

THE EXTERNALIZED SURFACE USER'S GUIDE  
V7.1

# Table of Contents

<b><u>1. How to install the software</u></b> .....	<b>1</b>
<b><u>1.1. Export off-line version of SURFEX</u></b> .....	<b>2</b>
<b><u>2. Overview of the externalized surface sequence</u></b> .....	<b>5</b>
<b><u>2.1. The sequence</u></b> .....	<b>6</b>
<b><u>2.2. The atmospheric models using the externalized surface</u></b> .....	<b>7</b>
<b><u>2.2.1. In offline mode</u></b> .....	<b>8</b>
<b><u>2.2.1.1. namelist NAM IO OFFLINE</u></b> .....	<b>9</b>
<b><u>2.2.1.2. namelist NAM LAND USE</u></b> .....	<b>11</b>
<b><u>2.2.1.3. Namelist NAM ZS FILTER</u></b> .....	<b>12</b>
<b><u>2.2.1.4. Namelist NAM NACVEG</u></b> .....	<b>13</b>
<b><u>2.2.2. in MESONH</u></b> .....	<b>14</b>
<b><u>2.2.2.1. Initialization of surface fields integrated in MESONH programs</u></b> .....	<b>15</b>
<b><u>2.2.2.2. Namelist NAM PGDFILE</u></b> .....	<b>16</b>
<b><u>2.2.2.3. MESONH run and diagnostics</u></b> .....	<b>17</b>
<b><u>2.2.3. in AROME</u></b> .....	<b>18</b>
<b><u>3. Off-line Guide</u></b> .....	<b>19</b>
<b><u>3.1. The input files</u></b> .....	<b>20</b>
<b><u>3.2. forcing files</u></b> .....	<b>21</b>
<b><u>3.2.1. creation of forcing files</u></b> .....	<b>22</b>
<b><u>3.2.2. installation of an experiment</u></b> .....	<b>24</b>
<b><u>3.3. One example of off-line surfex application</u></b> .....	<b>25</b>

# Table of Contents

<b><u>3.3.1. netcdf format file</u></b> .....	26
<b><u>3.3.1.1. FORCING.nc</u></b> .....	27
<b><u>3.3.1.2. list of parameters</u></b> .....	28
<b><u>3.3.2. ascii format files</u></b> .....	31
<b><u>3.3.2.1. PGD.txt</u></b> .....	32
<b><u>3.3.2.2. PREP.txt</u></b> .....	34
<b><u>3.3.2.3. Extracting 2d fields</u></b> .....	36
<b><u>3.3.2.4. I/O diagram</u></b> .....	37
<b><u>3.4. Some output of off-line simulation</u></b> .....	38
<b><u>3.4.1. Examples of prognostic variables output</u></b> .....	39
<b><u>3.4.2. list of available variables</u></b> .....	41
<b><u>4. The physiographic fields</u></b> .....	45
<b><u>4.1. Overview of physiographic fields computation: PGD</u></b> .....	46
<b><u>4.1.1. Choice of the grid</u></b> .....	47
<b><u>4.1.2. Choice of the physiographic fields</u></b> .....	48
<b><u>4.1.2.1. Namelist NAM_FRAC</u></b> .....	49
<b><u>4.1.2.2. ideal physiographic fields:</u></b> .....	51
<b><u>4.1.2.3. realistic physiographic fields:</u></b> .....	52
<b><u>4.1.2.4. user defined physiographic fields:</u></b> .....	53
<b><u>4.1.2.4.1. ISBA scheme</u></b> .....	54
<b><u>4.1.2.4.2. TEB scheme</u></b> .....	60

# Table of Contents

<u>4.1.2.4.3. TEB GARDEN scheme</u> .....	64
<u>4.1.2.4.4. SEAFLUX scheme</u> .....	65
<u>4.2. Choice of the surface schemes</u> .....	67
<u>4.2.1. Namelist NAM PGD SCHEMES</u> .....	68
<u>4.3. Definition of the grid</u> .....	69
<u>4.3.1. Choice of the grid type</u> .....	70
<u>4.3.1.1. Namelist NAM PGD GRID</u> .....	71
<u>4.3.2. Conformal projection grids (Mercator, Lambert, Polar stereographic)</u> .....	72
<u>4.3.2.1. Namelist NAM CONF PROJ</u> .....	73
<u>4.3.2.2. Namelist NAM CONF PROJ GRID</u> .....	74
<u>4.3.2.3. Namelist NAM INIFILE CONF PROJ</u> .....	75
<u>4.3.3. Cartesian grids</u> .....	76
<u>4.3.3.1. Namelist NAM CARTESIAN</u> .....	77
<u>4.3.3.2. Namelist NAM INIFILE CARTESIAN</u> .....	78
<u>4.3.4. Longitude-latitude grids</u> .....	79
<u>4.3.4.1. Namelist NAM LONLAT REG</u> .....	80
<u>4.3.4.2. Namelist NAM LONLATVAL</u> .....	81
<u>4.3.5. Regular Lambert grids</u> .....	82
<u>4.3.5.1. Namelist NAM IGN</u> .....	83
<u>4.3.6. Gaussian grids</u> .....	84
<u>4.3.6.1. Namelist NAMDIM</u> .....	85

# Table of Contents

<b><u>4.3.6.2. Namelist NAMRGRI</u></b> .....	<b>86</b>
<b><u>4.3.6.3. Namelist NAMGEM</u></b> .....	<b>87</b>
<b><u>4.4. Land cover fractions</u></b> .....	<b>88</b>
<b><u>4.4.1. Namelist NAM COVER</u></b> .....	<b>89</b>
<b><u>4.4.2. Namelist NAM PGD ARRANGE COVER</u></b> .....	<b>91</b>
<b><u>4.4.3. Namelist NAM READ DATA COVER</u></b> .....	<b>92</b>
<b><u>4.4.4. Namelist NAM WRITE COVER TEX</u></b> .....	<b>93</b>
<b><u>4.5. Specificities of ecoclimap II classification</u></b> .....	<b>94</b>
<b><u>4.5.1. Namelist NAM ECOCLIMAP2</u></b> .....	<b>95</b>
<b><u>4.6. Orography, subgrid orography gaussian indices and bathymetry</u></b> .....	<b>96</b>
<b><u>4.6.1. Namelist NAM ZS</u></b> .....	<b>97</b>
<b><u>4.6.2. Namelist NAM GAUSS INDEX</u></b> .....	<b>98</b>
<b><u>4.6.3. Namelist NAM SEABATHY</u></b> .....	<b>99</b>
<b><u>4.7. Namelist for ISBA scheme</u></b> .....	<b>100</b>
<b><u>4.7.1. Namelist NAM ISBA</u></b> .....	<b>101</b>
<b><u>4.8. Namelist for FLake scheme</u></b> .....	<b>104</b>
<b><u>4.8.1. Namelist NAM DATA FLAKE</u></b> .....	<b>105</b>
<b><u>4.9. Namelist to add user's own fields</u></b> .....	<b>107</b>
<b><u>4.9.1. Namelist NAM DUMMY PGD</u></b> .....	<b>108</b>
<b><u>4.10. Namelist for chemistry anthropogenic emissions</u></b> .....	<b>109</b>
<b><u>4.10.1. Namelist NAM CH EMIS PGD</u></b> .....	<b>110</b>

# Table of Contents

<b><u>5. Initialization of the prognostic fields</u></b> .....	<b>111</b>
<b><u>5.1. Overview of fields computation: PREP</u></b> .....	<b>112</b>
<b><u>5.2. Date initialization and default input data file for all schemes</u></b> .....	<b>113</b>
<b><u>5.2.1. Namelist NAM PREP SURF ATM</u></b> .....	<b>114</b>
<b><u>5.3. Sea scheme "SEAFLX"</u></b> .....	<b>116</b>
<b><u>5.3.1. Namelist NAM PREP SEAFLUX</u></b> .....	<b>117</b>
<b><u>5.4. Lake scheme "WATFLX"</u></b> .....	<b>119</b>
<b><u>5.4.1. Namelist NAM PREP WATFLUX</u></b> .....	<b>120</b>
<b><u>5.5. Lake scheme "FLAKE"</u></b> .....	<b>121</b>
<b><u>5.5.1. Namelist NAM PREP FLAKE</u></b> .....	<b>122</b>
<b><u>5.6. Vegetation scheme "ISBA "</u></b> .....	<b>124</b>
<b><u>5.6.1. Namelist NAM PREP ISBA</u></b> .....	<b>125</b>
<b><u>5.6.2. Namelist NAM PREP ISBA SNOW</u></b> .....	<b>128</b>
<b><u>5.6.3. Namelist NAM PREP ISBA CARBON</u></b> .....	<b>130</b>
<b><u>5.7. Town scheme "TEB "</u></b> .....	<b>131</b>
<b><u>5.7.1. Namelist NAM PREP TEB</u></b> .....	<b>132</b>
<b><u>5.7.2. Namelist NAM PREP TEB SNOW</u></b> .....	<b>135</b>
<b><u>5.7.3. Namelist NAM PREP TEB GARDEN</u></b> .....	<b>136</b>
<b><u>5.7.4. Namelist NAM PREP GARDEN SNOW</u></b> .....	<b>137</b>
<b><u>6. How to run the externalized surface physical schemes</u></b> .....	<b>138</b>
<b><u>6.1. "SURF ATM" general options available over all tiles</u></b> .....	<b>139</b>

# Table of Contents

<b><u>6.1.1. Namelist NAM SURF CSTS</u></b> .....	<b>140</b>
<b><u>6.1.2. Namelist NAM SURF ATM</u></b> .....	<b>141</b>
<b><u>6.1.3. Namelist NAM WRITE SURF ATM</u></b> .....	<b>143</b>
<b><u>6.1.4. Namelist NAM SSQ</u></b> .....	<b>144</b>
<b><u>6.2. "SEAFLX" sea scheme options</u></b> .....	<b>145</b>
<b><u>6.2.1. Namelist NAM SEAFLUXn</u></b> .....	<b>146</b>
<b><u>6.2.2. Namelist NAM SURF SLT</u></b> .....	<b>147</b>
<b><u>6.3. "WATFLX" inland water scheme options</u></b> .....	<b>148</b>
<b><u>6.3.1. Namelist NAM WATFLUXn</u></b> .....	<b>149</b>
<b><u>6.4. "FLAKE" lake scheme options</u></b> .....	<b>150</b>
<b><u>6.4.1. Namelist NAM FLAKEn</u></b> .....	<b>151</b>
<b><u>6.5. "ISBA " vegetation scheme options</u></b> .....	<b>152</b>
<b><u>6.5.1. Namelist NAM SGH ISBAn</u></b> .....	<b>153</b>
<b><u>6.5.2. Namelist NAM ISBAn</u></b> .....	<b>154</b>
<b><u>6.5.3. Namelist NAM SURF DST</u></b> .....	<b>156</b>
<b><u>6.5.4. Namelist NAM ASSIM</u></b> .....	<b>157</b>
<b><u>6.5.5. Namelist NAM AGRI</u></b> .....	<b>158</b>
<b><u>6.5.6. Namelist NAM DEEPSOIL</u></b> .....	<b>159</b>
<b><u>6.5.7. Namelist NAM TREEDRAG</u></b> .....	<b>160</b>
<b><u>6.6. "TEB" town scheme options</u></b> .....	<b>161</b>
<b><u>6.6.1. Namelist NAM TEBn</u></b> .....	<b>162</b>

# Table of Contents

<b><u>6.7. "IDEAL" ideal flux scheme options.....</u></b>	<b>163</b>
<b><u>6.6.1. Namelist NAM IDEAL FLUX.....</u></b>	<b>164</b>
<b><u>7. How to run the externalized surface chemical schemes.....</u></b>	<b>165</b>
<b><u>7.1. Chemical settings control.....</u></b>	<b>166</b>
<b><u>7.1.1. Namelist NAM CH CONTROLn.....</u></b>	<b>167</b>
<b><u>7.2. Chemical anthropogenic emissions.....</u></b>	<b>168</b>
<b><u>7.2.1. Namelist NAM CH SURFn.....</u></b>	<b>169</b>
<b><u>7.3. Chemical deposition over ocean.....</u></b>	<b>170</b>
<b><u>7.3.1. Namelist NAM CH SEAFLUXn.....</u></b>	<b>171</b>
<b><u>7.4. Chemical deposition over lakes.....</u></b>	<b>172</b>
<b><u>7.4.1. Namelist NAM CH WATFLUXn.....</u></b>	<b>173</b>
<b><u>7.5. Chemical deposition over towns.....</u></b>	<b>174</b>
<b><u>7.5.1. Namelist NAM CH TEBn.....</u></b>	<b>175</b>
<b><u>7.6. Chemical deposition and biogenic emissions over vegetation.....</u></b>	<b>176</b>
<b><u>7.6.1. Namelist NAM CH ISBAn.....</u></b>	<b>177</b>
<b><u>7.7. Chemical aerosol scheme (ORILAM).....</u></b>	<b>178</b>
<b><u>7.7.1. Namelist NAM CHS ORILAM.....</u></b>	<b>179</b>
<b><u>8. Externalized surface diagnostics.....</u></b>	<b>180</b>
<b><u>8.1. Diagnostics relative to the general surface monitor.....</u></b>	<b>181</b>
<b><u>8.1.1. Namelist NAM DIAG SURF ATMn.....</u></b>	<b>182</b>
<b><u>8.2. Diagnostics relative to the general surface monitor and to each surface scheme.....</u></b>	<b>183</b>



# Table of Contents

<b><u>8.2.1. Namelist NAM DIAG SURF<sub>n</sub></u></b> .....	184
<b><u>8.2.2. Namelist NAM WRITE DIAG SURF<sub>n</sub></u></b> .....	186
<b><u>8.3. Diagnostics relative to the ISBA vegetation scheme</u></b> .....	187
<b><u>8.3.1. Namelist NAM DIAG ISBA<sub>n</sub></u></b> .....	188
<b><u>8.4. Diagnostics relative to the TEB town scheme</u></b> .....	189
<b><u>8.4.1. Namelist NAM DIAG TEB<sub>n</sub></u></b> .....	190
<b><u>8.5. Diagnostics relative to the FLAKE scheme</u></b> .....	191
<b><u>8.5.1. Namelist NAM DIAG FLAKE<sub>n</sub></u></b> .....	192
<b><u>8.6. Diagnostics relative to the 1D oceanic scheme</u></b> .....	193
<b><u>8.6.1. Namelist NAM DIAG OCEAN<sub>n</sub></u></b> .....	194
<b><u>9. Externalized surface model output fields</u></b> .....	195
<b><u>9.1. Prognostic model output fields</u></b> .....	196
<b><u>9.1.1. ISBA</u></b> .....	197
<b><u>9.1.2. SEAFLUX</u></b> .....	200
<b><u>9.1.3. TEB</u></b> .....	201
<b><u>9.1.4. WATFLUX</u></b> .....	202
<b><u>9.1.5. FLAKE</u></b> .....	203
<b><u>10. Example of namelist features</u></b> .....	204
<b><u>10.1. How to define a target grid</u></b> .....	205
<b><u>10.2. How to use ECOCLIMAP I</u></b> .....	206
<b><u>10.3. How to use ECOCLIMAP II</u></b> .....	207

# Table of Contents

<b><u>10.4. How to use 1D Oceanic Model.....</u></b>	<b>208</b>
<b><u>10.5. How to initialize variables from grib file.....</u></b>	<b>209</b>
<b><u>10.6. How to initialize main ISBA scheme options.....</u></b>	<b>210</b>
<b><u>10.7. How to get lake temperature profile as output.....</u></b>	<b>211</b>
<b><u>10.8. user defined surface parameters.....</u></b>	<b>212</b>
<b><u>10.8.1. Uniform values prescribed: 1d example without patches.....</u></b>	<b>213</b>
<b><u>10.8.2. Uniform values prescribed: 1d example with patches.....</u></b>	<b>218</b>
<b><u>10.8.3. Surface parameters read from external files.....</u></b>	<b>227</b>

# 1. How to install the software

# 1.1. Export off-line version of SURFEX

Instructions to install surfex on a linux-PC and to run a 1d example.

1. select a directory where installation has to be done: for example \$HOME or \$HOME/MYDIR, where MYDIR is an existing directory (if not, it has to be created by the user).  
From now on, it is supposed that the user has defined a MYDIR directory.
2. download EXPORT\_v7.1.gz from surfex web site and move it into \$HOME/MYDIR/EXPORT\_v7.1.gz
3. extract files from archive: tar zxvf EXPORT\_v7.1.gz (or gunzip EXPORT\_v7.1.gz and then tar xvf EXPORT\_v7.1.tar) at this stage directory EXPORT\_v7.1 is created in MYDIR and contains all software peaces.
4. initialize environment variables needed for surfex:
  1. set main environment variable:
  2. export SURFEX\_EXPORT="\$HOME/MYDIR/EXPORT\_v7.1"
  3. then run a configuration script included in the archive:
  4. . \$SURFEX\_EXPORT/conf/profile\_surfex
5. run install\_surfex:  
if your linux distribution is ubuntu, don't run install\_surfex but install\_surfex\_ubuntu.  
install\_surfex / install\_surfex\_ubuntu:
  1. realizes the compilation of surfex sources
  2. prepares executable files for pgd, prep, offline and sxpost applications stored in \$SURFEX\_EXPORT/src/exe directory  
If everything goes well until this step, then surfex has been successfully installed on you computer.
6. How to install a pre-defined experiment:
  1. go to \$SURFEX\_EXPERIMENT/forcing directory
  2. run 1\_compile\_and\_link.bash script in order to prepare executable that will then be used to install an experiment.
  3. run 2\_prepare\_files.bash to know the available experiments
  4. run 2\_prepare\_files.bash with a name of experiment as argument:  
for example "2\_prepare\_files.bash hapex"  
a namelist MY\_PARAM.nam will open (vi editor), simply quit (use command :q)  
Some information will then be written on the screen and should look like:  
\$SURFEX\_EXPORT="/home/lemoine/surfex/EXPORT\_v7.1/"  
-- namelist NAM\_MY\_PARAM read  
> =====  
> PREP\_INPUT\_EXPERIMENT: YEXPER = HAPEX  
> PREP\_INPUT\_EXPERIMENT: INI = 1  
> PREP\_INPUT\_EXPERIMENT: INPTS = 17521

```

> PREP_INPUT_EXPERIMENT: JNPTS = 17521
> PREP_INPUT_EXPERIMENT: ZTSTEPFRC = 1800.
> PREP_INPUT_EXPERIMENT: YFORCING_FILETYPE = NETCDF
> =====
YFILE_FORCIN=./Data/hapex/HAPEX.DAT.30
-rw-r--r-- 1 lemoigne mc2 1543644 jui 22 16:51
/home/lemoigne/surfex/EXPORT_v7.0//EXP/forcing/FORCING.nc
-rw-r--r-- 1 lemoigne mc2 644 jui 22 16:51
/home/lemoigne/surfex/EXPORT_v7.0//EXP/forcing/PARAMS.nc
=====
> input files moved to /home/lemoigne/surfex/EXPORT_v7.0//EXP/rundir/hapex
=====

```

7. once the installation is done, go to \$SURFEX\_EXPERIMENT/rundir/hapex directory and launch successively:

1. pgd.exe
2. prep.exe
3. offline.exe

8. to view output 1d, you can use vespa tool. Type vespa to get available fields and then vespa FIELDNAME to plot FIELDNAME

9. How to rerun a pre-defined experiment with new inputs :

1. you can define new surface characteristics by modifying file \$SURFEX\_EXPERIMENT/rundir/hapex/OPTIONS.nam and then run pgd.exe, prep.exe and offline.exe
2. you can define new initial values for state variables by modifying file \$SURFEX\_EXPERIMENT/rundir/hapex/OPTIONS.nam and then run prep.exe and offline.exe
3. you can modify the forcing characteristics
  1. you can rerun \$SURFEX\_EXPERIMENT/forcing/2\_prepapre\_files.bash and modify namelist MY\_PARAM to select the number of time steps you want to treat (parameter NUMBER\_OF\_TIME\_STEPS\_FINAL) the format of the input forcing files (parameter YFORCING\_FILETYPE)
  2. then go to \$SURFEX\_EXPERIMENT/rundir/hapex and rerun pgd.exe, prep.exe and offline.exe

10. How to create a new experiment:

1. you need to modify \$SURFEX\_EXPERIMENT/forcing/src/my\_forcing.f90 to introduce the call to the new program that is going to read your dataset
2. you need to create a new subroutine named \$SURFEX\_EXPERIMENT/forcing/src/my\_forc\_xxxx.f90 that corresponds to experiment xxxx
3. go then to \$SURFEX\_EXPERIMENT/forcing and run successively :
  1. 1\_compile\_and\_link.bash (to account for your new subroutine)

2. 2\_prepare\_files.bash (to create input files related to your experiment)
3. then go to \$SURFEX\_EXPERIMENT/rundir/xxxx and run pgd.exe, prep.exe and offline.exe

11. How to compile your own source for surfex

1. cp the sources (from OFF\_LINE or SURFEX directories) that you want to modify onto \$SURFEX\_EXPORT/src/MYSRC
2. go to \$SURFEX\_EXPORT/src/MYSRC and make your modifications
3. go to \$SURFEX\_EXPORT/src and simply launch command "make" new executable files will be created in exe directory and will replace the old ones.

12. How to include the BUGFIX (when exists)

1. download BUGFIX\_EXPORT\_v7.1.gz
2. extract files from archive BUGFIX\_EXPORT\_v7.1.gz, a directory named MYSRC and containing bug fixes will be created
3. place extracted MYSRC in \$SURFEX\_EXPORT/src/ as replacement of the old one
4. go to \$SURFEX\_EXPORT/src and simply launch command "make" new executable files will be created in exe directory and will replace the old ones.

## 2. Overview of the externalized surface sequence

The externalized surface facilities do not contain only the program to run the physical surface schemes, but also those producing the initial surface fields (before the run) and the diagnostics (during or after the run). All these facilities are listed, below, and they separate in 4 main parts:

## 2.1. The sequence

1. **PGD** (routine `pgd_surf_atm.f90`): this program computes the physiographic data file (called PGD file below). At this step, you perform 3 main tasks:
  1. You choose the surface schemes you will use.
  2. You choose and define the grid for the surface
  3. The physiographic fields are defined on this grid.  
Therefore, the PGD file contains the spatial characteristics of the surface and all the physiographic data necessary to run the interactive surface schemes for vegetation and town.
2. **PREP** (routine `prep_surf_atm_n.f90`): this program performs the initialization of the surface scheme prognostic variables, as temperatures profiles, water and ice soil contents, interception reservoirs, snow reservoirs.
3. **run of the schemes** (routine `coupling_surf_atm_n.f90`): this performs the physical evolution of the surface schemes. It is necessary that this part, contrary to the 2 previous ones, is to be coupled within an atmospheric forcing (provided either in off-line mode or via a coupling with an atmospheric model).
4. **DIAG** (routine `diag_surf_atm_n.f90`): this computes diagnostics linked to the surface (e.g. surface energy balance terms, variables at 2m of height, etc...). It can be used either during the run (adding these diagnostics in the output file(s) of the run), or independantly from the run, for a given surface state (still, an instantaneous atmospheric forcing is necessary for this evaluation).

In addition, in order to read or write the prognostic variables or the diagnostics variables, respectively, in the surface files, the following subroutines are used: `init_surf_atm_n.f90`, `write_surf_atm_n.f90` and `write_diag_surf_atm_n.f90`.



## 2.2. The atmospheric models using the externalized surface

The externalized surface can presently be used in:

1. in offline mode
2. MESONH
3. AROME

For each model, additionnal possibilities of the surface, especially the ability to read and write in files with particular formats, are added:

## 2.2.1. In offline mode

In this case, several types of files can be used:

- *ASCII files*, not efficient in term of storage, but completely portable.
- *TEXTE files*, not efficient in term of storage, but completely portable.
- *netcdf files*, that can be used by the program code "OFFLIN".
- *BINARY files*, increases the efficiency of the system.
- *LFI files*, increases the efficiency of the system. This special format is used in meso-NH and Arôme models for surface fields.
- *FA files*, This special format is used for Arpege and Aladin models.

currently, **PGD** and **PREP** steps may be done using ASCII, LFI or FA files, and also the run produces time series of each variable (prognostic or diagnostic) in any of the formats listed above files and the output instant of the run in an ASCII, LFI or FA file.

The namelists are all included in the namelist file named `OPTIONS.nam`

## 2.2.1.1. namelist NAM\_IO\_OFFLINE

This namelist is the main namelist used in the off-line mode.

Fortran name	Fortran type	values	default value
CSURF_FILETYPE	string of 6 characters	"FA ", "ASCII ", "LFI "	"ASCII "
CTIMESERIES_FILETYPE	string of 6 characters	"NETCDF", "OFFLIN", "NONE ", "ASCII ", "TEXTE ", "BINARY", "FA ", "LFI "	"NONE "
CFORCING_FILETYPE	string of 6 characters	"NETCDF", "BINARY", "ASCII"	"NETCDF"
CPGDFILE	string of 28 characters		"PGD"
CPREPROFILE	string of 28 characters		"PREP"
CSURFFILE	string of 28 characters		"SURFOUT"
LPRINT	logical		.FALSE.
LRESTART	logical		.FALSE.
LINQUIRE	logical		.FALSE.
XTSTEP_SURF	real		300.
XTSTEP_OUTPUT	real		1800.
LSET_FORC_ZS	logical		.FALSE.
LWRITE_COORD	logical		.FALSE.
LOUT_TILENAME	logical		.FALSE.
LLIMIT_QAIR	logical		.FALSE.
NB_READ_FORC	integer		Number of forcing time steps
LLAND_USE	logical		.FALSE.

- CSURF\_FILETYPE: type of Surfex surface files created during PGD or PREP steps .
- CTIMESERIES\_FILETYPE: type of the files containing the output diagnostic time series .
- CFORCING\_FILETYPE: type of atmospheric forcing files .
- CPGDFILE: name of the PGD file .
- CPREPROFILE: name of the PREP file .
- CSURFFILE: name of the final output surfex file (restart file) .
- LPRINT: write information on screen during run .
- LRESTART: write restart file .
- LINQUIRE: enable test of inquiry mode .
- XTSTEP\_SURF: surface time step .
- XTSTEP\_OUTPUT: time step of the output time series .
- LSET\_FORC\_ZS: if T, the orography of the forcing file is set to the same value as in surface file .

- `LWRITE_COORD`: enables write of fields `XLAT` and `XLON` in output file .
- `LOUT_TIMENAME`: change name of output file at the end of the day .
- `LLIMIT_QAIR`: General flag for coherence between forcing `Qair` and calculated `Qsat(Tair)`.
- `NB_READ_FORC`: subdivisions of the reading of forcings. Can vary from 1 (all forcing data read in one time) to the total number of forcing time steps (what was done until now). It's usefull especially for netcdf forcing files on tori and yuki.
- `LLAND_USE`: if `LLAND_USE = .TRUE.`, fractions of vegtypes can be given at INIT level, by the namelist `NAM_LAND_USE`, and other surface parameters are calculated through `ECOCLIMAP`. It allows to make a restart with new fractions of vegtypes. But for the moment, the water balance is not kept in this case (it will be done in next version).

## 2.2.1.2. namelist NAM\_LAND\_USE

This namelist is needed when LLAND\_USE = .TRUE. (NAM\_IO\_OFFLINE). The file referenced in this namelist has to be formatted as a Surfex PREP file and to contain at least 13 record:

DIM\_FULL, VEGTYPE\_P1, .... VEGTYPE\_P12.

If CFTYP\_VEGTYPE = 'OFFLIN', the file is a NETCDF file and its name needs to be PARAMS.nc.

Fortran name	Fortran type	values	default value
CFNAM_VEGTYPE	character (LEN=28)	' '	
CFTYP_VEGTYPE	character (LEN=6)	'ASCII', 'FA ', 'LFI ', 'OFFLIN'	none

### 2.2.1.3. Namelist NAM\_ZS\_FILTER

Fortran name	Fortran type	values	default value
NZSFILTER	integer		1

- NZSFILTER : number of iterations of the spatial filter applied to smooth the orography (integer, 1 iteration removes the  $2\Delta x$  signal, 50% of the  $4\Delta x$  signal, 25% of the  $6\Delta x$  signal, etc) [1]

## 2.2.1.4. Namelist NAM\_NACVEG

declaration of keys for ISBA assimilation scheme (2DVAR, Bouyssel et al.)

Fortran name	Fortran type	values	default value
NECHGU	integer		6
RCLIMCA	real		0.
RCLISST	real		0.05
SIGH2M0	real		0.1
SIGT2M0	real		1.0
SIGWG0	real		
SIGWGB	real		0.06
SIGW2B	real		0.03
LOBSWG	logical		
LOBS2M	logical		F
LIMVEG	logical		T
SPRECIP2	real		4.0
RTHR_QC	real		3.0
SIGWG0_MAX	real		6.0
RSCAL_JAC	real		4.0
LPRINT	logical		T
LAROME	logical		T

## 2.2.2. in MESONH

In this case, MESONH FM files are used. The parallelization of the surface fields is done during the reading or writing of the fields by the FMREAD and FMWRIT routines.



## 2.2.2.1. Initialization of surface fields integrated in MESONH programs

In MESONH, there are usually 2 ways to produce initial files, depending if you want to use real or ideal atmospheric conditions. However, from the surface point of view, there is no difference between these 2 main possibilities of fields (real -e.g. from operational surface scheme in an operational model- or ideal -e.g. uniform-), whatever the treatment done for the atmospheric fields. This is allowed because the same externalized routines corresponding to PGD and PREP are used:

In the case of realistic atmospheric fields, the MESONH programs calling the surface are:

1. PREP\_PGD : it uses the **PGD** facility of the surface
2. PREP\_NEST\_PGD : surface fields are only read and rewritten, except the orography that is modified (the modification of the orography itself is considered as an atmospheric model routine, as orography is also a field of the atmospheric model).
3. PREP\_REAL\_CASE : it uses the **PREP** facility of the surface, that can produce either ideal or realistic surface fields.
4. SPAWNING : it does not produce surface fields any more. The surface fields will be recreated during the PREP\_REAL\_CASE step following the SPAWNING.

In the case of ideal atmospheric fields, the MESONH program calling the surface is PREP\_IDEAL\_CASE : it uses both the **PGD** and **PREP** facilities of the surface. Ideal or realistic (the latter only in conformal projection) physiographic fields can be either produced or read from a file. Then the prognostic surface variables, either ideal or realistic, can be computed by **PREP**.

If you use MESONH atmospheric model, the input and output surface files are the same as the atmospheric ones, so there is no need to specify via surface namelists any information about the input or output file names.

## 2.2.2.2. Namelist NAM\_PGDFILE

Note however that, in PREP\_PGDFILE (just before the call to the surface physiographic computation in PGDFILE, for which the namelists are described in the next chapter), there is a namelist to define the output physiographic file:

Fortran name	Fortran type
CPGDFILE	string of 28 characters

### **2.2.2.3. MESONH run and diagnostics**

Then, the MESONH run can be done. During this one, the diagnostics can be, or not, be computed.

In DIAG, the surface diagnostics can also be recomputed.

## 2.2.3. in AROME

In this case, MESONH FM files are also used, for the surface only. The parallelization of the surface fields is done during the reading or writing of the fields by parallelization routines of ALADIN atmospheric model.

# 3. Off-line Guide

## 3.1. The input files

The use of the externalized surface software in off-line mode requires the preparation of several types of file, especially the input data necessary for the run and the definition of the options specified in the namelist .

- **OPTIONS.nam** is the namelist name used in the off-line model. The same namelist is used for the PGD, PREP and RUN facilities. The description of the different namelist blocks for PGD ("The physiographic fields") and PREP ("Initialization of the prognostic fields") tools are described in the next chapter. The namelist block where functionalities of the off-line run in terms of Input/Output is named **NAM\_IO\_OFFLINE**
- the princip of an off-line simulation is that the atmospheric variables are known in advance. Thus, time series of air temperature, humidity, wind speed, precipitation, pressure and radiation terms are known. These data are pre-treated in order to be written in specific files (see below) called forcing files.
- like for any model, some parameters related to the scheme have to be set and state variable have to be initialized. These two tasks are performed by mean of tools PGD and PREP which lead to create the initial file used in the simulation

## 3.2. forcing files

## 3.2.1. creation of forcing files

For the preparation of forcing files, specific programs are used and are located in \$SURFEX\_EXPERIMENT/forcing/src. The structure of \$SURFEX\_EXPERIMENT directory looks like:

```
EXP
|-- Data
|   |-- Alp_for_0203
|   |-- Alqueva0206
|   |-- cdp9697
|   |-- hapex
|   |-- ma01
|   |-- me93
|   |-- v192
|-- forcing
|   |-- Comp
|   |-- src
|-- namelist
|   |-- Alp_for_0203
|   |-- Alqueva0206
|   |-- cdp9697
|   |-- hapex
|   |-- ma01
|   |-- me93
|   |-- v192
|-- rundir
    |-- hapex
```

- Data directory contains subdirectories (one per experiment) in which atmospheric time-series ascii files are stored
- namelist directory contains subdirectories (one per experiment) in which at least 2 namelists are stored: the first one named MY\_PARAM.nam contains information related to the forcing. For example for the "hapex" experiment, MY\_PARAM.nam looks like:

```
&NAM_MY_PARAM  YEXPERIMENT_NAME   = 'HAPEX'           ,
                NUMBER_GRID_CELLS  = 1                    ,
                NUMBER_OF_TIME_STEPS_INPUT = 17521           ,
                NUMBER_OF_TIME_STEPS_FINAL = 17521           ,
                ZATM_FORC_STEP       = 1800.                ,
                YFORCING_FILETYPE    = 'NETCDF'              /
```

- YEXPERIMENT\_NAME  
is the name associated to the experiment (12 characters)
- NUMBER\_GRID\_CELLS  
is the grid cell number
- NUMBER\_OF\_TIME\_STEPS\_INPUT  
number of time steps of forcing serie
- NUMBER\_OF\_TIME\_STEPS\_FINAL  
number of time steps used for the simulation (should be lower or equal to



- NUMBER\_OF\_TIME\_STEPS\_INPUT)
- ZATM\_FORC\_STEP  
frequency of atmospheric forcing
  - YFORCING\_FILETYPE  
is the type of the forcing files asked by the user:
    - ◆ **NETCDF**:  
1 file will be created: FORCING.nc
    - ◆ **ASCII or BINARY** : Params\_config.txt ascii file describing the configuration of the run will be created if forcing file type is ASCII or BINARY. The content of this file is :
      - ◇ number of grid cells of the domain
      - ◇ number of atmospheric time steps
      - ◇ atmospheric time step
      - ◇ year corresponding to the beginning of the simulation
      - ◇ month corresponding to the beginning of the simulation
      - ◇ day corresponding to the beginning of the simulation
      - ◇ seconds corresponding to the beginning of the simulation
      - ◇ longitudes of grid cells
      - ◇ latitudes of grid cells
      - ◇ elevation (meters) of grid cells
      - ◇ reference height for thermodynamical variables for each grid cell
      - ◇ reference height for wind for each grid cell
    - ◆ **ASCII** : 11 ascii files, one per parameter: Forc\_CO2.txt, Forc\_DIR.txt, Forc\_PS.txt, Forc\_RAIN.txt, Forc\_SNOW.txt, Forc\_WIND.txt, Forc\_DIR\_SW.txt, Forc\_LW.txt, Forc\_QA.txt, Forc\_SCA\_SW.txt, Forc\_TA.txt.
    - ◆ **BINARY** : 11 binary files, one per parameter: Forc\_CO2.bin, Forc\_DIR.bin, Forc\_PS.bin, Forc\_RAIN.bin, Forc\_SNOW.bin, Forc\_WIND.bin, Forc\_DIR\_SW.bin, Forc\_LW.bin, Forc\_QA.bin, Forc\_SCA\_SW.bin, Forc\_TA.bin.

All forcing files will be placed in \$SURFEX\_EXPERIMENT/rundir/hapex/ and a consistency test between MY\_PARAM.nam and OPTIONS.nam will be done in case the forcing filetype would be different.

## 3.2.2. installation of an experiment

Go to \$SURFEX\_EXPERIMENT/forcing directory and launch **1\_compile\_and\_link.bash** to create executable file that will then be used to prepare the files when you'll launch **2\_prepare\_files.bash** giving the experiment name as argument. If you want to create a new experiment named for example 'MYTEST', you'll have to modify \$SURFEX\_EXPERIMENT/forcing/src/my\_forcing.f90 program in order to refer to the new subroutine that you'll have created and that must be named my\_forc\_mytest.f90 and stored in \$SURFEX\_EXPERIMENT/forcing/src. You simply have to add few lines in my\_forcing.f90 program:

```
      CASE ( 'MYTEST      ' )
          CALL MY_FORC_MYTEST (HEXPER, KNI, KNPTS, PTSTEP_FORC,      &
                               KYEAR, KMONTH, KDAY, PTIME,          &
                               PLON, PLAT, PZS, PZREF, PUREF,       &
                               PTA, PQA, PPS, PWINDSPEED, PWINDDIR, &
                               PDIR_SW, PSCA_SW, PLW, PRAIN, PSNOW, PCO2 )
```

Then copy my\_forc\_hapex.f90 into my\_forc\_mytest.f90, replace HAPEX by MYTEST, refer to the correct input file and adapt the reading sequence.

Create \$SURFEX\_EXPERIMENT/namelist/mytest and namelist MY\_PARAM.nam and OPTIONS.nam inside this directory

When this is done, go to \$SURFEX\_EXPERIMENT/forcing and run **1\_compile\_and\_link.bash** and **2\_prepare\_files.bash mytest**

### **3.3. One example of off-line surfex application**

This example is based on the situation of the 25th of October 2004 at 06UTC and covers a temporal period of 24 hours. During this day an unstable weather was observed in France, especially in the Southern part.

Here, two different file formats are used as input for the externalized surface off-line software. Both are portable: the first format is netcdf and the second is the ascii one. Netcdf format has been chosen because of several participations of PILPS intercomparison projects that requires such format, due to its portability. It follows the Alma concept (proposed by Polcher in 1998).

## 3.3.1. netcdf format file

### **3.3.1.1. FORCING.nc**

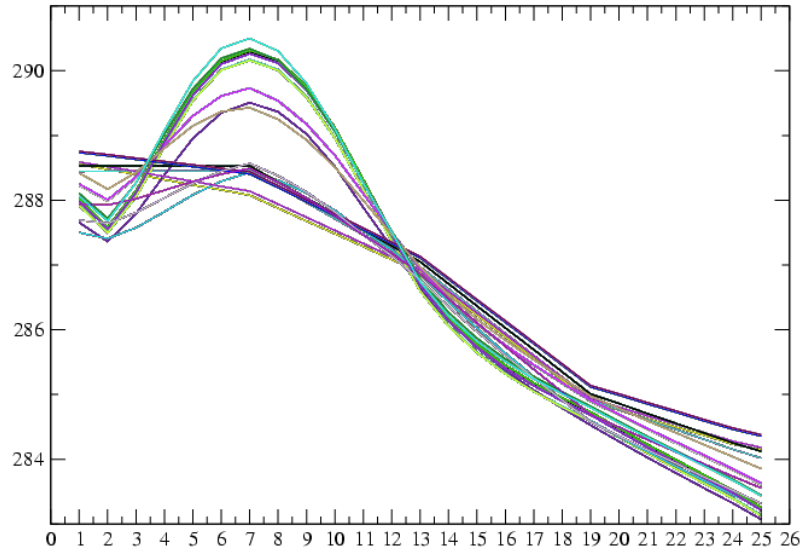
For this experiment, atmospheric forcing is extracted from French database named BDAP (Base de Donnees Analysees et Prevues). Data come from the analysis of surface parameters performed by Safran analysis system devoted to hydrological applications. A constant value in space is applied for each gridbox.

### 3.3.1.2. list of parameters

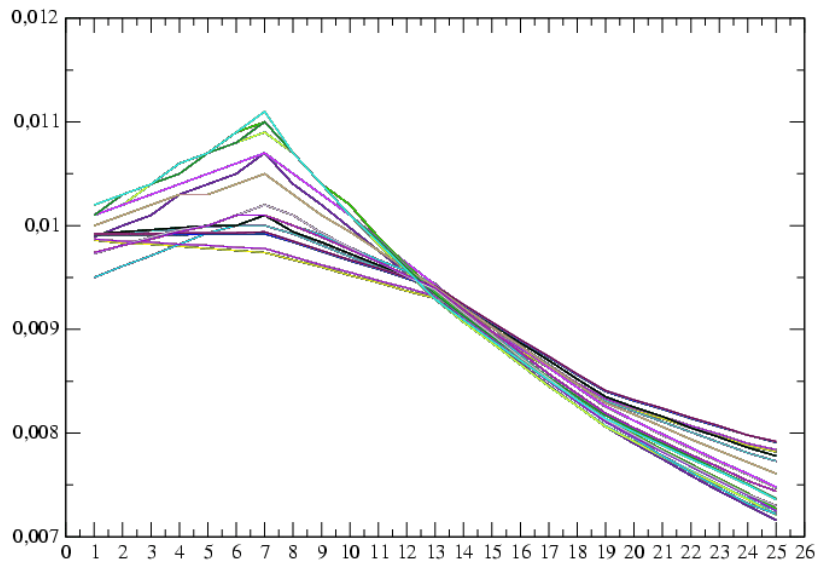
Variable name	Dimensions	Unit	Description
time	time	days/hours/minutes/seconds since YYYY-MM-DD HH:MM:SS	time
FORC_TIME_STEP		s	forcing time step
LAT	Number_of_points	degrees	latitudes
LON	Number_of_points	degrees	longitudes
UREF	Number_of_points	m	Reference_Height_for_Wind
ZREF	Number_of_points	m	Reference_Height
ZS	Number_of_points	m	surface orography
Tair	time, Number_of_points	K	air temperature
Qair	time, Number_of_points	Kg/Kg	air specific humidity
Wind	time, Number_of_points	m/s	wind speed
DIR_SWdown	time, Number_of_points	W/m2	downward direct shortwave radiation
SCA_SWdown	time, Number_of_points	w/m2	downward diffuse shortwave radiation
LWdown	time, Number_of_points	W/m2	downward longwave radiation
PSurf	time, Number_of_points	Pa	surface pressure
Rainf	time, Number_of_points	Kg/m2/s	rainfall rate
Snowf	time, Number_of_points	Kg/m2/s	snowfall rate
CO2air	time, Number_of_points	Kg/m3	CO2 concentration
Wind_DIR	time, Number_of_points	deg	wind direction

Dimensions **Number\_of\_points** and **time** represent respectively the total number of gridboxes in the area of interest and the number of atmospheric time steps.

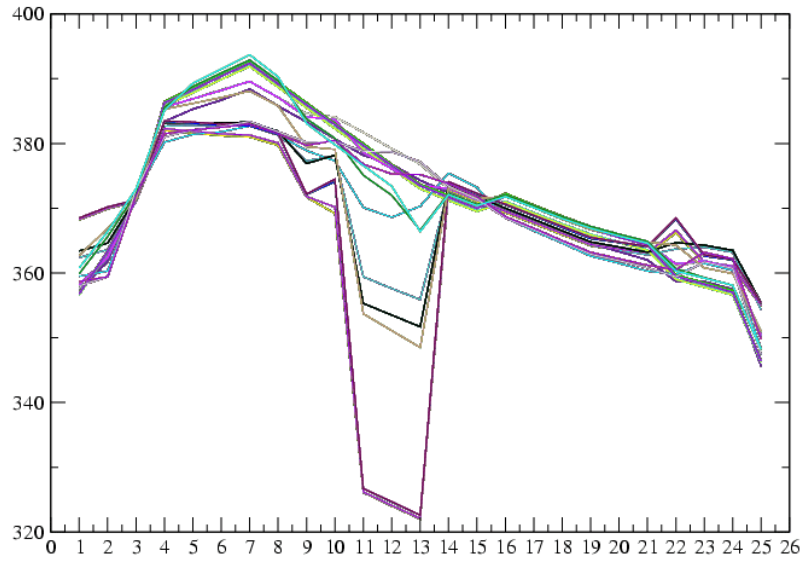
The following pictures show the time evolution of forcing quantities for the integration period, over the region of interest. Each curve correspond to a grid point.



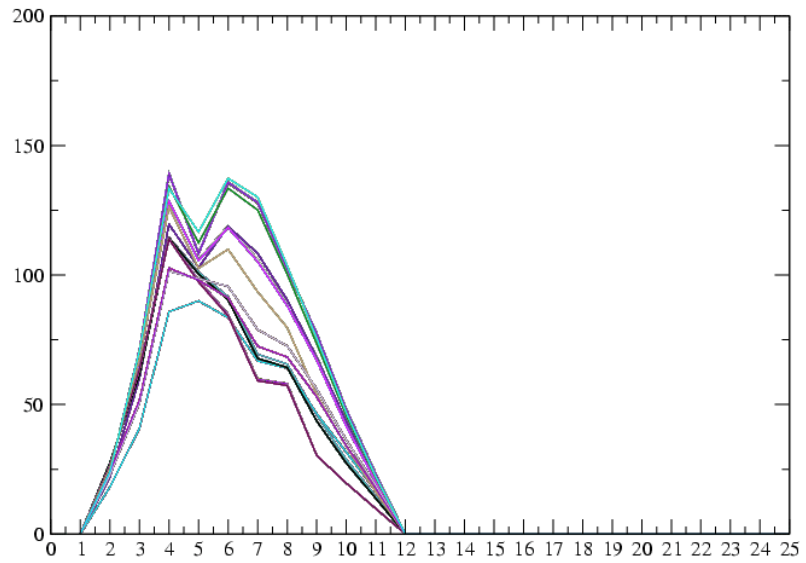
**Figure 3.1:** *Time evolution of temperature over the working area*



**Figure 3.2:** *Time evolution of specific humidity over the working area*



**Figure 3.3:** *Time evolution of longwave incoming radiation over the working area*



**Figure 3.4:** *Time evolution of shortwave incoming radiation over the working area*



## 3.3.2. ascii format files

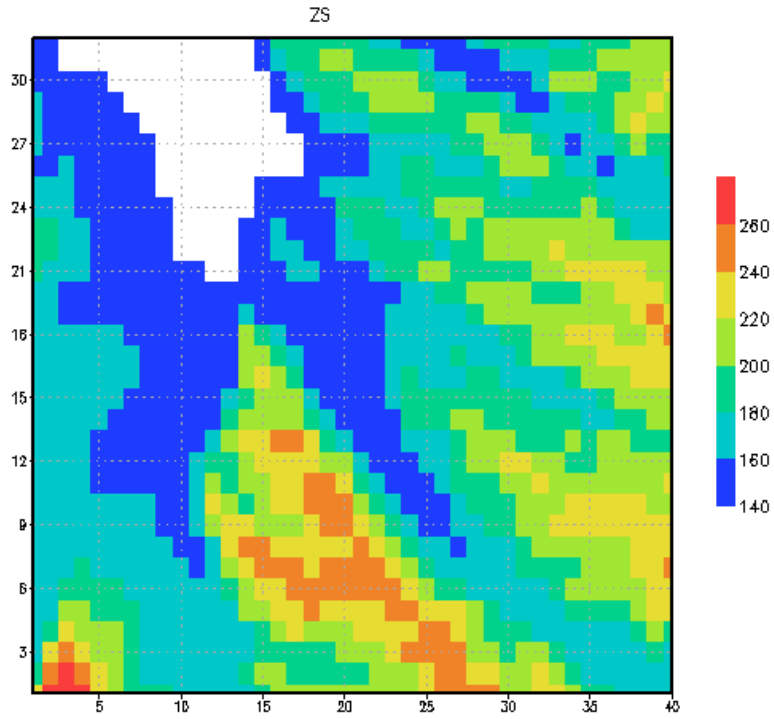
This format is used in the off-line applications because it's the simplest format that may replace more complex (in terms of file handling) like lfi format used in the meso-NH framework or FA used in the Arome framework.

### 3.3.2.1. PGD.txt

This file contains the information related to physiography and orography essentially. The file is splitted into several parts. The first one corresponds to the gridbox as seen as a single pixel where quantities are aggregated. The corresponding field names are prefixed with **FULL**. This gridbox may be separated into four tiles respectively associated to nature, town, sea/ocean and lake). The corresponding field names are respectively prefixed with **NATURE**, **TOWN**, **SEA** and **WATER**. The physiographic parameters written out into PGD.txt file are mainly the fraction of land covers contained in each gridbox. These fractions are computed from ECOCLIMAP database.

```
&FULL DIM_FULL
(-)
1280
&FULL DIM_SEA
(-)
0
&FULL DIM_NATURE
(-)
1279
&FULL DIM_WATER
(-)
1
&FULL DIM_TOWN
(-)
240
&FULL ECOCLIMAP
(-)
T
&NATURE ISBA
ISBA
3-L
&NATURE PHOTO
PHOTO
NON
&NATURE GROUND_LAYER
GROUND_LAYER
3
&NATURE PATCH_NUMBER
PATCH_NUMBER
1
```

Value of **DIM\_FULL** indicates that this is a 2D exercise, and the gridbox contains a non-zero fraction of nature, of water (lake) and of town, but there's no fraction of sea because the number of points with a non-zero fraction of sea **DIM\_SEA** is zero. 1279 gridboxes contain vegetation and 240 gridboxes contain a fraction of town, onmly one gridbow contains a fraction ok inland water(lake). Surface scheme to treat vegetation is Isba 3-L which means that soil is represented with 3 layers. The number of patches is 1, it indicates that the vegetation is not splitted into patches like it could (this should be the case for the A-gs option of Isba that treats explicitly photosynthesis).



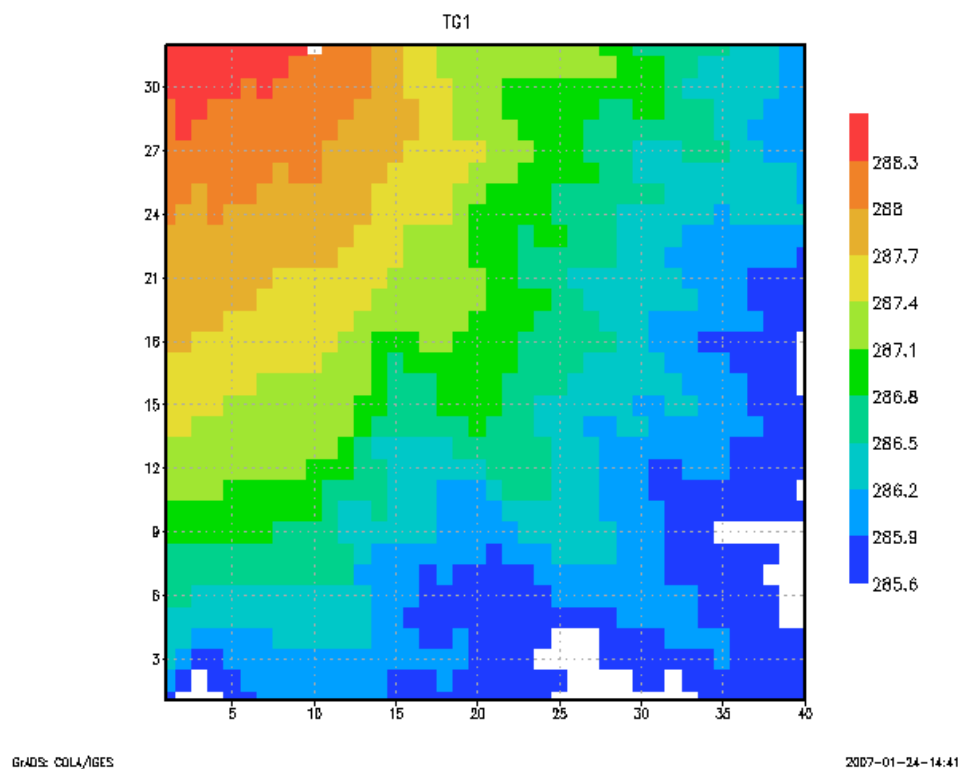
**Figure 3.5:** *Orography field taken from PGD.txt file*

## 3.3.2.2. PREP.txt

This file contains the information related to physiography and orography read from PGD.txt file and rewritten, as well as initial values of prognostic variables of the different schemes contained in SURFEX.

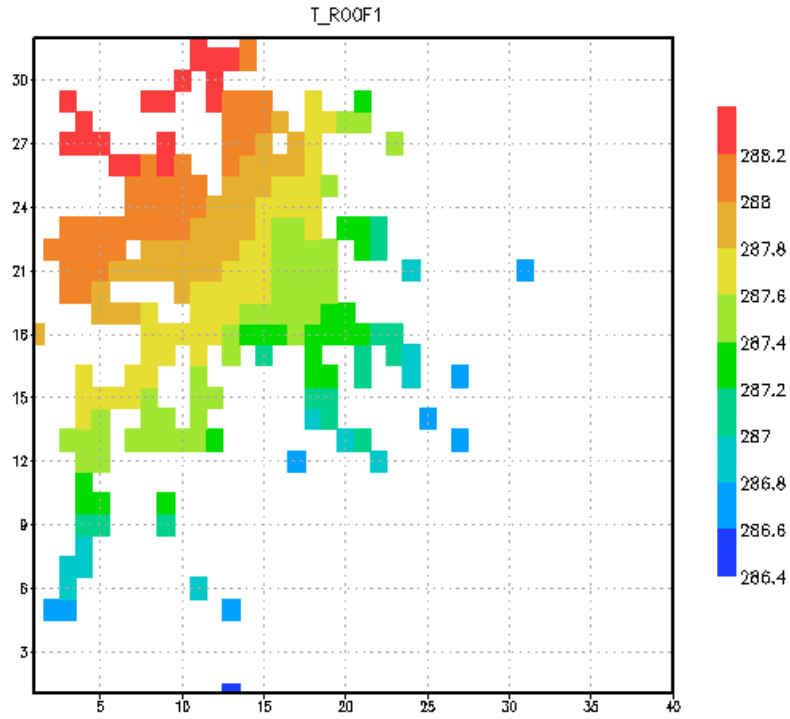
```
&NATURE TG1
X_Y_TG1 (K)
  2.8576409563069382E+02  2.8548082006251650E+02  2.8540527530138650E+02  2.8546873
```

For example here are shown the first initial values of skin surface temperature over natural area for Isba surface scheme as they are written in PREP.txt file. This field is represented on figure 3.6.



**Figure 3.6:** Initial surface temperature for vegetation taken from PREP.txt file

An other example shows the roof surface temperature over the working area (Toulouse city is located roughly at  $x=15$ ,  $y=22$ )



GRADS: COLA/IGES

2007-01-24-14:58

**Figure 3.7:** Initial surface temperature for the roofs taken from PREP.txt file

### 3.3.2.3. Extracting 2d fields

It's possible to extract 2d fields from PGD.txt (covers, orography, etc.) and from PREP.txt (initial prognostic variables like soil temperature profile, soil water content profile, etc.). For that purpose you need to run SXPOST tool (located at the same place as PGD, PREP and OFFLINE: `$SURFEX_EXPORT/src/exe/`). Input files for SXPOST are PGD.txt or PREP.txt if exists, and a namelist containing the number of fields to be extracted, the name and a flag indicating if the variable depends on patches or not. The name of a given field is the name written in PGD.txt or PREP.txt file where characters " SPMamp;" have been removed. For example, to extract orography, the name of the field is ZS in SURFEX, the mask over which it's defined can be FULL (total gridbox). To extract surface temperature over vegetation, the name will be TG1 and the mask NATURE. (`grep "&" PGD.txt` returns all variables of this file). To distinguish variables defined or not over patches, a flag is used: if the variable is patch dependant, the flag must be set to '+', in the contrary, it should be '-'. For example orography (ZS) doesn't depend on patches, but surface temperature (TG1) does. If the simulation uses patches and the flag is '-' then only the first patch will be treated (bare ground).

The namelist SXPOST.nam looks like:

```
2
- FULL    ZS
+ NATURE  TG1
```

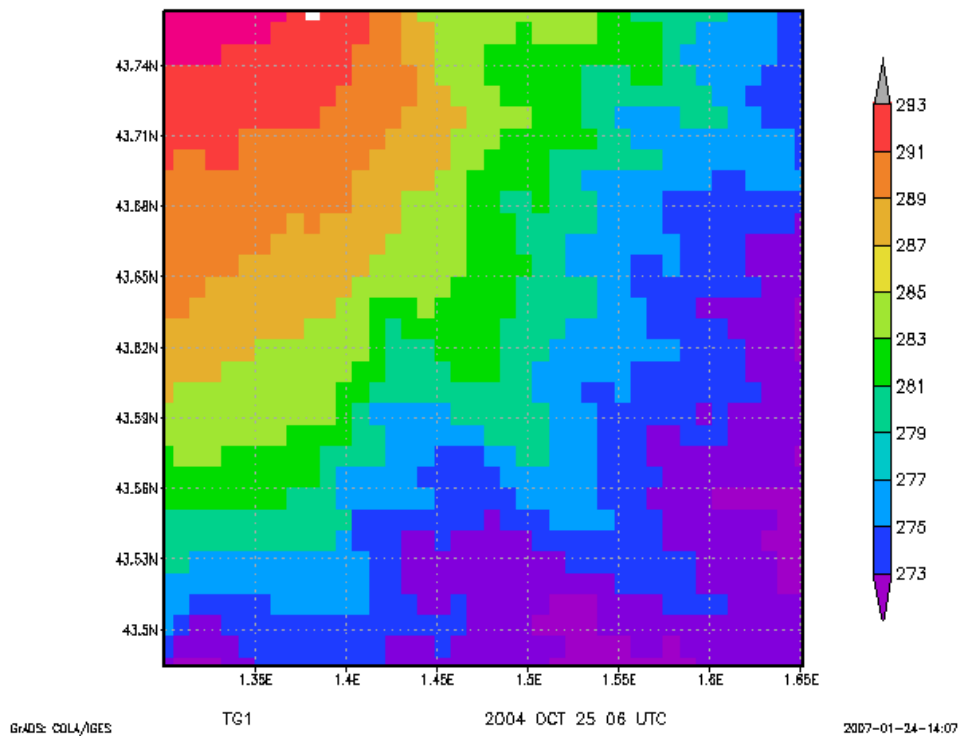
Running SXPOST will return a file per variable, which will contain the longitude, the latitude and the value of the field for each gridbox over which the field is defined (For example, TG1 which is known only over nature won't have a value for each gridbox of the domain).



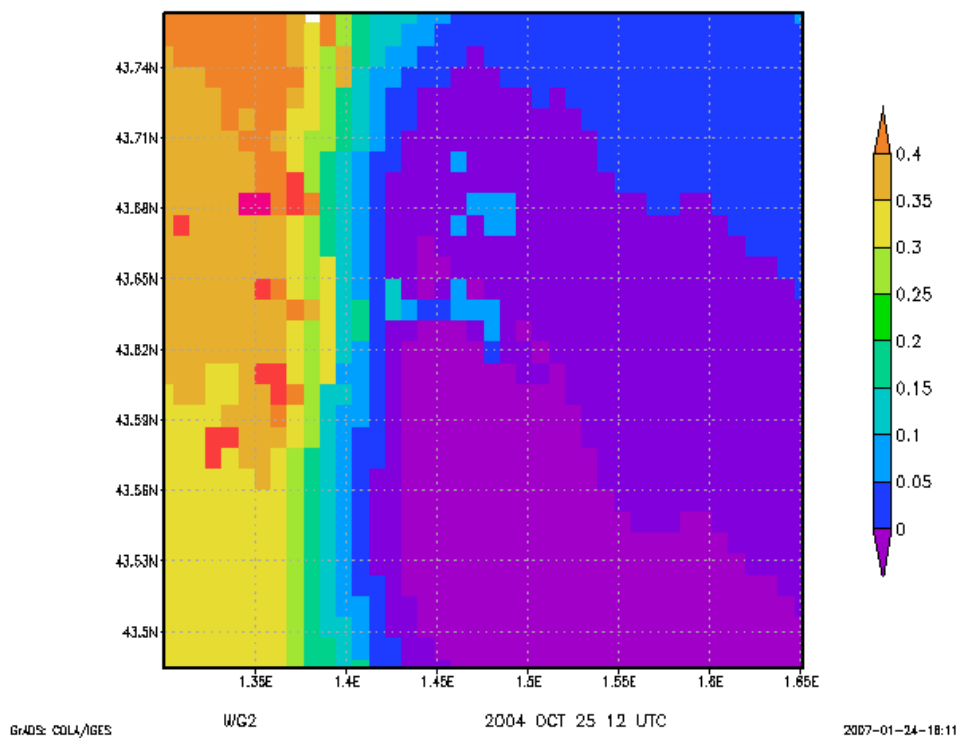
## 3.4. Some output of off-line simulation



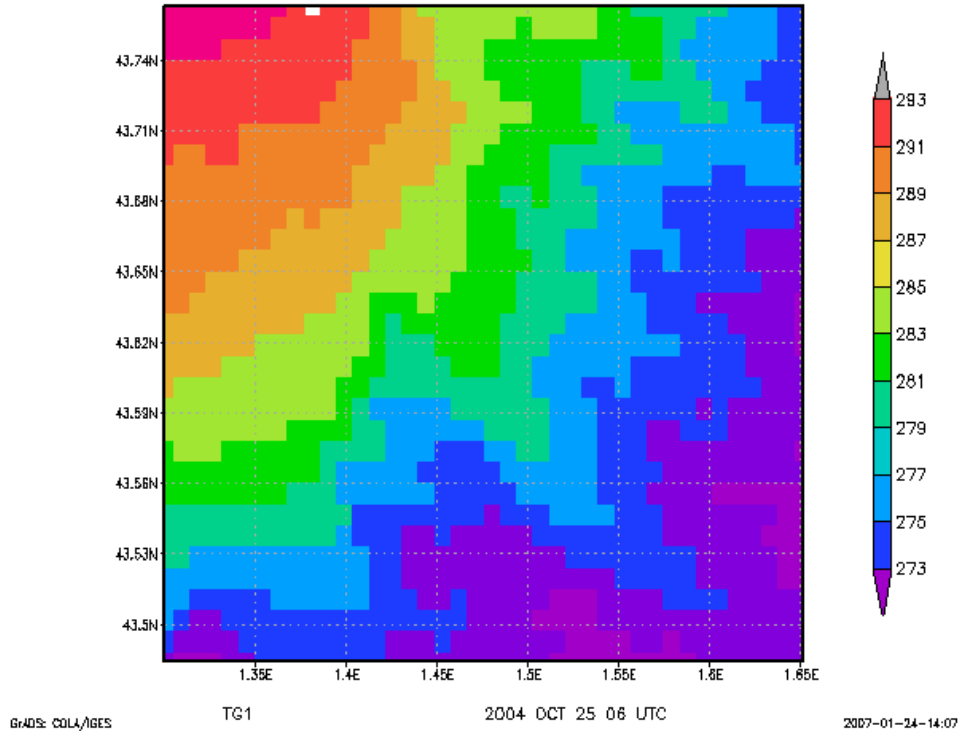
### 3.4.1. Examples of prognostic variables output



**Figure 3.9:** *Initial surface temperature field*



**Figure 3.10:** *Root layer water content after 12 hours of integration*



**Figure 3.11:** *Interception (by vegetation) water content after 12 hours of integration*

## 3.4.2. list of available variables

This list has been made by using vespa which is a tool that has been used to realize the plots of this document. This is not the complete possible list since only the fields that are present in netcdf output files are listed.

```
*****
***** V E S P A *****
*****
***
***      V-aluable      ***
***      E-externalized  ***
***      S-urface      ***
***      P-plot         ***
***      A-pplication   ***
***
*****
*****
```

### ISBA PROGNOSTIC VARIABLES

=====

```
TG1: "Soil_temp_layer_1" ;
WG1: "Soil_liquid_layer_1" ;
WGI1: "Soil_ice1" ;
TG2: "Soil_temp_layer_2" ;
WG2: "Soil_liquid_layer_2" ;
WGI2: "Soil_ice2" ;
TG3: "Soil_temp_layer_3" ;
WG3: "Soil_liquid_layer_3" ;
WGI3: "Soil_ice3" ;
WR: "Interception_reservoir" ;
RESA: "Aerodynamic_resistance" ;
WSNOW_VEG1: "Snow_Water_Equivalent_layer_1" ;
RSNOW_VEG1: "Snow_density_layer_1" ;
TSNOW_VEG1: "Snow_temperature_layer1" ;
HSNOW_VEG1: "Snow_heat_layer1" ;
WSNOW_VEG2: "Snow_Water_Equivalent_layer_2" ;
RSNOW_VEG2: "Snow_density_layer_2" ;
TSNOW_VEG2: "Snow_temperature_layer2" ;
HSNOW_VEG2: "Snow_heat_layer2" ;
WSNOW_VEG3: "Snow_Water_Equivalent_layer_3" ;
RSNOW_VEG3: "Snow_density_layer_3" ;
TSNOW_VEG3: "Snow_temperature_layer3" ;
HSNOW_VEG3: "Snow_heat_layer3" ;
ASNOW_VEG: "Snow_albedo" ;
```

### ISBA DIAGNOSTIC VARIABLES

=====

```
RI_ISBA: "Averaged_Richardson_Number" ;
```

```

RN_ISBA: "Averaged_Net_Radiation" ;
H_ISBA: "Averaged_Sensible_Heat_Flux" ;
LE_ISBA: "Averaged_Latent_Heat_Flux" ;
GFLUX_ISBA: "Averaged_Ground_Heat_Flux" ;
LEG: "Ground_Evaporation_Heat_Flux" ;
LEGI: "Soil_Ice_Sublimation" ;
LEV: "Vegetation_Evaporation_Heat_Flux" ;
LES: "Snow_Evaporation_Heat_Flux" ;
LER: "Canopy_Water_Interception_Evaporation" ;
LETR: "Vegetation_Evapotranspiration" ;
EVAP: "Evapotranspiration" ;
DRAIN: "Soil_Drainage_Flux" ;
RUNOFF: "Supersaturation_Runoff" ;
LEG_ISBA: "Averaged_Ground_Evaporation_Heat_Flux" ;
LEGI_ISBA: "Averaged_Soil_Ice_Sublimation" ;
LEV_ISBA: "Averaged_Vegetation_Evaporation_Heat_Flux" ;
LES_ISBA: "Averaged_Snow_Evaporation_Heat_Flux" ;
LER_ISBA: "Averaged_Canopy_Water_Interception_Evaporation" ;
LETR_ISBA: "Averaged_Vegetation_Evapotranspiration" ;
EVAP_ISBA: "Averaged_Evapotranspiration" ;
DRAIN_ISBA: "Averaged_Soil_Drainage_Flux" ;
RUNOFF_ISBA: "Averaged_Supersaturation_Runoff" ;
CH_ISBA: "Averaged_thermal_diffusion_coefficient" ;
HV_ISBA: "Halstead_coefficient" ;
Z0REL: "Output_Z0REL" ;
VEGTYPE_PATCH_1: "fraction_of_vegetation_type_1" ;
VEGTYPE_PATCH_2: "fraction_of_vegetation_type_2" ;
VEGTYPE_PATCH_3: "fraction_of_vegetation_type_3" ;
VEGTYPE_PATCH_4: "fraction_of_vegetation_type_4" ;
VEGTYPE_PATCH_5: "fraction_of_vegetation_type_5" ;
VEGTYPE_PATCH_6: "fraction_of_vegetation_type_6" ;
VEGTYPE_PATCH_7: "fraction_of_vegetation_type_7" ;
VEGTYPE_PATCH_8: "fraction_of_vegetation_type_8" ;
VEGTYPE_PATCH_9: "fraction_of_vegetation_type_9" ;
VEGTYPE_PATCH_10: "fraction_of_vegetation_type_10" ;
VEGTYPE_PATCH_11: "fraction_of_vegetation_type_11" ;
VEGTYPE_PATCH_12: "fraction_of_vegetation_type_12" ;

```

#### ISBA PHYSIOGRAPHIC VARIABLES

=====

```

VEG: "Output_vegetation_fraction" ;
Z0_ISBA: "Output_Z0_ISBA" ;
LAI: "Output_LAI_ISBA" ;
ALBNIR_SOIL: "Output_ALBNIR_SOIL" ;
ALBVIS_SOIL: "Output_ALBVIS_SOIL" ;

```

#### TEB PROGNOSTIC VARIABLES

=====

```

T_ROOF1: "Roof_Temperature_Layer_1" ;
T_ROOF2: "Roof_Temperature_Layer_2" ;
T_ROOF3: "Roof_Temperature_Layer_3" ;
T_ROAD1: "Road_Temperature_Layer_1" ;

```

```

T_ROAD2: "Road_Temperature_Layer_2" ;
T_ROAD3: "Road_Temperature_Layer_3" ;
T_WALL1: "Wall_Temperature_Layer_1" ;
T_WALL2: "Wall_Temperature_Layer_2" ;
T_WALL3: "Wall_Temperature_Layer_3" ;
TI_BLD: "Internal_Building_Temperature" ;
TI_ROAD: "Deep_Road_Temperature" ;
WS_ROOF1: "Roof_Water_Content_Layer_1" ;
WS_ROOF2: "Roof_Water_Content_Layer_2" ;
WS_ROOF3: "Roof_Water_Content_Layer_3" ;
WS_ROAD1: "Road_Water_Content_Layer_1" ;
WS_ROAD2: "Road_Water_Content_Layer_2" ;
WS_ROAD3: "Road_Water_Content_Layer_3" ;
T_CANYON: "Canyon_Air_Temperature" ;
Q_CANYON: "Canyon_Air_Humidity" ;

```

TEB DIAGNOSTIC VARIABLES

=====

```

RI_TEB: "Averaged_Richardson_Number" ;
CD_TEB: "Averaged_Drag_Momentum_Coef" ;
CDN_TEB: "Averaged_Neutral_Drag_Coef" ;
CH_TEB: "Averaged_Drag_Thermal_Coef" ;
RESA_TEB: "Averaged_Aerodyn_Resistance" ;
RN_TEB: "Averaged_Net_Radiation" ;
H_TEB: "Averaged_Sensible_Heat_Flux" ;
LE_TEB: "Averaged_Latent_Heat_Flux" ;
GFLUX_TEB: "Averaged_Ground_Heat_Flux" ;

```

WATER PROGNOSTIC VARIABLES

=====

```

TS_WATER: "Averaged_Water_Temperature" ;
Z0_WATER: "Roughness length" ;

```

WATER DIAGNOSTIC VARIABLES

=====

```

RI_WAT: "Averaged_Richardson_Number" ;
CD_WAT: "Averaged_Drag_Momentum_Coef" ;
CDN_WAT: "Averaged_Neutral_Drag_Coef" ;
CH_WAT: "Averaged_Drag_Thermal_Coef" ;
RESA_WAT: "Averaged_Aerodyn_Resistance" ;
RN_WAT: "Averaged_Net_Radiation" ;
H_WAT: "Averaged_Sensible_Heat_Flux" ;
LE_WAT: "Averaged_Latent_Heat_Flux" ;
GFLUX_WAT: "Averaged_Ground_Heat_Flux" ;

```

SURF\_ATM DIAGNOSTICS VARIABLES

=====

```

RI: "Averaged_Richardson_Number" ;

```

```
RN: "Averaged_Net_Radiation" ;
H: "Averaged_Sensible_Heat_Flux" ;
LE: "Averaged_Latent_Heat_Flux" ;
GFLUX: "Averaged_Ground_Heat_Flux" ;
```

SURF\_ATM FRACTIONS

=====

```
FRAC_SEA: "Fraction_of_sea" ;
FRAC_WATER: "Fraction_of_water" ;
FRAC_TOWN: "Fraction_of_town" ;
FRAC_NATURE: "Fraction_of_nature" ;
```

FORCING FIELDS

=====

```
TA: "air temperature" ;
QA: "air specific humidity" ;
WIND: "wind speed" ;
DIR_SW: "downward direct shortwave radiation" ;
SCA_SW: "downward diffuse shortwave radiation" ;
LW: "downward longwave radiation" ;
PS: "surface pressure" ;
RAIN: "rainfall rate" ;
SNOW: "snowfall rate" ;
CO2: "CO2 concentration" ;
DIR_SW: "downward direct shortwave radiation" ;
DIR: "wind direction" ;
```

## 4. The physiographic fields

## 4.1. Overview of physiographic fields computation: PGD

The physiographic fields are averaged or interpolated on the specified grid by the program **PGD**. They are stored in a file, called PGD file, but only with the physiographic 2D fields, the geographic and grid data written in it.

During the PGD facility :

1. You choose the surface schemes you will use.
2. You choose and define the grid for the surface.
3. The physiographic fields are defined on this grid.



## 4.1.1. Choice of the grid

There are 3 possibilities. 2 are always possible, one is available only if the **PGD** routine is integrated into an atmospheric model initialization facility.

1. The grid is chosen via namelists options (see below)
2. The grid is defined as a part of the grid of an already existing surface file, indicated via namelists (see below)
3. The grid is defined as being identical to the one of an atmospheric model, which is given as fortran argument in the coupling of the **PGD** surface facilities (routine PGD\_SURF\_ATM) into an atmospheric model initialization procedures. In this case, **all namelists that are usually used to define the surface grid are ignored**. Note that, in addition to the grid, the orography can also be given from the atmospheric file.

## 4.1.2. Choice of the physiographic fields

There are 3 main possibilities depending on LECOCLIMAP flag.

## 4.1.2.1. Namelist NAM\_FRAC

This namelist defines if ECOCLIMAP mechanism based on fractions of covers will be used or not.

Fortran name	Fortran type	values	default value
LECOCLIMAP	Logical		.TRUE.
XUNIF_SEA	real	between 0 and 1	none
CFNAM_SEA	character (LEN=28)		''
CFTYP_SEA	character (LEN=6)	'DIRECT', 'BINLLF'	none
		'BINLLV', 'ASCLLV'	
XUNIF_WATER	real	between 0 and 1	none
CFNAM_WATER	character (LEN=28)		''
CFTYP_WATER	character (LEN=6)	'DIRECT', 'BINLLF'	none
		'BINLLV', 'ASCLLV'	
XUNIF_NATURE	real	between 0 and 1	none
CFNAM_NATURE	character (LEN=28)		''
CFTYP_NATURE	character (LEN=6)	'DIRECT', 'BINLLF'	none
		'BINLLV', 'ASCLLV'	
XUNIF_TOWN	real	between 0 and 1	none
CFNAM_TOWN	character (LEN=28)		''
CFTYP_TOWN	character (LEN=6)	'DIRECT', 'BINLLF'	none
		'BINLLV', 'ASCLLV'	

- XUNIF\_SEA : uniform prescribed value of sea fraction. If XUNIF\_SEA is set, file CFNAM\_SEA is not used.
- CFNAM\_SEA: sea fraction data file name. If XUNIF\_SEA is set, file CFNAM\_SEA is not used.
- CFTYP\_SEA: type of sea data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_WATER : uniform prescribed value of water fraction. If XUNIF\_WATER is set, file CFNAM\_WATER is not used.
- CFNAM\_WATER: water fraction data file name. If XUNIF\_WATER is set, file CFNAM\_WATER is not used.
- CFTYP\_WATER: type of water data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_NATURE : uniform prescribed value of nature fraction. If XUNIF\_NATURE is set, file CFNAM\_NATURE is not used.
- CFNAM\_NATURE: nature fraction data file name. If XUNIF\_NATURE is set, file CFNAM\_NATURE is not used.
- CFTYP\_NATURE: type of nature data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_TOWN : uniform prescribed value of town fraction. If XUNIF\_TOWN is set, file CFNAM\_TOWN is not used.
- CFNAM\_TOWN: town fraction data file name. If XUNIF\_TOWN is set, file CFNAM\_TOWN is not used.

- CFTYP\_TOWN: type of town data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')

## **4.1.2.2. ideal physiographic fields:**

These fields are either uniform (fraction of each 215 ecoclimap ecosystem, orography, or any field needed by the surface schemes). As mentioned above, orography can be, in the case of the coupling with an atmospheric model, imposed as the atmospheric model (non-uniform) field.

### 4.1.2.3. realistic physiographic fields:

**PGD** can use files to build accurate physiographic fields from geographical information. This is possible only if the grid chosen can be linked to geographical coordinates (latitude and longitude), i.e. if the grid type is "LATLONREG " or "CONF PROJ ".

The files that can be used are :

- A file describing the type of cover of the surface. This describes where are located the different cover types (forests, towns, seas, etc...). At the time being, the file provided contains the ecoclimap data (215 land covers) on the world, with a resolution of 30". **PGD** computes the fraction of surface coverage occupied by each type in the grid mesh. From this information, the surface parameter convenient for the surface schemes (such as building fraction, leaf area index, etc..) are deduced, using correspondance arrays: **a parameter has always the same value for a given cover type, anywhere in the world.**
- A file containing the orography: GTOPO30. The resolution of the file is 30" on the world. This allows to compute the model orography, and the following subgrid-scale orographic characteristics:
  - ◆ the surface of frontal obstacle (A) over the surface of the grid mesh (S) in each direction (  $\frac{\sum A_{i+}}{S}$ ,  $\frac{\sum A_{i-}}{S}$ ,  $\frac{\sum A_{j+}}{S}$ ,  $\frac{\sum A_{j-}}{S}$  ), used to compute the directional  $z_{0_{eff}}$ ,
  - ◆ the half height of these obstacles (  $h_i^+/2$ ,  $h_j^-/2$ ,  $h_i^-/2$ ,  $h_j^+/2$  ), used to compute the directional  $z_{0_{eff}}$ ,
  - ◆ These 8 parameters are used to compute the total roughness length in the four directions given by the model axis (  $z_{0_{eff}}^+$ ,  $z_{0_{eff}}^-$ ,  $z_{0_{eff}}^+$ ,  $z_{0_{eff}}^-$  ),
  - ◆ the Subgrid-Scale Orography (SSO) parameters (standard deviation  $\frac{\mu_{z_x}}$ , anisotropy  $\frac{\gamma_{z_x}}$ , direction of the small main axis  $\theta_{z_x}$  and slope  $\frac{\sigma_{z_x}}$ ).
- For ISBA scheme, a file with the clay fraction of the (near-surface) soil. The resolution of the file provided is 5' on the world.
- For ISBA scheme, a file with the sand fraction of the (near-surface) soil. The resolution of the file provided is 5' on the world.

## **4.1.2.4. user defined physiographic fields:**

## 4.1.2.4.1. ISBA scheme

Over natural areas, all surface parameters for each vegtype, at a given frequency have to be specified by the user in namelist **NAM\_DATA\_ISBA**.

If LECOCLIMAP = .TRUE. (NAM\_FRAC), only part of the surface parameters for each vegtype can be given in NAM\_DATA\_ISBA. They are then completed by ECOCLIMAP data.

If only data for some of the 12 vegtypes are given, other vegtypes are filled with the values of the first given vegtype placed before in the list of 12.

parameters depending on the number of vegetation types:

Fortran name	Fortran type	values	default value	description	unit
XUNIF_VEGTYPE	real	between 0 and 1	none	vegetation type	
CFNAM_VEGTYPE	character (LEN=28)			file name	
CFTYP_VEGTYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none	file type	

parameters depending on the number of vegtypes and time:

Fortran name	Fortran type	values	default value	description	unit
NTIME	integer	1, 2, 12 or 36	36	time dimension	
XUNIF_VEG	real	between 0 and 1	none	vegetation fraction	(-)
CFNAM_VEG	character (LEN=28)		' '	file name	
CFTYP_VEG	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none	file type	
XUNIF_LAI	real		none	leaf area index	( $m^2/m^2$ )
CFNAM_LAI	character (LEN=28)		' '	file name	
CFTYP_LAI	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none	file type	
XUNIF_Z0	real		none	roughness length	( $m$ )
CFNAM_Z0	character (LEN=28)		' '	file name	
CFTYP_Z0	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none	file type	
XUNIF_EMIS	real		none	emissivity	(-)



CFNAM_EMIS	character (LEN=28)		' '	file name	
CFTYP_EMIS	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none	file type	

parameters depending on the number of vegtypes and soil levels:

Fortran name	Fortran type	values	default value	description	unit
XUNIF_DG	real		none	soil layer thickness	( <i>m</i> )
CFNAM_DG	character (LEN=28)		' '		
CFTYP_DG	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_ROOTFRAC	real		none	root fraction	(-)
CFNAM_ROOTFRAC	character (LEN=28)		' '		
CFTYP_ROOTFRAC	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		

parameters depending on number of vegtypes only:

Fortran name	Fortran type	values	default value	description	unit
XUNIF_RSMIN	real		none	minimal stomatal resistance	( <i>s/m</i> )
CFNAM_RSMIN	character (LEN=28)		' '		
CFTYP_RSMIN	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_GAMMA	real		none	coefficient used in the computation of RSMIN	(-)
CFNAM_GAMMA	character (LEN=28)		' '		
CFTYP_GAMMA	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_WRMALX_CF	real		none	coefficient for maximum interception water storage capacity	(-)
CFNAM_WRMALX_CF	character (LEN=28)		' '		
CFTYP_WRMALX_CF	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		

XUNIF_RGL	real		none	maximum solar radiation available for photosynthesis	( $W/m^2$ )
CFNAM_RGL	character (LEN=28)		' '		
CFTYP_RGL	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_CV	real		none	vegetation thermal inertia coefficient	( $K m^2/J$ )
CFNAM_CV	character (LEN=28)		' '		
CFTYP_CV	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_Z0_O_Z0H	real		none	ratio of surface roughness lengths	(-)
CFNAM_Z0_O_Z0H	character (LEN=28)		' '		
CFTYP_Z0_O_Z0H	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_ALBNIR_VEG	real		none	vegetation near-infra-red albedo	(-)
CFNAM_ALBNIR_VEG	character (LEN=28)		' '		
CFTYP_ALBNIR_VEG	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_ALBVIS_VEG	real		none	vegetation visible albedo	(-)
CFNAM_ALBVIS_VEG	character (LEN=28)		' '		
CFTYP_ALBVIS_VEG	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_ALBUV_VEG	real		none	vegetation UV albedo	(-)
CFNAM_ALBUV_VEG	character (LEN=28)		' '		
CFTYP_ALBUV_VEG	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_ALBNIR_SOIL	real		none	soil near-infra-red albedo	(-)
CFNAM_ALBNIR_SOIL	character (LEN=28)		' '		
CFTYP_ALBNIR_SOIL	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		

XUNIF_ALBVIS_SOIL	real		none	soil visible albedo	(-)
CFNAM_ALBVIS_SOIL	character (LEN=28)		''		
CFTYP_ALBVIS_SOIL	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_ALBUV_SOIL	real		none	soil UV albedo	(-)
CFNAM_ALBUV_SOIL	character (LEN=28)		''		
CFTYP_ALBUV_SOIL	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		

Isba-A-gs parameters depending on number of vegtypes only:

Fortran name	Fortran type	values	default	description	unit
XUNIF_GMES	real		none	mesophyll conductance	( $m/s^{-1}$ )
CFNAM_GMES	character (LEN=28)		''		
CFTYP_GMES	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_BSLAI	real		none	ratio d(biomass)/d(lai)	( $kg/m^2$ )
CFNAM_BSLAI	character (LEN=28)		''		
CFTYP_BSLAI	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_LAIMIN	real		none	minimum LAI	( $m^2/m^2$ )
CFNAM_LAIMIN	character (LEN=28)		''		
CFTYP_LAIMIN	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_SEFOLD	real		none	e-folding time for senescence	(s)
CFNAM_SEFOLD	character (LEN=28)		''		
CFTYP_SEFOLD	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_GC	real		none	cuticular conductance	( $m/s$ )
CFNAM_GC	character (LEN=28)		''		

CFTYP_GC	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_DMAX	real		none	maximum air saturation deficit	(kg/kg)
CFNAM_DMAX	character (LEN=28)		''		
CFTYP_DMAX	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_F2I	real		none	critical normlized soil water content for stress parameterization	(-)
CFNAM_F2I	character (LEN=28)		''		
CFTYP_F2I	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_H_TREE	real		none	height of trees	(m)
CFNAM_H_TREE	character (LEN=28)		''		
CFTYP_H_TREE	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_RE25	real		none	Ecosystem respiration parameter	(kg/kgms <sup>-1</sup> )
CFNAM_RE25	character (LEN=28)		''		
CFTYP_RE25	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_CE_NITRO	real		none	leaf aera ratio sensivity to [nitrogen]	(m <sup>2</sup> /kg)
CFNAM_CE_NITRO	character (LEN=28)		''		
CFTYP_CE_NITRO	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_CF_NITRO	real		none	lethal minimum value of leaf area ratio	(m <sup>2</sup> /kg)
CFNAM_CF_NITRO	character (LEN=28)		''		
CFTYP_CF_NITRO	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_CNA_NITRO	real		none	nitrogen concentration of active biomass	(kg/kg)
CFNAM_CNA_NITRO	character (LEN=28)		''		
CFTYP_CNA_NITRO			none		

	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'			
LUNIF_STRESS	logical		T		

## 4.1.2.4.2. TEB scheme

Over urban areas, all surface parameters have to be specified by the user in namelist **NAM\_DATA\_TEB**.

But, if LECOCLIMAP = .TRUE. (NAM\_FRAC), only some of them can be specified and the missing parameters are completed with ECOCLIMAP database.

Fortran name	Fortran type	values	default	description	unit
XUNIF_URBTYPE	real	13-21 corresponding to covers 151, 152-154, 155, 156, 157, 158, 159, 160, 161	none	type of urban area	(-)
CFNAM_URBTYPE	character (LEN=28)		' '		
CFTYP_URBTYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_BLD	real		none	fraction of buildings	(-)
CFNAM_BLD	character (LEN=28)		' '		
CFTYP_BLD	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_BLD_HEIGHT	real		none	buildings height	(m)
CFNAM_BLD_HEIGHT	character (LEN=28)		' '		
CFTYP_BLD_HEIGHT	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_WALL_O_HOR	real		none	wall surf. / hor. surf.	(-)
CFNAM_WALL_O_HOR	character (LEN=28)		' '		
CFTYP_WALL_O_HOR	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_Z0_TOWN	real		none	roughness length for momentum	(m)
CFNAM_Z0_TOWN	character (LEN=28)		' '		
CFTYP_Z0_TOWN	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_ALB_ROOF	real		none	roof albedo	(-)
CFNAM_ALB_ROOF	character (LEN=28)		' '		
CFTYP_ALB_ROOF	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		

XUNIF_EMIS_ROOF	real		none	roof emissivity	(-)
CFNAM_EMIS_ROOF	character (LEN=28)		' '		
CFTYP_EMIS_ROOF	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_HC_ROOF	real		none	roof layers heat capacity	( $J/K/m^3$ )
CFNAM_HC_ROOF	character (LEN=28)		' '		
CFTYP_HC_ROOF	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_TC_ROOF	real		none	roof layers thermal conductivity	( $W/K/m$ )
CFNAM_TC_ROOF	character (LEN=28)		' '		
CFTYP_TC_ROOF	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_D_ROOF	real		none	roof layers depth	( $m$ )
CFNAM_D_ROOF	character (LEN=28)		' '		
CFTYP_D_ROOF	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_ALB_ROAD	real		none	road albedo	(-)
CFNAM_ALB_ROAD	character (LEN=28)		' '		
CFTYP_ALB_ROAD	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_EMIS_ROAD	real		none	road emissivity	(-)
CFNAM_EMIS_ROAD	character (LEN=28)		' '		
CFTYP_EMIS_ROAD	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_HC_ROAD	real		none	road layers heat capacity	( $J/K/m^3$ )
CFNAM_HC_ROAD	character (LEN=28)		' '		
CFTYP_HC_ROAD	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_TC_ROAD	real		none	road layers thermal conductivity	( $W/K/m$ )

CFNAM_TC_ROAD	character (LEN=28)		' '		
CFTYP_TC_ROAD	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_D_ROAD	real		none	road layers depth	(m)
CFNAM_D_ROAD	character (LEN=28)		' '		
CFTYP_D_ROAD	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
Fortran name	Fortran type	values	default	description	unit
XUNIF_ALB_WALL	real		none	wall albedo	(-)
CFNAM_ALB_WALL	character (LEN=28)		' '		
CFTYP_ALB_WALL	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_EMIS_WALL	real		none	wall emissivity	(-)
CFNAM_EMIS_WALL	character (LEN=28)		' '		
CFTYP_EMIS_WALL	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_HC_WALL	real		none	wall layers heat capacity	(J/K/m <sup>3</sup> )
CFNAM_HC_WALL	character (LEN=28)		' '		
CFTYP_HC_WALL	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_TC_WALL	real		none	wall layers thermal conductivity	(W/K/m)
CFNAM_TC_WALL	character (LEN=28)		' '		
CFTYP_TC_WALL	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_D_WALL	real		none	wall layers depth	(m)
CFNAM_D_WALL	character (LEN=28)		' '		
CFTYP_D_WALL	character (LEN=6)	'DIRECT', 'BINLLF','BINLLV', 'ASCLLV'	none		
XUNIF_H_TRAFFIC	real		none	anthropogenic sensible	(W/m <sup>2</sup> )



				heat fluxes due to traffic	
CFNAM_H_TRAFFIC	character (LEN=28)		' '		
CFTYP_H_TRAFFIC	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_LE_TRAFFIC	real		none	anthropogenic latent heat fluxes due to traffic	(W/m <sup>2</sup> )
CFNAM_LE_TRAFFIC	character (LEN=28)		' '		
CFTYP_LE_TRAFFIC	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_H_INDUSTRIES	real		none	anthropogenic sensible heat fluxes due to factories	(W/m <sup>2</sup> )
CFNAM_H_INDUSTRIES	character (LEN=28)		' '		
CFTYP_H_INDUSTRIES	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
XUNIF_LE_INDUSTRIES	real		none	anthropogenic latent heat fluxes due to factories	(W/m <sup>2</sup> )
CFNAM_LE_INDUSTRIES	character (LEN=28)		' '		
CFTYP_LE_INDUSTRIES	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		

### **4.1.2.4.3. TEB\_GARDEN scheme**

Over urban areas, all vegetation surface parameters have to be specified by the user in namelist **NAM\_DATA\_TEB\_GARDEN**, a duplication of **NAM\_DATA\_ISBA**.

## 4.1.2.4.4. SEAFLUX scheme

Treat SST as a forcing variable. For that purpose, several SST files at a given time are required and namelist `NAM_DATA_SEAFLUX` should be filled.

Fortran name	Fortran type	values	default	description	unit
LSST_DATA	logical		none	flag to activate this option	(-)
NTIME	integer	12	12	number of SST data	(-)
CFNAM_SST	character (LEN=28)		' '		
CFTYP_SST	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none		
NYEAR_SST	integer				
NMONTH_SST	integer				
NDAY_SST	integer				
XTIME_SST	real				

- LSST\_DATA : flag to initialize SST from a climatology
- NTIME : number of SST input files
- CFNAM\_SST: SST data file name
- CFTYP\_SST: type of SST data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- NYEAR\_SST: year of SST data file
- NMONTH\_SST: month of SST data file
- NDAY\_SST: day of SST data file
- XTIME\_SST: time in seconds of SST data file

How to initialise SST from external files: an example with 3 SST input files (lat, lon, value type).

```
&NAM_DATA_SEAFLUX      NTIME = 3 , LSST_DATA = T ,
                        CFNAM_SST (1)   = â sst_1.dat                â , CFTYP_SS
                        CFNAM_SST (2)   = â sst_2.dat                â , CFTYP_SS
                        CFNAM_SST (3)   = â sst_3.dat                â , CFTYP_SS
                        NYEAR_SST (1)=1985, NMONTH_SST (1)=12,  NDAY_SST (1)=31 , XTIME_SS
                        NYEAR_SST (2)=1986, NMONTH_SST (2)=1 ,  NDAY_SST (2)=1  , XTIME_SS
                        NYEAR_SST (3)=1986, NMONTH_SST (3)=1 ,  NDAY_SST (3)=2  , XTIME_SS
/
```

- XUNIF\_xxx : uniform prescribed value of parameter xxx. If XUNIF\_xxx is set, file CFNAM\_xxx is not used.
- CFNAM\_xxx: data file name associated to parameter xxx. If XUNIF\_xxx is set, file CFNAM\_xxx is not used.
- CFTYP\_xxx: type of sea data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')



## 4.2. Choice of the surface schemes

You must first choose the surface schemes you will use. It is not possible, once chosen, to modify the surface schemes in the later steps (**PREP, running of the schemes, DIAG**).

Depending on the schemes you use, some additional physiographic fields will be computed if they are needed for the surface scheme chosen. For example, the ISBA scheme (used for vegetation and soil) needs the fractions of clay and sand.

## 4.2.1. Namelist NAM\_PGD\_SCHEMES

This namelist defines the four schemes that will be used, one for each type of surface (sea, inland water, town, vegetation).

Fortran name	Fortran type	values	default value
CNATURE	string of 6 characters	"NONE ", "FLUX ", "TSZ0 ", "ISBA "	"ISBA "
CSEA	string of 6 characters	"NONE ", "FLUX ", "SEAFLX"	"SEAFLX"
CWATER	string of 6 characters	"NONE ", "FLUX ", "WATFLX", "FLAKE "	"WATFLX"
CTOWN	string of 6 characters	"NONE ", "FLUX ", "TEB "	"TEB "
LGARDEN	logical		F

- CNATURE: scheme used for vegetation and natural soil covers . The different possibilities are:
  - ◆ "NONE " : no scheme used. No fluxes will be computed at the surface.
  - ◆ "FLUX " : ideal fluxes are prescribed. The have to be set in the fortran routine `init_ideal_flux.f90`.
  - ◆ "TSZ0 " : In this cheme, the fluxes are computed according to the ISBA physics, but the surface characteristics (temperature, humidity, etc...) remain constant with time.
  - ◆ "ISBA " : this is the full ISBA scheme (Noilhan and Planton 1989), with all options developed since this initial paper.
- CSEA : scheme used for sea and ocean . The different possibilities are:
  - ◆ "NONE " : no scheme used. No fluxes will be computed at the surface.
  - ◆ "FLUX " : ideal fluxes are prescribed. The have to be set in the fortran routine `init_ideal_flux.f90`.
  - ◆ "SEAFLX" : this is a relatively simple scheme, using the Charnock formula.
- CWATER : scheme used for inland water . The different possibilities are:
  - ◆ "NONE " : no scheme used. No fluxes will be computed at the surface.
  - ◆ "FLUX " : ideal fluxes are prescribed. The have to be set in the fortran routine `init_ideal_flux.f90`.
  - ◆ "WATFLX" : this is a relatively simple scheme, using the Charnock formula.
  - ◆ "FLAKE " : this is lake scheme from Mironov, 2005.
- CTOWN : scheme used for towns . The different possibilities are:
  - ◆ "NONE " : no scheme used. No fluxes will be computed at the surface.
  - ◆ "FLUX " : ideal fluxes are prescribed. The have to be set in the fortran routine `init_ideal_flux.f90`.
  - ◆ "TEB " : this is the Town Energy Balance scheme (Masson 2000), with all the susequent ameliorations of the scheme.
- LGARDEN : general flag to activate TEB\_GARDEN

## 4.3. Definition of the grid

Note that all the namelists presented in this section are ignored if the grid is imposed, in the fortran code, from an atmospheric model. This is the case when one already have defined the atmospheric grid and one want to be sure that the surface has the same grid. For example, this is what happens in the MESONH program PREP\_IDEAL\_CASE (when no physiographic surface file is used).

If you are in this case, ignore all the namelists presented in this section, and only the namelists for cover and the following ones, have to be used.

## **4.3.1. Choice of the grid type**



## 4.3.1.1. Namelist NAM\_PGD\_GRID

This namelist defines the grid type, either specified or from an existing surface file

Fortran name	Fortran type	default value
CGRID	string of 10 characters	"CONF PROJ "
YINIFILE	string of 28 characters	none
YFILETYPE	string of 6 characters	none

- CGRID: type of grid and projection . It is used **only** if a file is not prescribed (see below).  
The different grid possibilities are:
  - ◆ "GAUSS " : this grid is a gaussian grid (global grid, that may be stretched, rotated, ...)
  - ◆ "CONF PROJ " : this grid is a regular grid (in meters in x and y perpendicular directions) on conformal projection plan (Mercator, Lambert or polar stereographic).
  - ◆ "CARTESIAN " : this grid is a regular grid (in meters in x and y perpendicular directions), with no reference to real geographical coordinates.
  - ◆ "LONLAT REG" : this grid is defined as a regular latitude - longitude grid.
  - ◆ "LONLATVAL " : this grid is defined as a not regular latitude - longitude grid (all points and mesh sizes are defined).
  - ◆ "IGN " : this grid type contains all IGN (French National Geographical Institute) possible Lambert projections
  - ◆ "NONE " : this grid is not regular. Only the number of points and the size of each grid mesh is prescribed. There is no positioning of each point compared to any other.
- YINIFILE: name of the file used to define the grid. It is possible to define the grid as a subgrid of a previously created file. This is currently possible only for files that have a "CONF PROJ " or "CARTESIAN " grid type. The exact definition of the subgrid grid chosen is prescribed in a namelist (described below), depending on the type of grid available in the file chosen. **The use of a file has priority on the CGRID type.**
- YFILETYPE: type of the YINIFILE file, if the latter is provided. YFILETYPE must be given. The following values are currently usable :
  - ◆ "MESONH" : the file type is a MESONH file.

## **4.3.2. Conformal projection grids (Mercator, Lambert, Polar stereographic)**

## 4.3.2.1. Namelist NAM\_CONF\_PROJ

This namelist defines the projection in case CGRID="CONF PROJ "

Fortran name	Fortran type	default value
XLAT0	real	none
XLON0	real	none
XRPK	real	none
XBETA	real	none

- XLAT0: reference latitude for conformal projection (real, decimal degrees)
- XLON0: reference longitude for conformal projection (real, decimal degrees)
- XRPK: cone factor for the projection (real):
  - ◆ XRPK=1: polar stereographic projection from south pole
  - ◆  $1 > \text{XRPK} > 0$ : Lambert projection from south pole
  - ◆ XRPK=0: Mercator projection from earth center
  - ◆ -1
  - ◆ XRPK=-1: polar stereographic projection from north pole
- XBETA: rotation angle of the simulation domain around the reference longitude (real)

## 4.3.2.2. Namelist NAM\_CONF\_PROJ\_GRID

This namelists defines the horizontal domain in case CGRID="CONF PROJ ".

Fortran name	Fortran type	
XLATCEN	real	
XLONCEN	real	
NIMAX	integer	
NJMAX	integer	
DX	real	
DY	real	

- XLATCEN: latitude of the point of the center of the domain (real, decimal degrees)
- XLONCEN: longitude of the point of the center of the domain (real, decimal degrees)
- NIMAX: number of surface points of the grid in direction  $x$ .
- NJMAX: number of surface points of the grid in direction  $y$ .
- DX: grid mesh size on the conformal plane in  $x$  direction (real, meters).
- DY: grid mesh size on the conformal plane in  $y$  direction (real, meters).

### 4.3.2.3. Namelist NAM\_INIFILE\_CONF\_PROJ

This namelist defines the horizontal domain from an existing surface file in which grid type is "CONF PROJ ". If nothing is set in the namelist, a grid identical as the one in the file is chosen.

Fortran name	Fortran type	default value
IXOR	integer	1
IYOR	integer	1
IXSIZE	integer	YINIFILE size
IYSIZE	integer	YINIFILE size
IDXRATIO	integer	1
IDYRATIO	integer	1

- IXOR: first point I index, according to the YINIFILE grid, left to and out of the new physical domain.
- IYOR: first point J index, according to the YINIFILE grid, under and out of the new physical domain.
- IXSIZE: number of grid points in I direction, according to YINIFILE grid, recovered by the new domain. If to be used in MESONH, it must only be factor of 2,3 or 5.
- IYSIZE: number of grid points in J direction, according to YINIFILE grid, recovered by the new domain. If to be used in MESONH, it must only be factor of 2,3 or 5.
- IDXRATIO: resolution factor in I direction between the YINIFILE grid and the new grid. If to be used in MESONH, it must only be factor of 2,3 or 5.
- IDYRATIO: resolution factor in J direction between the YINIFILE grid and the new grid. If to be used in MESONH, it must only be factor of 2,3 or 5.

### 4.3.3. Cartesian grids

### 4.3.3.1. Namelist NAM\_CARTESIAN

This namelist defines the projection in case CGRID="CARTESIAN "

Fortran name	Fortran type	default value
XLAT0	real	none
XLON0	real	none
NIMAX	integer	none
NJMAX	integer	none
XDY	real	none
XDX	real	none

- XLAT0: reference latitude (real, decimal degrees)
- XLON0: reference longitude (real, decimal degrees)
- NIMAX: number of surface points of the grid in direction  $x$ .
- NJMAX: number of surface points of the grid in direction  $y$ .
- XDX: grid mesh size on the conformal plane in  $x$  direction (real, meters).
- XDY: grid mesh size on the conformal plane in  $y$  direction (real, meters).

## 4.3.3.2. Namelist NAM\_INIFILE\_CARTESIAN

This namelist defines the horizontal domain from an existing surface file in which grid type is "CARTESIAN ". If nothing is set in the namelist, a grid identical as the one in the file is chosen.

Fortran name	Fortran type	default value
IXOR	integer	1
IYOR	integer	1
IXSIZE	integer	YINIFILE size
IYSIZE	integer	YINIFILE size
IDXRATIO	integer	1
IDYRATIO	integer	1

- IXOR: first point I index, according to the YINIFILE grid, left to and out of the new physical domain.
- IYOR: first point J index, according to the YINIFILE grid, under and out of the new physical domain.
- IXSIZE: number of grid points in I direction, according to YINIFILE grid, recovered by the new domain. If to be used in MESONH, it must only be factor of 2,3 or 5.
- IYSIZE: number of grid points in J direction, according to YINIFILE grid, recovered by the new domain. If to be used in MESONH, it must only be factor of 2,3 or 5.
- IDXRATIO: resolution factor in I direction between the YINIFILE grid and the new grid. If to be used in MESONH, it must only be factor of 2,3 or 5.
- IDYRATIO: resolution factor in J direction between the YINIFILE grid and the new grid. If to be used in MESONH, it must only be factor of 2,3 or 5.



## 4.3.4. Longitude-latitude grids

## 4.3.4.1. Namelist NAM\_LONLAT\_REG

This namelist defines the projection in case CGRID="LONLAT REG"

Fortran name	Fortran type	default value
XLONMIN	real	none
XLONMAX	real	none
XLATMIN	real	none
XLATMAX	real	none
NLON	integer	none
NLAT	integer	none

- XLONMIN: minimum longitude covered by the grid, i.e. corresponding to the west border of the domain (real, decimal degrees). XLONMIN must be smaller than XLONMAX, but no more than 360 smaller.
- XLONMAX: maximum longitude covered by the grid, i.e. corresponding to the east border of the domain (real, decimal degrees). XLONMAX must be larger than XLONMIN, but no more than 360 larger.
- XLATMIN: minimum latitude covered by the grid, i.e. corresponding to the south border of the domain (real, decimal degrees). XLATMIN must be between -90 and +90, and smaller than XLATMAX.
- XLATMAX: maximum longitude covered by the grid, i.e. corresponding to the 'right' border of the domain (real, decimal degrees). XLATMAX must be between -90 and +90, and larger than XLATMIN.
- NLON: number of surface points in the longitude direction.
- NLAT: number of surface points in the latitude direction.

## 4.3.4.2. Namelist NAM\_LONLATVAL

This namelist defines the projection in case CGRID="LONLATVAL "

Fortran name	Fortran type	values	default value	unit
NPOINTS	integer		none	
XX	real		none	degrees East
XY	real		none	degrees North
XDX	real		none	degrees
XDY	real		none	degrees

- NPOINTS: number of grid points defining the grid
- XX: longitude of grid mesh center
- YY: latitude coordinate of grid mesh center
- XDX: grid mesh size in  $x$  direction (real, degrees East).
- XDY: grid mesh size in  $y$  direction (real, degrees North).

## 4.3.5. Regular Lambert grids

## 4.3.5.1. Namelist NAM\_IGN

This namelist defines the projection in case CGRID="IGN "

Fortran name	Fortran type	values	default value
CLAMBERT	character (len=3)	'L1 ', 'L2 ', 'L3 ', 'L4 ', 'L2E', 'L93'	none
NPOINTS	integer		none
XX	real		none
XY	real		none
XDX	real		none
XDY	real		none

- CLAMBERT: type of Lambert prjection
  - ◆ "L1 " : Lambert I
  - ◆ "L2 " : Lambert II
  - ◆ "L3 " : Lambert III
  - ◆ "L4 " : Lambert IV
  - ◆ "L2E" : Extended Lambert II
  - ◆ "L93" : Lambert 93
- NPOINTS: number of grid points defining the grid
- XX: X coordinate of grid mesh center
- YY: Y coordinate of grid mesh center
- XDX: grid mesh size on the conformal plane in *x* direction (real, meters).
- XDY: grid mesh size on the conformal plane in *y* direction (real, meters).

## 4.3.6. Gaussian grids

These namelists define the projection in case CGRID="GAUSS "

## 4.3.6.1. Namelist NAMDIM

Fortran name	Fortran type	default value
NDGLG	integer	none

- NDGLG: number of pseudo-latitudes

## 4.3.6.2. Namelist NAMRGRI

Fortran name	Fortran type	default value
NRGRI	integer	none

- NRGRI: number of pseudo-longitudes on each pseudo-latitude circle starting from the rotated pole



### 4.3.6.3. Namelist NAMGEM

Fortran name	Fortran type	default value
RMUCEN	real	none
RLOCEN	real	none
RSTRET	real	none

- RMUCEN: sine of the latitude of the rotated pole
- RLOCEN: longitude of the rotated pole (radian)
- RSTRET: stretching factor (must be greater than or equal to 1)

## 4.4. Land cover fractions

## 4.4.1. Namelist NAM\_COVER

This namelist gives the information to compute the surface cover fractions.

Fortran name	Fortran type	values	default value
XUNIF_COVER	array of 573 reals	$\geq 0$ and $\sum_{i=1}^{573} \text{XUNIF\_COVER}(i) = 1$	none
YCOVER	character (LEN=28)		' '
YFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
XRM_COVER	real	$\geq 0$	$10^{-6}$
XRM_COAST	real	$\geq 0$	1.
XRM_LAKE	real	$\geq 0$	0.
XRM_SEA	real	$\geq 0$	0.
LORCA_GRID	logical		F
XLAT_ANT	real		-77.
LIMP_COVER	logical		F

- XUNIF\_COVER : specified values for uniform cover fractions. For each index  $i$  between 1 and 573, XUNIF\_COVER( $i$ ) is the fraction of the  $i^{th}$  ecosystem of ecoclimap. The same fraction of each ecosystem is set to all points of the grid. The sum of all ecosystem fractions must be equal to one  $\hat{A}$  :  $\sum_{i=1}^{573} \text{XUNIF\_COVER}(i) = 1$   
If XUNIF\_COVER is set, it has priority on the use of an ecosystem file (see next item  $\hat{A}$  : YCOVER). In the case of grid without any reference to geographical coordinates ("CARTESIAN " or "NONE " ), XUNIF\_COVER **must** be set.
- YCOVER: ecoclimap data file name. It is used only if XUNIF\_COVER is not set.
- YFILETYPE: type of YCOVER file ('DIRECT', 'BINLLV', 'BINLLF', 'ASCLLV').
- XRM\_COVER: for each point, all fractions of ecosystems that are below XRM\_COVER are removed (i.e. set to zero), and the corresponding area fractions are distributed among the remaining ecosystem fractions. Whatever the value of XRM\_COVER, at least one ecosystem remains for each grid point.
- XRM\_COAST: limit of coast coverage under which the coast is replaced by sea or inland water.
- XRM\_LAKE: limit of inland lake coverage under which the water is removed.
- XRM\_SEA: limit of sea coverage under which the sea is removed.
- LORCA\_GRID: flag to ensure the compatibility between surfex and Orca grid which minimal latitude over Antarctica is 77S
- XLAT\_ANT: minimum Orca grid latitude over Antarctica

- **LIMP\_COVER**: reads the cover fractions in an existing PGD file to avoid their computation

## 4.4.2. Namelist NAM\_PGD\_ARRANGE\_COVER

This namelist initialyses change water (not lake) to nature and/or town to rock keys.

Fortran name	Fortran type	values	default value
LWATER_TO_NATURE	logical		F
LTOWN_TO_ROCK	logical		F

- LWATER\_TO\_NATURE: Change Wetland treated as inland water into nature
- LTOWN\_TO\_ROCK : Change Town into Rock

### 4.4.3. Namelist `NAM_READ_DATA_COVER`

Fortran name	Fortran type	values	default value
<code>LREAD_DATA_COVER</code>	logical		T

- `LREAD_DATA_COVER`: if T, covers data are read in .bin files; if F, in fortran routines.

## 4.4.4. Namelist NAM\_WRITE\_COVER\_TEX

Fortran name	Fortran type	values	default value
CLANG	character(LEN=2)	"	'EN'

- CLANG: language used in the file class\_cover\_tex.tex

## **4.5. Specificities of ecoclimap II classification**



## 4.5.1. Namelist NAM\_ECOCLIMAP2

This namelist allows to choose which LAI is used: a climatological one (average over years 2002-2006) or a specific year (between 2002 and 2006). This is the place to define irrigation file.

Fortran name	Fortran type	values	default value
LCLIM_LAI	logical		.TRUE.
YIRRIG	character (LEN=28)		' '

- LCLIM\_LAI: if .TRUE., climatological LAI is computed otherwise, the LAI corresponding to current year (if between 2002 and 2006) is used.
- YIRRIG: irrigation file name

## **4.6. Orography, subgrid orography gaussian indices and bathymetry**

## 4.6.1. Namelist NAM\_ZS

This namelist defines the orography file and orographic treatment to be done.

Fortran name	Fortran type	values	default value
XUNIF_ZS	real		none
YZS	character (LEN=28)		' ' (default orography is 0.)
YFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
COROGTYPE	character (LEN=3)	'AVG', 'ENV', 'SIL ', 'MAX'	'ENV'
XENV	real		0.
LIMP_ZS	logical		F

- XUNIF\_ZS : uniform value of orography imposed on all points (real,meters). If XUNIF\_ZS is set, file YZS is not used.
- YZS: data file name. If XUNIF\_ZS is set, file YZS is not used. If neither XUNIF\_ZS and YZS is set, then orography is set to zero.
- YFILETYPE: type of data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- COROGTYPE: type of orography (string of 3 characters):
  - ◆ 'AVG': mean orography  $\overline{z_s}$ .
  - ◆ 'ENV': envelope relief, defined from mean orography and the subgrid orography standard deviation as  $\overline{z_s} + XENV * \sigma_{z_s}$ .
  - ◆ 'SIL': silhouette relief, defined as the mean of the two subgrid silhouettes in directions x and y (if two main directions can be defined for the grid chosen).
  - ◆ 'MAX': maximum orography over grid box (avoid averaging in case of sea/land grid box).
- XENV: enhance factor in envelope orography definition (real).
- LIMP\_ZS: reads orography from an existing PGD file

## 4.6.2. Namelist NAM\_GAUSS\_INDEX

This namelist computes the gaussian grid indices once and for all

Fortran name	Fortran type	values	default value
LINDEX_STORE	logical		F
LSTOP_PGD	logical		F
YINDEX_1KM	character (LEN=28)		' '
YINDEX_10KM	character (LEN=28)		' '
YINDEX_100KM	character (LEN=28)		' '

- LINDEX\_STORE : flag to write out gaussian grid indices in a binary file
- LSTOP\_PGD : flag to stop PGD execution once the binary file containing the gaussian grid indices is ready
- YINDEX\_1KM : name of the file where gaussian grid indices at 1km resolution are written
- YINDEX\_10KM : name of the file where gaussian grid indices at 10km resolution are written
- YINDEX\_100KM : name of the file where gaussian grid indices at 100km resolution are written

### 4.6.3. Namelist NAM\_SEABATHY

This namelist defines the bathymetry file

Fortran name	Fortran type	values	default value
XUNIF_SEABATHY	real		300
YSEABATHY	character (LEN=28)		' '
YSEABATHYFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV', 'NETCDF'	none
YNCVARNAME	character (LEN=28)		

- XUNIF\_SEABATHY : uniform value of bathymetry imposed on all points (real,meters). If XUNIF\_SEABATHY is set, file YSEABATHY is not used.
- YSEABATHY: data file name. If XUNIF\_SEABATHY is set, file YSEABATHY is not used. If neither XUNIF\_SEABATHY and YSEABATHY is set, then bathymetry is set to zero.
- YSEABATHYFILETYPE: type of data file ('NETCDF')
- YNCVARNAME: name of variable to be read in NETCDF file

# 4.7. Namelist for ISBA scheme

## 4.7.1. Namelist NAM\_ISBA

Fortran name	Fortran type	values	default value
NPATCH	integer	between 1 and 12	1
CISBA	character (LEN=3)	'2-L', '3-L', 'DIF'	'3-L'
CPHOTO	string of 3 characters	'NON', 'AGS', 'LAI', 'AST', 'LST', 'NIT', 'NCB'	'NON'
CPEDO_FUNCTION	string of 4 characters	'CH78', 'C084', 'CP88', 'W099'	'CH78'
NGROUND_LAYER	integer	>0	3
XUNIF_CLAY	real	between 0 and 1	0.33
YCLAY	character (LEN=28)		''
YCLAYFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
XUNIF_SAND	real	between 0 and 1	0.33
YSAND	character (LEN=28)		''
YSANDFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
XUNIF_RUNOFFB	real		0.5
YRUNOFFB	character (LEN=28)		''
YRUNOFFBFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
XUNIF_WDRAIN	real		0.
YWDRAIN	character (LEN=28)		''
YWDRAINFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
YCTI	character (LEN=28)		''
YCTIFILETYPE	character (LEN=6)	'DIRECT'	none
XUNIF_ORGMAT	real		0.
YORGMAT	character (LEN=28)		''
YORGMATFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
XUNIF_DENSITY	real		0.
YDENSITY	character (LEN=28)		''
YDENSITYFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV'	none
LIMP_SAND	logical		F
LIMP_CLAY	logical		F
LIMP_CTI	logical		F
LIMP_DENSITY	logical		F
LIMP_ORGMAT	logical		F

- NPATCH : number of patches used in ISBA. One patch corresponds to aggregated parameters. 12 patches correspond to separate energy budgets for all vegetation types present in ISBA. 3 patches correspond to bare soil types, low vegetation, trees. If CPHOTO equals 'NON' any number of patches between 1 and 12 is possible, for the other values of CPHOTO,

12 patches are required. The order and the signification of each patch is the following:

- ◆ 1: bare ground
- ◆ 2: rocks
- ◆ 3: permanent snow
- ◆ 4: deciduous forest
- ◆ 5: conifer forest
- ◆ 6: evergreen broadleaf trees
- ◆ 7: C3 crops
- ◆ 8: C4 crops
- ◆ 9: irrigated crops
- ◆ 10: grassland (C3)
- ◆ 11: tropical grassland (C4)
- ◆ 12: garden and parks
- CISBA: type of soil discretization and physics in ISBA:
  - ◆ '2-L' : force-restore method with 2 layers for hydrology
  - ◆ '3-L' : force-restore method with 3 layers for hydrology
  - ◆ 'DIF' : diffusion layer, with any number of layers
- CPHOTO: type of photosynthesis physics. The following options are currently available:
  - ◆ "NON" : none is used. Jarvis formula is used for plant transpiration.
  - ◆ "AGS" : ISBA-AGS, without evolving Leaf Area Index
  - ◆ "LAI" : ISBA-AGS, with evolving Leaf Area Index
  - ◆ "AST" : ISBA-AGS with offensive/defensive stress, without evolving Leaf Area Index
  - ◆ "LST" : ISBA-AGS with offensive/defensive stress, with evolving Leaf Area Index
  - ◆ "NIT" : ISBA-AGS with nitrogen, with evolving Leaf Area Index
- CPEDO\_FUNCTION: Pedo-transfer function for DIF. The following options are currently available:
  - ◆ "CH78" : Clapp and Hornberger 1978 for BC
  - ◆ "C084" : Cosby et al. 1988 for BC
  - ◆ "CP88" : Carsel and Parrish 1988 for VG
  - ◆ "W099" : Wosten et al. 1999 for VG
- NGROUND\_LAYER : number of soil layer used in case of diffusion physics in the soil (CISBA = 'DIF')
- XUNIF\_CLAY : uniform prescribed value of clay fraction.
- YCLAY: clay fraction data file name.
- YCLAYFILETYPE: type of clay data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_SAND : uniform prescribed value of sand fraction.
- YSAND: sand fraction data file name.
- YSANDFILETYPE: type of sand data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_RUNOFFB: uniform prescribed value of subgrid runoff coefficient.
- YRUNOFFB: subgrid runoff coefficient data file name.
- YRUNOFFBFILETYPE: type of subgrid runoff data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_WDRAIN: uniform prescribed value of subgrid drainage.
- YWDRAIN: subgrid drainage data file name.
- YWDRAINFILETYPE: type of subgrid drainage data file ('DIRECT', 'BINLLF', 'BINLLV',



'ASCLLV')

- YCTI: topographic indices file name.
- YCTIFILETYPE: type of topographic file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_ORGMAT: uniform prescribed value of organic matter.
- YORGMAT: organic matter data file name.
- YORGMATFILETYPE: type of organic matter data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- XUNIF\_DENSITY: uniform prescribed value of soil density.
- YDENSITY: soil density data file name.
- YDENSITYFILETYPE: type of soil density data file ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV')
- LIMP\_SAND: reads sand fraction in an existing PGD file
- LIMP\_CLAY: reads clay fraction in an existing PGD file
- LIMP\_CTI: reads topographic indices in an existing PGD file
- LIMP\_ORGMAT: reads organic matter in an existing PGD file
- LIMP\_DENSITY: reads soil density in an existing PGD file

## 4.8. Namelist for FLake scheme

## 4.8.1. Namelist NAM\_DATA\_FLAKE

Over lakes, if one wants to use Flake scheme, some parameters have to be specified by the user in the namelist **NAM\_DATA\_FLAKE**.

Fortran name	Fortran type	values	default	advice	description	unit
				value		
XUNIF_WATER_DEPTH	real			20.	Lake depth	(m)
YWATER_DEPTH	character (LEN=28)		''		filename	
YWATER_DEPTHFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'ASCLLV', 'BINLLV'	none			
YWATER_DEPTH_STATUS	character (LEN=28)		''		status file name	
XUNIF_WATER_FETCH	real			1000.	wind fetch	(m)
YWATER_FETCH	character (LEN=28)		''		filename	
YWATER_FETCHFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'ASCLLV', 'BINLLV'	none			
XUNIF_T_BS	real			286.	temperature at the outer edge of the thermally active layer of the of the bottom sediments	(K)
YT_BS	character (LEN=28)		''		filename	
YT_BSFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'ASCLLV', 'BINLLV'	none			
XUNIF_DEPTH_BS	real			1.	depth of the sediments layer	(m)
YDEPTH_BS	character (LEN=28)		''		filename	
YDEPTH_BSFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'ASCLLV', 'BINLLV'	none			
XUNIF_EXTCOEF_WATER	real			3.	extinction coefficient of solar radiation in water	(m <sup>-1</sup> )
YEXTCOEF_WATER	character (LEN=28)		''		filename	

YEXTCOEF_WATERFILETYPE	character (LEN=6)	'DIRECT', 'BINLLF', 'ASCLLV', 'BINLLV'	none			
------------------------	----------------------	---	------	--	--	--

## **4.9. Namelist to add user's own fields**

## 4.9.1. Namelist NAM\_DUMMY\_PGD

This namelist allows to incorporate into the physiographic file any surface field.

You can treat up to 999 such fields. These fields will be written on all the files you will use later(after prognostic fields initialization, or during and after run, etc...). Their name in the files are 'DUMMY\_GRnnn', where nnn goes from 001 to 999.

During the execution of the programs, these fields are stored in the XDUMMY\_FIELDS(:, :) (first dimension: spatial dimension, second dimension  $\hat{A}$  : total number of fields), in the module MODD\_DUMMY\_SURF\_FIELD\$n. You must modify the fortran source, where you want to use them.

Fortran name	Fortran type	default value
NDUMMY_PGD_NBR	integer	0
CDUMMY_PGD_NAME(:)	1000 * character (LEN=20)	1000 * ''
CDUMMY_PGD_FILE(:)	1000 * character (LEN=28)	1000 * ''
CDUMMY_PGD_COMMENT(:)	1000 * character (LEN=40)	1000 * ''
CDUMMY_PGD_FILETYPE(:)	1000 * character (LEN=6)	1000 * ''
CDUMMY_PGD_AREA(:)	1000 * character (LEN=3)	1000 * 'ALL'
CDUMMY_PGD_ATYPE(:)	1000 * character (LEN=3)	1000 * 'ARI'

Only the first NDUMMY\_PGD\_NBR values in these arrays are meaningful.

- NDUMMY\_PGD\_NBR: number of dummy fields.
- CDUMMY\_PGD\_NAME(:): list of the dummy fields you want to initialize with your own data. You can give any name you want. This is a way to describe what is the field. This information is not used by the program. It is just written in the FM files.
- CDUMMY\_PGD\_FILE(:): list of the names of the files containing the data for the fields you have specified in CDUMMY\_PGD\_NAME(:).
- CDUMMY\_PGD\_FILETYPE(:): list of the types of the files containing the data for the fields you have specified in CDUMMY\_PGD\_NAME(:) ('DIRECT', 'LATLON', 'BINLLF', 'BINLLV', 'ASCLLV').
- CDUMMY\_PGD\_AREA(:): area of meaningfulness of the fields you have specified in CDUMMY\_PGD\_NAME(:) ('ALL', 'NAT', 'TWN', 'SEA', 'WAT', 'LAN', respectively for everywhere, natural areas, town areas, sea, inland waters, land = natural cover + town). For example, oceanic emission of DNS is relevant on 'SEA'.
- CDUMMY\_PGD\_ATYPE(:): type of averaging (during **PGD** for the fields you have specified in CDUMMY\_PGD\_NAME(:) ('ARI', 'INV', 'LOG', respectively for arithmetic, inverse and logarithmic averaging).

## **4.10. Namelist for chemistry anthropogenic emissions**

## 4.10.1. Namelist NAM\_CH\_EMIS\_PGD

This namelist is used to initialize chemistry components emissions.

You can treat up to 999 such fields. These fields will be written on all the files you will use later (after prognostic fields initialization, or during and after run, etc...). Their name in the files are 'EMIS\_GRnnn', where nnn goes from 001 to 999.

During the execution of the programs, these fields are stored in the XEMIS\_GR\_FIELDS(:,:) (first dimension: spatial dimension, second dimension  $\hat{A}$  : total number of fields), in the module MODD\_EMIS\_GR\_FIELD\$n. The temporal evolution, the aggregation of prescribed emissions and the link with the corresponding chemical prognostic variables are handled by the subroutine CH\_EMISSION\_FLUXn.f90

Fortran name	Fortran type	default value
NEMIS_PGD_NBR	integer	0
CEMIS_PGD_NAME(:)	1000 * character (LEN=20)	1000 * ''
CEMIS_PGD_FILE(:)	1000 * character (LEN=28)	1000 * ''
CEMIS_PGD_COMMENT(:)	1000 * character (LEN=40)	1000 * ''
NEMIS_PGD_TIME	integer	0
CEMIS_PGD_FILETYPE(:)	1000 * character (LEN=6)	1000 * 'DIRECT'
CEMIS_PGD_AREA(:)	1000 * character (LEN=3)	1000 * 'ALL'
CEMIS_PGD_ATYPE(:)	1000 * character (LEN=3)	1000 * 'ARI'

Only the first NEMIS\_PGD\_NBR values in these arrays are meaningful.

- NEMIS\_PGD\_NBR: number of dummy fields.
- CEMIS\_PGD\_NAME(:): list of the dummy fields you want to initialize with your own data. You can give any name you want. This is a way to describe what is the field. This information is not used by the program. It is just written in the FM files.
- CEMIS\_PGD\_FILE(:): list of the names of the files containing the data for the fields you have specified in CEMIS\_PGD\_NAME(:).
- CEMIS\_PGD\_COMMENT(:): list of the comments associated to each emission field.
- NEMIS\_PGD\_TIME(:): list of the time of the files containing the data for the fields you have specified in CEMIS\_PGD\_NAME(:).
- CEMIS\_PGD\_FILETYPE(:): list of the types of the files containing the data for the fields you have specified in CEMIS\_PGD\_NAME(:) ('DIRECT', 'BINLLF', 'BINLLV', 'ASCLLV').
- $\hat{A}$  :

area of meaningfulness of the fields you have specified in CEMIS\_PGD\_NAME(:) ('ALL', 'NAT', 'TWN', 'SEA', 'WAT', 'LAN', respectively for everywhere, natural areas, town areas, sea, inland waters, land = natural cover + town). For example, oceanic emission of DNS is relevant on 'SEA'.



## **5. Initialization of the prognostic fields**

## 5.1. Overview of fields computation: PREP

The prognostic fields (temperature, humidity, ice, snow, etc...) are averaged or interpolated on the specified grid by the program **PREP**. They are stored in the surface file. The computation is done separately for each surface scheme.

During the **PREP** facility :

1. You initializes the date of the surface
2. You initializes the prognostic variables of the chosen sea scheme
3. You initializes the prognostic variables of the chosen lake scheme
4. You initializes the prognostic variables of the chosen vegetation scheme
5. You initializes the prognostic variables of the chosen town scheme

Here are presented the initialization procedures for the schemes that need such information (for example, scheme "IDEAL " does not need any information here, but modificaton of the code source *init\_ideal\_flux.f90*).

Note that for each scheme, and for some for each variable of the scheme, it is possible to initialize the prognostic fields either form an operationnal or research model, or using prescribed (usually uniform) fields.

## **5.2. Date initialization and default input data file for all schemes**

## 5.2.1. Namelist NAM\_PREP\_SURF\_ATM

This namelist information is used to (possibly):

- initialize the date of all surface schemes. The namelist information is used only if no input data file is used, either from namelist or by fortran code (as in MESONH programs). If a file is used, the date is read in it.
- define the default file in which each scheme can read the needed data (e.g. temperature).

Note that, all the information given in this namelist can be erased for each scheme by the namelist corresponding to this scheme, as the information in the scheme namelists have priority on namelist NAM\_PREP\_SURF\_ATM.

Fortran name	Fortran type	values	default value
CFILE	string of 28 characters		- atmospheric file used in the program
			calling the surface facilities, if any
			- none otherwise
CFILETYPE	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	- type of the atmospheric file, if any
NYEAR	integer		none
NMONTH	integer		none
NDAY	integer		none
XTIME	real		none

- CFILE: name of the file used to define
  1. the date.
  2. the file in which to read the needed data (e.g. temperature).  
**The use of a file or prescribed value in each scheme namelist has priority on the data in CFILE file of namelist NAM\_PREP\_SURF\_ATM.**
- CFILETYPE: type of the CFILE file, if the latter is provided. CTYPE must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
- NYEAR : year of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NMONTH : month of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NDAY : day of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).

- **XTIME** : time from midnight of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read). (seconds).

## 5.3. Sea scheme "SEAFLX"

## 5.3.1. Namelist NAM\_PREP\_SEAFLUX

This namelist information is used to initialize the "SEAFLX" sea scheme temperature.

Fortran name	Fortran type	values	default value
XSST_UNIF	real		none
CFILE_SEAFLX	string of 28 characters		CFILE in NAM_PREP_SURF_ATM
CTYPE	string of 6 characters	'MESONH', 'GRIB ', 'NETCDF'	CFILETYPE in NAM_PREP_SURF_ATM
NYEAR	integer		none
NMONTH	integer		none
NDAY	integer		none
XTIME	real		none
LSEA_SBL	logical		F
LOCEAN_MERCATOR	logical		F
LOCEAN_CURRENT	logical		F

- XSST\_UNIF : uniform prescribed value of Sea Surface Temperature. This prescribed value, if defined, has priority on the use of CFILE\_SEAFLX data.
- CFILE\_SEAFLX: name of the file used to define the Sea surface Temperature. **The use of a file or prescribed value XSST\_UNIF has priority on the data in CFILE\_SEAFLX file.**
- CTYPE: type of the CFILE\_SEAFLX file, if the latter is provided. CTYPE must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
  - ◆ "NETCDF" : the file type is a NETCDF file, coming from MERCATOR
- NYEAR : year of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NMONTH : month of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NDAY : day of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- XTIME : time from midnight of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read). (seconds).
- LSEA\_SBL : activates surface boundary multi layer scheme over sea.
- LOCEAN\_MERCATOR : oceanic variables initialized from MERCATOR if T
- LOCEAN\_CURRENT : initial ocean state with current (if F ucure=0, vcur=0)





## 5.4. Lake scheme "WATFLX"

## 5.4.1. Namelist NAM\_PREP\_WATFLUX

This namelist information is used to initialize the "WATFLX" sea scheme temperature.

Fortran name	Fortran type	values	default value
XTS_WATER_UNIF	real		none
CFILE_WATFLX	string of 28 characters		CFILE in NAM_PREP_SURF_ATM
CTYPE	string of 6 characters	'MESONH', 'GRIB ', 'NETCDF'	CFILETYPE in NAM_PREP_SURF_ATM
NYEAR	integer		none
NMONTH	integer		none
NDAY	integer		none
XTIME	real		none
LWAT_SBL	logical		F

- XTS\_WATER\_UNIF : uniform prescribed value of water surface temperature supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_WATFLX data.
- CFILE\_WATFLX: name of the file used to define the Sea surface Temperature. **The use of a file or prescribed value XTS\_WATER\_UNIF has priority on the data in CFILE\_WATFLX file.**
- CTYPE: type of the CFILE\_WATFLX file, if the latter is provided. CTYPE must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
  - ◆ "NETCDF" : the file type is a NETCDF file, coming from MERCATOR
- NYEAR : year of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NMONTH : month of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NDAY : day of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- XTIME : time from midnight of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read). (seconds).
- LWAT\_SBL : activates surface boundary multi layer scheme over inland water.

## 5.5. Lake scheme "FLAKE"

## 5.5.1. Namelist NAM\_PREP\_FLAKE

This namelist information is used to initialize the "FLAKE" sea scheme temperature.

Fortran name	Fortran type	values	default value
XTS_WATER_UNIF	real		none
XUNIF_T_SNOW	real		273.15
XUNIF_T_ICE	real		min(273.15,XTS_WATER)
XUNIF_T_MNW	real		depends on XTS_WATER
XUNIF_T_BOT	real		depends on XTS_WATER
XUNIF_T_B1	real		depends on XTS_WATER
XUNIF_H_SNOW	real		0.
XUNIF_H_ICE	real		0. or 0.01 if XTS_WATER < 273.15
XUNIF_H_ML	real		3.
XUNIF_H_B1	real		1.
CFILE_FLAKE	string of 28 characters		CFILE in NAM_PREP_SURF_ATM
CTYPE	string of 6 characters	'MESONH', 'GRIB ', 'NETCDF'	CFILETYPE in NAM_PREP_SURF_ATM
NYEAR	integer		none
NMONTH	integer		none
NDAY	integer		none
XTIME	real		none
LWAT_SBL	logical		F

- XTS\_WATER\_UNIF : uniform prescribed value of water surface temperature supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_FLAKE data.
- XUNIF\_T\_SNOW : surface temperature of snow (K)
- XUNIF\_T\_ICE : surface temperature at the ice-atmosphere or at the ice-snow interface (K)
- XUNIF\_T\_MNW : mean water column temperature (K)
- XUNIF\_T\_BOT : water temperature at the bottom of the lake (K)
- XUNIF\_T\_B1 : temperature at the bottom of the upper layer of sediments (K)
- XUNIF\_H\_SNOW : snow layer thickness (m)
- XUNIF\_H\_ICE : ice layer thickness (m)
- XUNIF\_H\_B1 : thickness of the upper level of the active sediments (m)
- CFILE\_FLAKE: name of the file used to define the Sea surface Temperature. **The use of a file or prescribed value XTS\_WATER\_UNIF has priority on the data in CFILE\_FLAKE file.**

- CTYPE: type of the CFILE\_FLAKE file, if the latter is provided. CTYPE must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
  - ◆ "NETCDF" : the file type is a NETCDF file, coming from MERCATOR
- NYEAR : year of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NMONTH : month of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NDAY : day of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- XTIME : time from midnight of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read). (seconds).
- LWAT\_SBL : activates surface boundary multi layer scheme over inland water.

## 5.6. Vegetation scheme "ISBA "

## 5.6.1. Namelist NAM\_PREP\_ISBA

This namelist information is used to initialize the "ISBA " vegetation scheme variables: soil temperature profile, soil water and ice profiles, water intercepted by leaves, snow.

Fortran name	Fortran type	values	default value
XHUG_SURF	real		none
XHUG_ROOT	real		none
XHUG_DEEP	real		none
XHUGI_SURF	real		none
XHUGI_ROOT	real		none
XHUGI_DEEP	real		none
CFILE_HUG_SURF	string of 28 characters		CFILE_ISBA in this namelist
CFILE_HUG_ROOT	string of 28 characters		CFILE_HUG in this namelist
CFILE_HUG_DEEP	string of 28 characters		CFILE_HUG in this namelist
CFILE_HUG	string of 28 characters		CFILE_ISBA in this namelist
CTYPE_HUG	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	CTYPE in this namelist
XTG_SURF	real		none
XTG_ROOT	real		none
XTG_DEEP	real		none
CFILE_TG_SURF	string of 28 characters		CFILE_TG in this namelist
CFILE_TG_ROOT	string of 28 characters		CFILE_TG in this namelist
CFILE_TG_DEEP	string of 28 characters		CFILE_TG in this namelist
CFILE_TG	string of 28 characters		CFILE_ISBA in this namelist
CTYPE_TG	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	CTYPE in this namelist
CFILE_ISBA	string of 28 characters		CFILE in NAM_PREP_SURF_ATM
CTYPE	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	CFILETYPE in NAM_PREP_SURF_ATM
NYEAR	integer		none
NMONTH	integer		none
NDAY	integer		none
XTIME	real		none
LISBA_CANOPY	logical		F

- XHUG\_SURF : uniform prescribed value of liquid soil water index (SWI) for the surface soil layer. This prescribed value, if defined, has priority on the use of CFILE\_HUG and CFILE\_ISBA data.
- XHUG\_ROOT : uniform prescribed value of liquid soil water index (SWI) for the root zone soil layer(s). This prescribed value, if defined, has priority on the use of CFILE\_HUG and

CFILE\_ISBA data.

- XHUG\_DEEP : uniform prescribed value of liquid soil water index (SWI) for the deep soil layer(s). This prescribed value, if defined, has priority on the use of CFILE\_HUG and CFILE\_ISBA data.
- XHUGI\_SURF : uniform prescribed value of ice soil water index (SWI) for the surface soil layer. This prescribed value, if defined, has priority on the use of CFILE\_HUG and CFILE\_ISBA data.
- XHUGI\_ROOT : uniform prescribed value of ice soil water index (SWI) for the root zone soil layer(s). This prescribed value, if defined, has priority on the use of CFILE\_HUG and CFILE\_ISBA data.
- XHUGI\_DEEP : uniform prescribed value of ice soil water index (SWI) for the deep soil layer(s). This prescribed value, if defined, has priority on the use of CFILE\_HUG and CFILE\_ISBA data.
- CFILE\_HUG\_SURF: name of the file used to define the liquid soil water index (SWI) for the surface soil layer.
- CFILE\_HUG\_ROOT: name of the file used to define the liquid soil water index (SWI) for the root zone soil layer(s).
- CFILE\_HUG\_DEEP: name of the file used to define the liquid soil water index (SWI) for the deep soil layer(s).
- CFILE\_HUG: name of the file used to define the soil water profiles.

**The use of a file or prescribed value of XHUG\_SURF, XHUG\_ROOT and XHUG\_DEEP has priority on the data in CFILE\_HUG file.**

- CTYPE\_HUG: type of the CFILE\_HUG file, if the latter is provided. CTYPE\_HUG must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
- XTG\_SURF : uniform prescribed value of temperature for the surface soil layer, supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_TG and CFILE\_ISBA data.
- XTG\_ROOT : uniform prescribed value of temperature for the root zone soil layer(s), supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_TG and CFILE\_ISBA data.
- XTG\_DEEP : uniform prescribed value of temperature for the deep soil layer(s), supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_TG and CFILE\_ISBA data.
- CFILE\_TG\_SURF: name of the file used to define the surface soil temperature profile.
- CFILE\_TG\_ROOT: name of the file used to define the root zone soil temperature profile.
- CFILE\_TG\_DEEP: name of the file used to define the deep soil temperature profile.



- CFILE\_TG: name of the file used to define the soil temperature profile.  
**The use of a file or prescribed value of XTG\_SURF, XTG\_ROOT and XTG\_DEEP has priority on the data in CFILE\_TG file.**
- CTYPE\_TG: type of the CFILE\_TG file, if the latter is provided. CTYPE\_TG must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
- CFILE\_ISBA: name of the file used to define any ISBA variable. **The use of a file or prescribed value XHUG\_SURF, XHUG\_ROOT, XHUG\_DEEP, XTG\_SURF, XTG\_ROOT, XTG\_DEEP, CFILE\_WG and CFILE\_TG has priority on the data in CFILE\_ISBA file.**
- CTYPE: type of the CFILE\_ISBA file, if the latter is provided. CTYPE must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
- NYEAR : year of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NMONTH : month of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NDAY : day of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- XTIME : time from midnight of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read). (seconds).
- LISBA\_CANOPY : activates surface boundary multi layer scheme over vegetation.

## 5.6.2. Namelist NAM\_PREP\_ISBA\_SNOW

This namelist defines the type of snow scheme used in ISBA scheme.

Fortran name	Fortran type	values	default value
CSNOW	string of 3 characters	'D95', '3-L', 'EBA','CRO'	'D95'
NSNOW_LAYER	integer		1
CFILE_SNOW	string of 28 characters		CFILE_ISBA in NAM_PREP_ISBA
CTYPE_SNOW	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	CTYPE in NAM_PREP_ISBA
LSNOW_IDEAL	logical		.FALSE.
LSNOW_FRAC_TOT	logical		.FALSE.
XWSNOW	real(20)		0.
XTSNOW	real(20)		273.16
XRSNOW	real(20)		300.
XASNOW	real		0.5
XSG1SNOW	real(20)		none
XSG2SNOW	real(20)		none
XHISTSNOW	real(20)		none
XAGESNOW	real(20)		none

- CSNOW : type of snow scheme. Possible snow schemes are:
  1. 'D95' : Douville et al (1995) snow scheme.
  2. '3-L' : Boone and Etchevers (2000) three layers snow scheme.
  3. 'EBA' : Bogatchev and Bazile (2005), Arpege operational snow scheme.
  4. 'CRO' : Crocus model
- NSNOW\_LAYER : number of snow layers
- CFILE\_SNOW : name of the file used to define the snow profiles. **The use of a file or prescribed value of XRSNOW, XTSNOW, XWSNOW and XASNOW (and XSG1SNOW, XSG2SNOW, XHISTSNOW and XAGESNOW in case of CSNOW = CROCUS) has priority on the data in CFILE\_SNOW file**
- CTYPE\_SNOW : type of the CFILE\_SNOW file, if the latter is provided. CTYPE\_SNOW must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
- LSNOW\_IDEAL : if LSNOW\_IDEAL = .FALSE. , only one value can be given for following snow parameters and a vertical interpolation is processed. If LSNOW\_IDEAL = .TRUE., values are given for each layer and there is no vertical interpolation performed.

- **LSNOW\_FRAC\_TOT** : if **LSNOW\_FRAC\_TOT** = **.TRUE.**, the total snow fraction  $XPSN = MIN(1.0, ZSNOWSWE(:)/XWCRN\_EXPL)$  where **ZSNOWSWE** is the snow liquid water content, and **XWCRN\_EXPL** is the critical value of the equivalent water content of the snow reservoir.
- **XWSNOW** : uniform value to initialize snow content, one for each layer
- **XTSNOW** : uniform value to initialize snow temperature, one for each layer
- **XRSNOW** : uniform value to initialize snow density, one for each layer
- **XASNOW** : uniform value to initialize snow albedo
- **XSG1SNOW** : uniform value to initialize snow layers grain feature 1 for Crocus, one for each layer
- **XSG2SNOW** : uniform value to initialize snow layers grain feature 2 for Crocus, one for each layer
- **XHISTSNOW** : uniform value to initialize snow layer grain historical parameter for Crocus, one for each layer
- **XAGESNOW** : uniform value to initialize snow grain age for Crocus, one for each layer

### 5.6.3. Namelist NAM\_PREP\_ISBA\_CARBON

Fortran name	Fortran type	values	default value
CRESPSL	string of 3 characters	'DEF', 'PRM', 'CNT'	'DEF'

- CRESPSL : soil respiration option. Possible values are :
  1. 'DEF' : Norman 1992
  2. 'PRM' : Rivalland 2003
  3. 'CNT' : Century model 2007

## 5.7. Town scheme "TEB "

## 5.7.1. Namelist NAM\_PREP\_TEB

This namelist information is used to initialize the "TEB " urban scheme variables: road, roof and wall temperature profiles, water intercepted by roofs and roads, snow, building internal temperature.

Fortran name	Fortran type	values	default value
XWS_ROAD	real		none
XWS_ROOF	real		none
CFILE_WS	string of 28 characters		CFILE_TEB in this namelist
CTYPE_WS	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	CTYPE in this namelist
XTS_ROAD	real		none
XTS_ROOF	real		none
XTS_WALL	real		none
XTI_BLD	real		none
XTI_ROAD	real		none
CFILE_TS	string of 28 characters		CFILE_TEB in this namelist
CTYPE_TS	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	CTYPE in this namelist
CFILE_TEB	string of 28 characters		CFILE in NAM_PREP_SURF_ATM
CTYPE	string of 6 characters	'MESONH', 'GRIB ', 'LFI '	CFILETYPE in NAM_PREP_SURF_ATM
NYEAR	integer		none
NMONTH	integer		none
NDAY	integer		none
XTIME	real		none
LTEB_CANOPY	logical		F

- XWS\_ROAD : uniform prescribed value of soil water interception for the road reservoir. This prescribed value, if defined, has priority on the use of CFILE\_WS and CFILE\_TEB data.
- XWS\_ROOF : uniform prescribed value of soil water interception for the roof reservoir. This prescribed value, if defined, has priority on the use of CFILE\_WS and CFILE\_TEB data.
- CFILE\_WS: name of the file used to define the soil water reservoirs. **The use of a file or prescribed value of XWS\_ROAD and XWS\_ROOF has priority on the data in CFILE\_WS file.**
- CTYPE\_WS: type of the CFILE\_WS file, if the latter is provided. CTYPE\_WS must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model

4. "MOCAGE" : Mocage french research chemistry model

- XTS\_ROAD : uniform prescribed value of temperature for road, supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_TS and CFILE\_TEB data.
- XTS\_ROOF : uniform prescribed value of temperature for roof, supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_TS and CFILE\_TEB data.
- XTS\_WALL : uniform prescribed value of temperature for wall, supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_TS and CFILE\_TEB data.
- XTI\_BLD : uniform prescribed value of internal building temperature. This temperature is not dependent on altitude. This prescribed value, if defined, has priority on the use of CFILE\_TS and CFILE\_TEB data.
- XTI\_ROAD : uniform prescribed value of deep road temperature, supposed at an **altitude of 0m** (mean sea level altitude). The temperature is then modified for each point depending on its altitude, following a uniform vertical gradient of  $-6.5 K km^{-1}$ . This prescribed value, if defined, has priority on the use of CFILE\_TS and CFILE\_TEB data.
- CFILE\_TS: name of the file used to define the soil temperature profile. **The use of a file or prescribed value of XTS\_ROAD, XTS\_ROOF, XTS\_WALL, XTI\_BLD or XTI\_ROAD has priority on the data in CFILE\_TS file.**
- CTYPE\_TS: type of the CFILE\_TS file, if the latter is provided. CTYPE\_TS must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
- CFILE\_TEB: name of the file used to define any TEB variable. **The use of a file or prescribed value XWS\_ROAD, XWS\_ROOF, XTS\_ROAD, XTS\_ROOF, XTS\_WALL, XTI\_BLD, XTI\_ROAD, CFILE\_WS or CFILE\_TS has priority on the data in CFILE\_TEB file.**
- CTYPE: type of the CFILE\_TEB file, if the latter is provided. CTYPE must then be given. The following values are currently usable:
  - ◆ "MESONH" : the file type is a MESONH file.
  - ◆ "GRIB " : the file type is a GRIB file, coming from any of these models:
    1. "ECMWF " : european center forecast model
    2. "ARPEGE" : Arpege french forecast model
    3. "ALADIN" : Aladin french forecast local model
    4. "MOCAGE" : Mocage french research chemistry model
- NYEAR : year of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- NMONTH : month of surface UTC time. It is used only if no atmospheric file or no surface

file is given (in those the date can be read).

- **NDAY** : day of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read).
- **XTIME** : time from midnight of surface UTC time. It is used only if no atmospheric file or no surface file is given (in those the date can be read). (seconds).
- **LTEB\_CANOPY** : activates surface boundary multi layer scheme over town.



## 5.7.2. Namelist NAM\_PREP\_TEB\_SNOW

Fortran name	Fortran type	values	default value
CSNOW_ROOF	string of 6 characters	'1-L'	'1-L'
CSNOW_ROAD	string of 6 characters	'1-L'	'1-L'
XWSNOW_ROOF	real		none
XWSNOW_ROAD	real		none
XTSNOW_ROOF	real		none
XTSNOW_ROAD	real		none
XASNOW_ROOF	real		none
XASNOW_ROAD	real		none
XRSNOW_ROOF	real		none
XRSNOW_ROAD	real		none

- CSNOW\_ROAD : snow scheme used over roads
- CSNOW\_ROOF : snow scheme used over roofs
- XWSNOW\_ROOF : roof snow content
- XWSNOW\_ROAD : road snow content
- XTSNOW\_ROAD : road temperature
- XTSNOW\_ROOF : roof temperature
- XRSNOW\_ROOF : roof density
- XRSNOW\_ROAD : road density
- XASNOW\_ROAD : road albedo
- XASNOW\_ROOF : road albedo

### **5.7.3. Namelist NAM\_PREP\_TEB\_GARDEN**

duplication of NAM\_PREP\_ISBA to initialize vegetation in urban areas

## 5.7.4. Namelist NAM\_PREP\_GARDEN\_SNOW

duplication of NAM\_PREP\_ISBA\_SNOW to initialize vegetation in urban areas

## **6. How to run the externalized surface physical schemes**

Here are described the options available during the run of the several surface schemes.

## **6.1. "SURF\_ATM" general options available over all tiles**

## 6.1.1. Namelist NAM\_SURF\_CSTS

Fortran name	Fortran type	values	default value
XEMISSN	real		1.0
XANSMIN	real		0.5
XANSMAX	real		0.85
XAGLAMIN	real		0.8
XAGLAMAX	real		0.85
XALBWAT	real		0.135
XALBCOEF_TA96	real		0.037
XALBSCA_WAT	real		0.06
XEMISWAT	real		0.98
XALBWATICE	real		0.85
XEMISWATICE	real		1.0
XHGLA	real		33.3
XWSNV	real		5.0
XCFFV	real		3.0

- XEMISSN : snow emissivity
- XANSMIN : minimum value for snow albedo
- XANSMAX : maximum value for snow albedo
- XAGLAMIN : minimum value for permanent snow/ice albedo
- XAGLAMAX : maximum value for permanent snow/ice albedo
- XALBCOEF\_TA96 : coefficient used in the computation of albedo if 'TA96' option selected
- XALBSCA\_WAT : water diffuse albedo
- XEMISWAT: water emissivity
- XALBWATICE : sea ice albedo
- XEMISWATICE : sea ice emissivity
- XHGLA : Height of aged snow in glacier case (allows Pn=1)
- XWSNV : Coefficient for calculation of snow fraction over vegetation
- XCFFV : Coefficient for calculation of floodplain fraction over vegetation

## 6.1.2. Namelist NAM\_SURF\_ATM

Fortran name	Fortran type	values	default value
XCISMIN	real		6.7E-5
XVMODMIN	real		0.
LALDTHRES	logical		F
LALDZ0H	logical		F
LDRAG_COEF_ARP	logical		F
LNOSOF	logical		F
XEDB	real		5.
XEDC	real		5.
XEDD	real		5.
XEDK	real		1.
XUSURIC	real		1.
XUSURID	real		0.035
XUSURICL	real		4.
XVCHRNK	real		0.015
XVZ0CM	real		0.0
XRIMAX	real		0.0
XDELTA_MAX	real		1.0
LVZIUSTAR0_ARP	logical		F
LRRGUST_ARP	logical		F
XVZIUSTAR0	real		0.
XRZHZ0M	real		1.
XRRSCALE	real		1.15E-4
XRRGAMMA	real		0.8
XUTILGUST	real		0.125
LCPL_ARP	logical		F
LQVNPLUS	logical		F

- LALDTHRES: flag to set a minimum wind and shear like done in Aladin model.
- XCISMIN: minimum wind shear to compute turbulent exchange coefficient (used only if LALDTHRES)
- XVMODMIN: minimum wind speed to compute turbulent exchange coefficient (used only if LALDTHRES)
- LALDZ0H: to take into account orography in heat roughness length
- LDRAG\_COEF\_ARP: to use drag coefficient computed like in Arpege/Aladin models
- LNOSOF: no parameterization of subgrid orography effects on atmospheric forcing

- XEDB, XEDC, XEDD, XEDK: coefficients used in Richardson critical numbers computation
- XUSURIC, XUSURID, XUSURICL: Richardson critical numbers
- XVCHRNK, XVZ0CM: Charnock's constant and minimal neutral roughness length over sea (formulation of roughness length over sea)
- XRIMAX: limitation of Richardson number in drag computation
- XDELTA\_MAX: maximum fraction of the foliage covered by intercepted water for high vegetation
- LVZIUSTAR0\_ARP: flag to activate aladin formulation for zoh over sea
- LRRGUST\_ARP: flag to activate the correction of CD, CH, CDN due to moist gustiness
- XVZIUSTAR0: aladin formulation for zoh over sea
- XRZHZ0M: aladin formulation for zoh over sea
- XRRSCALE: aladin formulation for zoh over sea
- XRRGAMMA: aladin formulation for zoh over sea
- XUTILGUST: correction of CD, CH, CDN due to moist gustiness
- LCPL\_ARP: activate aladin formulation for Cp and L
- LQVNPLUS: An option for the resolution of the surface temperature equation (Arpege)



### 6.1.3. Namelist NAM\_WRITE\_SURF\_ATM

Fortran name	Fortran type	values	default value
LNOWRITE_COVERS	logical		F
LNOWRITE_CANOPY	logical		F
LNOWRITE_TEXFILE	logical		F

- LNOWRITE\_COVERS: if T, do not write covers in initial/restart or LBC files
- LNOWRITE\_CANOPY: if T, do not write canopy prognostic variables in initial/restart or LBC files
- LNOWRITE\_TEXFILE: if T, do not fill class\_cover\_data.tex file during the model setup

## 6.1.4. Namelist NAM\_SSO

The namelist NAM\_SSO\_n concerns the roughness parameterization for orography.

Fortran name	Fortran type	values	default value
CROUGH	string of 4 characters	'Z01D', 'Z04D', 'NONE', 'BE04'	BE04
XFRACZ0	real		2.

- CROUGH: type of orographic roughness length. The following options are currently available:
  - ◆ "Z01D" : orographic roughness length does not depend on wind direction
  - ◆ "Z04D" : orographic roughness length depends on wind direction
  - ◆ "BE04" : Beljaars 2004 orographic drag
  - ◆ "NONE" : no orographic treatment
- XFRACZ0 :  $Z0 = \text{Min}(Z0, H_{\text{ref}}/XFRACZ0)$

## 6.2. "SEAFLX" sea scheme options

## 6.2.1. Namelist NAM\_SEAFLUXn

Fortran name	Fortran type	values	default value
CSEA_FLUX	string of 6 characters	'DIRECT', 'ITERAT', 'COARE3', 'ECUME '	'ECUME '
CSEA_ALB	string of 4 characters	'UNIF', 'TA96'	'TA96'
LPWG	logical		F
LPRECIP	logical		F
LPWEBB	logical		F
LPROGSST	logical		F
NTIME_COUPLING	integer		
CINTERPOL_TS	string of 6 characters		'NONE '
XICHCE	real		0.

- CSEA\_FLUX: type of flux computation physics. The following option is currently available:
  - ◆ "DIRECT" : direct Charnock computation. No effect of convection in the the boundary layer on the fluxes formulae.
  - ◆ "ITERAT" : iterative method proposed by Fairall et al (1996) from TOGA-COARE experiment, amended by Mondon and Redelsperger (1998) to take into account effect of atmospheric convection on fluxes.
  - ◆ "COARE3" : iterative method proposed by Fairall et al (1996) from TOGA-COARE experiment, amended by cnrm/memo to take into account effect of atmospheric convection, precipitation and gustiness on fluxes.
  - ◆ "ECUME " : iterative method proposed by Fairall et al (1996) from TOGA-COARE experiment, amended by cnrm/memo to take into account effect of atmospheric convection, precipitation and gustiness on fluxes.  $\hat{\alpha}$  : improvement of surface exchange coefficients representation.
  - ◆ LPWG: correction of fluxes due to gustiness
  - ◆ LPRECIP: correction of fluxes due to precipitation
  - ◆ LPWEBB: correction of fluxes due to convection (Webb effect)
- CSEA\_ALB: type of albedo formula. The following options are currently available:
  - ◆ "UNIF" : a uniform value of 0.135 is used for water albedo
  - ◆ "TA96" : Taylor et al (1996) formula for water direct albedo, depending on solar zenith angle  $\theta$ :
 
$$\alpha_{dir} = 0.037 / (1.1 \cos(\theta)^{1.4} + 0.15)$$
- LPROGSST: set it to .TRUE. to make SST evolve with tendency when using the 1d oceanic model
- NTIME\_COUPLING: coupling time frequency between surface and the 1d oceanic model
- CINTERPOL\_TS: interpolate monthly SST to daily SST (function interpol\_quadra)
- XICHCE: coefficient used in the Ecume formulation (computation of exchange coefficients over sea)

## 6.2.2. Namelist NAM\_SURF\_SLT

Fortran name	Fortran type	values	default value
CEMISPARAM	string of 5 characters	'Vig01','Sch04'	'Sch04'

- "CEMISPARAM\_SLT" : One-line sea salt emission parameterization type. This namelist gives the distribution of emitted sea salt of SURFEX. For Each parameterization type, a geometric standard deviation and a median radius is given. See the code `init_sltn.f90` (MesoNH) or `init_sltn.mnh` (AROME, ALADIN) for values associated to these parameterizations. Note that if the default value is change, it is necessary to uses the same modes in the sea initialisation in the atmospheric model. It concerns the value of `XINIRADIUS_SLT` (initial radius), `XINISIG_SLT` (standard deviation) and `CRGUNITS` (mean radius definition) to have the same aerosol size distribution emitted and in the atmosphere. It is possible to do it directly in the fortran code (`modd_salt.mnh` in case of aladin/arome, `modd_salt.f90` for MesoNH) or for MesoNH only, change the values of these variables in `NAM_AERO_CONF` (`prep_real_case` or `prep_ideal_case`).

## 6.3. "WATFLX" inland water scheme options

### 6.3.1. Namelist NAM\_WATFLUXn

Fortran name	Fortran type	values	default value
CINTERPOL_TS	string of 6 characters		'NONE '
CWAT_ALB	string of 4 characters	'UNIF', 'TA96'	'UNIF'

- CWAT\_ALB: type of formulation used to set albedo over water
- CINTERPOL\_TS: interpolate monthly SST to daily SST (function interpol\_quadra)

## 6.4. "FLAKE" lake scheme options



## 6.4.1. Namelist NAM\_FLAKEn

Fortran name	Fortran type	values	default value
LSEDIMENTS	logical		T
CFLAKE_SNOW	string of 6 characters	'NON ', 'FLAKE ', 'ISBAES'	'NON '
CFLAKE_FLUX	string of 6 characters	'FLAKE ', 'ISBA '	'FLAKE '

- CFLAKE\_SNOW: snow scheme to be used. For the time being only option 'FLAKE ' is active
- CFLAKE\_FLUX: scheme to be used to compute surface fluxes of moment, energy and water vapor. For the time being only option 'FLAKE ' is active
- LSEDIMENTS: to use the bottom sediments scheme of Flake (default)

## 6.5. "ISBA " vegetation scheme options

## 6.5.1. Namelist NAM\_SGH\_ISBAn

Fortran name	Fortran type	values	default value
CRUNOFF	string of 4 characters	'WSAT', 'DT92', 'SGH'	'WSAT'
CTOPREG	string of 4 characters	'DEF', 'NON'	'DEF'
CKSAT	string of 4 characters	'DEF', 'SGH'	'DEF'
CRAIN	string of 3 characters	'DEF', 'SGH'	'DEF'
CHORT	string of 4 characters	'DEF', 'SGH'	'DEF'
LTRIP	logical		F
LFLOOD	logical		F

- CRUNOFF: type of subgrid runoff. The following options are currently available:
  - ◆ "WSAT" : runoff occurs only when saturation is reached
  - ◆ "DT92" : Dumenill and Todini (1992) subgrid runoff formula
  - ◆ "SGH" : Decharme et al. (2006) Topmodel like subgrid runoff
- CTOPREG: kind of regression. Option activated only if CRUNOFF = 'SGH'. The following options are currently available:
  - ◆ "DEF" : Wolock and MacCabe regression between topographic indices computed at 1km and 100m resolution (recommended)
  - ◆ "NON" : no regression
- CKSAT: Activates the exponential profile for Ksat. The following options are currently available:
  - ◆ "DEF" : homogeneous profile
  - ◆ "SGH" : exponential decreasing profile with depth (due to compaction of soil)
- CRAIN: Activates the spatial distribution of rainfall intensity. The following options are currently available:
  - ◆ "DEF" : homogeneous distribution
  - ◆ "SGH" : exponential distribution which depends on the fraction of the mesh where it rains. This fraction depends on the mesh resolution and the intensity of hourly precipitation. (If the horizontal mesh is lower than 10km then the fraction equals 1).
- CHORT: Activates the Horton runoff due to water infiltration excess. The following options are currently available:
  - ◆ "DEF" : no Horton runoff
  - ◆ "SGH" : Horton runoff computed
- LTRIP: Activates TRIP river routing model (RRM) scheme
- LFLOOD: Activates the flooding scheme

## 6.5.2. Namelist NAM\_ISBAn

Fortran name	Fortran type	values	default value
CC1DRY	string of 4 characters	'DEF ', 'GB93'	'DEF '
CSCOND	string of 4 characters	'NP89', 'PL98'	'PL98'
CSOILFRZ	string of 3 characters	'DEF', 'LWT'	'DEF'
CDIFSFCOND	string of 4 characters	'DEF ', 'MLCH'	'DEF '
CSNOWRES	string of 3 characters	'DEF', 'RIL'	'DEF'
CALBEDO	string of 4 characters	'MEAN', 'DRY ', 'WET ', 'EVOL'	'DRY '
CROUGH	string of 4 characters	'Z01D', 'Z04D', 'NONE', 'BE04', 'UNDE'	'UNDE'
CCPSURF	string of 3 characters	'DRY ', 'HUM'	'DRY '
LCANOPY_DRAG	logical		.FALSE.
LGLACIER	logical		.FALSE.
XTSTEP	real		none
XCGMAX	real		2.E-5
LVEGUPD	logical		T

- CC1DRY: type of C1 formulation for dry soils. The following options are currently available:
  - ◆ "DEF " : Giard-Bazile formulation
  - ◆ "GB93" : Giordani 1993, Braud 1993
- CSCOND: type of thermal conductivity. The following options are currently available:
  - ◆ "NP89" : Noilhan and Planton (1989) formula
  - ◆ "PL98" : Peters-Lidard et al. (1998) formula
- CSOILFRZ: type of soil freezing-physics option. The following options are currently available:
  - ◆ "DEF" : Boone et al. 2000<sup>Â</sup> ; Giard and Bazile 2000
  - ◆ "LWT" : Phase changes as above, but relation between unfrozen water and temperature considered
- CDIFSFCOND: type of Mulch effects. The following options are currently available:
  - ◆ "DEF " : no mulch effect
  - ◆ "MLCH" : include the insulating effect of leaf litter/mulch on the surf. thermal cond.
- CSNOWRES: type of turbulent exchanges over snow. The following options are currently available:
  - ◆ "DEF" : Louis
  - ◆ "RIL" : Maximum Richardson number limit for stable conditions ISBA-SNOW3L turbulent exchange option
- CALBEDO: type of bare soil albedo. The following options are currently available:
  - ◆ "DRY " : dry bare soil albedo
  - ◆ "WET " : wet bare soil albedo
  - ◆ "MEAN" : albedo for bare soil half wet, half dry

- ◆ "EVOL" : albedo of bare soil evolving with soil humidity
- CROUGH: type of orographic roughness length. The following options are currently available:
  - ◆ "Z01D" : orographic roughness length does not depend on wind direction
  - ◆ "Z04D" : orographic roughness length depends on wind direction
  - ◆ "BE04" : Beljaars 2004 orographic drag
  - ◆ "NONE" : no orographic treatment

By default, CROUGH is set to "UNDE" (undefined) since its initialization depends if "LISBA\_CANOPY" is activated or not, finally default value of CROUGH is updated as follows:

  - ◆ "CROUGH = Z04D" if "LISBA\_CANOPY" = F
  - ◆ "CROUGH = BE04" if "LISBA\_CANOPY" = T
- CCPSURF: type of specific heat at surface. The following options are currently available:
  - ◆ "DRY" : specific heat does not depend on humidity at surface
  - ◆ "HUM" : specific heat depends on humidity at surface.
- LCANOPY\_DRAG: drag activated in SBL scheme within the canopy
- LGLACIER: If activated, specific treatment (as in Arpege) over permanent snow/ice regions. Snow depth initialised to 10m and soil ice to porosity. During the run, snow albedo ranges from 0.8 to 0.85
- XTSTEP: time step for ISBA. Default is to use the time-step given by the atmospheric coupling (seconds).
- XCGMAX: maximum value for soil heat capacity.
- LVEGUPD: True = update vegetation parameters every decade

## 6.5.3. Namelist NAM\_SURF\_DST

Fortran name	Fortran type	values	default value
CEMISPARAM	string of 5 characters	'AMMA','Dal87','EXPLI','alf98','EXPLI'	'AMMA'
CVERMOD	string of 6 characters	'CMDVER'	'NONE'
XFLX_MSS_FDG_FCT	real		8.0e-4

- "CEMISPARAM" : One-line dust emission parameterization type. This namelist gives the distribution of emitted dust of SURFEX. For Each parameterization type, a geometric standard deviation and a median radius is given. Moreover, the repartition of mass flux could be derive from the friction velocity (case of "AMMA" or "EXPLI") or imposed (case of "Dal87", "alf98", "She84" or "PaG77"). See the code `init_dstn.f90` (MesoNH) or `init_dstn.mnh` (AROME, ALADIN) for values associated to these parameterizations. Note that if the default value is change, it is necessary to uses the same modes in the dust initialisation in the atmospheric model. It concerns the value of `XINIRADIUS` (initial radius), `XINISIG` (standard deviation) and `CRGUNITD` (mean radius definition) to have the same aerosol size distribution emitted and in the atmosphere. It is possible to do it directly in the fortran code (`modd_dust.mnh` in case of `aladin/arome`, `modd_dust.f90` for MesoNH) or for MesoNH only, change the values of these variables in `NAM_AERO_CONF` (`prep_real_case` or `prep_ideal_case`).
- "XFLX\_MSS\_FDG\_FCT" : Value of the  $\hat{I}_{\pm}$  factor representing the ratio of the vertical mass flux over the horizontal mass flux in the saltation layer (use only If `CVERMOD="NONE"`). This  $\hat{I}_{\pm}$  factor depend on the size distribution of the aerosol consider in the model.
- "CVERMOD" New parameterization of the dust emission formulation. In development, not recommended to uses it in this version.

## 6.5.4. Namelist NAM\_ASSIM

Declaration of keys for ISBA assimilation scheme (2DVAR, Bouyssel et al.).

Fortran name	Fortran type	values	default value
LASSIM	logical		F
CASSIM	character(LEN=5)	"PLUS", "AVERA", "2DVAR"	"PLUS "

- LASSIM: Assimilation or not
- CASSIM: type of correction

## 6.5.5. Namelist NAM\_AGRI

Agricultural Practices

Fortran name	Fortran type	values	default value
LAGRIP	logical		F

- LAGRIP : General switch for agricultural practices (seeding and irrigation)



## 6.5.6. Namelist NAM\_DEEPSOIL

deep soil characteristics

Fortran name	Fortran type	values	default value
LDEEPSOIL	logical		F
LPHYSDOMC	logical		F

- LDEEPSOIL: General switch for deep soil fields (temperature and relaxation time).  
XTDEEP\_CLI=(/236.,236.,220.,209.,206.,211.,214.,210.,207.,212.,220.,229./)  
XGAMMAT\_CLI=(/4.,4.,4.,3.,1.,2.,3.,1.,1.,1.,1.,2./)
- LPHYSDOMC : General switch to impose CT and soil water/ice contents CT(:) = 9.427757E-6

## 6.5.7. Namelist NAM\_TREEDRAG

Declaration to take into account tree drag in the atmospheric model instead of SURFEX. The Z0 forest is therefore reduced to the Z0 grass.

Fortran name	Fortran type	values	default value
LTREEDRAG	logical		F

- LTREEDRAG:flag used to take into account tree drag in the atmospheric model instead of SURFEX.

## 6.6. "TEB" town scheme options

## 6.6.1. Namelist NAM\_TEBn

Fortran name	Fortran type	values	default value
CZ0H	character(LEN=6)	'MASC95','BRUT82','KAND07'	'MASC95'

- CZ0H: TEB option for z0h roof & road:
  - ◆ 'MASC95' : Mascart et al 1995
  - ◆ 'BRUT82' : Brustart 1982
  - ◆ 'KAND07' : Kanda 2007

## 6.7. "IDEAL" ideal flux scheme options

## 6.6.1. Namelist NAM\_IDEAL\_FLUX

Fortran name	Fortran type	values	default value
XSFTH	real(NFORC+1)		0.
CSFTQ	character(LEN=7)	'kg/m2/s', 'W/m2 '	'kg/m2/s'
XSFTQ	real(NFORC+1)		0.
XSFCO2	real(NFORC+1)		
CUSTARTYPE	character(LEN=5)	'Z0 ', 'USTAR'	'Z0 '
XUSTAR	real(NFORC+1)		0.
XZ0	real		0.01
XALB	real		
XEMIS	real		1.
XTSRAD	real(NFORC+1)		XTT=273.15K

- NFORC : number of surface forcing instants. The default value is NFORC=48.
- XSFTH: hourly data of heat surface flux ( $W/m^2$ )
- CSFTQ: Unit for the evaporation flux ( $kg/m^2/s$ ) or ( $W/m^2$ )
- XSFTQ: hourly data of water vapor surface flux
- XSFCO2: hourly data of CO2 surface flux ( $kg/m^2/s$ )
- CUSTARTYPE: type of computation for friction
- XUSTAR: hourly data of friction ( $m^2/s^2$ )
- XZ0: roughness length (m)
- XALB: albedo (-)
- XEMIS:emissivity (-)
- XTSRAD: radiative temperature (K)

## 7. How to run the externalized surface chemical schemes

Here are described the options available during the run of the several schemes for emission and deposition of chemical species. Note that all the schemes for deposition and emission of chemical species do activate only if chemical species are present (i.e. if the coupling between atmosphere and surface include the chemical species concentrations and fluxes).

## **7.1. Chemical settings control**



## 7.1.1. Namelist NAM\_CH\_CONTROLn

Fortran name	Fortran type	values	default value
CCHEM_SURF_FILE	string of 28 characters		' '

- CCHEM\_SURF\_FILE: name of general (chemical) purpose ASCII input file.

## **7.2. Chemical anthropogenic emissions**

## 7.2.1. Namelist NAM\_CH\_SURFn

Fortran name	Fortran type	values	default value
LCH_SURF_EMIS	logical		.FALSE.

- LCH\_SURF\_EMIS: flag to use anthropogenic emissions or not.

## 7.3. Chemical deposition over ocean

### 7.3.1. Namelist NAM\_CH\_SEAFLUXn

Fortran name	Fortran type	values	default value
CCH_DRY_DEP	string of 6 characters	'NONE ', 'WES89 '	'WES89 '

- CCH\_DRY\_DEP: type of deposition scheme.
  - ◆ " NONE " : no chemical deposition scheme.
  - ◆ "WES89 " : Wesley (1989) deposition scheme.

## 7.4. Chemical deposition over lakes

## 7.4.1. Namelist NAM\_CH\_WATFLUXn

Fortran name	Fortran type	values	default value
CCH_DRY_DEP	string of 6 characters	'NONE ', 'WES89 '	'WES89 '

- CCH\_DRY\_DEP: type of deposition scheme.
  - ◆ " NONE " : no chemical deposition scheme.
  - ◆ "WES89 " : Wesley (1989) deposition scheme.

## 7.5. Chemical deposition over towns



## 7.5.1. Namelist NAM\_CH\_TEBn

Fortran name	Fortran type	values	default value
CCH_DRY_DEP	string of 6 characters	'NONE ', 'WES89 '	'WES89 '

- CCH\_DRY\_DEP: type of deposition scheme.
  - ◆ " NONE " : no chemical deposition scheme.
  - ◆ "WES89 " : Wesley (1989) deposition scheme.

## **7.6. Chemical deposition and biogenic emissions over vegetation**

## 7.6.1. Namelist NAM\_CH\_ISBAn

Fortran name	Fortran type	values	default value
CCH_DRY_DEP	string of 6 characters	'NONE ', 'WES89 '	'WES89 '
LCH_BIO_FLUX	logical		.FALSE.

- CCH\_DRY\_DEP: type of deposition scheme.
  - ◆ " NONE " : no chemical deposition scheme.
  - ◆ "WES89 " : Wesley (1989) deposition scheme.
- LCH\_BIO\_FLUX: flag to activate the biogenic emissions.

## 7.7. Chemical aerosol scheme (ORILAM)

## 7.7.1. Namelist NAM\_CHS\_ORILAM

Fortran name	Fortran type	values	default value
LCH_AERO_FLUX	logical		.FALSE.
LCO2PM	logical		.FALSE.
XEMISRADIUSI	real		0.05
XEMISRADIUSJ	real		0.2
XEMISSIGI	real		1.80
XEMISSIGJ	real		2.00
CRGUNIT	character	"MASS", "NUMB"	"NUMB"

- LCH\_AERO\_FLUX: switch to active aerosol surface flux for ORILAM
- "LCO2PM" : switch to activate emission of primary aerosol (Black and Organic carbon) compute from CO emission. Uses only if CO emission is defined in the surface field (see PREP\_PGD) and if there is no data for primary aerosol emission.
- "XEMISRADIUSI" : Aerosol flux, mean radius of aitken mode in  $\hat{I}/4m$  (only if LCH\_AERO\_FLUX=.TRUE.).
- "XEMISRADIUSJ" : Aerosol flux, mean radius of accumulation mode in  $\hat{I}/4m$  (only if LCH\_AERO\_FLUX=.TRUE.).
- "XEMISSIGI" : Aerosol flux, standard deviation of aitken mode in  $\hat{I}/4m$  (only if LCH\_AERO\_FLUX=.TRUE.).
- "XEMISSIGJ" : Aerosol flux, standard deviation of accumulation mode in  $\hat{I}/4m$  (only if LCH\_AERO\_FLUX=.TRUE.).
- "CRGUNIT" : Aerosol flux, Definition of XEMISRADIUSI or XEMISRADIUSJ : mean radius can be define in mass ("MASS") or in number ("NUMB").

## 8. Externalized surface diagnostics

The diagnostics for the surface require the call to the complete physics of the surface. Therefore, they can be computed either during the run of the schemes (in order to have for example continuous time series of these diagnostics), or can be computed at a given instant only, if atmospheric forcing is given at this instant for the surface scheme to do one time step. The cumulated diagnostics are of course significant only when computed during a run.

## **8.1. Diagnostics relative to the general surface monitor**

## 8.1.1. Namelist NAM\_DIAG\_SURF\_ATMn

Fortran name	Fortran type	default value
LFRAC	logical	.FALSE.
LDIAG_GRID	logical	.FALSE.

- LFRAC: flag to save in the output file the sea, inland water, natural covers and town fractions.
- LDIAG\_GRID: flag for mean grid diagnostics



## **8.2. Diagnostics relative to the general surface monitor and to each surface scheme**

## 8.2.1. Namelist NAM\_DIAG\_SURF<sub>n</sub>

Fortran name	Fortran type	values	default value
N2M	integer	0, 1, 2	0
LSURF_BUDGET	logical		.FALSE.
LSURF_BUDGETC	logical		.FALSE.
LRESET_BUDGETC	logical		.FALSE.
LRAD_BUDGET	logical		.FALSE.
LCOEF	logical		.FALSE.
LSURF_VARS	logical		.FALSE.
L2M_MIN_ZS	logical		.FALSE.

- N2M : flag to compute surface boundary layer characteristics:
  - ◆ N2M=1 : computes temperature at 2 m, specific humidity at 2 m, relative humidity, zonal and meridian wind at 10 m, and Richardson number. 2m and 10m quantities are calculated extrapolating atmospheric forcing variables with Paulson laws using surface heat, water and momentum fluxes.
  - ◆ N2M=2 : computes temperature at 2 m, specific humidity at 2 m, relative humidity, zonal and meridian wind at 10 m, and Richardson number. 2m and 10m quantities are calculated interpolating between atmospheric forcing variables and surface temperature and humidity.
- LSURF\_BUDGET: flag to save in the output file the terms of the surface energy balance (net radiation, sensible heat flux, latent heat flux, ground flux), for each scheme (on the four separate tiles), on each patch of the vegetation scheme if existing, and aggregated for the whole surface. The diagnosed fields are (\* stands for the scheme considered (\*=nothing : field aggregated on the whole surface;\*=name of a scheme : field for this scheme):
  - ◆ RN\_\* : net radiation
  - ◆ H\_\* : turbulent sensible heat flux
  - ◆ LE\_\* : turbulent latent heat flux
  - ◆ GFLUX\_\*: ground or storage heat flux
  - ◆ FMU\_\* : zonal wind stress
  - ◆ FMV\_\* : meridian wind stress

If both LSURF\_BUDGET and LRAD\_BUDGET are T then downward and upward shortwave radiation per spectral band will be written into output file (they are computed even if LRAD\_BUDGET is false). The following output fields are then available:

  - ◆ SWD\_\* : downward short wave radiation
  - ◆ SWU\_\*: upward short wave radiation
  - ◆ SWBD\_\* : downward short wave radiation for each spectral band
  - ◆ SWBU\_\* : upward short wave radiation for each spectral band
  - ◆ LWD\_\*:downward long wave radiation
  - ◆ LWU\_\* : upward long wave radiation

- LSURF\_BUDGETC: flag to save in the output file the time integrated values of all budget terms that have been activated
- LRESET\_BUDGETC: flag to reset cumulatives variables at the beginning of a run
- LCOEF: flag to save in the output file the transfer coefficients used in the computation of the surface energy fluxes, for each scheme (on the four separate tiles) and aggregated for the whole surface. The diagnosed fields are (\* stands for the scheme considered (\*=nothing : field aggregated on the whole surface; \*=name of a scheme : field for this scheme) :
  - ◆ CD\_\* : drag coefficient for momentum
  - ◆ CH\_\* : drag coefficient for heat
  - ◆ CE\_\* : drag coefficient for evaporation (differs from CH only over sea)
  - ◆ Z0\_\* : roughness length
  - ◆ ZOH\_\* : thermal roughness length
- LSURF\_VARS: flag to save in the output file the surface specific humidity for each scheme (on the four separate tiles), on each patch of the vegetation scheme if existing. The diagnosed fields are (\* stands for the scheme considered (\*=nothing : field aggregated on the whole surface; \*=name of a scheme :
- QS\_\* : specific humidity
  
- L2M\_MIN\_ZS: flag for 2 meters quantities evaluated on the minimum orography of the grid

## 8.2.2. Namelist NAM\_WRITE\_DIAG\_SURF<sub>n</sub>

Fortran name	Fortran type	values	default value
LSELECT	logical		.FALSE.
CSELECT	array of string of characters		
LPROVAR_TO_DIAG	logical		.FALSE.

- LSELECT: if true it indicates that a selection will be used as output .
- CSELECT: array containing the list of output fields .
- LPROVAR\_TO\_DIAG: used to write out prognostic variables like diagnostic one, on average over all patches.

## **8.3. Diagnostics relative to the ISBA vegetation scheme**

### 8.3.1. Namelist NAM\_DIAG\_ISBAn

Fortran name	Fortran type	default value
LPGD	logical	.FALSE.
LSURF_EVAP_BUDGET	logical	.FALSE.
LSURF_MISC_BUDGET	logical	.FALSE.
LPATCH_BUDGET	logical	.TRUE.
LWOOD_SPIN	logical	.FALSE.
LSOILCARB_SPIN	logical	.FALSE.

- LPGD : flag to save in the output file the physiographic fields of ISBA scheme that are computed from ecoclimap data from the ecosystem fractions.
- LSURF\_EVAP\_BUDGET: flag to save in the output file the detailed terms of the water vapor fluxes, on each patch of the vegetation scheme if existing, and aggregated for the natural surface. The diagnosed fields are:
- LSURF\_MISC\_BUDGET: flag to save in the output file miscellaneous fields. The diagnosed fields are:
  - ◆ HV : Halstead coefficient
  - ◆ SNG : snow fraction over bare ground
  - ◆ SNV : snow fraction over vegetation
  - ◆ SN : total snow fraction
  - ◆ SWI : soil wetness index for each ground layer  $(w_g - w_{wilt}) / (w_{fc} - w_{wilt})$  where  $w_g$  is the volumic water content,  $w_{fc}$  is the porosity and  $w_{wilt}$  corresponds to the plant wilting point.
  - ◆ GPP : Gross primary production
  - ◆ RDK : Dark respiration
- LPATCH\_BUDGET: flag to save in the output file the diagnostics for each patch (default is .T.)
- LWOOD\_SPIN: diagnostics related to ISBA-CC model
- LSOILCARB\_SPIN: diagnostics related to ISBA-CC model

## **8.4. Diagnostics relative to the TEB town scheme**

## 8.4.1. Namelist NAM\_DIAG\_TEBn

Fortran name	Fortran type	default value
LSURF_MISC_BUDGET	logical	.FALSE.
LPGD	logical	.FALSE.
LPGD_FIX	logical	.FALSE.

- **LSURF\_MISC\_BUDGET**: flag to save in the output file miscellaneous fields. The diagnosed fields are:
  - ◆ **Z0\_TOWN** : roughness length for town
  - ◆ **QF\_BLD** : domestic heating
  - ◆ **QF\_BLDWFR** : domestic heating
  - ◆ **FLX\_BLD** : heat flux from bld
  - ◆ **TI\_BLD\_EQ** : internal temperature without heating
  - ◆ **TI\_BLDWFR** : internal temperature without heating
  - ◆ **QF\_TOWN** : total anthropogenic heat
  - ◆ **DQS\_TOWN** : storage inside building
  - ◆ **H\_WALL** : wall sensible heat flux
  - ◆ **H\_ROOF**: roof sensible heat flux
  - ◆ **H\_ROAD**: road sensible heat flux
  - ◆ **RN\_WALL** : net radiation at wall
  - ◆ **RN\_ROOF** : net radiation at roof
  - ◆ **RN\_ROAD**: net radiation at road
  - ◆ **GFLUX\_WALL** : net wall conduction flux
  - ◆ **GFLUX\_ROOF** : net roof conduction flux
  - ◆ **GFLUX\_ROAD** : net road conduction flux
  - ◆ **LE\_ROOF** : roof latent heat flux
  - ◆ **LE\_ROAD** : road latent heat flux
- **LPGD** : flag to save PGD fields if TEB garden is activated
- **LPGD\_FIX** : flag to save fixed PGD fields if TEB garden is activated



## **8.5. Diagnostics relative to the FLAKE scheme**

## 8.5.1. Namelist NAM\_DIAG\_FLAKEn

Fortran name	Fortran type	default value
LWATER_PROFILE	logical	.FALSE.
XZWAT_PROFILE	real	

- LWATER\_PROFILE: flag to save in the output file miscellaneous fields. The diagnostic is temperature at the depths defined by:
  - ◆ XZWAT\_PROFILE : depth of output levels (m) in namelist

## **8.6. Diagnostics relative to the 1D oceanic scheme**

## 8.6.1. Namelist NAM\_DIAG\_OCEANn

Fortran name	Fortran type	default value
LDIAG_OCEAN	logical	.FALSE.

- LDIAG\_OCEAN: flag for ocean variables

## 9. Externalized surface model output fields

Model output fields depend on the tile and on the configuration of run.

## 9.1. Prognostic model output fields

## 9.1.1. ISBA

The definition of the representation of soil, vegetation, snow and surface boundary layer components is done during PGD and PREP. The description of soil is done with parameter CISBA (2 layers, 3 layers or more if diffusion treatment selected) from NAM\_ISBA. The representation of vegetation is done with parameters NPATCH (number of patches over vegetation) and CPHOTO (type of photosynthesis). During PREP, the snow scheme is chosen by parameter CSNOW in NAM\_PREP\_ISBA and if the surface boundary layer (SBL) scheme is activated then LISBA\_CANOPY key has to be set to T.

name	soil (CISBA)				vegetation (CPHOTO)			snow (CSNOW)			SBL
	layer	2-L	3-L	DIF	AST/AGS	LAI/LST	NIT	layer	D95/EBA	DIF	layer 1 to 6
	1	*	*	*							
TG	2	*	*	*							
	$n \geq 3$			*							
	1	*	*	*							
WG	2	*	*	*							
	3		*	*							
	$n \geq 4$			*							
	1	*	*	*							
WGI	2	*	*	*							
	$n \geq 3$			*							
WR		*	*	*							
RESA		*	*	*							
LAI					*	*	*				
AN					*	*	*				
ANDAY					*	*	*				
ANFM					*	*	*				
LE_AGS					*	*	*				
RESPBSTR							*				
RESPBSTR2							*				
BIOMASSTR2							*				
BIOMASSTR2_LAST							*				
BIOMASSTR_LAST							*				
								1	*	*	
WSNOW_VEG								2		*	

								3		☼	
								1	☼	☼	
RSNOW_VEG								2		☼	
								3		☼	
								1		☼	
HSNOW_VEG								2		☼	
								3		☼	
TSNOW_VEG								1	☼		
ASNOW_VEG									☼	☼	
ISBA_CAN_U											☼
ISBA_CAN_T											☼
ISBA_CAN_Q											☼
ISBA_CAN_E											☼
ISBA_CAN_P											☼

variable	dimension	unit	description
TG1	[2D]	[K]	surface temperature
TG2	[2D]	[K]	deep soil temperature
TG3	[2D]	[K]	third layer temperature
WG1	[2D]	[m3/m3]	surface liquid volumetric water content
WG2	[2D]	[m3/m3]	root liquid volumetric water content
WG3	[2D]	[m3/m3]	deep liquid volumetric water content
WGI1	[2D]	[m3/m3]	surface frozen volumetric water content
WGI2	[2D]	[m3/m3]	root frozen volumetric water content
WGI3	[2D]	[m3/m3]	third layer frozen volumetric water content
WR	[2D]	[Kg/m2]	liquid water retained by the foliage
RESA	[2D]	[s/m]	aerodynamical resistance
LAI	[2D]	[m2/m2]	leaf area index
AN	[2D]	[kgCO2/m2/s]	CO2 net assimilation
ANDAY	[2D]	[kgCO2/m2]	daily CO2 net assimilation
ANFM	[2D]	[mg/m2/s]	maximum leaf assimilation
LE_AGS	[2D]	[W/m2]	latent heat flux
RESPBSTR	[2D]	[W/m2]	respiration of above-ground structural biomass
RESPBSTR2	[2D]	[W/m2]	respiration of deep structural biomass
BIOMASSTR2	[2D]	[Kg/m2]	total dry above-ground structural biomass
BIOMASSTR2_LAST	[2D]	[Kg/m2]	total dry below-ground structural biomass of day-1
BIOMASSTR_LAST	[2D]	[Kg/m2]	total dry above-ground structural biomass of day-1
TSNOW_VEG1	[2D]	[K]	1st layer snow temperature



WSNOW_VEG1	[2D]	[Kg/m2]	1st layer snow water equivalent
RSNOW_VEG1	[2D]	[Kg/m3]	1st layer snow density
HSNOW_VEG1	[2D]	[W/m2]	1st layer snow heat content
ASNOW_VEG	[2D]	[-]	snow albedo
WSNOW_VEG2	[2D]	[Kg/m2]	2nd layer snow water equivalent
RSNOW_VEG2	[2D]	[Kg/m3]	2nd layer snow density
HSNOW_VEG2	[2D]	[W/m2]	2nd layer snow heat content
WSNOW_VEG3	[2D]	[Kg/m2]	3rd layer snow water equivalent
RSNOW_VEG3	[2D]	[Kg/m3]	3rd layer snow density
HSNOW_VEG3	[2D]	[W/m2]	3rd layer snow heat content
ISBA_CAN_Un	[1D]	[m/s]	wind in canopy at layer n (n=1 to 6)
ISBA_CAN_Tn	[1D]	[m/s]	temperature in canopy at layer n (n=1 to 6)
ISBA_CAN_Qn	[1D]	[m/s]	humidity in canopy at layer n (n=1 to 6)
ISBA_CAN_En	[1D]	[m/s]	TKE in canopy at layer n (n=1 to 6)
ISBA_CAN_Pn	[1D]	[m/s]	pressure in canopy at layer n (n=1 to 6)

## 9.1.2. SEAFLUX

	SEAFLUX	SBL
SST	*	
ZOSEA	*	
SEA_SBL_U		*
SEA_SBL_T		*
SEA_SBL_Q		*
SEA_SBL_E		*
SEA_SBL_P		*

## 9.1.3. TEB

name	Standard TEB		CANOPY
	layer		
	1	☛	
T_ROOF	2	☛	
	3	☛	
WS_ROOF		☛	
	1	☛	
T_ROAD	2	☛	
	3	☛	
WS_ROAD		☛	
	1	☛	
T_WALL	2	☛	
	3	☛	
TI_BLD		☛	
TI_ROAD		☛	
WSNOW_ROOF	1	☛	
RSNOW_ROOF	1	☛	
TSNOW_ROOF	1	☛	
ASNOW_ROOF		☛	
WSNOW_ROAD	1	☛	
RSNOW_ROAD	1	☛	
TSNOW_ROAD	1	☛	
ASNOW_ROAD		☛	
T_CANYON		☛	
Q_CANYON		☛	
TEB_CAN_U			☛
TEB_CAN_T			☛
TEB_CAN_Q			☛
TEB_CAN_E			☛
TEB_CAN_P			☛

## 9.1.4. WATFLUX

	WATERFLUX	SBL
TS_WATER	*	
ZOWATER	*	
WAT_SBL_U		*
WAT_SBL_T		*
WAT_SBL_Q		*
WAT_SBL_E		*
WAT_SBL_P		*

## 9.1.5. FLAKE

	FLAKE	SBL
TS_WATER	*	
T_SNOW	*	
T_ICE	*	
T_MNW	*	
T_WML	*	
T_BOT	*	
T_B1	*	
CT	*	
H_SNOW	*	
H_ICE	*	
H_ML	*	
H_B1	*	
WAT_SBL_U		*
WAT_SBL_T		*
WAT_SBL_Q		*
WAT_SBL_E		*
WAT_SBL_P		*

## 10. Example of namelist features

## 10.1. How to define a target grid

```
&NAM_PGDFILE          CPGDFILE='PGDFILE.2.5km_AROME_FRANCE'  
/  
&NAM_PGD_GRID        CGRID = 'CONF PROJ '  
/  
&NAM_CONF_PROJ      XLAT0=46.401460686331625,  
                    XLON0=2.20000000000000273,  
                    XRPK=0.7241894422,  
                    XBETA=0.00  
/  
&NAM_CONF_PROJ_GRID XLATCEN=46.401460686331625,  
                    XLONCEN=2.20000000000000273,  
                    NIMAX=588, NJMAX=500,  
                    XDX=2499.7648911167489,  
                    XDY=2499.7648911167489  
/  
/
```

## 10.2. How to use ECOCLIMAP I

This is the classical way how to use ecoclimap and other databases (orography, sand and clay). In previous version, the name of ecoclimap database was ecoclimats\_v2, it has been replaced by ECOCLIMAP\_I\_GLOBAL.

```
&NAM_FRAC          LECOCLIMAP = T
/
&NAM_COVER         YCOVER      = 'ECOCLIMAP_I_GLOBAL' ,
                   YFILETYPE = 'DIRECT'
/
&NAM_ZS           YZS         = 'gtopo30' ,
                   YFILETYPE = 'DIRECT'
/
&NAM_ISBA         YCLAY       = 'clay_fao' ,
                   YCLAYFILETYPE = 'DIRECT' ,
                   YSAND       = 'sand_fao' ,
                   YSANDFILETYPE = 'DIRECT' ,
                   CISBA       = '3-L' ,
                   CPHOTO      = 'NON' ,
                   NPATCH     = 1 ,
                   NGROUND_LAYER = 3
/
```



## 10.3. How to use ECOCLIMAP II

```
&NAM_FRAC          LECOCLIMAP = T
/
&NAM_COVER         YCOVER      = 'ECOCLIMAP_II_EUROP'
/                   YFILETYPE  = 'DIRECT'
/                   ,
/                   ,
```

## 10.4. How to use 1D Oceanic Model

```
&NAM_SEABATHY          YSEABATHY          = & etopo2.ncâ ,
                       YSEABATHYFILETYPE = & NETCDFâ ,
                       YNCVARIABLE       = & topoâ
/
&NAM_PREP_SEAFLUX      CFILE_SEAFLX        = & mercator_20031203.ncâ ,
                       CTYPE             = & NETCDFâ ,
                       LOCEAN_MERCATOR   = T,
                       LOCEAN_CURRENT    = F
/
&NAM_SEAFLUXn          CSEA_ALB = "TA96" ,
                       LPROGSST = T
/
```

## 10.5. How to initialize variables from grib file

```
&NAM_PREP_SURF_ATM      CFILE          = 'arpifs.AN.20030101.00' ,
                        CFILETYPE       = 'GRIB   '
/
&NAM_PREP_TEB           CFILE_TEB       = 'arpifs.AN.20030101.00' ,
                        CTYPE           = 'GRIB   '
/
&NAM_PREP_SEAFLUX      CFILE_SEAFLX    = 'arpifs.AN.20030101.00' ,
                        CTYPE           = 'GRIB   '
/
&NAM_PREP_WATFLUX      CFILE_WATFLX    = 'arpifs.AN.20030101.00' ,
                        CTYPE           = 'GRIB   '
/
&NAM_PREP_ISBA          CFILE_ISBA      = 'arpifs.AN.20030101.00' ,
                        CTYPE           = 'GRIB   '
/
&NAM_PREP_ISBA_SNOW    CSNOW           = '3-L'
```

## 10.6. How to initialize main ISBA scheme options

```
&NAM_SGH_ISBAn      CRUNOFF      = "WSAT"
/
&NAM_ISBAn          CROUGH       = "Z04D"      ,
                    CSCOND      = "NP89"      ,
                    CALBEDO     = "DRY"       ,
                    CC1DRY      = 'DEF '      ,
                    CSOILFRZ    = 'DEF '      ,
                    CDIFSFCOND  = 'DEF '      ,
                    CCPSURF     = 'DRY '      ,
                    CSNOWRES    = 'DEF '
/
&NAM_CH_ISBAn       CCH_DRY_DEP = "WES89 "
/
```

## 10.7. How to get lake temperature profile as output

```
&NAM_DIAG_FLAKEn          LWATER_PROFILE = T ,
                           XZWAT_PROFILE (1) = 1. ,
                           XZWAT_PROFILE (2) = 2. ,
                           XZWAT_PROFILE (3) = 3. ,
                           XZWAT_PROFILE (4) = 4. ,
                           XZWAT_PROFILE (5) = 5. ,
                           XZWAT_PROFILE (6) = 6. ,
                           XZWAT_PROFILE (7) = 7.

/
&NAM_WRITE_DIAG_SURFm     LSELECT = T ,
                           CNAME_SELECT (1) = 'TW_1.0'
                           CNAME_SELECT (2) = 'TW_2.0'
                           CNAME_SELECT (3) = 'TW_3.0'
                           CNAME_SELECT (4) = 'TW_4.0'
                           CNAME_SELECT (5) = 'TW_5.0'
                           CNAME_SELECT (6) = 'TW_6.0'
                           CNAME_SELECT (7) = 'TW_7.0'

&END
```

## 10.8. user defined surface parameters

Ecoclimap is not used (LECOCLIMAP = F). Information is not read from databases but the user defines his own surface parameters. Uniform field is used in this 1D case.

## 10.8.1. Uniform values prescribed: 1d example without patches

```
&NAM_DATA_ISBA          NTIME = 12 ,
XUNIF_VEGTYPE (1)      = 0.,
XUNIF_VEGTYPE (2)      = 0.,
XUNIF_VEGTYPE (3)      = 0.,
XUNIF_VEGTYPE (4)      = 0.,
XUNIF_VEGTYPE (5)      = 1.,
XUNIF_VEGTYPE (6)      = 0.,
XUNIF_VEGTYPE (7)      = 0.,
XUNIF_VEGTYPE (8)      = 0.,
XUNIF_VEGTYPE (9)      = 0.,
XUNIF_VEGTYPE (10)     = 0.,
XUNIF_VEGTYPE (11)     = 0.,
XUNIF_VEGTYPE (12)     = 0.,
XUNIF_VEG (1,1)        = 1.,
XUNIF_VEG (1,2)        = 1.,
XUNIF_VEG (1,3)        = 1.,
XUNIF_VEG (1,4)        = 1.,
XUNIF_VEG (1,5)        = 1.,
XUNIF_VEG (1,6)        = 1.,
XUNIF_VEG (1,7)        = 1.,
XUNIF_VEG (1,8)        = 1.,
XUNIF_VEG (1,9)        = 1.,
XUNIF_VEG (1,10)       = 1.,
XUNIF_VEG (1,11)       = 1.,
XUNIF_VEG (1,12)       = 1.,
XUNIF_LAI (1,1)        = 0.789,
XUNIF_LAI (1,2)        = 0.213,
XUNIF_LAI (1,3)        = 0.345,
XUNIF_LAI (1,4)        = 0.467,
XUNIF_LAI (1,5)        = 0.888,
XUNIF_LAI (1,6)        = 0.621,
XUNIF_LAI (1,7)        = 0.743,
XUNIF_LAI (1,8)        = 0.855,
XUNIF_LAI (1,9)        = 0.976,
XUNIF_LAI (1,10)       = 0.123,
XUNIF_LAI (1,11)       = 0.134,
XUNIF_LAI (1,12)       = 0.155,
XUNIF_Z0 (1,1)         = 2.5,
XUNIF_Z0 (1,2)         = 2.5,
XUNIF_Z0 (1,3)         = 2.5,
XUNIF_Z0 (1,4)         = 2.5,
XUNIF_Z0 (1,5)         = 2.5,
XUNIF_Z0 (1,6)         = 2.5,
XUNIF_Z0 (1,7)         = 2.5,
XUNIF_Z0 (1,8)         = 2.5,
XUNIF_Z0 (1,9)         = 2.5,
XUNIF_Z0 (1,10)        = 2.5,
XUNIF_Z0 (1,11)        = 2.5,
XUNIF_Z0 (1,12)        = 2.5,
XUNIF_EMIS (1,1)       = 0.97,
XUNIF_EMIS (1,2)       = 0.97,
```

```

XUNIF_EMIS (1, 3)      = 0.97,
XUNIF_EMIS (1, 4)      = 0.97,
XUNIF_EMIS (1, 5)      = 0.97,
XUNIF_EMIS (1, 6)      = 0.97,
XUNIF_EMIS (1, 7)      = 0.97,
XUNIF_EMIS (1, 8)      = 0.97,
XUNIF_EMIS (1, 9)      = 0.97,
XUNIF_EMIS (1, 10)     = 0.97,
XUNIF_EMIS (1, 11)     = 0.97,
XUNIF_EMIS (1, 12)     = 0.97,
XUNIF_DG (1, 1)        = 0.01,
XUNIF_DG (1, 2)        = 0.70,
XUNIF_DG (1, 3)        = 1.30,
XUNIF_ROOTFRAC (1, 1)  = -999.,
XUNIF_ROOTFRAC (1, 2)  = -999.,
XUNIF_ROOTFRAC (1, 3)  = -999.,
XUNIF_RSMIN (1)        = 150.,
XUNIF_GAMMA (1)        = 0.04,
XUNIF_WRMX_CF (1)      = 0.1,
XUNIF_RGL (1)          = 30.,
XUNIF_CV (1)           = 0.00001,
XUNIF_Z0_O_Z0H (1)    = 10.,
XUNIF_ALBNIR_VEG (1)   = 0.15,
XUNIF_ALBVIS_VEG (1)   = 0.05,
XUNIF_ALBUV_VEG (1)    = 0.0425,
XUNIF_ALBNIR_SOIL (1)  = 0.15,
XUNIF_ALBVIS_SOIL (1)  = 0.05,
XUNIF_ALBUV_SOIL (1)   = 0.0425,
XUNIF_GMES (1)         = 0.001,
XUNIF_RE25 (1)         = 0.00000015,
XUNIF_BSLAI (1)        = 0.25,
XUNIF_LAIMIN (1)       = 1.0,
XUNIF_SEFOLD (1)       = 31536000.,
XUNIF_GC (1)           = 0.,
XUNIF_DMAX (1)         = 0.1,
XUNIF_F2I (1)          = 0.3,
XUNIF_H_TREE (1)       = 20.,
XUNIF_CE_NITRO (1)     = 4.85,
XUNIF_CF_NITRO (1)     = -0.24,
XUNIF_CNA_NITRO (1)    = 2.8,

```

/

&NAM\_DATA\_TEB

```

NROOF_LAYER           = 3,
XUNIF_ALB_ROOF        = 0.2,
XUNIF_EMIS_ROOF       = 0.97,
XUNIF_HC_ROOF (1)     = 2110000.,
XUNIF_HC_ROOF (2)     = 2800000.,
XUNIF_HC_ROOF (3)     = 2900000.,
XUNIF_TC_ROOF (1)     = 1.51,
XUNIF_TC_ROOF (2)     = 0.08,
XUNIF_TC_ROOF (3)     = 0.05,
XUNIF_D_ROOF (1)      = 0.05,
XUNIF_D_ROOF (2)      = 0.4,
XUNIF_D_ROOF (3)      = 0.1,
NROAD_LAYER            = 3,
XUNIF_ALB_ROAD        = 0.2,
XUNIF_EMIS_ROAD       = 0.97,

```



```

XUNIF_HC_ROAD(1) = 2110000.,
XUNIF_HC_ROAD(2) = 2800000.,
XUNIF_HC_ROAD(3) = 2900000.,
XUNIF_TC_ROAD(1) = 1.51,
XUNIF_TC_ROAD(2) = 0.08,
XUNIF_TC_ROAD(3) = 0.05,
XUNIF_D_ROAD(1) = 0.05,
XUNIF_D_ROAD(2) = 0.4,
XUNIF_D_ROAD(3) = 0.1,
NWALL_LAYER      = 3,
XUNIF_ALB_WALL   = 0.2,
XUNIF_EMIS_WALL  = 0.97,
XUNIF_HC_WALL(1) = 2110000.,
XUNIF_HC_WALL(2) = 2800000.,
XUNIF_HC_WALL(3) = 2900000.,
XUNIF_TC_WALL(1) = 1.51,
XUNIF_TC_WALL(2) = 0.08,
XUNIF_TC_WALL(3) = 0.05,
XUNIF_D_WALL(1)  = 0.05,
XUNIF_D_WALL(2)  = 0.4,
XUNIF_D_WALL(3)  = 0.1,
XUNIF_Z0_TOWN    = 1.,
XUNIF_BLD        = 0.5,
XUNIF_BLD_HEIGHT = 10.,
XUNIF_WALL_O_HOR = 0.5,
XUNIF_H_TRAFFIC  = 10.,
XUNIF_LE_TRAFFIC = 0.,
XUNIF_H_INDUSTRY = 5.,
XUNIF_LE_INDUSTRY = 0.

/
&NAM_FRAC          LECOCLIMAP = F,
                   XUNIF_SEA    = 0.,
                   XUNIF_WATER   = 0.5,
                   XUNIF_TOWN    = 0.,
                   XUNIF_NATURE  = 0.5

/
&NAM_PGD_GRID      CGRID = 'LONLAT REG'

/
&NAM_LONLAT_REG    XLONMIN = 0.          ,
                   XLONMAX = 0.          ,
                   XLATMIN = 0.          ,
                   XLATMAX = 0.          ,
                   NLON    = 1           ,
                   NLAT   = 1           ,

/
&NAM_PGD_SCHEMES   CNATURE = 'ISBA  ' ,
                   CSEA    = 'SEAFIX' ,
                   CTOWN   = 'TEB  ' ,
                   CWATER  = 'WATFIX'

/
&NAM_ZS            XUNIF_ZS = 0.

/
&NAM_ISBA          XUNIF_CLAY   = 0.4      ,
                   XUNIF_SAND   = 0.2      ,
                   XUNIF_RUNOFFB = 0.5      ,
                   CISBA       = '3-L'     ,

```

```

        CPHOTO          = 'NON'          ,
        NPATCH         = 1              ,
        NGROUND_LAYER  = 3
/
&NAM_PREPFILE        CPREPFILE = 'PREP'
/
&NAM_PREP_SURF_ATM  NYEAR  = 2004,
                    NMONTH = 10,
                    NDAY   = 25,
                    XTIME  = 21600.
/
&NAM_PREP_SEAFLUX   XSST_UNIF = 285.,
                    NYEAR  = 2004,
                    NMONTH = 10,
                    NDAY   = 25,
                    XTIME  = 21600.
/
&NAM_PREP_WATFLUX   XTS_WATER_UNIF = 285.,
                    NYEAR  = 2004,
                    NMONTH = 10,
                    NDAY   = 25,
                    XTIME  = 21600.
/
&NAM_PREP_TEB       XTI_ROAD= 285.,
                    XTI_BLD = 285.,
                    XTS_ROAD= 285.,
                    XTS_ROOF= 285.,
                    XTS_WALL= 285.,
                    XWS_ROAD= 0.,
                    XWS_ROOF= 0.,
                    NYEAR  = 2004,
                    NMONTH = 10,
                    NDAY   = 25,
                    XTIME  = 21600.
/
&NAM_PREP_ISBA      XHUG_SURF = 0.2,
                    XHUG_ROOT = 0.2,
                    XHUG_DEEP = 0.2,
                    XTG_SURF  = 285.,
                    XTG_ROOT  = 288.,
                    XTG_DEEP  = 292.,
                    NYEAR  = 2004,
                    NMONTH = 10,
                    NDAY   = 25,
                    XTIME  = 21600.
/
&NAM_PREP_ISBA_SNOW CSNOW = '3-L'
/

&NAM_IO_OFFLINE     LPRINT  = T          ,
                    CFORCING_FILETYPE = 'NETCDF' ,
                    CSURF_FILETYPE   = 'LFI'   ,
                    CTIMESERIES_FILETYPE = 'NETCDF' ,
                    LWRITE_COORD = T,
                    LSET_FORC_ZS=T

```

```

/

&NAM_DIAG_SURFm      LSURF_BUDGET = T      ,
                      N2M      = 1

/
&NAM_DIAG_SURF_ATMn  LFRAC      = T

/
&NAM_DIAG_ISBAn      LPGD          = T ,
                      LSURF_EVAP_BUDGET = T ,
                      LSURF_MISC_BUDGET = T

/
&NAM_DIAG_TEBn       LSURF_MISC_BUDGET = T

/
&NAM_SGH_ISBAn       CRUNOFF      = "WSAT"

/
&NAM_ISBAn           CROUGH       = "Z04D"      ,
                      CSCOND      = "NP89"      ,
                      CALBEDO     = "DRY"       ,
                      CC1DRY      = 'DEF '      ,
                      CSOILFRZ    = 'DEF '      ,
                      CDIFSFCOND  = 'DEF '      ,
                      CSNOWRES     = 'DEF '      ,
                      CCPSURF     = 'DRY'

/
&NAM_CH_ISBAn        CCH_DRY_DEP = "WES89 "

/
&NAM_SEAFLUXn        CSEA_ALB    = "TA96"

/
&NAM_CH_SEAFLUXn     CCH_DRY_DEP = "WES89 "

/
&NAM_CH_WATFLUXn     CCH_DRY_DEP = "WES89 "

/
&NAM_CH_TEBn         CCH_DRY_DEP = "WES89 "

/

```

## 10.8.2. Uniform values prescribed: 1d example with patches

```

&NAM_DATA_ISBA
      NTIME = 12 ,
      XUNIF_VEGTYPE (1)      = 0.,
      XUNIF_VEGTYPE (2)      = 0.,
      XUNIF_VEGTYPE (3)      = 0.,
      XUNIF_VEGTYPE (4)      = 0.,
      XUNIF_VEGTYPE (5)      = 0.,
      XUNIF_VEGTYPE (6)      = 0.,
      XUNIF_VEGTYPE (7)      = 1.,
      XUNIF_VEGTYPE (8)      = 0.,
      XUNIF_VEGTYPE (9)      = 0.,
      XUNIF_VEGTYPE (10)     = 0.,
      XUNIF_VEGTYPE (11)     = 0.,
      XUNIF_VEGTYPE (12)     = 0.,
      XUNIF_VEG (1,1)        = 0.,      XUNIF_VEG (5,1)      = 0.,
      XUNIF_VEG (1,2)        = 0.,      XUNIF_VEG (5,2)      = 0.,
      XUNIF_VEG (1,3)        = 0.,      XUNIF_VEG (5,3)      = 0.,
      XUNIF_VEG (1,4)        = 0.,      XUNIF_VEG (5,4)      = 0.,
      XUNIF_VEG (1,5)        = 0.5,     XUNIF_VEG (5,5)      = 0.5,
      XUNIF_VEG (1,6)        = 0.9,     XUNIF_VEG (5,6)      = 0.9,
      XUNIF_VEG (1,7)        = 0.9,     XUNIF_VEG (5,7)      = 0.9,
      XUNIF_VEG (1,8)        = 0.9,     XUNIF_VEG (5,8)      = 0.9,
      XUNIF_VEG (1,9)        = 0.9,     XUNIF_VEG (5,9)      = 0.9,
      XUNIF_VEG (1,10)       = 0.,      XUNIF_VEG (5,10)     = 0.,
      XUNIF_VEG (1,11)       = 0.,      XUNIF_VEG (5,11)     = 0.,
      XUNIF_VEG (1,12)       = 0.,      XUNIF_VEG (5,12)     = 0.,
      XUNIF_LAI (1,1)        = 0.,      XUNIF_LAI (5,1)      = 0.,
      XUNIF_LAI (1,2)        = 0.,      XUNIF_LAI (5,2)      = 0.,
      XUNIF_LAI (1,3)        = 0.,      XUNIF_LAI (5,3)      = 0.,
      XUNIF_LAI (1,4)        = 0.,      XUNIF_LAI (5,4)      = 0.,
      XUNIF_LAI (1,5)        = 1.,      XUNIF_LAI (5,5)      = 1.,
      XUNIF_LAI (1,6)        = 3.,      XUNIF_LAI (5,6)      = 3.,
      XUNIF_LAI (1,7)        = 3.,      XUNIF_LAI (5,7)      = 3.,
      XUNIF_LAI (1,8)        = 3.,      XUNIF_LAI (5,8)      = 3.,
      XUNIF_LAI (1,9)        = 3.,      XUNIF_LAI (5,9)      = 3.,
      XUNIF_LAI (1,10)       = 0.,      XUNIF_LAI (5,10)     = 0.,
      XUNIF_LAI (1,11)       = 0.,      XUNIF_LAI (5,11)     = 0.,
      XUNIF_LAI (1,12)       = 0.,      XUNIF_LAI (5,12)     = 0.,
      XUNIF_Z0 (1,1)         = 0.01,     XUNIF_Z0 (5,1)       = 0.01,
      XUNIF_Z0 (1,2)         = 0.01,     XUNIF_Z0 (5,2)       = 0.01,
      XUNIF_Z0 (1,3)         = 0.01,     XUNIF_Z0 (5,3)       = 0.01,
      XUNIF_Z0 (1,4)         = 0.01,     XUNIF_Z0 (5,4)       = 0.01,
      XUNIF_Z0 (1,5)         = 0.05,     XUNIF_Z0 (5,5)       = 0.05,
      XUNIF_Z0 (1,6)         = 0.15,     XUNIF_Z0 (5,6)       = 0.15,
      XUNIF_Z0 (1,7)         = 0.15,     XUNIF_Z0 (5,7)       = 0.15,
      XUNIF_Z0 (1,8)         = 0.15,     XUNIF_Z0 (5,8)       = 0.15,
      XUNIF_Z0 (1,9)         = 0.15,     XUNIF_Z0 (5,9)       = 0.15,
      XUNIF_Z0 (1,10)        = 0.01,     XUNIF_Z0 (5,10)      = 0.01,
      XUNIF_Z0 (1,11)        = 0.01,     XUNIF_Z0 (5,11)      = 0.01,
      XUNIF_Z0 (1,12)        = 0.01,     XUNIF_Z0 (5,12)      = 0.01,
      XUNIF_EMIS (1,1)       = 0.98,     XUNIF_EMIS (5,1)     = 0.98,
      XUNIF_EMIS (1,2)       = 0.98,     XUNIF_EMIS (5,2)     = 0.98,

```

XUNIF_EMIS (1, 3)	= 0.98,	XUNIF_EMIS (5, 3)	= 0.98,
XUNIF_EMIS (1, 4)	= 0.98,	XUNIF_EMIS (5, 4)	= 0.98,
XUNIF_EMIS (1, 5)	= 0.98,	XUNIF_EMIS (5, 5)	= 0.98,
XUNIF_EMIS (1, 6)	= 0.98,	XUNIF_EMIS (5, 6)	= 0.98,
XUNIF_EMIS (1, 7)	= 0.98,	XUNIF_EMIS (5, 7)	= 0.98,
XUNIF_EMIS (1, 8)	= 0.98,	XUNIF_EMIS (5, 8)	= 0.98,
XUNIF_EMIS (1, 9)	= 0.98,	XUNIF_EMIS (5, 9)	= 0.98,
XUNIF_EMIS (1, 10)	= 0.98,	XUNIF_EMIS (5, 10)	= 0.98,
XUNIF_EMIS (1, 11)	= 0.98,	XUNIF_EMIS (5, 11)	= 0.98,
XUNIF_EMIS (1, 12)	= 0.98,	XUNIF_EMIS (5, 12)	= 0.98,
XUNIF_VEG (2, 1)	= 0.,	XUNIF_VEG (6, 1)	= 0.,
XUNIF_VEG (2, 2)	= 0.,	XUNIF_VEG (6, 2)	= 0.,
XUNIF_VEG (2, 3)	= 0.,	XUNIF_VEG (6, 3)	= 0.,
XUNIF_VEG (2, 4)	= 0.,	XUNIF_VEG (6, 4)	= 0.,
XUNIF_VEG (2, 5)	= 0.5,	XUNIF_VEG (6, 5)	= 0.5,
XUNIF_VEG (2, 6)	= 0.9,	XUNIF_VEG (6, 6)	= 0.9,
XUNIF_VEG (2, 7)	= 0.9,	XUNIF_VEG (6, 7)	= 0.9,
XUNIF_VEG (2, 8)	= 0.9,	XUNIF_VEG (6, 8)	= 0.9,
XUNIF_VEG (2, 9)	= 0.9,	XUNIF_VEG (6, 9)	= 0.9,
XUNIF_VEG (2, 10)	= 0.,	XUNIF_VEG (6, 10)	= 0.,
XUNIF_VEG (2, 11)	= 0.,	XUNIF_VEG (6, 11)	= 0.,
XUNIF_VEG (2, 12)	= 0.,	XUNIF_VEG (6, 12)	= 0.,
XUNIF_LAI (2, 1)	= 0.,	XUNIF_LAI (6, 1)	= 0.,
XUNIF_LAI (2, 2)	= 0.,	XUNIF_LAI (6, 2)	= 0.,
XUNIF_LAI (2, 3)	= 0.,	XUNIF_LAI (6, 3)	= 0.,
XUNIF_LAI (2, 4)	= 0.,	XUNIF_LAI (6, 4)	= 0.,
XUNIF_LAI (2, 5)	= 1.,	XUNIF_LAI (6, 5)	= 1.,
XUNIF_LAI (2, 6)	= 3.,	XUNIF_LAI (6, 6)	= 3.,
XUNIF_LAI (2, 7)	= 3.,	XUNIF_LAI (6, 7)	= 3.,
XUNIF_LAI (2, 8)	= 3.,	XUNIF_LAI (6, 8)	= 3.,
XUNIF_LAI (2, 9)	= 3.,	XUNIF_LAI (6, 9)	= 3.,
XUNIF_LAI (2, 10)	= 0.,	XUNIF_LAI (6, 10)	= 0.,
XUNIF_LAI (2, 11)	= 0.,	XUNIF_LAI (6, 11)	= 0.,
XUNIF_LAI (2, 12)	= 0.,	XUNIF_LAI (6, 12)	= 0.,
XUNIF_Z0 (2, 1)	= 0.01,	XUNIF_Z0 (6, 1)	= 0.01,
XUNIF_Z0 (2, 2)	= 0.01,	XUNIF_Z0 (6, 2)	= 0.01,
XUNIF_Z0 (2, 3)	= 0.01,	XUNIF_Z0 (6, 3)	= 0.01,
XUNIF_Z0 (2, 4)	= 0.01,	XUNIF_Z0 (6, 4)	= 0.01,
XUNIF_Z0 (2, 5)	= 0.05,	XUNIF_Z0 (6, 5)	= 0.05,
XUNIF_Z0 (2, 6)	= 0.15,	XUNIF_Z0 (6, 6)	= 0.15,
XUNIF_Z0 (2, 7)	= 0.15,	XUNIF_Z0 (6, 7)	= 0.15,
XUNIF_Z0 (2, 8)	= 0.15,	XUNIF_Z0 (6, 8)	= 0.15,
XUNIF_Z0 (2, 9)	= 0.15,	XUNIF_Z0 (6, 9)	= 0.15,
XUNIF_Z0 (2, 10)	= 0.01,	XUNIF_Z0 (6, 10)	= 0.01,
XUNIF_Z0 (2, 11)	= 0.01,	XUNIF_Z0 (6, 11)	= 0.01,
XUNIF_Z0 (2, 12)	= 0.01,	XUNIF_Z0 (6, 12)	= 0.01,
XUNIF_EMIS (2, 1)	= 0.98,	XUNIF_EMIS (6, 1)	= 0.98,
XUNIF_EMIS (2, 2)	= 0.98,	XUNIF_EMIS (6, 2)	= 0.98,
XUNIF_EMIS (2, 3)	= 0.98,	XUNIF_EMIS (6, 3)	= 0.98,
XUNIF_EMIS (2, 4)	= 0.98,	XUNIF_EMIS (6, 4)	= 0.98,
XUNIF_EMIS (2, 5)	= 0.98,	XUNIF_EMIS (6, 5)	= 0.98,
XUNIF_EMIS (2, 6)	= 0.98,	XUNIF_EMIS (6, 6)	= 0.98,
XUNIF_EMIS (2, 7)	= 0.98,	XUNIF_EMIS (6, 7)	= 0.98,
XUNIF_EMIS (2, 8)	= 0.98,	XUNIF_EMIS (6, 8)	= 0.98,
XUNIF_EMIS (2, 9)	= 0.98,	XUNIF_EMIS (6, 9)	= 0.98,
XUNIF_EMIS (2, 10)	= 0.98,	XUNIF_EMIS (6, 10)	= 0.98,

XUNIF_EMIS (2, 11)	= 0.98,	XUNIF_EMIS (6, 11)	= 0.98,
XUNIF_EMIS (2, 12)	= 0.98,	XUNIF_EMIS (6, 12)	= 0.98,
XUNIF_VEG (3, 1)	= 0.,	XUNIF_VEG (7, 1)	= 0.,
XUNIF_VEG (3, 2)	= 0.,	XUNIF_VEG (7, 2)	= 0.,
XUNIF_VEG (3, 3)	= 0.,	XUNIF_VEG (7, 3)	= 0.,
XUNIF_VEG (3, 4)	= 0.,	XUNIF_VEG (7, 4)	= 0.,
XUNIF_VEG (3, 5)	= 0.5,	XUNIF_VEG (7, 5)	= 0.5,
XUNIF_VEG (3, 6)	= 0.9,	XUNIF_VEG (7, 6)	= 0.9,
XUNIF_VEG (3, 7)	= 0.9,	XUNIF_VEG (7, 7)	= 0.9,
XUNIF_VEG (3, 8)	= 0.9,	XUNIF_VEG (7, 8)	= 0.9,
XUNIF_VEG (3, 9)	= 0.9,	XUNIF_VEG (7, 9)	= 0.9,
XUNIF_VEG (3, 10)	= 0.,	XUNIF_VEG (7, 10)	= 0.,
XUNIF_VEG (3, 11)	= 0.,	XUNIF_VEG (7, 11)	= 0.,
XUNIF_VEG (3, 12)	= 0.,	XUNIF_VEG (7, 12)	= 0.,
XUNIF_LAI (3, 1)	= 0.,	XUNIF_LAI (7, 1)	= 0.,
XUNIF_LAI (3, 2)	= 0.,	XUNIF_LAI (7, 2)	= 0.,
XUNIF_LAI (3, 3)	= 0.,	XUNIF_LAI (7, 3)	= 0.,
XUNIF_LAI (3, 4)	= 0.,	XUNIF_LAI (7, 4)	= 0.,
XUNIF_LAI (3, 5)	= 1.,	XUNIF_LAI (7, 5)	= 1.,
XUNIF_LAI (3, 6)	= 3.,	XUNIF_LAI (7, 6)	= 3.,
XUNIF_LAI (3, 7)	= 3.,	XUNIF_LAI (7, 7)	= 3.,
XUNIF_LAI (3, 8)	= 3.,	XUNIF_LAI (7, 8)	= 3.,
XUNIF_LAI (3, 9)	= 3.,	XUNIF_LAI (7, 9)	= 3.,
XUNIF_LAI (3, 10)	= 0.,	XUNIF_LAI (7, 10)	= 0.,
XUNIF_LAI (3, 11)	= 0.,	XUNIF_LAI (7, 11)	= 0.,
XUNIF_LAI (3, 12)	= 0.,	XUNIF_LAI (7, 12)	= 0.,
XUNIF_Z0 (3, 1)	= 0.01,	XUNIF_Z0 (7, 1)	= 0.01,
XUNIF_Z0 (3, 2)	= 0.01,	XUNIF_Z0 (7, 2)	= 0.01,
XUNIF_Z0 (3, 3)	= 0.01,	XUNIF_Z0 (7, 3)	= 0.01,
XUNIF_Z0 (3, 4)	= 0.01,	XUNIF_Z0 (7, 4)	= 0.01,
XUNIF_Z0 (3, 5)	= 0.05,	XUNIF_Z0 (7, 5)	= 0.05,
XUNIF_Z0 (3, 6)	= 0.15,	XUNIF_Z0 (7, 6)	= 0.15,
XUNIF_Z0 (3, 7)	= 0.15,	XUNIF_Z0 (7, 7)	= 0.15,
XUNIF_Z0 (3, 8)	= 0.15,	XUNIF_Z0 (7, 8)	= 0.15,
XUNIF_Z0 (3, 9)	= 0.15,	XUNIF_Z0 (7, 9)	= 0.15,
XUNIF_Z0 (3, 10)	= 0.01,	XUNIF_Z0 (7, 10)	= 0.01,
XUNIF_Z0 (3, 11)	= 0.01,	XUNIF_Z0 (7, 11)	= 0.01,
XUNIF_Z0 (3, 12)	= 0.01,	XUNIF_Z0 (7, 12)	= 0.01,
XUNIF_EMIS (3, 1)	= 0.98,	XUNIF_EMIS (7, 1)	= 0.98,
XUNIF_EMIS (3, 2)	= 0.98,	XUNIF_EMIS (7, 2)	= 0.98,
XUNIF_EMIS (3, 3)	= 0.98,	XUNIF_EMIS (7, 3)	= 0.98,
XUNIF_EMIS (3, 4)	= 0.98,	XUNIF_EMIS (7, 4)	= 0.98,
XUNIF_EMIS (3, 5)	= 0.98,	XUNIF_EMIS (7, 5)	= 0.98,
XUNIF_EMIS (3, 6)	= 0.98,	XUNIF_EMIS (7, 6)	= 0.98,
XUNIF_EMIS (3, 7)	= 0.98,	XUNIF_EMIS (7, 7)	= 0.98,
XUNIF_EMIS (3, 8)	= 0.98,	XUNIF_EMIS (7, 8)	= 0.98,
XUNIF_EMIS (3, 9)	= 0.98,	XUNIF_EMIS (7, 9)	= 0.98,
XUNIF_EMIS (3, 10)	= 0.98,	XUNIF_EMIS (7, 10)	= 0.98,
XUNIF_EMIS (3, 11)	= 0.98,	XUNIF_EMIS (7, 11)	= 0.98,
XUNIF_EMIS (3, 12)	= 0.98,	XUNIF_EMIS (7, 12)	= 0.98,
XUNIF_VEG (4, 1)	= 0.,	XUNIF_VEG (8, 1)	= 0.,
XUNIF_VEG (4, 2)	= 0.,	XUNIF_VEG (8, 2)	= 0.,
XUNIF_VEG (4, 3)	= 0.,	XUNIF_VEG (8, 3)	= 0.,
XUNIF_VEG (4, 4)	= 0.,	XUNIF_VEG (8, 4)	= 0.,
XUNIF_VEG (4, 5)	= 0.5,	XUNIF_VEG (8, 5)	= 0.5,
XUNIF_VEG (4, 6)	= 0.9,	XUNIF_VEG (8, 6)	= 0.9,

XUNIF_VEG (4, 7)	= 0.9,	XUNIF_VEG (8, 7)	= 0.9,
XUNIF_VEG (4, 8)	= 0.9,	XUNIF_VEG (8, 8)	= 0.9,
XUNIF_VEG (4, 9)	= 0.9,	XUNIF_VEG (8, 9)	= 0.9,
XUNIF_VEG (4, 10)	= 0.,	XUNIF_VEG (8, 10)	= 0.,
XUNIF_VEG (4, 11)	= 0.,	XUNIF_VEG (8, 11)	= 0.,
XUNIF_VEG (4, 12)	= 0.,	XUNIF_VEG (8, 12)	= 0.,
XUNIF_LAI (4, 1)	= 0.,	XUNIF_LAI (8, 1)	= 0.,
XUNIF_LAI (4, 2)	= 0.,	XUNIF_LAI (8, 2)	= 0.,
XUNIF_LAI (4, 3)	= 0.,	XUNIF_LAI (8, 3)	= 0.,
XUNIF_LAI (4, 4)	= 0.,	XUNIF_LAI (8, 4)	= 0.,
XUNIF_LAI (4, 5)	= 1.,	XUNIF_LAI (8, 5)	= 1.,
XUNIF_LAI (4, 6)	= 3.,	XUNIF_LAI (8, 6)	= 3.,
XUNIF_LAI (4, 7)	= 3.,	XUNIF_LAI (8, 7)	= 3.,
XUNIF_LAI (4, 8)	= 3.,	XUNIF_LAI (8, 8)	= 3.,
XUNIF_LAI (4, 9)	= 3.,	XUNIF_LAI (8, 9)	= 3.,
XUNIF_LAI (4, 10)	= 0.,	XUNIF_LAI (8, 10)	= 0.,
XUNIF_LAI (4, 11)	= 0.,	XUNIF_LAI (8, 11)	= 0.,
XUNIF_LAI (4, 12)	= 0.,	XUNIF_LAI (8, 12)	= 0.,
XUNIF_Z0 (4, 1)	= 0.01,	XUNIF_Z0 (8, 1)	= 0.01,
XUNIF_Z0 (4, 2)	= 0.01,	XUNIF_Z0 (8, 2)	= 0.01,
XUNIF_Z0 (4, 3)	= 0.01,	XUNIF_Z0 (8, 3)	= 0.01,
XUNIF_Z0 (4, 4)	= 0.01,	XUNIF_Z0 (8, 4)	= 0.01,
XUNIF_Z0 (4, 5)	= 0.05,	XUNIF_Z0 (8, 5)	= 0.05,
XUNIF_Z0 (4, 6)	= 0.15,	XUNIF_Z0 (8, 6)	= 0.15,
XUNIF_Z0 (4, 7)	= 0.15,	XUNIF_Z0 (8, 7)	= 0.15,
XUNIF_Z0 (4, 8)	= 0.15,	XUNIF_Z0 (8, 8)	= 0.15,
XUNIF_Z0 (4, 9)	= 0.15,	XUNIF_Z0 (8, 9)	= 0.15,
XUNIF_Z0 (4, 10)	= 0.01,	XUNIF_Z0 (8, 10)	= 0.01,
XUNIF_Z0 (4, 11)	= 0.01,	XUNIF_Z0 (8, 11)	= 0.01,
XUNIF_Z0 (4, 12)	= 0.01,	XUNIF_Z0 (8, 12)	= 0.01,
XUNIF_EMIS (4, 1)	= 0.98,	XUNIF_EMIS (8, 1)	= 0.98,
XUNIF_EMIS (4, 2)	= 0.98,	XUNIF_EMIS (8, 2)	= 0.98,
XUNIF_EMIS (4, 3)	= 0.98,	XUNIF_EMIS (8, 3)	= 0.98,
XUNIF_EMIS (4, 4)	= 0.98,	XUNIF_EMIS (8, 4)	= 0.98,
XUNIF_EMIS (4, 5)	= 0.98,	XUNIF_EMIS (8, 5)	= 0.98,
XUNIF_EMIS (4, 6)	= 0.98,	XUNIF_EMIS (8, 6)	= 0.98,
XUNIF_EMIS (4, 7)	= 0.98,	XUNIF_EMIS (8, 7)	= 0.98,
XUNIF_EMIS (4, 8)	= 0.98,	XUNIF_EMIS (8, 8)	= 0.98,
XUNIF_EMIS (4, 9)	= 0.98,	XUNIF_EMIS (8, 9)	= 0.98,
XUNIF_EMIS (4, 10)	= 0.98,	XUNIF_EMIS (8, 10)	= 0.98,
XUNIF_EMIS (4, 11)	= 0.98,	XUNIF_EMIS (8, 11)	= 0.98,
XUNIF_EMIS (4, 12)	= 0.98,	XUNIF_EMIS (8, 12)	= 0.98,
XUNIF_DG (1, 1)	= 0.01,	XUNIF_DG (5, 1)	= 0.01,
XUNIF_DG (1, 2)	= 1.60,	XUNIF_DG (5, 2)	= 1.60,
XUNIF_DG (1, 3)	= 1.60,	XUNIF_DG (5, 3)	= 1.60,
XUNIF_DG (2, 1)	= 0.01,	XUNIF_DG (6, 1)	= 0.01,
XUNIF_DG (2, 2)	= 1.60,	XUNIF_DG (6, 2)	= 1.60,
XUNIF_DG (2, 3)	= 1.60,	XUNIF_DG (6, 3)	= 1.60,
XUNIF_DG (3, 1)	= 0.01,	XUNIF_DG (7, 1)	= 0.01,
XUNIF_DG (3, 2)	= 1.60,	XUNIF_DG (7, 2)	= 1.60,
XUNIF_DG (3, 3)	= 1.60,	XUNIF_DG (7, 3)	= 1.60,
XUNIF_DG (4, 1)	= 0.01,	XUNIF_DG (8, 1)	= 0.01,
XUNIF_DG (4, 2)	= 1.60,	XUNIF_DG (8, 2)	= 1.60,
XUNIF_DG (4, 3)	= 1.60,	XUNIF_DG (8, 3)	= 1.60,
XUNIF_ROOTFRAC (1, 1)	= -999.,	XUNIF_ROOTFRAC (5, 1)	= -999.,
XUNIF_ROOTFRAC (1, 2)	= -999.,	XUNIF_ROOTFRAC (5, 2)	= -999.,

XUNIF_ROOTFRAC (1, 3)	= -999.,	XUNIF_ROOTFRAC (5, 3)	= -999.,
XUNIF_ROOTFRAC (2, 1)	= -999.,	XUNIF_ROOTFRAC (6, 1)	= -999.,
XUNIF_ROOTFRAC (2, 2)	= -999.,	XUNIF_ROOTFRAC (6, 2)	= -999.,
XUNIF_ROOTFRAC (2, 3)	= -999.,	XUNIF_ROOTFRAC (6, 3)	= -999.,
XUNIF_ROOTFRAC (3, 1)	= -999.,	XUNIF_ROOTFRAC (7, 1)	= -999.,
XUNIF_ROOTFRAC (3, 2)	= -999.,	XUNIF_ROOTFRAC (7, 2)	= -999.,
XUNIF_ROOTFRAC (3, 3)	= -999.,	XUNIF_ROOTFRAC (7, 3)	= -999.,
XUNIF_ROOTFRAC (4, 1)	= -999.,	XUNIF_ROOTFRAC (8, 1)	= -999.,
XUNIF_ROOTFRAC (4, 2)	= -999.,	XUNIF_ROOTFRAC (8, 2)	= -999.,
XUNIF_ROOTFRAC (4, 3)	= -999.,	XUNIF_ROOTFRAC (8, 3)	= -999.,
XUNIF_RSMIN	(1)= 40.,	XUNIF_RSMIN	(2)= 40.,
XUNIF_GAMMA	(1)= 0.,	XUNIF_GAMMA	(2)= 0.,
XUNIF_WRMX_CF	(1)= 0.2,	XUNIF_WRMX_CF	(2)= 0.2,
XUNIF_RGL	(1)= 100.,	XUNIF_RGL	(2)= 100.,
XUNIF_CV	(1)= 0.00002,	XUNIF_CV	(2)= 0.00002,
XUNIF_Z0_O_ZOH	(1)= 10.,	XUNIF_Z0_O_ZOH	(2)= 10.,
XUNIF_ALBNIR_VEG	(1)= 0.3,	XUNIF_ALBNIR_VEG	(2)= 0.3,
XUNIF_ALBVIS_VEG	(1)= 0.1,	XUNIF_ALBVIS_VEG	(2)= 0.1,
XUNIF_ALBUV_VEG	(1)= 0.0425,	XUNIF_ALBUV_VEG	(2)= 0.0425,
XUNIF_ALBNIR_SOIL	(1)= 0.3,	XUNIF_ALBNIR_SOIL	(2)= 0.3,
XUNIF_ALBVIS_SOIL	(1)= 0.1,	XUNIF_ALBVIS_SOIL	(2)= 0.1,
XUNIF_ALBUV_SOIL	(1)= 0.06,	XUNIF_ALBUV_SOIL	(2)= 0.06,
XUNIF_GMES	(1)= 0.003,	XUNIF_GMES	(2)= 0.003,
XUNIF_RE25	(1)= 0.0000003,	XUNIF_RE25	(2)= 0.0000003,
XUNIF_BSLAI	(1)= 0.06,	XUNIF_BSLAI	(2)= 0.06,
XUNIF_LAIMIN	(1)= 0.3,	XUNIF_LAIMIN	(2)= 0.3,
XUNIF_SEFOLD	(1)= 5184000.,	XUNIF_SEFOLD	(2)= 5184000.,
XUNIF_GC	(1)= 0.00025,	XUNIF_GC	(2)= 0.00025,
XUNIF_DMAX	(1)= 0.1,	XUNIF_DMAX	(2)= 0.1,
XUNIF_F2I	(1)= 0.3,	XUNIF_F2I	(2)= 0.3,
XUNIF_H_TREE	(1)= 20.,	XUNIF_H_TREE	(2)= 20.,
XUNIF_CE_NITRO	(1)= 3.79,	XUNIF_CE_NITRO	(2)= 3.79,
XUNIF_CF_NITRO	(1)= 9.84,	XUNIF_CF_NITRO	(2)= 9.84,
XUNIF_CNA_NITRO	(1)= 1.3,	XUNIF_CNA_NITRO	(2)= 1.3,
XUNIF_RSMIN	(4)= 40.,	XUNIF_RSMIN	(5)= 40.,
XUNIF_GAMMA	(4)= 0.,	XUNIF_GAMMA	(5)= 0.,
XUNIF_WRMX_CF	(4)= 0.2,	XUNIF_WRMX_CF	(5)= 0.2,
XUNIF_RGL	(4)= 100.,	XUNIF_RGL	(5)= 100.,
XUNIF_CV	(4)= 0.00002,	XUNIF_CV	(5)= 0.00002,
XUNIF_Z0_O_ZOH	(4)= 10.,	XUNIF_Z0_O_ZOH	(5)= 10.,
XUNIF_ALBNIR_VEG	(4)= 0.3,	XUNIF_ALBNIR_VEG	(5)= 0.3,
XUNIF_ALBVIS_VEG	(4)= 0.1,	XUNIF_ALBVIS_VEG	(5)= 0.1,
XUNIF_ALBUV_VEG	(4)= 0.0425,	XUNIF_ALBUV_VEG	(5)= 0.0425,
XUNIF_ALBNIR_SOIL	(4)= 0.3,	XUNIF_ALBNIR_SOIL	(5)= 0.3,
XUNIF_ALBVIS_SOIL	(4)= 0.1,	XUNIF_ALBVIS_SOIL	(5)= 0.1,
XUNIF_ALBUV_SOIL	(4)= 0.06,	XUNIF_ALBUV_SOIL	(5)= 0.06,
XUNIF_GMES	(4)= 0.003,	XUNIF_GMES	(5)= 0.003,
XUNIF_RE25	(4)= 0.0000003,	XUNIF_RE25	(5)= 0.0000003,
XUNIF_BSLAI	(4)= 0.06,	XUNIF_BSLAI	(5)= 0.06,
XUNIF_LAIMIN	(4)= 0.3,	XUNIF_LAIMIN	(5)= 0.3,
XUNIF_SEFOLD	(4)= 5184000.,	XUNIF_SEFOLD	(5)= 5184000.,
XUNIF_GC	(4)= 0.00025,	XUNIF_GC	(5)= 0.00025,
XUNIF_DMAX	(4)= 0.1,	XUNIF_DMAX	(5)= 0.1,
XUNIF_F2I	(4)= 0.3,	XUNIF_F2I	(5)= 0.3,
XUNIF_H_TREE	(4)= 20.,	XUNIF_H_TREE	(5)= 20.,
XUNIF_CE_NITRO	(4)= 3.79,	XUNIF_CE_NITRO	(5)= 3.79,



XUNIF_CF_NITRO	(4) = 9.84,	XUNIF_CF_NITRO	(5) = 9.84,
XUNIF_CNA_NITRO	(4) = 1.3,	XUNIF_CNA_NITRO	(5) = 1.3,
XUNIF_RSMIN	(7) = 40.,	XUNIF_RSMIN	(8) = 40.,
XUNIF_GAMMA	(7) = 0.,	XUNIF_GAMMA	(8) = 0.,
XUNIF_WRMAX_CF	(7) = 0.2,	XUNIF_WRMAX_CF	(8) = 0.2,
XUNIF_RGL	(7) = 100.,	XUNIF_RGL	(8) = 100.,
XUNIF_CV	(7) = 0.00002,	XUNIF_CV	(8) = 0.00002,
XUNIF_Z0_O_ZOH	(7) = 10.,	XUNIF_Z0_O_ZOH	(8) = 10.,
XUNIF_ALBNIR_VEG	(7) = 0.3,	XUNIF_ALBNIR_VEG	(8) = 0.3,
XUNIF_ALBVIS_VEG	(7) = 0.1,	XUNIF_ALBVIS_VEG	(8) = 0.1,
XUNIF_ALBUV_VEG	(7) = 0.0425,	XUNIF_ALBUV_VEG	(8) = 0.0425,
XUNIF_ALBNIR_SOIL	(7) = 0.3,	XUNIF_ALBNIR_SOIL	(8) = 0.3,
XUNIF_ALBVIS_SOIL	(7) = 0.1,	XUNIF_ALBVIS_SOIL	(8) = 0.1,
XUNIF_ALBUV_SOIL	(7) = 0.06,	XUNIF_ALBUV_SOIL	(8) = 0.06,
XUNIF_GMES	(7) = 0.003,	XUNIF_GMES	(8) = 0.003,
XUNIF_RE25	(7) = 0.0000003,	XUNIF_RE25	(8) = 0.0000003,
XUNIF_BSLAI	(7) = 0.06,	XUNIF_BSLAI	(8) = 0.06,
XUNIF_LAIMIN	(7) = 0.3,	XUNIF_LAIMIN	(8) = 0.3,
XUNIF_SEFOLD	(7) = 5184000.,	XUNIF_SEFOLD	(8) = 5184000.,
XUNIF_GC	(7) = 0.00025,	XUNIF_GC	(8) = 0.00025,
XUNIF_DMAX	(7) = 0.1,	XUNIF_DMAX	(8) = 0.1,
XUNIF_F2I	(7) = 0.3,	XUNIF_F2I	(8) = 0.3,
XUNIF_H_TREE	(7) = 20.,	XUNIF_H_TREE	(8) = 20.,
XUNIF_CE_NITRO	(7) = 3.79,	XUNIF_CE_NITRO	(8) = 3.79,
XUNIF_CF_NITRO	(7) = 9.84,	XUNIF_CF_NITRO	(8) = 9.84,
XUNIF_CNA_NITRO	(7) = 1.3,	XUNIF_CNA_NITRO	(8) = 1.3,
XUNIF_RSMIN	(10) = 40.,	XUNIF_RSMIN	(11) = 40.,
XUNIF_GAMMA	(10) = 0.,	XUNIF_GAMMA	(11) = 0.,
XUNIF_WRMAX_CF	(10) = 0.2,	XUNIF_WRMAX_CF	(11) = 0.2,
XUNIF_RGL	(10) = 100.,	XUNIF_RGL	(11) = 100.,
XUNIF_CV	(10) = 0.00002,	XUNIF_CV	(11) = 0.00002,
XUNIF_Z0_O_ZOH	(10) = 10.,	XUNIF_Z0_O_ZOH	(11) = 10.,
XUNIF_ALBNIR_VEG	(10) = 0.3,	XUNIF_ALBNIR_VEG	(11) = 0.3,
XUNIF_ALBVIS_VEG	(10) = 0.1,	XUNIF_ALBVIS_VEG	(11) = 0.1,
XUNIF_ALBUV_VEG	(10) = 0.0425,	XUNIF_ALBUV_VEG	(11) = 0.0425,
XUNIF_ALBNIR_SOIL	(10) = 0.3,	XUNIF_ALBNIR_SOIL	(11) = 0.3,
XUNIF_ALBVIS_SOIL	(10) = 0.1,	XUNIF_ALBVIS_SOIL	(11) = 0.1,
XUNIF_ALBUV_SOIL	(10) = 0.06,	XUNIF_ALBUV_SOIL	(11) = 0.06,
XUNIF_GMES	(10) = 0.003,	XUNIF_GMES	(11) = 0.003,
XUNIF_RE25	(10) = 0.0000003,	XUNIF_RE25	(11) = 0.0000003,
XUNIF_BSLAI	(10) = 0.06,	XUNIF_BSLAI	(11) = 0.06,
XUNIF_LAIMIN	(10) = 0.3,	XUNIF_LAIMIN	(11) = 0.3,
XUNIF_SEFOLD	(10) = 5184000.,	XUNIF_SEFOLD	(11) = 5184000.,
XUNIF_GC	(10) = 0.00025,	XUNIF_GC	(11) = 0.00025,
XUNIF_DMAX	(10) = 0.1,	XUNIF_DMAX	(11) = 0.1,
XUNIF_F2I	(10) = 0.3,	XUNIF_F2I	(11) = 0.3,
XUNIF_H_TREE	(10) = 20.,	XUNIF_H_TREE	(11) = 20.,
XUNIF_CE_NITRO	(10) = 3.79,	XUNIF_CE_NITRO	(11) = 3.79,
XUNIF_CF_NITRO	(10) = 9.84,	XUNIF_CF_NITRO	(11) = 9.84,
XUNIF_CNA_NITRO	(10) = 1.3,	XUNIF_CNA_NITRO	(11) = 1.3,

/

&NAM\_DATA\_TEB

NROOF_LAYER	= 3,
XUNIF_ALB_ROOF	= 0.2,
XUNIF_EMIS_ROOF	= 0.97,
XUNIF_HC_ROOF (1)	= 2110000.,
XUNIF_HC_ROOF (2)	= 2800000.,

```

XUNIF_HC_ROOF(3) = 2900000.,
XUNIF_TC_ROOF(1) = 1.51,
XUNIF_TC_ROOF(2) = 0.08,
XUNIF_TC_ROOF(3) = 0.05,
XUNIF_D_ROOF(1) = 0.05,
XUNIF_D_ROOF(2) = 0.4,
XUNIF_D_ROOF(3) = 0.1,
NROAD_LAYER      = 3,
XUNIF_ALB_ROAD   = 0.2,
XUNIF_EMIS_ROAD  = 0.97,
XUNIF_HC_ROAD(1) = 2110000.,
XUNIF_HC_ROAD(2) = 2800000.,
XUNIF_HC_ROAD(3) = 2900000.,
XUNIF_TC_ROAD(1) = 1.51,
XUNIF_TC_ROAD(2) = 0.08,
XUNIF_TC_ROAD(3) = 0.05,
XUNIF_D_ROAD(1)  = 0.05,
XUNIF_D_ROAD(2)  = 0.4,
XUNIF_D_ROAD(3)  = 0.1,
NWALL_LAYER      = 3,
XUNIF_ALB_WALL   = 0.2,
XUNIF_EMIS_WALL  = 0.97,
XUNIF_HC_WALL(1) = 2110000.,
XUNIF_HC_WALL(2) = 2800000.,
XUNIF_HC_WALL(3) = 2900000.,
XUNIF_TC_WALL(1) = 1.51,
XUNIF_TC_WALL(2) = 0.08,
XUNIF_TC_WALL(3) = 0.05,
XUNIF_D_WALL(1)  = 0.05,
XUNIF_D_WALL(2)  = 0.4,
XUNIF_D_WALL(3)  = 0.1,
XUNIF_Z0_TOWN    = 1.,
XUNIF_BLD        = 0.5,
XUNIF_BLD_HEIGHT = 10.,
XUNIF_WALL_O_HOR = 0.5,
XUNIF_H_TRAFFIC  = 10.,
XUNIF_LE_TRAFFIC = 0.,
XUNIF_H_INDUSTRY = 5.,
XUNIF_LE_INDUSTRY= 0.

/
&NAM_FRAC          LECOCLIMAP = F,
                   XUNIF_SEA   = 0.,
                   XUNIF_WATER  = 0.,
                   XUNIF_TOWN   = 0.,
                   XUNIF_NATURE  = 1.

/
&NAM_PGD_GRID      CGRID = 'LONLAT REG'

/
&NAM_LONLAT_REG    XLONMIN = 0.           ,
                   XLONMAX = 0.           ,
                   XLATMIN = 0.           ,
                   XLATMAX = 0.           ,
                   NLON    = 1             ,
                   NLAT    = 1

/
&NAM_PGD_SCHEMES   CNATURE = 'ISBA ' ,

```

```

        CSEA      = 'SEAFIX'      ,
        CTOWN     = 'TEB      '    ,
        CWATER    = 'WATFIX'

/
&NAM_ZS          XUNIF_ZS = 113.
/
&NAM_ISBA        XUNIF_CLAY   = 0.37      ,
                  XUNIF_SAND   = 0.37      ,
                  XUNIF_RUNOFFB = 0.5       ,
                  CISBA        = '2-L'     ,
                  CPHOTO       = 'NIT'     ,
                  NPATCH      = 12        ,
                  NGROUND_LAYER = 2

/

&NAM_PREPFILE    CPREPFILE = 'PREP'
/
&NAM_PREP_SURF_ATM NYEAR = 1986,
                  NMONTH = 1,
                  NDAY   = 1,
                  XTIME  = 0.
/
&NAM_PREP_SEAFLUX XSST_UNIF = 285.,
                  NYEAR = 1986,
                  NMONTH = 1,
                  NDAY   = 1,
                  XTIME  = 0.
/
&NAM_PREP_WATFLUX XTS_WATER_UNIF = 285.,
                  NYEAR = 1986,
                  NMONTH = 1,
                  NDAY   = 1,
                  XTIME  = 0.
/
&NAM_PREP_TEB     XTI_ROAD= 285.,
                  XTI_BLD = 285.
                  XTS_ROAD= 285.
                  XTS_ROOF= 285.,
                  XTS_WALL= 285.,
                  XWS_ROAD= 0.,
                  XWS_ROOF= 0.,
                  NYEAR = 1986,
                  NMONTH = 1,
                  NDAY   = 1,
                  XTIME  = 0.
/
&NAM_PREP_ISBA    XHUG_SURF = 1.,
                  XHUG_ROOT = 1.,
                  XHUG_DEEP = 1.,
                  XTG_SURF  = 276.16,
                  XTG_ROOT  = 276.16,
                  XTG_DEEP  = 276.16,
                  NYEAR = 1986,
                  NMONTH = 1,
                  NDAY   = 1,
                  XTIME  = 0.

```

```

/
&NAM_PREP_ISBA_SNOW      CSNOW = '3-L'
/

&NAM_IO_OFFLINE          LPRINT = T ,
                          CFORCING_FILETYPE = 'NETCDF' ,
                          CSURF_FILETYPE = 'LFI' ,
                          CTIMESERIES_FILETYPE = 'NETCDF' ,
                          LWRITE_COORD = T,
                          LSET_FORC_ZS=T
/

&NAM_DIAG_SURFm          LSURF_BUDGET = F ,
                          N2M = 0 ,
                          LCOEF = F ,
                          LSURF_VARS = F
/

&NAM_DIAG_SURF_ATMn      LFRAC = F
/

&NAM_DIAG_ISBAn          LPGD = F ,
                          LSURF_EVAP_BUDGET = F ,
                          LSURF_MISC_BUDGET = F
/

&NAM_DIAG_TEBn           LSURF_MISC_BUDGET = F
/

&NAM_SGH_ISBAn           CRUNOFF = "WSAT"
/

&NAM_ISBAn               CROUGH = "Z04D" ,
                          CSCOND = "NP89" ,
                          CALBEDO = "DRY" ,
                          CC1DRY = 'DEF' ,
                          CSOILFRZ = 'DEF' ,
                          CDIFSFCOND = 'DEF' ,
                          CSNOWRES = 'DEF' ,
                          CCPSURF = 'DRY'
/

&NAM_CH_ISBAn            CCH_DRY_DEP = "WES89 "
/

&NAM_SEAFLUXn            CSEA_ALB = "TA96"
/

&NAM_CH_SEAFLUXn        CCH_DRY_DEP = "WES89 "
/

&NAM_CH_WATFLUXn        CCH_DRY_DEP = "WES89 "
/

&NAM_CH_TEBn            CCH_DRY_DEP = "WES89 "
/

```

## 10.8.3. Surface parameters read from external files

The following namelist is valid only for simulation without patches. In case of use of patches (like for A-gs options), it should be updated.

```
&NAM_DATA_ISBA      NTIME = 12 ,
                    CFNAM_VEGTYPE (1)    = 'VEGTYPE_01.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (2)    = 'VEGTYPE_02.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (3)    = 'VEGTYPE_03.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (4)    = 'VEGTYPE_04.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (5)    = 'VEGTYPE_05.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (6)    = 'VEGTYPE_06.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (7)    = 'VEGTYPE_07.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (8)    = 'VEGTYPE_08.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (9)    = 'VEGTYPE_09.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (10)   = 'VEGTYPE_10.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (11)   = 'VEGTYPE_11.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEGTYPE (12)   = 'VEGTYPE_12.DAT', CFTYP_VEGTYPE (
                    CFNAM_VEG (1, 1)     = 'VEG_01.DAT', CFTYP_VEG (1, 1)
                    CFNAM_VEG (1, 2)     = 'VEG_02.DAT', CFTYP_VEG (1, 2)
                    CFNAM_VEG (1, 3)     = 'VEG_03.DAT', CFTYP_VEG (1, 3)
                    CFNAM_VEG (1, 4)     = 'VEG_04.DAT', CFTYP_VEG (1, 4)
                    CFNAM_VEG (1, 5)     = 'VEG_05.DAT', CFTYP_VEG (1, 5)
                    CFNAM_VEG (1, 6)     = 'VEG_06.DAT', CFTYP_VEG (1, 6)
                    CFNAM_VEG (1, 7)     = 'VEG_07.DAT', CFTYP_VEG (1, 7)
                    CFNAM_VEG (1, 8)     = 'VEG_08.DAT', CFTYP_VEG (1, 8)
                    CFNAM_VEG (1, 9)     = 'VEG_09.DAT', CFTYP_VEG (1, 9)
                    CFNAM_VEG (1, 10)    = 'VEG_10.DAT', CFTYP_VEG (1, 10)
                    CFNAM_VEG (1, 11)    = 'VEG_11.DAT', CFTYP_VEG (1, 11)
                    CFNAM_VEG (1, 12)    = 'VEG_12.DAT', CFTYP_VEG (1, 12)
                    CFNAM_LAI (1, 1)     = 'LAI_01.DAT', CFTYP_LAI (1, 1)
                    CFNAM_LAI (1, 2)     = 'LAI_02.DAT', CFTYP_LAI (1, 2)
                    CFNAM_LAI (1, 3)     = 'LAI_03.DAT', CFTYP_LAI (1, 3)
                    CFNAM_LAI (1, 4)     = 'LAI_04.DAT', CFTYP_LAI (1, 4)
                    CFNAM_LAI (1, 5)     = 'LAI_05.DAT', CFTYP_LAI (1, 5)
                    CFNAM_LAI (1, 6)     = 'LAI_06.DAT', CFTYP_LAI (1, 6)
                    CFNAM_LAI (1, 7)     = 'LAI_07.DAT', CFTYP_LAI (1, 7)
                    CFNAM_LAI (1, 8)     = 'LAI_08.DAT', CFTYP_LAI (1, 8)
                    CFNAM_LAI (1, 9)     = 'LAI_09.DAT', CFTYP_LAI (1, 9)
                    CFNAM_LAI (1, 10)    = 'LAI_10.DAT', CFTYP_LAI (1, 10)
                    CFNAM_LAI (1, 11)    = 'LAI_11.DAT', CFTYP_LAI (1, 11)
                    CFNAM_LAI (1, 12)    = 'LAI_12.DAT', CFTYP_LAI (1, 12)
                    CFNAM_Z0 (1, 1)      = 'Z0_01.DAT', CFTYP_Z0 (1, 1)
                    CFNAM_Z0 (1, 2)      = 'Z0_02.DAT', CFTYP_Z0 (1, 2)
                    CFNAM_Z0 (1, 3)      = 'Z0_03.DAT', CFTYP_Z0 (1, 3)
                    CFNAM_Z0 (1, 4)      = 'Z0_04.DAT', CFTYP_Z0 (1, 4)
                    CFNAM_Z0 (1, 5)      = 'Z0_05.DAT', CFTYP_Z0 (1, 5)
                    CFNAM_Z0 (1, 6)      = 'Z0_06.DAT', CFTYP_Z0 (1, 6)
                    CFNAM_Z0 (1, 7)      = 'Z0_07.DAT', CFTYP_Z0 (1, 7)
                    CFNAM_Z0 (1, 8)      = 'Z0_08.DAT', CFTYP_Z0 (1, 8)
                    CFNAM_Z0 (1, 9)      = 'Z0_09.DAT', CFTYP_Z0 (1, 9)
                    CFNAM_Z0 (1, 10)     = 'Z0_10.DAT', CFTYP_Z0 (1, 10)
```

```

CFNAM_Z0(1,11)      = 'Z0_11.DAT           ', CFTYP_Z0(1,11)
CFNAM_Z0(1,12)      = 'Z0_12.DAT           ', CFTYP_Z0(1,12)
CFNAM_EMIS(1,1)     = 'EMIS_01.DAT          ', CFTYP_EMIS(1,1)
CFNAM_EMIS(1,2)     = 'EMIS_02.DAT          ', CFTYP_EMIS(1,2)
CFNAM_EMIS(1,3)     = 'EMIS_03.DAT          ', CFTYP_EMIS(1,3)
CFNAM_EMIS(1,4)     = 'EMIS_04.DAT          ', CFTYP_EMIS(1,4)
CFNAM_EMIS(1,5)     = 'EMIS_05.DAT          ', CFTYP_EMIS(1,5)
CFNAM_EMIS(1,6)     = 'EMIS_06.DAT          ', CFTYP_EMIS(1,6)
CFNAM_EMIS(1,7)     = 'EMIS_07.DAT          ', CFTYP_EMIS(1,7)
CFNAM_EMIS(1,8)     = 'EMIS_08.DAT          ', CFTYP_EMIS(1,8)
CFNAM_EMIS(1,9)     = 'EMIS_09.DAT          ', CFTYP_EMIS(1,9)
CFNAM_EMIS(1,10)    = 'EMIS_10.DAT         ', CFTYP_EMIS(1,10)
CFNAM_EMIS(1,11)    = 'EMIS_11.DAT         ', CFTYP_EMIS(1,11)
CFNAM_EMIS(1,12)    = 'EMIS_12.DAT         ', CFTYP_EMIS(1,12)
CFNAM_DG(1,1)       = 'DG_1.DAT             ', CFTYP_DG(1,1)
CFNAM_DG(1,2)       = 'DG_2.DAT             ', CFTYP_DG(1,2)
CFNAM_DG(1,3)       = 'DG_3.DAT             ', CFTYP_DG(1,3)
CFNAM_ROOTFRAC(1,1) = 'ROOTFRAC_1.DAT        ', CFTYP_ROOTFRAC(1,1)
CFNAM_ROOTFRAC(1,2) = 'ROOTFRAC_2.DAT        ', CFTYP_ROOTFRAC(1,2)
CFNAM_ROOTFRAC(1,3) = 'ROOTFRAC_3.DAT        ', CFTYP_ROOTFRAC(1,3)
CFNAM_RSMIN(1)      = 'RSMIN.DAT             ', CFTYP_RSMIN(1)
CFNAM_Z0_O_Z0H(1)  = 'Z0_O_Z0H.DAT         ', CFTYP_Z0_O_Z0H(1)
CFNAM_GAMMA(1)      = 'GAMMA.DAT             ', CFTYP_GAMMA(1)
CFNAM_WRMALX_CF(1) = 'WRMALX_CF.DAT          ', CFTYP_WRMALX_CF(1)
CFNAM_RGL(1)        = 'RGL.DAT             ', CFTYP_RGL(1)
CFNAM_CV(1)         = 'CV.DAT             ', CFTYP_CV(1)
CFNAM_ALBNIR_VEG(1) = 'ALBNIR_VEG.DAT         ', CFTYP_ALBNIR_VEG(1)
CFNAM_ALBVIS_VEG(1) = 'ALBVIS_VEG.DAT         ', CFTYP_ALBVIS_VEG(1)
CFNAM_ALBUV_VEG(1)  = 'ALBUV_VEG.DAT         ', CFTYP_ALBUV_VEG(1)
CFNAM_ALBNIR_SOIL(1) = 'ALBNIR_SOIL.DAT        ', CFTYP_ALBNIR_SOIL(1)
CFNAM_ALBVIS_SOIL(1) = 'ALBVIS_SOIL.DAT        ', CFTYP_ALBVIS_SOIL(1)
CFNAM_ALBUV_SOIL(1) = 'ALBUV_SOIL.DAT        ', CFTYP_ALBUV_SOIL(1)
CFNAM_GMES(1)       = 'GMES.DAT             ', CFTYP_GMES(1)
CFNAM_RE25(1)       = 'RE25.DAT             ', CFTYP_RE25(1)
CFNAM_BSLAI(1)      = 'BSLAI.DAT             ', CFTYP_BSLAI(1)
CFNAM_LAIMIN(1)     = 'LAIMIN.DAT            ', CFTYP_LAIMIN(1)
CFNAM_SEFOLD(1)     = 'SEFOLD.DAT            ', CFTYP_SEFOLD(1)
CFNAM_GC(1)         = 'GC.DAT             ', CFTYP_GC(1)
CFNAM_DMAX(1)       = 'DMAX.DAT             ', CFTYP_DMAX(1)
CFNAM_F2I(1)        = 'F2I.DAT             ', CFTYP_F2I(1)
CFNAM_H_TREE(1)     = 'H_TREE.DAT            ', CFTYP_H_TREE(1)
CFNAM_CE_NITRO(1)   = 'CE_NITRO.DAT          ', CFTYP_CE_NITRO(1)
CFNAM_CF_NITRO(1)   = 'CF_NITRO.DAT          ', CFTYP_CF_NITRO(1)
CFNAM_CNA_NITRO(1)  = 'CNA_NITRO.DAT         ', CFTYP_CNA_NITRO(1)

```

/

&NAM\_DATA\_TEB

```

NROOF_LAYER = 3 ,
CFNAM_ALB_ROOF      = 'ALB_ROOF.DAT           ', CFTYP_ALB_ROOF(1)
CFNAM_EMIS_ROOF     = 'EMIS_ROOF.DAT          ', CFTYP_EMIS_ROOF(1)
CFNAM_HC_ROOF(1)    = 'HC_ROOF.DAT           ', CFTYP_HC_ROOF(1)
CFNAM_HC_ROOF(2)    = 'HC_ROOF.DAT           ', CFTYP_HC_ROOF(2)
CFNAM_HC_ROOF(3)    = 'HC_ROOF.DAT           ', CFTYP_HC_ROOF(3)
CFNAM_TC_ROOF(1)    = 'TC_ROOF.DAT           ', CFTYP_TC_ROOF(1)
CFNAM_TC_ROOF(2)    = 'TC_ROOF.DAT           ', CFTYP_TC_ROOF(2)
CFNAM_TC_ROOF(3)    = 'TC_ROOF.DAT           ', CFTYP_TC_ROOF(3)
CFNAM_D_ROOF(1)     = 'D_ROOF.DAT           ', CFTYP_D_ROOF(1)
CFNAM_D_ROOF(2)     = 'D_ROOF.DAT           ', CFTYP_D_ROOF(2)

```

```

CFNAM_D_ROOF(3)      = 'D_ROOF.DAT', CFTYP_D_ROOF(3)
NROAD_LAYER = 3 ,
CFNAM_ALB_ROAD      = 'ALB_ROAD.DAT', CFTYP_ALB_ROAD
CFNAM_EMIS_ROAD     = 'EMIS_ROAD.DAT', CFTYP_EMIS_ROAD
CFNAM_HC_ROAD(1)    = 'HC_ROAD.DAT', CFTYP_HC_ROAD(
CFNAM_HC_ROAD(2)    = 'HC_ROAD.DAT', CFTYP_HC_ROAD(
CFNAM_HC_ROAD(3)    = 'HC_ROAD.DAT', CFTYP_HC_ROAD(
CFNAM_TC_ROAD(1)    = 'TC_ROAD.DAT', CFTYP_TC_ROAD(
CFNAM_TC_ROAD(2)    = 'TC_ROAD.DAT', CFTYP_TC_ROAD(
CFNAM_TC_ROAD(3)    = 'TC_ROAD.DAT', CFTYP_TC_ROAD(
CFNAM_D_ROAD(1)     = 'D_ROAD.DAT', CFTYP_D_ROAD(1)
CFNAM_D_ROAD(2)     = 'D_ROAD.DAT', CFTYP_D_ROAD(2)
CFNAM_D_ROAD(3)     = 'D_ROAD.DAT', CFTYP_D_ROAD(3)
NWALL_LAYER = 3 ,
CFNAM_ALB_WALL      = 'ALB_WALL.DAT', CFTYP_ALB_WALL
CFNAM_EMIS_WALL     = 'EMIS_WALL.DAT', CFTYP_EMIS_WALL
CFNAM_HC_WALL(1)    = 'HC_WALL.DAT', CFTYP_HC_WALL(
CFNAM_HC_WALL(2)    = 'HC_WALL.DAT', CFTYP_HC_WALL(
CFNAM_HC_WALL(3)    = 'HC_WALL.DAT', CFTYP_HC_WALL(
CFNAM_TC_WALL(1)    = 'TC_WALL.DAT', CFTYP_TC_WALL(
CFNAM_TC_WALL(2)    = 'TC_WALL.DAT', CFTYP_TC_WALL(
CFNAM_TC_WALL(3)    = 'TC_WALL.DAT', CFTYP_TC_WALL(
CFNAM_D_WALL(1)     = 'D_WALL.DAT', CFTYP_D_WALL(1)
CFNAM_D_WALL(2)     = 'D_WALL.DAT', CFTYP_D_WALL(2)
CFNAM_D_WALL(3)     = 'D_WALL.DAT', CFTYP_D_WALL(3)
CFNAM_Z0_TOWN       = 'Z0_TOWN.DAT', CFTYP_Z0_TOWN
CFNAM_BLD           = 'BLD.DAT', CFTYP_BLD
CFNAM_BLD_HEIGHT    = 'BLD_HEIGHT.DAT', CFTYP_BLD_HEIG
CFNAM_WALL_O_HOR    = 'WALL_O_HOR.DAT', CFTYP_WALL_O_H
CFNAM_H_TRAFFIC     = 'H_TRAFFIC.DAT', CFTYP_H_TRAFFI
CFNAM_LE_TRAFFIC    = 'LE_TRAFFIC.DAT', CFTYP_LE_TRAFF
CFNAM_H_INDUSTRIY   = 'H_INDUSTRIY.DAT', CFTYP_H_INDUST
CFNAM_LE_INDUSTRIY  = 'LE_INDUSTRIY.DAT', CFTYP_LE_INDUS

/

&NAM_FRAC          LECOCLIMAP = F,
CFNAM_SEA          = 'SEA.DAT', CFTYP_SEA
CFNAM_WATER        = 'WATER.DAT', CFTYP_WATER
CFNAM_NATURE       = 'NATURE.DAT', CFTYP_NATURE
CFNAM_TOWN         = 'TOWN.DAT', CFTYP_TOWN

/

&NAM_PGD_GRID      CGRID = 'LONLAT REG'
/

&NAM_LONLAT_REG    XLONMIN = 0. ,
                   XLONMAX = 0.5 ,
                   XLATMIN = 0. ,
                   XLATMAX = 0.5 ,
                   NLON   = 1 ,
                   NLAT   = 1

/

&NAM_PGD_SCHEMES   CNATURE = 'ISBA ' ,
                   CSEA   = 'SEAFIX' ,
                   CTOWN  = 'TEB ' ,
                   CWATER  = 'WATFIX'

/

&NAM_ZS            XUNIF_ZS = 0.

```

```

/
&NAM_ISBA          XUNIF_CLAY    = 0.4          ,
                   XUNIF_SAND    = 0.2          ,
                   XUNIF_RUNOFFB  = 0.5          ,
                   CISBA         = '3-L'        ,
                   CPHOTO        = 'NON'       ,
                   NPATCH       = 1           ,
                   NGROUND_LAYER = 3           ,
/

&NAM_PREPFILE      CPREPFILE = 'PREP'
/
&NAM_PREP_SURF_ATM NYEAR  = 2004,
                   NMONTH  = 10,
                   NDAY    = 25,
                   XTIME   = 21600.
/
&NAM_PREP_SEAFLUX  XSST_UNIF = 285.,
                   NYEAR  = 2004,
                   NMONTH = 10,
                   NDAY   = 25,
                   XTIME  = 21600.
/
&NAM_PREP_WATFLUX  XTS_WATER_UNIF = 285.,
                   NYEAR  = 2004,
                   NMONTH = 10,
                   NDAY   = 25,
                   XTIME  = 21600.
/
&NAM_PREP_TEB      XTI_ROAD= 285.,
                   XTI_BLD = 285.,
                   XTS_ROAD= 285.,
                   XTS_ROOF= 285.,
                   XTS_WALL= 285.,
                   XWS_ROAD= 0.,
                   XWS_ROOF= 0.,
                   NYEAR  = 2004,
                   NMONTH = 10,
                   NDAY   = 25,
                   XTIME  = 21600.
/
&NAM_PREP_ISBA     XHUG_SURF = 0.2,
                   XHUG_ROOT = 0.2,
                   XHUG_DEEP = 0.2,
                   XTG_SURF  = 285.,
                   XTG_ROOT  = 288.,
                   XTG_DEEP  = 292.,
                   NYEAR  = 2004,
                   NMONTH = 10,
                   NDAY   = 25,
                   XTIME  = 21600.
/
&NAM_PREP_ISBA_SNOW CSNOW = '3-L'
/

```



```

&NAM_IO_OFFLINE      LPRINT = T ,
                     CFORCING_FILETYPE = 'NETCDF' ,
                     CSURF_FILETYPE = 'LFI' ,
                     CTIMESERIES_FILETYPE = 'NETCDF' ,
                     LWRITE_COORD = T,
                     LSET_FORC_ZS=T
/

&NAM_DIAG_SURFm     LSURF_BUDGET = T ,
                     N2M = 1
/
&NAM_DIAG_SURF_ATMn LFRAC = T
/
&NAM_DIAG_ISBAn     LPGD = T ,
                     LSURF_EVAP_BUDGET = T ,
                     LSURF_MISC_BUDGET = T
/
&NAM_DIAG_TEBn      LSURF_MISC_BUDGET = T
/
&NAM_SGH_ISBAn      CRUNOFF = "WSAT"
/
&NAM_ISBAn           CROUGH = "Z04D" ,
                     CSCOND = "NP89" ,
                     CALBEDO = "DRY" ,
                     CC1DRY = 'DEF' ,
                     CSOILFRZ = 'DEF' ,
                     CDIFSFCOND = 'DEF' ,
                     CSNOWRES = 'DEF' ,
                     CCPSURF = 'DRY'
/
&NAM_CH_ISBAn        CCH_DRY_DEP = "WES89 "
/
&NAM_SEAFLUXn        CSEA_ALB = "TA96"
/
&NAM_CH_SEAFLUXn     CCH_DRY_DEP = "WES89 "
/
&NAM_CH_WATFLUXn     CCH_DRY_DEP = "WES89 "
/
&NAM_CH_TEBn         CCH_DRY_DEP = "WES89 "
/

```