

## GROMACS - Task #2538

Feature # 2054 (Accepted): PME on GPU

### organize more of the PME GPU code along task-specific lines

05/31/2018 05:23 PM - Mark Abraham

<b>Status:</b>	New
<b>Priority:</b>	Normal
<b>Assignee:</b>	Mark Abraham
<b>Category:</b>	core library
<b>Target version:</b>	
<b>Difficulty:</b>	uncategorized

#### Description

Once we've merged more of the code in Jenkins, I suggest we reorganize some aspects, e.g. within src/gromacs/ewald we have

- pme-gpu-constants.h (plain C, algorithmic constants used across all GPU implementations)
- some kind of header (not yet clear in my mind) that expresses device-specific aspects like warp size needed for derived constants computed below

For each of (solve, gather, spread):

- solve-launch.h (declares high-level functions for kernel launch, preferably C++, but plain C if it has to be)
- solve-common-constants.h (plain C, computes anything specific to this task derived from the above constants)
- solve-constants.cuh, solve-constants.clh (plain C, computes anything specific to this task for this configuration)
- solve-launch.cpp (builds in all three configs, hopefully very low use of preprocessor, includes all the above headers as appropriate)
- solve-kernel.cu, solve-kernel\_ocl.clh (only device code, builds in the relevant config, includes the matching solve-constants.\*h)
- whatever is necessary to glue the device kernels together in a performant way

This will let us match the tests to the code better, also.

That will leave us with a bunch of high-level workflow and memory-management code in various headers and source files to also reorganize, but grappling with that task will be easier to reason about once we've got the above done.

#### History

#1 - 05/31/2018 05:24 PM - Mark Abraham

- Assignee set to Mark Abraham

#2 - 05/31/2018 05:41 PM - Mark Abraham

- Description updated