MOHID-LAGRANGIAN

INSTALLATION-GUIDE

PROLOGUE

This installation guide is done with the purpose of help for Windows and Linux users. The installation guide for Windows is more user friendly than the Linux one. The Linux one install makes some assumptions based on the user are a more experienced user.

Without plattaform dependence, the user should take into account that some links maybe are unavailable or outdated at the time you are reading this file. We will try to update this links but if you found some mistake, please notify us.

WINDOWS Installation guide

1. INTRODUCTION

On the installation step we will distinguish two types of installation.

- **End users:** people who just want to run the code and obtain results.
- **Developers:** people who wants to write and modify code and to make tests (a guide for this kind of installation will be done in a future).

This installation guide is intended for Windows 10 64 bit system for simplicity. If you have another windows version or 32 system version, select the right architecture when you download the additional software required. Linux installation guide will be added later

In the case of the **End users** the list of the list of the software require is:

1.1. Software required for **end users:**

- Python 3.7 with the following packages:
  - xarray, numba, h5py netcdf4 vtk
- Paraview

1.2. Software required for **developers:**

- The previous software for end users and:
- Cmake
- Visual Studio
- Intel Fortran Compiler
- Recommended: Anaconda or another python environment

2. INSTALLATION STEPS FOR END USERS

2.1. The MOHID-Lagrangian code

1) Inside a web navigator…
2) Open this link: https://github.com/Mohid-Water-Modelling-System/MOHID-Lagrangian/archive/master.zip (you can open it directly from here with Ctrl key pressed)
3) Once the file is downloaded unzip it wherever you want.

2.2. Paraview

1) Inside a web navigator…
2) Open this link:

   https://www.paraview.org/paraview-downloads/download.php?submit=Download&version=v5.7&type=binary&os=Windows&downloadFile=ParaView-5.7.0-Windows-Python3.7-msvc2015-64bit.exe (you can open it directly from here with control key pressed)

3) Install it. Double click, specify the folder and you can install wherever you want.

2.3. Python for end-users

If you have a python distribution (Anaconda) installed go to the next section. It is mandatory that your python version is 3.X. MOHID-Lagrangian doesn't work with Python 2.7

1. Open a web navigator
2. Open this link https://www.python.org/ftp/python/3.7.4/python-3.7.4-amd64.exe.

   (you can open it directly from here with control key pressed)

3. When you install it, this screen will appear:

4. NOTE: The box in red rectangle sets this python installation as your main system python distribution. It is recommended to check this box if you want a unique functional python and you don't require many different versions of python packages. If you have another python distribution like condas or another python console do not check it to avoid issues.

   If you have no idea about if you should check it or not or you don't understand the previous paragraph you should CHECK IT.

5. Open your windows cmd:
   a. Click on Windows icon
   b. Write: **cmd**
   c. Open it

6. Install all the required packages typing.

a. **pip install xarray numba netcdf4 h5py vtk**

7. You are ready to run MOHID-Lagrangian.

2.4. Python for end-users (if you have an Anaconda distribution installed)

If you have Anaconda installed.

1) Open your windows cmd but linked with your Anaconda Distribution
    a. Windows icon
    b. Write: **anaconda prompt**
    c. Open it
    d. Activate your conda environment with: **conda activate MyEnvironment**
2) Install the required packages typing.
    a. conda install xarray numba netcdf4 h5py vtk
    b. (if you have some problem install packages regarding to permissions open the anaconda prompt with right click and Run as administrator) and try it again.
3) You are ready to run MOHID-Lagrangian

3. RUN A MOHID-Lagrangian test

If python is in your system path (if you use the first python installation way):

1) Go to your MOHID-Lagrangian folder.
2) Open \MOHID-Lagrangian\RUN_Cases
3) Choose a case, go inside the folder case and …
4) Double click on RunCase.bat

If your python distribution is not in your system path:

1) Open your windows cmd but linked with your Anaconda Distribution
    a. Windows icon
    b. Write: **anaconda prompt**
    c. Open it
    d. Activate your conda environment with: conda activate MyEnvironment
2) Go to your MOHID-Lagrangian folder
    a. cd yourpath\MOHID-Lagrangian\RUN_Cases
3) Go inside a case folder.
    a. cd name_of_the_case_folder
4) RunCase.bat

Linux Installation guide

As we mention in the prologue this guide is for non beginner users. It assumes that at least you know how to use a terminal and to install packages at your linux distribution. During this installation steps. I am assuming that I am in the $HOME folder or ~.

1. Download the code

The first step will be to download code with:

>> wget 'https://github.com/Mohid-Water-Modelling-System/MOHID-Lagrangian/archive/master.zip' -0 master.zip

And uncompress it with:

>> unzip master.zip

Or in case you prefer to use git just clone it with:

>> git clone https://github.com/Mohid-Water-Modelling-System/MOHID-Lagrangian.git

2. Install required packages and additional software

2.1 Linux distribution-based software

This installation section was done for Debian based distributions. In case you run a Red-Had based installation or another distribution, please use the respective commands to replace the "apt".

Once it is cloned, we need to compile the libraries. In order to compile them, we need the following requirements:

- Cmake > 3.5
- Autotools, automake
- M4
- Gfortran > 8.0

If you have a recent linux distribution you must do (with admin privileges):

>>apt-get update

>>apt-get install automake autotools-dev m4 cmake gfortran-8

If you have an Intel Fortran compiler (ifort version 16 or higher), you don't need to install the gfortran.

Check the cmake and gfortran versions. Please check them with:

>> cmake –version

>> gfortran –version

To make the things run properly, the cmake version should be higher than 3.5 and gfortran higher than 8. The other packages (autotools, automake and m4) didn't notice a version conflict during the installation.

In case that the CM requirement is not satisfied the compilation will not be run ok and these packages should be installed by hand.

2.2 Cmake by-hand installation (optional)

If your linux distribution do not have these packages versions requirements, you must install them by hand. The cmake package can be install in your home directory with:

>> wget 'https://github.com/Kitware/CMake/releases/download/v3.17.0/cmake-3.17.0-Linux-x86_64.sh' - 0 cmake-3.17.0-Linux-x86_64.sh

>> chmod +x cmake-3.17.0-Linux-x86_64.sh

>> ./cmake-3.17.0-Linux-x86_64.sh

During the installation it gives you the option to specify the installation path. Please select your desire folder. If you don't have admin privileges you must select a folder where you want to install.

2.3 Python distribution

Python 3 is required in order to run property with the following packages:

- Xarray, numba, vtk, h5py, netcdf4

This packages can be installed with:

>> pip install xarray numba vtk h5py netcdf4

In case you run under anaconda environment just do:

>> conda install xarray numba vtk h5py netcdf4

3. Compile the libraries

If you could install everything is time to run the script **MakeLibraries.sh** in MOHID-Lagrangian/ExternalLibs:

If you use gfotran compiler:

>> ~/MOHID-Lagrangian/ExternalLibs/MakeLibraries.sh -gfortran

If you use ifort compiler:

>> ~/MOHID-Lagrangian/ExternalLibs/MakeLibraries.sh -intel

(if you are going to use ifort compiler)

It takes some time to finish, be patient!

4. Compile the MOHID-Lagrangian

Once all the libraries have been compiled. you can compile the MOHID-Lagrangian with:

>> ~/MOHID-Lagrangian/buildScripts/MakeMohidLagrangian.sh

5. Test the MOHID-Lagrangian

Go to one of the test cases inside folder /MOHID-Lagrangian/RUN_Cases and just run

>> ./RunCase.sh