

[DO NOT MERGE YET] Format and test OpenPlanner #1949

New issue

❌ Closed

sgvandijk wants to merge 8 commits into `autowarefoundation:develop` from `sgvandijk:feature/open_planner_format_test`

Conversation 8

Commits 8

Checks 0

Files changed



sgvandijk commented on Feb 1, 2019

Status

DEVELOPMENT

Description

As discussed with [@hatem-darweesh](#), this PR builds on and supersedes [#1818](#). This is an effort to reformat the code of the OpenPlanner and its libraries, and to add tests for this code. Formatting is done using Uncrustify and the [ROS 2 style configuration](#).
A few sentences describing the overall goals of the pull request's commits.

Todos

- ☒ Uncrustify openplanner libraries
- ☐ Uncrustify op_global_planner
- ☐ Uncrustify op_local_planner and related motion planning packages
- ☐ Write tests covering openplanner libraries
- ☐ Write tests covering op_global_planner
- ☐ Write tests covering op_local_planner and related motion planning packages
- ☐ Add documentation to the above packages, at the least for public methods

Reviewers

- amc-nu
- hatem-darweesh

Assignees

No one assigned

Labels

- type:new-feature
- version:autoware-ai

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

























5 participants





Steps to Test or Reproduce

No functionality changes. Tests will be run with `catkin_make run_tests`, and documentation will be generatable using Doxygen.

 **hatem-darweesh** and others added 8 commits
6 years ago



-   - Fix topic name for `lidar_kf_contour_track`
`/cloud_clusters` to `/detec...`  582b3e5
-   - Format `op_utility` code using ROS
C++ style auto format  2e3e4f3
-   Uncrustify openplanner libraries  117baae
-   Rename methods in `op_utility` to be
camelCased  a6d55b7
-   Add TestSuite for `op_utility`, with
tests for non-time functions  48d2ac3
-   PID controller documentation and
tests  d58a602
-   Fix var naming, use initializer list,
don't pass primitives by ref  c17bc2c
-   Integral controller test and fix  46d3dcb

  **gbiggs** requested review from **amc-nu** and **hatem-darweesh** 5 years ago

  **gbiggs** added `type:new-feature` `in progress` labels
on Mar 18, 2019

 **hatem-darweesh** commented on Mar 18, 2019

@sgvandijk thanks for the hard work, but for this PR to be merged I need to modify almost every file in OpenPlanner nodes, which are a lot, then test.
I started working on it, but it will take several days.

  **gbiggs** removed the `in progress` label
on Mar 24, 2019

 **hatem-darweesh** commented on Mar 25, 2019

@sgvandijk could you give me access to this branch "sgvandijk:feature/open_planner_format_test" so I can reflect the function name changes to all nodes.



gbiggs commented on Mar 26, 2019

You can fork the `sgvandijk` repository, make changes to that branch, and make a PR against `sgvandijk`. If that PR is accepted, it will update this PR automatically.



hatem-darweesh commented on Mar 26, 2019

@gbiggs thanks, I will try, although it is not efficient at all.



sgvandijk commented on Mar 26, 2019

Author

Hi **@hatem-darweesh** sorry for the late reply, I haven't been able to work on this much recently, but will pick it up again. Thanks for doing the work of getting through the name changes! It would indeed be great if you could put your changes into a PR against this one. Then I think it's best to probably reduce the scope of this PR (just uncrustify + style maybe) and get it in and then do the rest of the work in smaller chunks/PRs, it looks like there is enough to do and this PR could become very big then. What do you think?



hatem-darweesh commented on Mar 27, 2019

Hello **@sgvandijk**, I agree it is very big PR, specially when all files that uses `op_utility` are changed.

I have a suggestion,

- 1- keep (revert) the old `op_utility` library.
 - 2- add the updated one as new library (file name, class, functions, variables) + testing -> PR 1
 - 3- change `op_library` start using the new utility library replacing the old in packages one by one
 - 3.1 - `op_planner` -> PR 2
 - 3.2 - `op_ros_helpers` & `op_simu` -> PR 3
 - 3.3 - `op_global_planner` -> PR 4
- and so on ..

what do you think ?

once PR 1 is approved I will start working on the rest. and this could be the strategy for the other libraries



gbiggs commented on Mar 27, 2019

@sgvandijk Can you please re-target this to the master branch?



sgvandijk commented on May 15, 2019

Author

I will close this as this is pretty out of date with latest development and approach this with smaller specific PRs.



sgvandijk closed this on May 15, 2019



mitsudome-r added the **version:autoware-ai** label on Jun 14, 2022