Building the ASCOM Platform

# Last Tested

This script was last tested on 7th February 2025 while installing into a brand new Windows 11 install running as a new VM under VMWare Workstation 17.6.2.

# Prepare the Build Environment

A 64bit Windows OS version 11 or later is required for the build environment.

This build process was validated in a fully updated VMWare VM running Windows 11 64 bit. Please install the following components to create the ASCOM Platform Windows build environment.

Never install the ASCOM Platform in this instance. The purpose of this is to ensure that only project references are used within the solution and that there are no unintended dependencies on pre-installed ASCOM components.

## Windows

* Make sure that .NET 3.5 support is enabled on the PC. This must be turned on manually in Windows 10/11 through:
  + Win 10: the “Turn Windows features on and off” option in “Uninstall or change a program”.
  + Win 11: the “Additional features " link at the bottom of the “Enable features” menu.

## VS 2019

A VS 2019 install is required only in order to get the .NET 3.5 and 4.0 targeting packs.

* Download the Community VS2019 installer.
  + If not available from Microsoft it is available here: <https://www.techspot.com/downloads/7241-visual-studio-2019.html>
* Install the editor with no workloads after selecting these additional individual components:
  + .NET Framework 3.5 development tools
  + .NET Framework 4 targeting pack

## VS 2022

* Download the Visual Studio 2022 Community installer from [https://www.visualstudio.com](https://www.visualstudio.com/)
* Select these workloads:
  + .NET Desktop Development
  + Desktop Development with C++” workloads
  + Visual Studio Extension Development
* Select these Individual Components
  + C++/CLI support for v143 build tools (Latest)
  + MSVC v143 – VS 2022 ARM64/ARM64EC build tools (latest)
* Launch VS2022 and sign in with your Microsoft account.
* Close VS2022

## Notepad ++

* Download from <https://notepad-plus-plus.org>
* Run the installer and take default options.

## Git For Windows

* Download from <https://git-scm.com/download/win>
* Run the installer and select Notepad ++ on the default editor selection page
* Take other options at default

## GitHub Desktop

* Download from <https://desktop.github.com/>
* Run the installer with default options
* Sign in to GitHub.com with your GitHub credentials
* Close GitHub Desktop

## InstallAware X15

InstallAware is a purchased product and is only licensed for use by one developer at a time. However, our “Studio” license entitles us to install as many “build machines” as required. The components below represent a build machine install and no other IA components should be installed.

* Request the installer from Bob Denny or Peter Simpson who will also supply the access password.
* Run the installer and select "Studio" (not “Studio Admin”) as the product type.
* Select “Let me choose which features to install”
* Deselect the following items
  + App-V Builder
  + App-V Viewer
  + Database import wizard
  + Database validator
  + Digital Rights Management
  + DiskZip
  + Group Policy Wizard
  + Inline Update
  + MSIZ Builder
  + Open Source Desktop Bridge
  + Setup Decompressor
  + Sounds
  + Sample Projects
  + Team Foundation Server Integration
  + Transformer
  + Visual Studio Add-in
* Keep all “Plug-ins” but deselect all “Plug-ins\Runtimes” except for:
  + "Windows installer"
  + "Windows installer (x64)"
* Click “Next” several times to start installation

## MS Build Community Tasks

* Download the version 1.5 MSI installer from <https://github.com/loresoft/msbuildtasks/releases>
* Run the installer with default options.

## ASCOM Platform Build Script

* Create the directory C:\ASCOM Build on the build machine.
* Copy these files in the Repository “Build Process” folder to the C:\ASCOM Build folder.
  + BuildPlatform.msbuild
  + build.cmd
  + nuget.exe
* Edit the:
  + <GitCheckoutBranch> variable near the head of the BuildPlatform.msbuild file to select the develop or main branch as appropriate.
  + BuildType PropertyGroup content to be Release or ReleaseCandidate as appropriate.

## Sandcastle Help File Builder

* Download the guided installer from <https://github.com/EWSoftware/SHFB> making sure that the version matches the version last used to edit the Help file sources.
* Run the installer:
  + Install the HTML Help 1 Compiler from the MS web site when prompted, Ignore the message about already having a newer version of the Help
  + Go back to the SHFB installer and click the “Search again” button on the HTML Help 1 page, which will now find the HTML Help 1 compiler.
  + Click the “Next” button to get to the Sandcastle Help File Builder and Tools page and install SHFB.
  + Click next and install the Visual Studio packages for VS2019 and VS2022.
  + Don’t install the MAML snippets.
  + Move through remaining dialogues and close the installer script.
* Reboot the PC or VM.

## VSIX Sign Tool

* Install the dotnet “Sign” tool as described here: <https://learn.microsoft.com/en-us/visualstudio/extensibility/signing-vsix-packages?view=vs-2022>

## Code Signing Certificate – USB Key

ASCOM currently uses a personal code signing certificate in the name of “Peter Simpson”

* Install the management software for your EV token e.g. SafeNet for Thales hardware tokens.
* Configure the client to only prompt once for the password (otherwise you will have to enter the token password over 200 times per build! 😁)
* Create a C:\CodeSigningCertificate folder and copy your EV certificate here.
* Ensure that the InstallAware Authenticode settings in Project Options / Build are correctly set including:
  + The “Sign the package with Authenticode” checkbox is checked
  + The filename and path to your EV certificate (.CER file) are correct.
  + The “Key Container Name” of your EV certificate USB key is correct.
* Update the <SignToolBase256> tag in the BuildPlatform.msbuild script to use the individual’s name to whom the certificate was issued.
* Update the <VSIXSignCommand> tag with the:
  + Full path and file name of the certificate file
  + The certificate fingerprint which you can find with the Powershell command: Get-FileHash -Algorithm SHA256 <path to .cer file> | Format-Table -AutoSize
  + The cryptographic service provider name and container name for your USB token. These can usually be found through your e-token management tool. E.g.  
    A screenshot of a computer

    AI-generated content may be incorrect.

# Tailing the Build Log

It is optional but useful to monitor build progress by tailing the build log. This can be achieved using a PowerShell script that should be run before starting the build. The script opens a PowerShell command window and displays the contents of the build log as it grows.

To install the script:

* Copy the TailBuildLog.ps1 file in the Repository “Build Process” folder to the Desktop.
* Start a PowerShell prompt as administrator
* Enter the command: Get-ExecutionPolicy -List  
  A screen shot of a computer

  AI-generated content may be incorrect.
* If the CurrentUser setting is not “AllSigned” or “Unrestricted”, set it to “AllSigned” with the command:   
  Set-ExecutionPolicy -ExecutionPolicy AllSigned -Scope CurrentUser

Now run the script by right clicking it and selecting Run with PowerShell.

## Notes

* The BuildLog.txt must exist before the script will run
* You will need to approve the script to “Always run” on first use.  
  A screenshot of a computer

  AI-generated content may be incorrect.

# How to Build the ASCOM Platform

## Process

* Open a Visual Studio 64bit command prompt
* Change directory to C:\ASCOM Build
* Run the Build.cmd command file. (See note 1)
* At the end of a successful build the Platform installers, other installable or publishable components and the build log will be found in the C:\ASCOM Build folder.

# Notes

1. I had to change the Windows Network type to “private” from the default “public” to avoid GIT clone reporting a network error
2. I had to install the out of support .NET 6 and .NET 7 support packages

# Document History

* V0.1 - Original from 2012 - Updated periodically after that but retaining the same version number
* V1.0 - 8/2/25
  + Tested the full build process on a fresh Windows 11 instance.
  + Updated this document where necessary and added further detail.
  + Updated code signing to use USB based EV certificate tokens.