2014

EDSP BV

EDSP BV Application
Management Services

www.edsp.nl

[.NET 4 VIRTUALIZATION WITH APP-V 5]

Recipe for sequencing the Microsoft .NET 4 installation with Microsoft App-V 5.0 SP2 HF5 for Windows 7 & Windows 8

Introduction:

Hi, my name is Jeroen Spaander, I work for EDSP and currently am working on multiple projects for Genmab.

One of the projects I was struggling with was the Electronic Laboratory Notebook (ELN) / Sample Management (SM) deployment build by Waters. The current ELN / SM installation consists of a couple of Microsoft Runtimes, a customized Oracle Client (don't ask me why), multiple setups for all of the ELN components and some extra components for the Sample Management module specifically designed for my client. It's a 65 step setup > 1GB.

Issue:

The Sample Management components have a dependency on .NET 4 and although we are running the .NET 4.5.1. version on our Windows 7 & 8 desktop environment (which should %100 replace the .NET 4 installation being very backward compatible and all), it is in fact not working for all the Sample Management functions.

Luckily I am not the only one encountering issues with .NET 4 and .NET 4.5.1. backward compatibility and there is already a lot said on the subject.

Possible solutions:

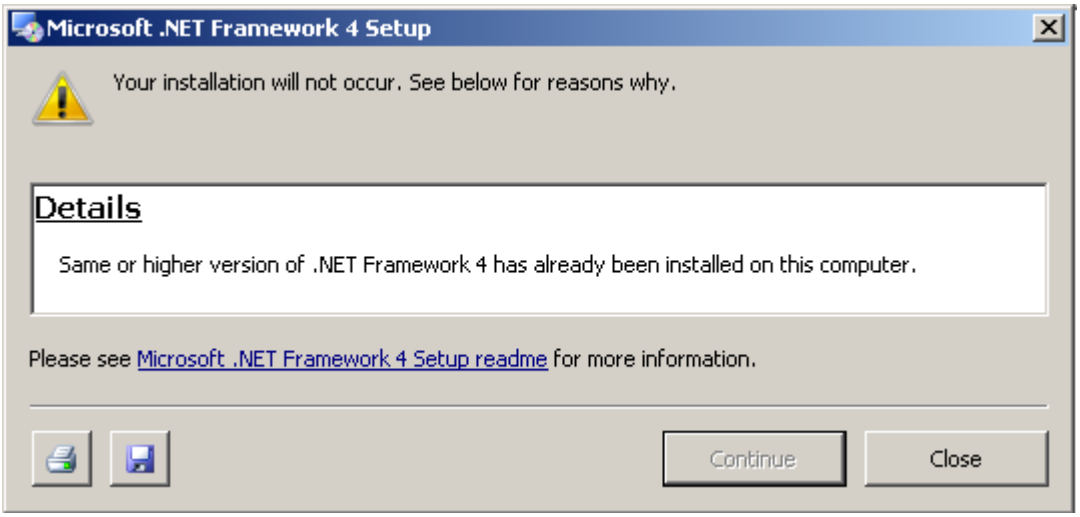
I've added a couple of the attempted solutions below:

Side by side installation of .NET 4 and .NET 4.5.1

Until the .NET 4 installation we could install all previous versions side by side next to each other so that of course was my first attempt to solve this issue.

The .NET 4 & .NET 4.5.1. in-place replacement install are however very different from the side by side installs of .NET 2.0 and 3.0/3.5. The two 3.x versions were basically library enhancements on top of the core .NET 2.0 runtime. The 4.5 update instead completely replaces the .NET 4.0 runtime and leaves the actual .NET version number set at v4.0.30319.

Trying to install the .NET 4 installation next to the already installed .NET 4.5.1 installation will return the following error on Windows 7: "Same or higher version of .NET Framework 4 has already been installed on this computer."

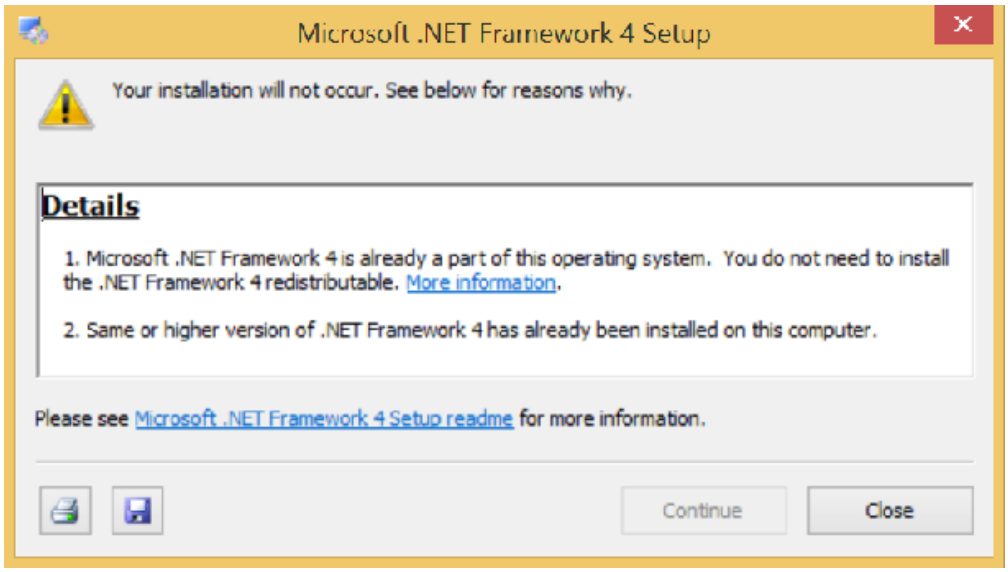


Trying to install the .NET 4 installation next to the already installed .NET 4.5.1 installation will return the following error on Windows 8:

1. Microsoft .NET Framework 4 is already a part of this operating system.

You do not need to install the .NET Framework 4 redistributable.

2. Same or higher version of .NET Framework 4 has already been installed on this computer.



Although this does seem to be the end of this attempt, it was however part of the solution to deliver .NET 4 as a virtualized installation.

Ask the vendor to fix his code:

Installing the required .NET version next to the latest .NET install wasn't possible and being fed up with wasting my clients money I contacted the vendor asking him to fix his code to use the new .NET 4.5.1. functions. Their reply wasn't very hopeful as it was a definite NO as they had to run their product in sync with other products and weren't planning an update or new release for some time for that reason.

Forcing the ELN.exe to use the .NET 4 Runtime:

The vendor wasn't going to change the code so if the mountain won't come to me, I will go and break it using google as my crowbar. Searched for "Force .NET 4.0 Runtime" which helped me to create an [applicationname.exe].config file next to the actual .exe you would like to force to run with a different Runtime and including the following code which should do the trick: (it didn't though…)

```
</configuration></appSettings>

<startup useLegacyV2RuntimeActivationPolicy="true">

<supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0"></supportedRuntime>

</startup></configuration>
```

Virtualizing an .NET 4.5.1 uninstall & .NET 4 install

Still a couple of options left and one of those was already attempted by my predecessor who uninstalled all .NET 4.5.1. and installed .NET 4 clients on all pc's (40+) from the pilot users which made other (new) applications crash or not function.

Having been busy with Microsoft App-V 5 on a couple of projects for the last years (and already having a custom build Oracle Client in the ELN / SM setup which couldn't be installed next to the same version of the normal Oracle Client which also was being used in my clients environment…) I thought I might give virtualizing the .NET 4 setup a go, including it into an already 65 step virtual package installation.

As google is my biggest friend, I turned for help on my quest for results but for once was let down:

I found some very promising results but none of them included the .NET 4 virtualization:

http://technet.microsoft.com/nl-nl/appvirtualization/dd146065.aspx

http://technet.microsoft.com/nl-nl/appvirtualization/dd146065.aspx

	.NET Framework dependency	Sequencing .NET Framework on an App-V 4.5 workstation	Deploying App-V 4.5 packages with .NET Framework dependencies	Deploying legacy App-V packages that include .NET Framework to App-V 4.5 clients
XP SP2	1.0	.NET Framework can be virtualized		
	1.1			
	2.0			
	3.0			
	3.5 RTM			
	3.5 SP1	.NET Framework must be locally installed		
Vista *RTM*	2.0	installed by default as part of Windows Vista		
	3.0			
	3.5 (all)	.NET Framework must be locally installed		
Vista *SP1* (Fully patched)	2.0	installed by default as part of Windows Vista		
	3.0			
	3.5 RTM	no explicit installation required		
	3.5 SP1	.NET Framework must be locally installed		
Windows 7	2.0	installed by default as part of Windows 7		
	3.0			
	3.5 RTM			
	3.5 SP1			

At the time App-V 5.0 SP2 HF5 was the latest release for the App-V sequencer and having a dependency on .NET 4.5.1. I already worried about uninstalling it but I was getting more desperate so gave it a shot anyway.

I tried different options during the uninstall of .NET 4.5.1. and install of .NET 4 during sequencing but all attempts ended at compile time with an error (which is really frustrating as it looks very promising until the end when you end up with a big pile of nothing…)

Knowing this suite had to be installed on more than half of all pc's in my clients environment I wasn't going to give up but man this was really getting annoying…

Hacking & virtualizing the .NET 4.0 install:

When all fails & problems seem to be too big to solve I always turn to alcohol and after a bottle of red wine & snacks I gave it yet another shot and this time I succeeded so if you are interested how I fixed it this is the point you should start paying attention ☺

Knowing my MSI stuff I reckoned that the .NET 4 setup.exe would be a wrapper for the .NET 4 MSI's which actually contain all installer logic for the .NET 4 install. That installer logic also contains the check if an higher version of the product is already installed. We would like to poke that eye and make sure the .NET 4 install doesn't see the .NET 4.5.1 install on the pc which is actually really easy to do.

Solution:

.NET 4 App-V 5 Sequencing Recipe:

Download Microsoft Orca.exe and install it on your Windows 7 PC (I didn't try this on Windows 8 as my virtualized .NET 4 package I created on Windows 7 worked on Windows 8 as well so there was no point in torturing myself again) on which you are going to virtualize your application and the .NET 4 install.

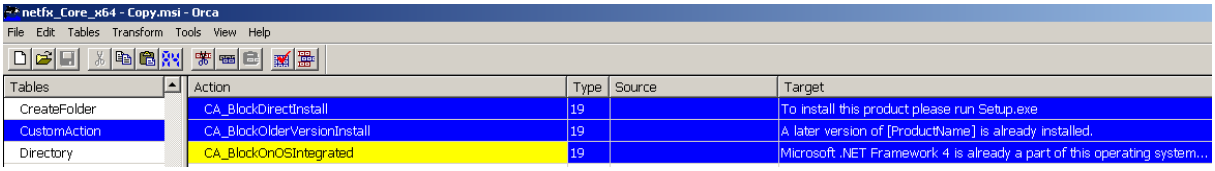
Download and run the dotNetFx40_Full_x86_x64.exe (Standalone) Installer and check in the %TEMP% (C:\Users\[UserName]\AppData\Local\Temp) directory for a new log file which is just being created telling you where the dotNetFx40_Full_x86_x64.exe setup is being unwrapped. (if you pay attention during the initial install you will see where it goes and it's probably on the root of your local disk containing the most amount of disk space in a folder named [GUID] (a lot of numbers and letters ☺)

Open the [GUID] folder and right click the netfx_Core_x64.msi & select "Edit with Orca"

Browse to the CustomAction (CA) table and remove (yes delete) the following Actions:

- CA_BlockDirectInstall - To install this product please run Setup.exe
- CA_BlockOlderVersionInstall - A later version of [ProductName] is already installed.
- CA_BlockOnOSIntegrated - Microsoft .NET Framework 4 is already a part of this operating system, but it is currently turned off. To enable the .NET Framework 4, use Turn Windows features on or off in Control Panel.

Browse to the InstallExecuteSequence table and delete the same 3 CA's too and save the MSI afterwards.



Do the same for **all** MSI's in the [GUID] folder: (two of them have no CA's)

- netfx_Core_x86.msi
- netfx_Extended_x64
- netfx_Extended_x86

Now we are ready to sequence your application and the .NET 4 install (and I might not get it all right as this was a couple of weeks ago and mind you, I figured this out while having drunk a bottle of wine so please let me know if it worked for you too ☺)

I already created the ELN / SM App-V package first and when I was going to add the .NET 4 installation I choose the option to "Modify an existing package" when I started my sequencer and choose to "Add a new application"

In your case you will start your App-V sequencer (as admin) and 'Create a New Virtual Package' and choose creating a 'Custom Installation' (instead of selecting an exe) during the sequence options.

If you are still using the pre App-V 5 SP3 sequencer add the PVAD (INSTALLDIR) for the application you want to work with the .NET 4 Runtime.

(Should you try to sequence the .NET 4 install by itself so you can add it to an App-V Connection Group choose "C:\Program Files (x86)\Microsoft.NET" as the PVAD (INSTALLDIR) if you are still working with the pre App-V 5 SP3 sequencer)

Start sequencing and start: "C:\Windows\System32\services.msc"

Right click the "Microsoft .NET Framework NGEN v4.0.30319_X64" service, select 'Properties' and set the 'startup type' to 'disabled' and 'Stop' the service if it is running.

Do the same for the "Microsoft .NET Framework NGEN v4.0.30319_X86" service.

Start: "C:\Windows\System32\cmd.exe"

Run command:

Msiexec .exe /i "[path to netfx_Core_x64.msi]\ netfx_Core_x64.msi" REBOOT=ReallySuppress /qb

Do this for **all** MSI's

Now reboot your pc.

After reboot and the sequencer is back up again:

Start: "C:\Windows\System32\cmd.exe"

And run command:

msiexec /i "[path to netfx_Core_x64.msi]\ netfx_Core_x64.msi" REINSTALLMODE=vamus REINSTALL=ALL REBOOT=ReallySuppress /qb

Do this for **all** MSI's

(.NET 4 will overwrite all newer .NET .dll's en registry which had the same name placed by .NET 4.5.1.)

Now install the application you would like to run with .NET 4

Now we will use ngen.exe to create a native image of the application you want to run with .NET 4:

Start: "C:\Windows\System32\cmd.exe"

Be aware for the bitness of the application you want to run with the .NET 4 runtime!

Mine was x86 (ELN.exe) so I used the following ngen.exe path:
C:\Windows\Microsoft.Net\Framework\v4.0.30319\ngen.exe"

Run command: (my application.exe = eln.exe)

"C:\Windows\Microsoft.Net\Framework\v4.0.30319**ngen.exe**" **Install** "C:\Program Files (x86)\Waters\NuGenesis ELN**eln.exe**"

(Should your application be x64 based you should use the following ngen.exe path:

"C:\Windows\Microsoft.Net\Framework64\v4.0.30319\ngen.exe")

Start: "C:\Windows\System32\services.msc" (Again? Yes again)

Right click the "Microsoft .NET Framework NGEN v4.0.30319_X64" service, select 'Properties' and set the 'startup type' to 'disabled' and 'Stop' the service if it is running.

Do the same for the "Microsoft .NET Framework NGEN v4.0.30319_X86" service.

Now you have ended the initial setup of .NET 4 and the installation of the application you want to run with the .NET 4 Runtime.

Start your application for adding the First Use Settings to your App-V package (although I do think you don't have to if you don't have anything to change, I did though)

Now create the package with your default settings and edit / clean your package before compilation.

(I didn't have to change anything for merge / overrule settings of the virtual files or registry keys)

You can understand I was pretty happy with myself seeing this work and having the .NET 4 functions working normally again!

I would really appreciate some feedback if it worked for you guys too so don't be shy, contact me and drop me a line.

Greetings

Jeroen