implement force calculation via ForceProviders containing collections of IForceProvider

In order to simplify high-level code (#1793), implement hardware- and locality-aware task parallelism, and expose functionality to APIs (gmxapi, #2229) we need a more flexible framework for code that calculates forces. That is likely to include all such code implementing the IForceProvider interface, and for ForceProviders to arrange for them to be called. For now, that will also still involve t_forcerec. In future, it is hoped the collaboration between the ForceProviders, the available hardware, the state of any auto-tuners, user input, and the integration schedule (#1137) will be able to implement a highly flexible, modular, yet optimizable, run-time framework to replace the hard-coded execution schedules in e.g. do_md() and do_force() (and sub-functions).

This will take a while and the exact endpoint is rather unclear!

Related issues:
- Related to GROMACS - Task #2623: Allow extensible MDModules and forceProviders.
- Related to GROMACS - Task #3040: Refactor Restraint module

Associated revisions

Revision 26ba7a31 - 05/24/2018 06:29 PM - Mark Abraham
Move responsibility for bonded threading decomposition

This is an aspect of force calculation, not of the topology needed for that force calculation.

Removed use of assert no longer needed now that the responsibility has been moved.

Also updated some use of struct keyword.

Refs #2492
Change-Id: lf9d356dc9c4de48b84123e9c432baa84a8335731

Revision 9864b201 - 08/29/2018 02:32 PM - Eric Irrgang
Allow extensible MDModules and forceProviders.

supports gmxapi milestone 6, described at #2585.

MDModules::impl gets a std::vector for (shared) ownership of objects providing the IMDModule interface. An add() method is added to the MDModules public member functions, but the binding protocols are separated into separate issues to allow minimal changes and varying dependencies on other pending changes.

Relates to #1972, #2229, #2492, #2574, #2590.

Refs #2623
Change-Id: lbb16d1453003213a49622810ed8bad4ed4b06e2d

History

#1 - 05/04/2018 12:26 AM - Gerrit Code Review Bot
Gerrit received a related patchset `1` for Issue #2492.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
There will need to be a lot other cleanup as we progress to this point, e.g. that data structures for bonded threading move from t_idef to t_forcerec.

Perhaps some wrapper around do_force and the relax_shell_flexcon will be helpful too.

#3 - 06/04/2018 10:57 PM - Gerrit Code Review Bot
Gerrit received a related patchset ‘1’ for Issue #2492.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
Change-Id: gromacs~master~I3c75d75b13c93d3159dea9a364bfa3258484bd68
Gerrit URL: https://gerrit.gromacs.org/7971

#4 - 08/22/2018 05:03 PM - Gerrit Code Review Bot
Gerrit received a related patchset ‘1’ for Issue #2492.
Uploader: M. Eric Irrgang (ericirrgang@gmail.com)
Change-Id: gromacs~master~Ibb16d1453003213a49622810ed8bad4ed4b06e2d
Gerrit URL: https://gerrit.gromacs.org/8219

#5 - 08/22/2018 05:03 PM - Eric Irrgang
- Related to Task #2623: Allow extensible MDModules and forceProviders. added

#6 - 09/19/2018 03:13 PM - Mark Abraham
- Target version changed from 2019 to future

#7 - 07/18/2019 04:32 PM - Eric Irrgang
- Related to Task #3040: Refactor Restraint module added