SURFEX SETUP

P. Le Moigne

February 15, 2006

Contents

Part I

INITIALIZATION

Chapter 1

INIT_SURF_ATM_n

Setup of the externalized surface (surfex hereafter) is called from main program through init_surf_atm_n subroutine. In the model case, it's called outside temporal loop.

1.1 Arguments

1.1.1 input arguments

Variable name	Description	Type
HPROGRAM	name of the program which calls surface schemes, used to define the input file type.	Chain of 6 characters with the following possible values: 'ASCII ', 'OFFLIN', 'MESONH', 'AROME ' which are the currently available options
HINIT	configuration of initialization in order to define which field to initialize	Chain of 3 characters with the following possible values: 'PGD', 'PRE', 'ALL'. It decides which field to initialize
KI	number of points	integer
KSV	number of scalars	integer
KSW	number of short-wave spectral bands	integer
HSV	name of all scalar variables.	Character array of dimension KSV
PCO2	CO2 concentration (kg/m3).	Real array of dimension KI
PRHOA	Air density.	Real array of dimension KI
PZENITH	Solar zenital angle.	Real array of dimension KI
PAZIM	Solar azimutal angle (rad from North, clockwise).	Real array of dimension KI Real array of dimension KI
PSW_BANDS	Middle wavelength of each band.	Real array of dimension KSW
KYEAR	current year.	integer
KMONTH	current month.	integer
KDAY	current day.	integer
PTIME	current time since midnight (seconds).	real
HATMFILE	atmospheric file name.	Chain of 28 characters
HATMFILETYPE	atmospheric file type.	Chain of 6 characters
HTEST	must be equal to 'ok' at this stage.	Chain of 2 characters

1.1.2 output arguments

Variable name	Description	Type
PDIR_ALB	direct albedo for each band	Real array of dimension (KI,KSW)
PSCA_ALB	scattered albedo for each band	Real array of dimension (KI,KSW)
PEMIS	emissivity	Real array of dimension (KI)
PTSRAD	radiative temperature	Real array of dimension (KI)

1.2 Modules

1.2.1 data

- MODD_SURF_ATM_n: declaration of surface parameters
- MODD_CH_SURF_n: declaration of chemical surface parameters
- MODD_SV_n: declaration of scalar variables
- MODD_SURF_ATM_GRID_n: declaration of grid
- MODD_DIAG_SURF_ATM_n: declaration of diagnostics for the surface
- MODD_DATA_COVER_PAR: declaration of parameters related to surface physiography initialization if ecoclimap is used
- MODD_SURF_DST_n: declaration of parameters related to dust calculations
- MODD_SURF_PAR: declaration of surface parameters

1.2.2 interface

Module interfaces required for INIT_SURF_ATM_n:

MODI_INIT_IO_SURF_n
MODI_DEFAULT_SURF_ATM
MODI_DEFAULT_CH_SURF_ATM
MODI_DEFAULT_DIAG_SURF_ATM
MODI_READ_DEFAULT_SURF_ATM_n
MODI_READ_SURF_ATM_CONF_n
MODI_READ_SURF_ATM_DATE
MODI_READ_SURF
MODI_GET_SIZE_FULL_n
MODI_READ_COVER_n
MODI_READ_SSO_n
MODI_READ_DUMMY_n
MODI_READ_GRID
MODI_END_IO_SURF_n
MODI_DST_INIT_NAMES
MODI_AVERAGE_RAD
MODI_CONVERT_COVER_FRAC
MODI_WRITE_COVER_TEX_START
MODI_WRITE_COVER_TEX_END
MODI_CH_INIT_NAMES
MODI_CH_INIT_DEPCONST
MODI_CH_INIT_EMISSION_n
MODI_OPEN_NAMELIST
MODI_CLOSE_NAMELIST

1.3 Main calls

1.3.1 High level setup

SURF_VERSION
Initialization of the version and the bugfix number corresponding to the user's choice.

INI_CSTS
Initialization of physical MODD_CSTS module containing fundamental constants (like π, lightspeed, ...) as well as constant for astronomy (like day length, sideral day length, ...), terrestrial geoide description (earth radius and gravity strengh), radiation (solar, constant, ...), thermodynamics (capacities, latent heats, ...) and reference pressure.

1.3.2 setup of parameters from cover

INI_DATA_COVER
Initialization of cover-field correspondance arrays. See chapter 2 for more details.

1.3.3 Miscellaneous setup

WRITE_COVER_TEX_START
Opens the output tex file (named class_cover_data.tex) containing cover data description.

WRITE_COVER_TEX_COVER
Writes the cover data arrays corresponding to the area of interest into the tex file.

GET_LUOUT
Get output listing logical unit (interface to MNHGET_LUOUT for Meso-NH or AROGET_LUOUT for AROME, unit number 10 in the other cases (Offline and ascii modes))

1.3.4 Default setup

DEFAULT_SURF_ATM
Defines default values for prognostic fields writing time step.

DEFAULT_CH_SURF_ATM
Defines default values for surface chemistry.

DEFAULT_DIAG_SURF_ATM
Defines default values for the choice of diagnostics like flags for operational 2m meters quantities, for computation of budget, as well as the choice of diagnostics time step writing.

1.3.5 Default from file or namelist

READ_DEFAULT_SURF_ATM
Routine to read the default general configuration for surface by reading namelist like NAM_DIAG_SURFN, NAM_DIAG_SURF_ATMN, NAM_CH_CONTROLN, NAM_CH_SURFN: in a .des file (description file) this is done in the case of Meso-NH and AROME models only.

READ_SURF_ATM_CONF_n
Routine to read the general configuration for surface by reading model namelist like NAM_DIAG_SURFN, NAM_DIAG_SURF_ATMN, NAM_CH_CONTROLN, NAM_CH_SURFN (OPTIONS.nam in the off-line case, other names of file containing thoses namelists are used in Meso-NH or AROME models, however, namelists names remain the same).

1.3.6 General options

Date
HINIT = 'PGD': since date is unknown in PGD files, it's set to undefined default value. HINIT = 'PRE': idem as 'PGD' but in addition date is searched in namelist NAM_SURF_ATM. If date not present in the namelist, then look for a file in the namelist. If there's no date and no file, it checks if atmospheric date is initialized, if it's the case then date gets atmospheric date values, if not it aborts. HINIT = 'ALL': corresponds to the model configuration, during which date is read from initial file.

At this stage IO mechanism starts because IO may be distributed over several processors.

Schemes used
The schemes are set up by reading information in the initial file that should contain it. CSEA, CWATER, CNATURE and CTOWN are respectively the names of sea, inland water, nature and town schemes.

Dimensions
Setup of all dimensions: NDIM_FULL represents the total number of grid points of the working area, NDIM_SEA is the total number of grid points containing a sea surface, NDIM_WATER is the total number of grid points containing an inland water surface, NDIM_NATURE is the total number of grid points containing a natural surface and NDIM_TOWN is the total number of grid points containing an urban surface. Locally, the total number of points by processor is named NSIZE_FULL and is retrieved with the routine called GET_SIZE_FULL_n.

Ecoclimap or not
Setup of logical variable LECOCLIMAP which will decide if surface parameters will be derived from ECOCLIMAP database or not (in this case they can be initialized from external files)

1.3.7 Setup of fields over each processor

Cover fields and grid
READ_COVER_n is the routine that reads the logical array LCOVER (1 : 255) and that initialze the fraction of each cover as well as the orography over the processor. READ_GRID will initialize the horizontal grid.

Chemical species and dust
If any passive scalar entity, it reads and filters the chemical species and dust, chemical emissions and chemical dry deposition scheme. Then it finds the number of dust modes to be transported and initializes the dust emission scheme.

Subgrid scale orography
READ_SSO reads subgrid scale orography (SSO) parameters like standard deviation, anisotropy, direction and slope of SSO. It reads also the average, the silhouette, the minimum and maximum orography and finally the parameters H/2 and A/S: variances of the SSO in the four possible directions.

Dummy fields
Possibility to read dummy fields.

End of the IO.

1.3.8 Fractions of each surface type

CONVERT_COVER_FRAC
Called only if ecoclimap is used, it returns XSEA, XNATURE, XWATER and XTOWN which are the fraction of sea, nature, inland water and town of the part of all grid boxes covered by the current processor.

1.3.9 Initialization of masks

The number of geographical points over the processor which have a non-zero fraction is computed as follows, for example for natural surface:
$NSIZE\_NATURE = COUNT(XNATURE(:) > 0.0)$. This is the dimension of the mask associated to the NATURE surface. Knowing this dimension, the total NSIZE_FULL over the processor and the fraction of nature XNATURE, it's possible to computed the 1D-vector mask NR_NATURE using routine GET_1D_MASK.

Packing principle:
Lets take the following example where the grid is composed with the 4 possible tiles: nature, water, sea and town, organized as follows (numbers from 1 to 12 represent the index of the grid boxes in the full grid), in this particular case, one grid box is represented with an entire tile (pure pixels), but in reality, each tile may be represented in each grid box:

1 NATURE	2 NATURE	3 TOWN	4 TOWN
5 WATER	6 NATURE	7 SEA	8 TOWN
9 NATURE	10 SEA	11 SEA	12 NATURE

In this case, $NSIZE\_FULL = 12$ and the fraction and the size of the mask of each tile is given in the following table:

XNATURE	=	(1.,1.,0.,0.,0.,1.,0.,0.,1.,0.,0.,1.)	NSIZE_NATURE	=	5
XTOWN	=	(0.,0.,1.,1.,0.,0.,0.,1.,0.,0.,0.,0.)	NSIZE_TOWN	=	3
XSEA	=	(0.,0.,0.,0.,0.,0.,1.,0.,0.,1.,1.,0.)	NSIZE_SEA	=	3
XWATER	=	(0.,0.,0.,0.,1.,0.,0.,0.,0.,0.,0.,0.)	NSIZE_WATER	=	1

Once the fraction and the size of the mask of each tile is computed, it becomes possible to pack the variables over each tile to deduce the effective mask:

N	N		
	N		
N			N

		T	T
			T

		S	S
		S	

W			

These four tables represent the repartition of each tile over the grid and define the masks:

$NR\_NATURE = (1, 2, 6, 9, 12)$
$NR\_TOWN = (3, 4, 8)$
$NR\_WATER = (5)$
$NR\_SEA = (7, 10, 11)$

If YP is the packed variable of a Y variable defined over the full grid, it comes:
$YP(INDEX) = Y(MASK(INDEX))$ for $INDEX$ going from 1 to the size of the mask.

1.3.10 Initialization of sea from XSEA fraction

PACK_SURF_INIT_ARG
Variables which are arguments to this routine are packed over sea.

INIT_SEA_n
The arguments of this routine are exactly the same as for INIT_SURF_ATM. The aim is to initialize sea tile. Different options, driven by variable CSEA are available. There are three possibilities for CSEA: NONE, FLUX or SEAFLUX. **NONE** means that no specific scheme

is specified over sea, then direct and scattered albedo are set to zero, emissivity to 1 and surface radiative temperature to triple point temperature. **FLUX** means that values of the surface fluxes for the potential temperature, the vapor, the horizontal components of the wind and the scalar variables are prescribed in (INIT_IDEAL_FLUX routine). **SEAFLUX** is the name of the parameterization used for sea. Initialization is done in INIT_SEAFLUX_n which has the same structure as INIT_SURF_ATM_n described in chapter 1.

<u>UNPACK_SURF_INIT_ARG</u>
Unpack will have the effect to go from NR_SEA mask to NSIZE_FULL mask which corresponds to the full mask for the given processor.

1.3.11 Initialization of lakes from XWATER fraction

<u>PACK_SURF_INIT_ARG</u>
Variables which are arguments to this routine are packed over water.

<u>INIT_WATER_n</u>
The arguments of this routine are exactly the same as for INIT_SURF_ATM. The aim is to initialize inland water tile. Different options, driven by variable CWATER are available. There are three possibilities for CWATER: NONE, FLUX or WATFLUX. **NONE** means that no specific scheme is specified over sea, then direct and scattered albedo are set to zero, emissivity to 1 and surface radiative temperature to triple point temperature. **FLUX** means that values of the surface fluxes for the potential temperature, the vapor, the horizontal components of the wind and the scalar variables are prescribed in (INIT_IDEAL_FLUX routine). **WATFLUX** is the name of the parameterization used for inland water. Initialization is done in INIT_WATFLUX_n which has the same structure as INIT_SURF_ATM_n described in chapter 1.

<u>UNPACK_SURF_INIT_ARG</u>
Unpack will have the effect to go from NR_WATER mask to NSIZE_FULL mask which corresponds to the full mask for the given processor.

1.3.12 Initialization of vegetation scheme from XNATURE fraction

<u>PACK_SURF_INIT_ARG</u>
Variables which are arguments to this routine are packed over nature.

<u>INIT_NATURE_n</u>
The arguments of this routine are exactly the same as for INIT_SURF_ATM. The aim is to initialize nature tile. Different options, driven by variable CNATURE are available. There are four possibilities for CNATURE: NONE, FLUX, ISBA or TSZ0. **NONE** means that no specific scheme is specified over sea, then direct and scattered albedo are set to zero, emissivity to 1 and surface radiative temperature to triple point temperature. **FLUX** means that values of the surface fluxes for the potential temperature, the vapor, the horizontal components of the wind and the scalar variables are prescribed in (INIT_IDEAL_FLUX routine). **ISBA** or **TSZ0** is the name of the parameterization used for natural areas. Initialization is done in INIT_ISBA_n which has the same structure as INIT_SURF_ATM_n described in chapter 1.

<u>UNPACK_SURF_INIT_ARG</u>
Unpack will have the effect to go from NR_NATURE mask to NSIZE_FULL mask which corresponds to the full mask for the given processor.

1.3.13 Initialization of town scheme from XTOWN fraction

<u>PACK_SURF_INIT_ARG</u>
Variables which are arguments to this routine are packed over town.

INIT_TOWN_n
The arguments of this routine are exactly the same as for INIT_SURF_ATM. The aim is to initialize town tile. Different options, driven by variable CTOWN are available. There are three possibilities for CTOWN: NONE, FLUX or TEB. **NONE** means that no specific scheme is specified over sea, then direct and scattered albedo are set to zero, emissivity to 1 and surface radiative temperature to triple point temperature. **FLUX** means that values of the surface fluxes for the potential temperature, the vapor, the horizontal components of the wind and the scalar variables are prescribed in (INIT_IDEAL_FLUX routine). **TEB** is the name of the parameterization used for town. Initialization is done in INIT_TEB_n which has the same structure as INIT_SURF_ATM_n described in chapter 1.

UNPACK_SURF_INIT_ARG
Unpack will have the effect to go from NR_TOWN mask to NSIZE_FULL mask which corresponds to the full mask for the given processor.

1.3.14 Output radiative fields

This is the last step of the initialization. It's done when the contribution of each tile has been set up. It consists in averaging the emissivity, the direct and diffuse albedo and the radiative surface temperature according to fraction of each tile in the mesh. This operation is done in routine AVERAGE_RAD.

Chapter 2

INI_DATA_COVER

Initialization of cover-field correspondance arrays. This is performed in three steps.

Default setup
At first, all arrays contained in MODD_DATA_COVER are allocated and set up with undefined values.

Cover initialization
Then effective initialization routines are called: one for each cover (ie from 1 to 243, plus 255). According to the cover type, different parameters are initialised. Fractions of town, water, nature and sea are set up for all the covers. For example, for cover number 001 which corresponds to sea, sea fraction is set to 1. and other fractions to zero.
Secondary parameters for each cover are initialized as follows (note that according to the cover, not all secondary parameters are set up):

parameters for cover including natural areas

XDATA_VEGTYPE	Vegetation type
XDATA_LAI	Leaf Area Index
XDATA_ROOT_DEPTH	Root depth
XDATA_GROUND_DEPTH	Total depth

But for cover 165 which corresponds to atlantic crops, these fractions will be set up as well as the leaf area index temporal cycle (given every decade) for the two different possible photosynthesis (C3 and C4), the fraction of each vegetation type, the root depth and the ground depth for the two types corresponding to C3 and C4.

parameters for cover including urban areas.

XDATA_ALB_ROOF	roof albedo
XDATA_ALB_ROAD	road albedo
XDATA_ALB_WALL	wall albedo
XDATA_EMIS_ROOF	roof emissivity
XDATA_EMIS_ROAD	road emissivity
XDATA_EMIS_WALL	wall emissivity
XDATA_HC_ROOF	roof heat capacity
XDATA_HC_ROAD	road heat capacity
XDATA_HC_WALL	wall heat capacity
XDATA_TC_ROOF	roof thermal conductivity
XDATA_TC_ROAD	road thermal conductivity
XDATA_TC_WALL	wall thermal conductivity
XDATA_D_ROOF	roof depth
XDATA_D_ROAD	road depth
XDATA_D_WALL	wall depth
XDATA_BLD_HEIGHT	building height
XDATA_WALL_O_HOR	building shape
XDATA_BLD	building fraction
XDATA_CAN_HW_RATIO	canyon shape
XDATA_H_TRAFFIC	anthropogenic flux (traffic)
XDATA_LE_TRAFFIC	anthropogenic flux (traffic)
XDATA_H_INDUSTRY	anthropogenic flux (industry)
XDATA_LE_INDUSTRY	anthropogenic flux (industry)

Other secondary parameters setup

2.1 Description of fixed parameters

These parameters are derived from lookup tables.
Finally, secondary parameters on natural covers are set up from VEGTYPE and LAI, in subroutine INI_DATA_PARAM. This stands for the following parameters dimensionned as (JPCOVER,NVEGTYPE) with JPCOVER = 255 and NVEGTYPE = 12:

XDATA_ALBNIR_VEG	near infra-red albedo over vegetation
XDATA_ALBVIS_VEG	visible albedo over vegetation
XDATA_ALBUV_VEG	ultra-violet albedo over vegetation
XDATA_RSMIN	minimal stomatal resistance of vegetation
XDATA_GAMMA	coefficient used in the computation of the minimal stomatal resistance
XDATA_WRMAX_CF	coefficient for maximum water interception storage capacity on the vegetation
XDATA_CV	inverse of vegetation thermal capacity
XDATA_GMES	mesophyll conductance
XDATA_GC	cuticular conductance
XDATA_RE25	ecosystem respiration
XDATA_F2I	critical normalized soil water content for stress parametrization
XDATA_BSLAI	ratio biomass over lai
XDATA_DMAX	maximum air saturation deficit tolerate by vegetation
XDATA_STRESS	defensive or offensive strategy
XDATA_SEFOLD	e-folding time for senescence
XDATA_LAIMIN	minimum LAI
XDATA_CE_NITRO	leaf area ratio sensitivity to nitrogen concentration
XDATA_CF_NITRO	lethal minimum value of leaf area ratio
XDATA_CNA_NITRO	nitrogen concentration of active biomass
XDATA_ROOT_EXTINCTION	coefficient for cumulative root fraction
XDATA_ROOT_LIN	Ponderation coefficient between formulations for cumulative root fraction
XDATA_SOILRC_SO2	Coefficient for chemistry deposition of SO2
XDATA_SOILRC_O3	Coefficient for chemistry deposition of O3
XDATA_Z0_O_Z0H	ratio between z0 and z0h (taken equal to 10.)

2.2 Description of monthly dependant parameters

Some parameters are time dependant (frequency decade) and the associated fields are dimensionned as (JPCOVER = 255, JPDECAD = 36, NVEGTYPE = 12), this stands for :

XDATA_VEG	fraction of vegetation from LAI computed in VEG_FROM_LAI routine
XDATA_GREEN	fraction of greeness from LAI computed in GREEN_FROM_LAI routine
XDATA_Z0	roughness length from LAI computed in Z0V_FROM_LAI routine
XDATA_EMIS_ECO	emissivity of vegetation from VEG computed in EMIS_FROM_VEG routine

VEG_FROM_LAI

Computes the vegetation fraction as a function of leaf area index and vegetation type. For a given grid box, the mean leaf area index is locally modified to take into account the presence of bare ground, rocks or permanent snow: if such vegetation types exist in the grid box, it means that the LAI over vegetation is bigger than the mean value. Then the fraction of vegetation is computed by using the fraction of the vegetation type, and a weighting coefficient depending on LAI for crops (C_3, C_4 and irrigated crops as well) and a constant value for the other types:

$$veg = \sum_i \beta_i F_i$$

where F_i represents the fraction of the vegetation type in the grid box and β_i is given in the following table:

vegetation type	weighting coefficient β
C_4 crops C_3 crops irrigated crops	1.-exp(-0.6LAI)
broadleaf forest coniferous forest grassland tropical grassland irrigated parks	0.95
equatorial forest	0.99
bare ground permanent snow rocks	0.

Let's take the example of a grid box composed of 10% of bare soil, 40% of crops of C_4 type and 50% of grassland. If the mean LAI over the box is 2., then LAI over vegetation will be 2.22 and the vegetation fraction will computed as $0.736 \times F_{C_4} + 0.95 \times F_{grassland} + 0. \times F_{bareground}$

Z0V_FROM_LAI

Computes the vegetation roughness length as a function of leaf area index and vegetation type. For a given grid box, the mean leaf area index is locally modified to take into account the presence of bare ground, rocks or permanent snow: if such vegetation types exist in the grid box, it means that the LAI over vegetation is bigger than the mean value. Then the vegetation roughness length of vegetation is computed like follows:

$$Z_0 = \max(0.001, Z_{ref} \times exp(-1./\sqrt{Z_{SUM}}))$$

where

$$Z_{SUM} = \sum_i \frac{F_i}{(\ln(0.13 \times Z_i/Z_{ref}))^2}$$

F_i is the fraction of vegetation type, Z_{ref} is the reference height of 10 meters and Z_i is the height of each vegetation type given in the following table:

vegetation type	height Z_i
C_4 crops	min(2.5,exp((LAI-3.5)/1.3))
C_3 crops	min(2.5,exp((LAI-3.5)/1.3))
irrigated crops	min(2.5,exp((LAI-3.5)/1.3))
broadleaf forest	H_{trees}: height of trees
coniferous forest	H_{trees}: height of trees
equatorial forest	H_{trees}: height of trees
grassland	LAI/6.
tropical grassland	LAI/6.
irrigated parks	LAI/6.
bare ground	0.1
permanent snow	0.01
rocks	1.

EMIS_FROM_VEG

Computes the emissivity as a function of the vegetation fraction and the vegetation type:

$$\epsilon = veg \times \epsilon_{veg} + (1 - veg) \times (\epsilon_{snow} F_{snow} + \epsilon_{soil}(1 - F_{snow}))$$

Chapter 3

INIT_SEA_n

3.1 Setup of physiographical fields

READ_PGD_SEAFLUX_n
After having retrieved the usefull dimension, the initial file is read to set up LCOVER (cover classes) and the fraction of each cover (if present for sea surface). Orography is set to zero and the grid is initialized (latitude, longitude and mesh_size)

3.2 Setup of prognostic fields

READ_SEAFLUX_n
The main difference between INIT_SURF_ATM_n and INIT_SEA_n is the reading of prognostic fields in READ_SEAFLUX_n. At first dimension of sea surface is recovered from GET_TYPE_DIM_n routine. Then SST and sea roughness length are read from initial file.

3.3 Output radiative fields

There are two possibilities to compute albedo over sea. The first consists in setting direct and scattered component of albedo to a uniform value, taken equals to 0.135. The second option takes into account the solar zenithal angle and compute the direct albedo component, while the scattered is set constant and equals to 0.06. Emissivity is set to 0.98 over sea. If SST is lower than 271.16 (2 degrees below triple point) then sea ice is considered and both components of albedo are set to 0.85. Emissivity over ice is the same as for snow and is set to 1. Surface radiative temperature is not modified and is set to the SST read in the initial file.

Chapter 4

INIT_WATER_n

4.1 Setup of physiographical fields

READ_PGD_WATFLUX_n
After having retrieved the usefull dimension, the initial file is read to set up LCOVER (cover classes) and the fraction of each cover (if present for inland water surface). Orography is read from file and the grid is initialized (latitude, longitude and mesh_size)

4.2 Setup of prognostic fields

READ_WATFLUX_n
Prognostic fields are read via routine READ_WATFLUX_n. Like for seas and oceans, the dimension of this tile is set up and then surface temperature and roughness lenght are read from initial file.

4.3 Output radiative fields

If surface temperature for lakes is greater than triple point temperature then lake isn't frozen and albedo is set to 0.135 and emissivity to 0.98. In the contrary case albedo is set to 0.85 and emissivity to 1.

Chapter 5

INIT_ISBA_n

5.1 Setup of physiographical fields

READ_PGD_ISBA_n
After having retrieved the usefull dimension, the initial file is read to set up some high level options or dimensions like the soil scheme or the type of photosynthesis that will be used, the number of layers in the soil as well as the number of patches. Then LCOVER (cover classes) and the fraction of each cover (if present for nature surface) are set up from initial file. Orography is read and the grid is initialized (latitude, longitude and mesh_size). Fraction of clay and sand for the first layer are read in the initial file (this is the only available value) and then the clay and sand profiles are initialized constant for the other layers. Subgrid scale orography parameters are then read (they'll be used to compute dynamical roughness length and the effective surface of energy exchange (slope of SSO). Finally, orographic runoff coefficient and chemical emissions (if activated) are read from file.

5.2 Setup of physiographical fields from land cover

CONVERT_COVER_ISBA
Before calling this conversion routine, the current decade is set up (for parameters that vary in time). Converts surface cover classes into secondary physiographic variables for ISBA. When several covers are present in the same grid mesh, the grid mesh value of a given parameter is obtained by averaging of the values for each cover (according to the relative fraction of the covers in the grid mesh). There are three kinds of averaging: arithmetic, inverse and an arithmetic averaging of the neutral Cd coefficient (used for roughness length only), they are realized in routine AV_PGD. For that purpose, weighting functions are defined for each cover according to the surface type as describe in th following table:

surface type	name of surface type	weighting function ω
full grid box	'ALL'	1.
nature	'NAT'	f_{NAT} fraction of nature
sea	'SEA'	f_{SEA} fraction of sea
water	'WAT'	f_{WAT} fraction of water
town	'TWN'	f_{TWN} fraction of town
building	'BLD'	$f_{TWN}.f_{BLD}$ where f_{BLD} is the fraction of building
street	'STR'	$f_{STR}.(1. - f_{BLD})$
trees	'TRE'	$f_{NAT}. \sum f_{VEGTYPE}$ for vegetation types containing trees

Knowing the weighting function ω_j for each cover and the fraction f_j of the cover in the grid mesh, the parameter Y_j defined for each cover can be averaged using arithmetic averaging:

$$\overline{Y} = \frac{\sum_j f_j \omega_j Y_j}{\sum_j f_j \omega_j}$$

or inverse averaging:

$$\overline{Y} = \frac{\sum_j f_j \omega_j}{\sum_j \frac{f_j \omega_j}{Y_j}}$$

or averaging the neutral coefficient for roughness length:

$$\overline{Y} = h_{ref} \exp(- \sqrt{\frac{\sum_j f_j \omega_j}{\sum_j \frac{f_j \omega_j}{(\ln(\frac{z_{ref}}{Y_j}))^2}}})$$

where h_{ref} is the first model level if specified as argument to AV_PGD or $h_{ref} = 20.$ meters by default.

The following table summarizes the different parameters that are initialized (the description of these parameters is done in chapter 2). The type of surface defines the weight function for the averaging

parameter	type of surface where field is defined	type of averaging
H_TREE	TRE	ARI
VEGTYPE	NAT	ARI
VEG	NAT	ARI
LAI	VEG	ARI
Z_0	NAT	CDN
Z_0_O_Z_0H	NAT	ARI
EMIS_ECO	NAT	ARI
RSMIN	LAI	INV
GAMMA	VEG	ARI
WRMAX_CF	VEG	ARI
RGL	VEG	ARI
CV	VEG	INV
DG	NAT	ARI
ROOTFRAC	NAT	ARI
ALBNIR_VEG	VEG	ARI
ALBVIS_VEG	VEG	ARI
ALBUV_VEG	VEG	ARI
GMES	VEG	ARI
RE25	NAT	ARI
BSLAI	VEG	ARI
LAIMIN	VEG	ARI
EFOLD	VEG	ARI
GC	VEG	ARI
DMAX	TRE	ARI
CE_NITRO	VEG	ARI
CF_NITRO	VEG	ARI
CNA_NITRO	VEG	ARI
F2I	VEG	ARI
LSTRESS	VEG	ARI

The setup of soil depth DG is based on predifined values of XDATA_GROUND_DEPTH (2-layer isba option) and XDATA_ROOT_DEPTH (3-layer isba option) as defined in chapter 2 or on the computation of a vertical profile (N layers isba diffusion option) where first layer thickness is set to 3 cm and the total depth is divided into N-1 layers using a geometrical relation (For example, in case of 5 layers in the soil, a total depth of 2.7 meters will return the following base layer depths (from top to buttom): 3 cm (first layer), 10 cm, 30 cm, 90 cm and 2.70 m.

5.3 User defined setup of physiographical fields

INIT_FROM_DATA_ISBA
Parameters that have been described above, may be prepared by *PGD* facility and in this case they're written in the output file of *PGD*. In the case ecoclimap is used, only description of covers are written in order to retrieve the surface parameters during the initialization process.

5.4 Fraction and mask of each patch

SURF_PATCH
Returns the fraction of each patch for each vegetation type. The mask associated to the patch is then constructed.

5.5 Isba-A-gs case

CO2_INIT_n
Initialization of the concentration of CO2. Setup of the ratio biomass over LAI in the nitrogen case.

5.6 Roughness length

SUBSCALE_Z0EFF
The computation takes into account the subscale orography parameters in all directions and returns the components of the roughness length in these directions, the orography roughness length and the surface roughness length.

5.7 Soil characteristics and other specific parameters

Wilting point, field capacity and soil water content at saturation are computed as function of soil texture.

5.8 Dry and wet soil albedo

DRY_WET_SOIL_ALBEDOS
Computes the albedo of bare soil, for dry or wet conditions.

5.9 Prognostic fields

READ_ISBA_n
Reads prognostic variables like soil temperatures, soil liquid water content, soil ice water content, water intercepted on leaves, Leaf Area Index (Ags option), snow mantel. It reads also semi-prognostic variable like the aerodynamic resistance and the latent heat flux as well as the several assimilation terms of Ags and the respiration and biomass from nitrogen version.

5.10 Snow long-wave properties

INIT_SNOW_LW
Setup of snow surface temperature and emissivity depending on the different available snow schemes.

5.11 Soil albedo

SOIL_ALBEDO
Computes the soil albedo of the natural continental parts in each wavelength (visible, near-infrared and UV). Several options are available: "EVOL" = soil albedo evolves with soil wetness, "DRY " = constant soil albedo value for dry soil, "WET " = constant soil albedo value for wet soil, "MEAN" = constant isoil albedo value for medium soil wetness. But in case ECOCLIMAP computations are not chosen, then spectral albedo is initialized during call to $init\_from\_data\_isba$ subroutine.

5.12 Averaged albedo and emissivity

AVERAGED_ALBEDO_EMIS_ISBA
Computes visible, near infra-red and UV albedo for the ecosystem (soil+vegetation), the averaged direct and diffuse albedo, the averaged emissivity and the radiative surface temperature.

Chapter 6

INIT_TEB_n

6.1 Setup of physiographical fields

READ_PGD_TEB_n
After having retrieved the usefull dimension, the initial file is read to set up the number of layers for roads, roofs and walls. Then LCOVER (cover classes) and the fraction of each cover (if present for town surface) are set up from initial file. Orography is read and the grid is initialized (latitude, longitude and mesh_size).

6.2 Setup of physiographical fields from land cover

CONVERT_COVER_TEB
Converts surface cover classes into physiographic variables for TEB. When several covers are present in the same grid mesh, the grid mesh value of a given parameter is obtained by averaging of the values for each cover (according to the relative fraction of the covers in the grid mesh). There are three kinds of averaging: arithmetic, inverse and an arithmetic averaging of the neutral Cd coefficient (used for roughness length only), they are realized in routine AV_PGD. The following table summarizes the different parameters that are initialized (the description of these parameters is done in chapter 2). The type of surface defines the weight function for the averaging: 'TWN' for town, 'BLD' for building only and STR for street only:

parameter	type of surface where field is defined	type of averaging
Z0_TOWN	TWN	CDN
BLD	TWN	ARI
ALB_ROOF	BLD	ARI
EMIS_ROOF	BLD	ARI
HC_ROOF	BLD	INV
TC_ROOF	BLD	ARI
D_ROOF	BLD	ARI
ALB_ROAD	STR	ARI
EMIS_ROAD	STR	ARI
HC_ROAD	STR	INV
TC_ROAD	STR	ARI
D_ROAD	STR	ARI
ALB_WALL	BLD	ARI
EMIS_WALL	BLD	ARI
HC_WALL	BLD	INV
TC_WALL	BLD	ARI
D_WALL	BLD	ARI
BLD_HEIGHT	BLD	ARI
WALL_O_HOR	BLD	ARI
CAN_HW_RATIO	TWN	ARI
H_TRAFFIC	TWN	ARI
LE_TRAFFIC	TWN	ARI
H_INDUSTRY	TWN	ARI
LE_INDUSTRY	TWN	ARI

6.3 User defined setup of physiographical fields

INIT_FROM_DATA_TEB
Parameters that have been described above, may be prepared by *PGD* facility and in this case they're written in the output file of *PGD*. In the case ecoclimap is used, only description of covers are written in order to retrieve the surface parameters during the initialization process.

6.4 Prognostic fields

READ_TEB_n
Reads prognostic variables like roof temperatures, roof water content, road temperatures, road water content, wall temperatures, internal building temperature, deep road temperature, snow cover, temperature and water vapor in canyon air.

6.5 Snow long-wave properties

INIT_SNOW_LW
Setup of snow surface temperature and emissivity depending on the different available snow schemes.

6.6 Averaged emissivity and radiative temperature

AVERAGED_TSRAD_TEB
Computes the averaged emissivity and the radiative surface temperature for TEB scheme.

6.7 Averaged albedo

AVERAGED_ALBEDO_TEB and URBAN_SOLAR_ABS
Computes direct and diffuse albedo from town.