DYNAMICO

Status and outlook

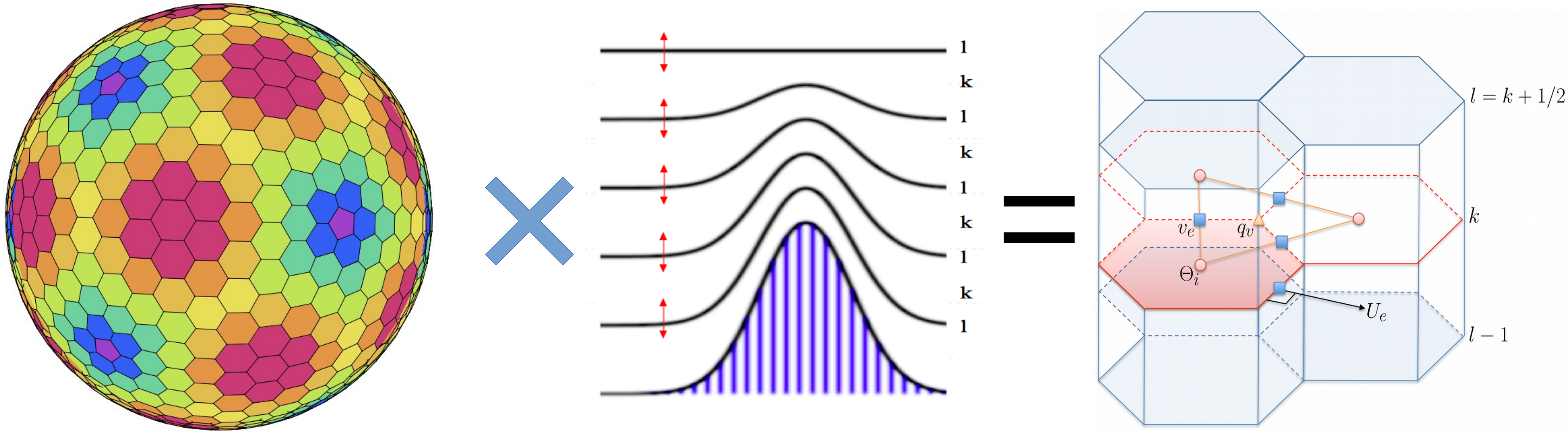
Thomas Dubos
École Polytechnique, LMD/IPSL

*with F. Hourdin, A. Traore, M. Tort, E. Millour, A. Spiga & co (LMD/IPSL),
J. Gattas (IPSL), Y. Meurdesoif, J. Servonnat, M. Kageyama, P. Braconnot (LSCE/IPSL),
S. Dubey (IIT Delhi), E. Kritsikis (LAGA/Paris XIII), ...*

- Status
- Ongoing developments
- Prospects for kilometer-scale climate modelling

DYNAMICO

Equations of motion	*shallow-water* *shalllow-atmosphere, hydrostatic*
Conservation properties	*Mass (air and species)* *Energy*
Formulation	*Mass : flux-form* *Momentum : vector-invariant form*
Vertical coordinate	*Terrain-following mass-based* *(often conflated with pressure-based)*
Numerics	*Mass : finite volume* *Momentum : low-order **mimetic finite difference*** *Mesh : **icosahedral-hexagonal C-grid*** *Time : (additive) Runge-Kutta (HEVI)*
Computing	*MPI / OpenMP* *XIOS I/O server* *Scales at least to O(10^4), including **I/O***

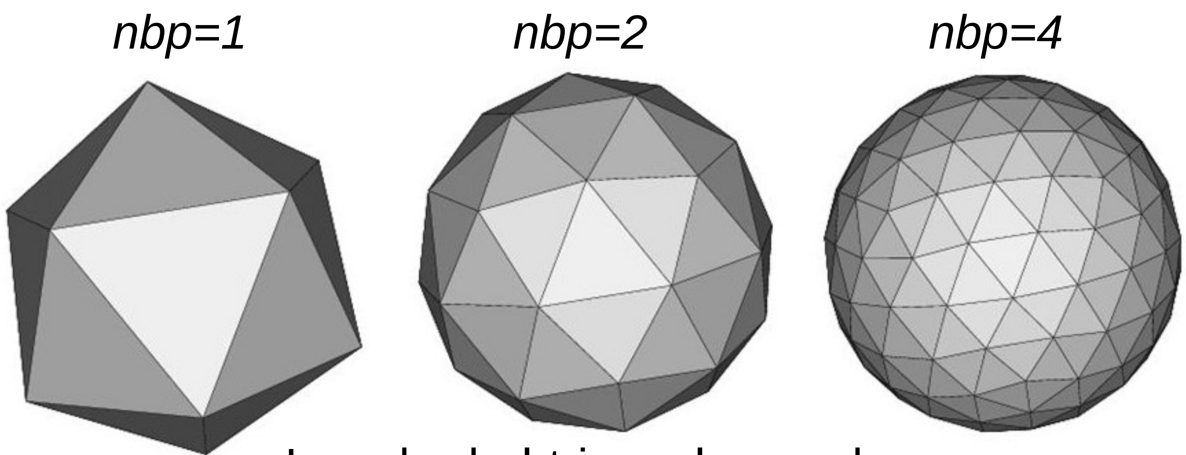


- Discrete integration by parts (Bonaventura & Ringler, 2005 ; Taylor, 2010)
- Energy- and vorticity- conserving Coriolis discretization (TRiSK : Thuburn et al., 2009 ; Ringler et al., 2010)

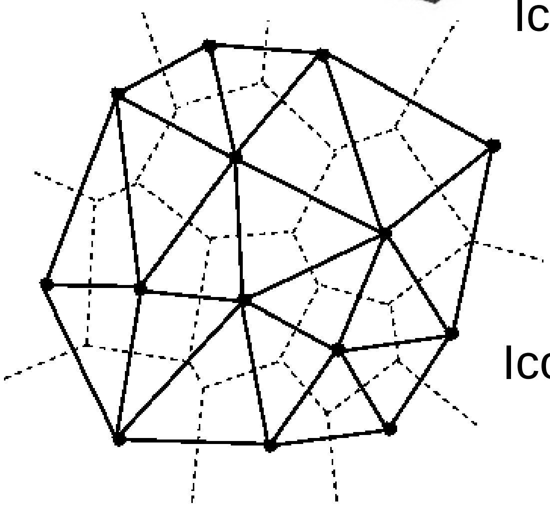
Energy-conserving 3D core

(Tort & Dubos, 2015 ; Dubos et al., 2015)

Mesh partitioning for parallel computing

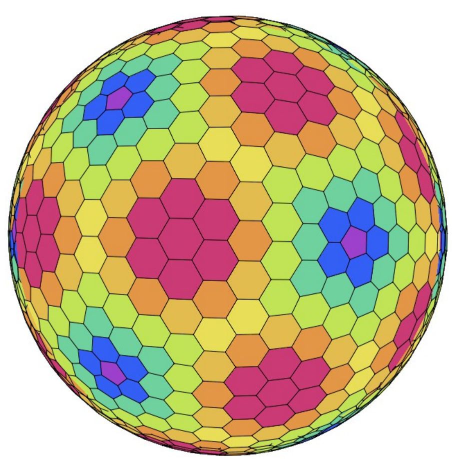


Icosahedral-triangular mesh
$10*nbp^2+2$ *vertices*

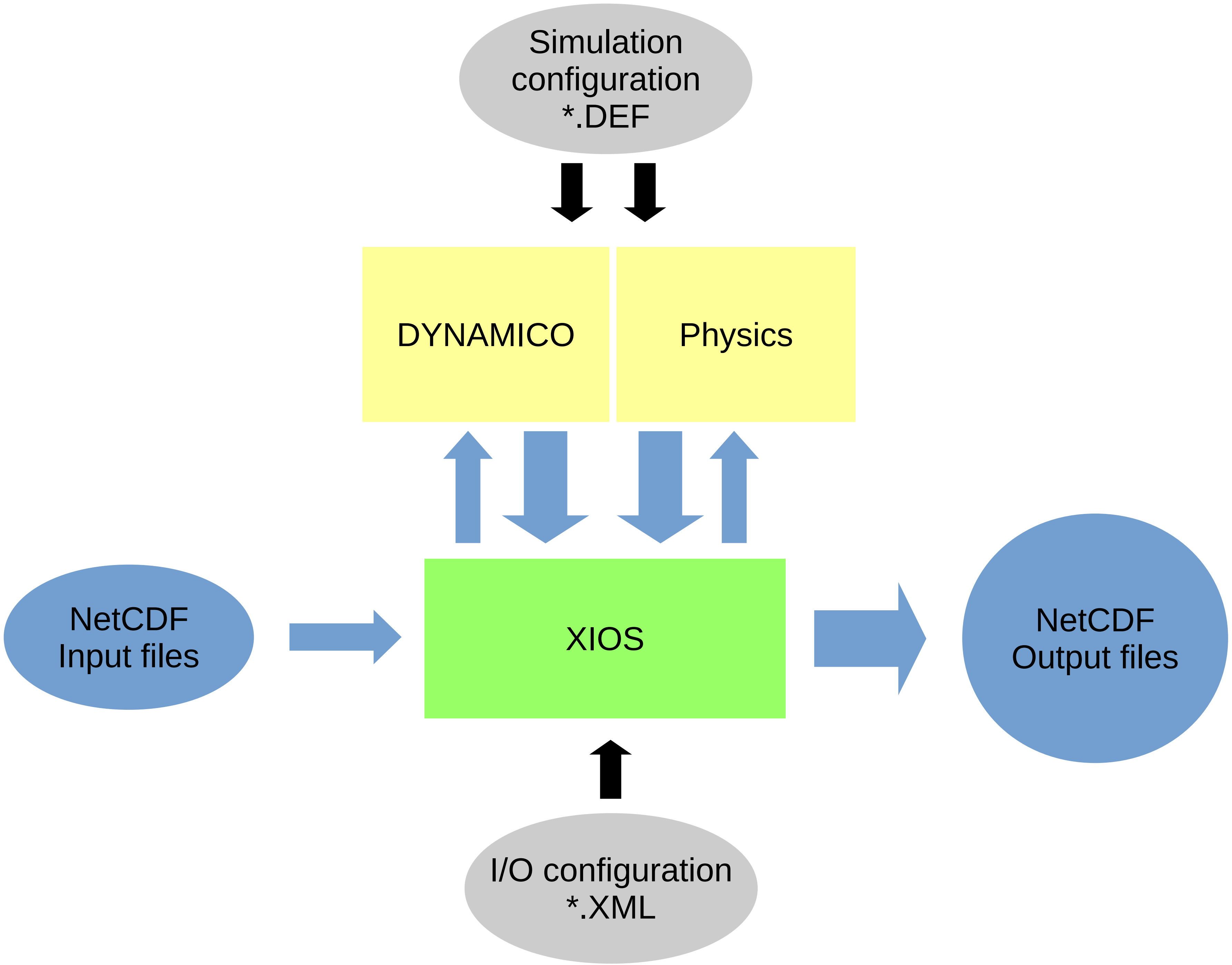


Voronoi dual

Icosahedral-hexagonal mesh
$10*nbp^2+2$ *cells*

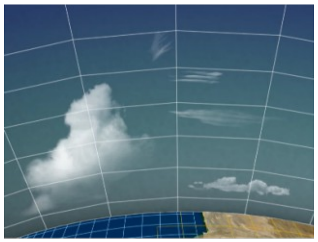


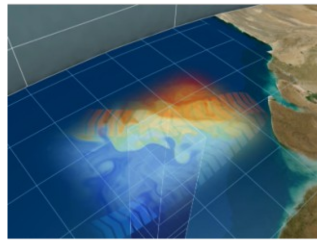
- Easy to partition into $10 \times nsplit^2$ domains
- About $(nbp/nsplit)^2$ cells per domain = MPI process
- Nbp/nsplit>10 for performance

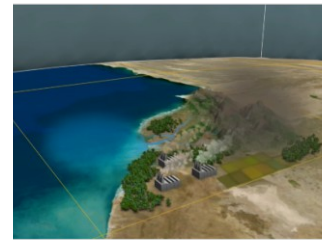

Simulation
configuration
*.DEF
DYNAMICO
Physics
NetCDF
Input files
XIOS
NetCDF
Output files
I/O configuration
*.XML

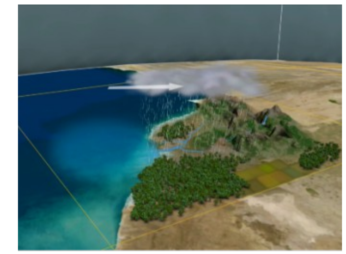
Target : IPSL Earth System Model IPSL-CM

HighResMIP : 2x50yr @ 25km (1 million horizontal grid points)





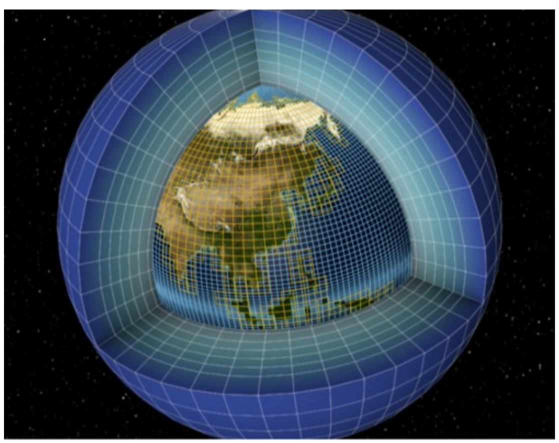




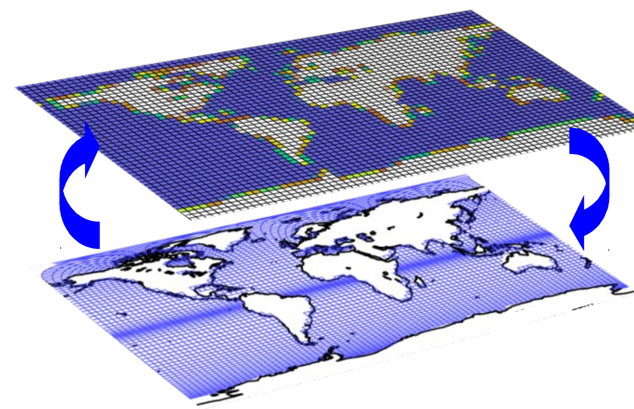
INCA / REPROBUS
(chimie atmosphérique)
(aérosol)

ORCHIDEE
(surfaces continentales)
(végétation)

LMDZ
(atmosphère)



OASIS
(coupleur)



OPA
(océan)

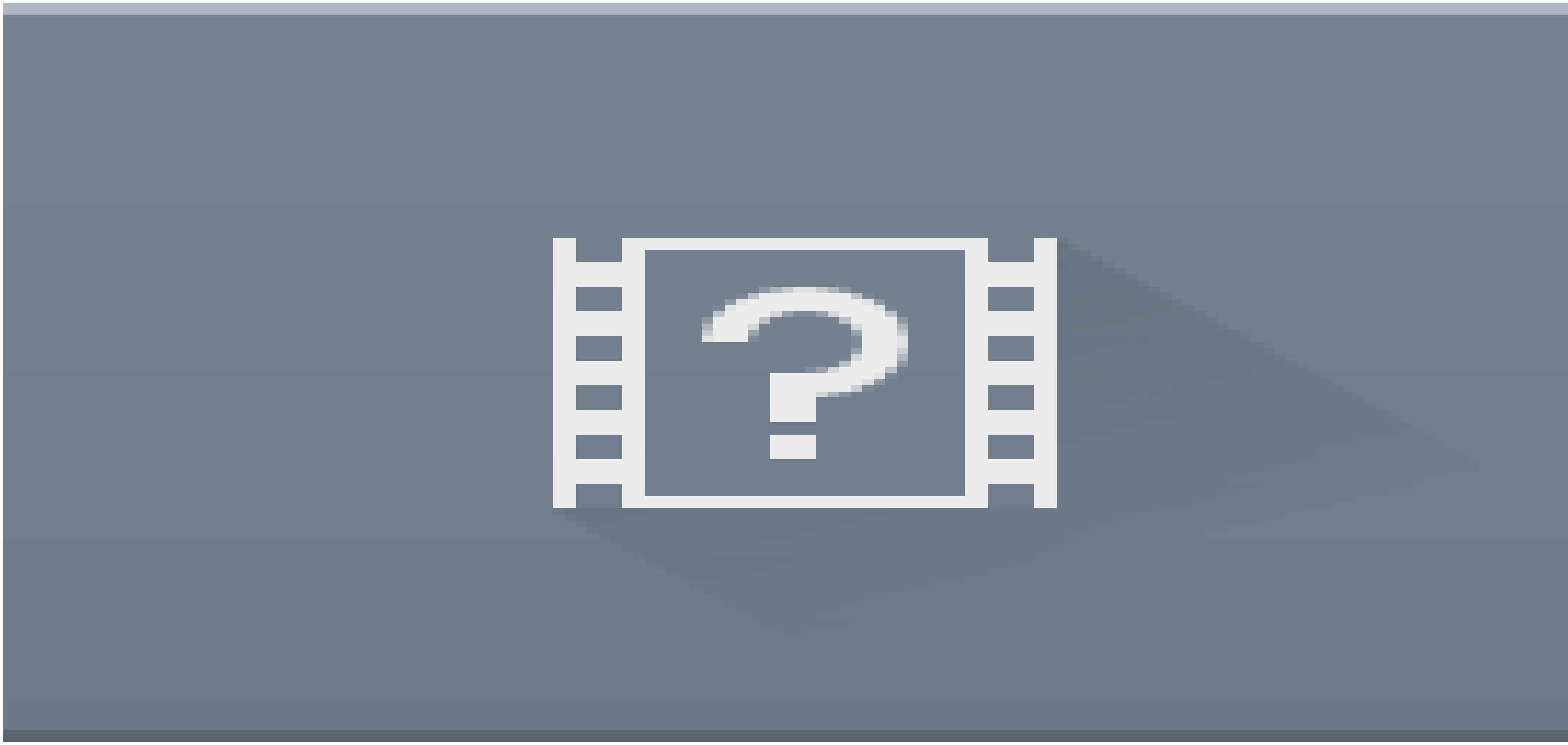
LIM
(glace de mer)

PISCES
(biogéochimie marine)

NEMO

Couplage DYNAMICO – physique de LMDZ

Ehouarn Millour, Yann Meurdesoif (LSCE)



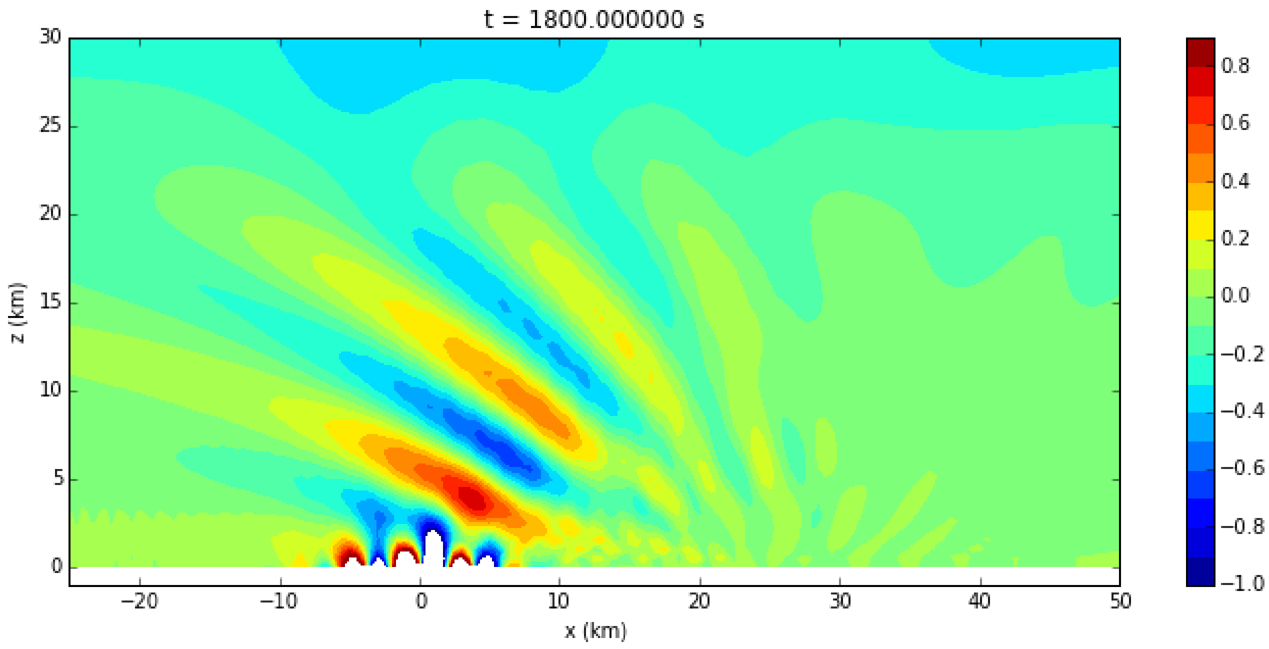
Precipitable water in an aquaplanet experiment (IPSL-CM5a physics, 1/4 degree)

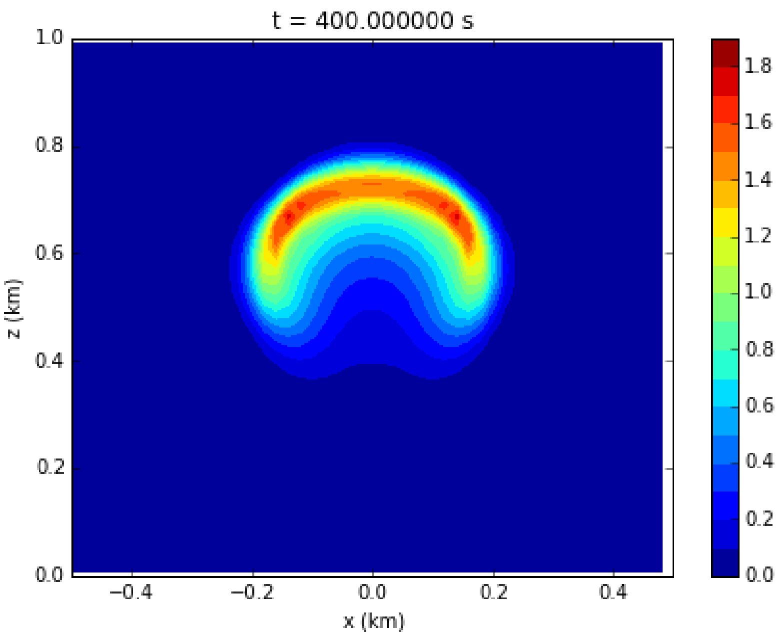
Ongoing : ORCHIDEE+routing

Ongoing developments and outlook

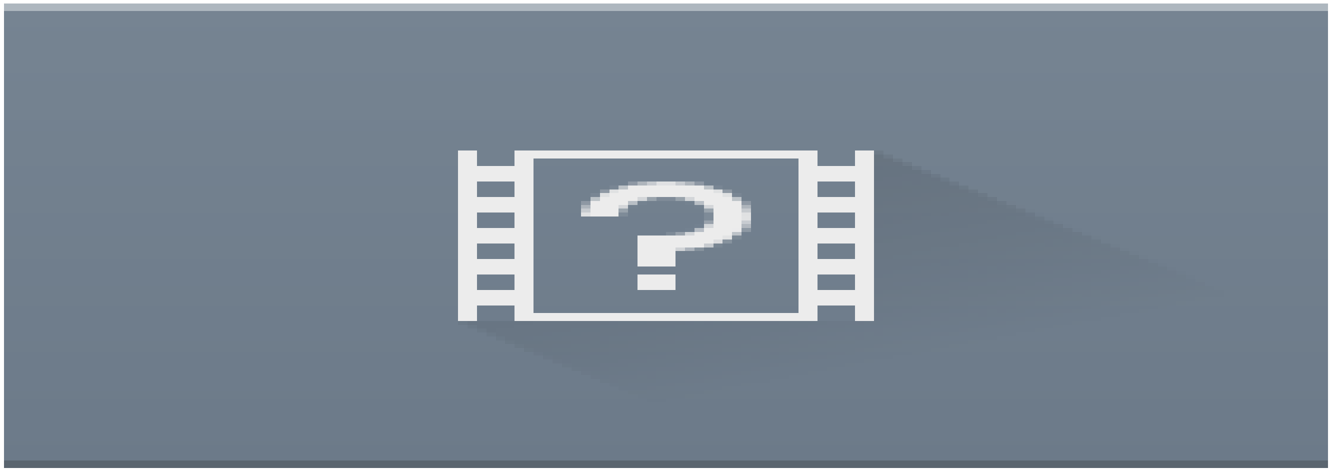
DYNAMICO-NH : non-hydrostatic (fully compressible) dynamics

T. Dubos, F. Voitus, C. Colavolpe (CNRM-GAME)



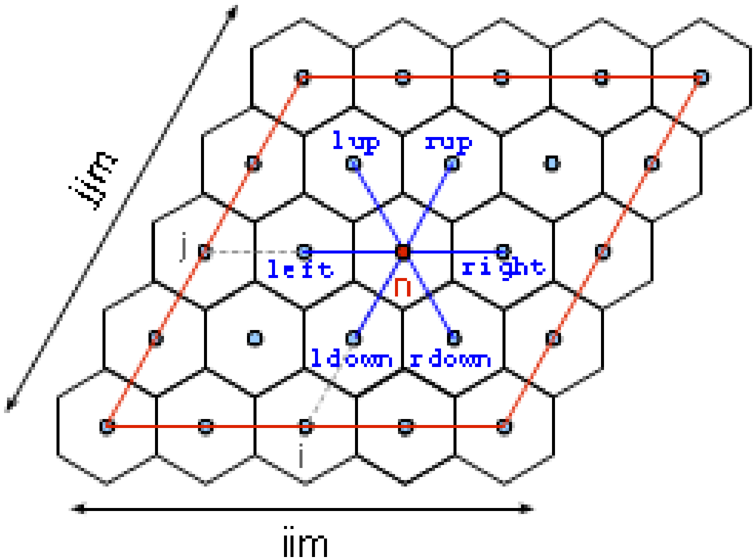


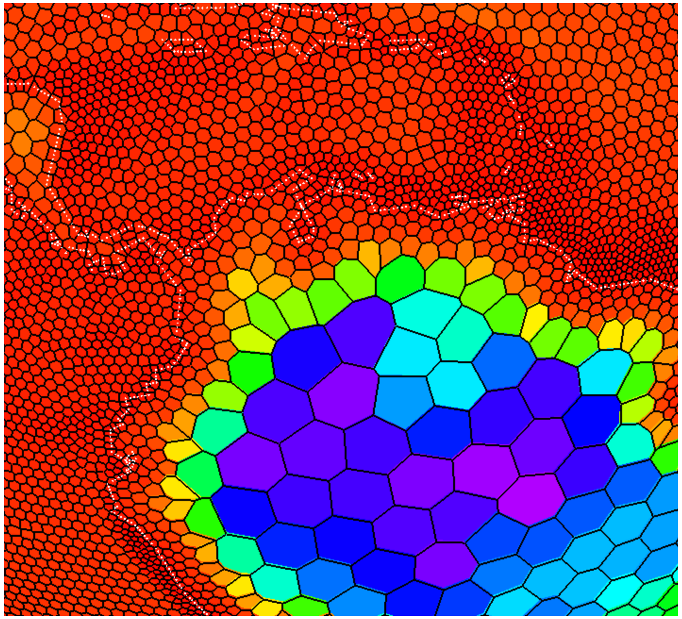
- *Runs DCMIP2016 test cases*
- *About 2x more expensive than hydrostatic*
- *current LMDz physics designed for more than ~10km resolution*



Subsidence of a zonally-symmetric cold pool

Unstructured-mesh capability





Structured mesh

- Quasi-uniform resolution
- Zoom possible but limited
- Regular data access / compute pattern

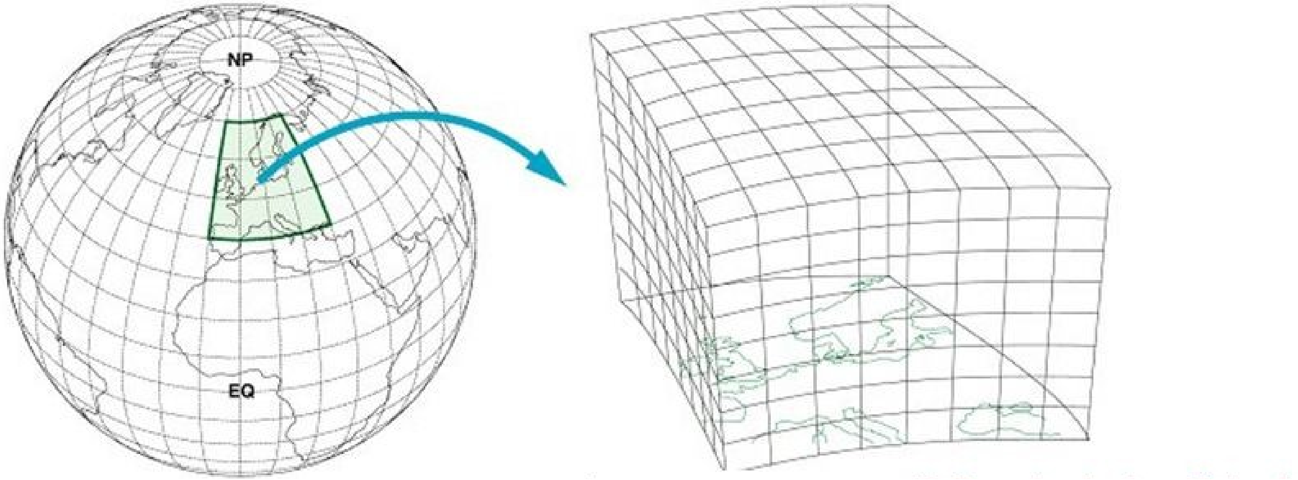
Unstructured mesh

- Variable resolution
- Very flexible zoom capability
- Irregular data access / compute pattern
- Needs scale-aware physics

Status :

- *Prototype available in the « devel» branch*
- *MPI/OpenMP parallelism*
- *More work ahead (performance, transport scheme, ...)*

Limited-area capability



Status :

- Not started
- IPSL Labex funding (12-month post-doc) to build **convection-resolving demonstrator** by 2019
- Physics will initially be LMDZ with convection shut down

Roadmap

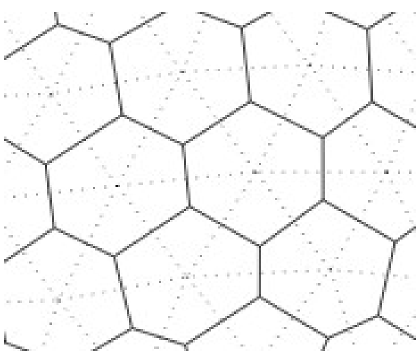
- Build periodic-box DYNAMICO, either from structured or unstructured code
- Use already developed coupling to LMD-Z
- Davies relaxation at lateral domain boundaries
- Build workflow to get boundary data through XIOS

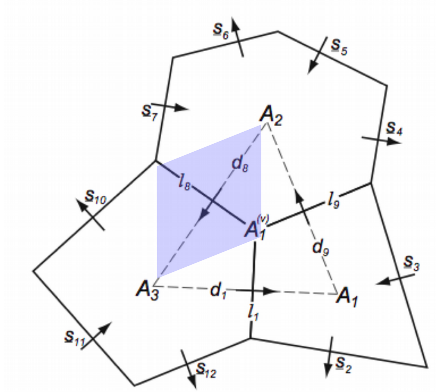
DySL : Dynamico-specific « language »

A low-tech experiment in code maintainability

and separation of concerns

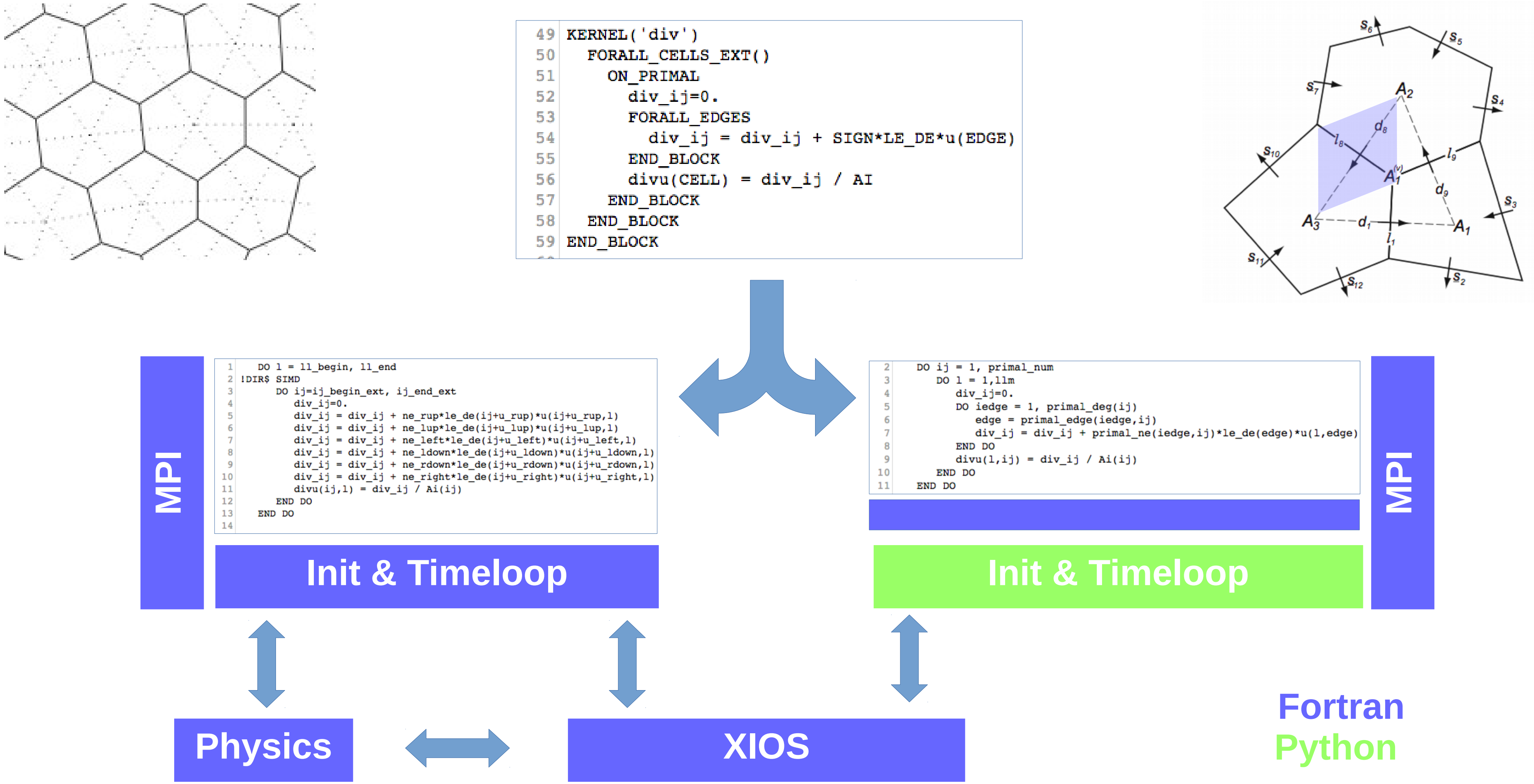
```
KERNEL('div')
  FORALL_CELLS_EXT()
    ON_PRIMAL
      div_ij=0.
      FORALL_EDGES
        div_ij = div_ij + SIGN*LE_DE*u(EDGE)
      END_BLOCK
      divu(CELL) = div_ij / AI
    END_BLOCK
  END_BLOCK
END_BLOCK
```





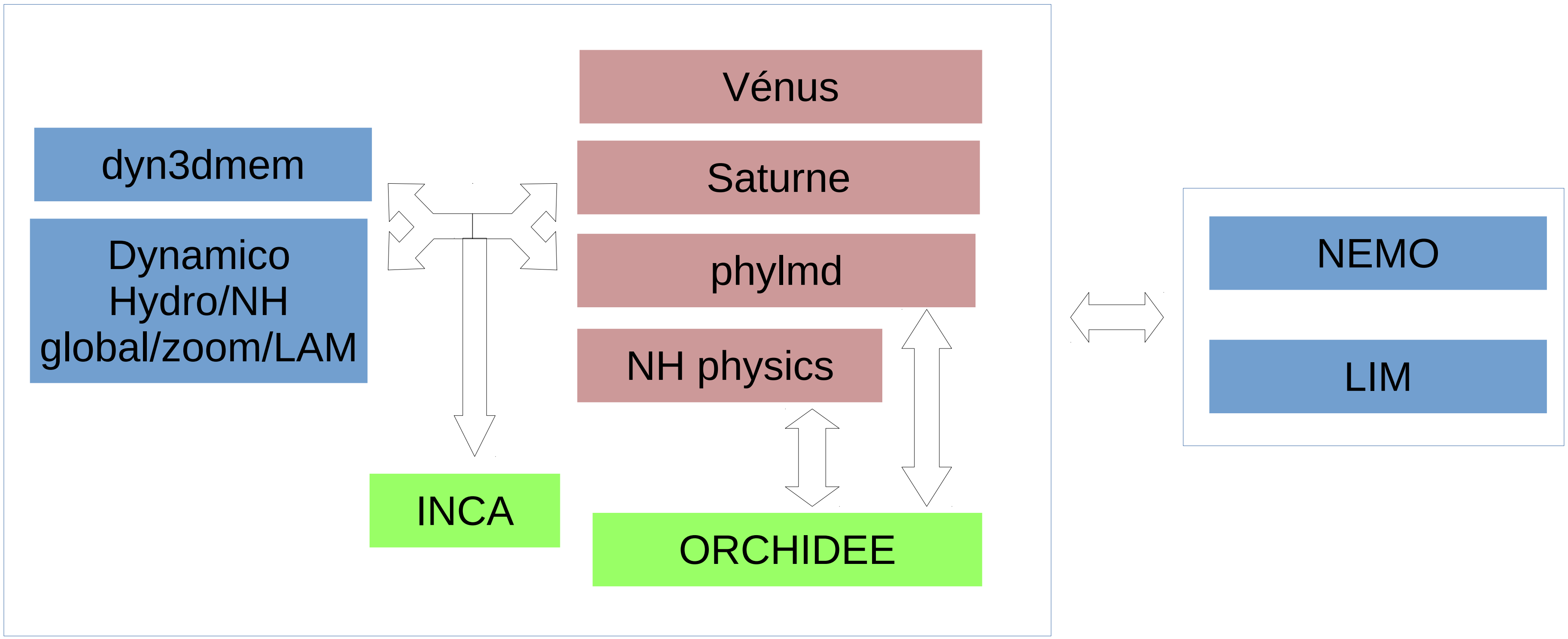

```
   DO l = ll_begin, ll_end
!DIR$ SIMD
      DO ij=ij_begin_ext, ij_end_ext
         div_ij=0.
         div_ij = div_ij + ne_rup*le_de(ij+u_rup)*u(ij+u_rup,l)
         div_ij = div_ij + ne_lup*le_de(ij+u_lup)*u(ij+u_lup,l)
         div_ij = div_ij + ne_left*le_de(ij+u_left)*u(ij+u_left,l)
         div_ij = div_ij + ne_ldown*le_de(ij+u_ldown)*u(ij+u_ldown,l)
         div_ij = div_ij + ne_rdown*le_de(ij+u_rdown)*u(ij+u_rdown,l)
         div_ij = div_ij + ne_right*le_de(ij+u_right)*u(ij+u_right,l)
         divu(ij,l) = div_ij / Ai(ij)
      END DO
   END DO
```

```
   DO ij = 1, primal_num
      DO l = 1,llm
         div_ij=0.
         DO iedge = 1, primal_deg(ij)
            edge = primal_edge(iedge,ij)
            div_ij = div_ij + primal_ne(iedge,ij)*le_de(edge)*u(l,edge)
         END DO
         divu(l,ij) = div_ij / Ai(ij)
      END DO
   END DO
```



- Code generator purely text-based (no grammar / parser / syntax tree)
- Macro substitution (cpp) and inlining (jinja2)
- Generates *human-readable* code chunks to be #included in hand-written Fortran code
- Hexagonal version executes either generated code or hand-written code
- Non-hydrostatic dynamics based on generated code
- Unstructured code : only performance-critical pieces in Fortran, interfaced to higher-level code in Python

Mid-term outlook :
Towards a global-regional IPSL ESM



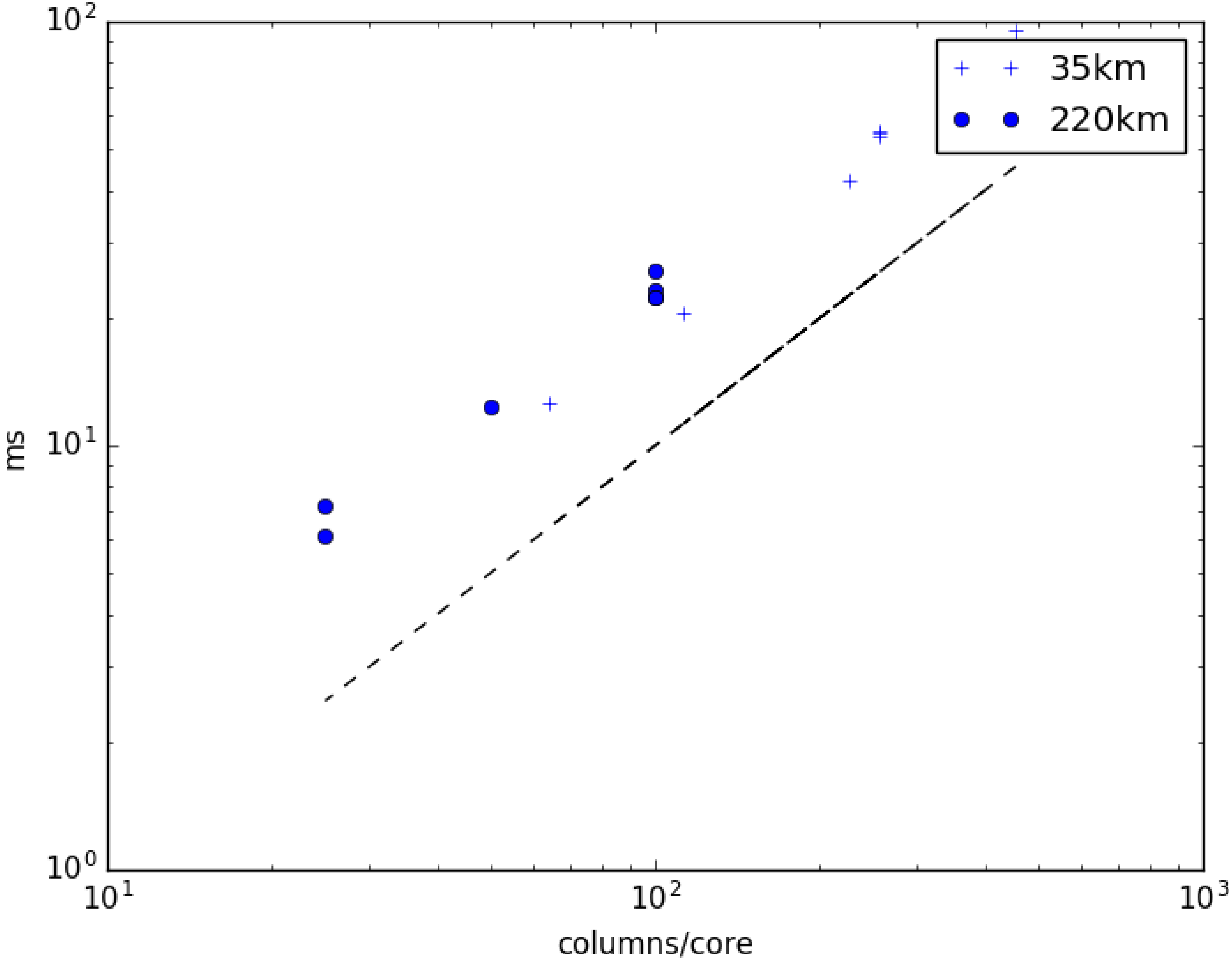
Prospects for kilometer-scale *climate* modelling

- CMIP requires a throughput of x10000 (30SYPD)
- Some climate modelling still doable with x1000 (3SYPD)
- Ability to attain x1000 depends on maximum stable time step (numerics) and walltime needed to perform one time step (implementation)
- Assuming a large enough machine, reducing walltime is a **strong scaling problem**
- For DYNAMICO, dt (in sec) is about 2.5*dx (in km)

  => 3SYPD

  - At 25km resolution requires about 60 ms per full time step
  - At 8km resolution requires about 20 ms per full time step
  - At 1km resolution requires about 2.5 ms per full time step

Curie (Intel Sandy Bridge) MPI/OpenMP Dynamics + 4 tracers no phys, no I/O



CMIP6 physics (79 vertical levels) cost 2-3 ms per column per call
(24 SYPD with 96 calls per day, 36 columns per core)

If physics are called every 5 dynamics time steps :

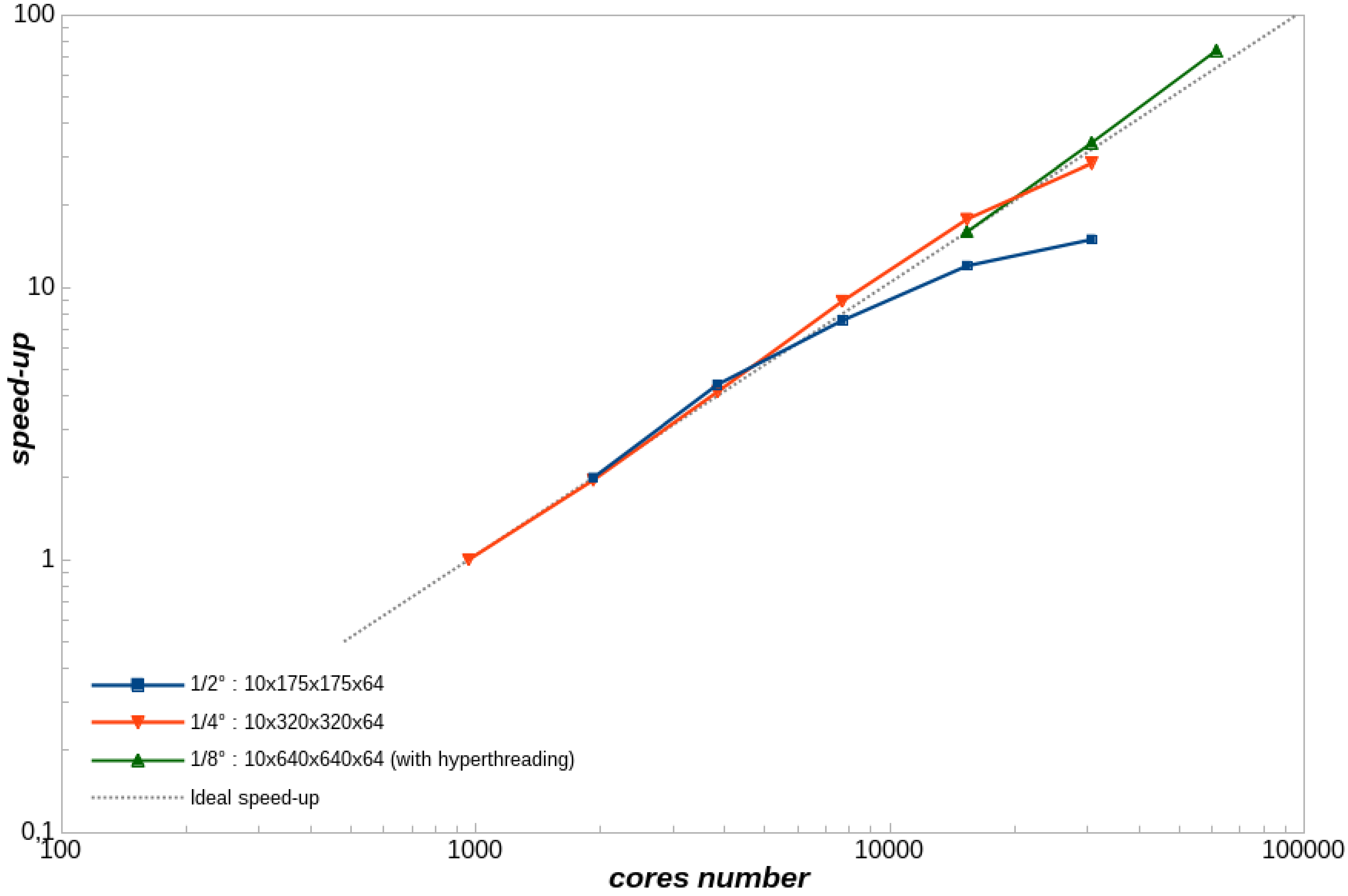
100 columns/core	=> 20+(100*2/5)=60 ms	25 km OK with 10 000 cores
30 columns/core	=> 5+(30*2/5)=17 ms	8 km doable with 300 000 cores
10 columns/core	=> 3ms+(10*2/5)=7 ms	1km not doable

Thanks for your attention

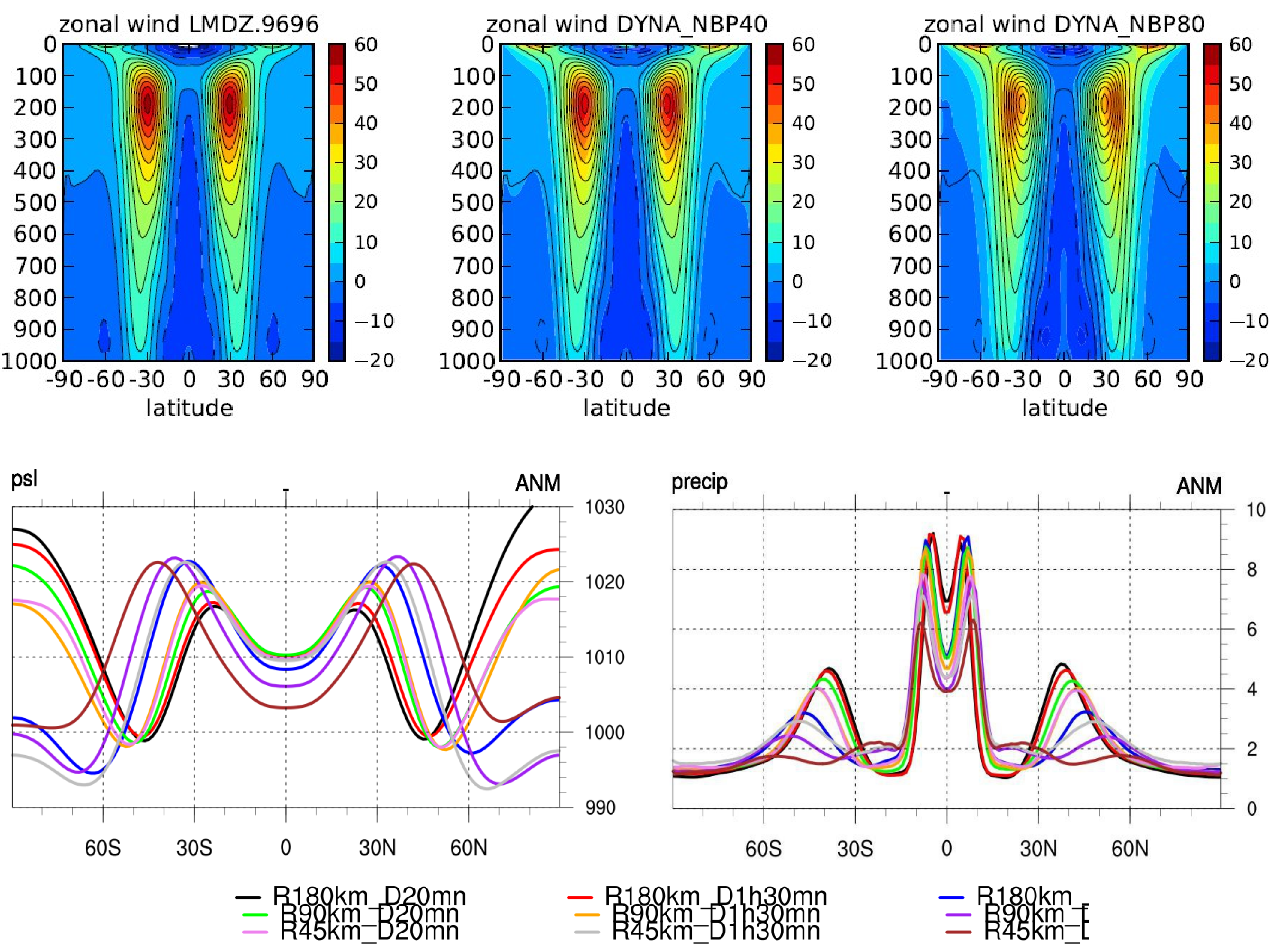
- T. Dubos, S. Dubey, M. Tort, R. Mittal, Y. Meurdesoif and F. Hourdin (2015) *DYNAMICO, a hydrostatic icosahedral dynamical core designed for consistency and versatility* Geosci. Mod. Dev.
- S. Dubey, T. Dubos, F. Hourdin, H.C. Upadhyaya (2015) *On the inter-comparison of two tracer transport schemes on icosahedral grids* Applied Math. Mod. 39(16) 4828-4847 doi:10.1016/j.apm.2015.04.015
- E. Kritsikis, M. Aechtner, Y. Meurdesoif, and T. Dubos *Conservative interpolation between general spherical meshes* Geosci. Mod. Dev.
- T. Dubos and M. Tort (2014) *Equations of atmospheric motion in non-Eulerian vertical coordinates : vector-invariant form and Hamiltonian formulation* Mon. Wea. Rev. 142(10) : 3860-3880

- http://forge.ipsl.jussieu.fr/dynamico/wiki
- https://forge.ipsl.jussieu.fr/heat/wiki

Extra slides



100
10
1
0,1
100
1000
10000
100000
speed-up
cores number
1/2° : 10x175x175x64
1/4° : 10x320x320x64
1/8° : 10x640x640x64 (with hyperthreading)
Ideal speed-up


zonal wind LMDZ.9696
zonal wind DYNA_NBP40
zonal wind DYNA_NBP80
latitude
psl
ANM
precip
60S
30S
0
30N
60N
R180km_D20mn
R90km_D20mn
R45km_D20mn
R180km_D1h30mn
R90km_D1h30mn
R45km_D1h30mn
R180km_
R90km_
R45km_

Example configurations

LMDZ

		MPI
3°	96x95	48
2°	144x142	72
(2/3)°	512x360	180

DYNAMICO

		MPI	SYPD (no phys)	SYPD* (CMIP5a)	Mh per 100 yr*
2°	10x40x40	160	300	120	0,01
1°	10x80x80	640	150	40	0,15
(1/2)°	10x160x160	640		18	0,4
(1/4)°	10x320x320	*2560*		12	2

**assuming 4x with 4 OpenMP threads, no attempt to optimize/tune MPI or XIOS, very few runs* slow | fast

$$H = K_H[\mu, \Phi, \mathbf{v}, W] + K_V[\mu, \Phi, W] + P[\mu, \Theta, \Phi]$$

$$\partial_t \mu + \partial_{\mathbf{x}} \cdot \frac{\delta H}{\delta \mathbf{v}} + \partial_\eta (\mu \dot{\eta}) = 0$$

$$\partial_t \Theta + \partial_{\mathbf{x}} \cdot \theta \frac{\delta H}{\delta \mathbf{v}} + \partial_\eta (\theta \mu \dot{\eta}) = 0$$

$$\partial_t \mathbf{v} + \dot{\eta} (\partial_\eta \mathbf{v} - \partial_{\mathbf{x}} w) + \frac{\partial_{\mathbf{x}} \times \mathbf{v}}{\mu} \times \frac{\delta H}{\delta \mathbf{v}} + \partial_{\mathbf{x}} \frac{\delta K_H}{\delta \mu} + \partial_{\mathbf{x}} \frac{\delta (K_V + P)}{\delta \mu} + \theta \partial_{\mathbf{x}} \frac{\delta P}{\delta \Theta} = 0$$

$$\partial_t \Phi + \dot{\eta} \partial_\eta \Phi - \frac{\delta K_H}{\delta W} - \frac{\delta K_V}{\delta W} = 0$$

$$\partial_t W + \partial_\eta (\dot{\eta} \Phi) + \frac{\delta K_H}{\delta \Phi} + \frac{\delta (K_V + P)}{\delta \Phi} = 0$$

- The implicit problem only couples vertical position and vertical momentum
- eliminate W and obtain a scalar tridiagonal implicit problem for Φ