

## GROMACS - Bug #3086

### gmxapi fails with MPI build of GROMACS 2020

09/12/2019 11:28 AM - Paul Bauer

<b>Status:</b>	New	
<b>Priority:</b>	Normal	
<b>Assignee:</b>		
<b>Category:</b>		
<b>Target version:</b>		
<b>Affected version - extra info:</b>	<a href="https://gerrit.gromacs.org/c/gromacs/+13088">https://gerrit.gromacs.org/c/gromacs/+13088</a>	<b>Difficulty:</b> uncategorized
<b>Affected version:</b>	git master	

#### Description

When building GROMACS 2020 in release mode and MPI enabled and trying to use it with gmxapi, I get the following error

```
Python 3.7.4+ (default, Sep 4 2019, 08:03:05)
[GCC 9.2.1 20190827] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import gmxapi as gmx
>>> test = gmx.mdrun("/home/acmpnv/data/gerrit/refactor/src/gromacs/trajectoryanalysis/tests/clust
size.tpr")
>>> output = test.run()
```

```
-----
Program:      GROMACS, version 2020-dev-20190911-2e9e083d6a
Source file:  src/gromacs/gmxlib/network.cpp (line 70)
*** The MPI_Comm_size() function was called before MPI_INIT was invoked.
*** This is disallowed by the MPI standard.
*** Your MPI job will now abort.
[debian-xps13:19677] Local abort before MPI_INIT completed successfully, but am not able
to aggregate error messages, and not able to guarantee that all other processes were killed!
```

#### GROMACS CMake flags

```
cmake .. -DCMAKE_BUILD_TYPE=Release -DCMAKE_CXX_COMPILER=clang++-7 -DCMAKE_C_COMPILER=clang-7 -DGM
XAPI=ON -DGMX_HWLOC=AUTO -DGMX_USE_RDTSCP=DETECT -DGMX_MPI=ON '-DCMAKE_CXX_LINK_FLAGS=-Wl,-rpath,/
usr/bin/./lib64 -L/usr/bin/./lib64' -DGMX_GPLUSPLUS_PATH=/usr/bin/gcc-5 -DCMAKE_INSTALL_PREFIX=/
home/acmpnv/data/gerrit/refactor/install-clang-7-simd-mpi -DGMX_GPU=OFF
```

#### History

##### #1 - 09/25/2019 01:24 PM - Eric Irrgang

Can you provide the command line you used or a reproducible test case / Dockerfile? I would only expect to see this if mpiexec or similar wrapper were not called.

##### #2 - 09/25/2019 04:18 PM - Eric Irrgang

I have not been able to reproduce this error, but I was able to produce a similar error when executing Python **with** mpiexec. It seems like the gmx\_mpi binary tries to initialize MPI, but runs into trouble because it is a subprocess of a subprocess of mpiexec. I don't know how to resolve that easily, other than to disallow gmx\_mpi when wrapping command line tools. (Thoughts?) It would also make sense to replace the use case of wrapped gmx command lines with direct C++ access to the tool launcher.

In the case of mdrun, specifically, I thought I had a RAII/sentinel object in place as part of the simulation launcher to make sure MPI was properly initialized and deinitialized, but there has been some tinkering in that machinery and we never resolved the issue of MPI communicator sharing. As such, behavior is undefined when launching a script with more than 1 MPI rank and MPI-enabled libgromacs. 1 MPI rank should work, though, so we should figure out what's going on if it doesn't.