

Setting up a new Developer Environment for Manitou

12/29/2023 1:55 pm EST

Overview

This is a working document to help on-board any new developers and get them running in the Manitou system. It walks through the process to take a stock Windows environment and be able to work effectively in the Bold Development process. This should be a quick start for new developers as well as a repository for knowledge that anyone can refer back to should they need a refresher.

Estimated time to complete all tasks and build code: 3 hours

Installs

We need several tool and packages that must be installed. Many of these can be found online but we can also get them from internal sources.

Shared Installation Files: \\BoldFiles\\Development\\Software\\CD_Images\\

- Microsoft IIS
- Microsoft SQL Server 2012R2 or higher
- [SQL Server Management Studio \(SSMS\)](#)
- Chrome browser
- Visual Studio Enterprise
 - .NET Desktop Development
 - Desktop development with C++
 - ASP.NET and web development
 - Individual Components

- > Visual Studio core editor
- > .NET desktop development
- > Desktop development with C++
- > ASP.NET and web development
- > Node.js development
- > Mobile development with .NET
- > .NET Core cross-platform development
- ✓ Individual components
 - ✓ .NET Framework 4.6.2 SDK
 - ✓ .NET Framework 4.6.2 targeting pack
 - ✓ .NET Framework 4.7.1 SDK
 - ✓ .NET Framework 4.7.1 targeting pack
 - ✓ .NET Framework 4.7.2 SDK
 - ✓ .NET Framework 4.7.2 targeting pack
 - ✓ Visual C++ ATL for x86 and x64
 - ✓ Visual C++ MFC for x86 and x64
 - ✓ Windows 10 SDK (10.0.16299.0) for UWP: C#, VB, JS
 - ✓ Windows 10 SDK (10.0.16299.0) for UWP: C++
 - ✓ Windows 10 SDK (10.0.16299.0) for Desktop C++ [x...

- Git for Windows
- Notepad++
- Build_Manitou_All.bat (copy to C:\DevOps)
- Bold_Manitou_Install_Services.bat (copy to C:\DevOps)
- Node.js - <https://nodejs.org/en/download/>
- Create folder c:\DevOps\Manitou for the code
- Download nuget.exe - <https://www.nuget.org/downloads>

As an option to run the Visual Studio installer and individually select the items to install, you can use the attached VS2017Install.vsconfig file and run that with the installer like this: `vs_enterprise_installer.exe --config "C:\VS2017Install.vsconfig"`. This will bring up the Visual Studio installer and have all the necessary items selected to install, just select "Install" to start the installation process.

Windows Junction file

You will need to create a Junction that allows us to register the services to a path but then change where that path points. This is very helpful if we have to switch to another file path for any reason.

Open a command prompt as an administrator run the following command:

```
mklink /J C:\ManitouDev C:\DevOps\Manitou
```

NuGet

NuGet is an executable that needs to be accessible from your command line. A very easy way to get this working is to download NuGet from [here](#) and place the EXE in your C:\Program Files\dotnet folder.

If you want to place NuGet in another folder, you will need to make sure that folder is accessible by your %PATH%

Environmental Variable.

Pulling code base

Log into dev.azure.com/boldgroup

Navigate to the Manitou project and the Repo section.

Choose the Manitou Repository and hit the clone button. Change the IDE to Clone is Visual Studio.

This should launch Visual Studio and bring you to the clone page. We are using C:\DevOps as the base path for all our code. You can use another drive letter if needed but all repositories should be in the same base path.

You can also use command line for GIT

- Delete all items from folder including the .git folder
- Init Git and add repo URL
- Run git pull origin master (all code is now clean)

IIS Setup

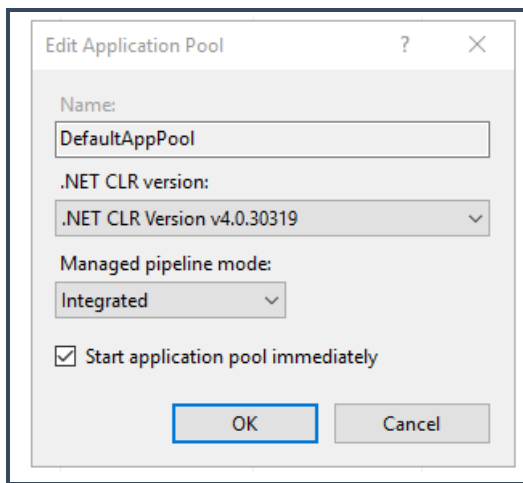
You must have a self-signed cert created and bound to the default website before working on NEO API solutions.

1. Open IIS and click on the server
2. On the right, in the IIS section, open "Server Certificates"
3. On the right select "Create Self-Signed Certificate"
4. Provide the Fully Qualified Domain Name (FQDN) of the machine, certificate store should be "Personal", click OK
5. Expand Sites and click on the "Default Web Site", click "Bindings" on the right
6. Click "Add", set the "Type" as 'https', using the drop-down list at the bottom, select the self-signed certificate, no other selections are needed, click "OK"

IIS Website Setup

When building the Neo Client, Neo Api, and Neo Oauth solutions, these actions will also create the necessary items for the Neo website to work. Should you need to setup the website:

1. Open IIS manager, expand your machine name and select "Application Pools"
2. On the right, select "Add Application Pool", name it what you want and give it the following settings



3. Right-click on "Default Web Site" and select "Add Application"

a. Select the App Pool you created previously

b. Add one application for each of the following

i. Alias: api, Physical Path: C:\ManitouDev\Neo\Api\Manitou.Api, Connect as: Application User, Enable Preload not selected

ii. Alias: oauth, Physical Path: C:\ManitouDev\Neo\Oauth\Manitou.Oauth, Connect as: Application User, Enable Preload not selected

iii. Alias: manitou, Physical Path: C:\ManitouDev\Neo\Client\Manitou.Client, Connect as: Application User, Enable Preload not selected

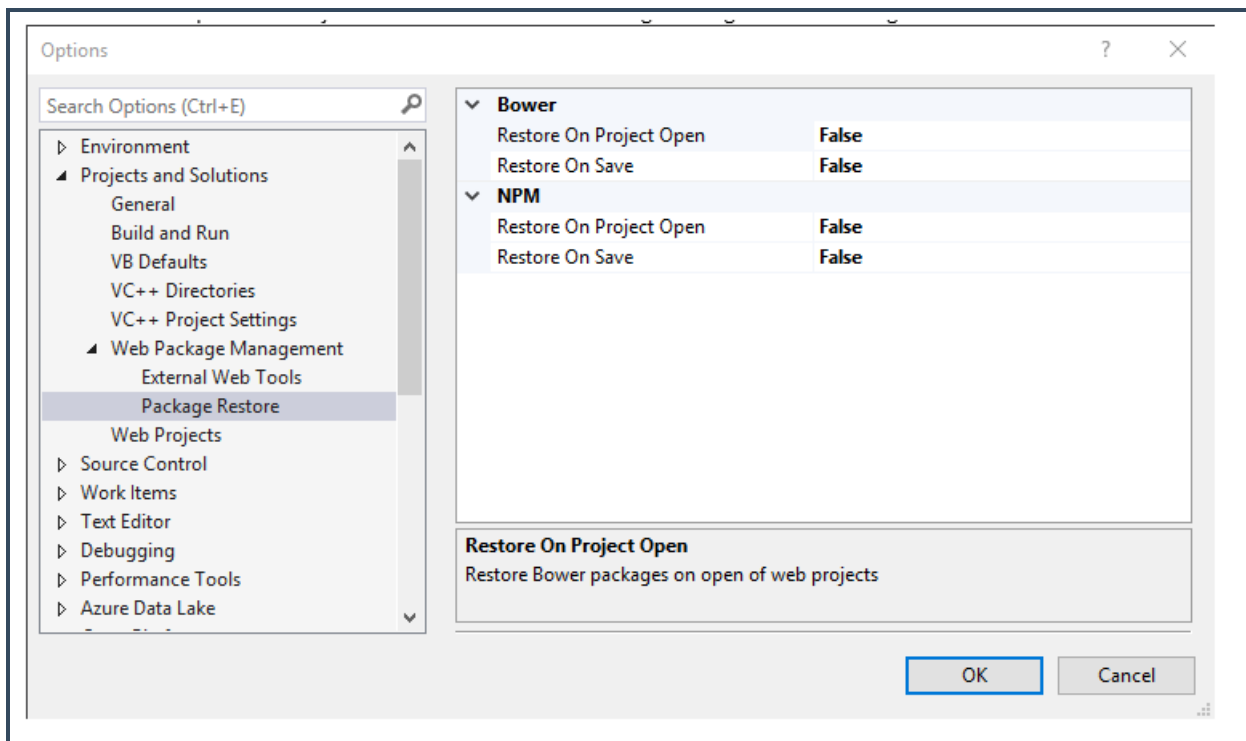
Visual Studio Setup

Bower Update

There is a setting in Visual Studio where it runs the bower install every time you open the project. This, in my case, was causing bower to make decisions like choosing what angular version to grab and what to update, etc; breaking the gulp and VS build.

If you already haven't done this or you have no idea what I am talking about, open your Visual Studio 2017:

Go to : Tools > Options > Projects and Solutions > Web Package Management > Package Restore

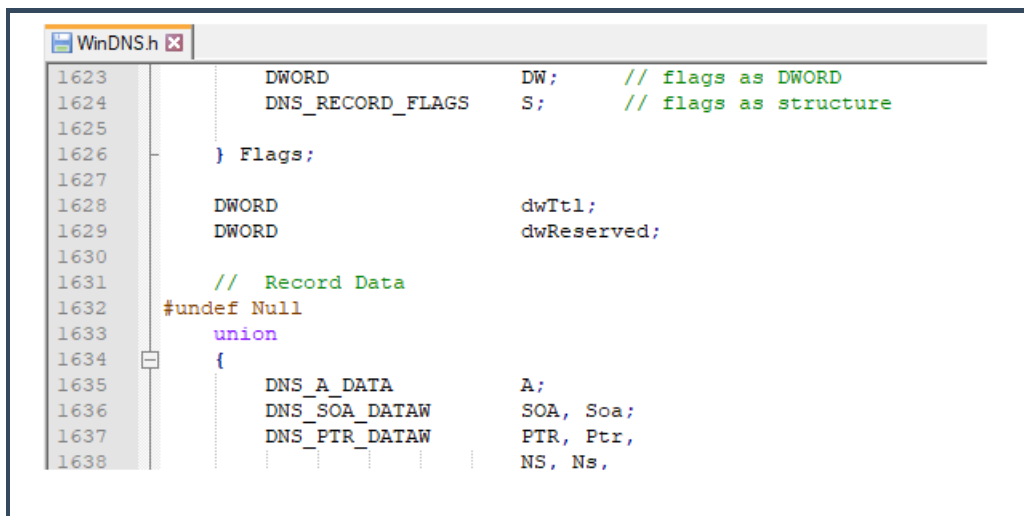


Set Bower: Restore On Project Open = false, Restore On Save = false.

Manual Changes

In the following file, make the following change: c:\Program Files (x86)\Windows Kits\10\Include\10.0.16299.0\um\WinDNS.h

At line 1633, insert the following (before the 'union' that was at 1633 (case sensitive)): #undef Null



Bypass Code Signing In Dev Environment

A command needs to be run to exclude the dev environment from needing code signing.

This command needs to be run in both an x86 Visual Studio Command Prompt and a x64 Visual Studio Command Prompt, open each as an administrator and run:

```
sn.exe -Vr *,151ae431f239ddf0
```

NEO Client

Gulp Setup

1. Run the node.js command prompt as an administrator (or open windows command prompt as administrator and run the following:

- a. For x86

- i. "C:\Program Files (x86)\nodejs\nodevars.bat" & cd "C:\ManitouDev\Neo\Client\Manitou.Client" & npm i bower@1.8.2 canonical-path@0.0.2 del@3.0.0 gulp@3.9.1 gulp-angular-filesort@1.2.1 gulp-angular-templatecache@2.2.0 gulp-autoprefixer@5.0.0 gulp-concat@2.6.1 gulp-debug@3.2.0 gulp-expect-file@0.0.7 gulp-flatten@0.4.0 gulp-inject@4.3.1 gulp-livereload@3.8.1 gulp-minify-css@1.2.4 gulp-minify-css@1.2.4 gulp-rename@1.2.2 gulp-rev@8.1.1 gulp-sass@3.2.1 gulp-sourcemaps@2.6.4 gulp-uglify@3.0.2 require-nocache@1.0.0 run-sequence@2.2.1 stream-series@0.1.1 typescript@2.8.1 --save-exact & npm i -g bower@1.8.2 canonical-path@0.0.2 del@3.0.0 gulp@3.9.1 gulp-angular-filesort@1.2.1 gulp-angular-templatecache@2.2.0 gulp-autoprefixer@5.0.0 gulp-concat@2.6.1 gulp-debug@3.2.0 gulp-expect-file@0.0.7 gulp-flatten@0.4.0 gulp-inject@4.3.1 gulp-livereload@3.8.1 gulp-minify-css@1.2.4 gulp-minify-css@1.2.4 gulp-rename@1.2.2 gulp-rev@8.1.1 gulp-sass@3.2.1 gulp-sourcemaps@2.6.4 gulp-uglify@3.0.2 require-nocache@1.0.0 run-sequence@2.2.1 stream-series@0.1.1 typescript@2.8.1 --save-exact & gulp build

- b. For x64

- i. "C:\Program Files\nodejs\nodevars.bat" & cd "C:\ManitouDev\Neo\Client\Manitou.Client" & npm i bower@1.8.2 canonical-path@0.0.2 del@3.0.0 gulp@3.9.1 gulp-angular-filesort@1.2.1 gulp-angular-templatecache@2.2.0 gulp-autoprefixer@5.0.0 gulp-concat@2.6.1 gulp-debug@3.2.0 gulp-expect-file@0.0.7 gulp-flatten@0.4.0 gulp-inject@4.3.1 gulp-livereload@3.8.1 gulp-minify-css@1.2.4 gulp-minify-css@1.2.4 gulp-rename@1.2.2 gulp-rev@8.1.1 gulp-sass@3.2.1 gulp-sourcemaps@2.6.4 gulp-uglify@3.0.2 require-nocache@1.0.0 run-sequence@2.2.1 stream-series@0.1.1 typescript@2.8.1 --save-exact & npm i -g bower@1.8.2 canonical-path@0.0.2 del@3.0.0 gulp@3.9.1 gulp-angular-filesort@1.2.1 gulp-angular-templatecache@2.2.0 gulp-autoprefixer@5.0.0 gulp-concat@2.6.1 gulp-debug@3.2.0 gulp-expect-file@0.0.7 gulp-flatten@0.4.0 gulp-inject@4.3.1 gulp-livereload@3.8.1 gulp-minify-css@1.2.4 gulp-minify-css@1.2.4 gulp-

```
rename@1.2.2 gulp-rev@8.1.1 gulp-sass@3.2.1 gulp-sourcemaps@2.6.4 gulp-tsd@0.1.1 gulp-  
typescript@4.0.1 gulp-uglify@3.0.2 require-nocache@1.0.0 run-sequence@2.2.1 stream-series@0.1.1  
typescript@2.8.1 --save-exact & gulp build
```

Database

You will need to have a database for Manitou. We have a seed database and several others that can be used.

Add the Manitou Database to your system using SSMS.

You may need to upgrade your copy of the database before running Manitou system. This is done with the DBManager tool.

If you are using a database from another machine you will need to make the following edits:

- In the DEVCONF table, update the NAME and IPADDR for this machine
- In the DEVCONF_D table, update all logger reference that might be there for the other machine.
- Remove all entries in the WORKSTN table

Initial Manitou Build

Manitou is a very large system involving many solutions. Instead of having to run each solution in Visual Studio, we have a bat file to build all the key solutions. This must be run from a Developer Command Prompt for VS 2017 in administrator mode.

Change directory to C:\DevOps and run the following command:

```
Build_Manitou_All.bat
```

You can expect the "DBManager" solution to fail its build, this package requires additional setup not covered in this document.

Note: When running Build_Manitou_All.bat, occasionally one of the solutions may get stuck and not continue. If you notice a solution being built that seems to be taking longer than 25 minutes, press CTRL+C in the command prompt window. You will see a message, "Terminate Batch Job (Y/N)?", type N and press enter, this should allow the rest of the packages to be built and you can open the failed package in Visual Studio to resolve any issues and build.

Running the Build_Manitou_All.bat will take a while. During this time you can open solutions in Visual Studio, however, if you use any of the clean or build commands before the Build_Manitou_All.bat has finished, you may see build failures in your command window, which will cause you to need to either run the Build_Manitou_All.bat again or rebuild the failed solutions manually.

Once the Manitou processes build, you will need to register all the services. This is done by opening a windows command prompt as an administrator running the following file:

```
Bold_Manitou_Install_Services.bat
```

Install Services

Once all of the code has been built, you will need to install/register the service executable with windows, open a windows command prompt as an administrator and run [Bold_Manitou_Install_Services.bat](#)

Configuration

1. Create a shortcut to C:\ManitouDev\bin\MSM.exe
2. Right-click on the shortcut and select "Properties"
3. Modify the "Target" to the following: C:\ManitouDev\bin\MSM.exe -small -config
4. Click "Apply" then "OK"
5. Double-click the shortcut
6. Under "Configuration" click "Add" and give your configuration a name (machine name will due), press "OK"
7. Ensure your new Configuration is selected and select "Edit"
8. Set "Configuration Name", "Computer", and "Server Name" to the short machine name (not FQDN)
9. The "Database Service" and "Backup Db Service" are the name of the SQL instance
10. Set the "Database Name", Database User", and "Database Password", click OK
11. Create a new shortcut to C:\ManitouDev\bin\MSM.exe and double-click the shortcut

Creating Manitou Shortcuts

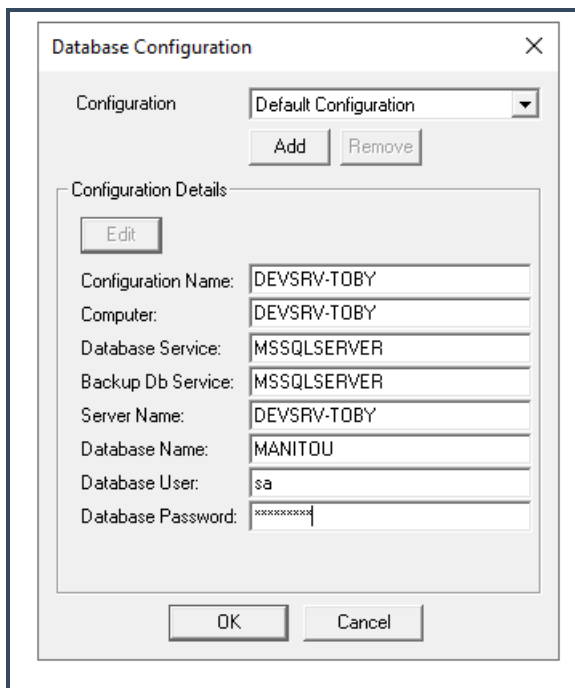
Open C:\ManitouDev\bin and create shortcuts from the following applications in that directory on the desktop:

- LogViewer
- MSM(2x)
 - One should be named MSM Config have the -config flag set under properties in the Target
- DistCommander
- FEPCCommander
- FEPTestClient

Setting Up Manitou

MSM Config

We first need to configure the Manitou System Monitor (MSM) for our system. This tells it what machine and database to use. We first open the MSM Config and click the Edit button. Populate the values with your machine specific ones.



Licensing Manitou

Each Manitou machine must be licensed individually. This is locked to the machine itself. There is an encrypted file called bold.lic that must be placed at the same location as the executables. For us this is C:\ManitouDev\bin.

Work with your manager or support to get the machine licensed.

Local Utility Service and Certificate

Creating the Local Certificate Utility Service certificate is performed while running `Bold_Manitou_Install_Services.bat` and may be unnecessary here.

Build the Local Certificate Utility Service if you have not already.

Open a windows command prompt as administrator and navigate to

C:\ManitouDev\Neo\LocalCertificateUtility\LocalCertificateUtility\bin\Debug and run the following command

```
LocalCertificateUtility.exe --p=7020 --ue --cn
```

If you are planning on running Firefox browser you will need to authorize the local certificate

- Go to <https://localhost:7020>
- Authorize the self-signed certificate
- Reload the NEO sign-in page

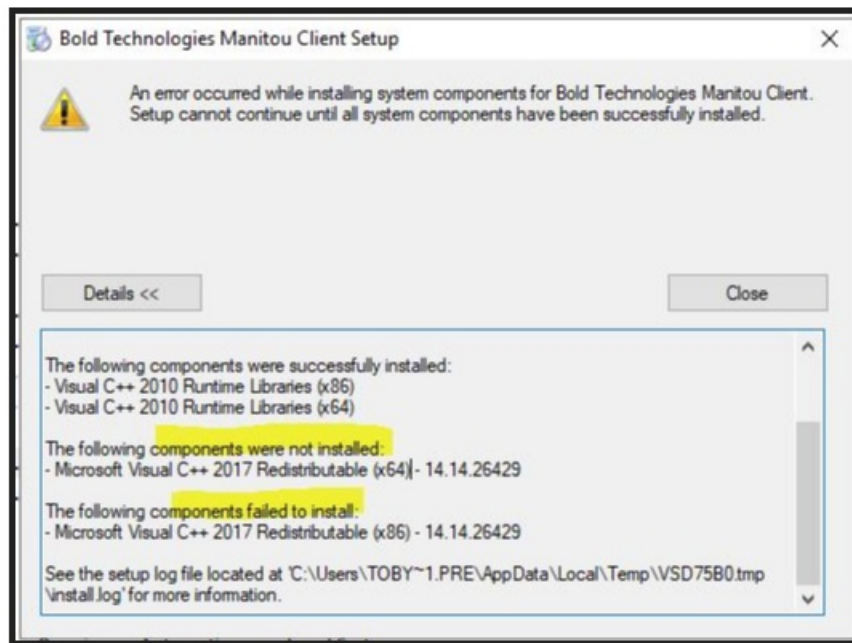
VB Clients

You will need to install the VB Clients in order to use the Supervisor Workstation. Instead of setting up the entire VB

development environment, we will use the installer located here \\boldfiles\Pre-Release\Manitou\Pre-Release-DevOps\Installers\ManitouClient_Setup.

Copy and run the installer. Set the path to install to C:\ManitouClients

If you get the following error, you may have to use Add and Remove Programs to remove these so the installer can finish.



Setting up Reports

1. From the Manitou\Libs\wRTE2PDF folder, copy wRTE2PDF04.dll and wRTE2PDF04x64.dll to Manitou\Bin.
2. Edit Manitou\Neo\Api\Manitou.Api\Web.config. change the path for "ReportTemplates" to the following:
"C:\DevOps\Manitou\Neo\Api\Manitou.Api.Clients\ReportTemplates"

SSO User

Get with IT/Manager to have your user added to the Azure ActiveDirectory so you can use the SSO in Manitou.