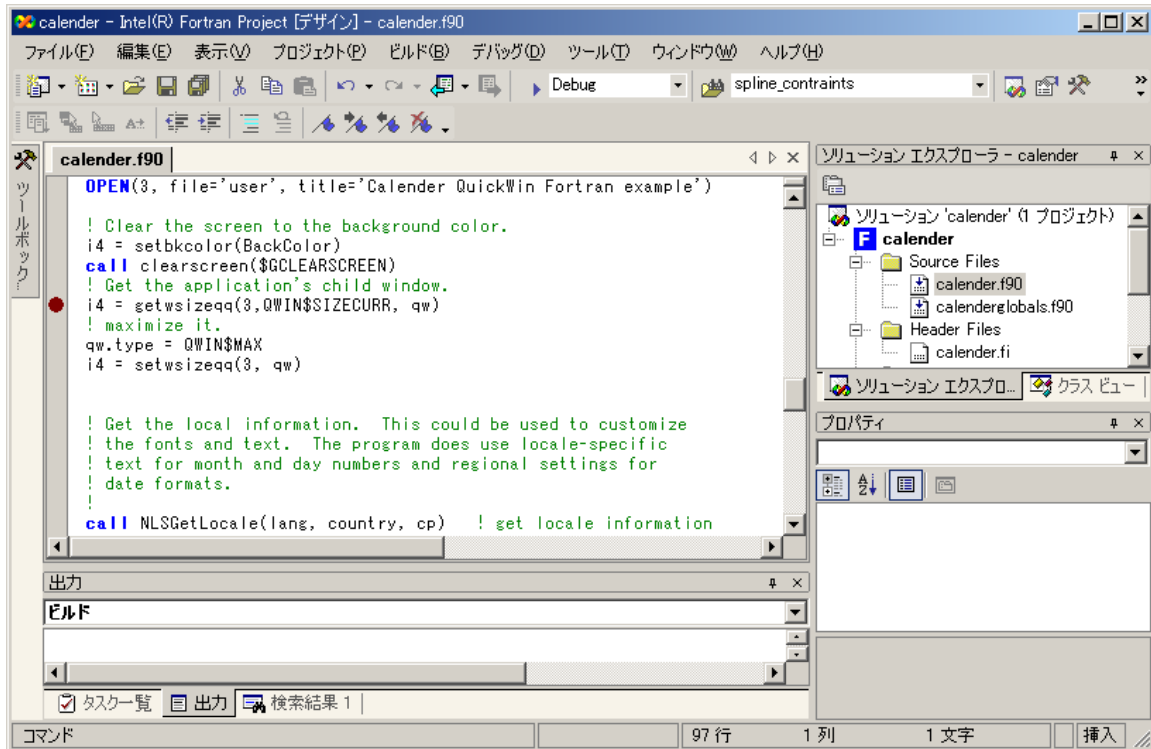


値をクリックして表示されたオプションの中から、必要なオプションを選択します:

上記の例の [Additional Include Directories] のように、項目をリストして指定できるプロパティの場合は、(ドット 3 つの) リストアイコンが表示されます。アイコンをクリックして表示されるダイアログで値を入力します。値が 1 つの場合は、プロパティ・ページに直接入力することもできます。同様に、単一のテキスト文字列のプロパティも、直接入力することができます。プロパティによっては、矢印アイコンをクリックした後、"<inherit from project defaults>" を選択して、プロジェクトのデフォルトを使用するように選択することもできます。

デバッグ

インテル Visual Fortran でのデバッグは CVF と機能的にはほとんど違いはありませんが、コントロールの場所が異なるものがあります。ブレークポイントをセットするには、文の左側の余白をクリックします。ブレークポイントを示す赤のアイコンが表示されます。デバッガで実行を開始するには、構成名の左にある青の右向きの三角形アイコン [続行] をクリックします。



ソースの変更

インテルは、ソースの変更を行わないで済むように (リビルドしてすぐに実行できるように)、これまで細心の注意を払ってきました。しかし、インテル・コンパイラと Compaq コンパイラの実装の差により、選択したアプリケーションでいくつかの変更が必要になることがあります。

"フロントエンド" とよく呼ばれる、Fortran の構文とセマンティクスを制御するコンパイラの部分は、CVF から継承され、正しくない使用法を検出するように改良されました。このため、新しいコンパイラでは、以前のコンパイラでは出力されなかった、使用法についての診断メッセージが発行されます。例えば、コンパイラは、純粋プロシージャから `RANDOM_NUMBER` のような非純粋組込み関数への呼び出しを行うソースをエラーにします。エラーが発生した場合は、コーディング・エラーを修正する必要があります。診断に不明な点がある場合は、インテル® プレミア・サポートまでお問い合わせください。

デフォルトの呼び出し規約の変更

CVF では、デフォルトの呼び出しメカニズムは `STDCALL` で、ルーチン名は最後に `@n` (`n` は引数リストのバイト数) が追加されていました。インテル Visual Fortran では、インテル Fortran の以前のバージョンで使用されていたものよりも、より一般的な C の呼び出しメカニズムを採用しました。ルーチン名は、デフォルトでは以前と同じように大文字に変換されますが、先頭に下線 (`_`) が追加され、`@n` は追加されません。

Fortran のみを含むアプリケーションでは、この変更はあまり重要ではありません。しかし、C を含む複数言語のアプリケーションでは、コンパイラのデフォルトが変更されたことによる影響に注意する必要があります。[External Procedures] プロパティ・ページで、または `/iface:cvf` コマンドライン・スイッチを使用して、CVF のデフォルトを使用するようにコンパイラに指示することができます。

GETARG、IARGC、および NARGS

コマンドライン問い合わせルーチン GETARG、IARGC、および NARGS は、コンパイラで組み込みプロシージャとして認識されるようになりました。アプリケーションのソースでこれらのいずれかの名前を EXTERNAL として宣言しているか、それらに明示的なプロシージャ・インターフェイスを提供している場合、リンク時にエラーが発生しないように、これらの宣言を削除する必要があります。

システムとライブラリ宣言の新しいモジュール名

Compaq Visual Fortran は、Fortran ライブラリ・ルーチン用のモジュールに加えて、Win32 API ルーチンとシンボルの定義を含むモジュールを提供していました。これらのモジュールには、DFWIN、DFLIB のような名前が付けられていました。インテル Visual Fortran でも同じ名前のモジュールが提供されていますが、これらは新しい名前のモジュールのラップです。既存のソースを変更する必要はありませんが、新しく作成する場合は新しい名前を使用するようにしてください。例えば、DFLIB の場合、シンボルは 3 つの新しいモジュール (IFCORE、IFPORT、および IFQWIN) のうちの 1 つに再配置されます。下記の表を参考にして、必要なモジュールを選択してください。

以前の名前	新しい名前	説明
DFAUTO	IFAUTO	オートメーション
DFCOM	IFCOM	COM、OLE
DFCOMTY	IFCOMTY	古い名前です。IFWINTY を使用してください。
DFLIB	IFCORE、IFPORT、IFQWIN	一般ライブラリ、移植ライブラリ、QuickWin
DFLOGM	IFLOGM	ダイアログ
DFLOGMT	IFLOGMT	ダイアログの種類と定数
DFMT	IFMT	マルチスレッド・ルーチン
DFNLS	IFNLS	National Language Support (NLS)

DFOPNGL	IFOPNGL	OpenGL ルーチン
DFOPNGLT	IFOPNGLT	OpenGL の種類と定数
DFPORT	IFPORT	移植ルーチン
DFWBASE	IFWBASE	古い WIN16 ルーチン
DFWIN	IFWIN (または、KERNEL32、 USER32 のような個々のモジュール)	WIN32 API ルーチン
DFWINA	IFWINA	QuickWin ルーチンと重複するため名前が変更された WIN32 ルーチン
DFWINTY	IFWINTY	WIN32 API の種類と定数

KERNEL32 のような個々の WIN32 API モジュールの名前は、インテル Visual Fortran でも Compaq Visual Fortran でも同じです。

ビルドの変更

ほとんどの場合、ビルド・プロシージャを変更する必要はありません。しかし、インテル Fortran コンパイラの将来のバージョンとの互換性のために、いくつかの点を変更することを推奨します。このセクションでは、アプリケーションのビルドに影響する違いについて説明します。

新しいコンパイル・コマンド ifort

インテル Fortran の以前のバージョンは、ifl および efl (Windows)、ifc および efc (Linux) のように、さまざまなコマンドを使用してコマンドラインからコンパイラを起動していました。バージョン 8.0 では、すべてのプラットフォームで同じコンパイル・コマンド ifort を使用するように変更されました。この変更により、複数のプラットフォーム上でアプリケーション開発がより簡単になりました。コマンド・セッションを開始するときに特定の "vars" スクリプトを使用することで、適切なコンパイラが選択されます。

バージョン 8.0 で、以前の名前を使用することもできますが、新しい名前への変更を促すメッセージが表示されます。このメッセージを表示しないようにするには、/quiet (Windows) または -quiet (Linux) オプションを使用してください。

新しい機能

厳密には移植に影響する部分ではありませんが、インテル Visual Fortran は、プログラムを新しい環境で実行する際に利用すると便利な、さまざまな新しい機能を提供しています。これらの機能には、次のようなものがあります：

- 4 倍精度浮動小数点 REAL(16) と COMPLEX(32)
- 自動並列化
- OpenMP
- 最新のインテル[®] プロセッサ用の高度な最適化
- コード・カバレッジ・ツールとテスト・プライオリタイゼーション・ツール

オンライン・ヘルプ

さまざまなヘルプを利用することができます。一般的な使い方に関する質問は、インテル Fortran ユーザ・フォーラム <http://softwareforums.intel.com/> (英語) を参照してください。他のユーザに質問したり、過去的话题を検索することで、問題解決の助けとなる場合があります。

製品の不具合を発見した場合は、インテル・プレミア・サポートまでご連絡ください。インテル・プレミア・サポートを利用するためには、まず登録を行う必要があります。コンパイラをインストールするときに行うか、または Registration Center <http://www.intel.com/software/products/registrationcenter/index.htm> (英語) で登録することができます。登録が完了したら、<https://premier.intel.com/> (英語) にログインして、サポートを依頼してください。