ML-Agents Configuration UI
Documentation

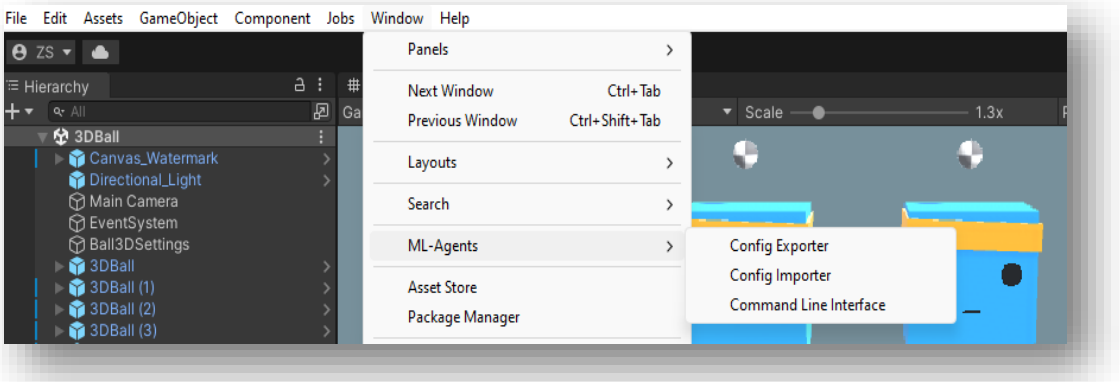
Introduction

This project is a user interface for Unity ML-Agents. It helps the users, that they can create/modify a yaml file inside Unity Editor.
Moreover, the users can run the mlagents-learn command without command line.
Please check the LICENSE in Xardas/ML-Agents Configuration folder.
Furthermore, you can find the source code on Github.

Requirements

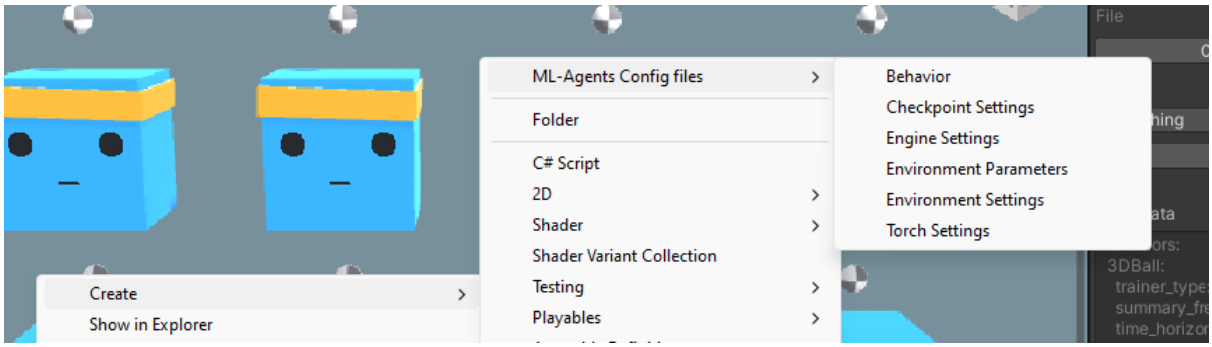
- Unity ML-Agents 2.3 or above / Release 20 or above.
  The last tested version is 4.0.0 / Release 23.
- OS
  - Windows (Tested: Windows 11)
  - Ubuntu (Tested: Ubuntu 20.04)
  - MacOS (Tested: MacOS 13, 14, 15, 26)
- Unity 2021.3.30f1 or later (It is depend on ML-Agents Release)
  The last tested version Unity 6.2.

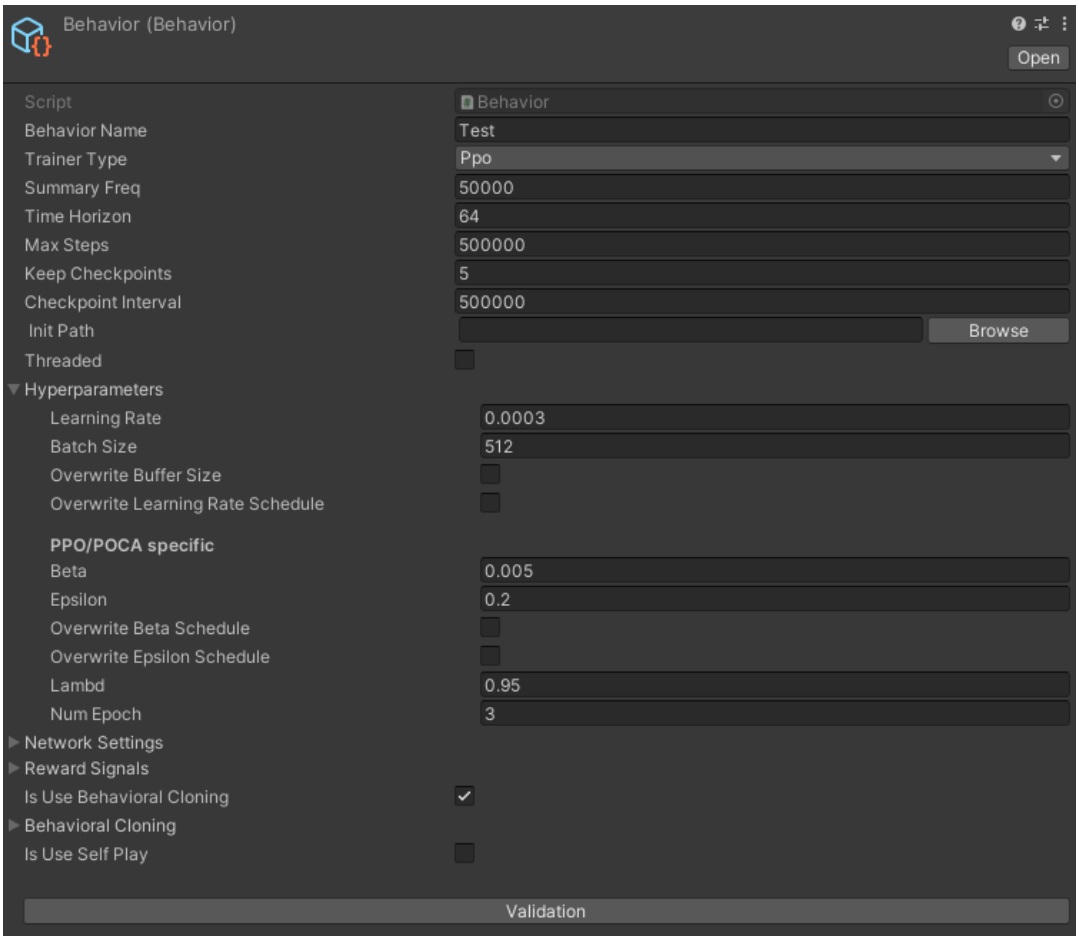
Features

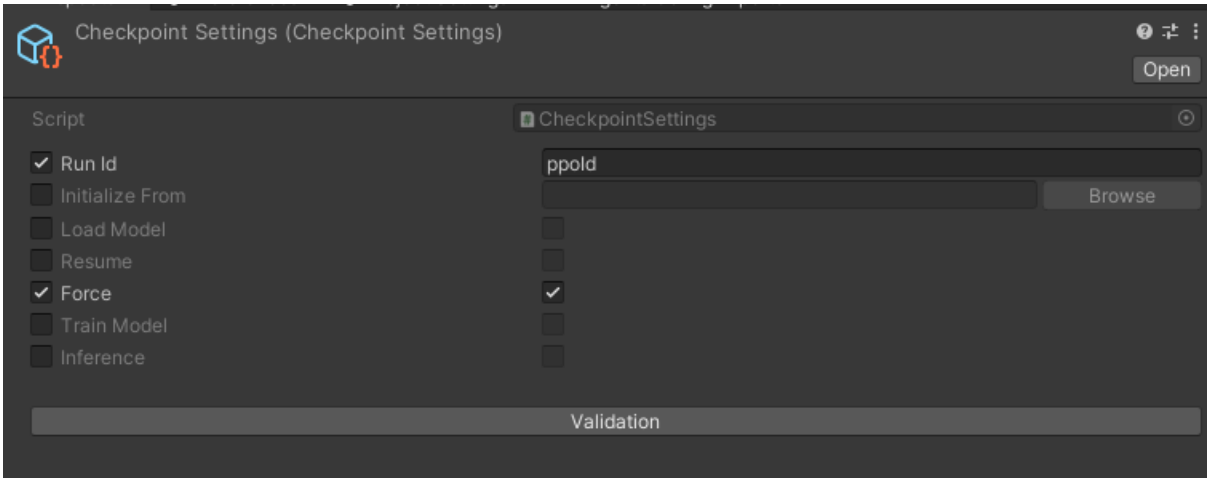


1. Creation

The users can create parts of the ML-Agents config yaml file as ScriptableObject. They contains tooltips, validation. Futhermore, the users can see only the actual fields. For example: If PPO Trainer Type is selected, the specific variables of SAC type is not visible.







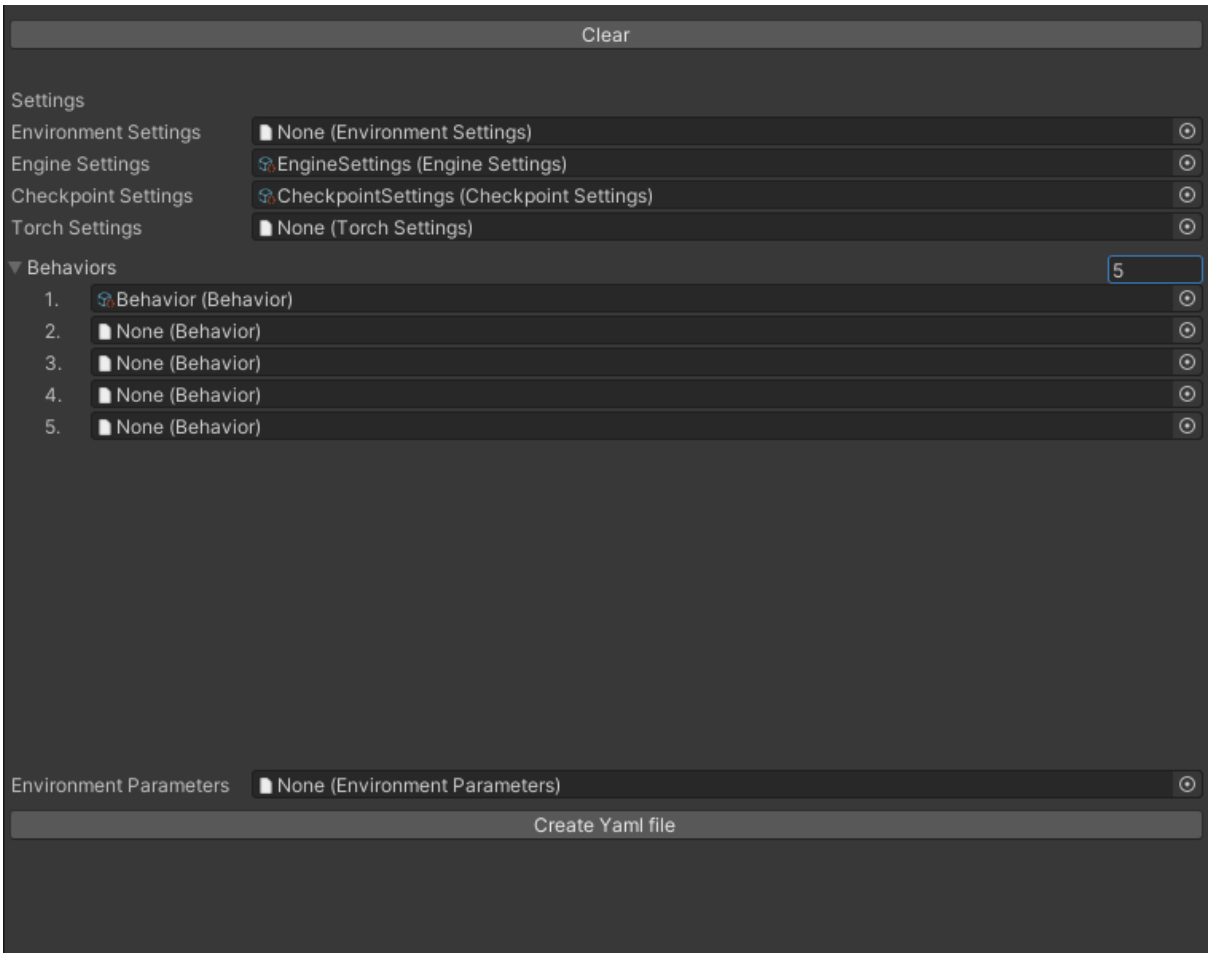
2. Import

The users can import any ML-Agents config yaml file from samples or their own custom project. The files will be created in Assets/Xardas/ML-Agents Configuration/Files folder. (If the folder doesn't exist, it will be created.)

Limit: If the yaml file has default_settings, it will be part of the Behaviors. After that, if the users want to modify the same variable in the Behaviors, they have to update them one by one.

3. Export

The users can export ML-Agents Config ScriptableObjects into a yaml file.

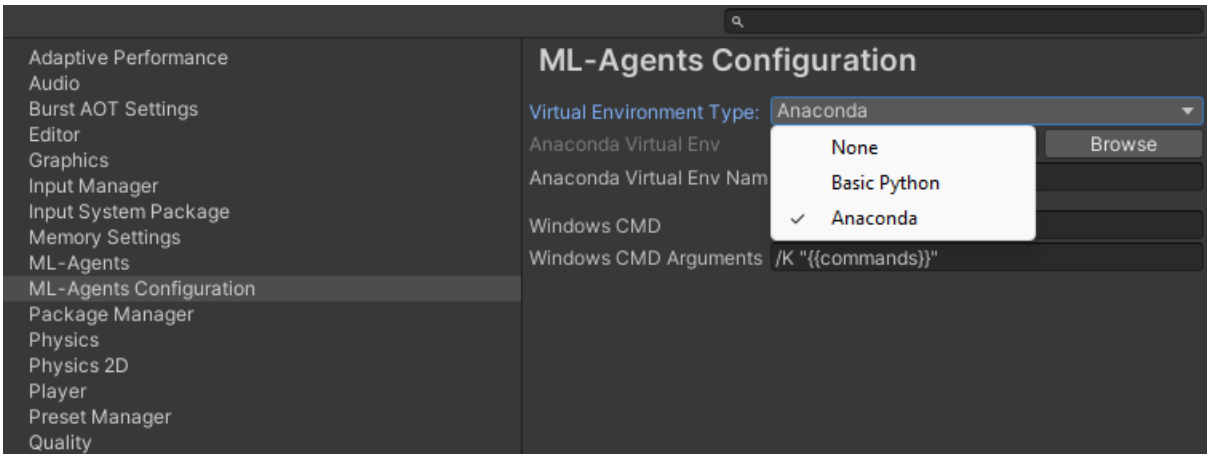


4. Run Command Line

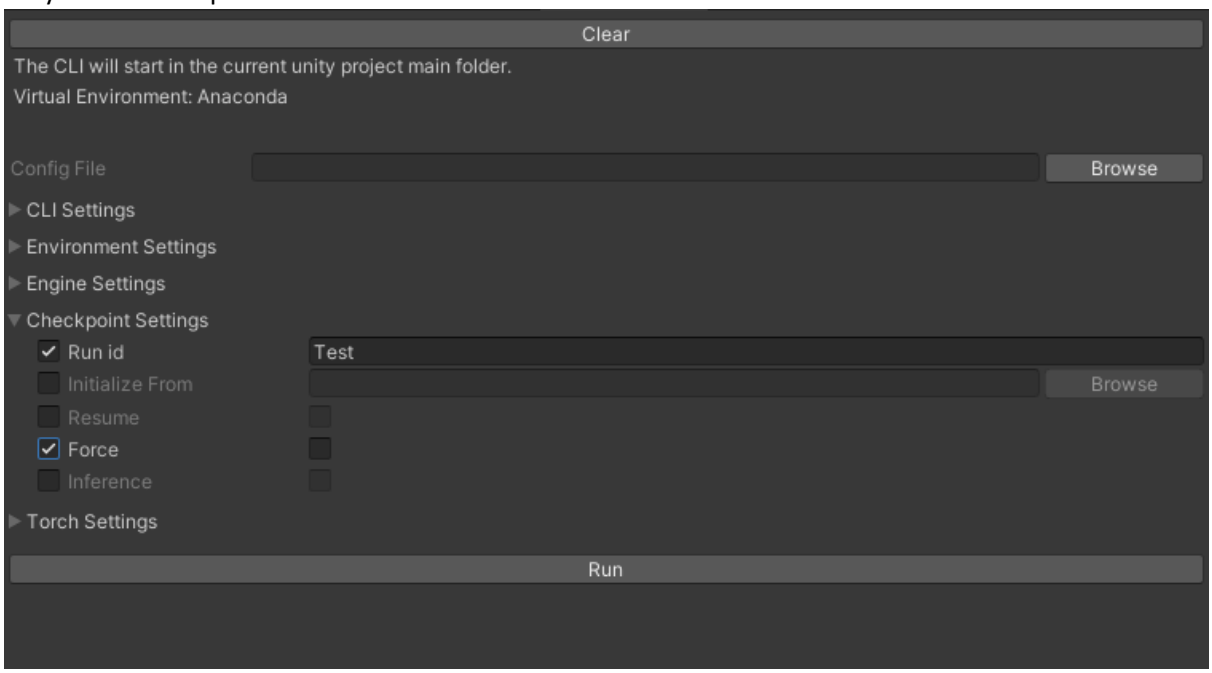
Firstly, it has to be configureated. It is in Project Settings/ML-Agents Configuration. The users can use Python Virtual Environment, if they want.

There are 3 options:

- None: The user don't use Virtual Environment.
- Basic Python: The Virtual Environment, which the Python can create by default. The users have to add the activate file.
- Anaconda: If the users use the Anaconda package management, they have to add the activate file and the name of Virtual Environment.



When the users want to run the mlagents-learn command, they have to select a yaml file and they can add the parameters of command.



Note: By default, the Windows uses CMD, the MacOS uses Terminal and the Linux uses gnome-terminal. On MacOS and on Linux, a mlAgentsCommand.sh file is always generated in the project folder. If the users use Git (or any source control system), they should add this sh file to .gitignore.