

Laboratoire de Météorologie-Physique UMR 6016/ OPGC

Wolfram Wobrock

Modélisation numérique des processus atmosphériques

Thématiques scientifiques:



- **Nuages troposphériques**
- **Pollution atmosphérique**



Type de modélisation

- ◆ modèles appliqués = *prévision*: modèles météorologiques
- du vent
 - de la température
 - du cycle de l'eau (vapeur, nuage, pluie, ...)
 - (des polluants)

- ◆ la modélisation de l'atmosphère au LaMP se restreint typiquement à l'échelle locale et régionale (50– 500km) (actuellement élargi: pour la modélisation du transport de pollution à l'échelle de l'Europe)
- ➔ *pas de modélisation « Climat »*

Modèles météorologiques

Éléments clés:

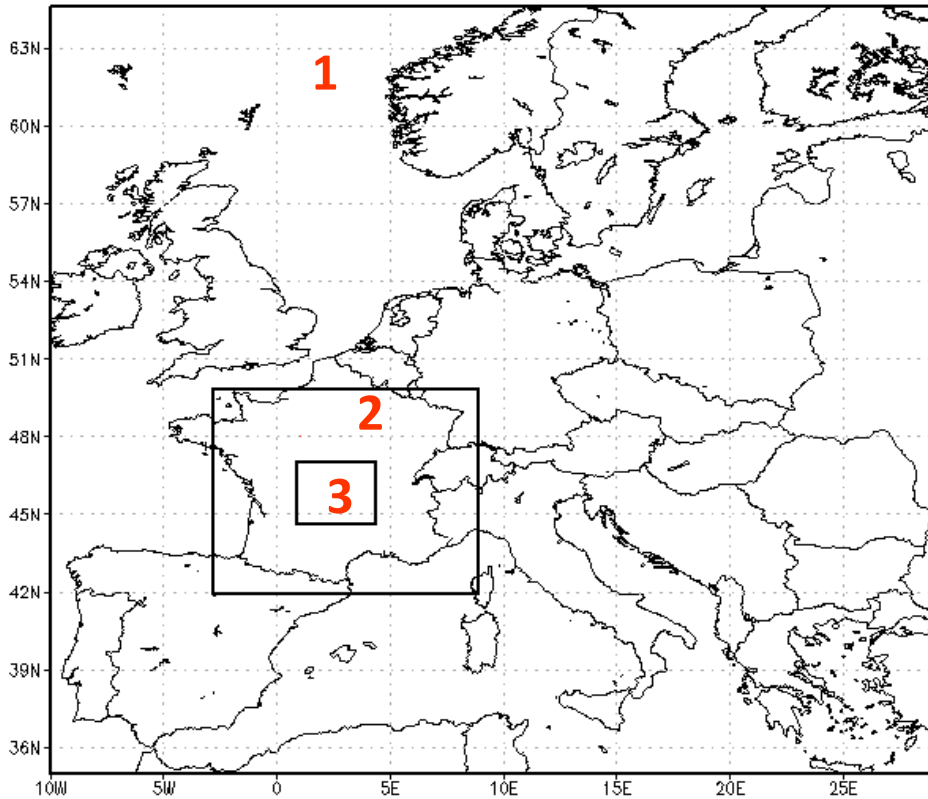
solution de l'équation du mouvement « type Navier-Stokes »
= 3 éqs. différentielles couplées, non-linéaires

+ solution des équations pour la température, pression, ...
= n éqs. différentielles couplées, non-linéaires

+ options modernes

- *projection dans un repère sphérique*
- *utilisation d'une projection cartographique (Lambert, Mercator, ...)*
- *grille verticale non-équidistante*
- *système de coordonnées non-orthogonales (coordonnée verticale suivant le terrain)*
- *imbriquer multi-domaines (grid nesting et « spawning »)*

Modèles météorologiques



Multi-domaines « nesting »

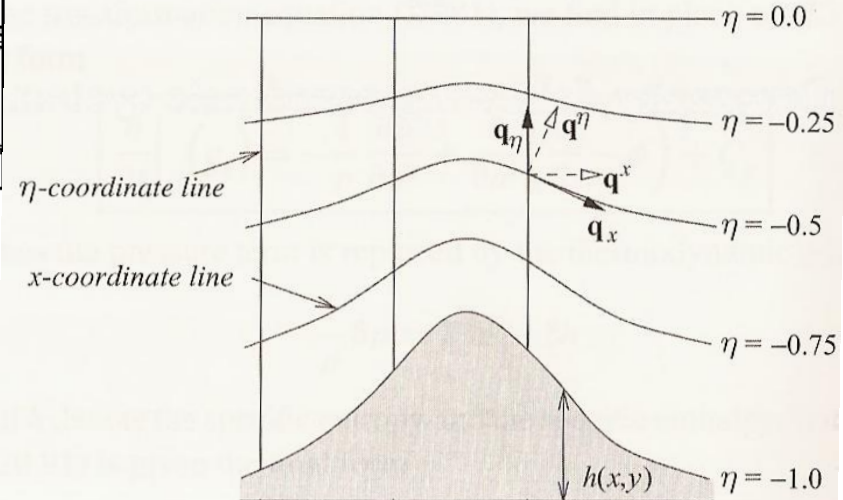
1 = 45 km

2 = 9 km

3 = 3 km

Repère non-orthogonal

Orography-following coordinate systems



Modèles météorologiques

Exemple: prévision de la vitesse verticale w en coordonnées sphériques

$$\frac{Dw}{Dt} - \frac{1}{r}(u^2 + v^2) = -\frac{1}{\rho} \frac{\partial p}{\partial r} + \frac{p}{\rho} \sigma + f^* u + gB + \frac{1}{\rho} \left[(\nabla \cdot \tau)_3 - \frac{1}{r}(\tau_{11} + \tau_{22}) \right]$$

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + u \frac{1}{r \cos \phi} \frac{\partial}{\partial \lambda} + v \frac{1}{r} \frac{\partial}{\partial \phi} + w \frac{\partial}{\partial r}$$

Après transformation dans le repère non-orthogonal

$$\begin{aligned} & \delta_t (\overline{\rho^\zeta w})^t + \delta_x (\overline{\rho^x U^\zeta w^x}) + \delta_y (\overline{\rho^y V^\zeta w^y}) + \delta_\zeta (\overline{\rho^\zeta \omega^\zeta w^\zeta}) - \frac{\overline{\rho^\zeta}}{\overline{\Gamma^\zeta R}} (\overline{u^{2x} + v^{2y}})^\zeta = \\ & -\overline{\Gamma^2}^\zeta \cos \phi \left[\delta_\zeta p - \left(\frac{p \overline{\Gamma^2 F_\zeta}}{\rho} \right)^\zeta \delta_\zeta \left(\frac{\rho}{\overline{\Gamma^2 F_\zeta}} \right) \right] + \overline{\rho^x U^x}^\zeta \cos \phi f^* + g \overline{\rho^\zeta B}^\zeta + \frac{1}{\overline{\Gamma^\zeta} \cos \phi} \delta_x \left(\overline{\sqrt{J}}^{x\zeta} \tau_{13} \right) \\ & + \frac{1}{\overline{\Gamma}} \delta_y \left(\overline{\sqrt{J}}^{y\zeta} \tau_{23} \right) + \delta_\zeta \tau_{33} + \frac{1}{\overline{\Gamma}} \delta_\zeta \left(\frac{\overline{\sqrt{J J^{13}}^\zeta \tau_{13}^\zeta}}{\cos \phi + \overline{\sqrt{J J^{23}}^\zeta \tau_{23}^\zeta}} \right) - \frac{\sqrt{J}}{\overline{\Gamma^\zeta R}} (\overline{\tau_{11}}^\zeta + \overline{\tau_{22}}^\zeta) \end{aligned}$$

Solution numérique

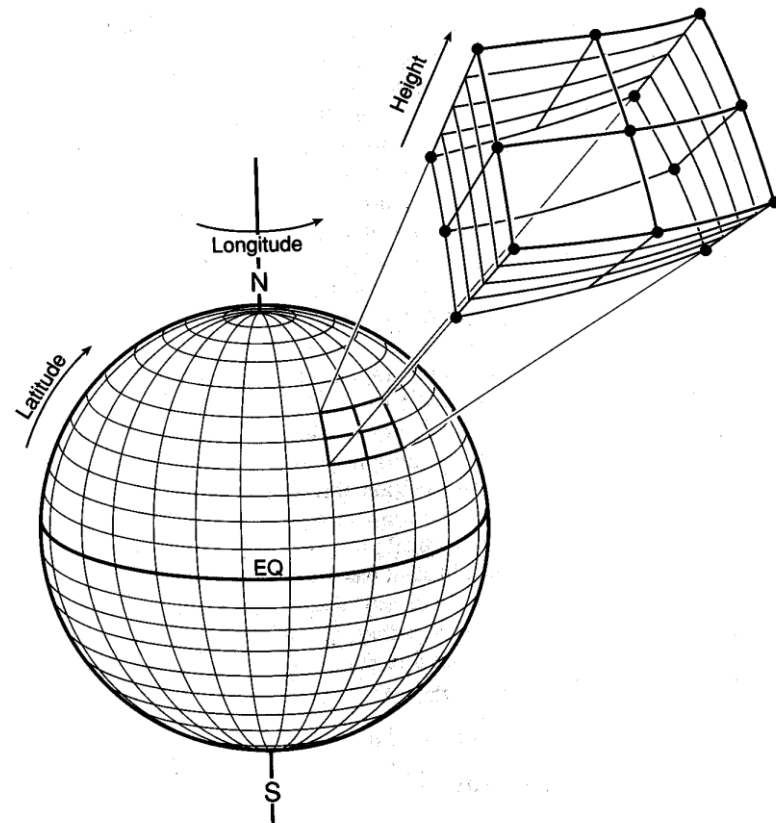
Numériser les équations de prévisions en espace et en temps:

➔ définir une grille horizontale (Δx , Δy) et verticale (Δz) et un pas de temps Δt

avec Δx , Δy , Δz , Δt fins

➔ meilleure solution numérique

➔ calcul lourd

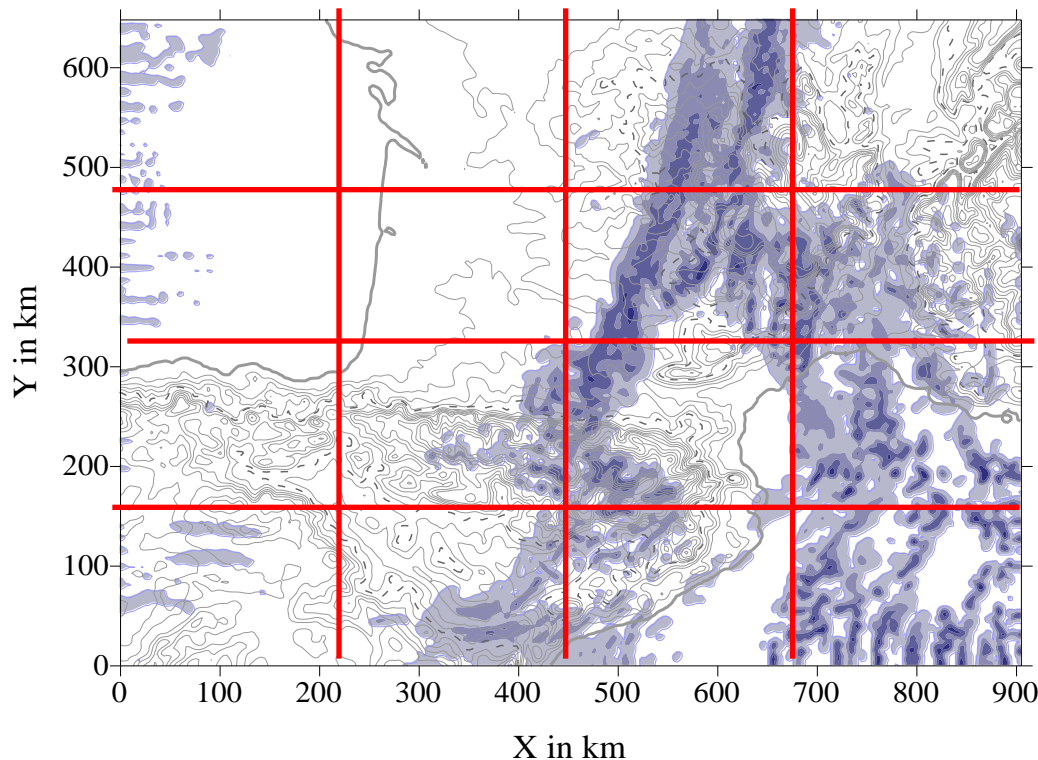


Aides techniques:

1. vectorisation
2. calcul parallèle
(openMP, MPI)

Parallélisme dans les modèles météorologiques

Méthodologie: séparation du domaine en sous-domaines « parallèles »



→
**16 sous-
domaines**
=
16 CPU

~~→
**16 fois plus
rapide**~~

**problèmes: I) échange des conditions limites après chaque Δt
II) le taux des opérations est non-équilibré**

Plateforme de calcul parallèle

IDRIS/CNRS : Orsay	<u>IBM SP Power6</u> - 3584procs; Vargas	<u>IBM Blue Gene/P</u> - 40960procs Babel
CINES: Montpellier	<u>SGI Altix ICE</u> - 23040procs Jade	<u>IBM Power7+</u> -256procs Yoda
au LaMP:	1 SGI et 1 DELL à 32 procs	

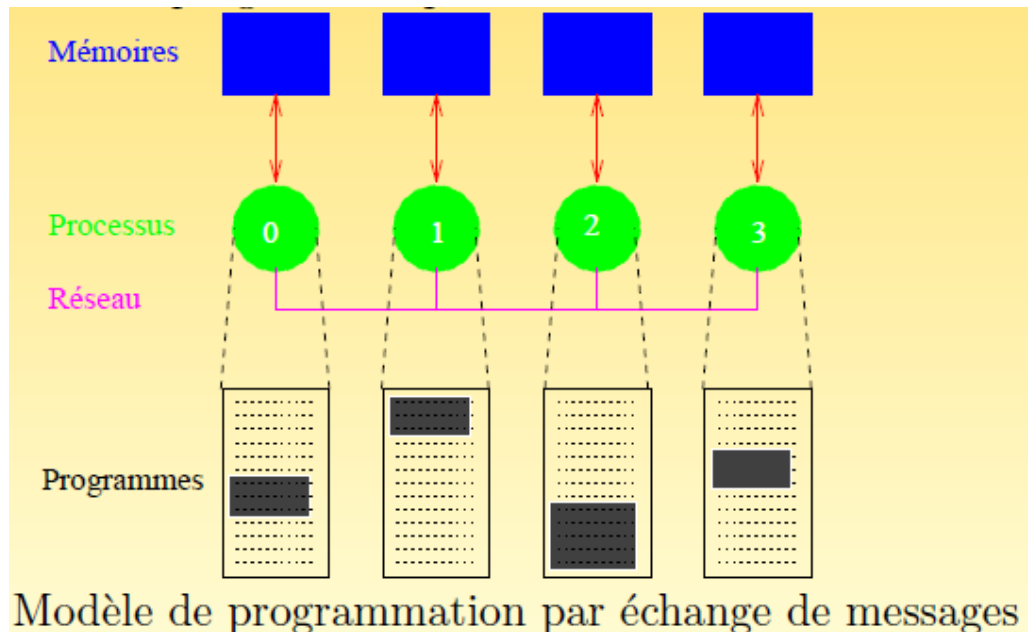
calcul parallèle est possible sur ces plateformes en:

openMP, MPI et « hybride » (MPI+openMP)

Calcul MPI

Dans le modèle de programmation par « échange de messages » :

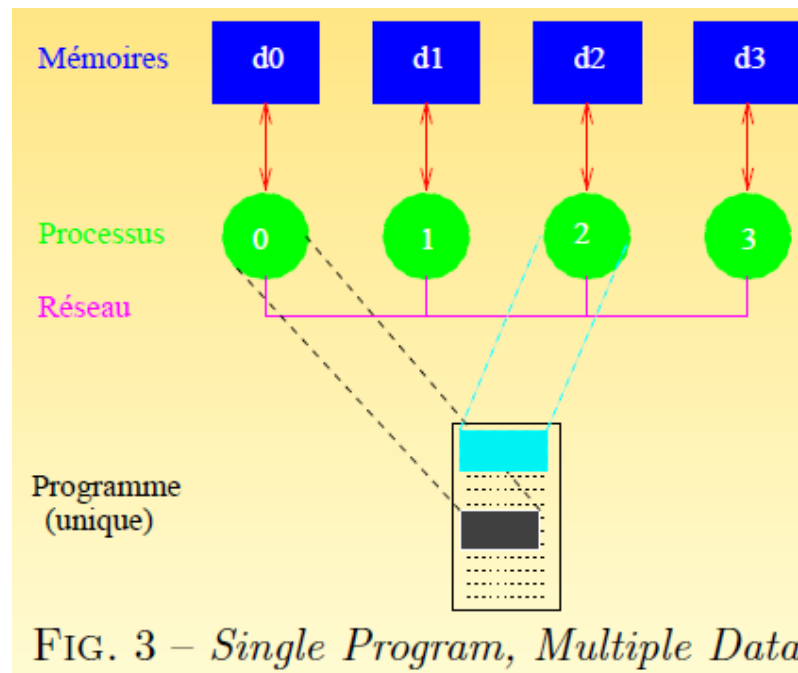
- le programme est écrit dans un langage classique (Fortran, C, C++, etc.) ;
- toutes les variables du programme sont privées et résident dans la mémoire locale allouée à chaque processus ;
- le même programme est exécuté par tous les processus
- chaque processus exécuté éventuellement des parties différentes d'un programme



Calcul MPI

Dans le modèle de programmation par « échange de messages » :

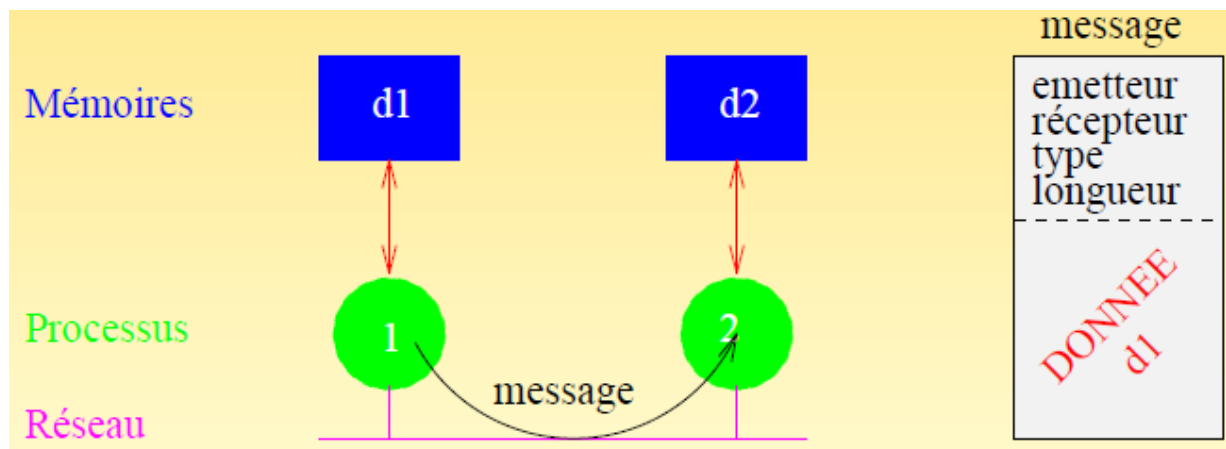
- une donnée est échangée entre deux ou plusieurs processus via un appel, dans le programme, à des sous-programmes particuliers.



Calcul MPI

Dans le modèle de programmation par « échange de messages » :

- Un message est constitué de paquets de données transitant du processus émetteur au(x) processus récepteur(s)
- En plus des données (variables scalaires, tableaux, etc.) à transmettre, un message doit contenir les informations suivantes :
 - l'identificateur du processus émetteur ;
 - le type de la donnée ;
 - sa longueur ;
 - l'identificateur du processus récepteur.



Exemple

```
1 program point_a_point
2   use mpi
3   implicit none
4
5   integer, dimension(MPI_STATUS_SIZE) :: statut
6   integer, parameter      :: etiquette=100
7   integer                  :: rang,valeur,code
8
9   call MPI_INIT(code)
10
11  call MPI_COMM_RANK(MPI_COMM_WORLD,rang,code)
12
13  if (rang == 2) then
14    valeur=1000
15    call MPI_SEND(valeur,1,MPI_INTEGER,5,etiquette,MPI_COMM_WORLD,code)
16  elseif (rang == 5) then
17    call MPI_RECV(valeur,1,MPI_INTEGER,2,etiquette,MPI_COMM_WORLD,statut,code)
18    print *,'Moi, processus 5, j''ai reçu ',valeur,' du processus 2.'
19  end if
20
21  call MPI_FINALIZE(code)
22
23 end program point_a_point
```

```
> mpiexec -n 7 point_a_point
```

```
Moi, processus 5, j'ai reçu 1000 du processus 2
```

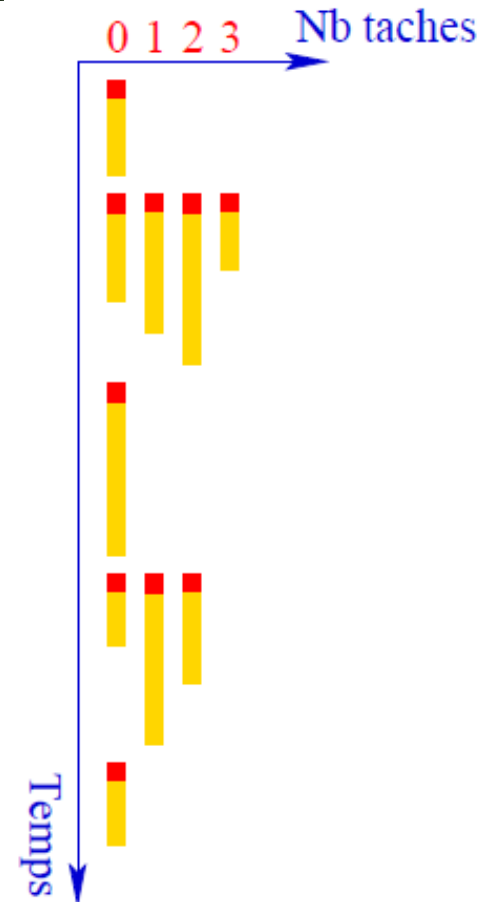
Calcul openMP

- Un programme OpenMP est une alternance de régions séquentielles et de régions parallèles.
- Une région séquentielle est toujours exécutée par la tâche maître, celle dont le rang vaut 0.
- Une région parallèle peut être exécutée par plusieurs tâches à la fois.
- Les tâches peuvent se partager le travail contenu dans la région parallèle

Difficultés: gestion de la mémoire « privée » et « partagée »

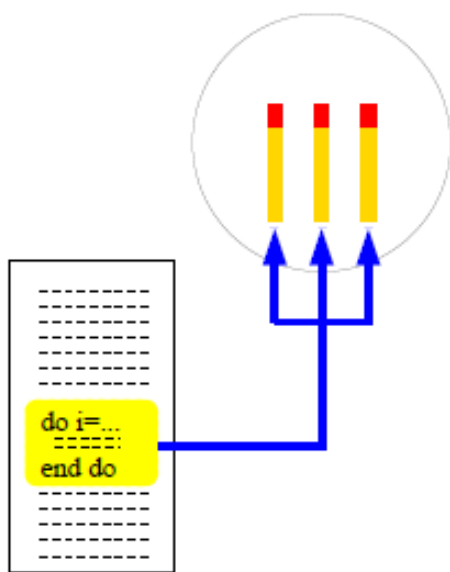
Espace **mémoire partagée** par tous les processus ; on parle alors de variable partagée

Espace **mémoire locale** d'un processus léger ; on parle alors de variable privée.

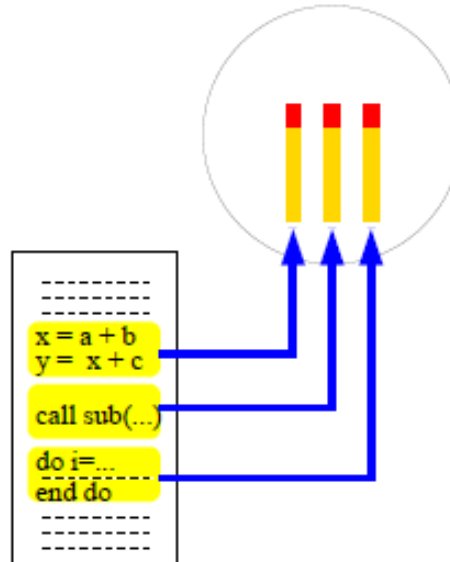


Calcul openMP

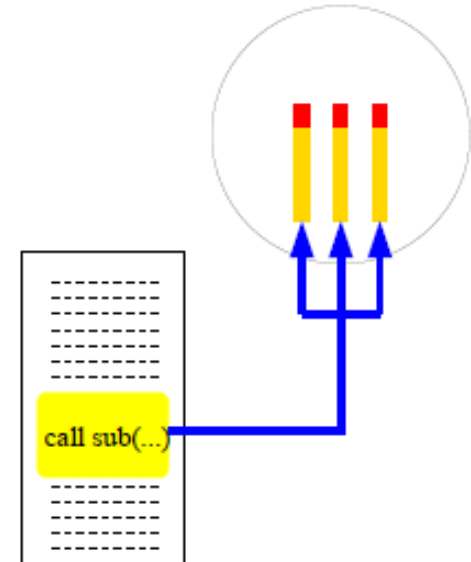
- ☞ Le partage du travail consiste essentiellement à :
- ⇒ exécuter une boucle par répartition des itérations entre les tâches ;
 - ⇒ exécuter plusieurs sections de code mais une seule par tâche ;
 - ⇒ exécuter plusieurs occurrences d'une même procédure par différentes tâches (*orphanning*).



Boucle parallèle
(Looplevel parallelism)



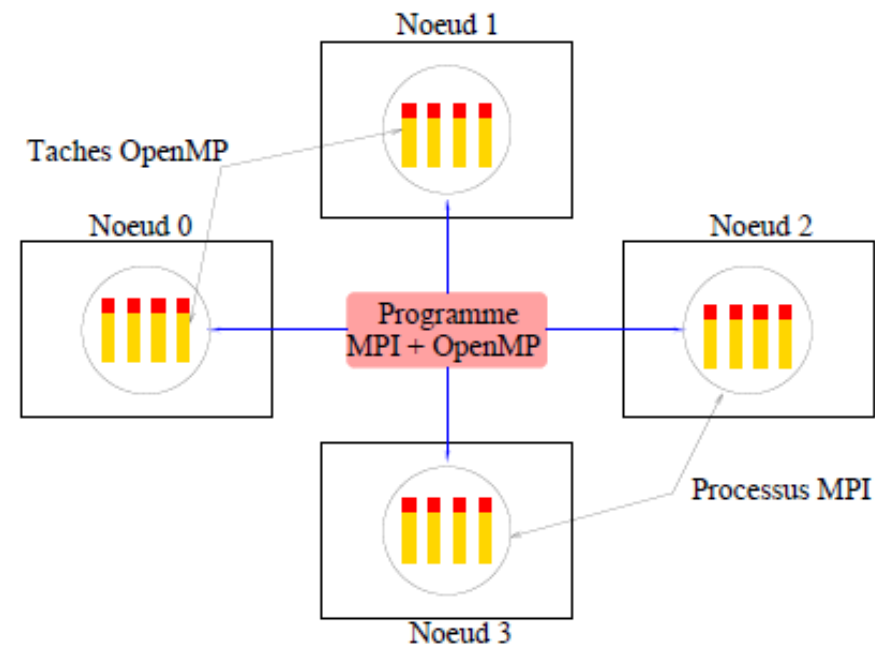
Sections parallèles



Procédure parallèle (orphanning)

Calcul hybride

- ➔ MPI est utilisé en général sur des machines multiprocesseurs à mémoire distribuée.
- ➔ OpenMP est utilisé sur des machines multiprocesseurs à mémoire partagée.
- ➔ Sur une grappe de machines indépendantes (nœuds) multiprocesseurs à mémoire partagée, la mise en œuvre d'une parallélisation à deux niveaux (MPI et OpenMP) dans un même programme peut être un atout majeur pour les performances parallèles du code.



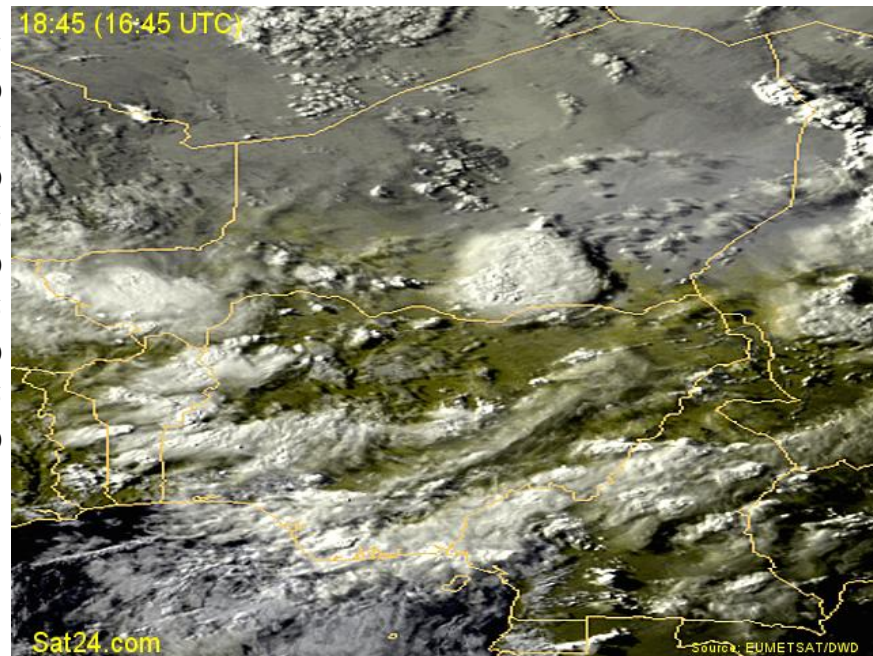
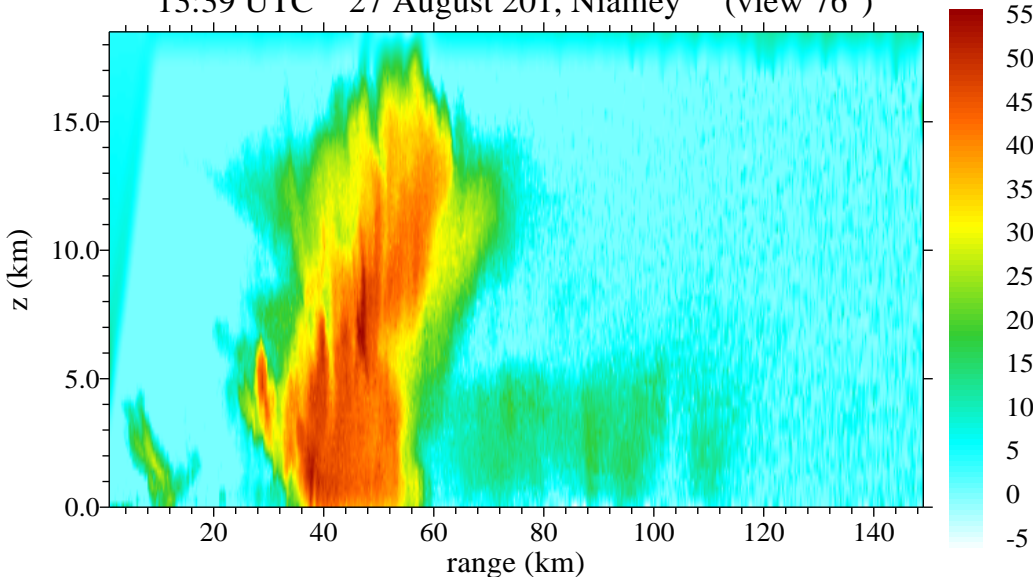


résultats

Résultats: - MeghaTropiques 2010

MIT-C-band radar at Niamey (by courtesy of Earle Williams)

13:39 UTC 27 August 201, Niamey (view 76°)

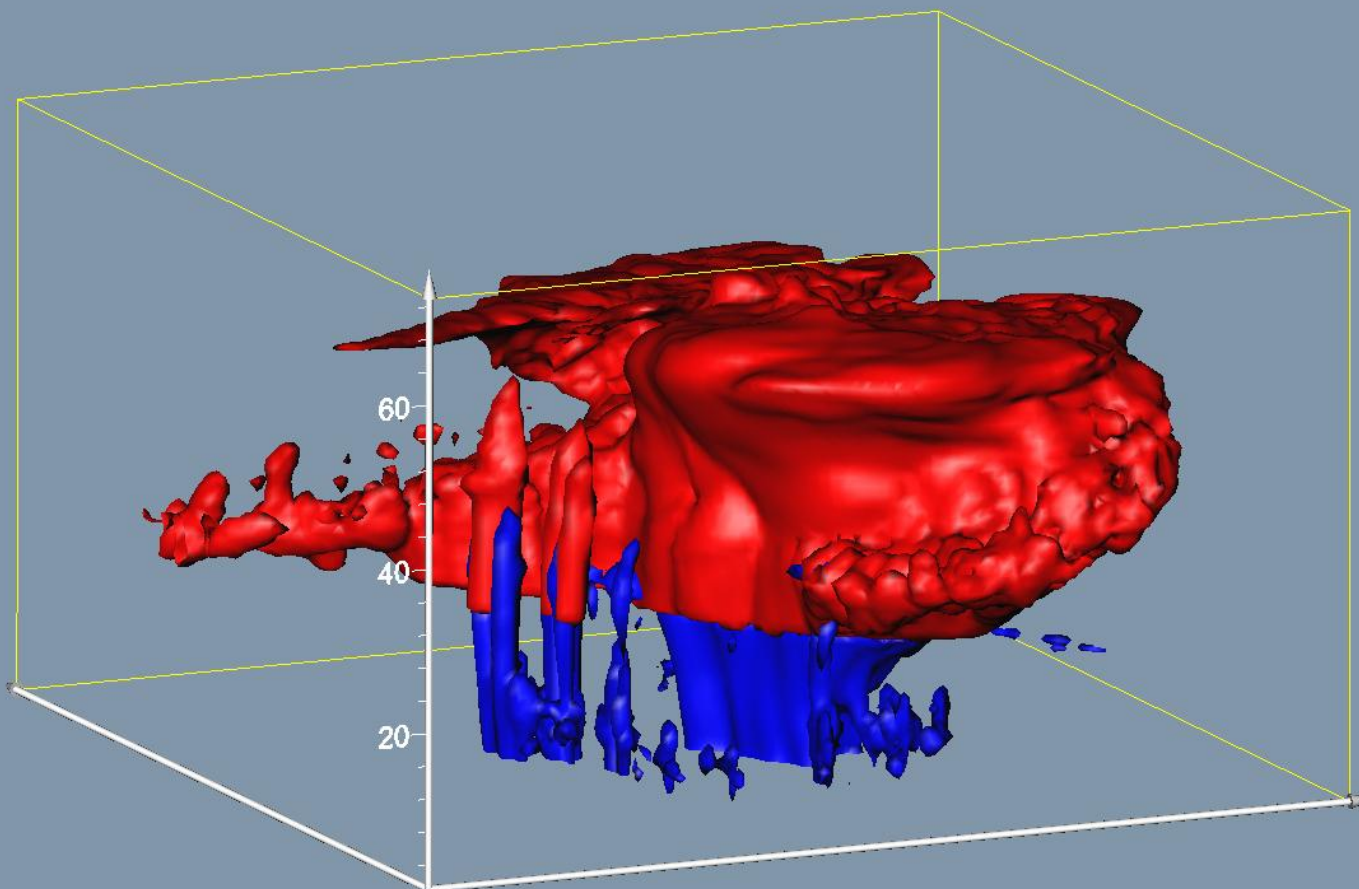


2006:
- FSSP
- 2DC
- 2DP

2010:
- 2DS - Rasta
- CIP - FSSP
- PIP - 2DC+P

+ surface:
2 Radars
(MIT, IRD)

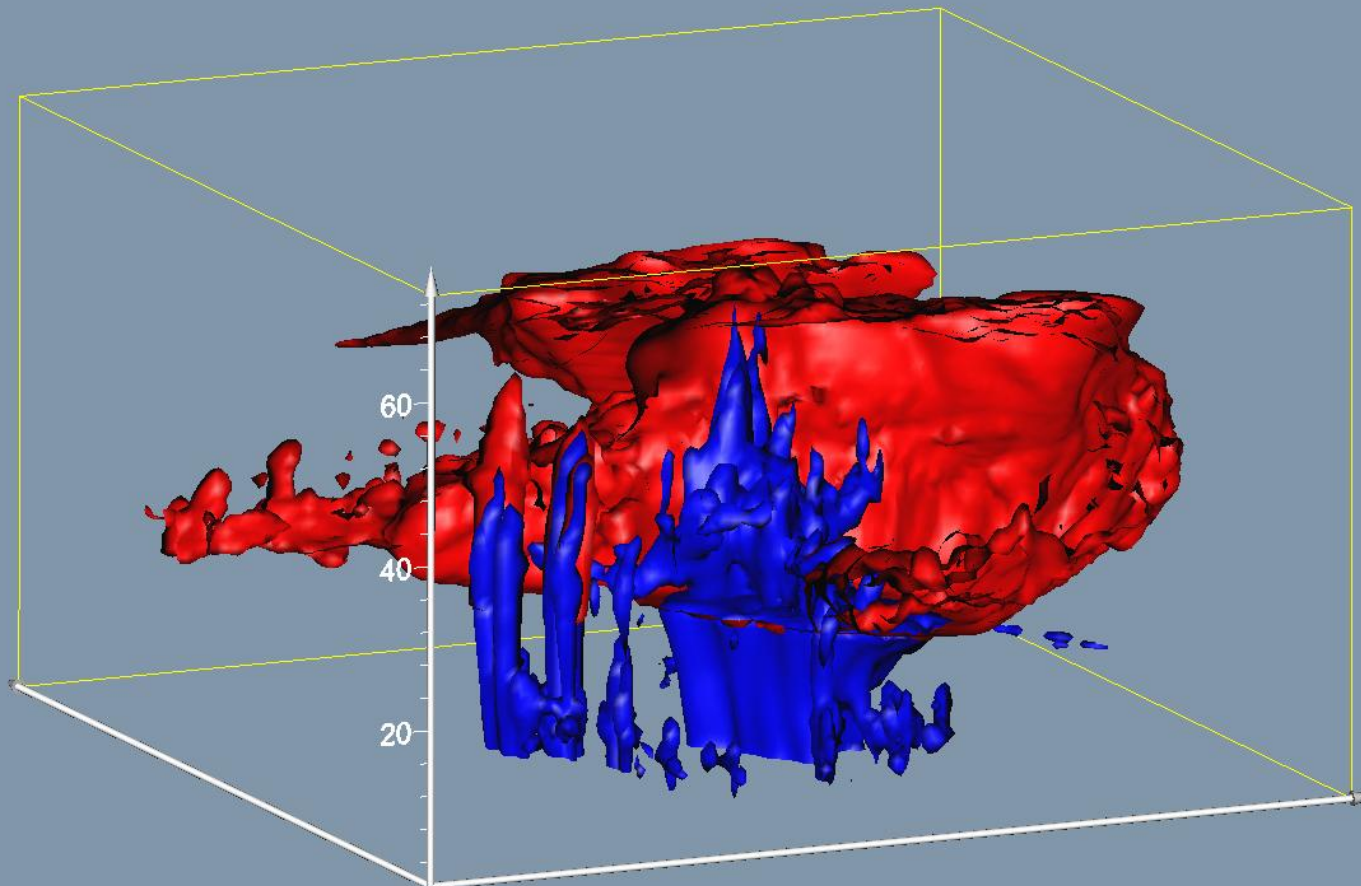
AMMA 2006 - MeghaTropiques 2010



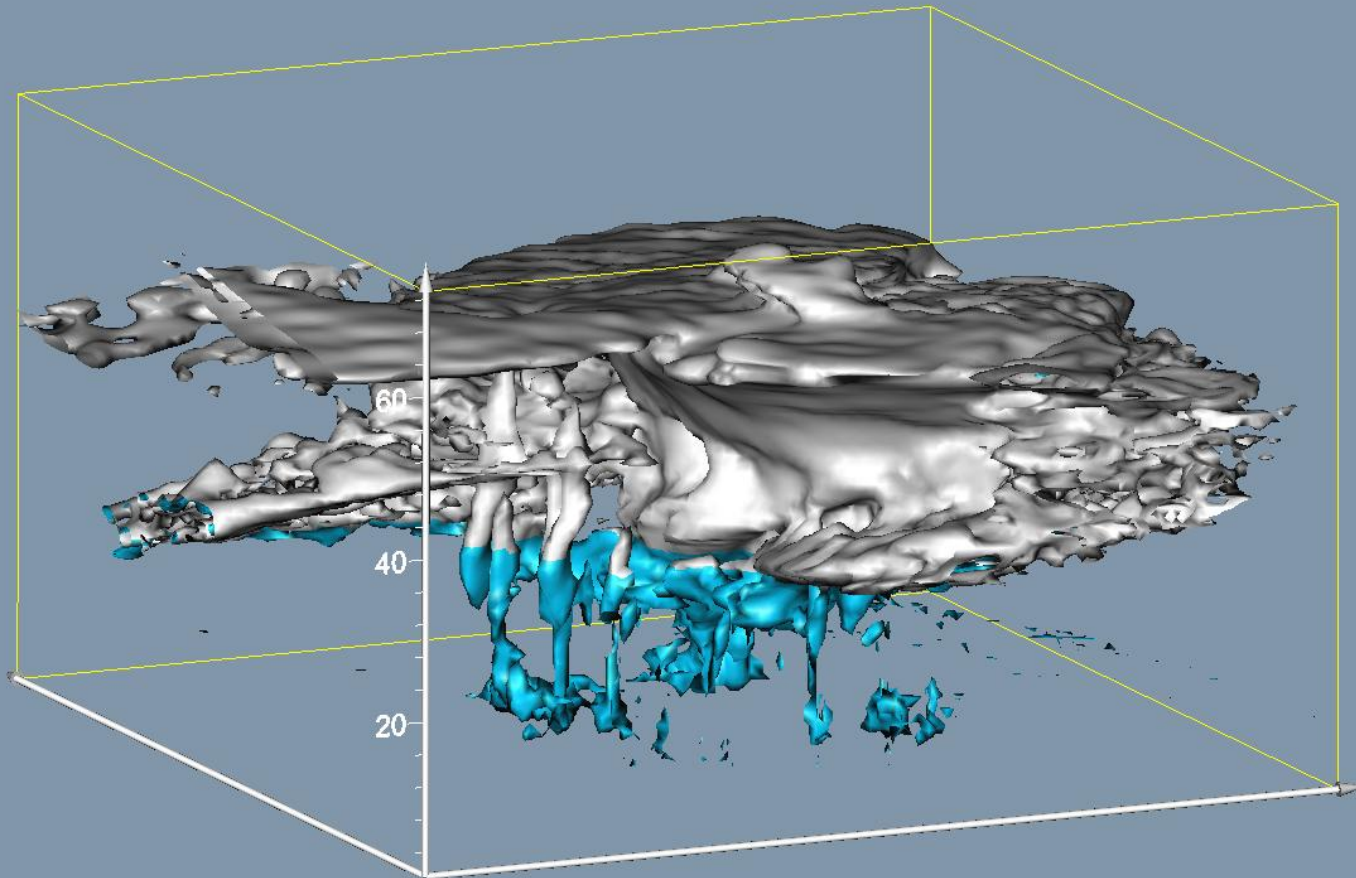
AMMA 2006 - MeghaTropiques 2010

Ice > 10 mg/m³

Water > 100 mg/m³

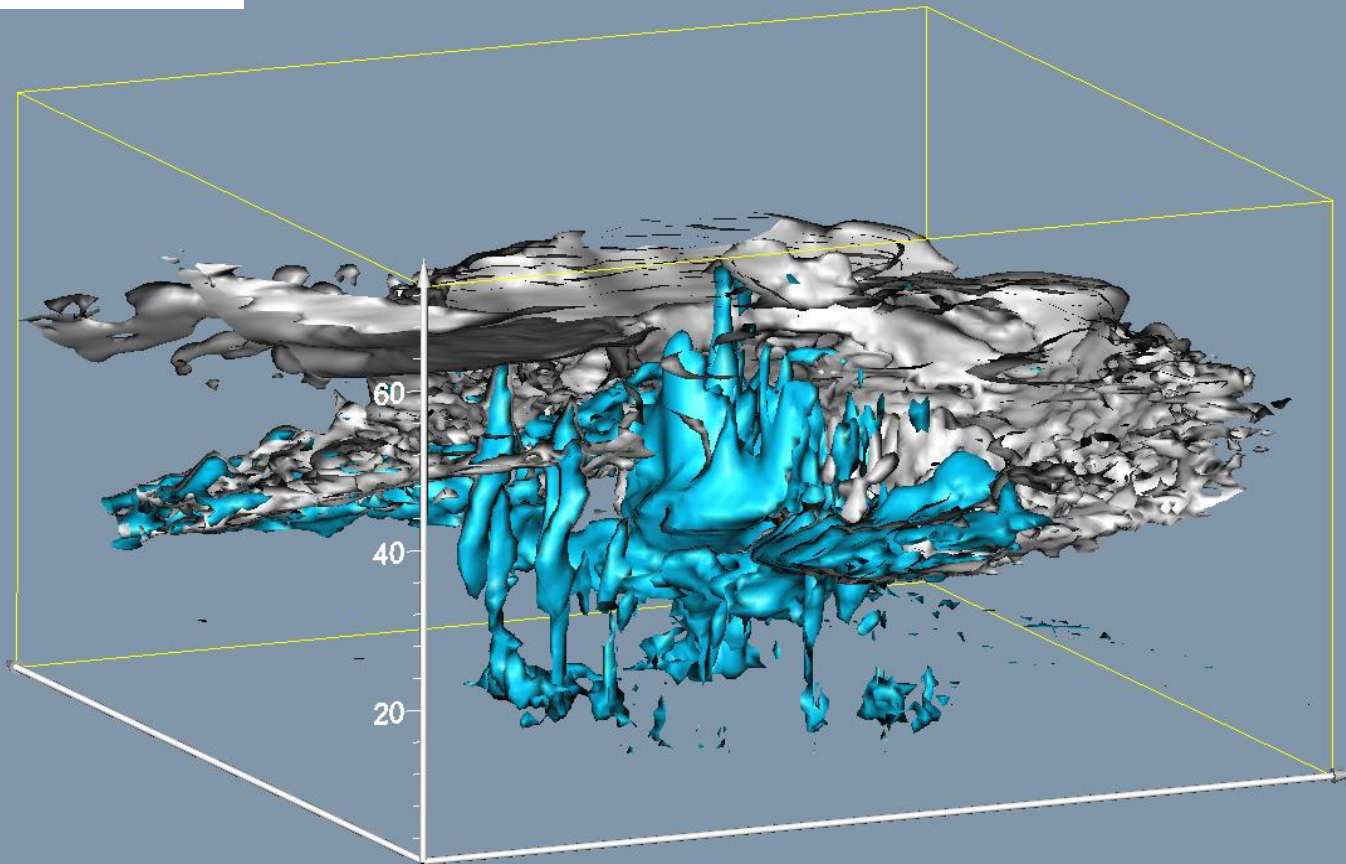


AMMA 2006 - MeghaTropiques 2010

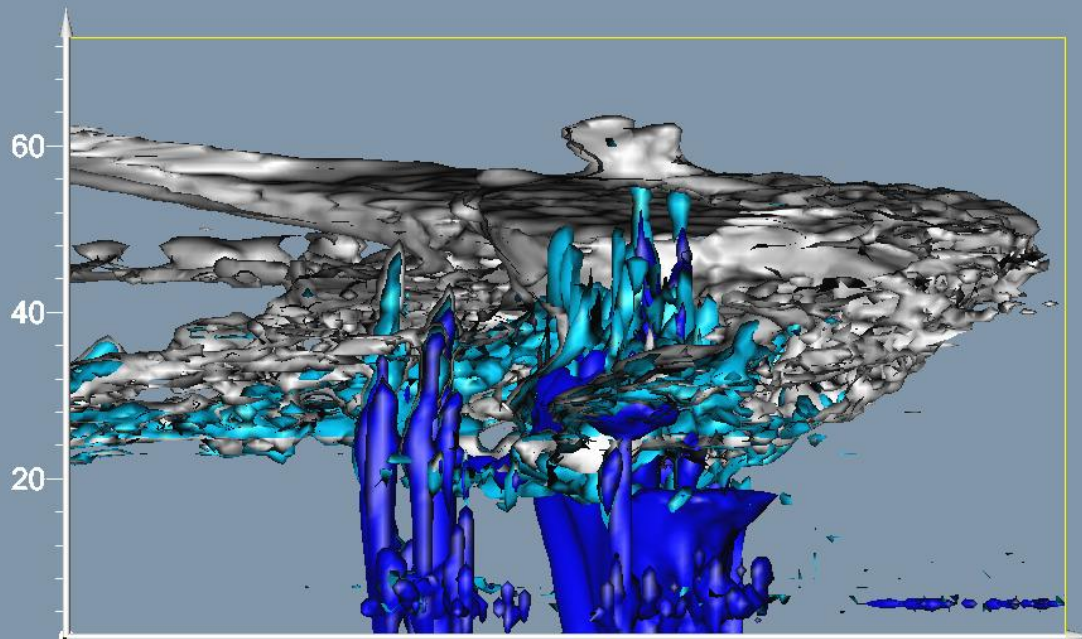


AMMA 2006 - MeghaTropiques 2010

RHI > 110 %
RH > 100 %



AMMA 2006 - MeghaTropiques 2010



Dynamique et Microphysique d'un orage

