Running the MPAS framework on the cluster

> The Model for Prediction Across Scales (MPAS) is a collaborative project for developing atmosphere, ocean and other earth-system simulation components for use in climate, regional climate and weather studies.

Compilation

First of all define a folder ${WORK} on the /work or the /scratch filesystem (somewhere where you have write permissions):

```
export WORK=/work/FAC/...
mkdir -p ${WORK}
```

Load the following relevant modules:

```
module load gcc/10.4.0
module load mvapich2/2.3.7
module load parallel-netcdf/1.12.2
module load parallelio/2.5.9-mpi

export PIO=$PARALLELIO_ROOT
export PNETCDF=$PARALLEL_NETCDF_ROOT
```

Download the MPAS framework:

```
cd ${WORK}
git clone https://github.com/MPAS-Dev/MPAS-Model
```

Patch the MPAS Makefile:

```
sed -i 's/-ffree-form/-ffree-form -fallow-argument-mismatch/' ${WORK}/MPAS-Model/Makefile
```

Compile:

```
cd ${WORK}/MPAS-Model

make gfortran CORE=init_atmosphere AUTOCLEAN=true PRECISION=single OPENMP=true USE_PIO2=true
make gfortran CORE=atmosphere AUTOCLEAN=true PRECISION=single OPENMP=true USE_PIO2=true
```

Running a basic global simulation

Here we aim at running a basic global simulation, just to test that the framework runs. we need to proceed in three steps:

1. Process time-invariant fields, which will be interpolated into a given mesh, this step produces a "static" file
2. Interpolating time-varying meteorological and land-surface fields from intermediate files (produced by the
   ungrib component of the WRF Pre-processing System), this step produces an "init" file
3. Run the basic simulation

Create the run folder and link to the binary files

```
cd ${WORK}
mkdir -p run
cd run
ln -s ${WORK}/MPAS-Model/init_atmosphere_model
ln -s ${WORK}/MPAS-Model/atmosphere_model
```

Get the mesh files

```
cd ${WORK}
wget https://www2.mmm.ucar.edu/projects/mpas/atmosphere_meshes/x1.40962.tar.gz
cd run
tar xvzf ../x1.40962.tar.gz
```

Create the configuration files for the "static" run

The `namelist.init_atmosphere` file:

```
cat << EOF > ${WORK}/run/namelist.init_atmosphere
&nhyd_model
config_init_case = 7
/
&data_sources
```

```
config_geog_data_path = '${WORK}/WPS_GEOG/'
config_landuse_data = 'MODIFIED_IGBP_MODIS_NOAH'
config_topo_data = 'GMTED2010'
config_vegfrac_data = 'MODIS'
config_albedo_data = 'MODIS'
config_maxsnowalbedo_data = 'MODIS'
/
&preproc_stages
config_static_interp = true
config_native_gwd_static = true
config_vertical_grid = false
config_met_interp = false
config_input_sst = false
config_frac_seaice = false
/
EOF
```

The `streams.init_atmosphere` file:

```
cat << EOF > ${WORK}/run/streams.init_atmosphere
<streams>
<immutable_stream name="input"
                  type="input"
                  precision="single"
                  filename_template="x1.40962.grid.nc"
                  input_interval="initial_only" />

<immutable_stream name="output"
                  type="output"
                  filename_template="x1.40962.static.nc"
                  packages="initial_conds"
                  output_interval="initial_only" />
</streams>
EOF
```

Proceed to the "static" run

You will need to make sure that the folder `${WORK}/WPS_GEOG` exists and contains all the appropriate data.

First create a `start_mpas_init.sbatch` file (carefully replace on line #4 `ACCOUNT_NAME` by your actual project name and on line #6 appropriately type your e-mail address, or double-comment with an additional `#` if you don't wish to receive job notifications):

```
cat << EOF > ${WORK}/run/start_mpas_init.sbatch
#!/bin/bash -l

#SBATCH --account ACCOUNT_NAME
#SBATCH --mail-type ALL
#SBATCH --mail-user <first.lastname>@unil.ch

#SBATCH --chdir ${WORK}/run
#SBATCH --job-name mpas_init
#SBATCH --output=mpas_init.job.%j

#SBATCH --partition cpu

#SBATCH --nodes 1
#SBATCH --ntasks 1
#SBATCH --cpus-per-task 1
#SBATCH --mem 8G
#SBATCH --time 00:59:59
#SBATCH --export ALL

module load gcc/10.4.0
module load mvapich2/2.3.7
module load parallel-netcdf/1.12.2
module load parallelio/2.5.9-mpi

export PIO=\$PARALLELIO_ROOT

srun ./init_atmosphere_model
EOF
```

Now start the job with `sbatch start_mpas_init.sbatch` and at the end of the run, make sure that the log file `${WORK}/run/log.init_atmosphere.0000.out` displays no error.

Create the configuration files for the "init" run

The `namelist.init_atmosphere` file:

```
cat << EOF > ${WORK}/run/namelist.init_atmosphere
&nhyd_model
config_init_case = 7
config_start_time = '2014-09-10_00:00:00'
/
&dimensions
config_nvertlevels = 55
config_nsoillevels = 4
config_nfglevels = 38
config_nfgsoillevels = 4
/
&data_sources
config_met_prefix = 'GFS'
config_use_spechumd = false
/
&vertical_grid
config_ztop = 30000.0
config_nsmterrain = 1
config_smooth_surfaces = true
config_dzmin = 0.3
config_nsm = 30
config_tc_vertical_grid = true
config_blend_bdy_terrain = false
/
&preproc_stages
config_static_interp = false
config_native_gwd_static = false
config_vertical_grid = true
config_met_interp = true
config_input_sst = false
config_frac_seaice = true
/
EOF
```

The `streams.init_atmosphere` file:

```
cat << EOF > ${WORK}/run/streams.init_atmosphere
<streams>
<immutable_stream name="input"
                  type="input"
```

```
                filename_template="x1.40962.static.nc"
                input_interval="initial_only" />

<immutable_stream name="output"
                type="output"
                filename_template="x1.40962.init.nc"
                packages="initial_conds"
                output_interval="initial_only" />
</streams>
EOF
```

Proceed to the "init" run

Just start again the job with `sbatch start_mpas_init.sbatch` and at the end of the run, make sure that the log file `${WORK}/run/log.init_atmosphere.0000.out` displays no error.

Create the configuration file for the global simulation

The `namelist.atmosphere` file:

```
cat << EOF > ${WORK}/run/namelist.atmosphere
&nhyd_model
   config_time_integration_order = 2
   config_dt = 720.0
   config_start_time = '2014-09-10_00:00:00'
   config_run_duration = '0_03:00:00'
   config_split_dynamics_transport = true
   config_number_of_sub_steps = 2
   config_dynamics_split_steps = 3
   config_h_mom_eddy_visc2 = 0.0
   config_h_mom_eddy_visc4 = 0.0
   config_v_mom_eddy_visc2 = 0.0
   config_h_theta_eddy_visc2 = 0.0
   config_h_theta_eddy_visc4 = 0.0
   config_v_theta_eddy_visc2 = 0.0
   config_horiz_mixing = '2d_smagorinsky'
   config_len_disp = 120000.0
   config_visc4_2dsmag = 0.05
   config_w_adv_order = 3
   config_theta_adv_order = 3
   config_scalar_adv_order = 3
```

```
    config_u_vadv_order = 3
    config_w_vadv_order = 3
    config_theta_vadv_order = 3
    config_scalar_vadv_order = 3
    config_scalar_advection = true
    config_positive_definite = false
    config_monotonic = true
    config_coef_3rd_order = 0.25
    config_epssm = 0.1
    config_smdiv = 0.1
/
&damping
    config_zd = 22000.0
    config_xnutr = 0.2
/
&limited_area
    config_apply_lbcs = false
/
&io
    config_pio_num_iotasks = 0
    config_pio_stride = 1
/
&decomposition
    config_block_decomp_file_prefix = 'x1.40962.graph.info.part.'
/
&restart
    config_do_restart = false
/
&printout
    config_print_global_minmax_vel = true
    config_print_detailed_minmax_vel = false
/
&IAU
    config_IAU_option = 'off'
    config_IAU_window_length_s = 21600.
/
&physics
    config_sst_update = false
    config_sstdiurn_update = false
    config_deepsoiltemp_update = false
```

```
    config_radtlw_interval = '00:30:00'
    config_radtsw_interval = '00:30:00'
    config_bucket_update = 'none'
    config_physics_suite = 'mesoscale_reference'
/
&soundings
    config_sounding_interval = 'none'
/
EOF
```

The `streams.atmosphere` file:

```
cat << 'EOF' > ${WORK}/run/streams.atmosphere
<streams>
<immutable_stream name="input"
                  type="input"
                  filename_template="x1.40962.init.nc"
                  input_interval="initial_only" />

<immutable_stream name="restart"
                  type="input;output"
                  filename_template="restart.$Y-$M-$D_$h.$m.$s.nc"
                  input_interval="initial_only"
                  output_interval="1_00:00:00" />

<stream name="output"
        type="output"
        filename_template="history.$Y-$M-$D_$h.$m.$s.nc"
        output_interval="6:00:00" >
</stream>

<stream name="diagnostics"
        type="output"
        filename_template="diag.$Y-$M-$D_$h.$m.$s.nc"
        output_interval="3:00:00" >
</stream>

<immutable_stream name="iau"
                  type="input"
                  filename_template="x1.40962.AmB.$Y-$M-$D_$h.$m.$s.nc"
```

```
            filename_interval="none"
            packages="iau"
            input_interval="initial_only" />

<immutable_stream name="lbc_in"
            type="input"
            filename_template="lbc.$Y-$M-$D_$h.$m.$s.nc"
            filename_interval="input_interval"
            packages="limited_area"
            input_interval="none" />

</streams>
EOF
```

Run the whole simulation

You will need to copy relevant data to the run folder:

```
cp ${WORK}/MPAS-Model/{GENPARM.TBL,LANDUSE.TBL,OZONE_DAT.TBL,OZONE_LAT.TBL,OZONE_PLEV.TBL,RRTMG_LW_DATA,RRTMG_SW_DATA,SOILPARM.TBL,VEGPARM.TBL} ${WORK}/run/.
```

Then create a `start_mpas.sbatch` file (carefully replace on line #4 `ACCOUNT_NAME` by your actual project name and on line #6 appropriately type your e-mail address, or double-comment with an additional `#` if you don't wish to receive job notifications):

```
cat << EOF > ${WORK}/run/start_mpas.sbatch
#!/bin/bash -l

#SBATCH --account ACCOUNT_NAME
#SBATCH --mail-type ALL
#SBATCH --mail-user <first.lastname>@unil.ch

#SBATCH --chdir ${WORK}/run
#SBATCH --job-name mpas_init
#SBATCH --output=mpas_init.job.%j

#SBATCH --partition cpu

#SBATCH --nodes 1
#SBATCH --ntasks 1
```

```
#SBATCH --cpus-per-task 16
#SBATCH --mem 8G
#SBATCH --time 00:59:59
#SBATCH --export ALL

module load gcc/10.4.0
module load mvapich2/2.3.7
module load parallel-netcdf/1.12.2
module load parallelio/2.5.9-mpi

export PIO=\$PARALLELIO_ROOT

srun ./atmosphere_model
EOF
```

Now start the job with `sbatch start_mpas.sbatch` and at the end of the run, make sure that the log file `${WORK}/run/log.atmosphere.0000.out` displays no error.

Révision #4
Créé 10 mars 2023 16:25:40 par Flavio Calvo
Mis à jour 18 décembre 2023 14:30:46 par Flavio Calvo