Coral, a home made workflow system tool to manage numerical climate simulations

HPC Workflow 2022

François Klein & Charles Pelletier

Climate models

The climate system is governed by physical principles

$$\frac{d\vec{v}}{dt} = -\frac{1}{\rho}\vec{\nabla}p + \vec{g} + \vec{F}_{fric} - 2\vec{\Omega} \times \vec{v} \tag{1}$$

$$\frac{\partial \rho}{\partial t} = -\vec{\nabla} \cdot (\rho \vec{v}) \tag{2}$$

$$\frac{\partial \rho q}{\partial t} = -\vec{\nabla} \cdot (\rho \vec{v} q) + \rho(E - C) \tag{3}$$

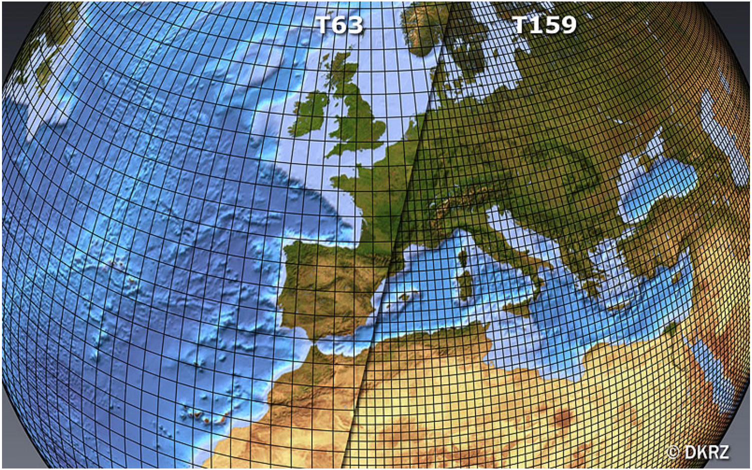
$$c_p \frac{dT}{dt} = Q + \frac{1}{\rho}\frac{dp}{dt} \tag{4}$$

$$\dots \tag{5}$$

Climate models

The climate system is governed by physical principles

that must be discretized in time and space



Climate models

The climate system is governed by physical principles

that must be discretized in time and space

so that it can be numerically simulated

```
ALLOCATE(zn0_sto(1:2*nn_sto_eos))
ALLOCATE(zn_sto(1:2*nn_sto_eos))
ALLOCATE(zsign(1:2*nn_sto_eos))
DO jsmp = 1, 2*nn_sto_eos, 2
  zsign(jsmp)   = 1._wp
  zsign(jsmp+1) = -1._wp
END DO
!
DO jk = 1, jpkm1
   DO jj = 1, jpj
      DO ji = 1, jpi
         !
         ! compute density (2*nn_sto_eos) times:
         ! (1) for t+dt, s+ds (with the random TS fluctutation computed in sto_pts)
         ! (2) for t-dt, s-ds (with the opposite fluctuation)
         DO jsmp = 1, nn_sto_eos*2
            jdof   = (jsmp + 1) / 2
            zh     = pdep(ji,jj,jk) * r1_Z0                                   ! depth
            zt     = (pts (ji,jj,jk,jp_tem) + pts_ran(ji,jj,jk,jp_tem,jdof) * zsign(jsmp)) * r1_T0    ! temperature
            zstemp = pts  (ji,jj,jk,jp_sal) + pts_ran(ji,jj,jk,jp_sal,jdof) * zsign(jsmp)
            zs     = SQRT( ABS( zstemp + rdeltaS ) * r1_S0 )   ! square root salinity
         END DO
         !
         ! compute stochastic density as the mean of the (2*nn_sto_eos) densities
         prhop(ji,jj,jk) = 0._wp ; prd(ji,jj,jk) = 0._wp
         DO jsmp = 1, nn_sto_eos*2
            prhop(ji,jj,jk) = prhop(ji,jj,jk) + zn0_sto(jsmp)                    ! potential density referenced at the surface
            !
            prd(ji,jj,jk) = prd(ji,jj,jk) + (  zn_sto(jsmp) * r1_rau0 - 1._wp  )   ! density anomaly (masked)
         END DO
         prhop(ji,jj,jk) = 0.5_wp * prhop(ji,jj,jk) * ztm / nn_sto_eos
         prd  (ji,jj,jk) = 0.5_wp * prd  (ji,jj,jk) * ztm / nn_sto_eos
      END DO
   END DO
END DO
DEALLOCATE(zn0_sto,zn_sto,zsign)
```

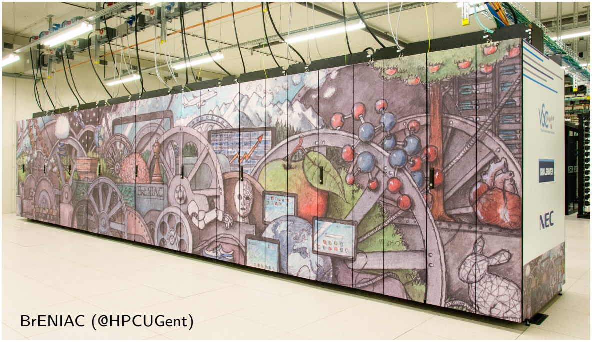
Climate models

The climate system is governed by physical principles

that must be discretized in time and space

so that it can be numerically simulated

using supercomputers



BrENIAC (@HPCUGent)

Climate models and HPC

The accuracy of a climate model simulation will depend on :

Climate models and HPC

The accuracy of a climate model simulation will depend on :

- The complexity of the model (components included, numerical core, ...)

Climate models and HPC

The accuracy of a climate model simulation will depend on :

- The complexity of the model (components included, numerical core, ...)
- The spatial and temporal resolutions of the model

Climate models and HPC

The accuracy of a climate model simulation will depend on :

- The complexity of the model (components included, numerical core, ...)
- The spatial and temporal resolutions of the model
- The spatial and temporal scales of the investigated physical processes
- ...

Climate models and HPC

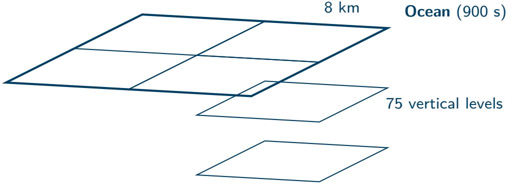
The accuracy of a climate model simulation will depend on :

- The complexity of the model (components included, numerical core, ...)
- The spatial and temporal resolutions of the model
- The spatial and temporal scales of the investigated physical processes
- ...

Climate model performance directly depends on the HPC facility performance

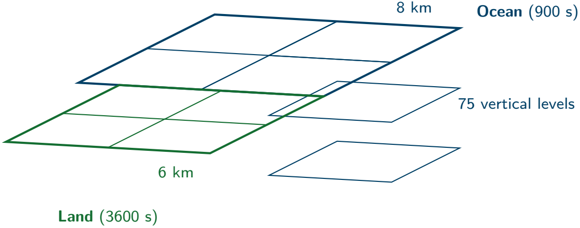
Currently running on BrENIAC

Simulation with the model NEMO-CCLM^2-F* over Antarctica :



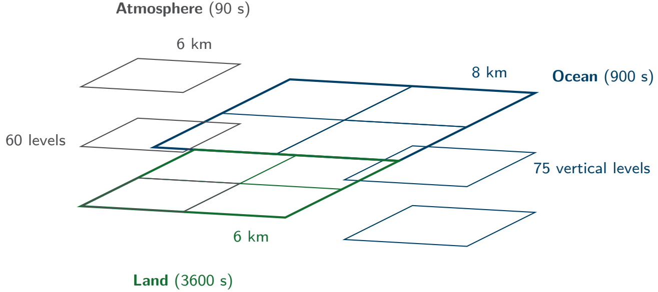
Currently running on BrENIAC

Simulation with the model NEMO-CCLM^2-F* over Antarctica :



Currently running on BrENIAC

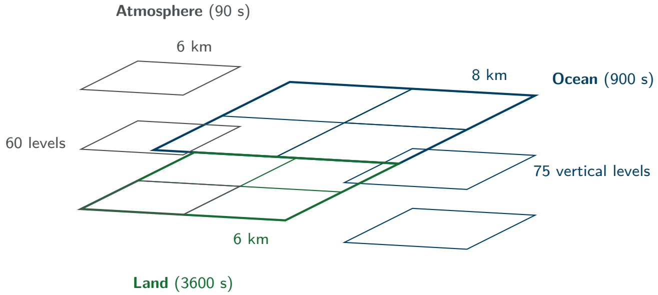
Simulation with the model NEMO-CCLM^2-F* over Antarctica :



*Pelletier et al., Geosci. Model Dev., 15, 553–594, 2022

Currently running on BrENIAC

Simulation with the model NEMO-CCLM^2-F* over Antarctica :



$+$ sea ice and ice sheet components.

- Huge amount of calculations to be performed.

*Pelletier et al., Geosci. Model Dev., 15, 553–594, 2022

Currently running on BrENIAC

Simulation with the model NEMO-CCLM^2-F* over Antarctica :

Simulated time: 1985 2015 2022

using 224 cores

Computer time: day 0 (10/12/2021) day 66

*Pelletier et al., Geosci. Model Dev., 15, 553–594, 2022

Currently running on BrENIAC

Simulation with the model NEMO-CCLM^2-F* over Antarctica :

The main HPC-related constraint is the walltime (max 3 days)

Coral allows for chunking an overwhelming large job (+60 days) into a sequence of shorter segments

*Pelletier et al., Geosci. Model Dev., 15, 553–594, 2022

Coral

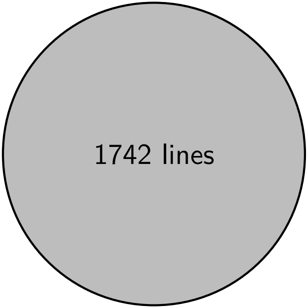
Home-made tool written in bash and hosted in our private git plateform, which allows for :

1. Initiating a set of working configuration files
   `coral init`
2. Creating a submission script from the configuration files
   `coral build`
3. Submitting a linear sequence of automatically launched jobs
   `coral submit`
4. Displaying the current status of the simulation
   `coral status`

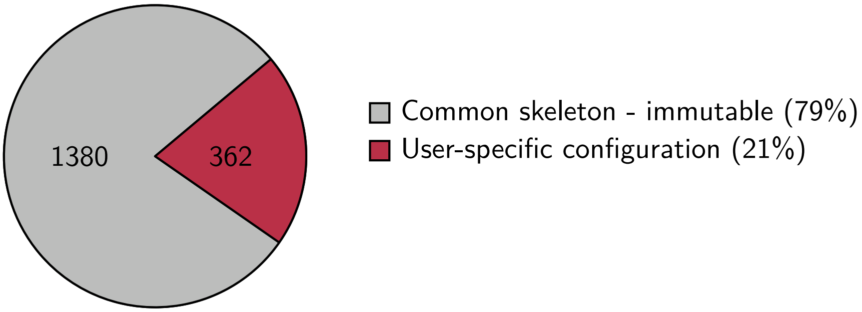
in a **clean** and **flexible** way.

Submission script

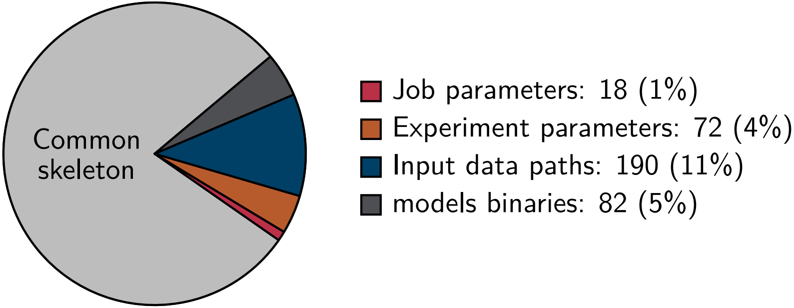
Submission script



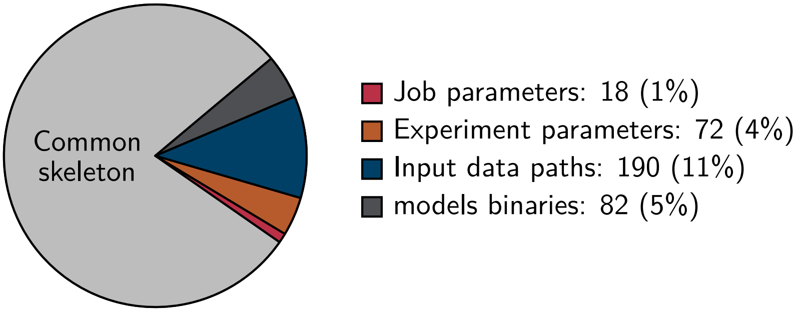
Submission script



Submission script



Submission script



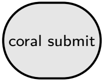
1. `coral init`

   copy the four machine-specific configuration files (from existing templates) into the submission directory

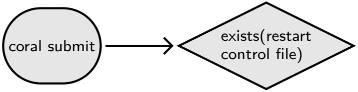
2. `coral build`

   merge them and append them to the skeleton to create one submission script

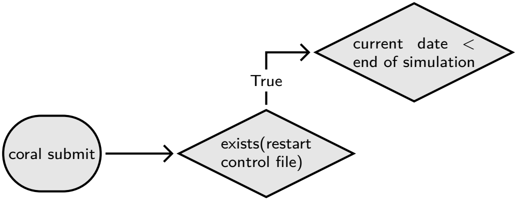
Submit the job



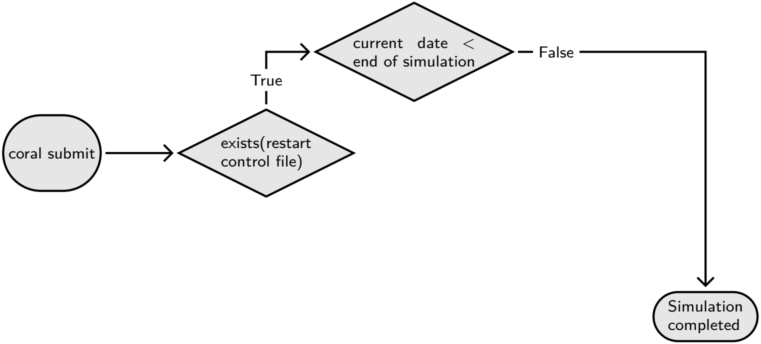
Submit the job



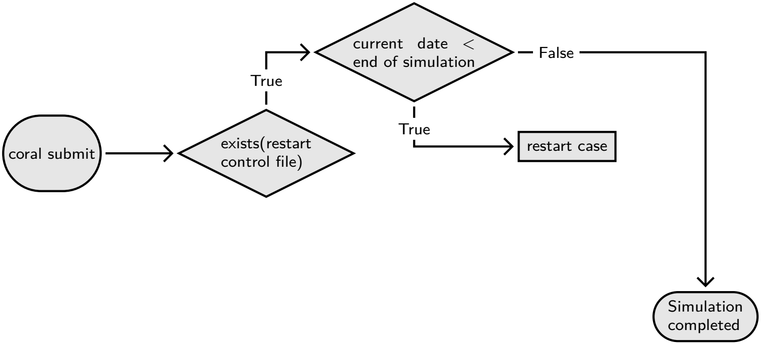
Submit the job



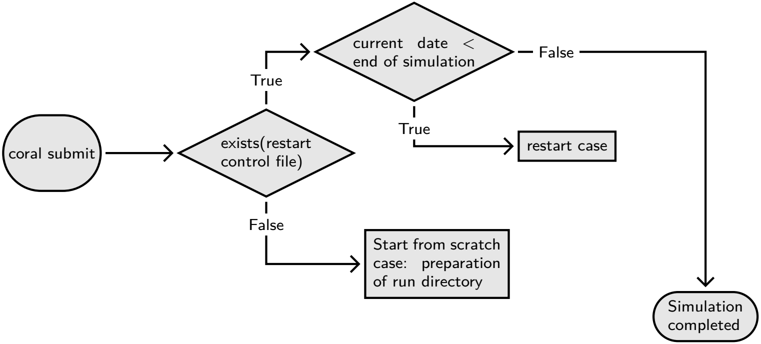
Submit the job



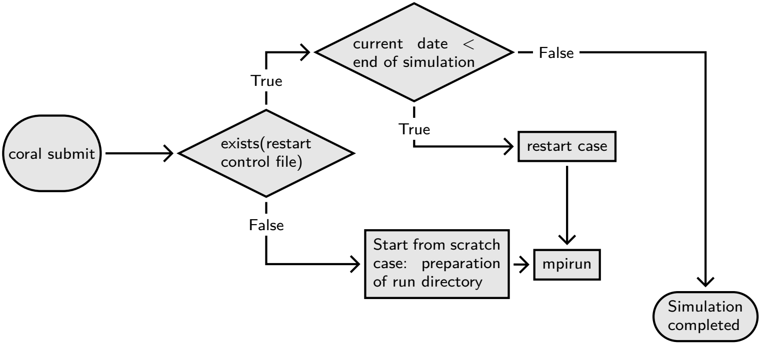
Submit the job



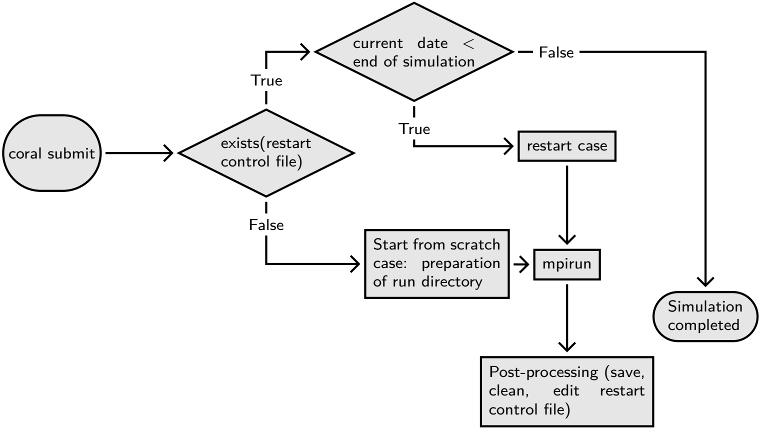
Submit the job



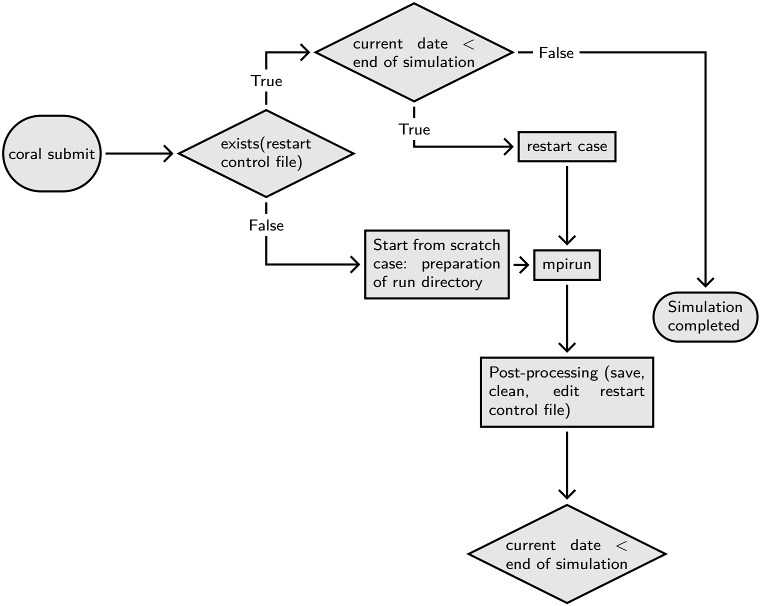
Submit the job



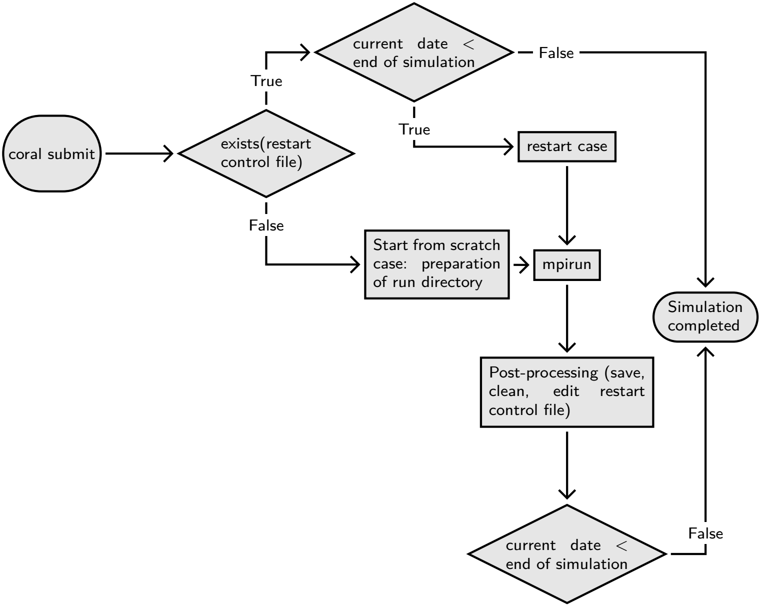
Submit the job



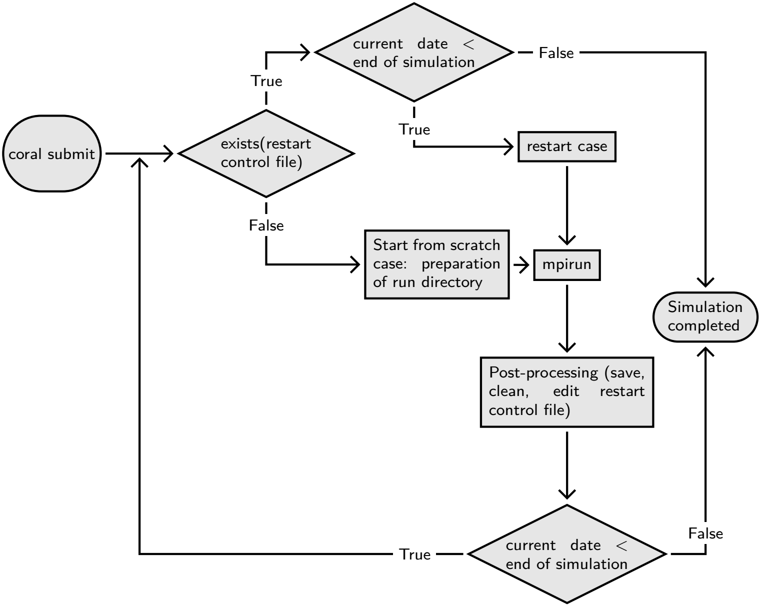
Submit the job



Submit the job



Submit the job



Take home message

- Climate model simulations involve **massively parallel jobs**
- 2 challenges need to be faced when launching a simulation :
  1. A huge amount of data and information is required, making the submission script long and potentially dirty
  2. On most supercomputers, the maximum walltime is a limiting factor
- Coral allows for easily and cleanly **setting up** an experiment and for **submitting** a sequence of automatically launched jobs, until the end of the simulation is reached