

## Step 1: Acquire the Migration Tool

- 2024-05-08 - Comments (0) - Migration

# Lasernet for Dynamics 365

To be able to migrate the configuration from one environment to another, we are required to download the migration tool provided by Microsoft.

### Prerequisites

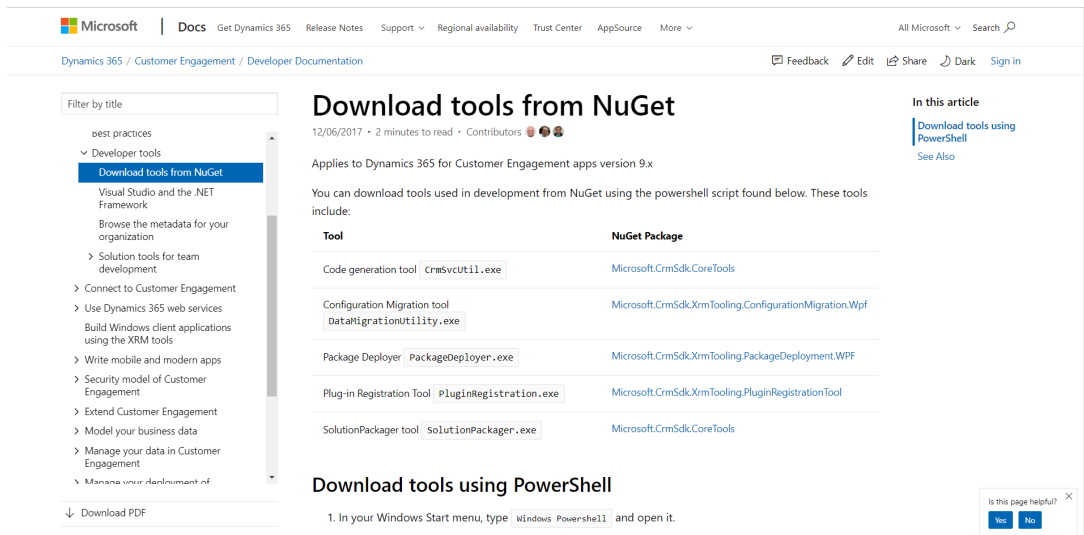
To be able to execute the steps required in this article the following is required:

- Administrative access to the installation location

### Steps

1. Navigate to

<https://docs.microsoft.com/en-us/dynamics365/customer-engagement/developer/download-tools-nuget>



The screenshot shows the Microsoft Docs page for "Download tools from NuGet". The page is titled "Download tools from NuGet" and is dated 12/06/2017. It applies to Dynamics 365 for Customer Engagement apps version 9.x. The page explains that you can download tools used in development from NuGet using a PowerShell script. It lists the following tools and their corresponding NuGet packages:

Tool	NuGet Package
Code generation tool <code>CmsSvcUtil.exe</code>	<code>Microsoft.CrmSdk.CoreTools</code>
Configuration Migration tool <code>DataMigrationUtility.exe</code>	<code>Microsoft.CrmSdk.XrmTooling.ConfigurationMigration.Wpf</code>
Package Deployer <code>PackageDeployer.exe</code>	<code>Microsoft.CrmSdk.XrmTooling.PackageDeployment.WPF</code>
Plug-in Registration Tool <code>PluginRegistration.exe</code>	<code>Microsoft.CrmSdk.XrmTooling.PluginRegistrationTool</code>
SolutionPacker tool <code>SolutionPacker.exe</code>	<code>Microsoft.CrmSdk.CoreTools</code>

Below the table, there is a section titled "Download tools using PowerShell" with the following steps:

1. In your Windows Start menu, type `Windows PowerShell` and open it.

The page also includes a sidebar with a search filter and a "Download PDF" button. A small "Is this page helpful?" poll is visible in the bottom right corner.

2. Copy the script as instructed from the steps in the link above.

```
[Net.ServicePointManager]::SecurityProtocol =  
[Net.SecurityProtocolType]::Tls12  
$sourceNugetExe =
```

```

"https://dist.nuget.org/win-x86-commandline/latest/nuget.exe"
$targetNugetExe = ".\nuget.exe"
Remove-Item .\Tools -Force -Recurse -ErrorAction Ignore
Invoke-WebRequest $sourceNugetExe -OutFile $targetNugetExe
Set-Alias nuget $targetNugetExe -Scope Global -Verbose

##
##Download Plugin Registration Tool
##
./nuget install Microsoft.CrmSdk.XrmTooling.PluginRegistrationTool -0
.\Tools
md .\Tools\PluginRegistration
$prtFolder = Get-ChildItem ./Tools | Where-Object {$_.Name -match
'Microsoft.CrmSdk.XrmTooling.PluginRegistrationTool.'}
move .\Tools\$prtFolder\tools\*. * .\Tools\PluginRegistration
Remove-Item .\Tools\$prtFolder -Force -Recurse

##
##Download CoreTools
##
./nuget install Microsoft.CrmSdk.CoreTools -0 .\Tools
md .\Tools\CoreTools
$coreToolsFolder = Get-ChildItem ./Tools | Where-Object {$_.Name -
match 'Microsoft.CrmSdk.CoreTools.'}
move .\Tools$coreToolsFolder\content\bin\coretools\*. *
.\Tools\CoreTools
Remove-Item .\Tools$coreToolsFolder -Force -Recurse

##
##Download Configuration Migration
##
./nuget install
Microsoft.CrmSdk.XrmTooling.ConfigurationMigration.Wpf -0 .\Tools
md .\Tools\ConfigurationMigration
$configMigFolder = Get-ChildItem ./Tools | Where-Object {$_.Name -
match 'Microsoft.CrmSdk.XrmTooling.ConfigurationMigration.Wpf.'}
move .\Tools$configMigFolder\tools\*. *
.\Tools\ConfigurationMigration
Remove-Item .\Tools$configMigFolder -Force -Recurse

##

```

```
##Download Package Deployer
```

```
##
```

```
./nuget install Microsoft.CrmSdk.XrmTooling.PackageDeployment.WPF -O  
.\Tools
```

```
md .\Tools\PackageDeployment
```

```
$pdFolder = Get-ChildItem ./Tools | Where-Object {$_.Name -match  
'Microsoft.CrmSdk.XrmTooling.PackageDeployment.Wpf.'}
```

```
move .\Tools\${pdFolder}\tools\*. * .\Tools\PackageDeployment
```

```
Remove-Item .\Tools\${pdFolder} -Force -Recurse
```

```
##
```

```
##Download Package Deployer PowerShell module
```

```
##
```

```
./nuget install
```

```
Microsoft.CrmSdk.XrmTooling.PackageDeployment.PowerShell -O .\Tools
```

```
$pdPoshFolder = Get-ChildItem ./Tools | Where-Object {$_.Name -match  
'Microsoft.CrmSdk.XrmTooling.PackageDeployment.PowerShell.'}
```

```
move .\Tools\${pdPoshFolder}\tools\*. *  
.\Tools\PackageDeployment.PowerShell
```

```
Remove-Item .\Tools\${pdPoshFolder} -Force -Recurse
```

```
##
```

```
##Remove NuGet.exe
```

```
##
```

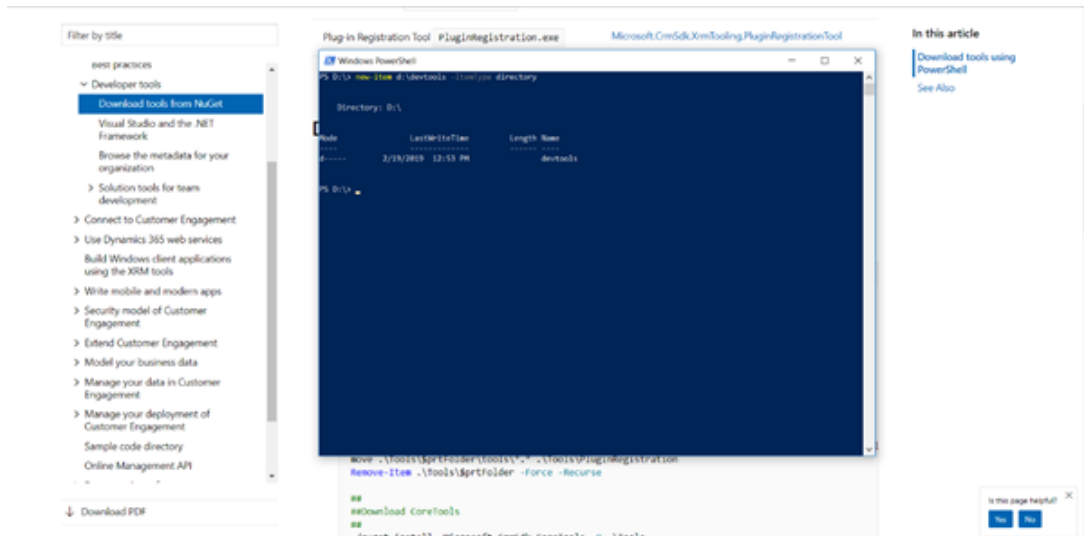
```
Remove-Item nuget.exe
```

3. In your Windows Start menu, type "Windows Powershell" and open it.

The screenshot shows a Windows Start menu search for "Windows PowerShell ISE", which is highlighted with a red box. To the right, a webpage titled "Download tools using PowerShell" is visible. The page contains a list of three steps: 1. In your Windows Start menu, type "Windows PowerShell" and open it. 2. Navigate to the folder you want to install the tools to. For example if you want to install them in a "tools" folder on your D drive, type "cd D:\tools". 3. Copy and paste the following PowerShell script into the PowerShell window and press Enter. Below the steps is a PowerShell script with the following content: \$sourceNuGetExe = "https://dist.nuget.org/win-x86-commandline/latest/nuget.exe" \$targetNuGetExe = ".\nuget.exe" Remove-Item .\tools -Force -Recurse -ErrorAction Ignore Invoke-WebRequest \$sourceNuGetExe -OutFile \$targetNuGetExe Set-Alias nuget \$targetNuGetExe -Scope Global -Verbose ## ##Download Plugin Registration tool ## ./nuget install Microsoft.CrmSdk.XrmTooling.PluginRegistrationTool -O .\tools md .\tools\PluginRegistration \$prFolder = Get-ChildItem .\tools | Where-Object {\$\_.Name -match 'Microsoft.CrmSdk.XrmTooling.P move .\tools\\${prFolder}\tools\\*. \* .\tools\PluginRegistration Remove-Item .\tools\\${prFolder} -Force -Recurse ## ##Download Coretools ## ./nuget install Microsoft.CrmSdk.CoreTools -O .\tools

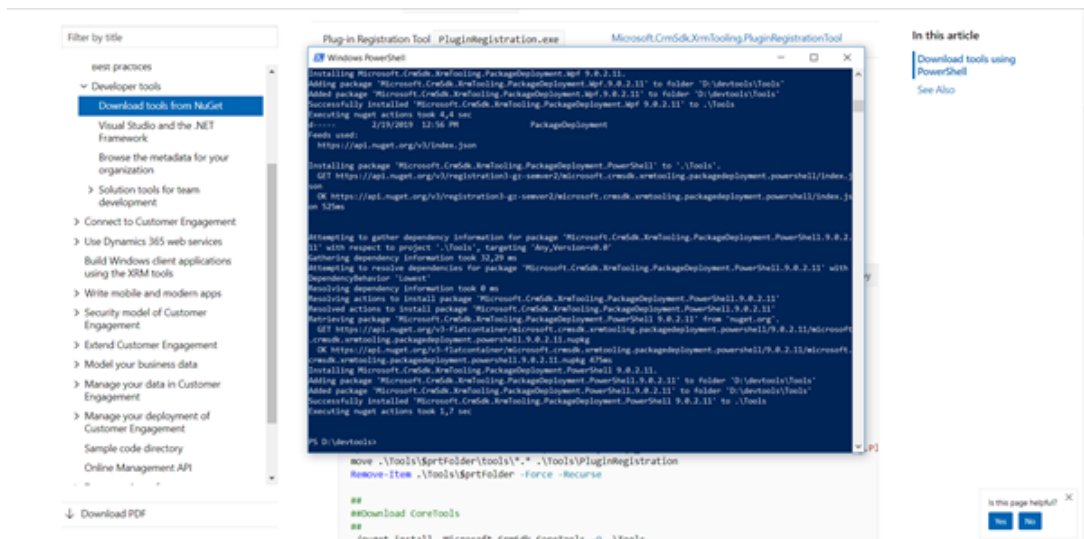
4. Create a new folder where the tools need to be installed. Type:

"new-item d:\devtools -ItemType directory"

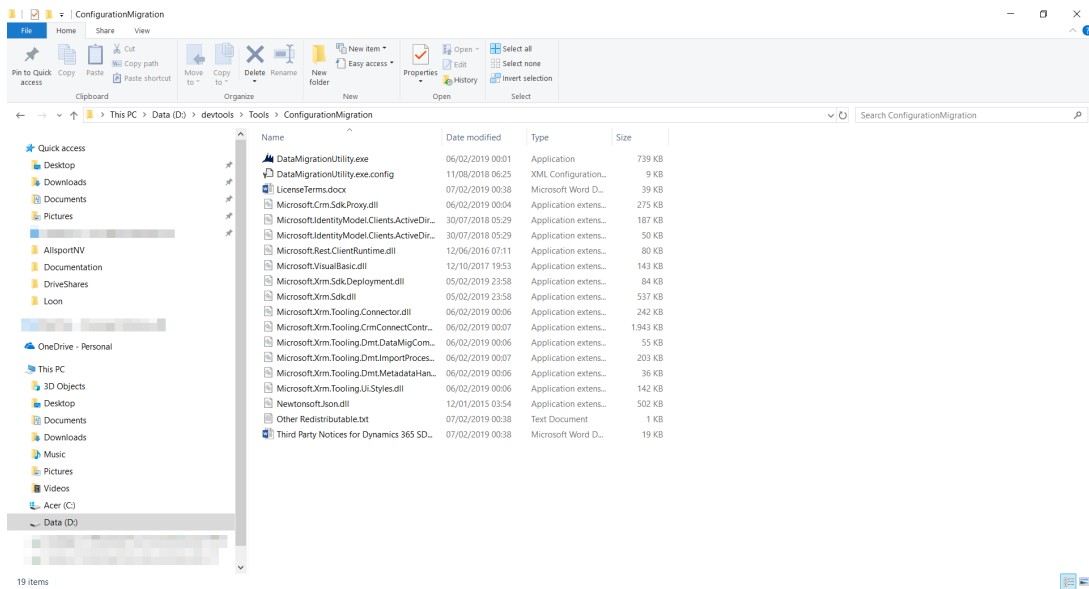


5. Navigate to the folder you want to install the tools to. For example, if you want to install them in a *devtools* folder on your D drive, type "cd D:\devtools".

6. Paste the script copied above into the PowerShell window and press the Enter key. PowerShell will finish with the following message:



7. Navigate to the corresponding folder. You will now have acquired the MigrationTool.



## Related Content

- [Step 3: Export Configuration Data](#)
- [Step 2: Create a Migration Schema](#)