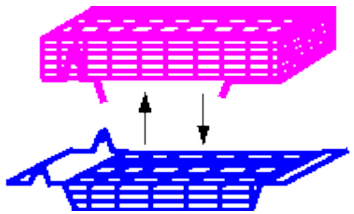
EUROPEAN CENTRE FOR RESEARCH AND ADVANCED TRAINING IN SCIENTIFIC COMPUTING

OASIS3-MCT, a coupler for climate modelling

S. Valcke, L. Coquart, CERFACS

- Introduction: Technical coupling solutions
- OASIS historical overview
- OASIS community today
- Use of OASIS3-MCT:
  - component model interfacing
  - coupled model configuration
- OASIS3-MCT communication
- OASIS3-MCT interpolations et transformations
- OASIS3-MCT performances
- Conclusions and perspectives

Why couple ocean and atmosphere (and sea-ice and land and …) models?



- ➢ Of course, to treat the Earth System globally

What does "coupling of codes" imply?

- ➢ Exchange and transform information at the code interface
- ➢ Manage the execution and synchronization of the codes

What are the constraints?

- ✓ Coupling should be easy to implement, flexible, efficient, portable
- ✓ Coupling algorithm dictated by science (sequ. vs conc. coupling)
- ✓ Start from existing and independently developed codes
- ✓ Global performance and load balancing issues are crucial
- ✓ Platform characteristics (OS, CPU, message passing efficiency, …)

Technical coupling solutions

1. merge the codes:

program prog1
…
call sub_prog2(data)
…
end prog1

→

~~program prog2~~
subroutine sub_prog2(data)
…
end prog2

☺ efficient (memory exchange)
☺ as portable as the codes
☺ one executable: easier to debug, easier for the OS
☺ sequential execution of the components

☹ not easy to implement with existing codes (splitting, conflicts in namespaces and I/O)
☹ not flexible (coupling algorithm hard coded)
☹ no use of generic transformations/ interpolations
☹ loss of one degree of parallelism in the execution of the components

Technical coupling solutions

2. use existing communication protocole (MPI, CORBA, UNIX pipe, files, …)

```
program prog1
…
call xxx_send (prog2, data, …)
end
```

```
program prog2
…
call xxx_recv (prog1,data)
end
```

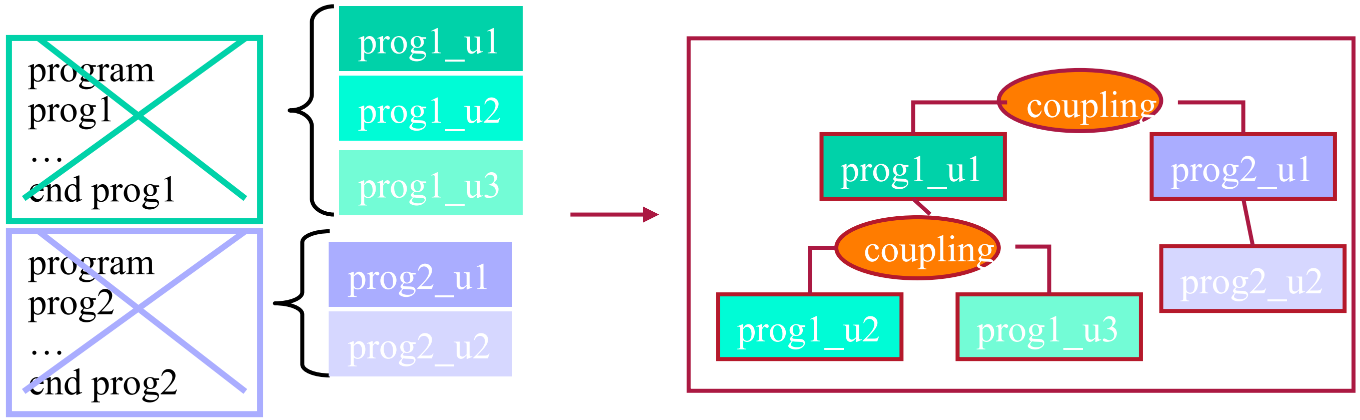
- ☺ existing codes
- ☺ natural parallelism in the execution of the components

- ☹ not easy to implement (needs protocol expert)
- ☹ not flexible (hard coded exchanges)
- ☹ multi-executable: possible waste of resources if forced sequential execution of the components
- ☹ multi-executable: more difficult to debug; harder to manage for the OS
- ☹ no use of generic transformations/interpolations
- 😐 efficient, portable

Technical coupling solutions

3. use coupling framework   ESMF **FMS**(GFDL) *CESM* (NCAR)

- Split code into elemental units
- Write or use coupling units
- Use the library to build a hierarchical merged code
- Adapt code data structure and calling interface

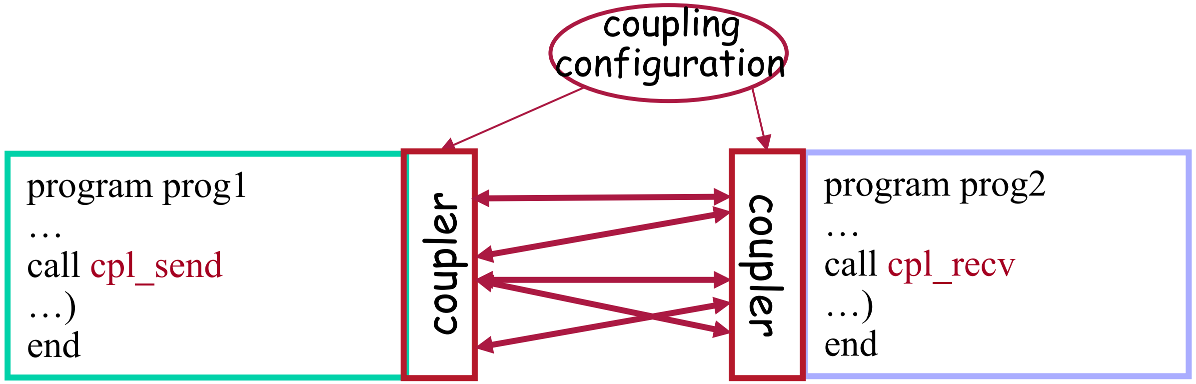


☺ efficient, ☺ sequential and concurrent components ☺ use of generic utilities (parallelisation, regridding, time management, etc.)	☹ existing codes 😐 (easy)

→ probably best solution in controlled development environment

Technical coupling solutions

4. use a coupler or coupling library

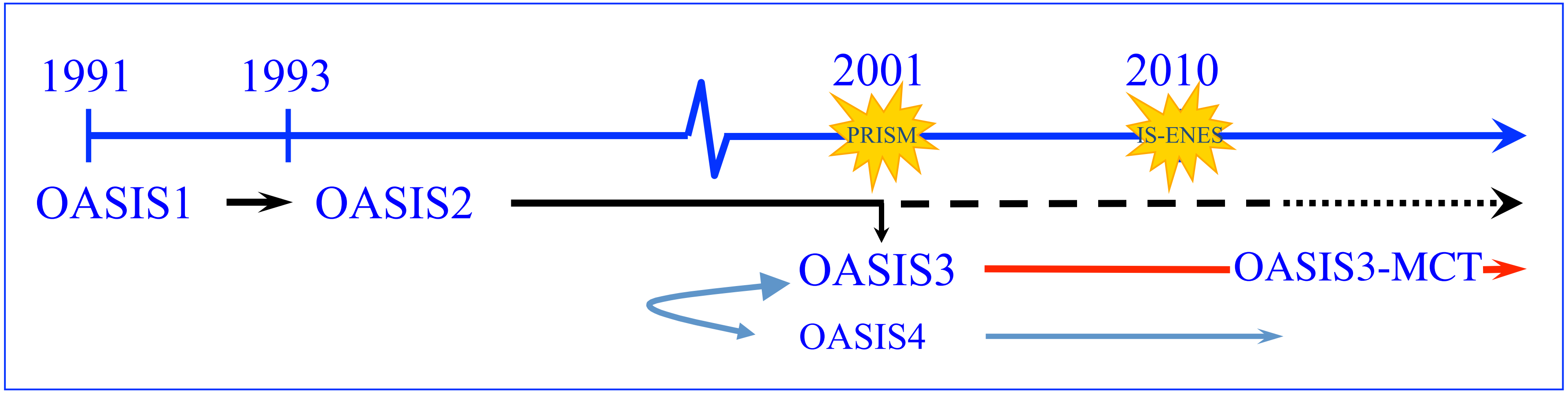


- ☺ existing codes
- ☺ use of generic transformations/ regridding
- ☺ concurrent coupling (parallelism)

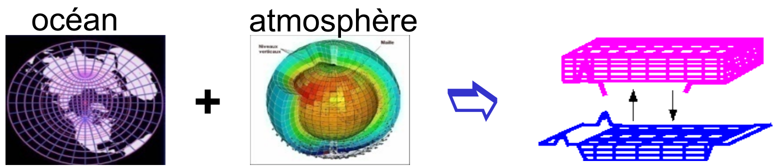
- ☹ multi-executable: possible waste of resources if sequential execution of the components is enforced
- ☹ multi-executable: more difficult to debug; harder to manage for the OS
- 😐 efficient

→ probably best solution to couple independently developed codes

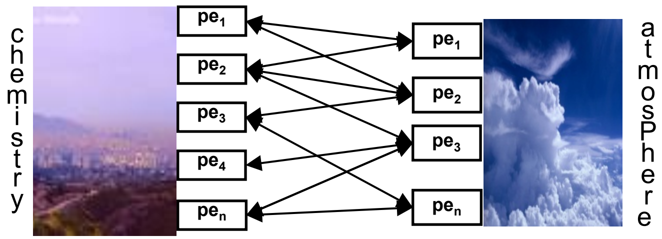
OASIS historical overview



- **OASIS1 -> OASIS2 -> OASIS3:**
  2D ocean-atmosphere coupling
  low resolution, low frequency
  → flexibility, modularity, 2D interpolations



- OASIS4 / **OASIS3-MCT:**
  2D/3D coupling of high resolution parallel components
  on massively parallel platforms
  →parallelism, efficiency, performance



OASIS community today

About 35 groups world-wide (climate modelling or operational monthly/seasonal forecasting):

- France: CERFACS, METEO-FRANCE, IPSL (LOCEAN, LMD, LSCE), OMP, LGGE, IFREMER
- Europe: ECMWF + Ec-Earth community
- Germany: MPI-M, IFM-GEOMAR, HZG, U. Frankfurt
- UK: MetOffice, NCAS/U. Reading, ICL
- Denmark: DMI
- Norway: U. Bergen
- Sweden: SMHI, U. Lund
- Ireland: ICHEC, NUI Galway
- The Netherland: KNMI
- Switzerland: ETH Zurich
- Italy: INGV, ENEA, CASPUR
- Czech_Republic :CHMI
- Spain: U. Castilla
- Tunisia: Inst. Nat. Met
- Japan: JMA, JAMSTEC
- China: IAP-CAS, Met. Nat. Centre, SCSIO
- Korea: KMA
- Australia: CSIRO
- New Zealand: NIWA
- Canada: RPN-Environment Canada, UQAM
- USA: Oregon State U., Hawaii U., JPL, MIT
- Peru: IGP + downloads from Belgium, Nigeria, Colombia, Saudi Arabia, Singapore, Russia

➢ OASIS3 is used in 5 of the 7 European ESMs that participate in IPCC AR5

OASIS3-MCT: some generalities

- All sources are written in F90 and C
- Uses the Model Coupling Toolkit (MCT) from Argonne National Lab
- Open source product distributed under a LGPL license
- All external libraries used are public domain (MPI, NetCDF) or open source (LANL SCRIP, MCT)
- Current developers are:
  - 1.3 permanent FTEs (CERFACS, CNRS)
  - 1 consultant (T. Craig, previously from NCAR)

IS-ENES (InfraStructure for ENES) EU FP7 project 2009-2012 - 18 partners – 7,6 MEuros ; coord: IPSL

- 93 pm for OASIS development and supp

IS-ENES2, EU FP7 project 2013-2016 – 25 partners - 8MEuros

- 27 pm for OASIS3-MCT development and support

Use of OASIS3-MCT

At run time, the component models remain separate executables and OASIS3-MCT acts as a communication library linked to the models.

To use OASIS3-MCT:

- Download the sources, compile and run the tutorial on your platform
- Identify your component models, grids, coupling fields to be exchanged
- Identify the transformations to go from the source to the target grids
- Use the "test_interpolation" environment (offline) to test the quality
- Adapt your codes i.e. insert calls to OASIS3-MCT communication library
- Choose the other parameters (source and target, frequency, field trans - formations, etc.) and create the *namcouple* configuration file with the GUI
- Compile OASIS3-MCT, your components **with same compiler**, and link the components models with OASIS3-MCT library
- Start the models and let OASIS3-MCT manage the coupling exchanges

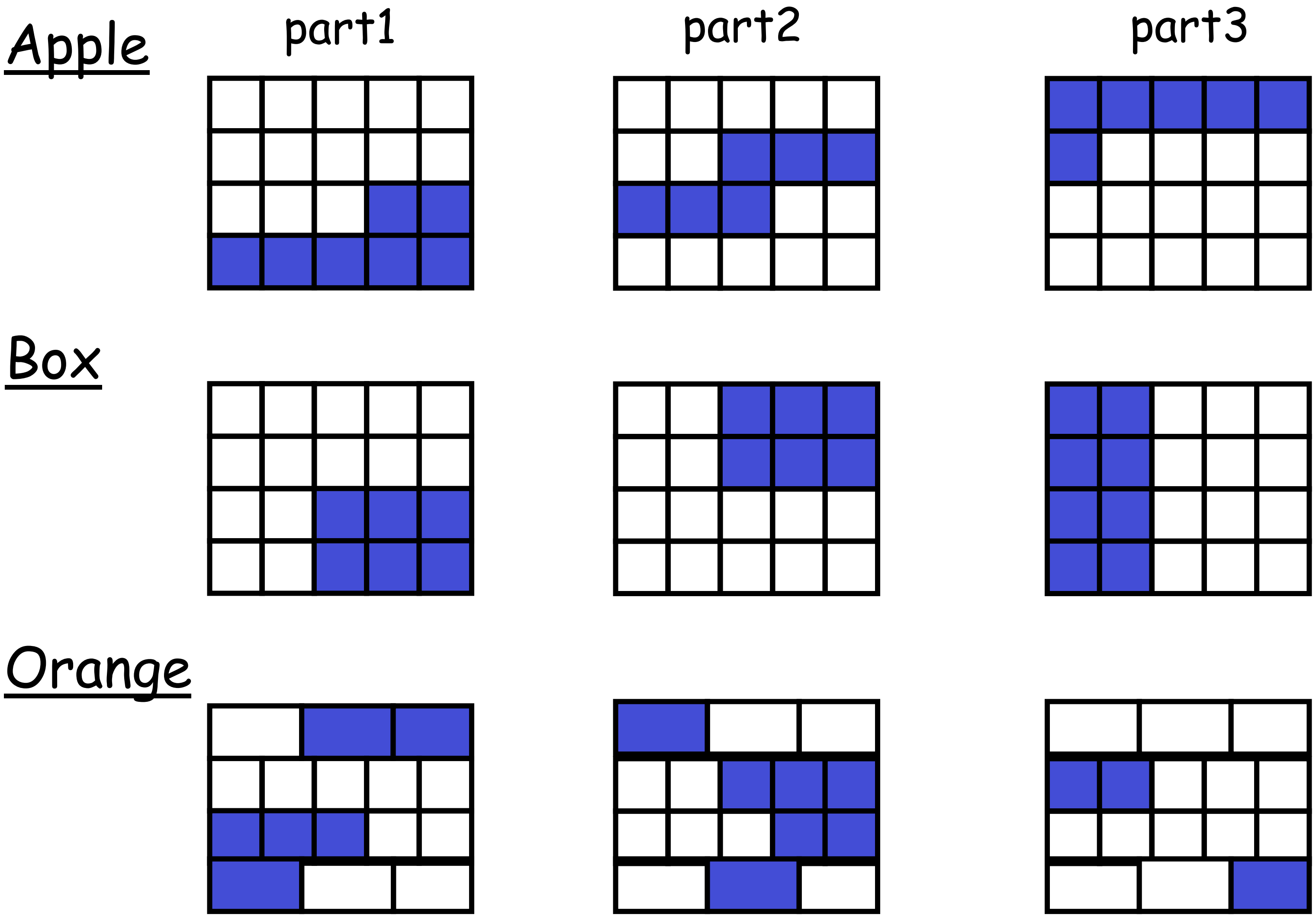
OASIS3-MCT: component model interfacing

- Initialization: `call oasis_init_comp(...)`
- Grid definition: `call oasis_write_grid (...)`
- Local partition definition: `call oasis_def_partition (...)`
- Coupling field declaration: `call oasis_def_var (...)`
- End of definition phase: `call oasis_enddef (…)`
- **Coupling field exchange**:
  - in model time stepping loop

```
call oasis_put (…, date, var_array. …)
call oasis_get (…, date, var_array, …)
```

  - user's defined source or target (end-point communication)
  - sending or receiving at appropriate time only
  - automatic averaging/accumulation if requested
  - automatic writing of coupling restart file at end of run
- Termination: `call oasis_terminate (…)`

OASIS3-MCT: partitioning supported



Apple and orange applicable to unstructured grids

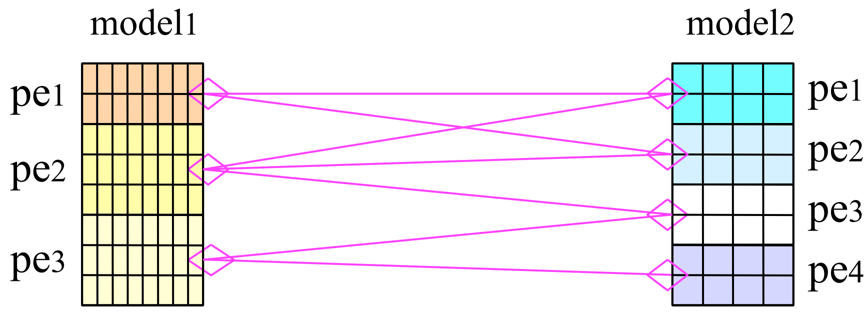
OASIS3-MCT coupled model configuration

Configuration in a **text** file *namcouple*
** now generated with OASIS GUI based on CERFACS OPENTEA **

- general characteristics of a coupled run
    - total duration
    - components
    - …
- for each exchange of coupling field :
    - source and target symbolic name (end-point communication)
    - exchange period
    - transformations/interpolations

OASIS3-MCT communication

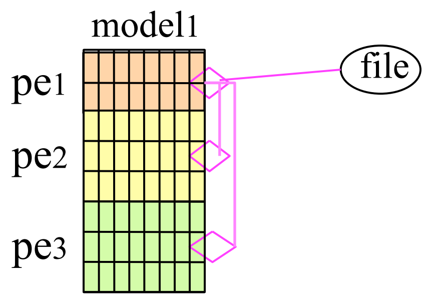
•Fully parallel communication between parallel models based on Message Passing Interface (MPI)



If required, the interpolation weights and addresses are calculated onto one model process

Interpolation per se from the source grid to the target grid is done in parallel on the source or on the target processes

•I/O functionality
(switch between coupled and forced mode):



OASIS3-MCT: interpolations & transformations

- on 2D or 3D scalar fields
- on different types of grids: lat-lon, rotated (logically rectangular), gaussian reduced, unstructured

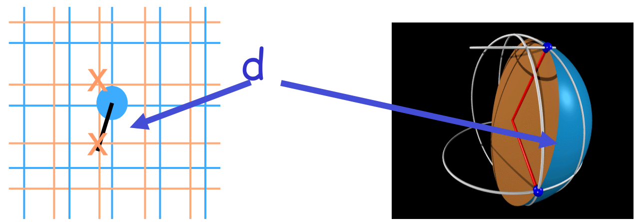
❖ <u>Transformations</u>: statistics, addition/multiplication by scalar, global conservation

❖ <u>Interpolations/regridding</u> SCRIP (Jones, 1999)

x source grid point
● target grid point

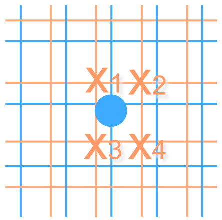
<u>n-nearest-(gaussian-weighted)-neighbours</u>: weight(x) α 1/d

d: great circle distance on the sphere:



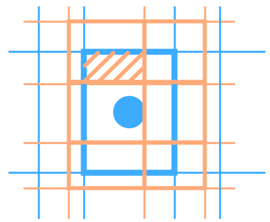
<u>bilinear interpolation</u>

- general bilinear iteration in a continuous local coordinate system using f(x) at x_1, x_2, x_3, x_4



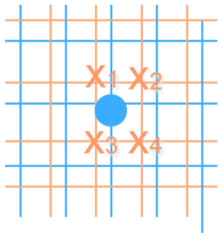
<u>conservative remapping</u>

- weight of a source cell % to intersected area

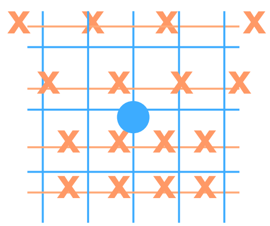


<u>bicubic interpolation:</u>

- general bicubic iterations in a continuous local coordinate system: f(x), $\delta f(x)/\delta i$, $\delta f(x)/\delta j$, $\delta^2 f/\delta i \delta j$ in x_1, x_2, x_3, x_4 for logically-rectangular grids (i,j)

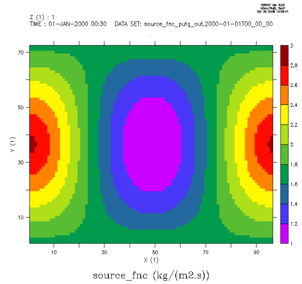


- standard bicubic algorithm: 16 neighbour points for Gaussian Reduced grids



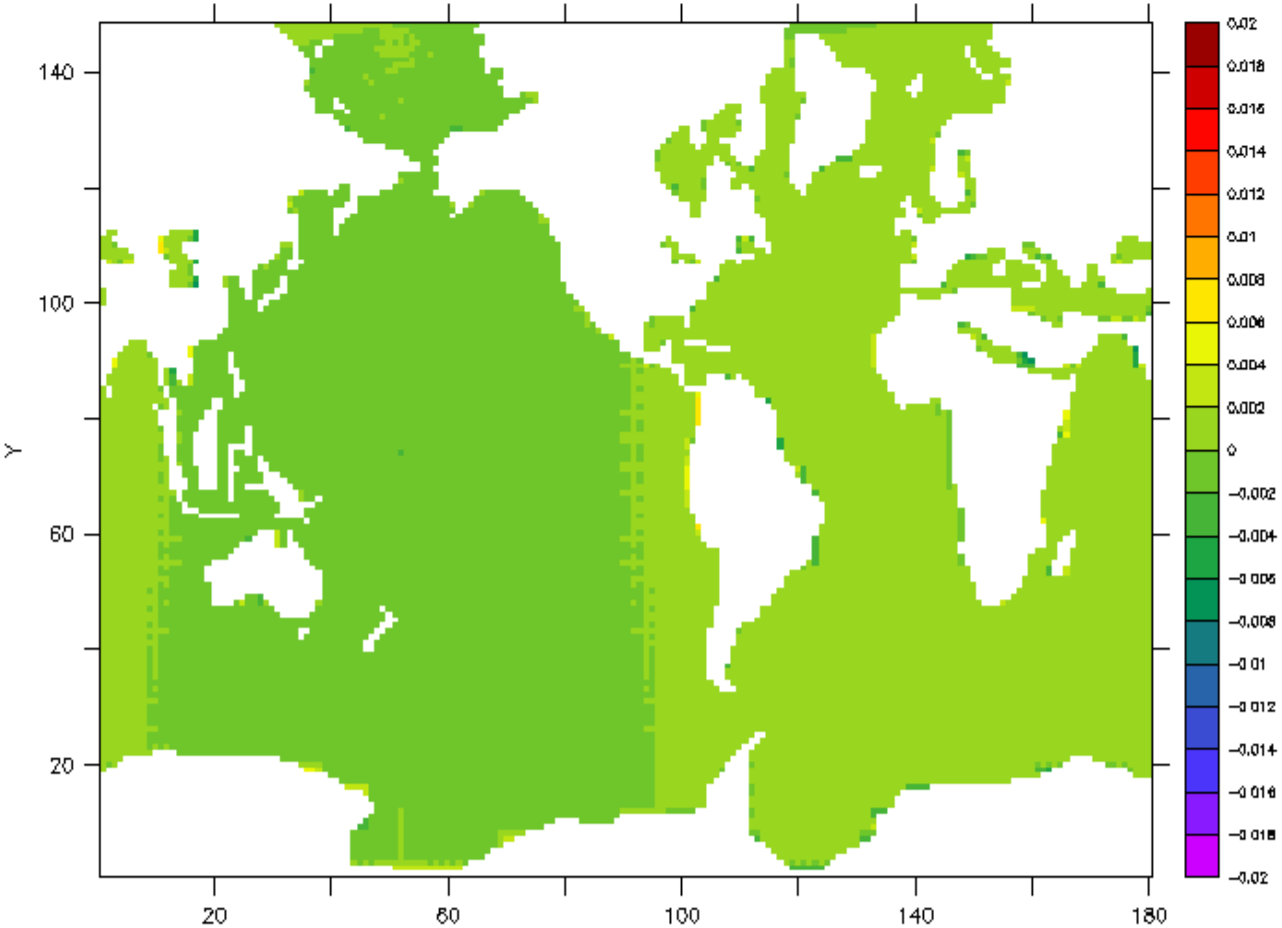
** gradients must be given as extra arguments to the oasis_put

OASIS3-MCT: interpolations & transformations



One example of bilinear interpolation error

F = 2 + cos[π * acos(cos(lon)cos(lat))]   LMDz grid (96 x 72) -> ORCA2



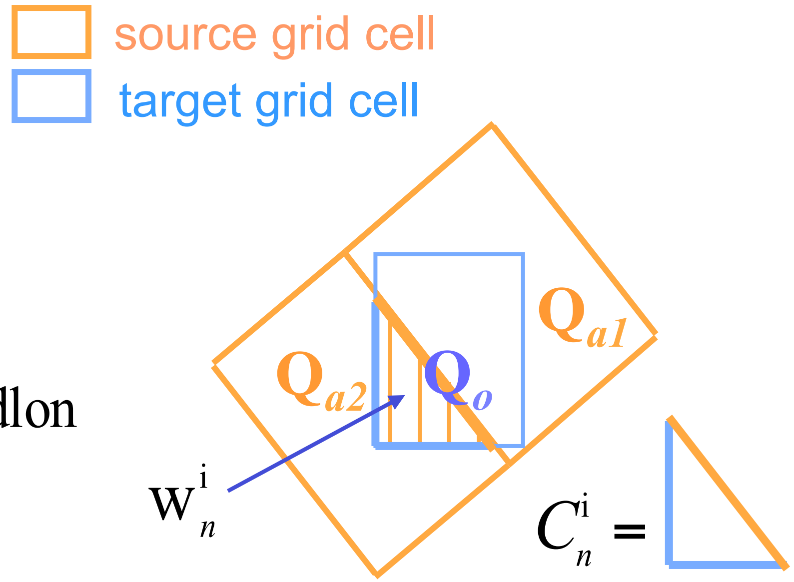
- < 0.2% whole domain; ~1% near the coastline

OASIS3-MCT: interpolations & transformations

<u>1st order conservative regridding</u>:

- conserves integral of extensive properties
- weight of a source cell α to intersected area

$$Q_o^i = \frac{1}{A_o} \sum_{n=1}^{N} Q_{a_n} w_n^i \quad \text{with} \quad w_n^i = \oint_{C_n^i} -\sin(\text{lat})\,d\text{lon}$$



<u>Actual limitations:</u>

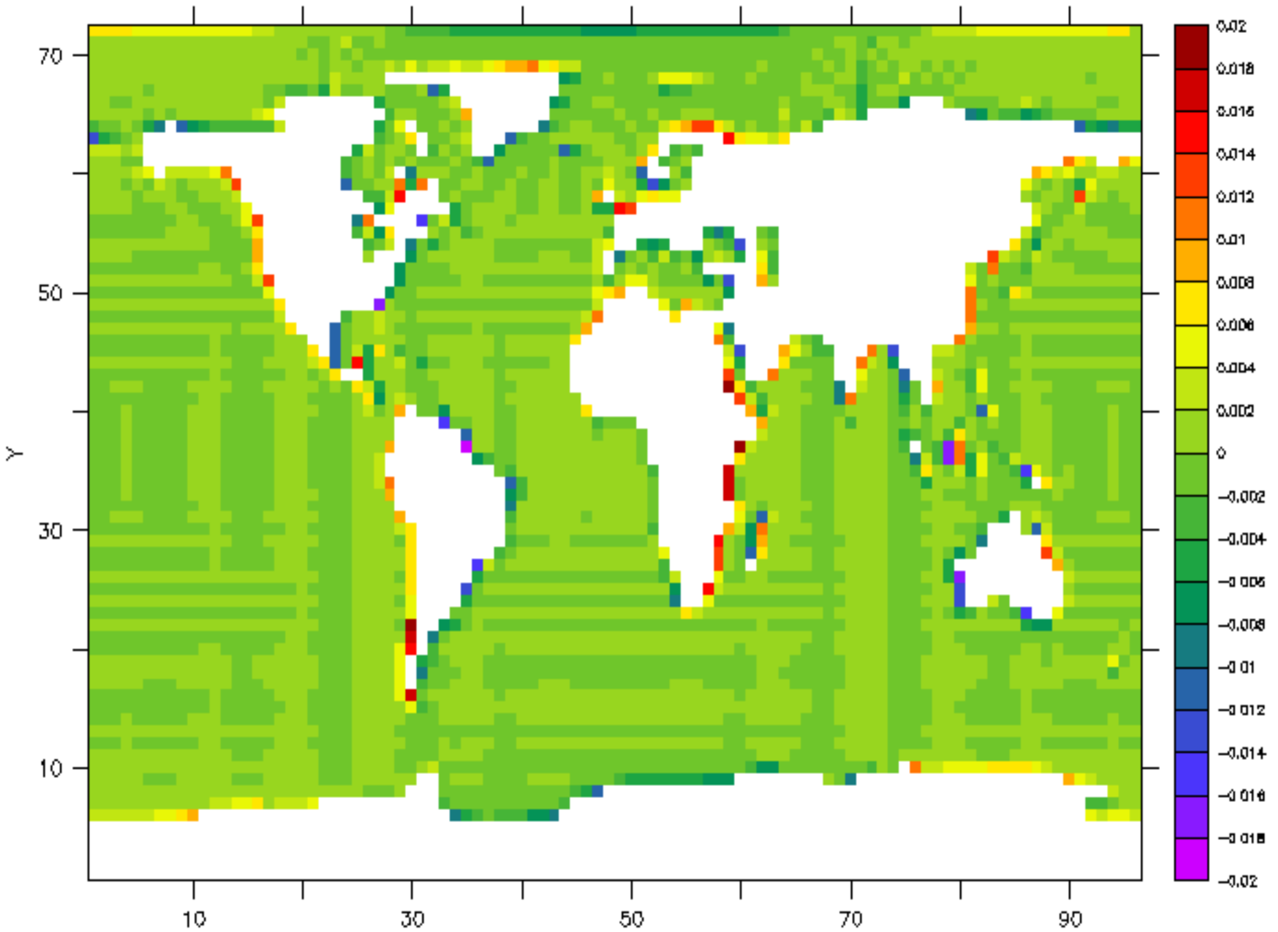
- assumes borders are linear in (lat,lon) ; uses Lambert equivalent azimuthal projection near the pole for intersection calculation
- assumes sin(lat) linear function of lon for line integral calculation
  - need to use a projection near the pole (as done for intersect. calc.)
- exact calculation is not possible as "real shape" of the borders are not known
  - could use of border middle point
  - to ensure conservation, need to normalize by true area of the cells (under work)

- ESMF new conservative regridding under evaluation (see examples/test_rmp_esmf)

OASIS3-MCT: interpolations & transformations

• One example of conservative remapping error

F = 2 - cos[π*acos(cos(lon)cos(lat))]
ORCA2 -> LMDz (96x72)



➢ < 0.2% everywhere except
~ 0.8% for LMDz last row close to the North pole
~ 2% near the coastline

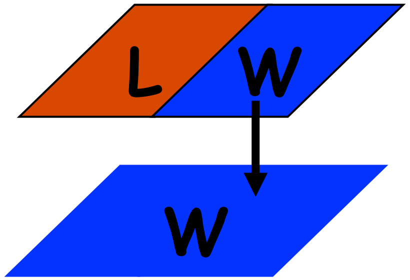
OASIS3-MCT: interpolations & transformations

Problem with non-matching sea-land masks

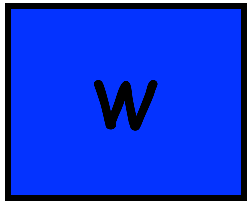
$$Q_o^i = \frac{1}{A_o}\sum_{n=1}^{N} Q_{a_n} w_n^i$$

1- Support subsurfaces in the atmosphere

and use the ocean land-sea mask in the atmosphere to determine the fractional area of each type of surface

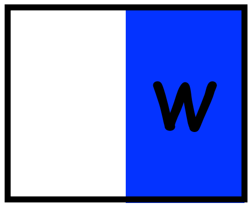


2-"DESTAREA" option

$A_o =$ 

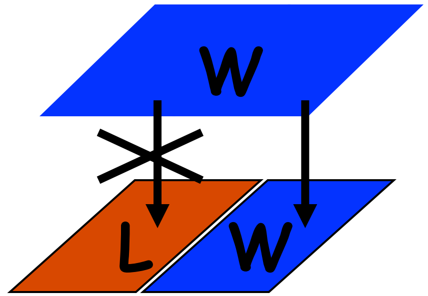
- local flux conservation
- possibly unrealistic flux values

3-"FRACAREA" option

$A_o =$ 

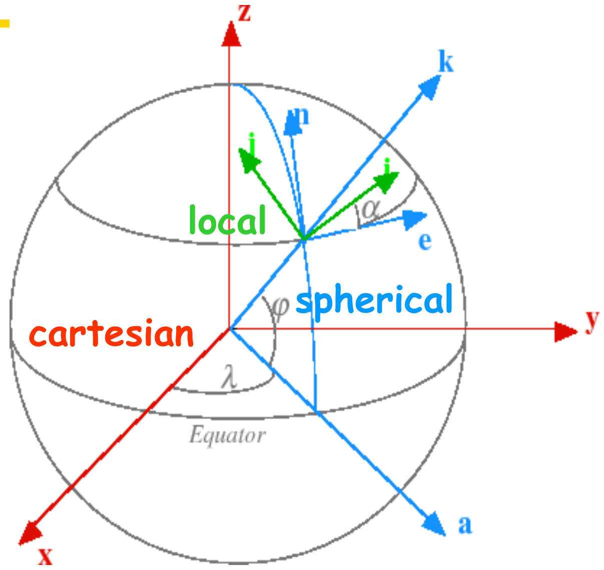
- no local conservation of flux
- realistic flux values

+ nearest non-masked value for ocean cells covered only with masked atmospheric cells (FRACNNEI)



- no local conservation of flux
- global conservation can be artificially imposed

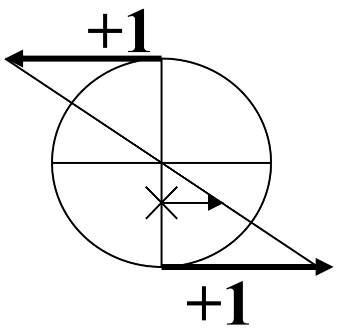
OASIS3-MCT: interpolations & transformations



Vector interpolation (winds, currents, …)

- ❖ interpolation of vectors component per component is not accurate, especially where the referential changes rapidly

Example interpolation of a zonal wind in the spherical referential near the pole



- ➢At x, one would expect a zonal wind between 0 and 1.
- ➢Interpolation comp. per comp. -> zonal wind of 1.

Solution (proposed by O. Marti, LSCE):

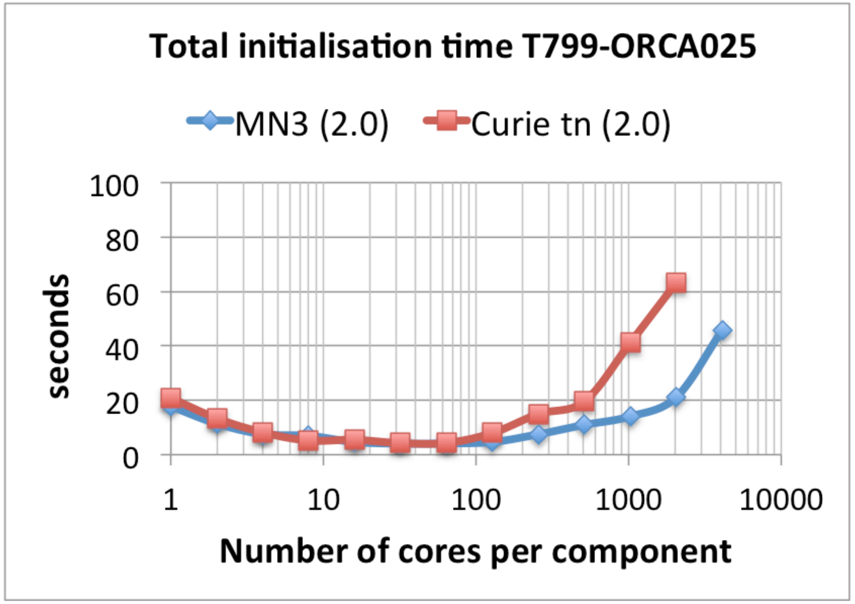
- "turn" the vector in the spherical ref. and project the resulting vector in a cartesian ref
- send the 3 components in the cartesian referential and let OASIS3-MCT interpolate them
- project back in spherical referential; check that k component is zero
- possibly "turn" the resulting vector in the target local referential

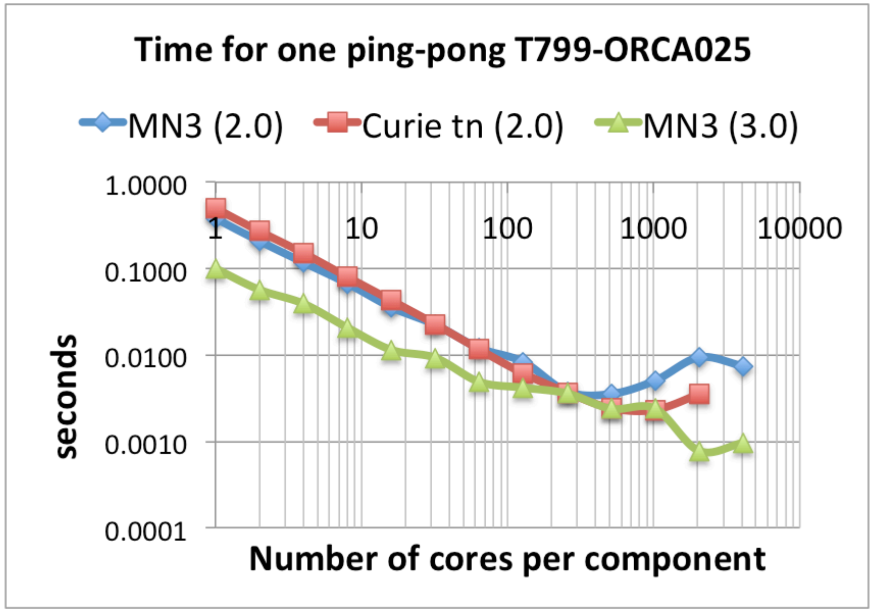
OASIS3-MCT users

- CERFACS (France):
  - NEMO ocean (ORCA025, 1021x1442) - ARPEGE atmosphere (Gaussian Red T359 grid, 181724 points).
  - Seasonal prediction exp, SPRUCE PRACE project, 27 Mhours, Bullx Curie.
  - Decadal prediction exp, SPECS PRACE project, 10 Mhours, MareNostrum3
- IPSL (France):
  - WRF atm – NEMO, 2-way nested zooms, 27-9 km (4322x1248 pts), 7 cpl fields, 1h cpl period.
  - PULSATION project (ANR), 22 Mhours on PRACE tiers-0 Bullx Curie.
- MPI-M (Germany):
  - MPI-ESM-XR: atmosphere ECHAM6 T255L95 (768x384 grid pts, ~50km) - ocean MPIOM TP6ML40 (3602x2394 grid pts, ~10km); 17 cpl fields, 1h cpl period.
- MetOffice (UK) :
  - UM global atm (N768, 1536x1152) – NEMO (ORCA012, 4320x3058), 38 cpl fields, 1h-3h cpl frequency
- BTU-Cottbus (Germany)
  - 3D coupling: COSMO-CLM regional atm (221x111x47, ~2 deg) - ECHAM global atm (T63, 192x96x47), + 2D coupling to MPI-OM ocean (254x220)
  - 6% coupling overhead observed for exchange of 6 x 3D fields every ECHAM time step
- … + many others (NICAM-NEMO, EC-Earth, …)

OASIS3-MCT performance

- Toy coupled model: ping-pong exchanges between NEMO ORCA025 grid (1021x1442) and Gaussian Reduced T799 grid (843 000)
- Bullx Curie thin nodes; Intel® procs Sandy Bridge EP; IFort 12.1.7.256, Bullx MPI 1.1.16.5
- IBM MareNostrum3: Intel Sandy Bridge processors, Intel MPI 4.1.0.024





Coupling overhead for one-year long simulation with one 1 coupling exchange every hour in each direction between codes with O(1 M) grid points running on 4000 cores/component:

- ~20 seconds for initialisation, ~9 seconds for data exchange

Conclusions and perspectives

Conclusions on OASIS3-MCT

- Good performance, removes OASIS3.3 bottleneck
- Very simple to use for traditional OASIS3.3 users (same API)

-> OASIS3-MCT most likely provides a satisfactory solution for fully parallel coupling in our climate models at the resolutions targeted operationally for the next ~5 years.

Perspectives:

- Release of OASIS3-MCT_3.0 in january 2015: more flexibility in coupling layout
- Evaluation of ESMF for off-line precomputing of interpolation weights (on going)
- IS-ENES2: Coupling technology benchmark + International Working Committee on Coupling Techologies (IWCCT, http://earthsystemcog.org/projects/iwcct/)
  - Performance of OASIS3-MCT for icosahedral grids
  - Evaluation of Open-PALM (including ONERA CWIPI library)
  - Evaluation of ESMF

The end