About types in SURFEX V81 / SURFEX V80

1. About GRID types

- Merging of the grid types for the tiles :

V80		*V81*	
TYPE	*ABBREVIATION*	*TYPE*	*ABBREVIATION*
FLAKE_GRID_t	FG	GRID_t	G
WATFLUX_GRID_t	WG		
SEAFLUX_GRID_t	SG		
ISBA_GRID_t	IG		
TEB_GRID_t	TG		

- SURF_ATM_GRID_t (UG) becomes an overtype of GRID_t (G), ie UG contains G ans other fields specific to the SURF_ATM general tile.

→ To refer for example to UG%CGRID, you now need to type UG%G%CGRID.

2. About CANOPY types

- Merging of the canopy types for the tiles :

V80		*V81*	
TYPE	*ABBREVIATION*	*TYPE*	*ABBREVIATION*
FLAKE_SBL_t	FSB	CANOPY_t	SB
WATFLUX_SBL_t	WSB		
SEAFLUX_SBL_t	SSB		
ISBA_CANOPY_t	ICP		
TEB_CANOPY_t	TCP		
SSO_CANOPY_t	SSCP		

3. About diagnostics common to the tiles

- Merging of the diagnostic common types for the tiles :

V80		*V81*	
TYPE	*ABBREVIATION*	*TYPE*	*ABBREVIATION*
DIAG_FLAKE_t	DGF	DIAG_OPTIONS_t + DIAG_t	DO + D
DIAG_WATFLUX_t	DGW		
DIAG_SEAFLUX_t	DGS		
DIAG_ISBA_t	DGI		
DIAG_TEB_t	DGT		
DIAG_SURF_ATM_t	DGU		
DIAG_IDEAL_t	DGL		

NB : the partition of diagnostic types are detailed further for each tile.

4. About ISBA_MODEL_t

4.a. About the management of the patches

- Definition of a type TTT_NP_t compared to a type TTT_t :

  TYPE TTT_NP_t
  TYPE(TTT_t), DIMENSION(:), POINTER :: AL=>NULL()
  END TYPE TTT_NP_t

This approach allows to store the fields defined by patches in those sorts of structures : TT_NP_t%AL(1)%FIELD(:) for patch1, TTT_NP_t%AL(2)%FIELD(:) for patch2, etc.

The main advantages of this approach are to :
- allocate directly each AL(JP) %FIELD(:) to the dimension of the corresponding PATCH
- replace the former packing process at each time step for each patch (hence « PACK » types are suppressed)

- Decomposition of the type ISBA_t :
  - ISBA_OPTIONS_t : options for the ISBA model
  - ISBA_S_t : fields not depending on patches and never packed by patch. S stands for « Stays the same ». Ex : XCOVER
  - ISBA_K_t : fields not depending on patches but packed by patch to be used in the patch loop. K stands for « to be pacKed ». Ex : XSAND
  - ISBA_NK_t : fields from ISBA_K_t packed on patches. N stands for « Numerous patches ».
  - ISBA_NP_t : fields depending on patches and defined by patch, not varying in time. P stands for « defined by Patch » and N for « Numerous patches ». Ex : XDG.

- ISBA_NPE_t : fields depending on patches and defined by patch, varying in time. P tands for « defined by Patch », N for « Numerous patches », E for « Evolving in time ». Ex : XTG.

4.b. General changes in V81

V80		*V81*	
TYPE	*ABBREVIATION*	*TYPE*	*ABBREVIATION*
ISBA_CANOPY_t	ICP	CANOPY_t	SB
DATA_ISBA_t	DTI	DATA_ISBA_t	DTV
AGRI_t	AG	AGRI_NP_t	NAG
ISBA_GRID_t	IG	GRID_NP_t + GRID_t	NG + G
CH_ISBA_t	CHI	CH_ISBA_NP_t + CH_ISBA_t	NCHI + CHI
GR_BIOG_t	GB	GR_BIOG_NP_t + GR_BIOG_t	NGB + GB
PACK_CH_ISBA_t	PKCI	Suppressed (see 4.b.)	
PACK_DIAG_ISBA_t	PKDI		
PACK_ISBA_t	PKI		
ISBA_t	I	ISBA_OPTIONS_t + ISBA_S_t + ISBA_K_t + ISBA_NK_t + ISBA_NP_t + ISBA_NPE_t	O + S + K + NK + NP + NPE
DIAG_ISBA_t	DGI	DIAG_OPTIONS_t + DIAG_t + DIAG_NP_t	O + D + DC + ND + NDC
DIAG_EVAP_ISBA_t	DGEI	DIAG_EVAP_ISBA_t + DIAG_EVAP_ISBA_NP_t	DE + DEC + NDE + NDEC
DIAG_MISC_ISBA_t	DGMI	DIAG_MISC_ISBA_t + DIAG_MISC_ISBA_NP_t	DM + NDM

4.c. About the overtype ISBA_DIAG_t

The overtype ISBA_DIAG_t contains :

- DIAG_OPTIONS_t : options for the diagnostics
- DIAG_t / D : general diagnostics
- DIAG_t / DC : cumulated general diagnostics
- DIAG_NP_t / ND : general diagnostics by patches
- DIAG_NP_t / NDC : cumulated general diagnostics by patches
- DIAG_EVAP_ISBA_t / DE : evaporation diagnostics
- DIAG_EVAP_ISBA_t / DEC : cumulated evaporation diagnostics
- DIAG_EVAP_ISBA_NP_t / NDE : evaporation diagnostics by patches

- DIAG_EVAP_ISBA_NP_t / NDEC : cumulated evaporation diagnostics by patches
- DIAG_MISC_ISBA_t / DM : miscellaneous diagnostics
- DIAG_MISC_ISBA_NP_t / NDM : miscellaneous diagnostics by patches

This decomposition, combined with points 3. and 4.a., allows to :

- defined each actual diagnostic field only one time.

Example :

V80		*V81*	
modd_diag_isban	XGFLUX	modd_diagn	XGFLUX
	XAVG_GFLUX		
modd_diag_evap_isban	XGFLUXC		
	XAVG_GFLUXC		
modd_diag_flaken	XGFLUX		
	XGFLUXC		
modd_diag_idealn	XGFLUX		
	XGFLUXC		
modd_diag_seafluxn	XGFLUX		
	XGFLUXC		
	XGFLUX_ICE		
	XGFLUXC_ICE		
modd_diag_surf_atmn	XGFLUX_TILE		
	XGFLUXC_TILE		
	XAVG_GFLUX		
	XAVG_GFLUXC		
modd_diag_teb_gardenn	XGFLUX		
modd_diag_teb_greenroofn	XGFLUX		
modd_diag_tebn	XGFLUX		
modd_diag_watfluxn	XGFLUX		
	XGFLUXC		
modd_pack_diag_isba	XP_GFLUX		
	XP_GFLUX_ISBA		

- Simplify initialization and averaging of diagnostic fields.

4.d. Examples of use in the Surfex code

4.d.1. Example of use of multi-patches types in a patch loop :

- Arguments of the routine :

TYPE(ISBA_NK_t), INTENT(INOUT) :: NK
TYPE(ISBA_NP_t), INTENT(INOUT) :: NP
TYPE(ISBA_NPE_t), INTENT(INOUT) :: NPE
TYPE(AGRI_NP_t), INTENT(INOUT) :: NAG
TYPE(SSO_NP_t), INTENT(INOUT) :: NISS
!

- Declaration of local types to simplify the reading :

TYPE(ISBA_K_t), POINTER :: KK
TYPE(ISBA_P_t), POINTER :: PK
TYPE(ISBA_PE_t), POINTER :: PEK
TYPE(AGRI_t), POINTER :: AGK
TYPE(SSO_t), POINTER :: ISSK
!

- Patch loop :

DO JP = 1, IO%NPATCH
!
KK => NK%AL(JP)
PK => NP%AL(JP)
PEK => NPE%AL(JP)
AGK => NAG%AL(JP)
ISSK => NISS%AL(JP)
!

This convention allows to write for example PEK instead of NPE%AL(JP) inside the loop, what lightens the writing.

In a general way, a patch instance *NTTT%AL(JP)* of a multi-patches type named *NTTT* is changed to *TTTK :*

- *i*nside a patch loop,
- if declared as an argument of a subroutine.

4.d.2. Example of use in the subroutines of calculation of diagnostics :

By convention, in the subroutines specific to diagnostics, diagnostic types used as arguments of local subroutines get the suffix « A ».

- coupling_isban :
  CALL AVERAGE_DIAG_EVAP_ISBA_n(..., ID%DE,..., ID%NDE, ...)

- average_diag_evap_isban :

SUBROUTINE AVERAGE_DIAG_EVAP_ISBA_n (..., DE, ..., NDE, ...)
…
USE MODD_DIAG_EVAP_ISBA_n, ONLY : DIAG_EVAP_ISBA_t, DIAG_EVAP_ISBA_NP_t
…
TYPE(DIAG_EVAP_ISBA_t), INTENT(INOUT) :: DE
TYPE(DIAG_EVAP_ISBA_t), INTENT(INOUT) :: DEC
…
IF (DE%LSURF_EVAP_BUDGET) CALL MAKE_AVERAGE_EVAP(DE,NDE)
…

```
SUBROUTINE MAKE_AVERAGE_EVAP(DEA,NDEA)
!
TYPE(DIAG_EVAP_ISBA_t), INTENT(INOUT) :: DEA
TYPE(DIAG_EVAP_ISBA_NP_t), INTENT(INOUT) :: NDEA
!
DO JP=1,KNPATCH
  DO JI=1,NP%AL(JP)%NSIZE_P
    IMASK = NP%AL(JP)%NR_P(JI)
    DEA%XLEG (IMASK) = DEA%XLEG (IMASK) + NP%AL(JP)%XPATCH(JI) * NDEA%AL(JP)
%XLEG(JI)
  ...
  ENDDO
ENDDO
```

4.d.3. Example of use of XTG in isba.F90

- offline.F90 :
  CALL COUPLING_SURF_ATM_n(YSC…)

- coupling_surf_atmn.F90 :
  SUBROUTINE COUPLING_SURF_ATM_n (YSC…)
  ...
  USE MODD_SURFEX_n, ONLY : SURFEX_t
  TYPE(SURFEX_t), INTENT(INOUT) :: YSC
  …
  CALL COUPLING_NATURE_n(..., YSC%IM,…)

- coupling_isba_svatn.F90 :
  SUBROUTINE COUPLING_ISBA_SVAT_n (..., IM…)
  ...
  USE MODD_SURFEX_n, ONLY : ISBA_MODEL_t
  TYPE(ISBA_MODEL_t), INTENT(INOUT) :: IM
  ...
  CALL COUPLING_ISBA_OROGRAPHY_n(..., IM%NPE…)

- coupling_isban.F90 :
  SUBROUTINE COUPLING_ISBA_n (...NPE…)
  ...
  USE MODD_ISBA_n, ONLY : ISBA_NPE_t, ISBA_PE_t
  TYPE(ISBA_NPE_t), INTENT(INOUT) ::NPE
  TYPE(ISBA_PE_t), POINTER :: PEK
  ...
  PATCH_LOOP: DO JP=1,IO%NPATCH
  CALL TREAT_PATCH(..., NPE%AL(JP)…)
  ...
  SUBROUTINE TREAT_PATCH(..., PEK…)
  ...
  USE MODD_ISBA_n, ONLY : ISBA_PE_t
  TYPE(ISBA_PE_t), INTENT(INOUT) :: PEK
  ...
  CALL ISBA(..., PEK,…)

- isba.F90 :
  SUBROUTINE ISBA(..., PEK...)
  USE MODD_ISBA_n, ONLY : ISBA_PE_t
  TYPE(ISBA_PE_t), INTENT(INOUT) :: PEK
  ...
  ZQSAT(:)=QSAT(PEK%XTG(:,1),PPS(:))

5. About GARDEN and GREENROOF

- The great change for GARDEN and GREENROOF is that the ISBA types are directly used, and no longer duplicated in types specific to GARDEN and GREENROOF.

V80		*V81*	
TYPE	*ABBR.*	*TYPE*	*ABBR.*
TEB_VEG_OPTIONS_t	TVG	ISBA_OPTIONS_t	O
TEB_GARDEN_OPTIONS_t	TGDO		
TEB_GREENROOF_OPTIONS_t	TGRO		
DATA_TEB_GARDEN_t	DTGD	DATA_ISBA_t	DTV
DATA8TEB_GREENROOF_t	DTGR		
DIAG_TEB_GARDEN_t	DGTGD	TEB_VEG_DIAG_t, contains DIAG_NP_t + DIAG_EVAP_ISBA_NP_t + DIAG_MISC_ISBA_NP_t	VD, contains ND + NDE + NDEC + NDM
DIAG_TEB_GREENROOF_t	DGTGR		
GR_BIOG_GARDEN_t	GBGD	GR_BIOG_t	GB
GR_BIOG_GREENROOF_t	GBGR		
TEB_GARDEN_PGD_t	TGDP	ISBA_S_t + ISBA_K_t	S + K
TEB_GREENROOF_PGD_t	TGRP		
TEB_GARDEN_t	TGD	ISBA_P_t	P
TEB_GREENROOF_t	TGR		
TEB_GARDEN_PGD_EVOL_t	TGDPE	ISBA_NPE_t	NPE
TEB_GREENROOF_PGD_EVOL_t	TGRPE		

NB : In GARDEN and GREENROOF, ISBA_NPE_t refers to the patches in TEB.

6. Other changes

6.a. SURFEX_t

V80		*V81*	
TYPE	*ABBR.*	*TYPE*	*ABBR.*
DIAG_SURF_ATM_t	DGU	DIAG_OPTIONS_t + DIAG_t + DIAG_NP_t	DUO + DU + DUC + DUP + DUPC
DIAG_IDEAL_t	DGL	DIAG_OPTIONS_t + DIAG_t	DL0 + DL + DLC
SURF_ATM_SSO_t	USS	SSO_t	USS
SSO_CANOPY_t	SSCP	CANOPY_t	SB
DST_t	DST	DST_NP_t	NDST

6.b. TEB_MODEL_t

- About diagnostics :

V80		*V81*	
TYPE	*ABBR.*	*TYPE*	*ABBR.*
DIAG_MISC_TEB_OPTIONS_t	DGMTO	DIAG_MISC_TEB_OPTIONS_t	MTO
DIAG_TEB_t	DGT	DIAG_OPTIONS_t + DIAG_t	O + D
DIAG_UTCI_TEB_t	DGUT	DIAG_UTCI_TEB_t	DUT
DIAG_CUMUL_TEB_t	DGCT	DIAG_MISC_TEB_NP_t	NDMT
DIAG_MISC_TEB_t	DGMT	DIAG_MISC_TEB_NP_t	NDMTC

NB : in V81, upper TEB diagnostic types are gathered in the overtype TEB_DIAG_t (TD).

- Other types :

V80		*V81*	
TYPE	*ABBR.*	*TYPE*	*ABBR.*
TEB_t	T	TEB_NP_t	NT
BEM_t	B	BEM_NP_t	NB

6.c. SEAFLUX_MODEL_t

V80		*V81*	
TYPE	*ABBR.*	*TYPE*	*ABBR.*
DIAG_SEAFLUX_t	DGS	DIAG_OPTIONS_t + DIAG_t	O + D + DC
DIAG_OCEAN_t	DGO	DIAG_OCEAN_t	GO
DIAG_SEAICE_t	DGSI	DIAG_t + DIAG_MISC_SEAICE_t	DI + DIC + DMI

NB : in V81, upper SEAFLUX diagnostic types are gathered in the overtype SEAFLUX_DIAG_t (SD).

6.d. WATFLUX_MODEL_t

V80		*V81*	
TYPE	*ABBR.*	*TYPE*	*ABBR.*
DIAG_WATFLUX_t	DGW	DIAG_OPTIONS_t + DIAG_t	DWO + DW+ DWC

6.e. FLAKE_MODEL_t

V80		*V81*	
TYPE	*ABBR.*	*TYPE*	*ABBR.*
DIAG_FLAKE_t	DGF	DIAG_OPTIONS_t + DIAG_t	DFO + DF + DFC
DIAG_MISC_FLAKE_t	DGMF	DIAG_MISC_FLAKE_t	DMF