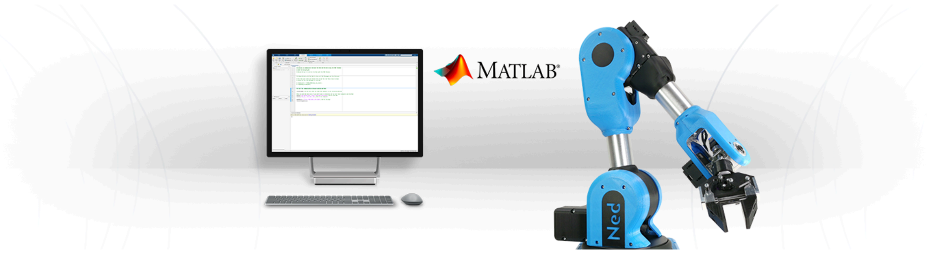
Setup Matlab and the ROS Toolbox for Ned/Ned2

V.2.1



Difficulty: `medium`

Time: `~1h` with all the installations (depending on your internet connection).

> **Note**
>
> **This tutorial is working from:**
>
> The version v4.1.0 of the ned_ros_stack
> The version v4.1.0 of Niryo Studio
>
> If you are using a Niryo One, please refer to this tutorial (https://niryo.com/docs/niryo-one/developer-tutorials/set-up-matlab/).

Objectives

- Prepare the Matlab work environment in order to use it for all of our future Matlab tutorials
- Be able to use Ned/Ned2 with Matlab
- Be able to create content with Ned/Ned2 and Matlab using ROS

Requirements

- Basic knowledge of Matlab
- Basic knowledge in robotics and ROS
- Being able to use Ned/Ned2

Note

Documentation of Ned: https://archive-docs.niryo.com/product/ned/index.html
Documentation of Ned2: https://archive-docs.niryo.com/product/ned2/index.html
Matlab documentation: https://fr.mathworks.com/help/matlab/
ROS documentation: http://wiki.ros.org/Documentation

What you will need

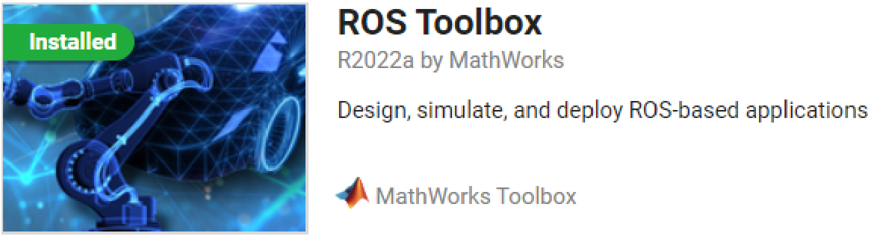
- A Ned or a Ned2
- Ubuntu or Windows
- Matlab installed on your computer (2014 version or more)
- Matlab ROS Toolbox: https://fr.mathworks.com/products/ros.html (https://fr.mathworks.com/products/ros.html)
- A Wi-Fi communication

Note

To do this tutorial we used Windows10, Matlab2022a and a Ned2.

Add-ons installation

It is necessary to install the Matlab add-ons by downloading them from the Matlab website (https://fr.mathworks.com/products/ros.html).



ROS Toolbox

R2022a by MathWorks

Design, simulate, and deploy ROS-based applications

Matlab ROS Toolbox

What is the Matlab Ros Toolbox?

ROS Toolbox provides an interface that connects Matlab® and Simulink® to the Robot Operating System (ROS and ROS 2), allowing you to create a network of ROS nodes. The toolbox includes Matlab functions and Simulink blocks for importing, analyzing, and reading ROS data stored in rosbag files. In addition, you can establish a connection to a live ROS network to access ROS messages.

Tip

We advise you to have a first look to the Matlab tutorial (https://fr.mathworks.com/help/ros/ug/get-started-with-ros.html) concerning the ROS Toolbox.

Download the “ned_ros” package from Github

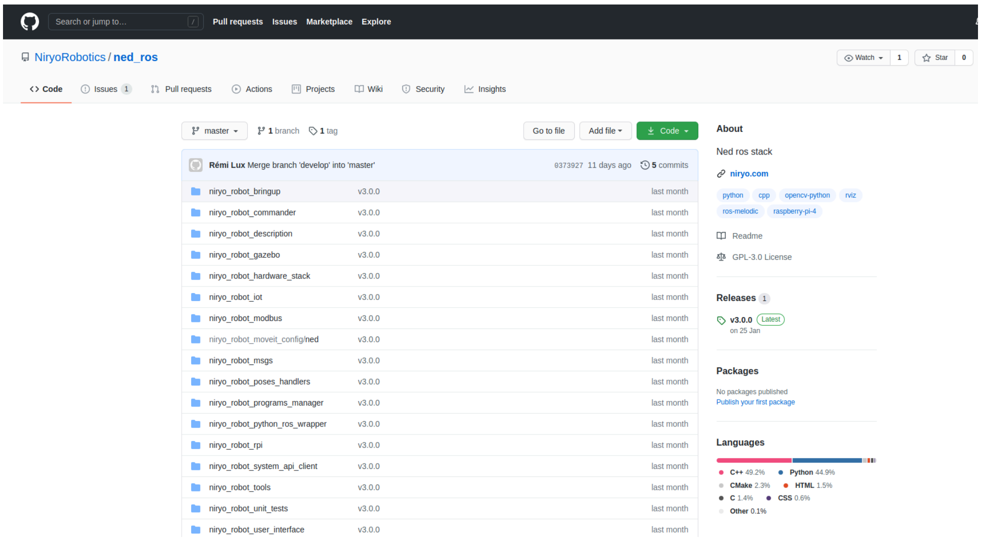
In order to control Ned/Ned2 with Matlab, it is essential to generate Ned’s messages on Matlab. To do so, you will first need to download the “ned_ros” package available on Niryo’s Github.

You can clone the “ned_ros” package from github

```
git clone https://github.com/NiryoRobotics/ned_ros.git
```

Or download as a ZIP

Extract it in an easy-to-access file (e.g. D:/ned_ros) and check that the file has the name “ned_ros” (not “ned_ros_master”).



Rémi Lux Merge branch 'develop' into 'master' 0373927 11 days ago 5 commits

niryo_robot_bringup	v3.0.0	last month
niryo_robot_commander	v3.0.0	last month
niryo_robot_description	v3.0.0	last month
niryo_robot_gazebo	v3.0.0	last month
niryo_robot_hardware_stack	v3.0.0	last month
niryo_robot_iot	v3.0.0	last month
niryo_robot_modbus	v3.0.0	last month
niryo_robot_moveit_config/ned	v3.0.0	last month
niryo_robot_msgs	v3.0.0	last month
niryo_robot_poses_handlers	v3.0.0	last month
niryo_robot_programs_manager	v3.0.0	last month
niryo_robot_python_ros_wrapper	v3.0.0	last month
niryo_robot_rpi	v3.0.0	last month
niryo_robot_system_api_client	v3.0.0	last month
niryo_robot_tools	v3.0.0	last month
niryo_robot_unit_tests	v3.0.0	last month
niryo_robot_user_interface	v3.0.0	last month

About

Ned ros stack

niryo.com

python cpp opencv-python rviz ros-melodic raspberry-pi-4

Readme

GPL-3.0 License

Releases 1

v3.0.0 Latest
on 25 Jan

Packages

No packages published
Publish your first package

Languages

● C++ 49.2% ● Python 44.9% ● CMake 2.3% ● HTML 1.5% ● C 1.4% ● CSS 0.6% ● Other 0.1%

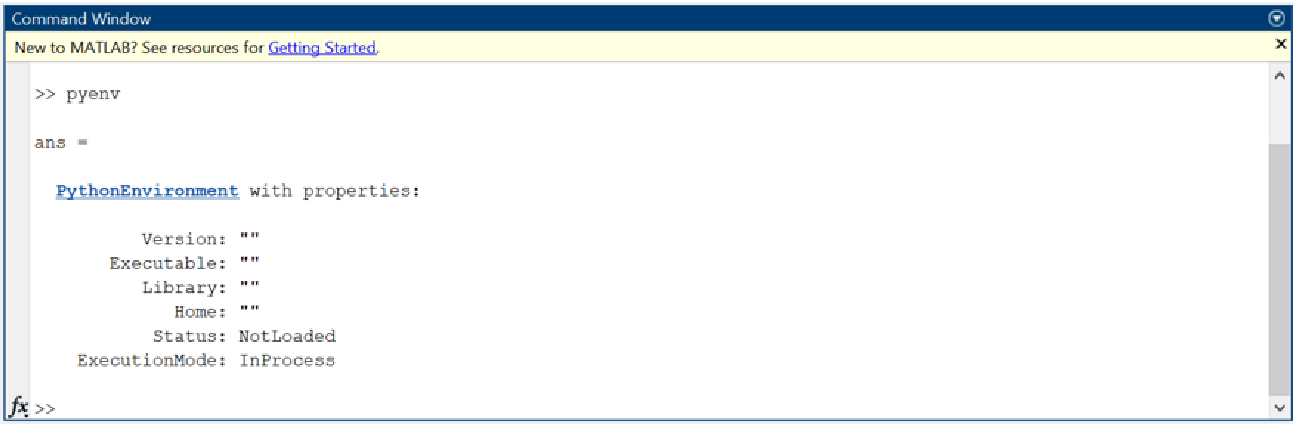
Ned ROS stack github repository

Setup pyenv in Matlab

In the Matlab Command Window, type:

```
pyenv
```

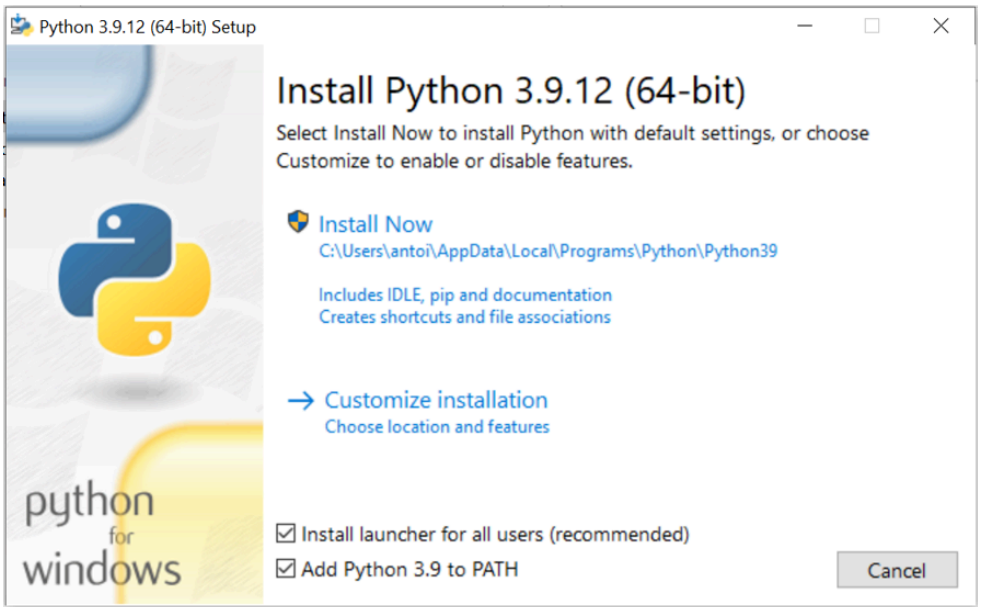
If you get this answer:



```
>> pyenv

ans = 

  PythonEnvironment with properties:

          Version: ""
       Executable: ""
          Library: ""
             Home: ""
           Status: NotLoaded
    ExecutionMode: InProcess

fx >>
```

Command Window answer

You have to install python3.9 (http://www.python.org/downloads/windows).



Python 3.9.12 (64-bit) Setup

Install Python 3.9.12 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\antoi\AppData\Local\Programs\Python\Python39
Includes IDLE, pip and documentation
Creates shortcuts and file associations
→ Customize installation
Choose location and features
python for windows
Cancel

☑ Install launcher for all users (recommended)

☑ Add Python 3.9 to PATH

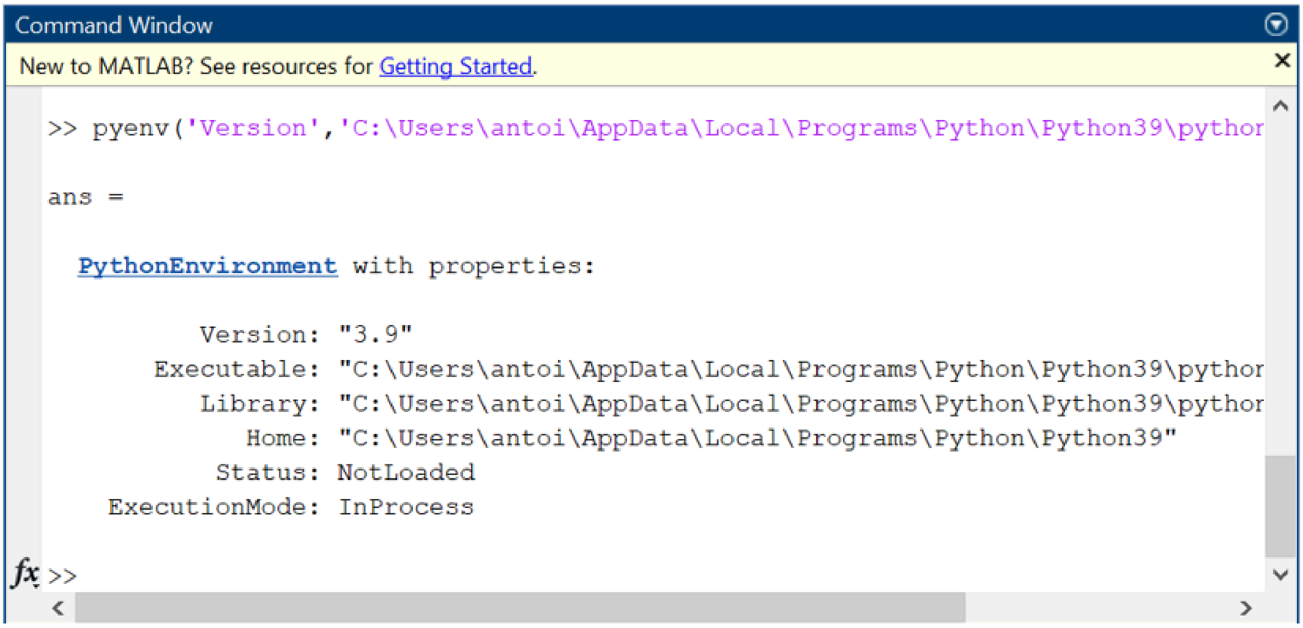
Python installer

Note

Don't forget to check the box "Add Python 3.9 to PATH"

Now, in the Matlab Command Window, type:

```
pyenv('Version'),('YOUR_PATH/python.exe')
```



Command Window

New to MATLAB? See resources for Getting Started.

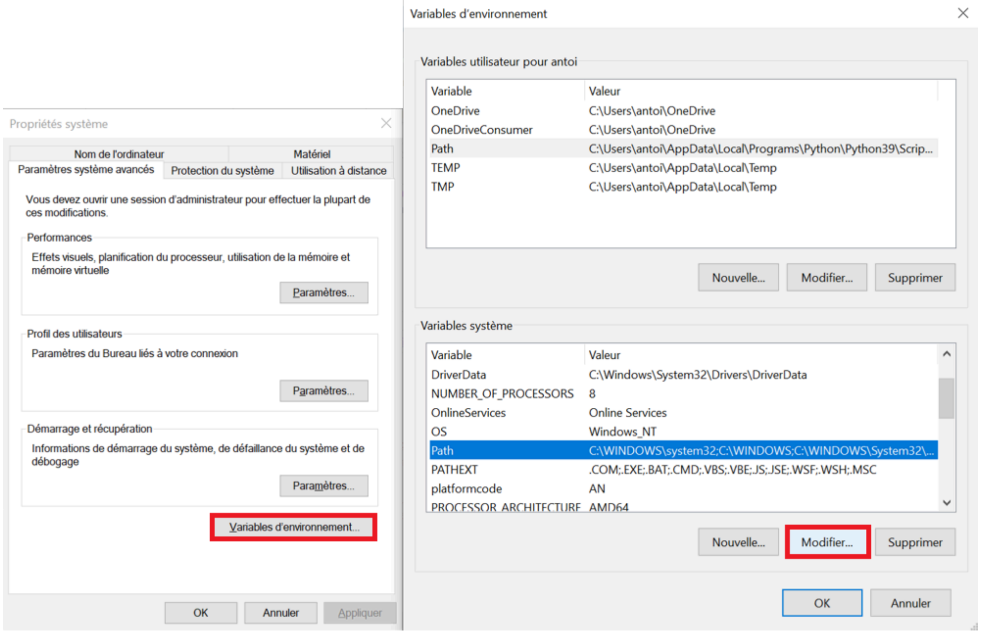
```
>> pyenv('Version','C:\Users\antoi\AppData\Local\Programs\Python\Python39\pythor

ans =

  PythonEnvironment with properties:

          Version: "3.9"
       Executable: "C:\Users\antoi\AppData\Local\Programs\Python\Python39\pythor
          Library: "C:\Users\antoi\AppData\Local\Programs\Python\Python39\pythor
             Home: "C:\Users\antoi\AppData\Local\Programs\Python\Python39"
           Status: NotLoaded
    ExecutionMode: InProcess

fx >>
```

Command Window with pyenv setup

Install Cmake version 3.19

Matlab will need the version 3.19 of Cmake. For this, you should first download CMake using this link, download a .zip file and extract it.

Copy the path of the /bin file, then go to the advanced settings of the computer and add a new variable to paste the path.



Propriétés système

Nom de l'ordinateur | Matériel | Paramètres système avancés | Protection du système | Utilisation à distance

Vous devez ouvrir une session d'administrateur pour effectuer la plupart de ces modifications.

Performances

Effets visuels, planification du processeur, utilisation de la mémoire et mémoire virtuelle

Paramètres...

Profil des utilisateurs

Paramètres du Bureau liés à votre connexion

Paramètres...

Démarrage et récupération

Informations de démarrage du système, de défaillance du système et de débogage

Paramètres...

Variables d'environnement...

OK | Annuler | Appliquer

Variables d'environnement

Variables utilisateur pour antoi

Variable	Valeur
OneDrive	C:\Users\antoi\OneDrive
OneDriveConsumer	C:\Users\antoi\OneDrive
Path	C:\Users\antoi\AppData\Local\Programs\Python\Python39\Scrip...
TEMP	C:\Users\antoi\AppData\Local\Temp
TMP	C:\Users\antoi\AppData\Local\Temp

Nouvelle... | Modifier... | Supprimer

Variables système

Variable	Valeur
DriverData	C:\Windows\System32\Drivers\DriverData
NUMBER_OF_PROCESSORS	8
OnlineServices	Online Services
OS	Windows_NT
Path	C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
platformcode	AN
PROCESSOR_ARCHITECTURE	AMD64

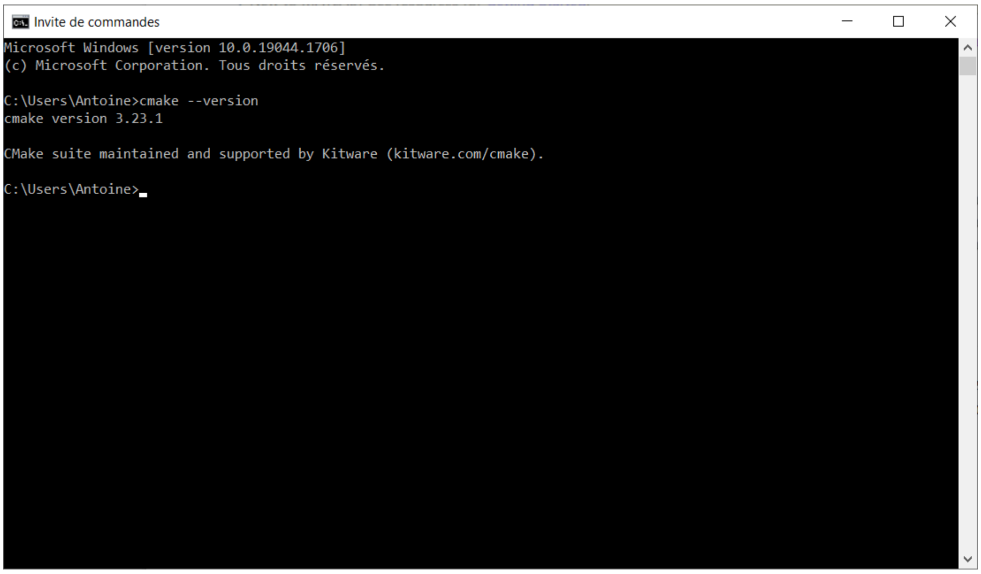
Nouvelle... | Modifier... | Supprimer

OK | Annuler

CMake setup in advanced settings

You can now test this command in a prompt:

```
cmake --version
```



```
Microsoft Windows [version 10.0.19044.1706]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\Antoine>cmake --version
cmake version 3.23.1

CMake suite maintained and supported by Kitware (kitware.com/cmake).

C:\Users\Antoine>
```

Command prompt after CMake setup

In Matlab, try to compile these lines:

```
folderpath = "/YOUR_PATH/ned_ros"
rosgenmsg(folderpath)
```

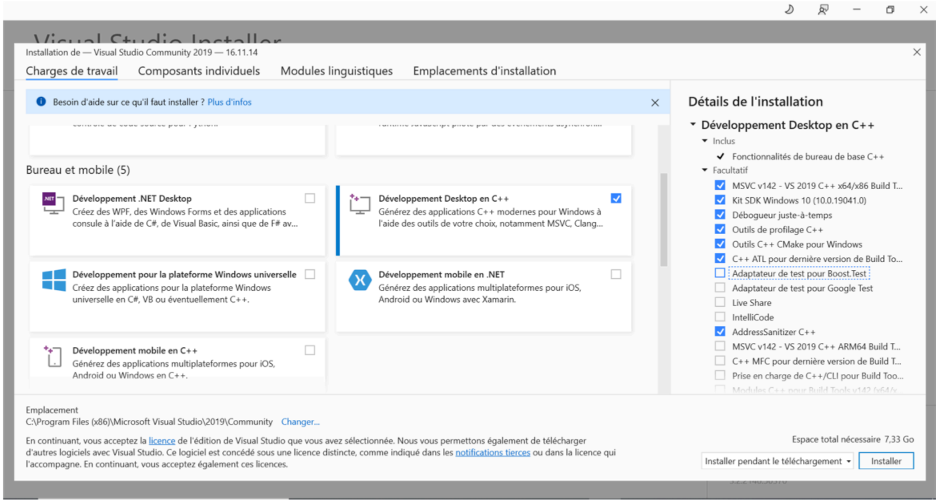
If CMake is not found, type in the Matlab Command Window:

```
>> oldPath = getenv('PATH');
>> newPath = strcat(oldPath,pathsep,'YOUR_PATH\bin'); %YOUR_PATH of the /bin folder
>> setenv('PATH',newPath);
>> system('cmake --version')
```

Now, retry the compilation of the two lines.

Possible error : “No C++ Compiler”

In this case, you have to install Visual Studio 2019 (only C++ compiler that ROS can support).



Visual Studio Installer

Installation de — Visual Studio Community 2019 — 16.11.14

Besoin d'aide sur ce qu'il faut installer ? Plus d'infos

Bureau et mobile (5)

□ **Développement .NET Desktop**
Créez des WPF, des Windows Forms et des applications consule à l'aide de C#, de Visual Basic, ainsi que de F# av...

☑ **Développement Desktop en C++**
Générez des applications C++ modernes pour Windows à l'aide des outils de votre choix, notamment MSVC, Clang...

□ **Développement pour la plateforme Windows universelle**
Créez des applications pour la plateforme Windows universelle en C#, VB ou éventuellement C++.

□ **Développement mobile en .NET**
Générez des applications multiplateformes pour iOS, Android ou Windows avec Xamarin.

□ **Développement mobile en C++**
Générez des applications multiplateformes pour iOS, Android ou Windows en C++.

Détails de l'installation

- Développement Desktop en C++
  - Inclus
    - ✓ Fonctionnalités de bureau de base C++
  - Facultatif
    - ☑ MSVC v142 - VS 2019 C++ x64/x86 Build T...
    - ☑ Kit SDK Windows 10 (10.0.19041.0)
    - ☑ Débogueur juste-à-temps
    - ☑ Outils de profilage C++
    - ☑ Outils C++ CMake pour Windows
    - ☑ C++ ATL pour dernière version de Build To...
    - □ Adaptateur de test pour Boost.Test
    - □ Adaptateur de test pour Google Test
    - □ Live Share
    - □ IntelliCode
    - ☑ AddressSanitizer C++
    - □ MSVC v142 - VS 2019 C++ ARM64 Build T...
    - □ C++ MFC pour dernière version de Build T...
    - □ Prise en charge de C++/CLI pour Build Too...
    - □ Modules C++ pour Build Tools v142 (x64/x...

Emplacement
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community Changer...

En continuant, vous acceptez la licence de l'édition de Visual Studio que vous avez sélectionnée. Nous vous permettons également de télécharger d'autres logiciels avec Visual Studio. Ce logiciel est concédé sous une licence distincte, comme indiqué dans les notifications tierces ou dans la licence qui l'accompagne. En continuant, vous acceptez également ces licences.

Espace total nécessaire 7,33 Go

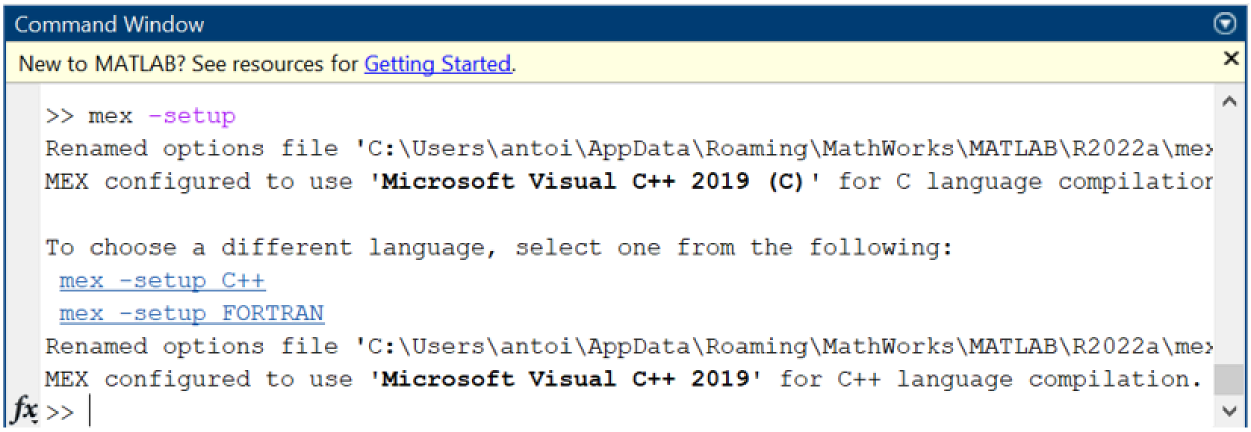
Virtual Studio 2019 installation

Then, in the Matlab Command Window, type:

```
mex -setup
```

And click on:

```
mex -setup C++
```



Command Window

New to MATLAB? See resources for Getting Started.

```
>> mex -setup
Renamed options file 'C:\Users\antoi\AppData\Roaming\MathWorks\MATLAB\R2022a\mex
MEX configured to use 'Microsoft Visual C++ 2019 (C)' for C language compilatior

To choose a different language, select one from the following:
 mex -setup C++
 mex -setup FORTRAN
Renamed options file 'C:\Users\antoi\AppData\Roaming\MathWorks\MATLAB\R2022a\mex
MEX configured to use 'Microsoft Visual C++ 2019' for C++ language compilation.
fx >>
```

C++ compiler setup in Matlab

Setup the working environment, generating Ned

messages and services from the ROS package

Now, you will have to link messages and services from Ned/Ned2 to Matlab.

For this, you have to create a Matlab script on the repository you want. Then, write on the Matlab script you just created the code below. It will specify the path of the ned_ros package which contains all messages and services and generate them.

```
folderpath = "/YOUR_PATH/ned_ros"
rosgenmsg(folderpath)
```

You should then have the message below

```
Identifying message files in folder '/YOUR_PATH/ned_ros'..Done.
Validating message files in folder '/YOUR_PATH/ned_ros'..Done.
[12/12] Generating MATLAB interfaces for custom message packages... Done.
Running catkin build in folder '/YOUR_PATH/ned_ros/matlab_msg_gen_ros1/win64'.
Build in progress. This may take several minutes...
Build succeeded.build log

To use the custom messages, follow these steps:

1. Add the custom message folder to the MATLAB path by executing:

addpath('/YOUR_PATH/ned_ros/matlab_msg_gen_ros1/win64/install/m')
savepath

1. Refresh all messages class definitions, which requires clearing the workspace, by
executing:

clear classes
rehash toolboxcache

1. Verify that you can use the custom messages.
Enter "rosmsg list" and ensure that the output contains the generated
custom message types.
```

For the step 1, you should use the command below on the command window

```
addpath('/YOUR_PATH/ned_ros/matlab_msg_gen_ros1/win64/install/m')
savepath
```

Note

Getting a warning after this step is normal. Continue to step 2.

For the step 2, you should use the command below on the command window

```
clear classes
rehash toolboxcache
```

Note

Getting a warning after this step is normal. Continue to step 3.

For the step 3, you should use the command below on the command window

```
rosmsg list
```

Make sure that you have all messages available on Matlab and more especially all the messages of niryo_robot.

```
niryo_robot_commander/ArmMoveCommand
niryo_robot_commander/GetFKRequest
niryo_robot_commander/GetFKResponse
niryo_robot_commander/GetIKRequest
niryo_robot_commander/GetIKResponse
niryo_robot_commander/JogShiftRequest
niryo_robot_commander/JogShiftResponse
niryo_robot_commander/PausePlanExecution
niryo_robot_commander/RobotCommand
niryo_robot_commander/ShiftPose
              (...)
niryo_robot_vision/DebugColorDetectionRequest
niryo_robot_vision/DebugColorDetectionResponse
niryo_robot_vision/DebugMarkersRequest
niryo_robot_vision/DebugMarkersResponse
niryo_robot_vision/ObjDetectionRequest
niryo_robot_vision/ObjDetectionResponse
niryo_robot_vision/TakePictureRequest
niryo_robot_vision/TakePictureResponse
```

You can now comment those two lines. Thus, you'll have to repeat this step again each time you connect a new robot.

> **Note**
>
> You can also see all services available on Matlab using "rosservice list" on the command window.

Well done! Matlab can now use Ned's ROS messages and services! You are now ready to turn on Ned/Ned2, connect it to internet and configure the communication between it and MATLAB.

Connect Ned/Ned2 to internet and get its IP address

Now that you have all Ned's messages and services from the "ned_ros" package available on Matlab, you can connect Ned/Ned2 to internet in order to configure the connexion between it and Matlab using the ROS Toolbox.

For this, you will first need to power Ned/Ned2 on and connect it to internet to get its IP address.

The easiest way to get Ned/Ned2's IP address is to use Niryo Studio. You can have all information here (https://archive-docs.niryo.com/product/niryo-studio/vx.x.x/en/source/connection.html).

> **Note**
>
> You have to connect your Ned/Ned2 with the Wi-Fi mode (https://archive-docs.niryo.com/product/niryo-studio/vx.x.x/en/source/connection.html#using-ned-on-your-wi-fi-network) (not Hotspot).

> **Note**
>
> For the rest of this tutorial, the IP address of Ned/Ned2 will be IP_OF_NED.

Configure the communication between Ned/Ned2 and Matlab

Now that Ned/Ned2 is connected to internet, you can configure the communication between your robot and Matlab.

Note

- The host computer is your computer with Matlab, with the IP_HOST_COMPUTER IP address
- The robot is Ned/Ned2 with IP_OF_NED IP address

To define that we will use the ‘setenv’ function from the Matlab ROS Toolbox, you have to add the lines below on your Matlab script.

```
ipaddress = "http://IP_0F_NED:11311";
setenv('ROS_MASTER_URI','ipaddress')
setenv('ROS_IP','IP_HOST_COMPUTER')
```

Run the communication between Ned/Ned2 and Matlab

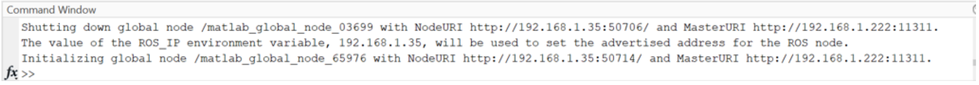
It’s now time to use Matlab and the ROS Toolbox and communicate with Ned/Ned2 from Matlab.

To communicate with Ned/Ned2 from Matlab with ROS, we will need to initialize ROS. To do that with the ROS Toolbox you need to use ‘rosinit’.
By default, rosinit creates a ROS master in Matlab and starts a global node that is connected to the master. In our case we will directly give the IP address of Ned/Ned2 to ‘rosinit’.

Add the line below on your Matlab script:

```
rosinit(ipaddress)
```

Normally, if everything is working properly, you should have this on the command window:



```
Shutting down global node /matlab_global_node_03699 with NodeURI http://192.168.1.35:50706/ and MasterURI http://192.168.1.222:11311.
The value of the ROS_IP environment variable, 192.168.1.35, will be used to set the advertised address for the ROS node.
Initializing global node /matlab_global_node_65976 with NodeURI http://192.168.1.35:50714/ and MasterURI http://192.168.1.222:11311.
fx >>
```

Rosinit setup

Before disconnecting or turning off the robot, don’t forget to type “rosshutdown” in the Matlab Command Window to cut the ROS connection.

If you write ‘rostopic list’ on the command window, you should be able to see all the topics of Ned/Ned2.

❗ Note

You are now ready to use Matlab with the ROS Toolbox and control Ned/Ned2 by following the tutorial below:

- Control Ned/Ned2 with Matlab (Use_ned_with_matlab.html)

Suggest a modification

Download as PDF