GROMACS - Task #1043

Contrib policy

11/21/2012 03:17 AM - Roland Schulz

Status:	Closed
Priority:	Normal
Assignee:	Mark Abraham
Category:	
Target version:	2019
Difficulty:	uncategorized

Description

https://gerrit.gromacs.org/#/c/1675/ showed that we need a contrib policy

Associated revisions

Revision b61de38b - 01/11/2018 11:27 AM - Mark Abraham

Remove contrib directory

Distributing code that can't be compiled is a service to nobody.
Version control means we can't lose whatever value remains.

Moved the build-own-fftw feature to external, which is anyway quite
appropriate for code that core developers (and the original
contributor maintain regularly), and is a very popular feature with
users.

Refs #1043

Change-Id: I964b07756b02068dec8473bc22fe7c465763aad6

History

#1 - 11/21/2012 03:18 AM - Roland Schulz

My suggestion: I think we would do much better with a "contrib" policy which is "it is tested from time to time and thrown out if it doesn't work (can be reinserted by the author after it is fixed)" then a "it just lays their and rots unless someone happens to fix it". Having a smaller contrib folder but one for which one knows the stuff is actually works would be more valuable.

#2 - 11/21/2012 11:50 PM - Szilárd Páll

One thing I would like to emphasize is that even if code is on contrib, it should be possible to build it without modifying source code, otherwise anyone who wants to use a contrib feature (and uses git code) will end up with "diry" version which is not desirable (e.g because we'll reject all bug reports that were obtained with a dirty version).

#3 - 01/17/2013 10:07 PM - Mark Abraham

Ok, but we're not going to treat this policy as meaning we're going to fix problems **because** it's moving to contrib.

I propose to throw out stuff that's been in contrib for (say) 1 year, if it is found to be broken and there's no volunteer to fix it.

So for 4.6.1 some heads might roll.

#4 - 01/17/2013 10:09 PM - Mark Abraham

Also, a README with command to find the commit that removed dead things will be appropriate.

#5 - 11/17/2016 04:32 PM - Mark Abraham

A contrib directory is not useful because the code will rot because it is neither compiled nor tested. Only a version where it worked is meaningful. Someone who has a feature that is either not proposed for the core distribution or has been rejected can still be useful for someone, and finding out if it is worth working on the code to make it potentially acceptable sometimes requires that users can find it and use it.

To that end, there have been suggestions that we make a place where it is easy for the community to share code.

- A gerrit instance is a natural fit for the existing developer community, but gerrit does a poor job of separating the activities of one project from another, so probably we'd want to make another instance. This would be fairly easy to integrate with Jenkins (for those who want that for their branch). This would partly separate GROMACS development into "for release" and "not for release" communities, which has both good and bad aspects to it.

- People with code they want to share can go solve their own problem on e.g. github/gitlab/bitbucket or their organization's webpage, and e.g. use pull-request workflows, but then there would be more of a role for some GROMACS-based wiki-like place that collects links to such unofficial projects that might be interesting to people with similar needs. This could probably also work with Jenkins, but is more work.

Either way, users can find code that is based off the version of GROMACS it was developed from, where it might actually work, rather than e.g. trying to use QM/MM in GROMACS 5.1 and finding that it is broken because the group scheme is no longer the default.

Either approach could also be viable for long-running feature development branches. There's a fundamental problem with such branches because you end up with a large amount of code that is not in a reviewable state. For example (and I'm not picking on Alexei here), you might have PME working on a GPU, but you're still going to need to break things up for review to accept into master branch. Separating off some infrastructure-change patches is often a useful approach in such situations, but now one has to make refactoring patches on the feature branch and then separately compose them into a new commit for inclusion into master branch, and then eventually merge the accepted version back into the feature branch. This avoids massive amounts of rebasing over long periods, while still leaving you with a useful diff against master HEAD to know what chunk might be reasonable to take off next. It's important that we remember that "reviewability" is a feature of code development process, and the longer the development has run without thinking about how to implement reviewability, the higher this deferred cost will be (particularly if other aspects like tests, docs and comments were also being neglected).

It has been suggested that gerrit doesn't handle branches well, despite e.g. we handle release and master branches in gerrit all the time. Someone on a sandbox gerrit can have a branch off whatever gromacs commit they like, and can have whatever policy they like about whether interested parties want to comment on code before or after they submit it to their gerrit branch, or when other gromacs branches might get merged into the development branch. Whether this branch is on a sandbox gerrit or github or a developer's laptop doesn't make it any easier or harder to move from "experimental feature development" to "production-worthy code."

#6 - 03/22/2017 11:29 AM - Mark Abraham

- Assignee set to Mark Abraham

- Target version set to 2018

Nothing currently in src/contrib can possibly work, and nobody on the core team should put effort into making it compile as C++. Does anybody know of anybody who's even thought about using this code in the last couple of years?

I propose removing the contents of src/contrib. It's now very easy for people to fork a GROMACS on e.g. github and share whatever thing they want to share. There's no value to anybody in GROMACS continuing to distribute this graveyard of code nobody seems to know or care about.

#7 - 12/19/2017 03:19 AM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #1043.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
Change-Id: gromacs~master~I964b07756b02068dec8473bc22fe7c465763aad6
Gerrit URL: https://gerrit.gromacs.org/7367

#8 - 12/19/2017 03:19 AM - Mark Abraham

- Target version changed from 2018 to 2019

Removing src/contrib, clearly nobody wants to even talk about it.

#9 - 01/16/2018 05:27 PM - Mark Abraham

- Status changed from New to Resolved

#10 - 01/16/2018 05:27 PM - Mark Abraham

- Status changed from Resolved to Closed