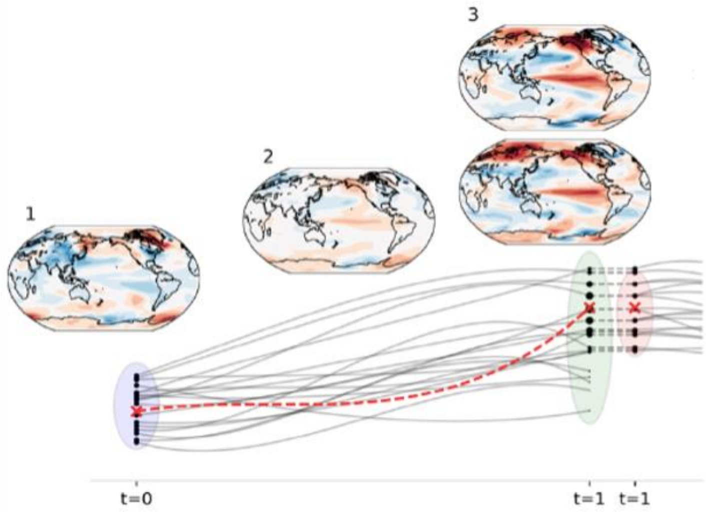


Ensemble simulations with LMDZ (and IPSL-CM) model

Arnaud Caubel (LSCE), Yann Meurdesoif (LSCE)

and M. Khodri (LOCEAN), R. Noyelle (LSCE)

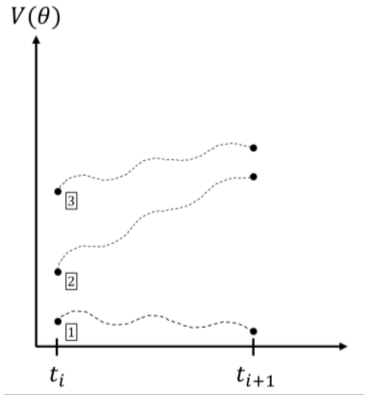
October, 3rd 2023, Journée des Utilisateurs de LMDZ

Context : Ensembles with LMDZ model

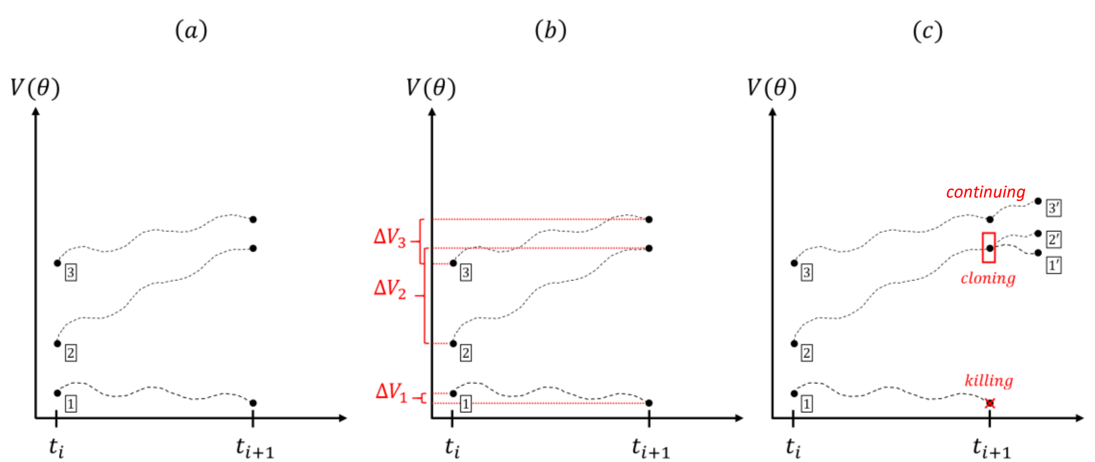
Starting point

- Discussion with Myriam Khodri
- Wish to run several (~ 80) instances/members of IPSL-CM over a period…
- …and periodically select and clone (with perturbation) trajectories which go in the favored direction (measured by a score function) and kill the other ones.

Standard simulations



Ensemble simulations

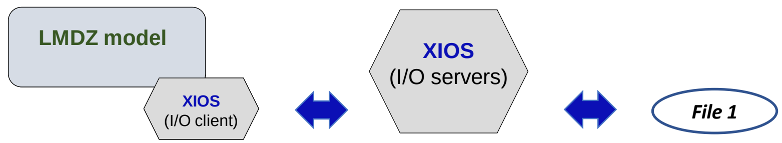


R. Noyelle ©

Context : Ensembles with LMDZ model

Launch many members of LMDZ model

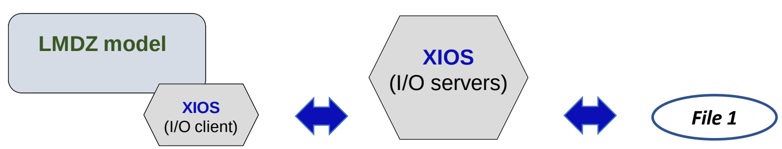
- Actual model : Each member of actual IPSL model produces its own output files



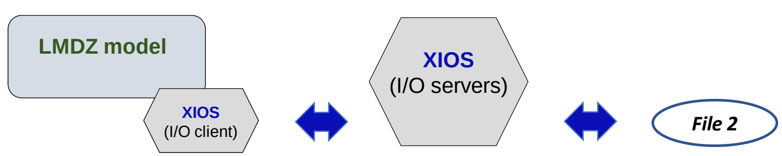
Member 1

- If the number of members is big (i.e > 10) :
  - A lot of inodes in computing centres filesystems and usually computing centres do not like inodes too much
  - A post-processing step is needed to “pack” output files to reduce inode footprint…
  - …but a post-processing step may be very loud (size of temporary buffer, computing resources, time to solution)

Member 1

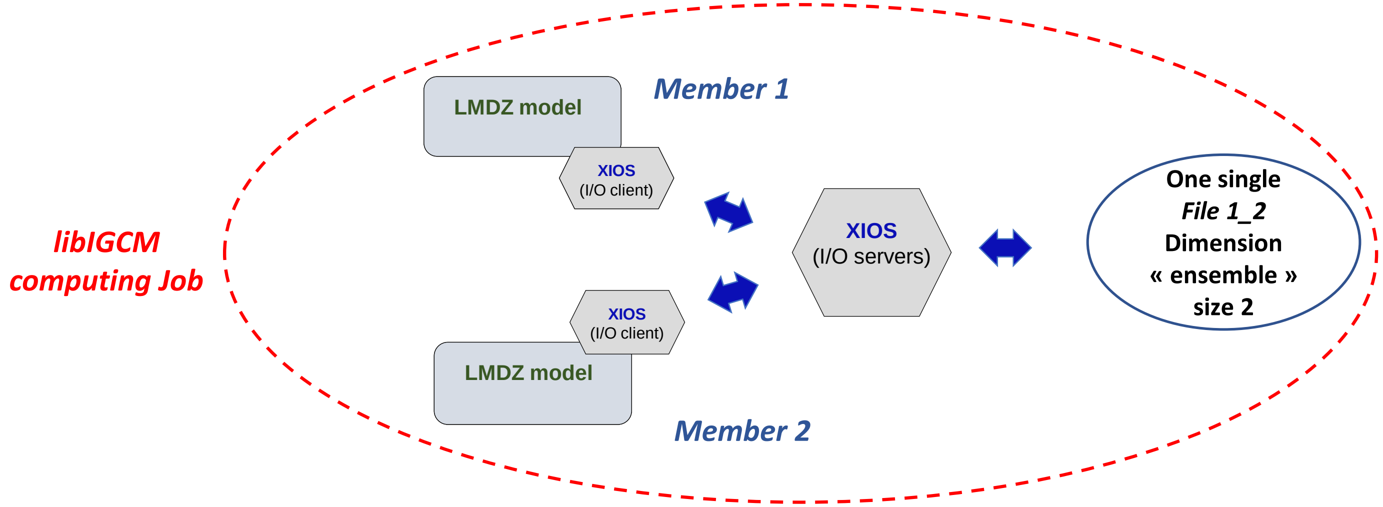


Member 2



Handling of many members

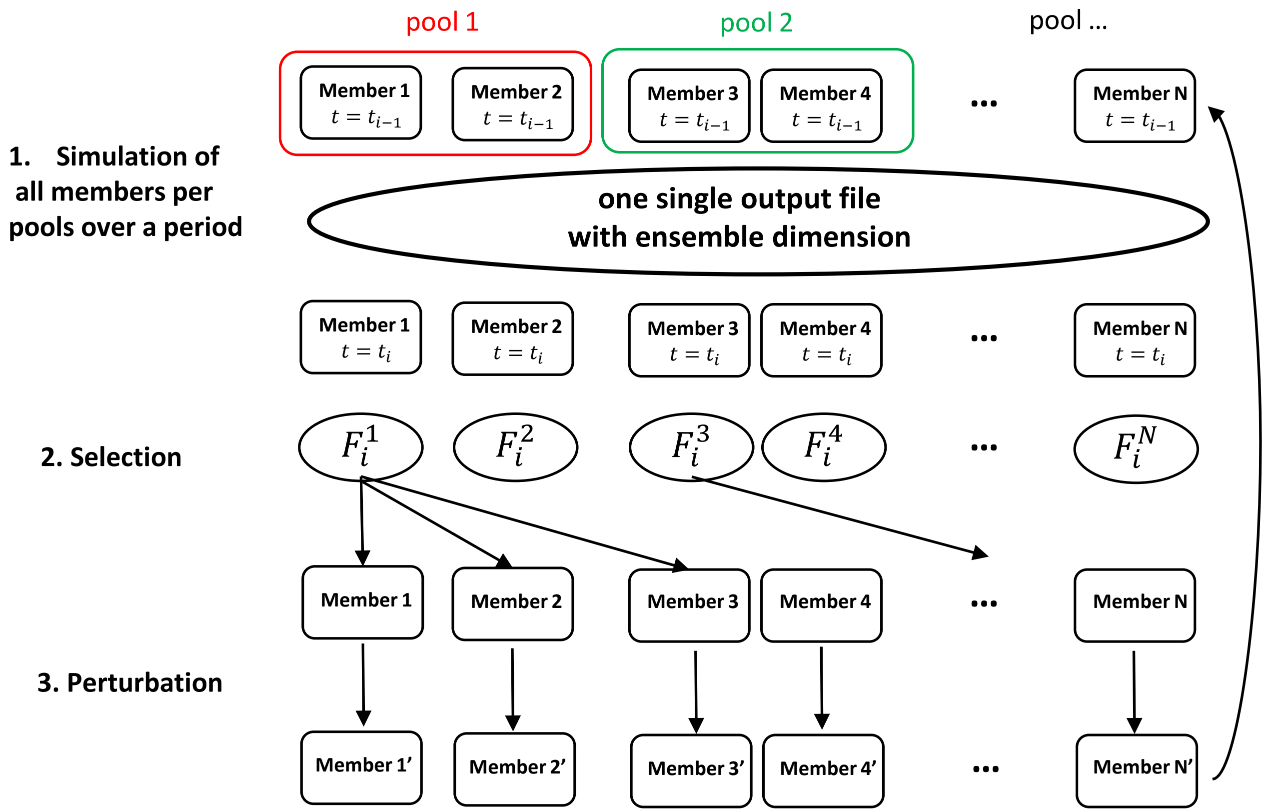
- One solution would be to adapt the model
  - to run **many instances (members)** of the model in parallel (simultaneously)
  - to be able to write out data from members into **a single file** (with additional ensemble dimension)
  - easily define the number of members, divided into pools by specifying the **number of pools** and the **size of the pools** in a parameter file



- Modifications needed in :
  - **XIOS** : handling of ensemble dimension as a axis
  - **LMDZ** (and other component) : MPI communicator containing all the members must be splitted so that each member recovers its own communicator
  - IPSL running environment (**libIGCM**) :
    - Management of execution of all instances of IPSL-CM model over a period in the same Job : the members of a pool run at the same time, the pools run sequentially one after the other.
    - Use of ensembles parameters : number of pools, size of pool
  - **OASIS coupler** (case IPSL-CM) : done by OASIS team

Selection and perturbation

- **Selection of members and perturbation**
  - Easy handling of selection of members using parameter file (according to result of a function score)
  - Perturbation of starting state of LMDZ atmospheric model by using a « white noise » on potential temperature (or ucov, vcov, ps) on all points of the domain, using parameter of run.def to define variable to perturbate, relative amplitude of perturbation,…
  - Perturbation is specific to one member (different perturbation between members), one period, one simulation.



Applications : Paleoclimatology

- **Reconstruction of 1500 years of paleoclimate (500 – 2014) by M Khodri**
  - IPSL-CM very low resolution version (IPSLCM6.5) : 200 km ocean NEMO4, 400 km atmosphere
  - 1 instance/member run on ~800 TGCC AMD Rome computing cores
  - 80 members of IPSL-CM6 to run and synchronize after given period
  - Use of data assimilation methods (SIR-LIM-Analogues) to select after every simulated year
  - Output data aggregated on the fly by XIOS into one single file.
  - Results
    - **8 pools of 10 members on 8 421 computing cores**
    - Gain of a factor 80 for number of inodes
    - Time to solution for one pool does not depend of the size of the pool
    - One single file at the end : very useful for analysis
    - Useful for low resolution that uses few cores : allows to apply for calls to benefit from computing hours (ex : PRACE) that are usually reserved for applications using several thousands cores.

Number of pools	Size of one pool	Nb cores	**Time to solution for 1 pool for 1 year period**	**Time to solution for 80 members for 1 year period**
80	1	843	**30 min**	**40 h**
40	2	1 685	**30 min**	**20 h**
16	5	4 211	**30 min**	**8 h**
10	8	6 737	**30 min**	**5 h**
8	10	8 421	**30 min**	**4 h**

Applications : Present and extreme events

- **Reconstruction of last 130 years of historical period (1880 – 2014), M Khodri**
  - IPSL-CM low resolution version (IPSLCM6.2.2) : 100 km ocean, 200 km atmosphere
  - 1 instance/member run on 1 800 AMD Rome computing cores
  - 80 members of IPSL-CM6 to run and synchronize after given period
  - Same data assimilation method and protocol of selection for previous case
  - Output data aggregated on the fly by XIOS into one single file
  - Results
    - **10 pools of 8 members on 15 360 computing cores**
    - Gain of a factor 80 for number of inodes
    - Time to solution for one pool does not depend of the size of the pool
    - Time to solution of all members is depending of the number of pools (and size of pools). Ex : 8h with 10 pools, 8 members per pool.

Number of pools	Size of one pool	Nb cores	**Time to solution for 1 pool for 1 year period**	**Time to solution for all members for 1 year period**
80	1	1857	**50 min**	**66 h**
40	2	3714	**50 min**	**33 h**
20	4	7927	**50 min**	**16 h**
16	5	9473	**50 min**	**13 h**
10	8	15360	**50 min**	**8 h**

- **Simulation of heat waves with LMDZ-ORCHIDEE (see Robin Noyelle’s talk)**
  - 100 members : 10 pools of 10 members.
  - Selection and perturbation every 5 days

Conclusions - Perspectives

The contract has been fulfilled

- many instances of the model (LMDZOR or IPSL-CM configuration) run in parallel generating one single output file (per type file)
- only one experiment directory (and one computing Job) to handle instead of one per member
- no impact on the inodes created for output files when running ensembles
- easy way to select and perturbate members
- useful (especially for low resolution) to target allocations reserved for many thousands computing cores per computing Job.
- useful for analysis to have one file with « ensemble » additional dimension rather than many files to use

Perspectives

- **Operations/reductions « in situ » between members**. In the actual version (XIOS2) :
  - it is possible to do some reductions/operations between members but only between members of a pool (and not between all members).
  - reductions/operations between members could have an impact on the time to solution because they are performed on model/client side and require synchronization between members
  - modifications are needed into the components to handle ensemble parameters (number of pools, size of pool). It would be better to be less invasive in the components.
- **XIOS3 coming soon** : asynchronous services in XIOS dedicated to such operations/reductions between different members with no modifications in component code
- Useful for **tuning exercise**
  - Many instances of the model run in parallel : same objective !
  - Missing generation of *.def from a parameter table : could be added
  - Need to integrate developments into updated sources and configurations, developments have been commited on specific sources of components (not on reference versions)
  - Need some work but why not…

Thank you !