UN POINT SUR L'EXTERNALISATION DES PARAMETRES

Auteur initial : Didier Solyga

Diffusion : Groupe Projet ORCHIDEE

Version 0 : 29/10/2010

Objectifs :

Ce document a pour but de récapituler les choix techniques pris et de résumer le travail accompli dans le cadre de l'externalisation des paramètres d'ORCHIDEE. Il fera le point sur les discussions en suspens. Sa lecture s'appuie sur le document « Dossier technique : externalisation des paramètres pour Orchidee » (disponible à l'adresse suivante : http://forge.ipsl.jussieu.fr/orchidee/wiki/ExternalisationParameters)

PLAN :

I. Notifications techniques

1. Rappels des décisions prises lors des réunions techniques:

- La version externalisée se base sur la version AR5 d'ORCHIDEE.
- Notion de métaclasse (MTC) : chaque PFT est associé à une MTC. Seul 13 MTCs sont prévues pour l'instant, correspondants aux 13 PFTs originaux. Les paramètres-MTCs sont fixés en dur dans le code. La table de correspondance est appelé PFT_TO_MTC qui est déclarée ligne par ligne dans orchidee.def. Le nombre de MTC est défini par le nouveau paramètre 'nvmc', celui des PFTs est toujours paramétré par 'nvm'. Le but final est de pouvoir définir un nombre quelconque de PFTs.
- Mise en place du FLAG IMPOS_PARAM permettant d'utiliser les valeurs par défauts sans avoir à commenter les valeurs des paramètres imposés.
- Tous les paramètres, qu'ils soient sur les PFTs ou non, seront rassemblés dans un premier temps dans le dossier src_parameters.
- Dans un deuxième temps, certains paramètres seront spatialisés : leurs valeurs dépendront des points de terre.

- Les valeurs standards des paramètres-MTC ont l'attribut PARAMETER pour protéger ces valeurs.
- La bibliothèque parallel est compilée avant la bibliothèque parameter pour utiliser getin_p au lieu de getin. (avec l'accord de Martial Mancip)
- Les paramètres utilisés avec un setvar ne sont pas externalisés et les paramètres externalisés ne sont pas restartés.

PS : Toutes les choix techniques listés par la suite ont été effectués avec l'accord de Nicolas Vuichard et/ou Nicolas Viovy.

2. Sur les MTCs :

- Remarque sur la routine getin : getin lit les tableaux 2d mais réindexe les éléments (ie : un tableau (nvm,x) sera considéré en sortie (ie dans les fichiers used_run_def) comme un tableau nvm*x) => difficulté à contrôler les valeurs et problème pour imposer les valeurs (pour modifier la valeur maint_resp_slope(pft7,3), il faut entrer maint_resp_slope_03 et déclarer les nvm valeurs).
- Par conséquent, les paramètres de dimensions (nvm,x) sont initialisés composante par composante lié à la routine getin.
- Reconstruction des paramètres(nvm,x) utilisés par la suite dans le code dans define_pft.
- Exemple : le paramètre maint_resp_slope est de taille (nvm,3). On définit au préalable 3 tableaux intermédiaires de taille (nvm) : maint_resp_slope1, maint_resp_slope2 et maint_resp_slope3 correspondants à chacune des composantes de main_resp_slope. Ces trois tableaux sont initialisés avec les valeurs standard ou modifiés. A la fin de ces initialisations préliminaires, on initialise maint_resp_slope qui sera utilisé dans le code par la suite.
- Sol nu : on impose que le premier pft soit le sol nu (ie pft_to_mtc(1) = 1) et que deux pfts ne peuvent être associés à la MTC sol nu => résout les problèmes sur les boucles de 2 à nvm

3. Sur stomate :

- Suppression des structures t_photo_type, tphoto, pheno_type et pheno_crit.
- Les valeurs du paramètre sla sont déclarés et non plus calculés => suppression de leaflife_tab
- Les valeurs paramètres suivants : tree, lai_initmin, migrate, bm_sapl, maxdia, cn_sapl, alpha, leaf_timecst dépendent des valeurs d'autres paramètres et ne sont donc pas externalisables.
- Modification d'un test dans stomate_lpj pour la prise en compte d'un nombre de PFTs variable.
- Les 3 points précédents ont été réalisés avec l'accord de Nicolas Viovy et Nicolas Vuichard

4. Sur sechiba :

- Réécriture de certaines boucles de sechiba afin qu 'elles prennent en compte un nombre variable de PFTs (slowproc et sechiba) .
- Externalisation de pref_soil_veg et déclaration des paramètres-pfts déjà externalisés dans define_pft.

II. Mise en oeuvre pratique : travail et choix effectués

Presque tous les choix techniques cités ci-dessous ont été faits en collaboration avec Nicolas Vuichard et/ou Nicolas Viovy.

1. Global

- Recherche et regroupement des paramètres-pft externalisables dans define_pft.f90. (annexe A)
- Recherche des coefficients d'équations pouvant être externalisés. (annexe B)
- Extension de la recherche aux paramètres scalaires : création d'une liste provisoire (annexe B)
-

2. Src_parameters

- Déclaration des PFTs dans le dossier src_parameters : ce dernier contiendra 3 fichiers sources constantes.f90, constantes_mtc.f90 et define_pft.f90, issus de la fusion de constantes_veg, constantes_soil et constantes_co2.
- constantes.f90 contient les paramètres scalaires et les dimensions et indices des tableaux; constantes_mtc.f90 contient les valeurs standards des paramètres-pfts et define_pft.f90 fait la liaison avec le reste du code.
- Création des sources define_pft.f90 et constantes_mtc.f90.
- constantes_mtc.f90 contient toutes les valeurs standard des paramètres-mtcs. define_pft contient tous les paramètres-pfts. On a une sorte de symétrie entre ces deux fichiers.
- define_pft.f90 contient 4 subroutines : pft_main, pft_init, pft_alloc et pft_clear. pft_main appelle pft_init et pft_alloc. pft_main est la routine principale . L'initialisation des paramètres-pfts se fait comme inidiqué dans le dossier technique :

```
DO j=1,nvm
        vcmax_opt(j) = vcmax_opt_mtc(pft_to_mtc(j))
ENDDO
```

- Regroupement des getin afin de faciliter l'emploi du flag impos_param.
- Prévention des erreurs avec 3 tests : existence de la métaclasse correspondance (pft_to_mtc(j) > nvmc), 2 tests pour le sol nu (cf I.2) . Test d'allocation mémoire.
- Suppression du module constantes_co2 => modification des Makefile et AA_make
- constantes_mtc contient tous les paramètres de taille nvm (environ 90 paramètres cf liste donnée en annexe)
- Classification des paramètres (cf annexes A à D)

3. Src_stomate

- Regroupement des paramètres-pfts de stomate_constants et stomate_data dans define_pft.
- Remplacement de tous les appels aux structures pheno_crit et t_photo .

4. Src_sechiba

- Suppression des variables doublons : ext_coef et ext_coeff, throughfall_by_pft dans hydrol et hydrolc. On garde ext_coeff et throughfall_by_pft est déclaré dans define_pft.
- La déclaration et l'initialisation des paramètres-pfts s'effectue au premier appel de sechiba .
- Modification des boucles distinguant les PFTs agricoles des autres PFTs (ie : elles indiquaient explicitement les PFTs 12 et 13, par exemple veget_max dans slowproc.f90)

III. A discuter

- Les choix techniques énoncés dans ce document
- Classification des paramètres-pfts et choix des paramètres scalaires externalisables
- Externalisation des paramètres sol et litière
- Déplacement des routines de constantes_veg.f90
- Organisation du orchidee.def : découpage, affichage de toutes les options
- Génération de la carte de végétation pour un nombre de PFTs quelconque
- Spatailisation : choix des paramètres 2D
- Sortie d'une première version sans la spatialisation avant la fin de l'année 2010

ANNEXE A : liste et classification des paramètres pfts

!-------------------------
! PFT global
!------------------------
!-
! *Table of conversion : we associate one pft to one mtc* :: pft_to_mtc
!-
! *Description of the PFT* :: PFT_name

!---------------------- ------
! Vegetation structure
!---------------------- ------
!-
! 1 .Sechiba
!-
! *Value for veget_ori for tests in 0-dim simulations* :: veget_ori_fixed_test_1
!-
! *laimax for maximum lai see also type of lai interpolation* :: llaimax
!-
! *laimin for minimum lai see also type of lai interpolation* :: llaimin
!-
! *prescribed height of vegetation.*
! *Value for height_presc : one for each vegetation type* :: height_presc
!-
! *Type of behaviour of the LAI evolution algorithm*
! *for each vegetation type.*
! *Value of type_of_lai, one for each vegetation type : mean or interp* :: type_of_lai
!-
! *Is the vegetation type a tree ?* :: is_tree
!-
! 2 .Stomate
!-
! *leaf type*
! *1=broad leaved tree, 2=needle leaved tree, 3=grass 4=bared ground* :: leaf_tab
!-
! *specif leaf area (m**2/gC)* :: sla
!-
! *natural?* :: natural

!---------------- ------
! Photosynthesis
!---------------- -----
!-
! 1 .CO2
!-
! *flag for C4 vegetation types* :: is_c4
!-
! *Slope of the gs/A relation (Ball & al.)* :: gsslope
!-
! *intercept of the gs/A relation (Ball & al.)* :: gsoffset
!-
! *values used for vcmax when STOMATE is not activated* :: vcmax_fix
!-

! *values used for vjmax when STOMATE is not activated* :: vjmax_fix
!-
! *values used for photosynthesis tmin when STOMATE is not activated* :: co2_tmin_fix
!-
! *values used for photosynthesis topt when STOMATE is not activated* :: co2_topt_fix
!-
! *values used for photosynthesis tmax when STOMATE is not activated* :: co2_tmax_fix
!-
! **2 .Stomate**
!-
! *extinction coefficient of the Monsi&Seaki relationship (1953)*:: ext_coeff ! = ext_coef in sechiba
!-
! *Maximum rate of carboxylation* :: vcmax_opt
!-
! *Maximum rate of RUbp regeneration* :: vjmax_opt
!-
! *minimum photosynthesis temperature,*
! *constant a of ax^2+bx+c (deg C),tabulated* :: tphoto_min_a
! *constant b of ax^2+bx+c (deg C),tabulated* :: tphoto_min_b
! *constant c of ax^2+bx+c (deg C),tabulated* :: tphoto_min_c
!-
! *optimum photosynthesis temperature,*
! *constant a of ax^2+bx+c (deg C),tabulated* :: tphoto_opt_a
! *constant b of ax^2+bx+c (deg C),tabulated* :: tphoto_opt_b
! *constant c of ax^2+bx+c (deg C),tabulated* :: tphoto_opt_c
!-
! *maximum photosynthesis temperature,*
! *constant a of ax^2+bx+c (deg C), tabulated* :: tphoto_max_a
! *constant b of ax^2+bx+c (deg C), tabulated* :: tphoto_max_b
! *constant c of ax^2+bx+c (deg C), tabulated* :: tphoto_max_c

!----------------------- --------
! **Respiration - stomate**
!----------------------- --------
!
! *slope of maintenance respiration coefficient (1/K),*
! *constant c of aT^2+bT+c , tabulated* :: maint_resp_slope1
! *constant b of aT^2+bT+c , tabulated* :: maint_resp_slope2
! *constant b of aT^2+bT+c , tabulated* :: maint_resp_slope3
!-
! *maintenance respiration coefficient (g/g/day) at 0 deg C,*
! *for leaves, tabulated* :: cm_zero_leaf
! for *sapwood above, tabulated* :: cm_zero_sapabove
! for *sapwood below, tabulated* :: cm_zero_sapbelow
! *for heartwood above, tabulated* :: cm_zero_heartabove
! *for heartwood below, tabulated* :: cm_zero_heartbelow
! *for roots, tabulated* :: cm_zero_root
! *for fruits, tabulated* :: cm_zero_fruit
! for carbohydrate reserve, tabulated ::cm_zero_carbres

!---------------- ---
! **Fire - stomate**
!---------------- ----
!-
! *flamability: critical fraction of water holding capacity* :: flam
!-
! *fire resistance* :: resist

!--------------- ---
! **Flux - LUC**
!--------------- ---
!
! *Coeff of biomass export for the year* :: coeff_lcchange_1
!-
! *Coeff of biomass export for the decade* :: coeff_lcchange_10
!-
! *Coeff of biomass export for the century* :: coeff_lcchange_100

!----------- -----
! **Phenology**
!----------- -----
!-
! **1 .Stomate**
!-
!
! *maximum LAI, PFT-specific* :: lai_max
!-
! *which phenology model is used? (tabulated)* :: pheno_model
!-
! *type of phenology*
! *0=bared ground 1=evergreen, 2=summergreen, 3=raingreen, 4=perennial*
! *Pour l'instant, le phénotype de sol nu n'est pas géré aussi on traitera les sols* :: pheno_type
!-
! **2. Leaf Onset**
!-
! *critical gdd,tabulated (C)*
! *constant c of aT^2+bT+c* :: gdd_crit1
! *constant b of aT^2+bT+c* :: gdd_crit2
! *constant a of aT^2+bT+c* :: gdd_crit3
!-
! *critical ngd,tabulated. Threshold -5 degrees* :: ngd_crit
!-
! *critical temperature for the ncd vs. gdd function in phenology* :: ncdgdd_temp
!-
! *critical humidity (relative to min/max) for phenology* :: hum_frac
!-
! *minimum duration of dormance (d) for phenology* :: lowgpp_time
!-
! *minimum time elapsed since moisture minimum (d)* :: hum_min_time
!-
! *sapwood -> heartwood conversion time (d)* :: tau_sap
!-
! *fruit lifetime (d)* :: tau_fruit
!-
! *fraction of primary leaf and root allocation put into reserve* :: ecureuil
!-
! *NEW - allocation above/below = f(age)* :: alloc_min
! :: alloc_max
! :: demi_alloc
!-
! **3. Senescence**
!-
! length of death of leaves,tabulated (d):: leaffall
!-
! critical leaf age,tabulated (d):: leafagecrit

!-
! type of senescence,tabulated
! List of avaible types of senescence :
! 'cold ', 'dry ', 'mixed ', 'none ':: senescence_type
!-
! critical relative moisture availability for senescence :: senescence_hum
!-
! relative moisture availability above which
! there is no humidity-related senescence :: nosenescence_hum
!-
! maximum turnover time for grasse:: max_turnover_time
!-
! minimum turnover time for grasse:: min_turnover_time
!-
! minimum leaf age to allow senescence g:: min_leaf_age_for_senescence
!-
! critical temperature for senescence (C),
! constant c of aT^2+bT+c , tabulated :: senescence_temp1
! constant b of aT^2+bT+c , tabulated :: senescence_temp2
! constant a of aT^2+bT+c , tabulated :: senescence_temp3

!-----------
! DGVM
!-----------
!-
! *residence time (y) of trees* :: residence_time
!-
! *critical tmin, tabulated (C)* :: tmin_crit
!-
! *critical tcm, tabulated (C)* :: tcm_crit

!------------------------------ ----------
! Evapotranspiration - sechiba
!------------------------------ ----------
!-
! *Structural resistance.*
! *Value for rstruct_const : one for each vegetation type* :: rstruct_const
!-
! *A vegetation dependent constant used in the calculation*
! *of the surface resistance.*
! *Value for kzero one for each vegetation type* :: kzero

!------------------ --
! Water - sechiba
!------------------ ---
!-
! *Maximum field capacity for each of the vegetations* (Temporary).
! *Value of wmax_veg : max quantity of water :*
! *one for each vegetation type en Kg/M3* :: wmax_veg
!-
! *Root profile description for the different vegetation types.*
! *These are the factor in the exponential which gets*
! *the root density as a function of depth* :: humcste

!------------------ -----
! Albedo - sechiba
!------------------ -----
!-
! *Initial snow albedo value for each vegetation type*
! *as it will be used in condveg_snow*
! *Values are from the Thesis of S. Chalita (1992)* :: snowa_ini
!-
! *Decay rate of snow albedo value for each vegetation type*
! *as it will be used in condveg_snow*
! *Values are from the Thesis of S. Chalita (1992)* :: snowa_dec
!-
! *leaf albedo of vegetation type, visible albedo* :: alb_leaf_vis
!-
! *leaf albedo of vegetation type, near infrared albedo* :: alb_leaf_nir

!------------------------
! Soil - vegetation
!------------------------
!
! *Table which contains the correlation between the soil types*
! *and vegetation type. Two modes exist :*
! *1) pref_soil_veg = 0 then we have an equidistribution*
! *of vegetation on soil types*
! 2*) Else for each pft the prefered soil type is given :*
! *1=sand, 2=loan, 3=clay*
! *The variable is initialized in slowproc.* :: pref_soil_veg1
! :: pref_soil_veg2
! :: pref_soil_veg3
!

!------------------------------ -----------------------------
! Parameters already externalised (from sechiba)
! to classify
!--------------------------------- ---------------------------
!
! *used in hydrolc and hydrol* :: throughfall_by_pft
!-
! *used in diffuco !! Nathalie le 28 mars 2006* **-** *sur proposition de Fred Hourdin, ajout*
!! d'un potentiometre pour regler la resistance de la vegetation :: rveg_pft

PS : Le nombre important de paramètres s'explique par le découpage composante par composante de chaque tableau à deux dimensions. Ce découpage concerne les variables suivantes : pheno_gdd_crit, senescence_temp, maint_resp_slope, coeff_maint_zero et alb_leaf.

ANNEXE B : liste des paramètres scalaires externalisables

I. LISTE DES PARAMETRES SCALAIRES (fichier par fichier)

Comme le titre l'indique, cette liste est incomplète. Elle est le résulat de mes conversations avec Nathalie de Noblet pour la partie sechiba, Nicolas Viovy et Nicolas Vuichard pour la partie stomate. La classification est le fruit de ces discussions.

```
!***********!
!  LPJ files    !
!***********!
```

!====================
! lpj_constraints.f90
!====================

! *longest sustainable time without regeneration (vernalization)* :: too_long = 5.

!==============
! lpj_fire.f90
!==============

! *Time scale for memory of the fire index (days)* :: tau_fire = 30.

! *Critical litter quantity for fire* :: litter_crit = 200.

!==============
! lpj_light.f90 :
!==============

! *maximum total number of grass individuals in a closed canopy* :: grass_mercy = 0.01

! *minimum fraction of trees that survive even in a closed canopy* :: tree_mercy = 0.01

! *for diagnosis of fpc increase, compare today's fpc to last year's maximum (T) or*
! *to fpc of last time step (F)?* :: annual_increase = .TRUE.

!================
! lpj_pftinout.f90
!================

! minimum availability :: min_avail = 0.01

```
!******************!
! STOMATE_FILES    !
!******************!
```

!==================
! stomate_alloc.f90
!==================

! *Do we try to reach a minimum reservoir even if we are severely stressed?* :: ok_minres = .TRUE.

! *time (d) to attain the initial foliage using the carbohydrate reserve* :: tau_leafinit = 10.

! *maximum time (d) during which reserve is used (trees)* :: reserve_time_tree = 30.

! *maximum time (d) during which reserve is used (grasses)* :: reserve_time_grass = 20.

! *Standard root allocation* :: R0 = 0.3

! *Standard sapwood allocation* :: S0 = 0.3

! *Standard leaf allocation* :: L0 = 1. - R0 - S0

! *Standard fruit allocation* :: f_fruit = 0.1

! *fraction of sapwood allocation above ground* :: alloc_sap_above_tree = 0.5
: :: alloc_sap_above_grass = 1.0

! *extrema of leaf allocation fraction* :: min_LtoLSR = 0.2
:: max_LtoLSR = 0.5

! *scaling depth for nitrogen limitation (m)* :: z_nitrogen = 0.2

!=====================
! stomate_constants.f90
!=====================

!- ---------------------------------------
! **parameters for the pipe model**
!- ---------------------------------------
! *crown area = pipe_tune1. stem diameter**(1.6) (Reinicke's theory)* :: pipe_tune1 = 100.0

! *height=pipe_tune2 * diameter**pipe_tune3* :: pipe_tune2 = 40.0
:: pipe_tune3 = 0.5

! *needed for stem diameter* :: pipe_tune4 = 0.3

! *Density* :: pipe_density = 2.e5

! *one more parameter* :: pipe_k1 = 8.e3

! *Maximum tree establishment rate* :: estab_max_tree = 0.12

! *Maximum grass establishment rate* :: estab_max_grass = 0.12

! *initial density of individuals* :: ind_0 = 0.02

! *For trees, minimum fraction of crown area occupied*
! (due to its branches etc.)
! This means that only a small fraction of its crown area
! can be invaded by other trees. :: min_cover = 0.05

!- ---------------------------------
! climatic parameters
!- ---------------------------------
! *minimum precip, in mm/year* :: precip_crit = 100.

! *minimum gdd for establishment of saplings* :: gdd_crit = 150.

! *critical fpc, needed for light competition and establishment* :: fpc_crit = 0.95
!-
! *fraction of GPP which is lost as growth respiration* :: frac_growthresp = 0.28

!=================
! stomate_data.f90
!=================

tau_hum_month = 20.
tau_hum_week = 7.
tau_t2m_month = 20.
tau_t2m_week = 7.
tau_tsoil_month = 20.
tau_soilhum_month = 20.
tau_gpp_week = 7.
tau_gdd = 40.
tau_ngd = 50.
tau_longterm = 3. * one_year

!===================
! stomate_litter.f90 :
!===================

! *scaling depth for soil activity (m)* :: z_decomp = 0.2

!=================
! stomate_npp.f90
!=================

! *maximum fraction of allocatable biomass used for maintenance respiration* :: tax_max = 0.8

!=====================
! stomate_phenology.f90
!=====================

! *take carbon from atmosphere if carbohydrate reserve too small?* :: always_init = .FALSE.

! *minimum time (d) since last beginning of a growing season* :: min_growthinit_time = 300.

!* parametres presents dans pheno_hum, pheno_humgdd et pheno_moigdd
! *moisture availability above which moisture tendency doesn't matter* :: moiavail_always_tree = 1.0
:: moiavail_always_grass = 0.6

!* parametre present dans pheno_humgdd et pheno_moigdd
! *monthly temp. above which temp. tendency doesn't matter* :: t_always = ZeroCelsius + 10.

!========================
! stomate_prescribe.f90
!========================

! *generic tree crown area (m**2)* :: cn_tree = 4.

!=====================
! stomate_season.f90
!=====================

! *rapport maximal GPP/GGP_max pour dormance* :: gppfrac_dormance = 0.2

! *minimum gpp considered as not "lowgpp"* :: min_gpp_allowed = 0.3

! *tau (year) for "climatologic variables* :: tau_climatology = 20

! *parameters for herbivore activity* :: hvc1 = 0.019
:: hvc2 = 1.38
:: leaf_frac=.33

!==================
! stomate_vmax.f90
!=================

! *offset (minimum relative vcmax)* :: vmax_offset = 0.3

! *leaf age at which vmax attains vcmax_opt (in fraction of critical leaf age)*:: leafage_firstmax = 0.03

! *leaf age at which vmax falls below vcmax_opt (in fraction of critical leaf age)* :: leafage_lastmax = 0.5

! *leaf age at which vmax attains its minimum (in fraction of critical leaf age)* :: leafage_old = 1.

!****************!
! SECHIBA_FILES !
!****************!

!===============
! condveg.f90
!===============

! *to get z0 from height* :: z0_over_height = un/16.

! *Magic number which relates the height to the displacement height* :: height_displacement = 0.75

!=============
! diffuco.f90
!=============

nlai = 20

laimax = 12.

!=============
! hydrol.f90
!=============

! *time constant in days to return to free drainage after return flow* :: drain_rest_cste = 15.0

! *Allowed moisture above mcs (boundary conditions)* :: dmcs = 0.002

! *Allowed moisture below mcr (boundary conditions)* :: dmcr = 0.002

!***********************!
! constantes_soil.f90 !
!***********************!

!*Déjà externalisé*
! Total depth of soil reservoir (for hydrolc) :: dpu_cste=2.0_r_std

!*Scaling depth for litter humidity (m)* :: hcrit_litter=0.08_r_std

! *Total depth of soil reservoir (m)* :: dpu = &
 & (/ 2.0_r_std, 2.0_r_std, 2.0_r_std /)
! => *correspond à dpu_cste (Nathalie)*

!************************!
! constantes_veg.f90 !
!************************!

! *The maximum mass (kg/m^2) of a glacier* .:: maxmass_glacier = 3000.

! *Minimal fraction of mesh a vegetation type can occupy* :: min_vegfrac=0.001

! *Limit of air temperature for snow* :: tsnow=273.

! *Sets the amount above which only sublimation occures [Kg/m^2]* :: snowcri=1.5

! *Critical value for computation of snow albedo [Kg/m^2]* :: snowcri_alb=10.

! *Lower limit of snow amount* :: sneige=snowcri/1000._r_std

! *The minimum wind* :: min_wind = 0.1

! *bare soil roughness length (m)* :: z0_bare = 0.01

! i*ce roughness length (m)* :: z0_ice = 0.001

!**
! To discuss with the concerned people (advice of Nathalie)
!**

with Jan Polcher for all Sechiba modules and for constantes_soil.f90

! and with Matthieu Guimberteau for routing
! magulod@locean-ipsl.upmc.fr

!=============
! routing.f90
!=============

! *The maximum number of basins we wish to have per grid-box (truncation of the model)* :: nbasmax=5

! *The maximum number of bassins we can handle at any time during the generation of the maps*. :: nbvmax = 220

! *The time constants are in days* .:: fast_tcst = 3.0, slow_tcst = 25.0, stream_tcst = 0.24
:: evap_cst = 0.18, wdelay_cst = 0.7

! *Maximum evaporation rate from lakes 7.5 kg/m^2/d transformed into kg/m^2/sec* :: maxevap_lake = 7.5/86400.

! *Parameter for the Kassel irrigation parametrization linked to the crops* :: crop_coef = 1.5

diagbox = (/ 1147, 1148 /)

!=================
! slowproc.f90
!=================

clayfraction_default = 0.2

veget_update=0 *!! update frequency in years for landuse*

veget_year_orig=0 *!! first year for landuse*

sec_old = 0.

! Also with Patricia de Rosnay or Isabelle Gouttevin

!************************!
! constantes_soil.f90 !
!************************!

!- ----------------------------------
! **Dimensioning parameters**
!- ----------------------------------
! *Number of soil level* :: ngrnd=7

! *Number of diagnostic levels in the soil* :: nbdl=11

! *Number of levels in CWRR* :: nslm=11

! *Number of soil types* :: nstm=3

!- --
! **Constantes from the Choisnel hydrology**
!- --
! *Wilting point (Has a numerical role for the moment)* :: qwilt = 5.0

! *Total depth of soil reservoir (for hydrolc)*:: dpu_cste=2.0_r_std

! *The minimal size we allow for the upper reservoir (m)* :: min_resdis = 2.e-5

! *Diffusion constant for the slow regime* :: min_drain = 0.001

! *Diffusion constant for the fast regime* :: max_drain = 0.1

! *The exponential in the diffusion law* :: exp_drain = 1.5

! *Transforms leaf area index into size of interception reservoir* :: qsintcst = 0.1

! *Maximum quantity of water (Kg/M3)* :: mx_eau_eau = 150.
!-
! *Constant in the computation of resistance for bare soil evaporation* :: rsol_cste = 33.E3

! *Scaling depth for litter humidity (m)* :: hcrit_litter=0.08_r_std
!- ---
! **Parameters for soil type distribution**
!- ---
! *Default soil texture distribution in the following order :*
! *sand, loam and clay* :: soiltype_default = (/ 0.0, 1.0, 0.0 /)

!- --

! **Parameters specific for the CWRR hydrology.**
!- --
! *Van genuchten coefficient n* :: nvan = (/ 1.89_r_std, 1.56_r_std, 1.31_r_std /)

! *Van genuchten coefficient a (mm^{-1})* :: avan = (/ 0.0075_r_std, 0.0036_r_std, 0.0019_r_std /)

! *CWRR linearisation* :: imin = 1

! *number of interval for CWRR* :: nbint = 100

! *number of points for CWRR* :: imax = nbint+1

! *Residual soil water content* :: mcr = (/ 0.065_r_std, 0.078_r_std, 0.095_r_std /)

! *Saturated soil water content* :: mcs = (/ 0.41_r_std, 0.43_r_std, 0.41_r_std /)

! *Total depth of soil reservoir (m)* :: dpu = (/ 2.0_r_std, 2.0_r_std, 2.0_r_std /)

! dpu must be constant over the different soil types
! Hydraulic conductivity Saturation (mm/d) :: ks = (/ 1060.8_r_std, 249.6_r_std, 62.4_r_std /)

! *Soil moisture above which transpir is max* :: pcent = (/ 0.5_r_std, 0.5_r_std, 0.5_r_std /)

! *Max value of the permeability coeff at the bottom of the soil* ::
free_drain_max = (/ 1.0_r_std, 1.0_r_std, 1.0_r_std /)

! *Volumetric water content field capacity* :: mcf = (/ 0.32_r_std, 0.32_r_std, 0.32_r_std /)

! *Volumetric water content Wilting pt* :: mcw = (/ 0.10_r_std, 0.10_r_std, 0.10_r_std /)

! *Vol. wat. cont. above which albedo is cst* :: mc_awet = (/ 0.25_r_std, 0.25_r_std, 0.25_r_std /)

! *Vol. wat. cont. below which albedo is cst* :: mc_adry = (/ 0.1_r_std, 0.1_r_std, 0.1_r_std /)

! *Matrix potential at saturation (mm)* :: psis = (/ -300.0_r_std, -300.0_r_std, -300.0_r_std /)

! *Time weighting for discretisation* :: w_time = 1.0_r_std

! With Gerhard Krinner
!krinner@ujf-grenoble.fr

!************************!
! constantes_veg.f90 !
!************************!

! *Time constant of the albedo decay of snow* :: tcst_snowa = 5._r_std

! *Maximum period of snow aging* :: max_snow_age = 50._r_std

! *Transformation time constant for snow (m)* :: snow_trans = 0.3_r_std

II. CLASSIFICATION PARTIELLE :

Voici une tentative de classification avec Nicolas Viovy et Nicolas Vuichard . On suit les rubriques indiquées dans l'annexe A :

!---------------------- ------
! **Vegetation structure**
!---------------------- ------
!-
! **1 .Sechiba**

! Minimal fraction of mesh a vegetation type can occupy :: min_vegfrac=0.001

!-
! **2 .Stomate**
!-
! *generic tree crown area (m**2)* :: cn_tree = 4.

!---------------- ------
! **Photosynthesis**
!--------------- -----
!-
! **1 .CO2**
!-
nlai = 20
laimax = 12.

!-
! **2 .Stomate**
!-
! *offset (minimum relative vcmax)* :: vmax_offset = 0.3

! *leaf age at which vmax attains vcmax_opt (in fraction of critical leaf age)* :: leafage_firstmax = 0.03

! *leaf age at which vmax falls below vcmax_opt (in fraction of critical leaf age)* :: leafage_lastmax = 0.5

! leaf age at which vmax attains its minimum (in fraction of critical leaf age) :: leafage_old = 1.

!----------------------- --------
! **Respiration - stomate**
!---------------------- -----------

! maximum fraction of allocatable biomass used for maintenance respiration :: tax_max = 0.8

!---------------- ---
! **Fire - stomate**
!--------------- -------

! Time scale for memory of the fire index (days) :: tau_fire = 30.

! Critical litter quantity for fire :: litter_crit = 200.

!----------- -----

! **Phenology**
!----------- -----
!-
! **1 .Stomate**
!-
! take carbon from atmosphere if carbohydrate reserve too small? :: always_init = .FALSE.

! minimum time (d) since last beginning of a growing season :: min_growthinit_time = 300.

! moisture availability above which moisture tendency doesn't matter :: moiavail_always_tree = 1.0
:: moiavail_always_grass = 0.6

! monthly temp. above which temp. tendency doesn't matter :: t_always = ZeroCelsius + 10.

!-
! 3. Season
!-
! rapport maximal GPP/GGP_max pour dormance :: gppfrac_dormance = 0.2

! minimum gpp considered as not "lowgpp" :: min_gpp_allowed = 0.3

! tau (year) for "climatologic variables :: tau_climatology = 20

! parameters for herbivore activity :: hvc1 = 0.019
:: hvc2 = 1.38
:: leaf_frac=.33

!-----------
! **DGVM**
!-----------

! *longest sustainable time without regeneration (vernalization)* :: too_long = 5.

! maximum total number of grass individuals in a closed canopy :: grass_mercy = 0.01

! minimum fraction of trees that survive even in a closed canopy :: tree_mercy = 0.01

! for diagnosis of fpc increase, compare today's fpc to last year's maximum (T) or
! to fpc of last time step (F)? :: annual_increase = .TRUE.

! minimum availability :: min_avail = 0.01

!--
! ALLOCATION
!-

! Do we try to reach a minimum reservoir even if we are severely stressed? :: ok_minres = .TRUE.

! time (d) to attain the initial foliage using the carbohydrate reserve :: tau_leafinit = 10.

! maximum time (d) during which reserve is used (trees) :: reserve_time_tree = 30.

! maximum time (d) during which reserve is used (grasses) :: reserve_time_grass = 20.

! Standard root allocation :: R0 = 0.3

! Standard sapwood allocation :: S0 = 0.3

! Standard leaf allocation :: L0 = 1. - R0 - S0

! Standard fruit allocation :: f_fruit = 0.1

! fraction of sapwood allocation above ground :: alloc_sap_above_tree = 0.5
:: alloc_sap_above_grass = 1.0

! extrema of leaf allocation fraction :: min_LtoLSR = 0.2
:: max_LtoLSR = 0.5

! scaling depth for nitrogen limitation (m) :: z_nitrogen = 0.2

!-
! SOIL & LITTER
!-

! scaling depth for soil activity (m) :: z_decomp = 0.2

III. LISTE DES PARAMETRES D'EQUATONS EXTERNALISABLES

Il existe d'autres paramètres que l'on veut externaliser qui les coefficients (codés en dur)qui intervienent dans les équations. Une listen (non exhaustive à nouveau) a été établie. Elle indique uniquement les modules et les variables concernées. Travail effectué avec Nicolas Viovy et Nicolas Vuichard.

PS : Le paramètre 'sla' de stomate_data n'est plus concerné .

FICHIER	VARIABLE	LIGNES	REFERENCE
lpj_establish	factor	273 & 288	
lpj_fire	bcfrac	424 & 521	Kuhlbush, JGR et GBC9
	firefrac	598 à 600	
lpj_gap	availbility	148	
stomate_data	sla	99 & 104	Reich & al 1997
stomate_litter	litterfrac	180	
	moistfunc_result	667 & 668	
	tempfunc_result	690	
stomate_resp	resp_maint_part_ radia	215	
stomate_season **	herbivores	884 & 886 & 891	Mc Naughton
diffuco	air_relhum	1052 à 1054	
	x_1	1137 & 1277	
	x_2	1279	
	x_3	1281	
	kt	1140	
	rt	1141	
	vc	1144	
	Kc	1182	
	Ko	1184	
	CP	1188	
	coeff_dew_veg	1671	
hydrol *	throughfall_by_pft	1465	
hydrolc		1802	
slowproc *	humcste	758	
	veget_next	2588	

* paramètre de taille nvm
** paramètre scalaire

Liste de variables susceptibles d'être externalisées

IV. LISTE DES PARAMETRES REDONDANTS

Comme le titre l'indique, il s'agit des paramètres définis plusieurs fois dans le code.

R_Earth (= 6378000)

Nom_src	Nom_fichier	Ligne	Modules utilisés
src_global	grid.f90	L 27	
	interpol_help.f90	L 34	
src_sechiba	routing.f90	L 1567 L 4862	grid.f90
	slowproc.f90	L 1654 L 2799 L 3438	grid.f90 interpol_help.f90 stomate_constants.f90
src_stomate	stomate_constants.f90	L 197	
	stomate_io.f90	L 1719	stomate_constants.f90

Pi

Nom_src	Nom_fichier	Ligne	Modules utilisés
	(=3,14159265358979328)		
src_parameters	constantes.f90	L 33	
	(= 4*arctan(1))		
src_global	grid.f90	L 191	constantes.f90
	solar.f90	L 56	constantes.f90
src_sechiba	routing.f90	L 1619 L 4876	constantes.f90
	slowproc.f90	L 1677 L 2820 L 3459 L 4061	constantes.f90
src_stomate	stomate_io.f90	L 1737	

Min_sechiba

src_parameters	constantes.f90	L 42
src_sechiba	slowproc.f90	L 1654 L 2799 L 3438

Min_stomate

src_global	grid.f90	L 190
src_stomate	stomate_constants.f90	L 47

ANNEXE C : liste des tableaux externalisables

On liste les paramètres tableaux non-pft pouvant être externalisés.

```
!***********!
! LPJ files !
!***********!

!===============
! lpj_fire.f90
!===============

! What fraction of a burned plant compartment goes into the atmosphere
 :: co2frac  (dimension(nparts))
      co2frac(ileaf) = .95
      co2frac(isapabove) = .95
      co2frac(isapbelow) = 0.
      co2frac(iheartabove) = 0.3
      co2frac(iheartbelow) = 0.
      co2frac(iroot) = 0.
      co2frac(ifruit) = .95
      co2frac(icarbres) = 0.95

!====================
! stomate_litter.f90 :
!===================

! C/N ratio  :: CN (DIM(nparts))
CN(ileaf) = 40.0
CN(isapabove) = 40.0
CN(isapbelow) = 40.0
CN(iheartabove) = 40.0
CN(iheartbelow) = 40.0
CN(iroot) = 40.0
CN(ifruit) = 40.0
CN(icarbres) = 40.0


! Lignine/C ratio of the different plant parts :: LC   (DIM(nparts))
LC(ileaf) = 0.22
LC(isapabove) = 0.35
LC(isapbelow) = 0.35
LC(iheartabove) = 0.35
LC(iheartbelow) = 0.35
LC(iroot) = 0.22
LC(ifruit) = 0.22
LC(icarbres) = 0.22


! decomposition flux fraction that goes into soil (litter -> carbon, above and below)
!   rest goes into atmosphere    DIMENSION(nlitt,ncarb,nlevs)  :: frac_soil

    frac_soil(istructural,iactive,iabove) = .55
    frac_soil(istructural,iactive,ibelow) = .45
    frac_soil(istructural,islow,iabove) = .7
    frac_soil(istructural,islow,ibelow) = .7
```

```
    frac_soil(imetabolic,iactive,iabove) = .45
    frac_soil(imetabolic,iactive,ibelow) = .45

                                             !****************!
                                             ! SECHIBA_FILES  !
                                             !****************!

!===============
! condveg.f90
!===============

! The correspondance table for the soil color numbers and their albedo :: classnb = 9
 vis_dry = (/0.24, 0.22, 0.20, 0.18, 0.16, 0.14, 0.12, 0.10, 0.27/)
 nir_dry = (/0.48, 0.44, 0.40, 0.36, 0.32, 0.28, 0.24, 0.20, 0.55/)
 vis_wet = (/0.12, 0.11, 0.10, 0.09, 0.08, 0.07, 0.06, 0.05, 0.15/)
 nir_wet = (/0.24, 0.22, 0.20, 0.18, 0.16, 0.14, 0.12, 0.10, 0.31/)
 albsoil_vis = (/0.18, 0.16, 0.16, 0.15, 0.12, 0.105, 0.09, 0.075, 0.25/)
 albsoil_nir = (/0.36, 0.34, 0.34, 0.33, 0.30, 0.25, 0.20, 0.15, 0.45/)
```

ANNEXE D : liste des paramètres fixes

On essaie de lister tous les paramètres qui ne devraient pas évoluer dans la version externalisée. Cette liste est issue des paramètres apparaissant dans les divers fichiers source constantes_ dans src_parameters et des constantes symboliques apparaissant dans le reste du code (ie: les variables ayant l'attribut PARAMETER dans le code). Le classement y est à nouveau purement formel et n'engage que ma responsabilité. Il y a des 'TO DELETE' qui apparaissent : ce sont des paramètres qui sont définis et qui ne sont pas utilisés dans la version AR5.

!-------- -------
! **INDICES** : list of all parameters which are index tables
!-------- ------

!================== ==============
! constantes_co2.f90 (now in define_pft.f90)
!================== ===============

!-
! indices for assimilation parameters
!-

INTEGER(i_std),PARAMETER :: itmin = 1
INTEGER(i_std),PARAMETER :: itopt = 2
INTEGER(i_std),PARAMETER :: itmax = 3
INTEGER(i_std),PARAMETER :: ivcmax = 4
INTEGER(i_std),PARAMETER :: ivjmax = 5
INTEGER(i_std),PARAMETER :: npco2 = 5

!=====================
! stomate_constants.f90
!=====================

! trees and litter: indices for the parts of heart- and sapwood above
! and below the ground
iabove = 1
ibelow = 2
nlevs = 2
!-
! litter: indices for metabolic and structural part
imetabolic = 1
istructural = 2
nlitt = 2
!-
! carbon pools: indices
iactive = 1
islow = 2
ipassive = 3
ncarb = 3

!-
! *leaf age discretisation (1 = no discretisation):*: nleafages = 4

! TO DELETE ?
! transformation between types of surface
INTEGER(i_std),PARAMETER :: ito_natagri = 1
INTEGER(i_std),PARAMETER :: ito_total = 2

!==================
! stomate_data.f90 (now in define_pft)
!==================

! litter fractions: indices :: ileaf = 1
:: isapabove = 2
:: isapbelow = 3
:: iheartabove = 4
:: iheartbelow = 5
:: iroot = 6
:: ifruit = 7
:: icarbres = 8
:: nparts = 8

!==============
! condveg.f90
!=============

ivis = 1
inir = 2

!----------------- ------
! INDEX for files
!----------------- -----

!===================
! constantes_co2.f90 (now in constantes.f90)
!===================
INTEGER(i_std),SAVE :: forcing_id

!-=================
! stomate_constants.f90
!-=================

! 0 = no, 4 = full online diagnostics
INTEGER(i_std),SAVE :: bavard=1

! time step :: itime

! STOMATE history file ID :: hist_id_stomate

! STOMATE history file ID for IPCC output :: hist_id_stomate_IPCC

! STOMATE restart file ID :: rest_id_stomate

!--------------------- ----------------------
! **DIMENSION_PARAMETERS** : list of all parameters which are index tables
!--------------------- ----------------------

!===================
! constantes_soil.f90
!===================
!-
! **Dimensioning parameters**
!-
! *Number of soil level* :: ngrnd=7

! *Number of diagnostic levels in the soil* :: nbdl=11

! *Number of levels in CWRR* :: nslm=11

! *Number of soil types* :: nstm=3

!=====================
! constantes_veg.f90
!====================

! *Number of other surface types: land ice (lakes,cities, ...)* :: nnobio=1

! I*ndex for land ice (see nnobio)* :: iice = 1

!=================
! stomate_io.f90
!================
nbvmax = 200

!==============
! stomate.f90
!==============

ndm = 10 (defined in 3 routines)

!===============
! intersurf.f90
!===============

max_hist_level = 11 (global)
llm = 1 (local)

!=============
! routing.f90
!=============

pickmax = 200

!==============
! slowproc.f90
!==============

classnb = 7 (local)

!----------------------------- ----------------
! CODE OPTION PARAMETERS
!------------------------------ -----------------

!================
! constantes.f90
!================

! *Specific value if no restart value* :: val_exp = 999999.
!-
! *Epsilon to detect a near zero floating point* :: min_sechiba = 1.E-8_r_std

! *The undef value used in SECHIBA* :: undef_sechiba = 1.E+20_r_std

!-
! *Numerical constant set to 0* :: zero = 0._r_std

! *Numerical constant set to 1/2* :: undemi = 0.5_r_std

! *Numerical constant set to 1* :: un = 1._r_std

! *Numerical constant set to -1* :: moins_un = -1._r_std

! *Numerical constant set to 2* :: deux = 2._r_std

! *Numerical constant set to 3* :: trois = 3._r_std

! *Numerical constant set to 4* :: quatre = 4._r_std

! *Numerical constant set to 5* :: cinq = 5._r_std

! *Numerical constant set to 6* :: six = 6._r_std

! *Numerical constant set to 8* :: huit = 8._r_std

! *Numerical constant set to 1000* :: mille = 1000._r_std

```
 TYPE control_type
   LOGICAL :: river_routing
   LOGICAL :: hydrol_cwrr
   LOGICAL :: ok_sechiba
   LOGICAL :: ok_co2
   LOGICAL :: ok_stomate
   LOGICAL :: ok_dgvm
   LOGICAL :: stomate_watchout
   LOGICAL :: ok_pheno
 END TYPE control_type
```

!====================
! constantes_co2.f90 (now in constantes.f90)
!===================

! NV080800 Name of STOMATE forcing file :: stomate_forcing_name='NONE'

! NV080800 Name of soil forcing file :: stomate_Cforcing_name='NONE'

!====================
! constantes_veg.f90
!===================

! *Constant in the computation of surface resistance* :: defc_plus=23.E-3

! *Constant in the computation of surface resistance* :: defc_mult=1.5

!=======================
! stomate_constants.f90
!======================

! write forcing file for carbon spinup? :: write_carbonforce

! Epsilon to detect a near zero floating point :: min_stomate = 1.E-8_r_std

! some large value :: large_value = 1.E33_r_std

! Special value :: undef = -9999.

!==============
! stomate.f90
!=============

r_typ =nf90_real4 (précision donnée)

!=============
! enerbil.f90
!============

missing = 999998

!============
! hydrol.f90
!============

allowed_err = 1.0E-8_r_std

!=============
! hydrolc.f90
!=============

dsg_min = 0.001
EPS1 = EPSILON(un)
nitermax = 100 (limite the number of loops)

!==============
! routing.f90
!==============

undef_int = 999999999

! *Just to make sure we do not get too large numbers* :: scaling = 1.0E+6
:: allowed_err = 50.

!==============
! watchout.f90
!==============

kind_r_watch=nf90_real8

!----------------------------- ------------------------
! PHYSICS AND MATHS CONSTANTS
!----------------------------- -----------------------

Math :

pi = 4*ATAN(1.) (which we define one time for all modules)

! e :: euler = 2.71828182846 (used in stomate, defined in stomate_constants)

Physics :

R_Earth (radius in meters) = 6378000 (same remark as pi)

!================
! constantes.f90
!================

! *One day in seconds* :: one_day

! *One year in seconds* :: one_year
!-
! *0 degre Celsius in degre Kelvin* :: tp_00=273.15

!====================
! constantes_veg.f90
!====================

! *Latent heat of sublimation* :: chalsu0 = 2.8345E06

! *Latent heat of evaporation* :: chalev0 = 2.5008E06

! *Latent heat of fusion* :: chalfu0 = chalsu0-chalev0

!$!!!TO DELETE
! Latent heat of evaporation 2 (?)
REAL(r_std),PARAMETER :: chalev1 = 2.5008E06

! Stefan-Boltzman constant :: c_stefan = 5.6697E-8

! Specific heat of air :: cp_air = 1004.675

! Constante molere :: cte_molr = 287.05

! Kappa :: kappa = cte_molr/cp_air

! in -- Kg/mole :: msmlr_air = 28.964E-03

! in -- Kg/mole :: msmlr_h2o = 18.02E-03
! cp_h2o = cp_air*(4._r_std*msmlr_air)/(3.5_r_std*msmlr_h2o)
!
cte_molr_h2o = cte_molr/4._r_std
!
retv = msmlr_air/msmlr_h2o-1._r_std
!
rvtmp2 = cp_h2o/cp_air-1._r_std
!
cepdu2 = (0.1_r_std) **2

! *Van Karmann Constante* :: ct_karman = 0.35_r_std

! *g acceleration* :: cte_grav = 9.80665_r_std

! *Transform pascal into hectopascal* :: pa_par_hpa = 100._r_std

!=====================
!stomate_constants.f90
!=====================

! *Freezing point* :: ZeroCelsius = 273.15

!------------------------------ ------------------------------
! **CONSTANTS SOIL THERMODYNAMICS**
!------------------------------ ----------------------------

!====================
! constantes_soil.f90
!====================

! *Average Thermal Conductivity of soils* :: so_cond = 1.5396

! *Average Heat capacity of soils* :: so_capa = 2.0514e+6
!-
! *Values taken from : PIELKE,'MESOSCALE METEOROLOGICAL MODELING',P.384*
! Dry soil heat capacity was decreased and conductivity increased.
!-
! *Dry soil Heat capacity of soils* :: so_capa_dry = 1.80e+6

! *Dry soil Thermal Conductivity of soils* :: so_cond_dry = 0.40
!-
! *Wet soil Heat capacity of soils* :: so_capa_wet = 3.03e+6

! *Wet soil Thermal Conductivity of soils* :: so_cond_wet = 1.89
!-
! *Thermal Conductivity of snow* :: sn_cond = 0.3

! *Snow density for the soil thermodynamics* :: sn_dens = 330.0

! *Heat capacity for snow* :: sn_capa = 2100.0_r_std*sn_dens

!------------------------------ ----------------------------
! **MODEL AND CLIMATIC PARAMETERS**
!------------------------------ --------------------------

!==================
! constantes_veg.f90
!==================

! *Constantes of the Louis scheme*
REAL(r_std),PARAMETER :: cb = 5._r_std
REAL(r_std),PARAMETER :: cc = 5._r_std
REAL(r_std),PARAMETER :: cd = 5._r_std
!-
! *Constant in the computation of surface resistance* :: rayt_cste = 125.
!-
! *Size of local array to keep saturated humidity*
! at each temperature level :: max_temp=370

! *Minimum temperature for saturated humidity* :: min_temp=100

!======================
! stomate_constants.f90
!======================

! *maximum reference long term temperature (K)*:: tlong_ref_max=303.1

! minimum reference long term temperature (K):: tlong_ref_min=253.1
!-

! **climatic parameters**
!-
! *critical value for being adapted (1-1/e)* :: adapted_crit = 1. - (1. / euler)

! *critical value for being regenerative (1/e)*:: regenerate_crit = 1. / euler
!-

!------------------------------ ----------------------
! OPTIONS (logical parameters and others)
!------------------------------ ----------------------

!================
! constantes.f90
!===============

! *To set for more printing* :: long_print = .FALSE.
!-
! *One of the most frequent problems is a temperature out of range*
! we provide here a way to catch that in the calling procedure. (JP)
:: diag_qsat = .TRUE.
!-
! Selects the type of output for the model.
! Value is read from run.def in intersurf_history. :: almaoutput

!===================
! constantes_veg.f90
!==================

l_qsat_first=.TRUE.
!-
! *Flags that (de)activate parts of the model* :: control

! allow agricultural PFTs agriculture = .TRUE.

! *Is veget_ori array stored in restart file*:: ldveget_ori_on_restart = .TRUE.
!-
! *Set to .TRUE. if you want q_cdrag coming from GCM* :: ldq_cdrag_from_gcm = .FALSE.

!======================
! stomate_constants.f90
!=====================

! *Do we treat PFT expansion across a grid point after introduction?*
! default = .FALSE.

LOGICAL,SAVE :: treat_expansion = .FALSE.
!-
! *herbivores?* :: ok_herbivores = .FALSE.
!-
! *harvesting ?* :: harvest_agri = .TRUE.
!-

!=============
! stomate.f90
!=============

! *Maximum STOMATE time step (days)* :: max_dt_days = 5.

! *Number max of time steps per year for carbon spinup* :: nparanmax=36

! *dummy time step, must be zero* :: dt_0 = 0.

```
!=================
! slowproc.f90
!=================
```

check = .FALSE.