Multi-tiling in ORCHIDEE (i.e. energy/water budgets at sub-grid scale) as a surrogate for high resolution !

(Julien Alleon, Philippe P., Catherine O., Josefine G., Frederique C., ….)

Few remarks / notations

- **Notion of PFT will become Surface Functional Type (SFT)**
  to group under the same terminology plants, lakes, glaciers, urban, etc…
  (Metaclass remains to help defining new SFT)

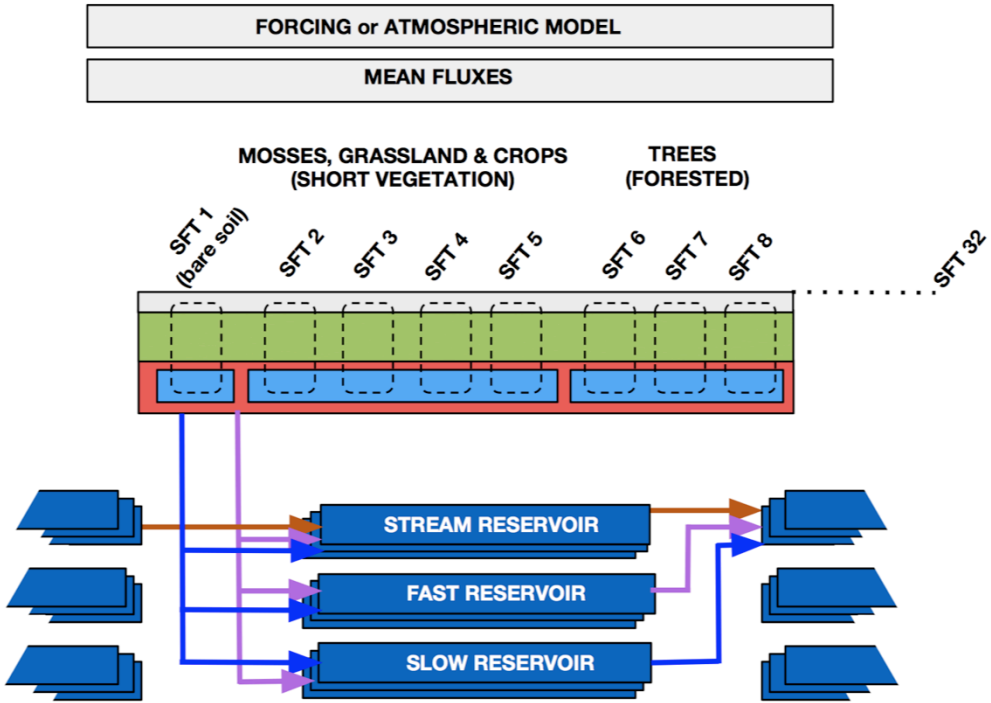
- **Key Parameters / Variables :**
  - **Nener** : number of energy budgets (excluding the snow specific budget) for each grid box;
  - **Nhydro** : number of water budgets for each grid box;
  - **Natm** : number of atmospheric columns (to be taken as 1 in a first approach)
  - **Nsft** : Number of SFT for each grid box
  - **⇒ Nener, Nhydro, Natm, Nsft could/should be variables of Npts (number of grid box)**
  - **SFT_type** : a variable that gives the types of SFT (Tree, grass, crop, lake, glacier, ….)

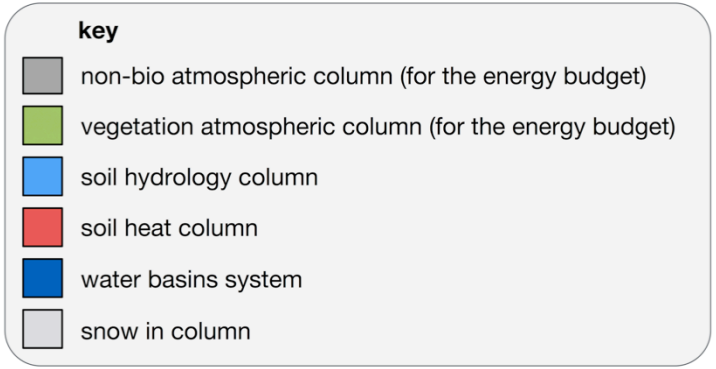
- **Tiling_flag** : parameter to define the options for the tiles (see later)

Current scheme !

N_energy = 1 ; N_hydro = 3 (bare soil, short veg, trees)

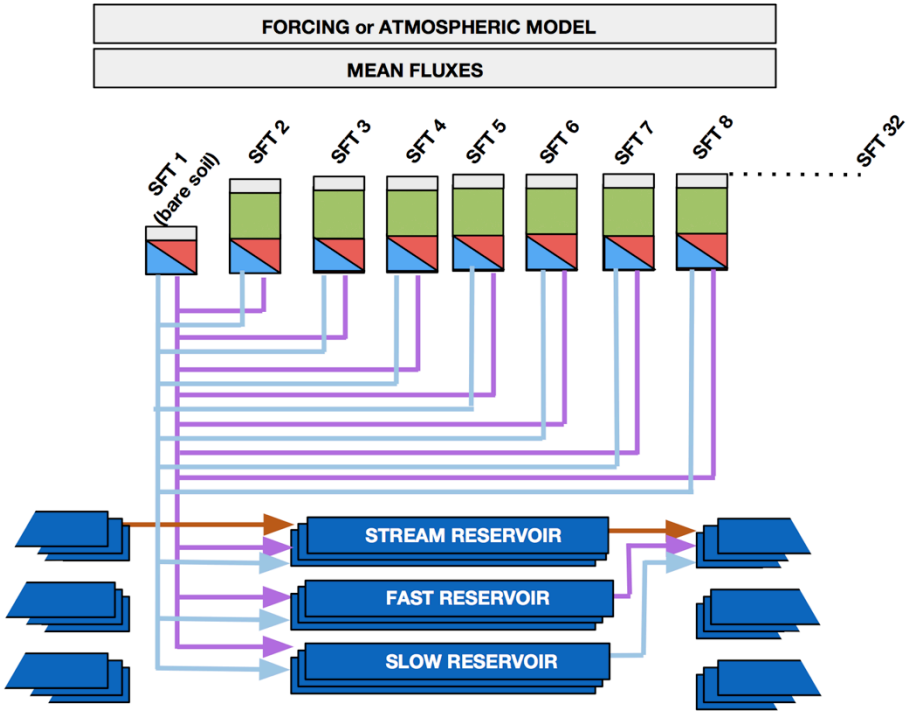
1 atmospheric column (mixing fluxes at first level) ; 1 routing scheme !

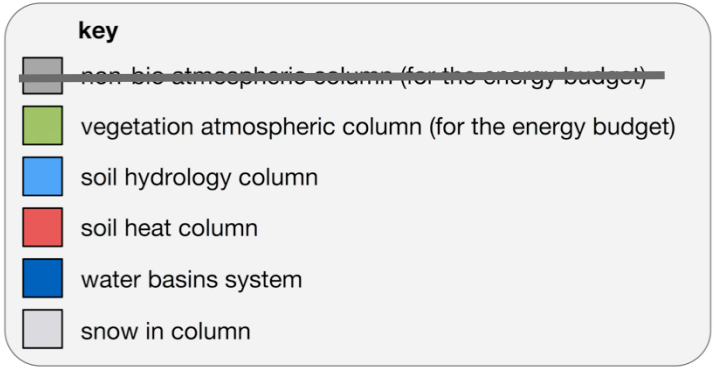




New multi-tiling approach : maximum split

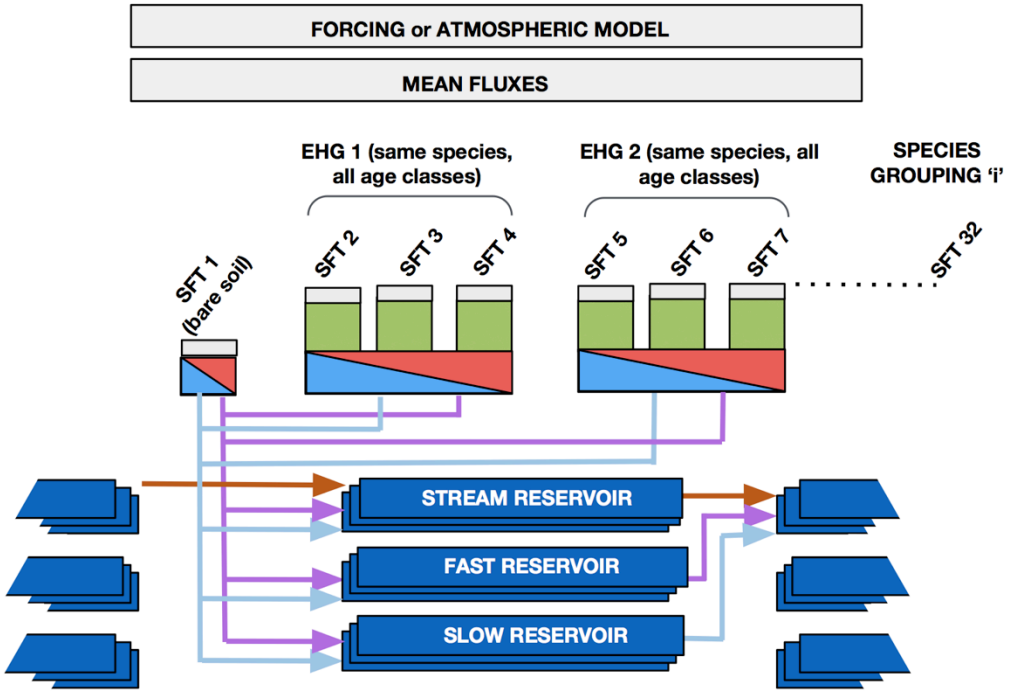
- Nhydro = Nener for each grid box ; But different across grid cells !
- Note: keep 2 parameters to reproduce current config with Nhydro = 3 / Nener =1

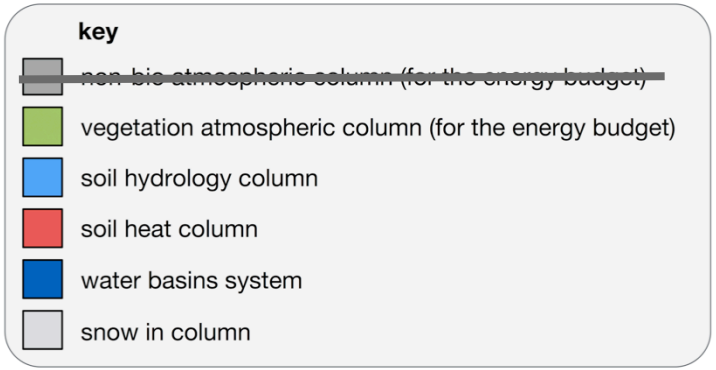




New multi-tiling approach : « intermediate » split

- Define a set of intermediate grouping with different options
- Variable grouping per grid cell





How to define the number : Nener and Nhydro ?

- Objective: describe for each pixel the grouping of SFTs that will share the same Energy and water budgets !

- **Tiling_flag** :
  - -1: To reproduce current implementation (should we keep it ?)
  - 0: to read an input map of tiling_flag per grid cell
  - 1: One E/W budget for each SFT_type (Trees, Grass, Crops, Bare Soil, lake, glaciers, cities...)
  - 2: One E/W budget for each SFT (i.e. all tree types, grass types, etc)
  - 3: One E/W budget for all the same Tree-pft (instead of 1 budget for each age class)
  - 4: ….
  - 5: … Any “eco-hydrological” grouping like low lands versus high lands OR Hydrological Transfert Unit (HTU) or ….
  - 6:...

- Special Tiling_flag = 0 to read a map that contains for each grid cell the values

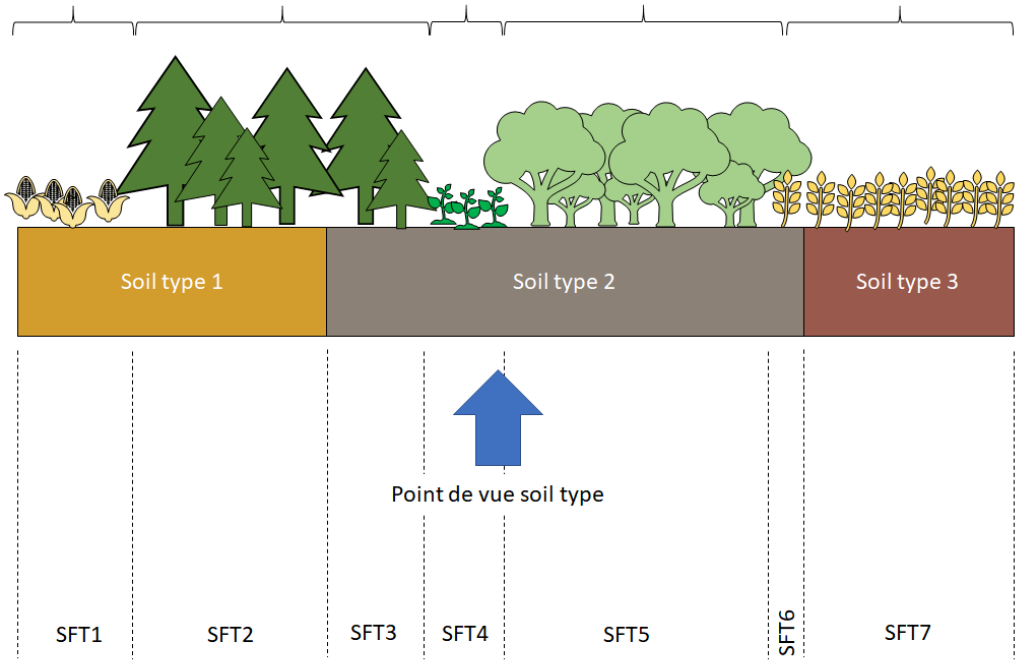
- Note: Keeping option “-1” (current set up) is not easy as Nener =/ Nhydro and having NOBIO complicates the code !

How to define the SFTs ?

- Nb of Metaclasses should increase with new types (lake, glacier, cities, ..)
- Define SFT similarly to the PFT maps (from ESA-LC, LUH,...)
- BUT with potentially more complex options
  - Split SFT depending on the soil texture ?
  - Split SFT depending on low vs high lands ?
  - Split SFT depending on eco-hydrological units ?

- SFTs are updated each year (like current PFTs)
- Get rid of “nobio” ⇒ becomes “glacier” SFT
- Bare soil becomes a unique SFT; separated from water bodies and cities

Potential SFT: combining “Soil texture” & Vegetation



Eco-hydrological units

⇒ **To be defined beforehand with high resolution land cover & soil type maps!**

Implicit coupling with the atmosphere (LMDZ)

- Some biblio on implicit coupling
  - A new page is created under Wiki/Documentation: implicit coupling with LMDZ
    https://forge.ipsl.jussieu.fr/orchidee/wiki/Documentation
  - It includes several notes / articles on the coupling with LMDZ

- Feasibility with respect to the general implicit coupling with atm PBL
  energy budget surface fluxes

$$C_0 \frac{\delta T_s}{\delta t} = F - k_g(T_s - T_g)| \quad \text{on} \quad F = F_{rad} + F_1^H + LF_1^q \quad \begin{cases} X_l = C_l^X + D_l^X X_{l-1} & (2 \le l \le n) \\ X_1 = A^X + B^X F_s^X \delta t & (l = 1) \end{cases}$$

⇒ **Assumption:** All surface fluxes Fs_i (each tile) are well mixed in the lowest atmospheric boundary layer !

⇒ Flux(Fs) becomes a sum of Fs_i (each tile)

⇒ Equations to be set up properly but in principle no conceptual issues.

WHO / WHEN / WHERE ?

- To be carried primarily by Julien Alleon (with technical supervision by Josefine) (Potential additional contribution by Aude’s CDD)
- To start after the summer break of 2022
- Targetting a first implementation in the Trunk
- Proceed step by step with evaluation of each step !
- FUTURE : improve the assumption of flux mixing in the first layer !!

Additional remarks / questions ?

Complementary suggestions / modifications:

- Rewording ß1, ß2, … => $ß_{snow}$, B_{inter}, B_{transp}, ….
- Separate ßx in two terms : Fraction of grid cell (“Frac_x”)
  & Resistance (“Res_x”)
- Remove ß23: Allow Transpiration to occur at the same time as “intercep loss”
- Others ??

Questions:

- Should we keep back compatibility with “Nener=1 and Nhydro=3” ?
- How to define mixed ecosystems? (See Slide 16)
- …

How to manage “sparse arrays” ?

Typical arrays are: X [Npts, Nsft_max, Ncircclass, (Ntracer)]
⇒ Large Nsft_max not needed for most pixels !! Ncirc only for Tree SFTs

Possible way forward:

- X ⇒ X [Nelements]
- Use two functions to relate i_element ⇐> (i_pts, i_sft, i_circ, i_trac)
  - ielement = F1 (i_pts, i_sft, i_circ, i_trac)
  - [i_pts,i_sft,i_circ,i_trac] = F2 (i_element)
- Careful implementation for the parallelisation (possible according to Yann M.)
- Maybe new feature of F99 can help ?
- ALL computations in the code would be made with X(Nelements) BUT the output could be transformed into X(Npts, Nsft_max, Ncirclass, (Ntracer))