# Simio API Note: Simio API Helper

May 2018 (Dhouck)

Contents

[Simio API Note: Simio API Helper 1](#_Toc31191445)

[Overview 2](#_Toc31191446)

[Using The API Helper. 3](#_Toc31191447)

[Tab DLL Helper 4](#_Toc31191448)

[Tab .Net Versions 5](#_Toc31191449)

[Tab Find User Extensions 6](#_Toc31191450)

[Headless Workflow and Recommendations 7](#_Toc31191451)

[Tabs Headless Builder and Headless Run 8](#_Toc31191452)

[Installing the Simio API Helper 11](#_Toc31191453)

# Overview

This API Note describes a utility that can:

1. Examine/test the DLLs that are used by the APIs.
2. Construct sample “headless” folders
3. Test the operation of the headless methods.

This is a very brief note, as this utility was quickly assembled and being put in place to help a few Simio users debug some API issues.

The intent is that this helper will expand/extend as needed.

# Using The API Helper.

The Simio API Tester is not part of the Simio product. Rather it is simply a test tool that was built to help debug problems.

You can use it to verify that your DLLs are:

1. In the right place
2. Is accessible
3. Has the Interface implemented
4. Referencing other Assemblies correctly

The UI architecture is a simple WinForms tabs, and the sections below are organized according to those tabs.

## Tab DLL Helper

This tab is used to examine and load DLL assemblies.

The top drop-down displays the locations where Simio DLLs can be found, and the next drop-down then shows the DLL files within that location. The Exclude filter can be used to reduce the number of DLL files displayed.

Once a file is chosen, general information about the contents of the DLL is shown, along with the definitions found within the file. If you wish to see only Simio information, check the “Simio Only” checkbox.

A screenshot of a computer screen

Description automatically generated

## Tab .Net Versions

A screenshot of a social media post

Description automatically generated

## Tab Find User Extensions

A screenshot of a cell phone

Description automatically generated

## Headless Workflow and Recommendations

Achieving success when building a headless application often depends upon selecting and using the correct components (DLLs). It is a bit confusing because these DLLs – although delivered with your Simio product – are best moved to a totally separate and independent location where your executable resides.

This has the advantage of protecting your application (which are often production oriented) from updates or upgrades to the Simio software.

Additionally, when you are building your application (e.g. with Visual Studio) you should reference the DLLs from your headless environment, since this is what your generated executable will expect.

The diagram below illustrates this workflow.

A close up of a map

Description automatically generated

### Tabs Headless Builder and Headless Run

A screenshot of a social media post

Description automatically generated

### Headless Debugging

One of the hardest things to determine is what DLLs are required, and/or what the dependencies between the DLLs is.

There are two free tools that can help with this:

1. Process Explorer from SysInternals (Microsoft)
2. DotnetPeek from JetBrains

ProcessExplorer can be used to examine a running program. This is incredibly useful because we can see what DLLs are employed regardless of when they were loaded.

A screenshot of a computer

Description automatically generated

A

So, in the example above the included DLLs are shown.

Below is ProcessExplorer being applied against Simio with the same model being run.

A screenshot of a computer

Description automatically generated

And below is the result of one of the example projects “HeadlessFormsExperiment” which uses the call “System.Appdomain.CurrentDomain.BaseDirectory” to pick up the location of the HeadlessFormsExperiment.exe to locate all of the DLLs.

Tab Logs and Settings

# Installing the Simio API Helper

The helper is just an EXE (SimioApiHelper.exe) and is built with .NET 4.5.2.

There are no other dependencies (e.g. DLLs) beyond .NET.