|  |  |
| --- | --- |
|  | LAB | Dependency Injection with Mef2 and .NET Core |
|  | WORKSHOP | Design Patterns |
|  | Crystal Tenn  crtenn@microsoft.com |

# SETUP LAB ENVRIOMENT

## Prerequisites

To perform the tasks in this lab you need following:

* Visual Studio 2015 or 2017, any version

## Objectives

In this lab you will complete following tasks:

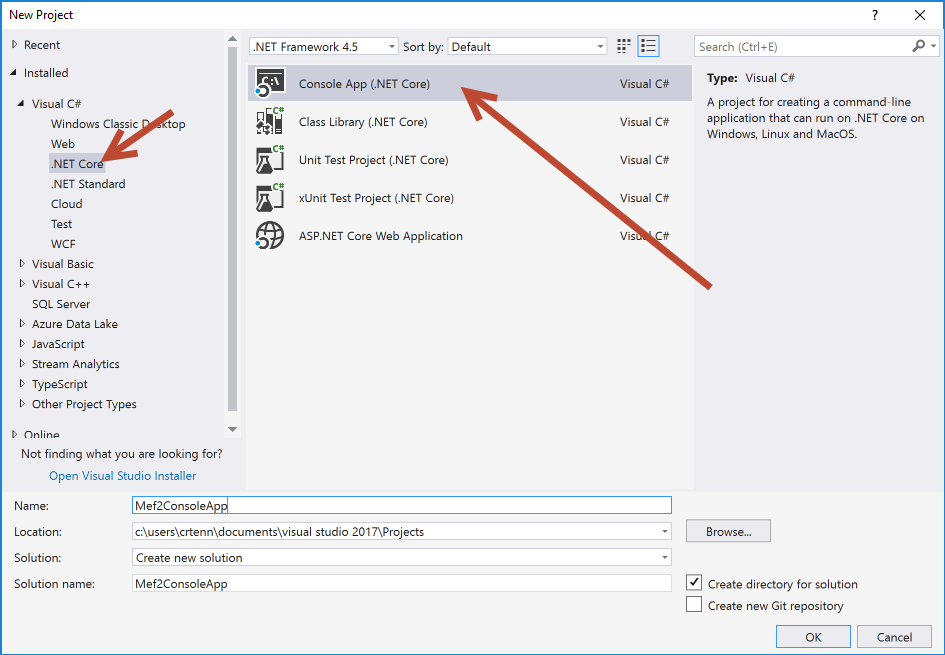
* Create a .NET Core application with Mef2

### Estimated Completion Time: 20 minutes.

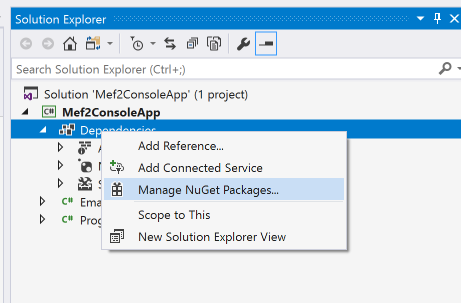
## Task: Create a Console Application

In this task you will create a new Console application.

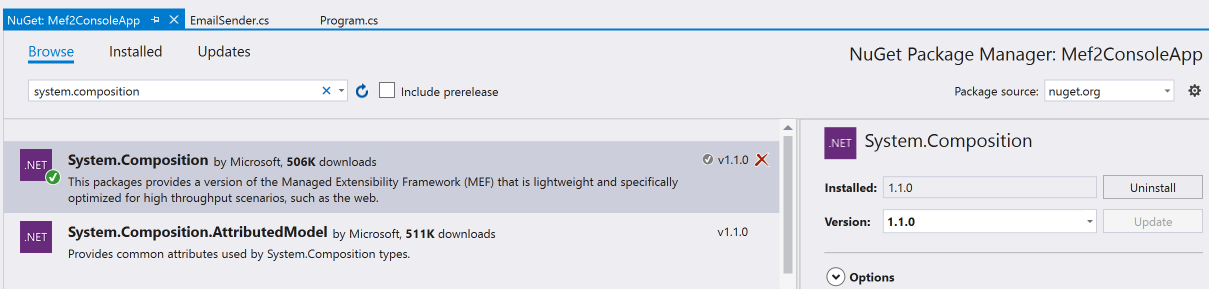
1. Open Visual Studio.
2. Choose New Project. On the left choose .NET Core. Then select a new Console App. Name it Mef2ConsoleAPp.



1. Right click dependencies, hit manage NuGet.



1. Hit Browse, search for System.composition. Hit Install. Once it is done, check that it exists in your dependencies.



1. Create a new interface called IMessageSender. Add the following code:

public interface IMessageSender

{

void Send(string message);

}

1. Create a new class called Email Sender.

[Export(typeof(IMessageSender))]

public class EmailSender : IMessageSender

{

public void Send(string message)

{

Console.WriteLine(message);

}

}

1. Add the following using statements to your EmailSender:

using System;

using System.Composition;

1. Add the following to your Program.cs class:

public void Run()

{

Compose();

MessageSender.Send("Hello MEF");

Console.ReadLine();

}

[Import]

public IMessageSender MessageSender { get; set; }

private void Compose()

{

var assemblies = new[] { typeof(Program).GetTypeInfo().Assembly };

var configuration = new ContainerConfiguration()

.WithAssembly(typeof(Program).GetTypeInfo().Assembly);

using (var container = configuration.CreateContainer())

{

MessageSender = container.GetExport<IMessageSender>();

}

}

1. Replace your Main method in Program.cs with the following:

public static void Main(string[] args)

{

Program p = new Program();

p.Run();

}

1. Add the following using statements to your Program.cs file:

using System;

using System.Composition;

using System.Composition.Hosting;

using System.Reflection;

1. Run the program. Drop a breakpoint and step through + understand the code. View the Compose method in Program.cs and see how the Containers are configured.

