**GROMACS - Bug #2041**

mdrun -resetstep can finish too early

08/26/2016 03:49 PM - Berk Hess

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Mark Abraham</td>
</tr>
<tr>
<td>Category:</td>
<td>mdrun</td>
</tr>
<tr>
<td>Target version:</td>
<td>2016.3</td>
</tr>
<tr>
<td>Affected version - extra info:</td>
<td></td>
</tr>
<tr>
<td>Affected version:</td>
<td>2016</td>
</tr>
<tr>
<td>Difficulty:</td>
<td>uncategorized</td>
</tr>
</tbody>
</table>

**Description**

With mdrun -resetstep the run can finish before the reset step is reached and print timings without warning. In contrast, we now generate a fatal error when PME tuning is still active at -resetstep. So we should also exit with a fatal error without printing timings when finishing before -resetstep.

**Related issues:**

- Related to GROMACS - Task #1781: re-design benchmarking functionality
  - Accepted
- Related to GROMACS - Bug #2131: mdrun hangs upon "-nsteps " or "-maxh" trigger...
  - Closed

**Associated revisions**

Revision 1d2d95e3 - 01/30/2017 04:01 PM
- Don't print invalid performance data

If mdrun finished before a scheduled reset of the timing information (e.g. from mdrun -resetstep or mdrun -rethway), then misleading timing information should not be reported.

Fixes #2041

Change-id: l4bd4383c924a342c01e9a3f06b521da128f96a35

**History**

**#1 - 10/31/2016 11:22 AM - Mark Abraham**
- Target version changed from 2016.1 to 2016.2

**#2 - 01/20/2017 04:15 PM - Mark Abraham**
- Assignee set to Mark Abraham

**#3 - 01/20/2017 04:16 PM - Gerrit Code Review Bot**
Gerrit received a related patchset '1' for Issue #2041.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
Change-id: gromacs-release-2016-l4bd4383c924a342c01e9a3f06b521da128f96a35
Gerrit URL: https://gerrit.gromacs.org/6428

**#4 - 01/20/2017 04:18 PM - Mark Abraham**
- Status changed from New to Fix uploaded

I didn't follow Berk's suggestion to add a new fatal error. The fatal error when PME tuning and reset interact is because it isn't valid to continue tuning after a reset, nor does it make sense to interpret the performance data if tuning would be allowed to continue. But in the present case, it suffices to simply not print performance data when it is known that it isn't what the user asked for.

**#5 - 01/30/2017 04:52 PM - Mark Abraham**
- Status changed from Fix uploaded to Resolved

Applied in changeset 1d2d95e3e34c79ca6360739a66f58e6be89270d33.
This fix may break mdrun -resethway with or without PME tuning, because bResetCountersHalfMaxH is then set on all ranks but only ever cleared on the master rank, which might break the new logic at the end of do_md. But perhaps the fix for #2131 also resolves this case.

Yes, looks like -resethway was broken, but now seems to be fixed by the solution for #2131.