

IJCAD 2026 アプリケーション開発での特記事項

2026年2月

本書では IJCAD 2026 のアプリケーション開発に関する特記事項をまとめています。

Visual Studio のバージョン変更

IJCAD 2026 では、GRX と .NET API で使用する Visual Studio のバージョンが変更されました。

変更前：Visual Studio 2017 (バージョン 15.9.17 以降)

変更後：Visual Studio 2022 (バージョン 17.8.0 以降)

開発環境の変更点

IJCAD 2024 以降では 32bit 版の提供がなくなり、64bit 版のみの提供となっています。

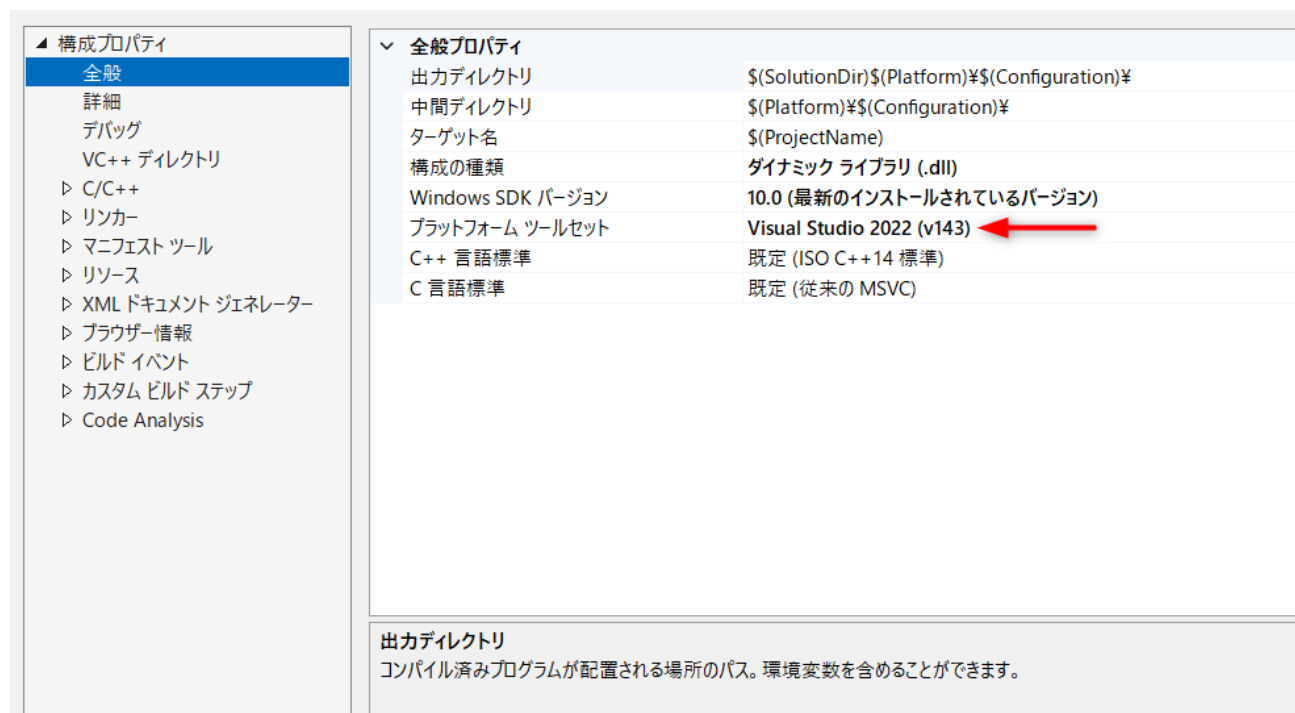
そのため、32bit 環境依存のソースコードを含む場合は、該当部分の変更が必要になります。

各詳細は開発ヘルプをご確認ください。

API	変更点
.NET API	IJCAD 2025 では .NET Framework 4.8 対応でしたが、IJCAD 2026 から .NET 8 対応となりました。 それに伴い、下記の注意点があります。 ① 従来のプロジェクトファイルは使用不可 ② .dll ファイルのプロジェクト参照が必要 ③ 名前空間の変更 ④ 名前空間の衝突
AutoLISP	開発環境に変更はありませんが、バージョンアップ時はテストの実施を推奨します。
DIESEL	開発環境に変更はありませんが、バージョンアップ時はテストの実施を推奨します。
VBA	開発環境に変更はありませんが、バージョンアップ時はテストの実施を推奨します。 ※ Microsoft Visual C++ 2010 再頒布可能パッケージがインストールされている環境が必要です。該当の環境は Microsoft のサイトにて提供されています。
GRX	IJCAD 2024 以降で、下記の変更があります。 ① モジュール構成の変更 ② SDK のフォルダ構成の変更 ③ 一部 API の廃止/変更 (AutoCAD 互換性の向上)

Visual Studio 2022 を用いた GRX モジュールの作成方法

Visual Studio 2022 で GRX モジュールを生成する場合、プロジェクトに設定する「プラットフォーム ツールセット」に「Visual Studio 2022 (v143)」を指定します。



The screenshot shows the 'Properties' window for a project, specifically the 'General' tab. The 'Platform Toolset' property is highlighted with a red arrow, indicating it is set to 'Visual Studio 2022 (v143)'. The left sidebar shows the 'Build Properties' tree with 'General' selected. Below the main table, the 'Output Directory' property is also visible.

▼ 全般プロパティ	
出力ディレクトリ	\$(SolutionDir)\$(Platform)¥\$(Configuration)¥
中間ディレクトリ	\$(Platform)¥\$(Configuration)¥
ターゲット名	\$(ProjectName)
構成の種類	ダイナミック ライブラリ (.dll)
Windows SDK バージョン	10.0 (最新のインストールされているバージョン)
プラットフォーム ツールセット	Visual Studio 2022 (v143) ←
C++ 言語標準	既定 (ISO C++14 標準)
C 言語標準	既定 (従来の MSVC)

出力ディレクトリ
コンパイル済みプログラムが配置される場所のパス。環境変数を含めることができます。

「プラットフォーム ツールセット」に「Visual Studio 2022 (v143)」が表示されない場合は、Visual Studio 2022 で追加インストールが必要です。

詳細は Visual Studio のヘルプをご覧ください。

(GRX) Visual Studio のスケルトンで stdafx.h が生成されない問題について

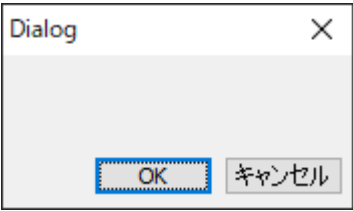
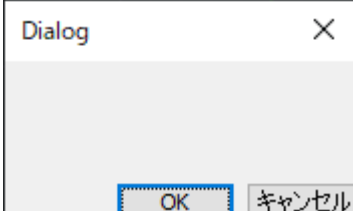
Visual Studio の最新版で GRX モジュールのスケルトンを生成した場合、プリコンパイル済みヘッダーのファイルが stdafx.h ではなく、pch.h と framework.h が作成される場合があります。

GRX SDK として必要なヘッダーファイルやマクロは、pch.h の #include "framework.h" の後ろに追加します。

```
6 // 頻りに更新するファイルをここに追加しないでください
7 #ifndef PCH_H
8 #define PCH_H
9
10 // プリコンパイルするヘッダーをここに追加します
11 #include "framework.h"
12
13 #include "arxHeaders.h"
14 AC_DECLARE_EXTENSION_MODULE(IJHelloDII)
15
16 #endif //PCH_H
17
```

4K ディスプレイ対応による変更点

IJCAD 2020 から 4K ディスプレイへ正式に対応しました。プログラムでダイアログサイズやコントロールの位置を変更している場合、以下のようなずれが発生する場合があります。

IJCAD 2019 以前	IJCAD 2020 以降
	

原因の多くはウィンドウの枠線の幅を固定値で計算しているためです。対策としてはウィンドウ枠線の幅を OS (例、Win32 API の GetSystemMetrics()関数) から取得するようにソースコードを変更する必要があります。

AutoCAD .NET API アプリケーションから移植する際の注意事項

AutoCAD で .NET API を利用したアプリケーションを IJCAD へ移植する際は以下にご注意ください。

(サンプルコードは C#用です)

using の使用

インスタンスを生成するコードでは以下のように using 構文を使用します。

変更前：

```
var mapping = new IdMapping();
```

変更後：

```
using( var mapping = new IdMapping())  
{  
    ...  
}
```

または、

```
using var mapping = new IdMapping();
```

ResultBuffer の使用

ResultBuffer クラスでは、UnmanagedObject の代わりに ResbufObject を利用します。

コンストラクタで ResultBuffer(IntPtr, bool)を利用したい場合は、ResultBuffer.Create(IntPtr, bool)で代用してください。

GetOrthoProjectedCurve を用いた作図の違い

GetOrthoProjectedCurve メソッドで得られた情報を元に作図をする場合、ブロックテーブルレコードへの AppendEntity および、トランザクションに対する AddNewlyCreatedDBObject の両方の呼び出しが必要です。(＃4078)

AutoCAD と IJCAD の既定のロックモードの違い

Application.DocumentManager.MdiActiveDocument.LockMode()で返される結果が AutoCAD と IJCAD とでは異なります。(＃1750)

- ① コマンドを CommandFlags.Modal で登録した場合
 - ・ AutoCAD : Write が返されます。
 - ・ IJCAD : NotLocked が返されます。
- ② コマンドを CommandFlags.Session で登録した場合
 - ・ AutoCAD : NotLocked が返されます。
 - ・ IJCAD : ProtectedAutoWrite が返されます。

DrawJig クラスで描画したオブジェクトがちらつく問題

DrawJig クラスを使ってラバーバンドなどを描画した場合、AutoCAD と比較すると、IJCAD ではちらついて見えるケースがあります。こちらは制限となります。(＃5636)

非 Session モードでブレークポイントを使用したデバッグができない問題

IJCAD 2026 において、.NET アプリケーションで CommandFlags.Modal の関数 または 呼び出された関数内にブレークポイントを設定した場合、デバッグの実行時に例外が発生します。

CommandFlags.Session の関数では発生しません。(正常にブレークポイントで停止する)

ダイアログのボタンを押したタイミングでブレークポイントを設定しても停止しない問題

.NET アプリケーションでダイアログのボタンをクリックしたイベント内でブレークポイントを設定しても、Visual Studio のデバッガではプログラムを停止することはできません。(＃1312)

AutoCAD から移植する際のその他の注意事項

.NET API 以外で AutoCAD から IJCAD へ移植する際の注意事項をまとめました。

複数図面を開いた状態でアクティブの図面を変更した際の挙動の違い

複数図面を開いている状態で、コマンド実行中にアクティブ図面を変更して再び元の図面をアクティブに戻した場合、IJCAD ではコマンドが強制終了されずに継続されます。AutoCAD (2015) 以降では実行中のコマンドが強制的にキャンセルされます。

(VBA) GcadDocument.EndCommand イベント内で他のコマンドを呼び出したときの制限について

VBA の GcadDocument.EndCommand イベント内で、SendCommand 文を使って他のコマンドを呼び出した場合、無限ループに陥ることがあります。これは IJCAD がコマンドキャンセル時に、CancelCommand ではなく EndCommand を呼び出すためです。EndCommand イベント内では他のコマンドを呼び出さないでください。(＃2121)