
build unknown

DEPRECATION notice

Groot is being rewritten from the ground up (<https://www.behaviortree.dev/groot>). This repository will be in “maintenance mode” and won’t receive any significant update. Issues will **not** be addressed by the main author but, occasionally, Pull Requests **might** be checked and merged.

Groot 1.0 is compatible only with BehaviorTree.CPP 3.8.x, and it is not expected to work correctly with BehaviorTree.CPP 4.x.

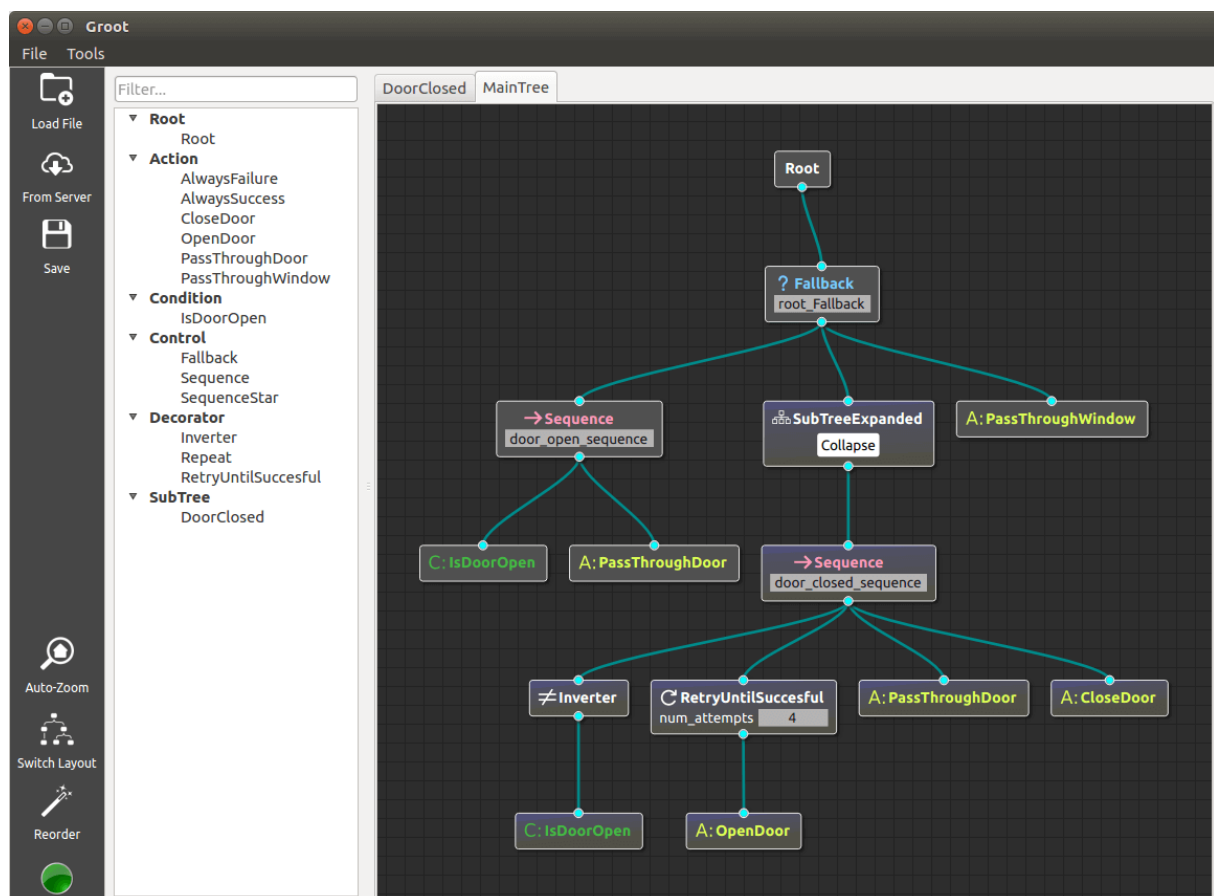
Groot 2.0 introduces many new functionalities and compatibility with BT.CPP 4.0.

If you use Groot at work and you want to know how we are redesigning it to be more flexible, reliable, fast and scalable, get in touch with the main author: dfaconti@aurynrobotics.com.

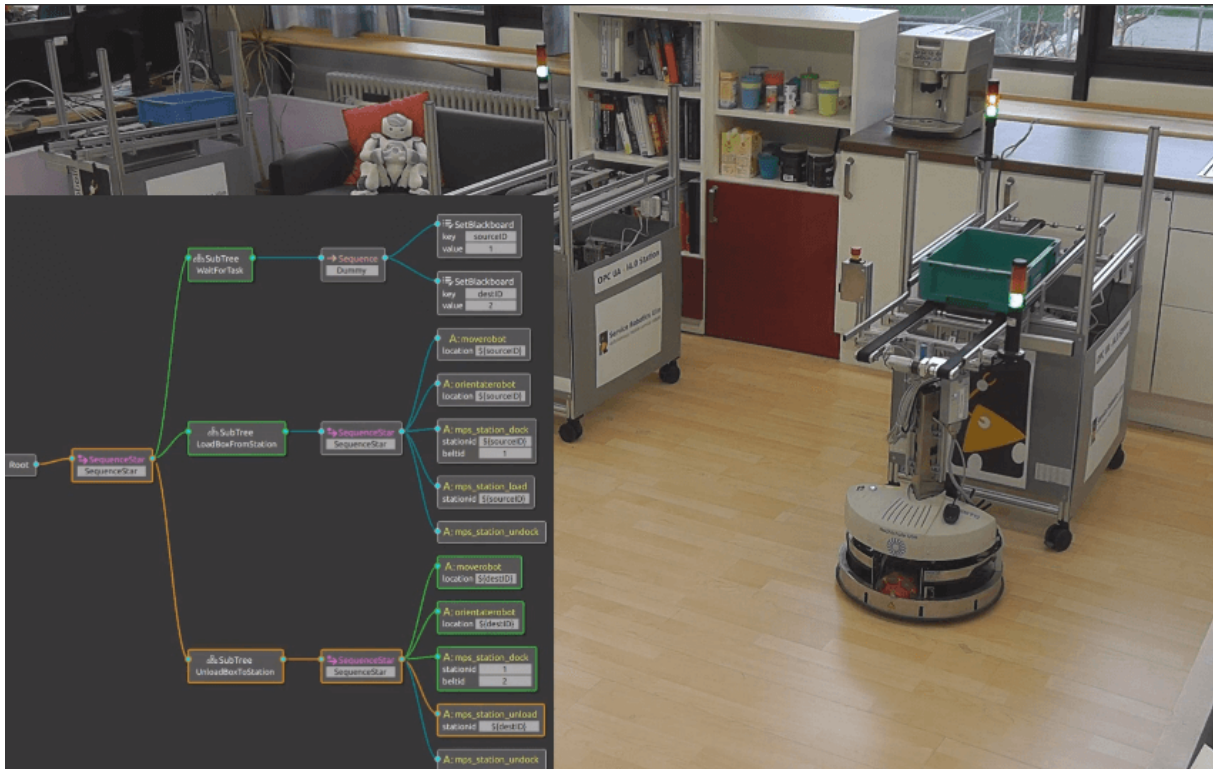
Groot

Groot is a Graphical Editor, written in C++ and Qt, to create BehaviorTrees.

It is compliant with the the library BehaviorTree.CPP.



In the following video you can see how the C++ library and the graphic user interface are used to design and monitor a Behavior Tree.



Dependencies, Installation, and Usage

To compile the project you need:

- CMake 3.2
- Qt5 (tested with version 5.5.1), including the SVG module.

On Ubuntu Xenial or later, you can install the dependencies with:

```
1 sudo apt install qtbase5-dev libqt5svg5-dev libzmq3-dev libdw-dev
```

Some functionalities of the code related to ROS will work **only** if the project is compiled with *catkin*.

Compilation instructions (Linux)

```
1 git clone --recurse-submodules https://github.com/BehaviorTree/Groot.
   git
2 cd Groot
3 cmake -S . -B build
4 cmake --build build
```

Note compiling “in-source” is not allowed.

Compilation instructions (ROS/ROS2)

Discouraged

If you want to compile using `catkin build`(ROS) or `colcon build` (ROS2) then you must be sure that version 3.8.x is used (branch V3.8).

You may probably want to compile BehaviorTree.CPP in the same workspace.

```
1 git clone --branch v3.8 https://github.com/BehaviorTree/BehaviorTree.CPP.git
```

Licence

Copyright (c) 2018-2019 FUNDACIO EURECAT

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Acknowledgment

This project is one of the main components of MOOD2Be, and it is developed at Eurecat by Davide Faconti.

MOOD2Be is one of the six **Integrated Technical Projects (ITPs)** selected from the RobMoSys first open call.

It received funding from the European Union's Horizon 2020 Research and Innovation Programme under the RobMoSys project.