AUTODESK

Autodesk Desktop API .NET Core Migration

Embracing Modern .NET

January 30, 2024

Autodesk Confidential. Do not share publicly. For the use by customers who are under NDA (Non-Disclosure Agreement) which you accepted when you joined feedback portal or ADN (Autodesk Developer Network) program.

Safe Harbor Statement

The presentations during this event may contain forward-looking statements about our outlook, future results and related assumptions, total addressable markets, acquisitions, products and product capabilities, and strategies. These statements reflect our best judgment based on currently known factors. Actual events or results could differ materially. Please refer to our SEC filings, including our most recent Form 10-K and Form 10-Q filings available at www.sec.gov, for important risks and other factors that may cause our actual results to differ from those in our forward-looking statements.

The forward-looking **Safe Harbor Statement** statements made in these presentations are being made as of the time and date of their live presentation. If these presentations are reviewed after the time and date of their live presentation, even if subsequently made available by us, on our website or otherwise, these presentations may not contain current or accurate information. We disclaim any obligation to update or revise any forward-looking statements.

Statements regarding planned or future development efforts for our products and services are not intended to be a promise or guarantee of future availability of products, services, or features but merely reflect our current plans and based on factors currently known to us. Purchasing decisions should not be made based upon reliance on these statements.

PLEASE NOTE: All Autodesk content is proprietary. Please Do Not Copy, Post or Distribute without authorization.

Today's Presenters

Jan Richter
Software Architect
Revit

Madhukar Moogala
Principal Developer Advocate
Engineer, Platform

Derek Stratton
Software Engineer
Revit

Chandra Shekar Gopal
Senior Developer Advocate Engineer,
Manufacturing

Michael Morris
Principal Engineer
Revit

Zhong Wu
Principal Developer Advocate
Engineer, AEC

Mikako Harada
Sr. Manager
Developer Technical Services, AEC

Today’s Webinar – Objectives

- Autodesk moving to .NET Core 8.0
  - Raise awareness among developer communities
  - Start migrating and testing now
  - Report any issues ASAP
- What is covered
  - Pointers to get started with migration
  - Share our experience so far
- What is NOT covered
  - product feature API update (.NET migration only)

Agenda

1. The Why - .NET Core overview
2. Migration basics
3. Product specific information
    - AutoCAD + verticals
    - Revit
    - Inventor
    - Others (including not migrating)
4. Q & A

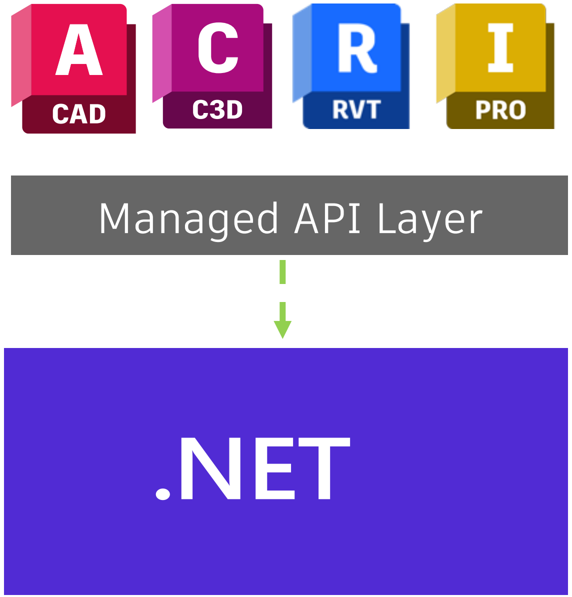
.NET Core Overview

Open-source Powerhouse for Modern Apps

Move to .NET Core

NET Core, NET 6.0 /7.0/8.0 and NET Platform all are synonyms indicating modern technology.

- Monolithic → Open Source & Cross Platform
- Windows Only → Multiplatform Evolution
- .NET Core Adoption → Unifying Codebases
- .NET 5.0 Launch → Modern Era Begins



Managed API Layer

.NET

Embracing Modern .NET Core Technology

State of .NET Framework

.NET Framework 4.8.x is the final major version.

Microsoft promises to keep the .NET Framework alive (it is part of Windows!)

Motivation

Why we are moving to NET Platform

Growth Opportunity

Connected Workflows

Performance Benefits

Growth Opportunity

NET is now cross platform.

One Consistent API

Global Appeal

Connected Workflows

Desktop to Cloud

SDKS and libraries

Minimal API’s and gRPC

Cloud Native Features

Performance Benefits

Smaller Footprint and Faster Startup

Higher Throughput and Lower Resource Consumption

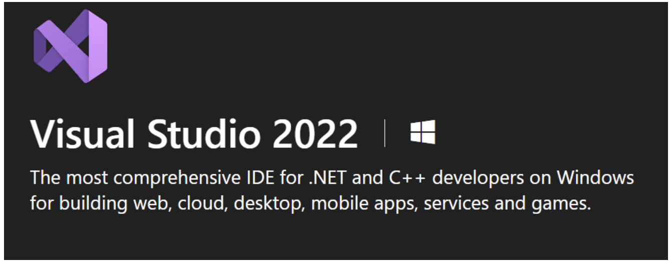
Scalability and Cloud-friendliness

- There is no change in performance on API due to change from .NET 4.x to .NET 8.0
- The Autodesk product-specific performance is still to be determined (TBD)

Migration Basics

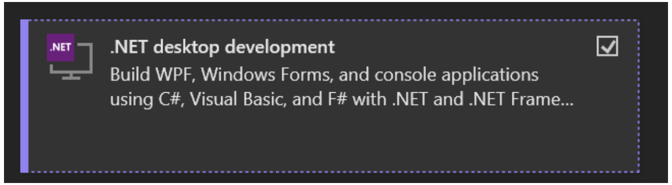
.NET 8: Development Environment

- **Visual Studio (VS) 2022** (Community Edition also works)
  - https://visualstudio.microsoft.com/downloads/
  - Select `.NET Desktop Development` in the VS installer page.
  - [(Optional) Download .NET 8]() (VS 17.8.3 includes .NET 8)
- **Upgrade Assistant –** Extension to update your app
  - [https://learn.microsoft.com/en-us/dotnet/core/porting/upgrade-assistant-install]()



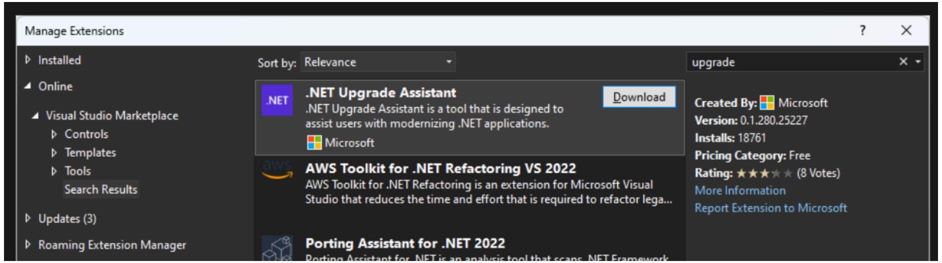
Visual Studio 2022

The most comprehensive IDE for .NET and C++ developers on Windows for building web, cloud, desktop, mobile apps, services and games.



☑ .NET desktop development

Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET and .NET Frame...



Sort by: Relevance

upgrade

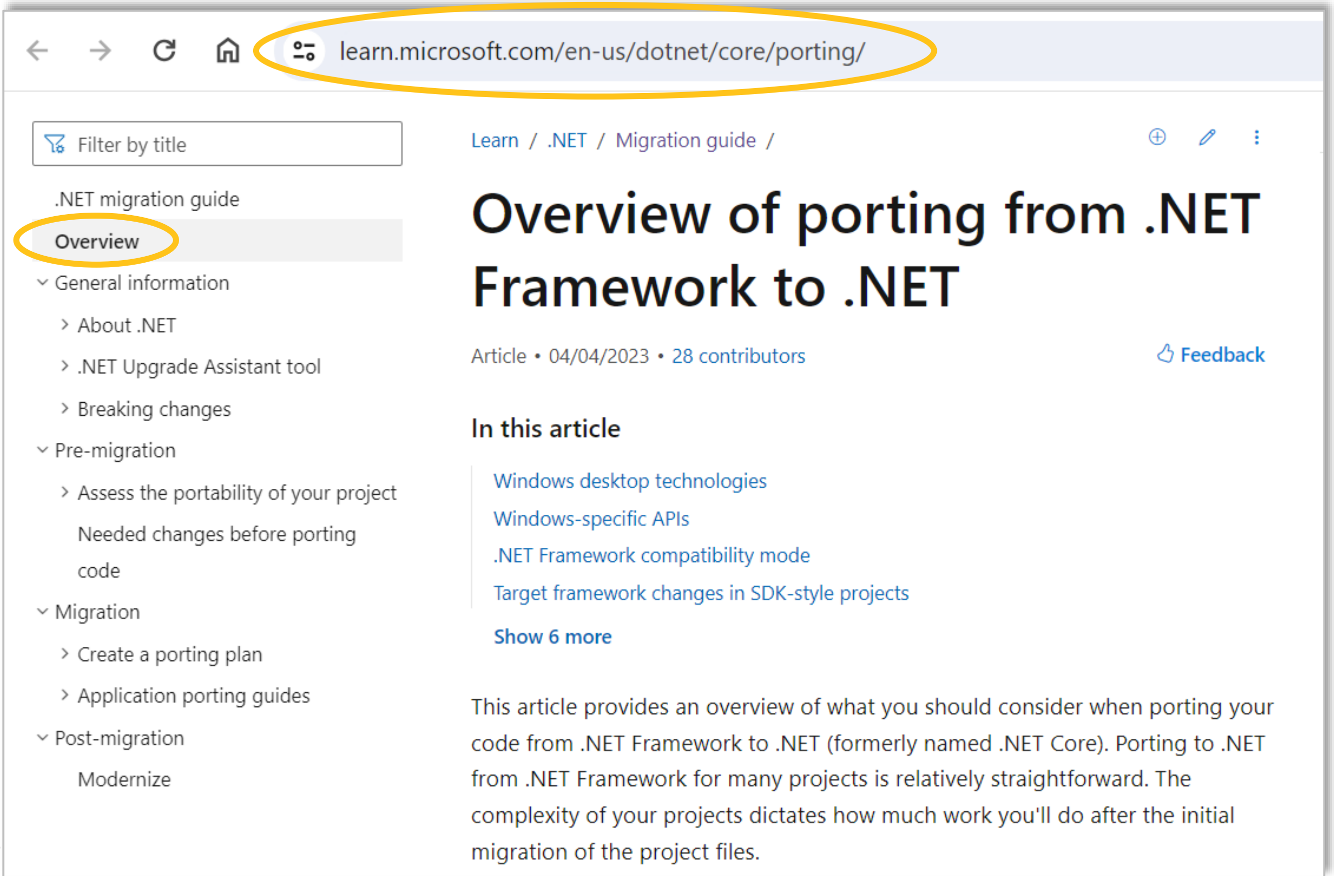
.NET Upgrade Assistant
.NET Upgrade Assistant is a tool that is designed to assist users with modernizing .NET applications.
Microsoft

AWS Toolkit for .NET Refactoring VS 2022
AWS Toolkit for .NET Refactoring is an extension for Microsoft Visual Studio that reduces the time and effort that is required to refactor lega...

Porting Assistant for .NET 2022
Porting Assistant for .NET is an analysis tool that scans .NET Framework

Created By: Microsoft
Version: 0.1.280.25227
Installs: 18761
Pricing Category: Free
Rating: ★★★☆☆ (8 Votes)
More Information
Report Extension to Microsoft

.NET 8 Porting – Learning Material on Microsoft



learn.microsoft.com/en-us/dotnet/core/porting/

Filter by title

.NET migration guide

Overview

General information

About .NET

.NET Upgrade Assistant tool

Breaking changes

Pre-migration

Assess the portability of your project

Needed changes before porting code

Migration

Create a porting plan

Application porting guides

Post-migration

Modernize

Learn / .NET / Migration guide /

Overview of porting from .NET Framework to .NET

Article • 04/04/2023 • 28 contributors

Feedback

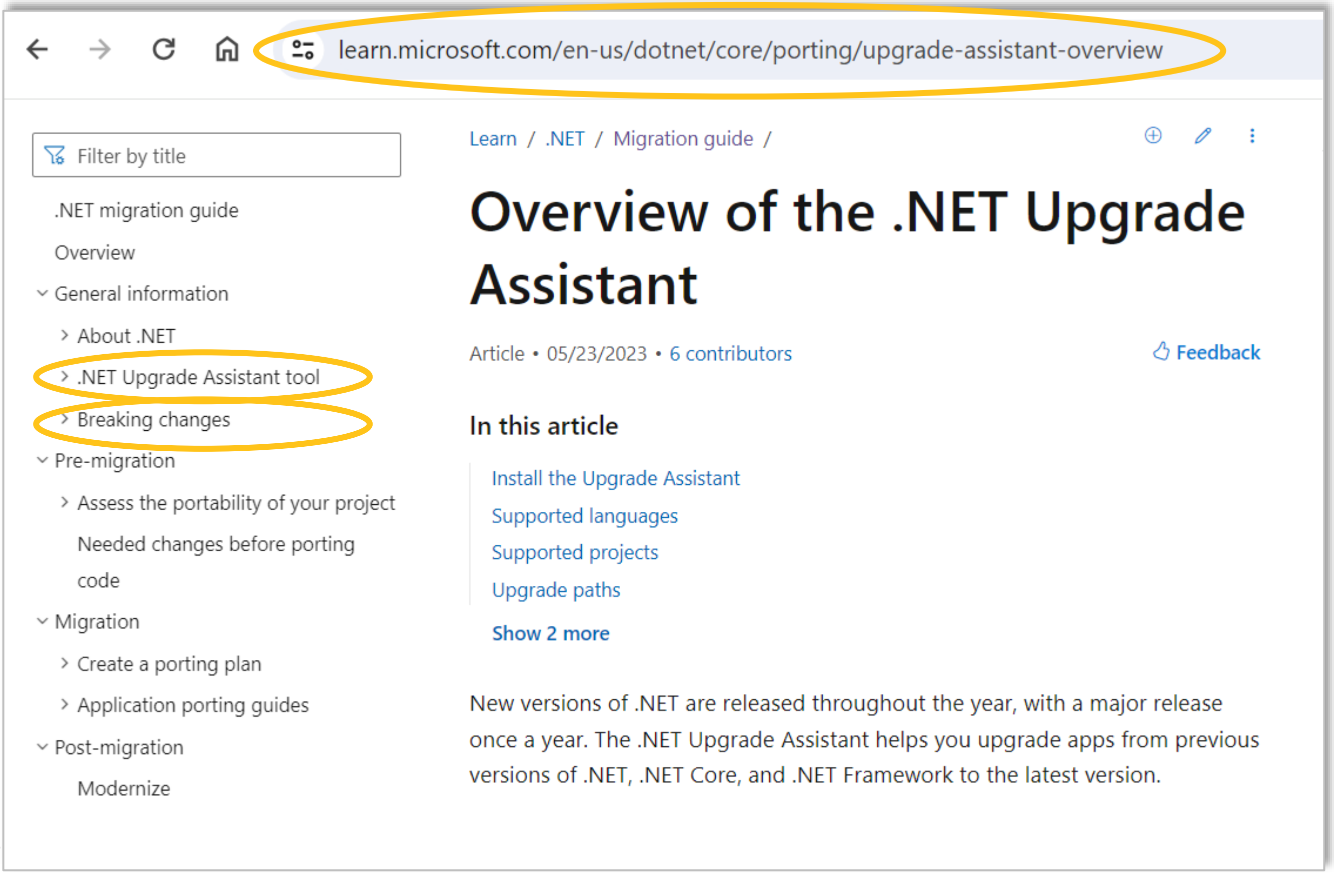
In this article

- Windows desktop technologies
- Windows-specific APIs
- .NET Framework compatibility mode
- Target framework changes in SDK-style projects

Show 6 more

This article provides an overview of what you should consider when porting your code from .NET Framework to .NET (formerly named .NET Core). Porting to .NET from .NET Framework for many projects is relatively straightforward. The complexity of your projects dictates how much work you'll do after the initial migration of the project files.

.NET 8 Porting – Learning Material on Microsoft



learn.microsoft.com/en-us/dotnet/core/porting/upgrade-assistant-overview

Filter by title

- .NET migration guide
- Overview
- General information
  - About .NET
  - .NET Upgrade Assistant tool
  - Breaking changes
- Pre-migration
  - Assess the portability of your project
  - Needed changes before porting code
- Migration
  - Create a porting plan
  - Application porting guides
- Post-migration
  - Modernize

Learn / .NET / Migration guide /

Overview of the .NET Upgrade Assistant

Article • 05/23/2023 • 6 contributors

Feedback

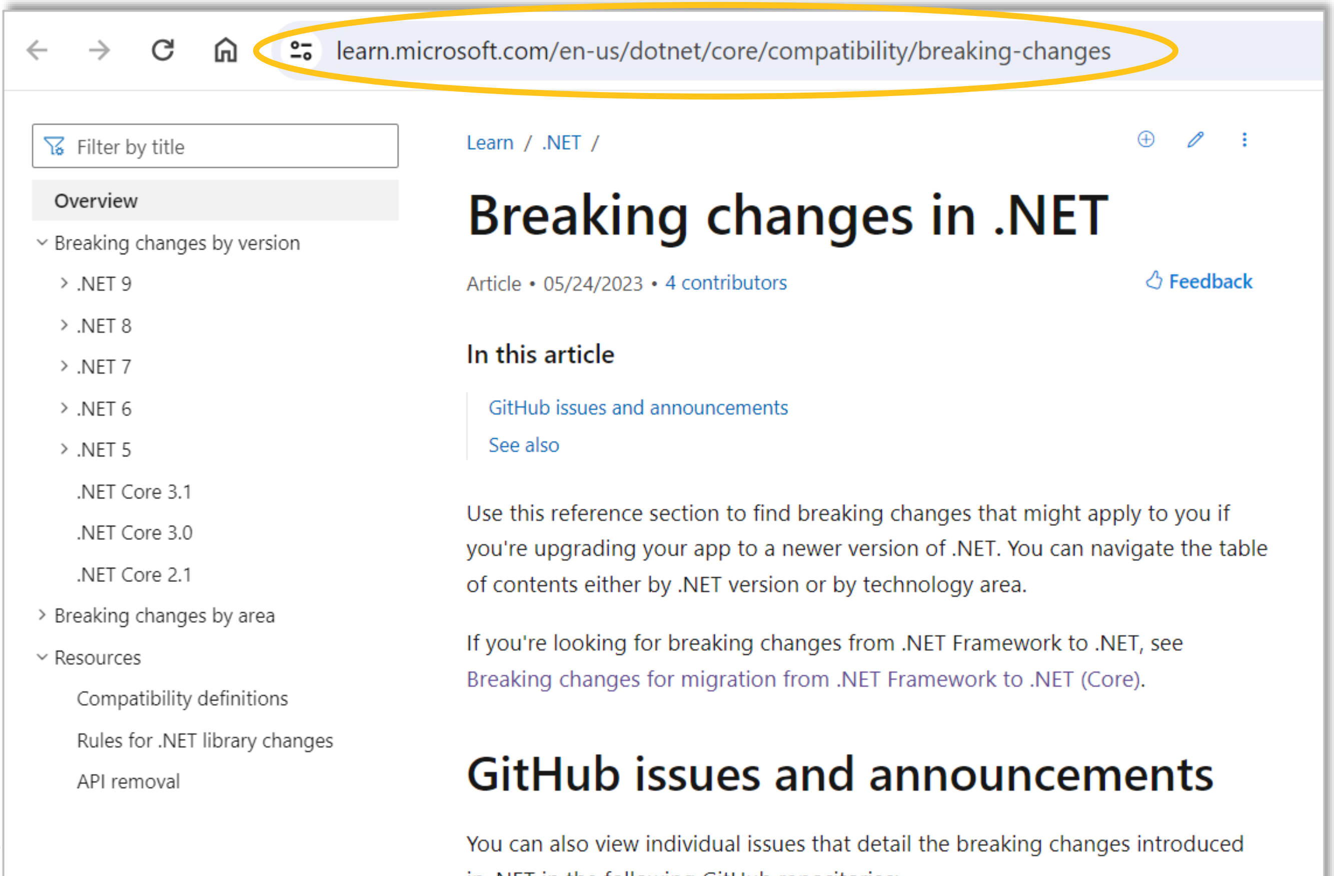
In this article

- Install the Upgrade Assistant
- Supported languages
- Supported projects
- Upgrade paths

Show 2 more

New versions of .NET are released throughout the year, with a major release once a year. The .NET Upgrade Assistant helps you upgrade apps from previous versions of .NET, .NET Core, and .NET Framework to the latest version.

.NET 8 Porting – Learning Material on Microsoft


learn.microsoft.com/en-us/dotnet/core/compatibility/breaking-changes

Filter by title

- Overview
- Breaking changes by version
  - .NET 9
  - .NET 8
  - .NET 7
  - .NET 6
  - .NET 5
  - .NET Core 3.1
  - .NET Core 3.0
  - .NET Core 2.1
- Breaking changes by area
- Resources
  - Compatibility definitions
  - Rules for .NET library changes
  - API removal

Learn / .NET /

Breaking changes in .NET

Article • 05/24/2023 • 4 contributors

Feedback

In this article

GitHub issues and announcements

See also

Use this reference section to find breaking changes that might apply to you if you're upgrading your app to a newer version of .NET. You can navigate the table of contents either by .NET version or by technology area.

If you're looking for breaking changes from .NET Framework to .NET, see Breaking changes for migration from .NET Framework to .NET (Core).

GitHub issues and announcements

You can also view individual issues that detail the breaking changes introduced

Desktop Products - preview build

feedback.autodesk.com >> join and look for opportunity >> Download

AutoCAD Customer Council

Revit Preview

Factory Design Utilities/Inventor

Infrastructure Futures

Under Non-Disclosure Agreement (NDA)

If you haven’t yet, please join and start testing.

Desktop Products – migration guide/discussion

feedback.autodesk.com >> join and look for opportunity >> **User Forums**

AutoCAD Customer Council

>> Forums >> ObjectARX API >> *“How to Create, Build and Debug .NET 8 Plugin for AutoCAD Venn”* (recording)

Revit Preview

>> Forums >> API General >> *“Enhancement Update - Revit .NET 8 Preview Release API”*

Factory Design Utilities/Inventor

>> Forums >> iLogic, API and Automation >> *“.Net Framework based Inventor projects upgrade to .Net 8”*

Infrastructure Futures

>> Forums >> Autodesk Civil 3D 2025

Keep the discussion within the feedback portal

Migration Steps – C# projects (csproj)

1. Convert C# projects to new SDK style format
   - Use **.NET Upgrade Assistant**

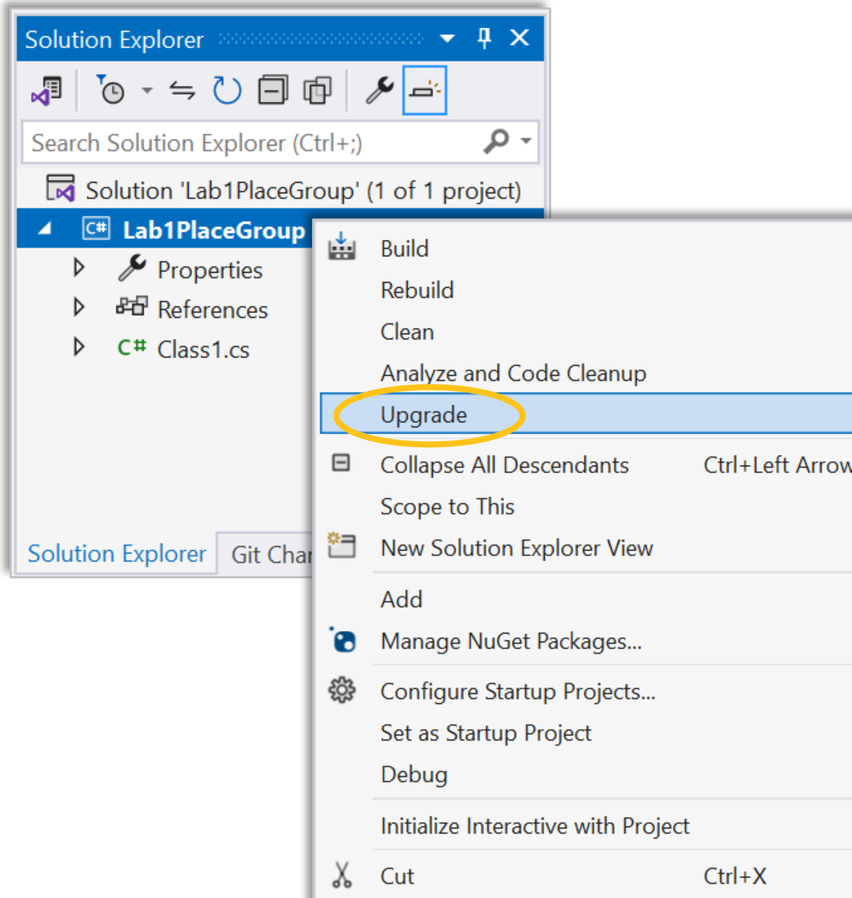
2. Update target framework to .NET 8.0-Windows
   <TargetFramework>**net8.0-windows**
   </TargetFramework>
   - WPF – add **<UseWPF>true</UseWPF>**
   - WinForms – add **<UseWindowsForms>true</UseWindowsForms>**

3. Update references. System references can be removed. (default in .NET 8)

4. Address incompatible packages, library references, obsolete codes

Migration Steps – C# projects (csproj)

1. Convert C# projects to new SDK style format
   - Use **.NET Upgrade Assistant**
2. Update target framework to .NET 8.0-Windows <TargetFramework>**net8.0-windows**</TargetFramework>
   - WPF – add **<UseWPF>true</UseWPF>**
   - WinForms – add **<UseWindowsForms>true</UseWindowsForms>**
3. Update references. System references can be removed. (default in .NET 8)
4. Address incompatible packages, library references, obsolete codes



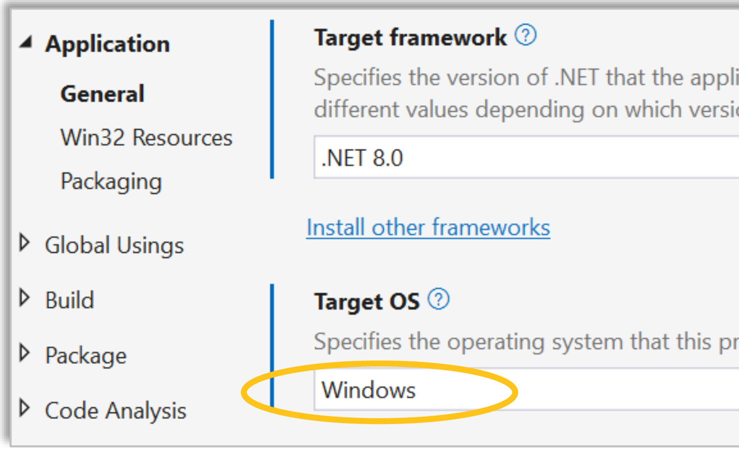
Migration Steps – C# projects (csproj)

1. Convert C# projects to new SDK style format
   - Use **.NET Upgrade Assistant**

2. Update target framework to .NET 8.0-Windows <TargetFramework>**net8.0-windows**</TargetFramework>
   - WPF – add **<UseWPF>true</UseWPF>**
   - WinForms – add **<UseWindowsForms>true</UseWindowsForms>**

3. Update references. System references can be removed. (default in .NET 8)

4. Address incompatible packages, library references, obsolete codes



Target framework

Specifies the version of .NET that the appli
different values depending on which versi

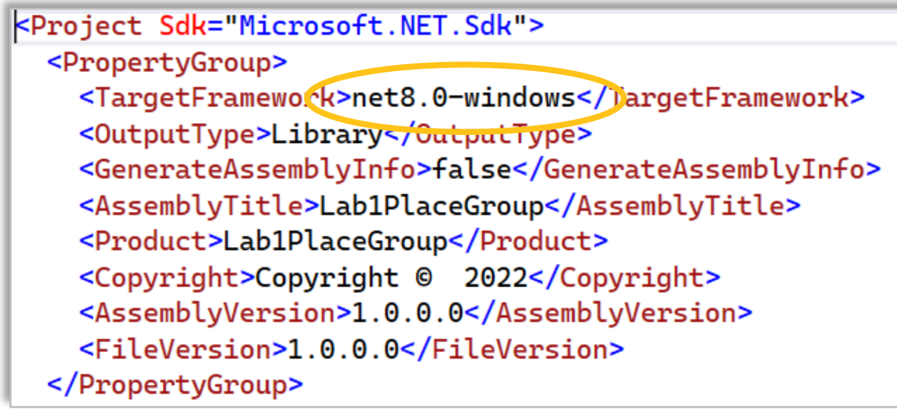
.NET 8.0

Install other frameworks

Target OS

Specifies the operating system that this pr

Windows



```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFramework>net8.0-windows</TargetFramework>
    <OutputType>Library</OutputType>
    <GenerateAssemblyInfo>false</GenerateAssemblyInfo>
    <AssemblyTitle>Lab1PlaceGroup</AssemblyTitle>
    <Product>Lab1PlaceGroup</Product>
    <Copyright>Copyright ©  2022</Copyright>
    <AssemblyVersion>1.0.0.0</AssemblyVersion>
    <FileVersion>1.0.0.0</FileVersion>
  </PropertyGroup>
```

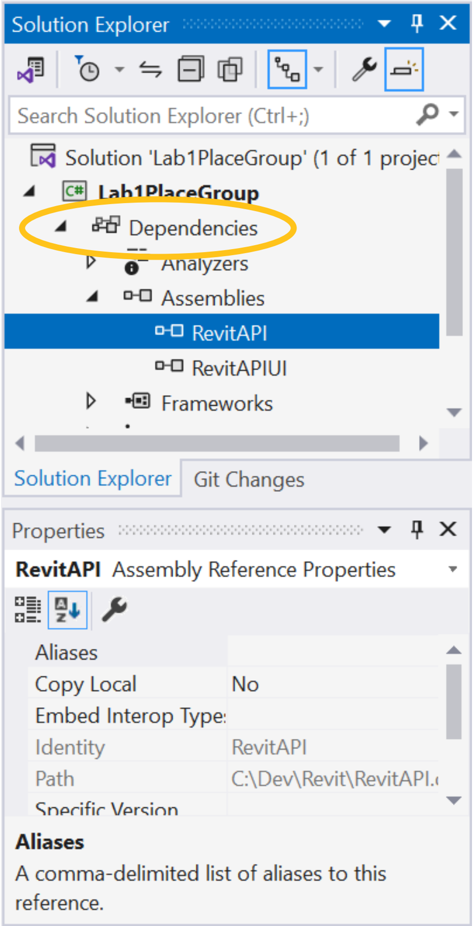
Migration Steps – C# projects (csproj)

1. Convert C# projects to new SDK style format
   - Use **.NET Upgrade Assistant**

2. Update target framework to .NET 8.0-Windows <TargetFramework>**net8.0-windows**</TargetFramework>
   - WPF – add **<UseWPF>true</UseWPF>**
   - WinForms – add **<UseWindowsForms>true</UseWindowsForms>**

3. Update references. System references can be removed. (default in .NET 8)

4. Address incompatible packages, library references, obsolete codes



RevitAPI Assembly Reference Properties

Aliases
Copy Local No
Embed Interop Type:
Identity RevitAPI
Path C:\Dev\Revit\RevitAPI.
Specific Version

Aliases
A comma-delimited list of aliases to this reference.

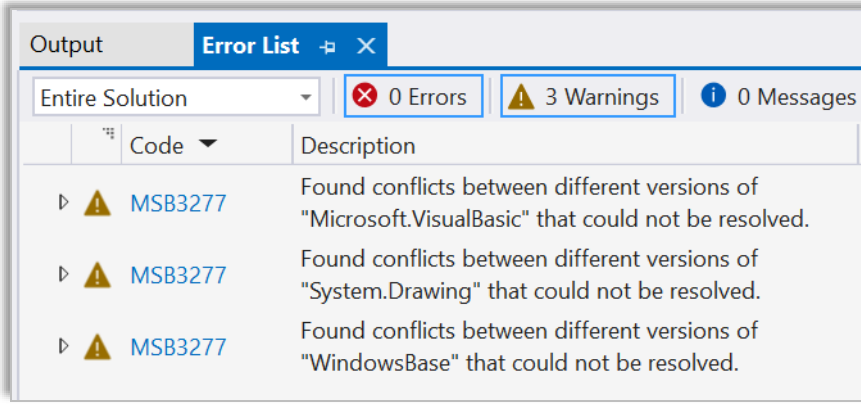
Migration Steps – C# projects (csproj)

1. Convert C# projects to new SDK style format
   - Use **.NET Upgrade Assistant**

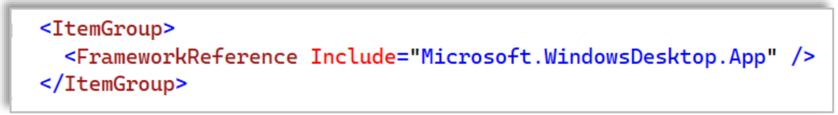
2. Update target framework to .NET 8.0-Windows <TargetFramework>**net8.0-windows**</TargetFramework>
   - WPF – add **<UseWPF>true</UseWPF>**
   - WinForms – add **<UseWindowsForms>true</UseWindowsForms>**

3. Update references. System references can be removed. (default in .NET 8)

4. Address incompatible packages, library references, obsolete codes



Code	Description
MSB3277	Found conflicts between different versions of "Microsoft.VisualBasic" that could not be resolved.
MSB3277	Found conflicts between different versions of "System.Drawing" that could not be resolved.
MSB3277	Found conflicts between different versions of "WindowsBase" that could not be resolved.



```
<ItemGroup>
  <FrameworkReference Include="Microsoft.WindowsDesktop.App" />
</ItemGroup>
```

Migration Steps – C# projects (csproj)

- Let’s see this in action using a MyFirstPlugin Revit sample

Common Issues

- Build warning MSB3277
    - When building the code referencing RevitAPI.dll, RevitUIAPI.dll or AcDbMgd.dll or AcCoreMgd.dll
    - Fix: add a reference to the Windows Desktop framework:

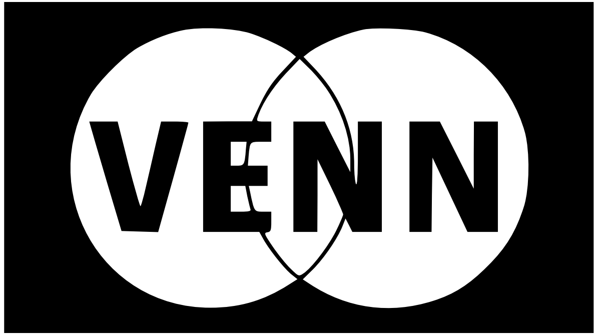
      ```
      <ItemGroup>
          <FrameworkReference Include="Microsoft.WindowsDesktop.App" />
        </ItemGroup>
      ```

- Build warning CA1416
    - When using Windows system only functions
    - Fix: add the following line to AssemblyInfo.cs:
      `[assembly:System.Runtime.Versioning.SupportedOSPlatformAttribute("windows")]`

AutoCAD

Next Release of AutoCAD & Civil API

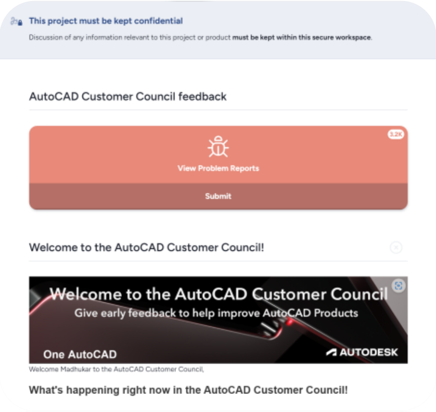
- **Binary Breaking Release**
  - Major version number is changed – R25.0
  - Visual Studio 2022
    - Visual Studio version: 17.8 or later
    - .NET version: ~~.NET Framework 4.8~~ .NET 8.0
    - Platform Toolset version: 143
- **Targeting the New AutoCAD:**
  - A Note for Appstore developers
  - Set SeriesMin and SeriesMax to R25 in Runtime Requirements
    - PackageContents.xml to support Autoloader



- **Update any source code that explicitly references the registry, interface versions.**
  - Examples:
    - ::CLSIDFromProgID(L"AutoCAD.Application.25", &clsid),

AutoCAD Plugin Prerequisites

- Register for AutoCAD Customer Council
  - ○ https://feedback.autodesk.com/project/home.html
  - ○ Find `AutoCAD Customer Council`
  - ○ Go to `Downloads` from Breadcrumb for SDKS\ NuGet
- AutoCAD NuGet Packages for Beta
  - ○ Required only if migrating .NET Projects.
- Object ARX SDK for Beta
  - ○ Required for migrating
    - C++,
    - C++\CLI and,
    - .NET Projects



This project must be kept confidential

Discussion of any information relevant to this project or product **must be kept within this secure workspace**.

AutoCAD Customer Council feedback

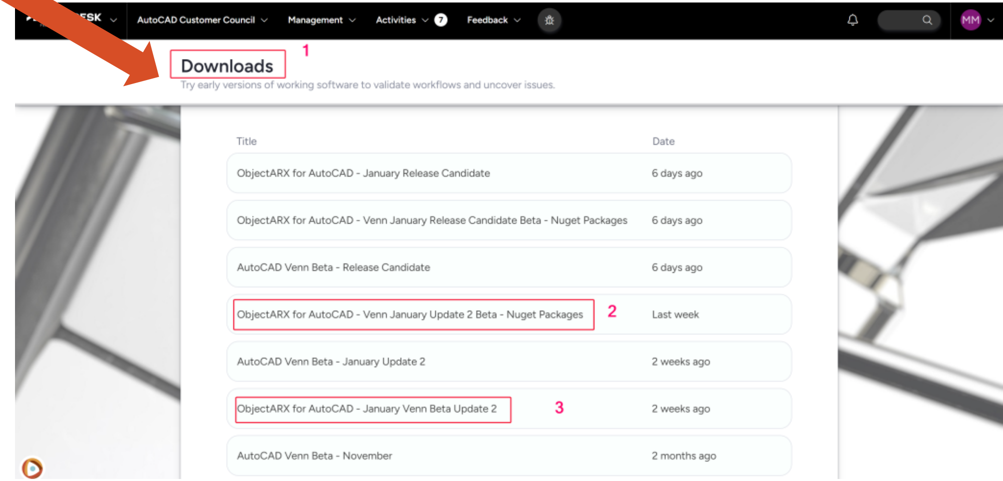
3.2K
View Problem Reports
Submit

Welcome to the AutoCAD Customer Council!

Welcome to the AutoCAD Customer Council
Give early feedback to help improve AutoCAD Products
One AutoCAD
AUTODESK

Welcome Madhukar to the AutoCAD Customer Council,

What's happening right now in the AutoCAD Customer Council!



Downloads

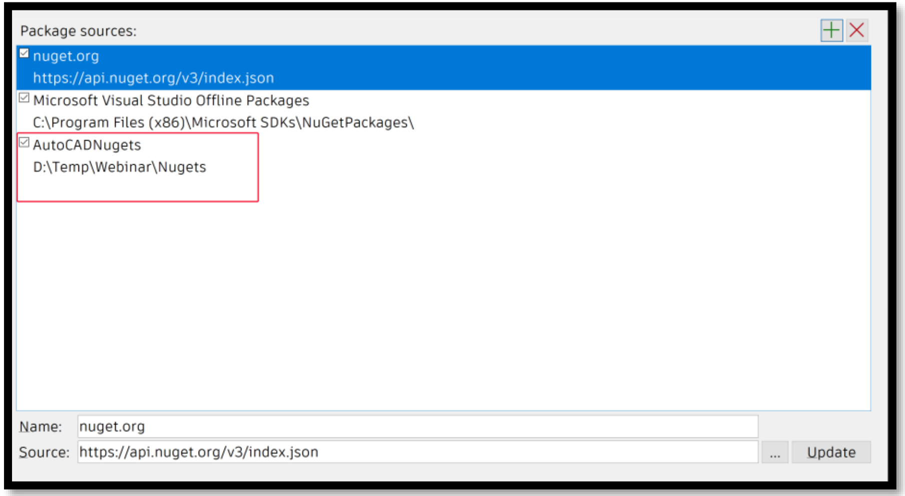
Try early versions of working software to validate workflows and uncover issues.

Title	Date
ObjectARX for AutoCAD - January Release Candidate	6 days ago
ObjectARX for AutoCAD - Venn January Release Candidate Beta - Nuget Packages	6 days ago
AutoCAD Venn Beta - Release Candidate	6 days ago
ObjectARX for AutoCAD - Venn January Update 2 Beta - Nuget Packages	Last week
AutoCAD Venn Beta - January Update 2	2 weeks ago
ObjectARX for AutoCAD - January Venn Beta Update 2	2 weeks ago
AutoCAD Venn Beta - November	2 months ago

Migration Steps – C# projects (csproj)

Install and manage packages in Visual Studio using the NuGet Package Manager

- Add Local Package source
- Set Package source to \`All\`
- Upgrade
- Let’s see this in action
  - TabExtension AutoCAD sample



Package sources:

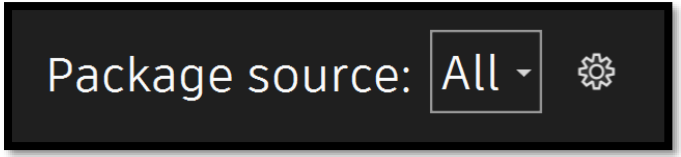
☑ nuget.org
https://api.nuget.org/v3/index.json

☑ Microsoft Visual Studio Offline Packages
C:\Program Files (x86)\Microsoft SDKs\NuGetPackages\

☑ AutoCADNugets
D:\Temp\Webinar\Nugets

Name: nuget.org

Source: https://api.nuget.org/v3/index.json



Package source: All

Example: MgdDbg (AutoCAD snoop tool)

Changes needed:

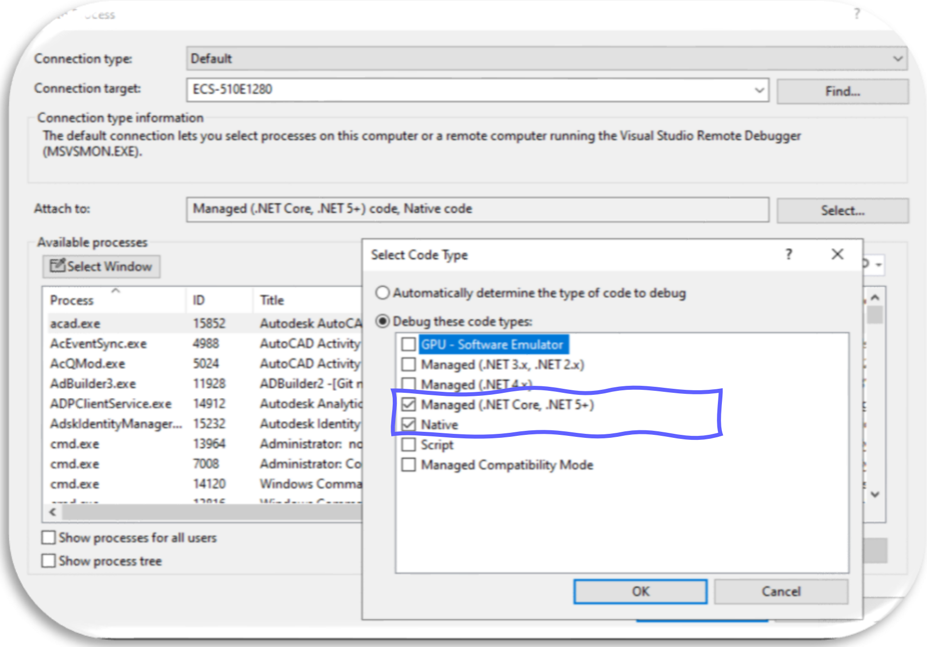
.NET Framework	.NET 8.0
ContextMenu	ContextMenuStrip
MenuItem	ToolStripMenuItem
Thread.Abort	CancellationToken
BinaryFormatter	EnableUnsafeBinaryFormatterSerialization=true https://github.com/dotnet/winforms/issues/9701

How to Debug

- In Visual Studio, select Debug > Attach to Process (or press Ctrl+Alt+P) to open the Attach to Process dialog box.
- Select the code types
  - Managed (Core CLR) - .NET Core
  - Native - C/C++

C++, C++\CLI & .NET Core

Pure .NET Core Plugin



Connection type: Default

Connection target: ECS-510E1280

Connection type information

The default connection lets you select processes on this computer or a remote computer running the Visual Studio Remote Debugger (MSVSMON.EXE).

Attach to: Managed (.NET Core, .NET 5+) code, Native code

Available processes

Process	ID	Title
acad.exe	15852	Autodesk AutoCA
AcEventSync.exe	4988	AutoCAD Activity
AcQMod.exe	5024	AutoCAD Activity
AdBuilder3.exe	11928	ADBuilder2 -[Git r
ADPClientService.exe	14912	Autodesk Analytic
AdskIdentityManager...	15232	Autodesk Identity
cmd.exe	13964	Administrator: nc
cmd.exe	7008	Administrator: Co
cmd.exe	14120	Windows Comma

□ Show processes for all users

□ Show process tree

Select Code Type

○ Automatically determine the type of code to debug

● Debug these code types:

□ GPU - Software Emulator

□ Managed (.NET 3.x, .NET 2.x)

□ Managed (.NET 4.x)

☑ Managed (.NET Core, .NET 5+)

☑ Native

□ Script

□ Managed Compatibility Mode

Revit

Component Versions (Revit January Preview)

- CefSharp
  - cef.redist.x64 and .x86 - version 120.1.10
  - CefSharp.Common.NetCore, CefSharp.Wpf.Hwndhost, CefSharp.Wpf.NetCore – version 120.1.110

cef.redist.x64

CefSharp.Common

- Newtonsoft.Json – version 13.0.1

Newtonsoft.Json

Disclaimer: May change in final version

Examples

- SDK Samples – will be migrated. Work in progress.

Additional error/warning we saw, e.g.,

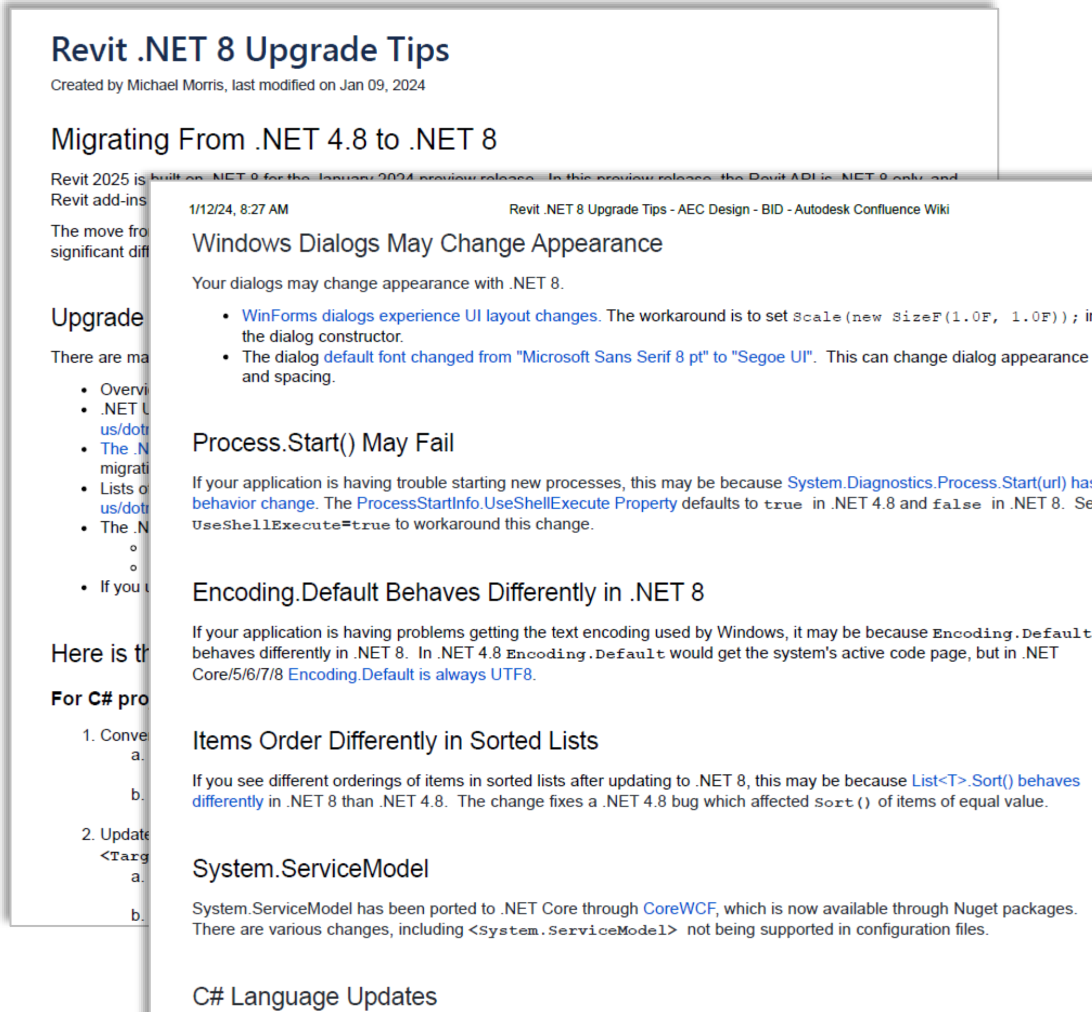
- if AssemblyInfo.cs file is not generated, set
  `<GenerateAssemblyInfo>true</GenerateAssemblyInfo>`

- Serializer/Deserializer – no more `using System.Web.Script.Serialization;`
  → `using Newtonsoft.Json;` Nuget Package

  ```
  e.g.,
  var jsonString = JsonConvert.SerializeObject(name);
  ```

More Tips

- Revit Preview
  >> User Forums
  >> API General
  >> "Enhancement Update – Revit .NET 8 Preview Release API"



Revit .NET 8 Upgrade Tips

Created by Michael Morris, last modified on Jan 09, 2024

Migrating From .NET 4.8 to .NET 8

Revit 2025 is built on .NET 8 for the January 2024 preview release. In this preview release, the Revit API is .NET 8 only, and
Revit add-ins

The move fro
significant diff

Upgrade

There are ma

- Overvi
- .NET U
  us/dotr
- The .N
  migrati
- Lists o
  us/dotr
- The .N
  -
  -
- If you

Here is th

For C# pro

1. Conve
   a.
   b.
2. Update
   <Targ
   a.
   b.

1/12/24, 8:27 AM Revit .NET 8 Upgrade Tips - AEC Design - BID - Autodesk Confluence Wiki

Windows Dialogs May Change Appearance

Your dialogs may change appearance with .NET 8.

- WinForms dialogs experience UI layout changes. The workaround is to set `Scale(new SizeF(1.0F, 1.0F));` in the dialog constructor.
- The dialog default font changed from "Microsoft Sans Serif 8 pt" to "Segoe UI". This can change dialog appearance and spacing.

Process.Start() May Fail

If your application is having trouble starting new processes, this may be because System.Diagnostics.Process.Start(url) has behavior change. The ProcessStartInfo.UseShellExecute Property defaults to `true` in .NET 4.8 and `false` in .NET 8. Se `UseShellExecute=true` to workaround this change.

Encoding.Default Behaves Differently in .NET 8

If your application is having problems getting the text encoding used by Windows, it may be because `Encoding.Default` behaves differently in .NET 8. In .NET 4.8 `Encoding.Default` would get the system's active code page, but in .NET Core/5/6/7/8 Encoding.Default is always UTF8.

Items Order Differently in Sorted Lists

If you see different orderings of items in sorted lists after updating to .NET 8, this may be because List<T>.Sort() behaves differently in .NET 8 than .NET 4.8. The change fixes a .NET 4.8 bug which affected `Sort()` of items of equal value.

System.ServiceModel

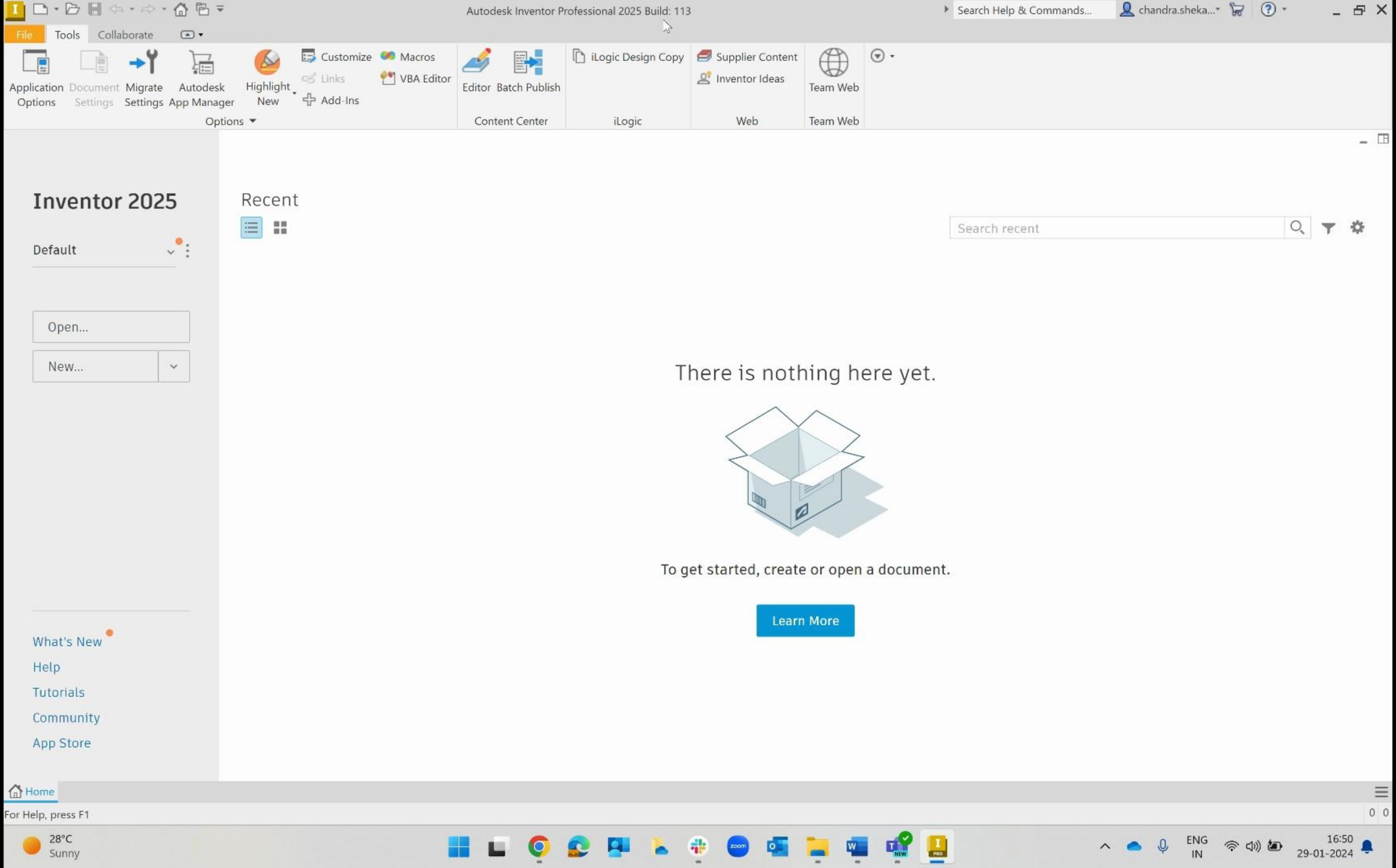
System.ServiceModel has been ported to .NET Core through CoreWCF, which is now available through Nuget packages. There are various changes, including `<System.ServiceModel>` not being supported in configuration files.

C# Language Updates

Inventor

Upgrade process for C# add-ins

- Re-reference: “**Autodesk.Inventor.interop.dll**” from **C:\Program Files\Autodesk\Inventor 2025\Bin\Public Assemblies** folder and set the **“Embed Interop Types”** to No, **“Copy Local”** to Yes.
- Edit the InventorAddinSampl.vbproj in text editor and edit **post build script.**
- iLogic –no change. If iLogic API is accessed through exteranl application (like .exe, add-in), Need to upgrade to .NET 8.0.
- Deploy the add-in at **“C:\Program Files\Autodesk\Inventor 2025\Bin\Addins\”** folder. and compiled binary at **“C:\Program Files\Autodesk\Inventor 2025\Bin\”**



There is nothing here yet.

To get started, create or open a document.

For Help, press F1

Other products

Other Desktop Products API

- **Vault**
  - Add-ins currently supports .NET 8.0
  - Two SDK available for .NET 8.0 and .NET Framework
- **ReCapSDK**
  - Migrated to .NET 8.0
- **Navisworks**
  - Remain .NET 4.8 (many dependencies that are not available in .NET 8.0 currently)

Q & A

Desktop Products – migration guide/discussion

feedback.autodesk.com >> join and look for opportunity >> **User Forums**

AutoCAD Customer Council

>> Forums >> ObjectARX API >> *"How to Create, Build and Debug .NET 8 Plugin for AutoCAD Venn"* (recording)

Revit Preview

>> Forums >> API General >> *"Enhancement Update - Revit .NET 8 Preview Release API"*

Factory Design Utilities/Inventor

>> Forums >> iLogic, API and Automation >> *".Net Framework based Inventor projects upgrade to .Net 8"*

Infrastructure Futures

>> Forums >> Autodesk Civil 3D 2025

Recording of this webinar will be posted in feedback portal and ADN site.

Keep the discussion within the feedback portal.

Thank you!

AUTODESK