# Spatial Analyzer SDK

Visual Studio C++ v16.32 Example

Install the latest version of Spatial Analyzer.

The Spatial Analyzer SDK provides a means by which to write custom applications that utilize Measurement Plan functionality within Visual C++. Once the SA SDK engine has been added to the development environment, Measurement Plan Steps can be executed within the programming language.

Step 1: Adding the SDK to the Visual Studio Application

Create a new C++/MFC project within Visual Studio 2019
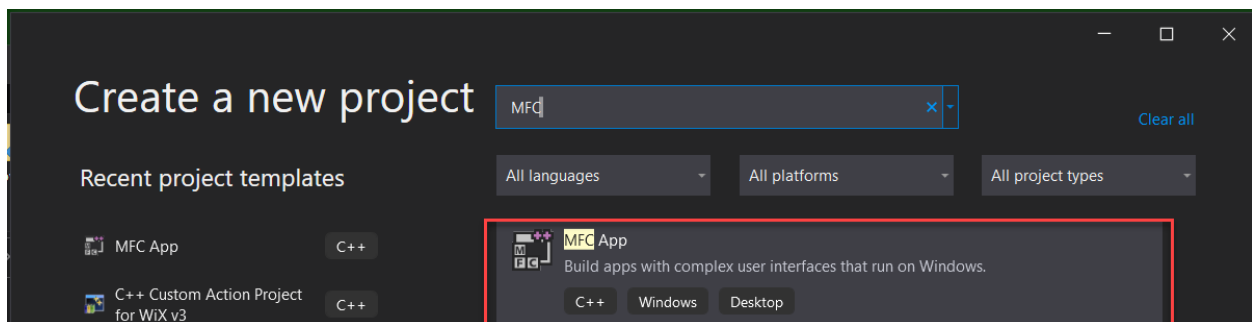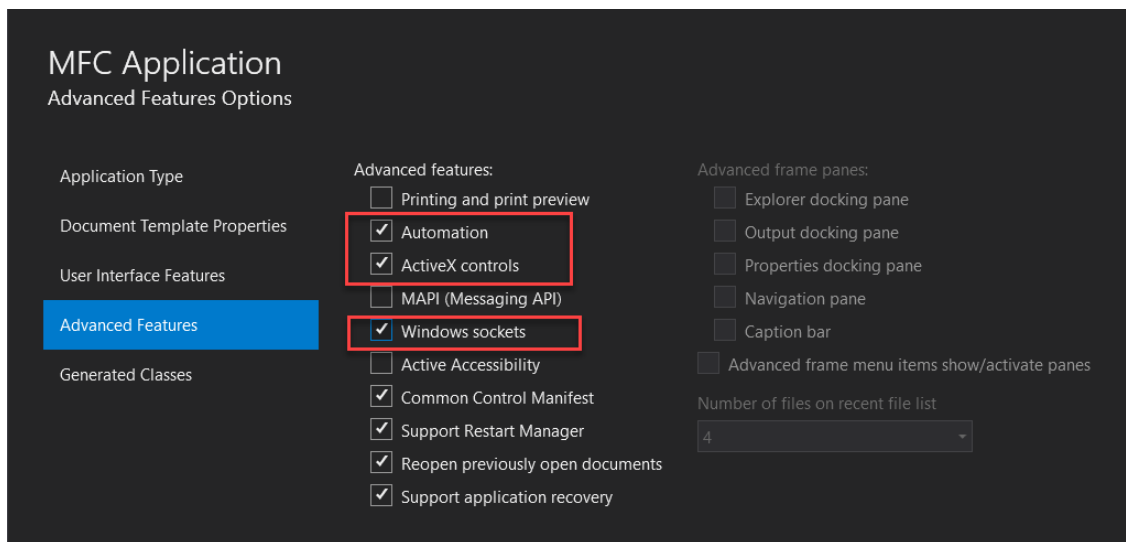

Step 2: Name New Project

Important: Name your new project using the Microsoft naming convention rules. **DO NOT** use underscores, hyphens, or any other nonalphanumeric characters in the project name.

https://learn.microsoft.com/en-us/dotnet/standard/design-guidelines/general-naming-conventions
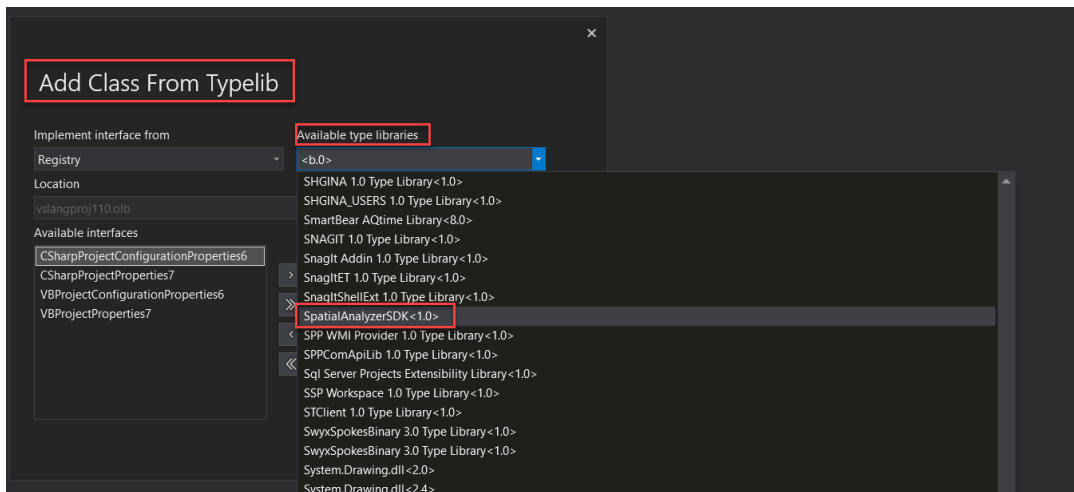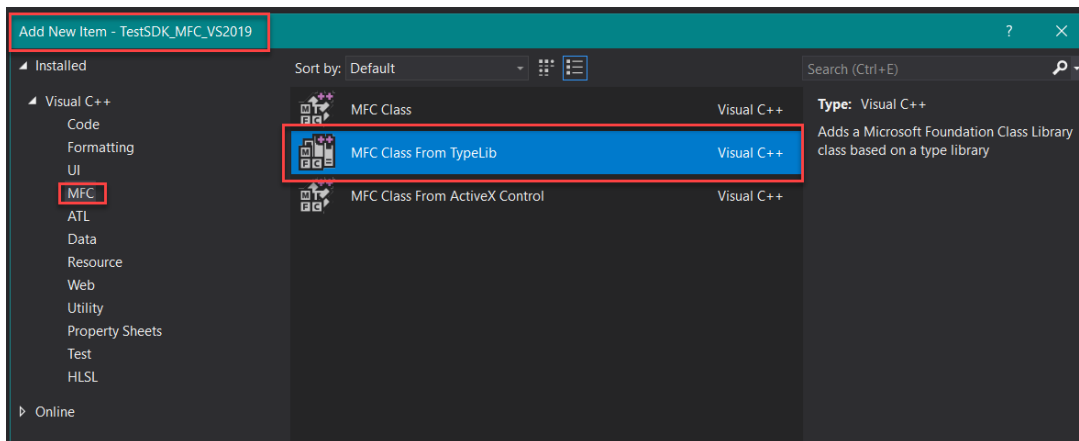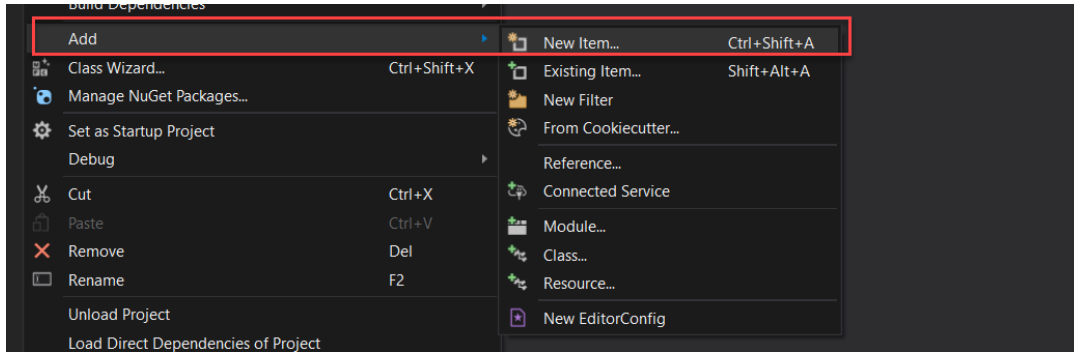
Step 3. Set Application Type Options
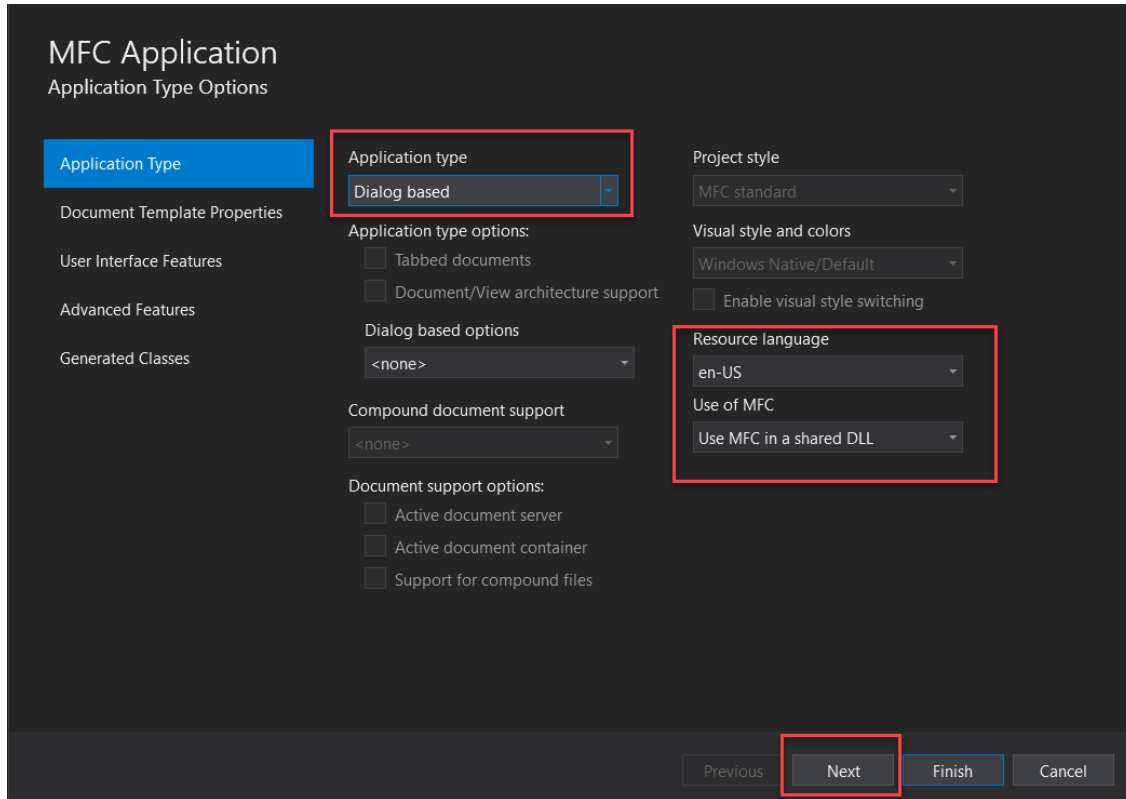
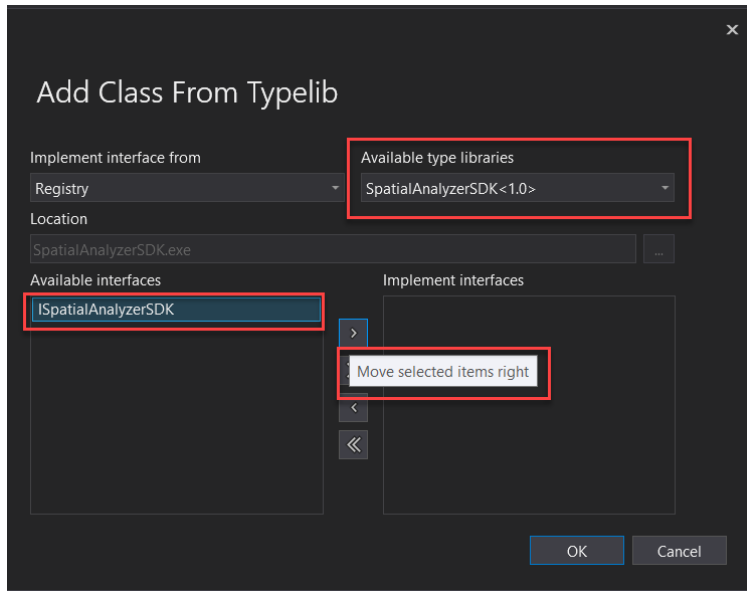Press "Next" to configure the Advanced Features

Press "Finish" to open the new project in the Visual Studio.

Step 4. Add SA SDK to the Project

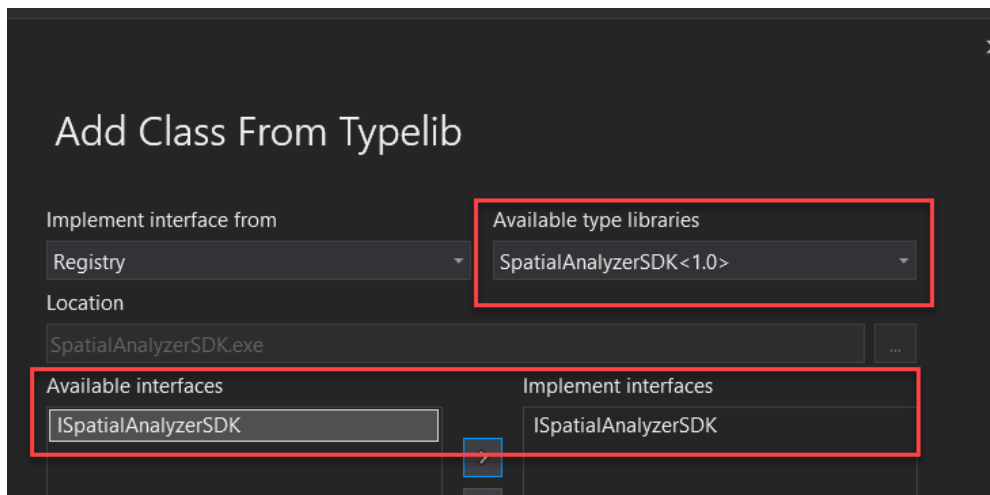From the Solution Explorer, right mouse click on the project name within the tree and select

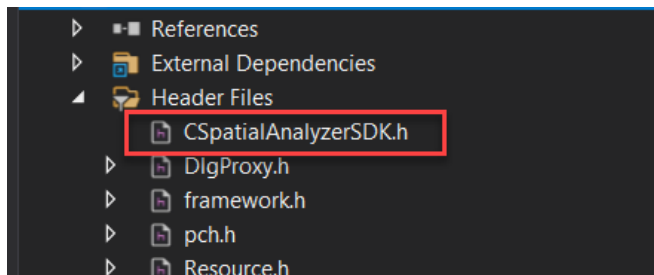
Select the **SpatialAnalyzerSDK<1.0>** from the **Available type libraries** drop-down.

Then press the **>>** button to add *ISpatialAnalyzerSDK* to the *Implement Interfaces* section.

Click "*OK*" button and check that in the project tree *CSpatialAnalyzerSDK.h* was automatically added

The "*CSpatialAnalyzerSDK.h*" contains all the SA SDK interfaces available in the current SA version.

SA SDK Code Examples
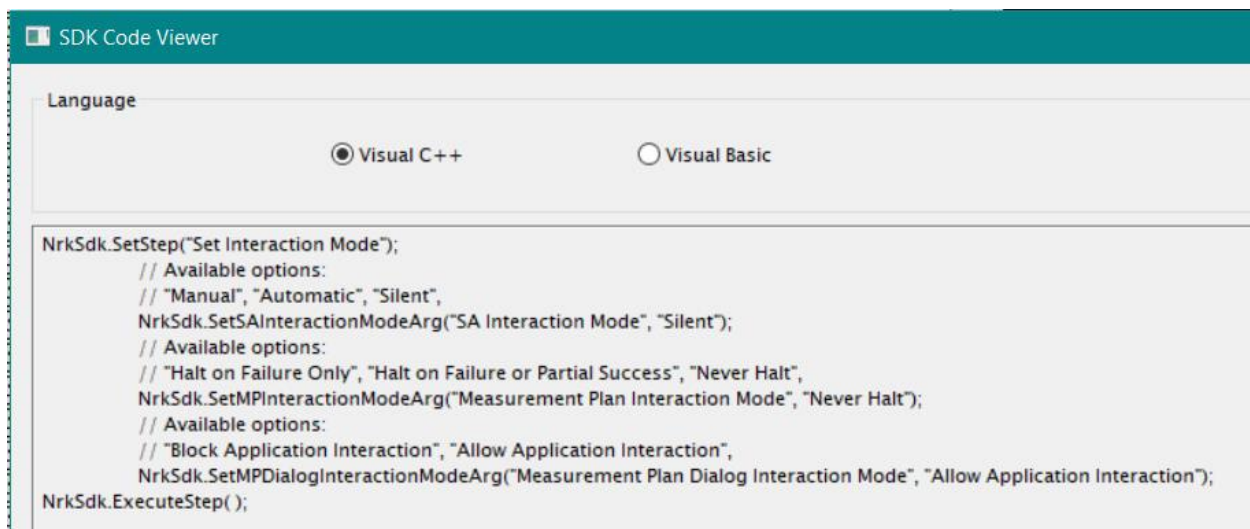
You can download SA SDK example projects written in C++, C#, and VB from the NRK FTP site:

https://www.kinematics.com/ftp/SA/Install/Driver%20Downloads/SA%20SDK/Examples/

Spatial Analyzer supports the generation of sample SA SDK code through the MP Editor.
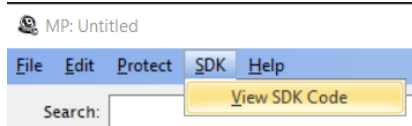
Setup an MP Step with the necessary arguments and press the code generation button to display sample code.

Generated code will automatically be placed on the Clipboard for pasting into a client application that is being developed.

The SA SDK Visual Studio 2019 C++ code example contains a helper class called **SDKHelper**. The SDKHelper class provides convenience functions when accessing arrays from a Visual C++ client application. When using the Spatial Analyzer code generation feature, the code generator will utilize this helper function in any code generated to Set/Get array data. Please refer to the Visual C++ sample client application source code for more details.

The updated SA SDK MFC C++ client application example demonstrates different MP commands use. Please check in the code comments if you need to load a specific SA job before executing the MP operation.
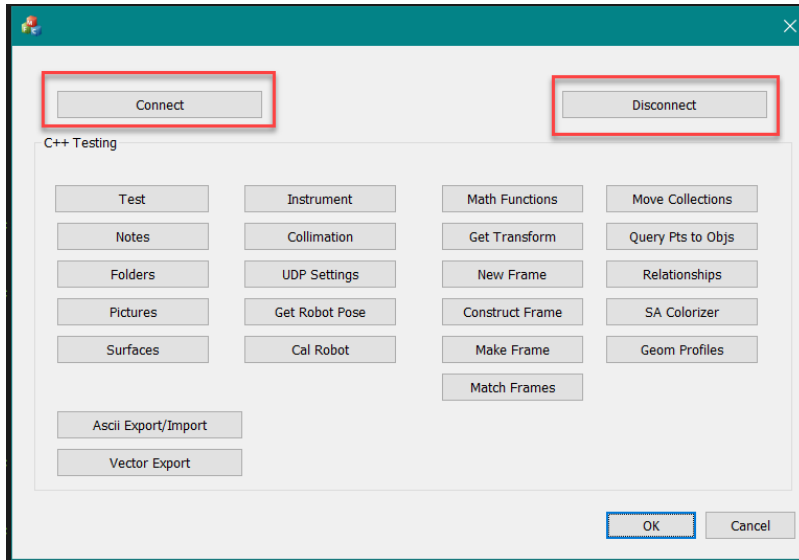
Important Rules

The SA SDK consists of a set of function calls that allow the specification and execution of MP Steps.

1. The client **must first** instruct the SDK engine **to connect to Spatial Analyzer** before performing any MP Step operations.

2. Once finished, the client application must **disconnect properly** to release the system resources.

3. Once connected, you can see in the task bar an icon

4. Click on icon to maximize the SA SDK engine application to check the SA SDK version and the connection status

5. The string names specified for the step name and its corresponding arguments need to **exactly** match the SA MP **Text strings in English**.

6. The return status of executing any MP command can be checked.

7. If an MP contains a "Result Only" argument, then the result argument can be retrieved via calling the appropriate **Get** function call.