

# Build and deploy the DLL

The example described in this section was implemented using Microsoft Visual Studio with an output type of *Class Library* and a target Framework using *.Net Standard 2.0*.

## Important

DLLs integrated with Aras Innovator need to be signed.

The DLL must also be compiled by linking with the `Aras.DynamicModelViewer.DataModel` and `Aras.Server.Core` libraries which are included with the Dynamic Model View Install and with the standard Aras Innovator installation respectively. Note also that the Rendering Configuration classes use the `Color` class as defined in `System.Drawing`. The resulting DLL needs to be stored within the `< Aras Innovator Install Dir>/server/bin` directory. Once the DLL is stored, update the `method-config.xml` file in the `<Aras Innovator Install Dir>/server` directory to include a reference to the DLL. See the following figure for an example.

```
<MethodConfig>
  <ReferencedAssemblies>
    ...
    <name>$(binpath) /Aras.DynamicModelViewer.DataModel.dll</name>
    <name>$(binpath) /Aras.DynamicModelViewer.QueryProcessor.dll</name>
    <name>$(binpath) /CustomDefaultQP.dll</name>
    <name>$(binpath) /CustomPartQP.dll</name>
  </ReferencedAssemblies>
```

## Important

Due to the differences between Windows and Linux file systems it is required to use OS-specific path separator in paths. Remember that the Linux file system is case sensitive so there is significant difference between file names `./path/to/file.xml` and `./path/to/File.xml`. For more information about cross-platform development, see section 2.3 Cross-platform development in the Aras Innovator 31 - Programmer's Guide.

