

# Pistes de développement LMDZ(-ISO)

# Plan

- Boucles (1)
  - Où mettre l'indice des isotopes ?
- Pointeurs et Tableaux
  - 2 tableaux mis dans 1 seul (tableau de structures pointeurs)
  - 1 seul tableau séparé en 2 (pointeurs sur éléments de tableau)
  - Accéder aux éléments d'un seul tableau avec un tableau d'indice (indirections)
- Boucles (2)
  - Se passer des #IFDEF ?
- Procédures
  - Arguments optionnels
  - Tableaux de taille nulle

# Où placer l'indice sur les isotopes ?

`tab_iso(niso, klon, klev)`

La boucle sur `niso` doit être la plus interne

`tab_iso(klon, klev, niso)`

La boucle sur `niso` doit être la plus externe

# Pointeurs et Tableaux : H2O+ISOTOPES

L'eau et les isotopes sont séparés dans 2 tableaux par exemple :

q\_seri(klon, klev) et xt\_seri(klon, klev, niso)

regrouper tout dans 1 seul tableau

qxt\_seri(klon, klev, niso+1)

simplification de l'écriture et de la maintenance du code

boucle sur les niso+1 grandeurs (eau + ses isotopes)

# Conserver les deux en même temps : 2 tableaux $\neq$ dans 1 grand tableau

On définit et initialise le grand tableau de cette façon :

```
REAL, DIMENSION(klon,klev),      TARGET :: tu
REAL, DIMENSION(klon,klev,niso),  TARGET :: xtu
```

```
TYPE ptr
```

```
    REAL, DIMENSION(:,.), POINTER :: p
```

```
END TYPE ptr
```

```
TYPE(ptr), DIMENSION(:), ALLOCATABLE :: tab_iso
```

```
ALLOCATE(tab_iso(1+niso))
```

```
tab_iso(1)%p => tu
```

```
DO K = 1, niso
```

```
    tab_iso(K+1)%p => xtu(:, :, K)
```

```
ENDDO
```

On utilise le grand tableau comme ceci :

```
DO K = 1, SIZE(tab_iso)
```

```
    DO J = 1, klev
```

```
        DO I = 1, klon
```

```
            tab_iso(K)%p(I,J) = 1.
```

```
        ENDDO
```

```
    ENDDO
```

```
ENDDO
```

# Conserver les deux en même temps : 1 grand tableau divisé en 2 tableaux

On définit et initialise les sous-tableaux de cette façon :

```
REAL, DIMENSION(klon,klev,niso+1), TARGET ::  
tab_iso
```

```
REAL, DIMENSION(klon,klev), POINTER :: tu
```

```
REAL, DIMENSION(klon,klev,niso), POINTER :: xtu
```

```
tu => tab_iso(:, :, 1)
```

```
xtu => tab_iso(:, :, 2:)
```

On doit passer les sous tableaux aux routines avec klon et klev :

```
CALL mysub( ... , tu, xtu)
```

```
SUBROUTINE mysub( ... , tu, xtu)
```

```
REAL, DIMENSION(klon,klev) :: tu
```

```
REAL, DIMENSION(klon,klev,niso) :: xtu
```

# ATTENTION À L'ALIASING !

Pointeurs sur tableaux :

le compilateur ne sait plus si il accède déjà à une zone de la mémoire avec le parallélisme...

# Indirections (tableau d'indice)

Similaire aux pointeurs sur les sous tableaux mais sans risque aliasing.

```
REAL, DIMENSION(klon,klev,niso) :: tab_iso
```

```
INTEGER :: ISO_IND = (/ iso_eau, iso_HDO, iso_H218O /)
```

```
DO K= 1, size(ISO_IND)
```

```
  Kiso = ISO_IND(K)
```

```
  tab_iso(l, J, Kiso)
```

```
ENDDO
```

On peut passer des sous tableaux en argument d'une SUBROUTINE :

```
CALL subiso(... , tab_iso(:, :, iso_eau) , ...)
```

```
SUBROUTINE subiso(... , tab_iso_eau , ...)
```



# Enlever les #IFDEF avec des boucles

```
REAL, ALLOCATABLE, DIMENSION(:, :, :) :: tab_iso, tab_iso_vider
```

```
ALLOCATE (tab_iso(klon,klev,niso), tab_iso_vider(klon,klev,0)) ! tab_iso_vider n'a pas d'éléments
```

un seul tableau pour tout et 1 seule boucle pour H2O+isotopes

```
DO K= 1, size(tab_iso,3)
```

```
  tab_iso(I,J,K) = 1. ! on execute la boucle normalement
```

```
ENDDO
```

un tableau séparé pour les isotopes de l'eau

```
DO K= 1, size(tab_iso_vider,3) ! (#IFDEF ISO : ISO n'est pas défini ⇔ size(tab_iso_vider,1) =0 )
```

```
  tab_iso_vider(I,J,K) = 1. ! on n'entre pas dans la boucle
```

```
ENDDO
```

un seul tableau pour tout et H2O + 1 boucle pour les isotopes

```
DO K= 2, size(tab_iso,3) ! (#IFDEF ISO : ISO n'est pas défini ⇔ size(tab_iso,1) =1 )
```

```
  tab_iso(I,J,K) = 1. ! on n'entre pas dans la boucle
```

```
ENDDO
```

# Procédures avec et sans les isotopes

- Passer des tableaux de taille nulle  
et utiliser des boucles non exécutées
- Utiliser des arguments optionels :

```
CALL twotabs(tu,xtu)
```

```
SUBROUTINE twotabs(tu,xtu)
```

```
REAL, DIMENSION(klon,klev) :: tu
```

```
REAL, DIMENSION(klon,klev,niso), OPTIONAL :: xtu
```

```
IF ( PRESENT(xtu) ) THEN
```

```
...
```

proposition: 1 tableau,  $k_{\text{iso}} = 1 \Leftrightarrow \text{eau}$

- 1 grand tableau H<sub>2</sub>O+isotopes
- plus de #IFDEF en utilisant : boucles vides (et optional)  
⇒ convergence de phylmdiso et phylmd
- définir des indices : iso\_eau, iso\_HDO, etc...
- tableaux d'indices : boucles sur des sous-ensembles d'isotopes
- plus adapté au HPC : parallélisme, GPU (pas d'aliasing)