

---

# **ech2o Documentation**

*Release 5.1.2ns\_beta*

**Marco Maneta**

**Dec 06, 2018**



<b>1</b>	<b>Installation</b>	<b>3</b>
<b>2</b>	<b>Tutorial</b>	<b>5</b>
2.1	Model set-up . . . . .	5
2.1.1	Introduction . . . . .	5
2.1.2	The CSF PCRaster format . . . . .	5
2.1.3	The configuration file . . . . .	5
2.1.4	Preparing the database . . . . .	5
2.1.5	Climate files . . . . .	7
2.1.6	Forest and species data . . . . .	10
2.2	Case study . . . . .	14
2.2.1	Configuration file . . . . .	15
2.2.2	Populating the database . . . . .	15
2.2.3	Reporting results . . . . .	21
2.2.4	Running the program . . . . .	21
2.2.5	Spinning-up the model . . . . .	22
2.3	Configuration file keywords . . . . .	24
<b>3</b>	<b>Indices and tables</b>	<b>37</b>



This document provides a tutorial that guides users through the process of preparing the files, running and visualizing an ech2o simulation.



# CHAPTER 1

---

## Installation

---

Download the installation package and follow the instructions for your operating system. The installation package includes the sample dataset used in this tutorial. Alternatively, you can download the sample files for the [ECH2O case study](#)

To pre-process inputs and to visualize model results you will need to instal PCRASTER, which can be downloaded for free from <http://pcraster.geo.uu.nl>





## 2.1 Model set-up

### 2.1.1 Introduction

### 2.1.2 The CSF PCRaster format

Ech2o reads spatial information using the binary raster format (cross-system format, CSF) used in the free GIS **PCRaster**. By using this format, full GIS capability for data pre-processing, post-processing and visualization is added to Ech2o.

### 2.1.3 The configuration file

The configuration file is the main communication interface with ECH2O. It is a plain text file with pairs of keywords and values that provides the information that needs to run. This includes information on the location of the files, simulation and time step length, module options and the choice of state and diagnostic variables that the user wants reported (written) to the drive.

The list of keywords in the current version of the configuration file (v1.22) is shown in appendix [appendixb]

### 2.1.4 Preparing the database

#### Creating a base map and importing the elevation model

The first recommended step in the preparation of the database to run is to prepare a base map holding information on the geometry of the domain grid (dimension, resolution, etc). This map can be generated when importing the digital elevation model (DEM) basemap as explained below.

The easiest way to generate the base maps is to obtain a DEM in ArcInfo ascii raster format. needs that all maps are in planar coordinates, with lat-long coordinates in meters, such as the UTM projection. If the map is obtained in other projection using degrees a reprojection of the map is necessary using ArcGIS or any other external tool.

Move to the example folder provided with the package, open the file named with a text editor and check the metadata header with information on the geometry of the raster image.

Within the PCRaster environment, type

```
mapattr base.map
```

to start the interface and create a new blank base map named `base.map`. Introduce the number of rows and columns as indicated in the metadata of the ascii raster image. Choose the *'scalar'* datatype and the *'small real'* cell representation. If the projection is UTM you may want to indicate a *'y increases from bottom to top'* projection. Provide the coordinates for the x upper left corner and for the y upper left corner and the cell resolution.

Please, note that the ArcInfo standard provides information for the lower left corner. You can calculate the value of the upper left y coordinate by adding to the lower left coordinate the result of multiplying the number of rows by the resolution.

Once this information is provided, hit 'q' and answer 'y' to write the newly created map to the drive. Display the map to ensure it has the correct dimensions:

```
aguila base.map
```

This base map will be used to import all other maps and to ensure all the maps in the database have the exact same geometry. To import the ArcInfo DEM map into the CSF PCRaster format type

```
asc2map -a --clone base.map dem.asc dem.map
```

This command indicates that we are importing an ascii file named `dem.asc` into the PCRaster format with name `dem.map`, that the imported file has ArcInfo ascii grid format and that we are cloning the geometry of our `base.map`.

Display the map to check it has been correctly imported

```
aguila dem.map
```

To display it in 3D you can type

```
aguila -3 dem.map
```

These maps will form the core of the database from which many of the other necessary maps can be derived.

### Delineating the drainage network

The drainage network is derived from the DEM using a steepest-descent algorithm on the 8 neighbor window around each cell. From a PCRaster environment type the command

```
pcrcalc ldd.map = lddcreate(dem.map, 1e9, 1e9, 1e9, 1e9)
```

This command instructs PCRaster to calculate the local drainage direction (ldd) for each cell using the `dem()` and save the drainage network in a map called `ldd.map`. The large numbers included as the final four arguments to the `lddcreate` function are options to remove pits and core areas (see PCRaster documentation on `lddcreate` for more details). Display the results with `aguila` to visually inspect the drainage network. You may have to zoom in to see the details of the network.

Pits and outlets are coded with the value 5 in the resulting map. These cells flow nowhere and are considered flow sinks. There is at least one sink in each basin (the outlet). Mostly we will want to have a continuous flow network towards the outlet (unless we are working on a karst area or similar), so if we see internal flow sinks it may be due to errors in the DEM that to some extent can be corrected with some of the functions in PCRaster (see PCRaster documentation for this)

For technical reasons, needs a buffer of at least 1 cell of no-data (MV) around the drainage network (i.e. the edges of the ldd image must be no-data or missing value cells). The easiest way is to calculate the ldd from a DEM image that has blank cells (no data or missing values) beyond the domain of interest and that the domain of interest does not reach the edge of the image.

## Soil characteristics and surface properties

needs information on the surface characteristics (slope and rugosity) and soil characteristics (porosity, depth, etc) of the area of interest. Because this information is spatially variable, it is introduced in as maps. While some terrain properties such as its slope can be directly calculated from the DEM, information on the spatial distribution of most other properties listed in Table [tab:soilvars] need to be obtained from surveys, external databases such as SSURGO, CONUS-SOIL, etc (e.g. <http://www.soilinfo.psu.edu>).

Property	Units
Slope	$mm^{-1}$
Rugosity	$m$
Hydraulic conductivity	$ms^{-1}$
Porosity	$m^3m^{-3}$
Air entry pressure	$m$
Brooks Corey $\lambda$	•
Residual soil moisture	$m^3m^{-3}$
Soil depth	$m$
Veg wat use par 1	•
Veg wat use par 2	•

Table: Table [tab:soilvars]: Soil/surface properties and corresponding units needed to run Ech2o

The  $\lambda$  parameter in the Brooks and Corey model is the inverse of the pore size distribution index. Typical values for the Brooks and Corey  $\lambda$  for a number of textures is shown in Figure [fig:BCValues].

### 2.1.5 Climate files

organizes the climate data in a set of binary files containing the necessary information to construct the time dependent spatial fields of atmospheric inputs. All maps related to climate must be placed in the folder identified in the *Clim\_Maps\_Folder* key of the configuration file.

The spatial distribution of climate data is done according to discrete climate zones with unique identifiers that define areas of the domain with constant values for a given climate input. These climate zones can be constructed using Voronoi polygons, using irregular regions following elevation and aspect bands, or simply using a regular orthogonal spatial grid. This information on the climate zones is provided as a CSF PcRaster map. Figure [fig:ClimZone] is an example of a climate zone map using an orthogonal grid.

A time series of climate information for each specific climate zone is associated with each of these zones through a unique identifier that links the climate zone and a specific column of the binary climate file.

ECH2O reads climate files in a specific binary format that can be constructed from a text file using the `asc2c` utility provided with the modeling package. The format of the text file needed to run is explained below and summarized in box [box:climformat]. Data must be space or tab separated except the first line that must end with a carriage return.

[box:climformat]

Table 1. Clapp and Hornberger (1978) parameters for equation (27) based on analysis of 1845 soils. Values in parentheses are standard deviations.

Soil Texture	Porosity $n$	$K_{sat}$ (cm/hr)	$ \psi_a $ (cm)	$\lambda$
Sand	0.395 (0.056)	63.36	12.1 (14.3)	4.05 (1.78)
Loamy sand	0.410 (0.068)	56.16	9 (12.4)	4.38 (1.47)
Sandy loam	0.435 (0.086)	12.49	21.8 (31.0)	4.9 (1.75)
Silt loam	0.485 (0.059)	2.59	78.6 (51.2)	5.3 (1.96)
Loam	0.451 (0.078)	2.50	47.8 (51.2)	5.39 (1.87)
Sandy clay loam	0.420 (0.059)	2.27	29.9 (37.8)	7.12 (2.43)
Silty clay loam	0.477 (0.057)	0.612	35.6 (37.8)	7.75 (2.77)
Clay loam	0.476 (0.053)	0.882	63 (51.0)	8.52 (3.44)
Sandy clay	0.426 (0.057)	0.781	15.3 (17.3)	10.4 (1.64)
Silty clay	0.492 (0.064)	0.371	49 (62.0)	10.4 (4.45)
Clay	0.482 (0.050)	0.461	40.5 (39.7)	11.4 (3.7)

Fig. 1: Brooke and Corey soil parameters for different textures. From Dingman, L(2002). Physical Hydrology, 2nd Ed. Prentice Hall, 646p .

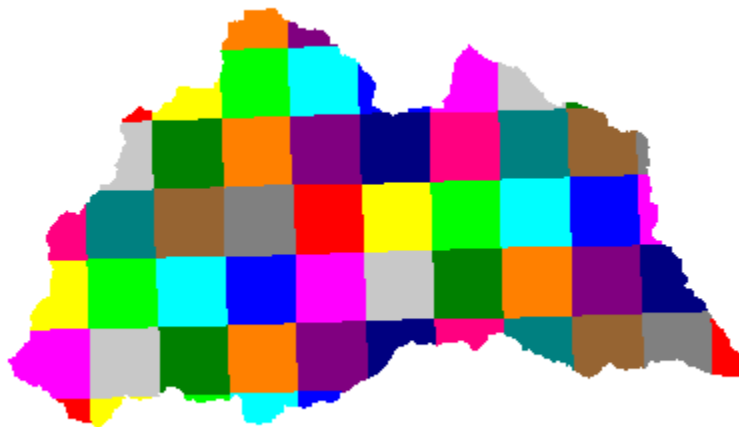


Fig. 2: Example of a climate zone map using a regular grid to accommodate input from a regional climate model

```

Comment [up to 256] (character)
NumTimeSteps [1] (integer number)
TimeSteps [NumTimeSteps] (real number)
NumZones [1] (integer number)
ZoneId [NumZones] (integer number)
Data [NumTimeSteps x NumZones] (real number)

```

Box [box:climformat]: ASCII climate file format. The number in square brackets is the number items allowed of the type indicated in parentheses

The first line of the file is a user's comment that typically includes a description of the contents of the file such as the what variable is represented in the file (precipitation, air temperature, etc), its source, units, etc. The size of the comment cannot exceed 256 characters including white spaces. The line may be left blank but the line must still exist (i.e. even if there is no information there must be a blank line).

The second line is the number of time steps included in the database. It must be a single integer.

The next line identifies the time steps in arbitrary units (e.g. 0.5 1 1.5... hours or 1 2 3 4... days). it is a space- or tab-separated list of real numbers containing exactly `NumTimeSteps` elements. The elements in this list are read with single precision (32 bits).

The next line is the number of spatial climate zones for which a time series is provided in the file. It must be a single integer.

The next line lists the climate zone identifiers as per the climate zone map that will be used during the simulations. This list is space- or tab-separated containing exactly `NumZones` integer numbers.

The final group of numbers contains the actual climate data. It is a matrix of real numbers with `NumTimeSteps` rows (a row per time step) and `NumZones` columns (one column per time zone listed in the header). Each column representing data for a zone must be ordered according to the order the zones were listed in the header. Elements in this matrix are read with single precision (32 bits).

Box [box:climfileex] gives An example of a climate file correctly formatted is

[box:climfileex]

```

Windspeed in m/s. Station 1b2. J Doe
4
0.5 1 1.5 2
2
1 2
2.4 2.1
2.0 2.8
1.9 2.0
0.5 1.2

```

Box [box:climfileex]: Example of ascii climate file with 4 time steps (0.5, 1, 1.5, and 2) and 2 climate zones (1 and 2)

Table [tab:climvars] File format of vegetation parameters needed to run the vegetation component of [tab:climvars]

Table 1: Climate Inputs

Variable	Units
Precipitation	$ms^{-1}$
Average air temperature	<i>centigrades</i>
Maximum air temperature	<i>centigrades</i>
Minimum air temperature	<i>centigrades</i>
Relative Humidity	fraction of saturation
Wind speed	$ms^{-1}$
Incoming long wave radiation	$Wm^{-2}$
Incoming solar radiation	$Wm^{-2}$

Text files with this format need to be converted into the appropriate binary climate format used by using the provided utility

```
asc2c input_text_file.asc output.bin
```

Where represents the name of the appropriately formatted text file containing the climate data and represents the name that will use to write the resulting binary file. The format of the binary file follows the same structure of the ascii file using 8 bit characters, 32 bit signed integers, and 32 bit signed floats.

Eight climate variables are needed to run ECH2O, each in its own binary file. expects the data in the files to be in some specific units. Table [tab:climvars] lists the eight needed climate variables and the corresponding units in which the data must be provided.

Two additional files in CSF PcRaster map format are necessary in *Clim\_Maps\_Folder*, one is a map with the temperature threshold (in  $^{\circ}C$ ) for rain to snow transition. This map can be constant or the threshold can change in space. The second file is a convenience map of precipitation multiplication factors that permits to manipulate and improve the spatial distribution of precipitation even when using coarse climate zones. The precipitation assigned to a pixel in the climate zone from the corresponding *.bin* file will be multiplied by the factor specified in the same pixel of this map before being used in .

## 2.1.6 Forest and species data

In this version is designed to simulate evergreen vegetation and a herbaceous understory. It is also designed to broad types of vegetation (e.g. firs, pines) with a general functional behavior instead of simulating specific species. Multiple vegetation types can be simulated, the number of them is supplied in the *Number\_of\_Species* keyword of the configuration file.

needs two type of information to set up the ecological module: 1) vegetation parameters, and 2)initial condition of the state variables tracked.

### Vegetation Parameters file

The vegetation parameters file must be located in the *Maps\_Folder* folder indicated in the configuration file. The name of the file must be indicated in the *Species\_Parameters* keyword.

The contents of the file is ascii text that describes the functional characteristics of the different vegetation types that will be included in the simulation. It contains the time-invariant parameters that define the behavior of plants.

The first line of the file contains two tab- or space-separated integers. The first integer indicates the number of vegetation types included in the file. The second integer must be the number 43, which is the number of information items that needs to be supplied for each vegetation type.

Below the first line there will be a line per vegetation type containing 43 items of information. The format and items of information are listed in Table [tab:vegparams].

Table [tab:vegparams]: Format of the vegetation parameters file

[tab:vegparams]

```

line 1: numSpecs      NumParams
In each line from line 1 to line numSpecs+1: 43 Comma or
tab separated numbers with the following elements:

SpeciesID NPP/GPPRatio      gsmax      CanopyQuantumEffic
MaxForestAge OptimalTemp MaxTemp MinTemp
FoliageAllocCoef_a FoliageAllocCoef_b
StemAllocCoef_a      StemAllocCoef_b gs_light_coeff gs_vpd_coeff
gs_psi_d gs_psi_c WiltingPnt      SpecificLeafArea
SpecificRootArea Crown2StemDRat
TreeShapeParam      WoodDens Fhdmax Fhdmin LeafTurnoverRate
MaxLeafTurnoverWaterStress LeafTurnoverWaterStressParam
MaxLeafTurnoverTempStress LeafTurnoverTempStressParam
ColdStressParam      RootTurnoverRate MaxCanStorageParam albedo
emissivity KBeers CanopyWatEffic
is_grass
DeadGrassLeafTurnoverRate DeadGrassLeafTurnoverTempAdjustment

```

**SpeciesID** A unique vegetation identifier (integer).

**NPP/GPPRatio** A NPP to GPP ratio representing a constant respiration loss. Positive real smaller than 1. Typical value around 0.47

**gsmax** Maximum stomatal conductance in  $ms^{-1}$ . Typical value around 0.006

**CanopyQuantumEffic** Canopy quantum efficiency representing the light use efficiency, in  $gCJ^{-1}$  (grams of carbon per absorbed joule of photosynthetically active radiation. Typical value around 0.0000018

**MaxForestAge** Typical maximum age for the vegetation, in years

**OptimalTemp** Optimal growth temperature for the vegetation type, in degrees C

**MaxTemp** Maximum temperature of comfort for the species, in degrees C

**MinTemp** Minimum temperature of comfort for the species, in degrees C

**FoliageAllocCoef\_a** Foliage allocation coefficient as per 3PG model. Typical value around 2.235

**FoliageAllocCoef\_b** Foliage allocation coefficient as per 3PG model. Typical value around 0.006

**StemAllocCoef\_a** Stem allocation coefficient as per 3PG model. Typical value around 3.3

**StemAllocCoef\_b** Stem allocation coefficient as per 3PG model. Typical value around 0.0000006

**gs\_light\_coeff** Parameter controlling stomatal sensitivity to light. Typical value around 300

**gs\_vpd\_coeff** Parameter controlling stomatal sensitivity to vapor pressure deficit. Typical value around 0.002

**gs\_psi\_d** Soil moisture suction potential at which stomatal function is reduced by 50%. Typical value around 0.5 m of suction head

**gs\_psi\_c** Parameter controlling stomatal sensitivity to soil moisture content. Typical value around 2

**WiltingPnt** Volumetric soil water content at wilting point, dependent on plant and soil characteristics.

**SpecificLeafArea** Specific leaf area, in  $m^2KgC^{-1}$

**SpecificRootArea** Specific root area, in  $m^2KgC^{-1}$

- Crown2StemDRat** Allometric parameter controlling the crown to stem diameter ratio as per TreeDyn.
- TreeShapeParam** Tree shape parameter as per TreeDyn. An often appropriate value is 0.4
- WoodDens** Wood density, in  $gCm^{-2}$
- Fhdmax** Maximum allowed ratio of tree height to stem diameter
- Fhdmin** Minimum allowed ratio of tree height to stem diameter
- LeafTurnoverRate** Base leaf turnover rate, in  $s^{-1}$
- MaxLeafTurnoverWaterStress** Maximum leaf turnover rate due to water stress, in  $s^{-1}$
- LeafTurnoverWaterStressParam** Parameter controlling increased leaf turnover due to water stress
- MaxLeafTurnoverTempStress** Maximum leaf turnover rate due to temperature stress, in  $s^{-1}$
- LeafTurnoverTempStressParam** Parameter controlling increased leaf turnover due to temperature stress
- ColdStressParam** (degC)
- RootTurnoverRate** Base root turnover rate, in  $s^{-1}$
- MaxCanStorageParam** Maximum water storage capacity of the canopy, in  $m$
- albedo** Albedo of vegetation
- emissivity** Emissivity of vegetation
- KBeers** Light extinction coefficient for the canopy as per Beer's law
- CanopyWatEffic** Water use efficiency of the canopy, in terms of grams of carbon assimilated per meter of transpired water,  $gCm^{-1}$
- is\_grass** Switch that indicates if the vegetation type is herbaceous (1) or not (0)
- DeadGrassLeafTurnoverRate** Base Rate of decomposition of dry grass leaves,  $s^{-1}$ . The value is used only if  $is\_grass=1$  although a value needs to be supplied in all cases
- DeadGrassLeafTurnoverTempAdjustment** Temperature threshold that triggers the decomposition of dry grass leaves, deg C. The value is used only if  $is\_grass=1$  although a value needs to be supplied in all cases

## Initial conditions for vegetation state variables

Information on the density of trees, relative canopy cover, root density, leaf area index, vegetation age, vegetation effective height, and tree basal area is necessary to initialize the status of vegetation. There is two ways to provide this information: using tables and using maps.

### Initialization using tables

Initialization of the state variables for vegetation using tables is often easier during the first model run. Ech2o can be initialized with tables by setting *Species\_State\_Variable\_Input\_Method* = tables in the configuration file.

This type of initialization relies on the concept of 'vegetation patches', which are discrete, arbitrarily-shaped regions in the study area where vegetation is initialized with constant values. A patch can have multiple vegetation types, each identified with the *SpeciesID* listed in the vegetation parameter file.

Patches are given to as a map in the '*ForestPatches*' keyword of the configuration file. This map must be included in the *Maps\_Folder* folder indicated in the configuration file. The map contains at least one discrete region (patch) identified with an integer. Please note that patches need not be continuous. A patch can be composed of different disconnected small regions scattered through the domain with the same integer identifier.



The initialization of vegetation types in each path is done through a number of ascii tables with a format described below. The tables must be placed in the *Maps\_Folder* folder indicated in the configuration file and the names for each variable paired with the appropriate key in the configuration file. A description of the tables is given below

### Species\_Proportion\_Table

: Table containing the proportion of each patch that is occupied by each vegetation type. In the current version of the model this is a time-invariant variable since there is no vegetation dispersal and encroachment module. If a vegetation type does not exist for a patch, indicate a zero in the column for that species in a patch.

### Species\_StemDensity\_Table

: Table containing the tree density of each vegetation type in their share of patch, in trees per sq. meter. In the current version of the model this is a time-invariant variable since there is no vegetation dispersal and encroachment module.

### Species\_LAI\_Table

: Table containing the initial LAI of each vegetation type. note that LAI is defined as the area of leaves over the projected canopy area and not area of leaves over patch or pixel area.

### Species\_AGE\_Table

: Table containing the average age of trees of each vegetation type in each patch. In years.

### Species\_BasalArea\_Table

: Table containing the total basal area of each type of vegetation in each patch, in square meters.

### Species\_Height\_table

: Table containing the effective height of each type of vegetation in each patch, in meters.

### Species\_RootMass\_table

: Table containing the average root mass of each type of vegetation in each patch, in grams per square meters.

All tables have identical format as described in Table [tab:vegvars].

Table [tab:vegvars]: Format of the vegetation variables file

[tab:vegvars]

```
line 1: numPatches NumSpecies+1
In each line from line 1 to line numPatches+1: PatchID
followed by NumSpecies comma or tab separated
numbers with initial information on vegetation variables.
The information for each vegetation type is listed in
the same order they appear in the vegetation parameter
file.
```

**numPatches** Number of patches with unique identifiers in file associated to *ForestPatches*.

**NumSpecies** Is the number of simulated vegetation types.

### PatchID

The unique integer identifier for the vegetation patch as identified in the patch map.

---

**Important:** The information for the vegetation type is introduced in the order in which the vegetation types are listed in the vegetation parameterfile (i.e. first number after the *PatchID* item corresponds to the topmost vegetation type listed in the vegetation parameter file, and so on).

---

## Initialization using maps

If distributed information is available to initialize the vegetation variables or if a complete run has already been performed it is possible to initialize the variables using maps instead of tables and provide variability within each patch.

To initialize the vegetation variables this way set *Species\_State\_Variable\_Input\_Method* = maps in the configuration file. With the configuration, will look for the following maps in the folder specified in *Maps\_Folder*:

The species are identifying by an index within square brackets in the file name. The index starts at 0, which identifying the topmost vegetation type identified in the vegetation parameter file (e.g. for a run with two vegetation types the leaf area index is initialized with two maps, *emlai[0].map* and *lai[1].map*, corresponding to the first and second vegetation types listed in the vegetation parameter file).

**p[0,...,NumSpecies-1].map** One map per vegetation type included in the simulation. The map contains the proportion of each pixel occupied by the vegetation type identifying by the index in the file name.

**root[0].map** One map per vegetation type included in the simulation. The map contains the root mass of the vegetation type identifying by the index in the file name, in  $gm-2$

**ntr[0,...,NumSpecies-1].map** One map per vegetation type included in the simulation. The map contains density of trees in the area of each pixel occupied by the vegetation type identified by the index in the file name. Trees per sq.meter.

**lai[0,...,NumSpecies-1].map** One map per vegetation type included in the simulation. The map contains the initial leaf area index in each pixel of the vegetation type identified by the index in the file name.

**hgt[0,...,NumSpecies-1].map** One map per vegetation type included in the simulation. The map contains the effective height in each pixel of the vegetation type identified by the index in the file name. In meters.

**bas[0,...,NumSpecies-1].map** One map per vegetation type included in the simulation. The map contains the total basal area in each pixel of the vegetation type identified by the index in the file name. In sq. meters.

**age[0,...,NumSpecies-1].map** One map per vegetation type included in the simulation. The map contains the age in each pixel of the vegetation type identified by the index in the file name. In years.

A way to produce these maps is to turn on the reporting flag for these maps during an initial run of using tables. Then rename the last time step of the corresponding files in the results folder with the appropriate names and copy these files to the maps folder. The case study included in this manual explains how initialize the model using this technique.

## 2.2 Case study

The folder named *CaseStudy*, which is distributed with the the model package, includes a sample dataset. This dataset is used in this case study to illustrate the process of creating a database and running for a mountain watershed. We will use the GIS functionality provided by PCRaster to assist us in the construction of the database. It is a good idea

to have the PCRaster documentation open in a browser tab and peruse it to learn more about the commands we will be using in these examples.

## 2.2.1 Configuration file

The configuration file presents the main communication interface with the model. The configuration file is a plain text file with pairs of keys and values. The values indicate options, paths to folders, or name of files that contain information needed by ECH2O. The easiest way to set the configuration file is to generate a template that can subsequently be edited. To generate a configuration file template, navigate to the *CaseStudy* folder and use the following command:

```
ech2o -g config.ini
```

Where the `-g` option indicates that we wish to generate a configuration file with the name `config.ini`. In the configuration file we provide information about the location of the database, the names of the files and also control the length of the run, the size of the time step, the outputs we want the model to produce and select a number of options.

Open the file with any text editor. In the *Folder* section of the file make sure the paths to the *Spatial* and *Climate* folders of the case study are correct. In these files we will be storing spatial and climate information. Also make sure the folder where Ech2o will write the results (*Results* folder) exists and the path is correct.

The maps to be read by ECH2O will be in the PCRaster cross-system format so make sure `MapTypes = csf`. Also we will be using tables to initialize the vegetation state variables so make sure `Species_State_Variable_Input_Method = tables`.

Turn on the infiltration and channel switches (1). We will use the aerodynamic resistance option as used in the Penman-Monteith equation (option 0) and the soil resistance as used by Ivanov (option 1).

We will simulate 1 year using daily time steps. We have to provide that information in seconds. The simulation will start at second 0 and will end at second 31536000. The simulation time step will be 86400. The Climate input time step and the report interval will be the same as the simulation time step (86400).

The next few sections is where the maps in the database are associated with parameters in the model. Give the appropriate file name for each parameter. A description of each parameter can be found in appendix a. At this point we should have generated all the needed files.

The report map section is a series of boolean switches (0-1) that turn on or off the reporting (writing to the results folder) of maps with the state variables. Turn on (=1) the variables that you like reported. Mind that writing maps to the disk is an expensive processes in terms of computer time and disk space.

## 2.2.2 Populating the database

### Creating a base map and importing the elevation model

First we will create the spatial section of the database. Open the command line and navigate to the *Spatial* folder of the case study. The maps created in this section should be placed in this folder.

The first recommended step in the preparation of the database to run is to prepare a base map holding information on the geometry of the domain grid (dimension, resolution, etc). This map can be generated when importing the digital elevation model (DEM) basemap as explained below.

The easiest way to generate the base maps is to obtain a DEM in ArcInfo ASCII raster format. needs maps in planar coordinates, with lat-long coordinates in meters, such as the UTM projection. If the map is obtained in other projection using degrees a reprojection of the map is necessary using ArcGIS or any other external tool.

Move to the example folder provided with the package, open the file named with a text editor and check the metadata header with information on the geometry of the raster image.

Within the PCRaster environment, type

```
mapattr base.map
```

to start the interface and create a new blank base map named `base.map`. Introduce the number of rows and columns as indicated in the metadata of the ascii raster image. Choose the *'scalar'* datatype and the *'small real'* cell representation. If the projection is UTM you may want to indicate a *'y increases from bottom to top'* projection. Provide the coordinates for the x upper left corner and for the y upper left corner and the cell resolution.

Please, note that the ArcInfo standard provides information for the lower left corner. You can calculate the value of the upper left y coordinate by adding to the lower left coordinate the result of multiplying the number of rows by the resolution.

Once this information is provided, press 'q' and answer 'y' to write the newly created map to the drive. Display the map to check it has the correct dimensions:

```
aguila base.map
```

This base map will be used to import all other maps and to ensure all the maps in the database have the exact same geometry. To import the ArcInfo DEM map into the CSF PCRaster format type

```
asc2map -a --clone base.map dem.asc DEM.map
```

This command indicates 1) that we are importing an ascii file named `dem.asc` into the PCRaster format with name `DEM.map`, 2) that the imported file has Arcinfo ascii grid format, and 3) that we are cloning the geometry of our `base.map`.

Display the map to check it has been correctly imported

```
aguila DEM.map
```

To display it in 3D you can type

```
aguila -3 DEM.map
```

These maps will form the core of the database from other necessary maps can be derived.

### Delineating the drainage network

The drainage network is derived from the DEM using a steepest-descent algorithm on the 8 neighbor window around each cell. From a PCRaster environment type the command:

```
pcrcalc ldd.map = lddcreate(DEM.map, 1e9, 1e9, 1e9, 1e9)
```

**Attention:** NOTE TO LINUX USERS Please, note that if you are following this tutorial in a linux computer you need to place the arguments to *pcrcalc* between quotes like

```
$ pcrcalc 'ldd.map = lddcreate(DEM.map, 1e9, 1e9, 1e9, 1e9)'
```

This command instructs PCRaster to calculate the local drainage direction (`ldd`) for each cell using the `dem` () and save the drainage network to a map called `ldd.map`. The large numbers included as the final four arguments to the *lddcreate* function are options to remove pits and core areas (see PCRaster documentation on *lddcreate* for more details). Display the results with `aguila` to visually inspect the drainage network. You may have to zoom in to see the details of the network.

Pits and outlets are coded with the value 5 in the resulting map. These cells flow nowhere and are considered flow sinks. There is at least one sink in each basin (the outlet). Mostly we will want to have a continuous flow network towards the outlet (unless we are working on a karst area or similar), so if we see internal flow sinks it may be due to errors in the DEM that to some extent can be corrected with some of the functions in PCRaster (see PCRaster documentation for this).

A map of the channels and the width of the channel is provided in the folder (). Inspect it using aguila and observe that cells with a channel have a positive number indicating the width of the channel in meters and cells without a channel have attribute 0 or nodata.

The resistance presented by the channel to flow is given by Manning's  $n$  coefficient. Values for Manning's  $n$  coefficient needs to be provided for each cell where the channel width is larger than 0. A map of Manning's  $n$  values in  $sm^{-\frac{1}{3}}$  for the example channel network is provided ()

The parameter controlling the seepage from the subsurface system to the channel lets us fine-tune subsurface-channel interactions. A good starting value for this parameter is 0.02 for the entire channel system. The larger the value, the more resistance to flow into the channel. We can produce this map using

```
pcrcalc chanparam.map = chanwidth.map/chanwidth.map * 0.02;
```

## Defining soil and surface properties

In this section we will create a set of maps that provide information on the soil and surface properties. Some of these properties can be derived from the DEM and for others we will use some simplifying assumptions about the spatial distribution of the properties.

The slope of the terrain can be obtained directly from the DEM using the following command

```
pcrcalc slope.map = slope(DEM.map)
```

This command will create a map named with the slope (rise over run) of the basin.

Now we will create a map with constant value 1 that will help us create maps of soil properties with a spatially uniform distribution.

```
pcrcalc unit.map = DEM.map/DEM.map
```

This operation divides the DEM map by itself to produce a map called holding 1 everywhere in the basin.

Now we use to construct maps of spatially uniform properties

```
pcrcalc albedo.map = unit.map * 0.3
pcrcalc emissivity.map = unit.map * 0.98
pcrcalc soilheatcap.map = unit.map * 2.205e6
pcrcalc soilthermalK.map = unit.map * 0.2
pcrcalc dampdepth.map = unit.map * 2
pcrcalc temp_damp.map = unit.map * 10
pcrcalc snowmeltCoeff.map = unit.map * 4.1e-8
pcrcalc randrough.map = unit.map * 0.05
pcrcalc psi_ae.map = unit.map * 0.2
pcrcalc BClambda.map = unit.map * 5.3
pcrcalc KvKh.map = unit.map * 0.4
pcrcalc theta_r.map = unit.map * 0.05
pcrcalc Wc.map = unit.map * 0.7
pcrcalc Wp.map = unit.map * 9
```

This will create maps of uniform albedo, surface emissivity, soil heat capacity, soil thermal conductivity, soil depth at which heat exchanges are negligible, initial soil temperature, snowmelt coefficient, terrain rugosity, soil air entry

pressure, Brooks and Corey  $\lambda$  parameter, vertical to horizontal hydraulic conductivity anisotropy ratio, residual soil moisture, and two soil parameter,  $W_c$ ,  $W_p$ , respectively, with values equal to the multiplying scalar in the right side of the expression.

To introduce some spatial variability in the simulation, we will assume that some geomorphologic sorting of soil particles distributes some key hydrologic properties throughout the basin. For instance, finer particles may get washed out of steep upslope areas and be deposited when water slows down in flatter areas downslope. This may produce deeper, more porous soils at the valley bottom with lower hydraulic conductivity than soils located higher up in the hillslopes.

To simulate such a geomorphologic driven distribution we can use a topographic index that is larger for flatter cells with large contributing areas (such as valley bottoms) and smaller for steep cells near the water divide.

```
pcrcalc topind.map = ln(accuflux(ldd.map,10000)/slope.map)
```

This expression uses the function *accuflux* to accumulate the area of the cells ( $10,000 m^2$ ) following the drainage direction and divides it by the map of slopes () that we created earlier. The function *ln* takes the logarithm of the result of the quotient to equalize the distribution of values, which is highly skewed due to the exponential distribution of the accumulated areas.

We will assume that this map describes the spatial distribution of soil depth, porosity and effective hydraulic conductivity. With the help of some scaling functions we produce the resulting fields for these soil properties:

```
pcrcalc soildepth.map = topind.map
    /areaaverage(topind.map,nominal(unit.map))
pcrcalc Keff.map = 1 / (soildepth.map * 36000)
pcrcalc poros.map = 1 / (1 + exp(0.01 * topind.map))
```

We will set initial conditions for the soil assuming the basin starts free of snow, with 50% of the pores saturated with water and with a temperature of  $10^\circ C$  throughout the basin:

```
pcrcalc swe.map = unit.map * 0
pcrcalc Soil_moisture_1.map = poros.map * 0.5
pcrcalc Soil_moisture_2.map = poros.map * 0.5
pcrcalc Soil_moisture_3.map = poros.map * 0.5
pcrcalc soiltemp.map = unit.map * 10
pcrcalc streamflow.map = unit.map * 0
```

We will also assume that the first hydraulic layer of the soil is 10 cm deep (0.1 m). We will also assume that the second hydraulic layer is 10 cm deep. will calculate the depth of the 3rd layer such that the sum of the three layers equals the soil depth at the pixel. Additionally, the first soil layer will contain 10% of the roots and the second layer will contain the remaining 90%. For simplicity we further assume that the bedrock at depth of the soil is impervious (leakance=0). This parameters varies between 0 (no flow boundary) and 1 (free drainage).

```
pcrcalc depth_soil1.map = unit.map * 0.1
pcrcalc depth_soil2.map = unit.map * 0.1
pcrcalc rootfrac1.map = unit.map * 0.1
pcrcalc rootfrac2.map = unit.map * 0.9
pcrcalc leakance.map = unit.map * 0.0
```

We will see later we will spin-up the model to equilibrate the initial conditions for the characteristics and climate of the basin.

## Defining vegetation parameters

For the sake of simplicity we will assume that there is only one type of forest homogeneously covering 60% of the basin (proportion of area covered in each forest patch is specified in file *SpecsProp.tab*).

The parameters that define the vegetation in the forest is provided in table [tab:exspecpars]

Table 2: Sample parameter configuration file for one tree species

Parameter	Species ID
SpeciesID	1
NPP/GPPRatio	0.47
gsmax(ms-1)	0.09
CanopyQuantumEffic(gCJ-1)	0.0000018
MaxForestAge(yrs)	290
OptimalTemp(C)	18
MaxTemp(C)	30
MinTemp(C)	0
FoliageAllocCoef_a	2.235
FoliageAllocCoef_b	0.006
StemAllocCoef_a	3.3
StemAllocCoef_b	6.00E-07
gs_light_coeff	300
gs_vpd_coeff	0.0019
gs_psi_d	0.2
gs_psi_c	2
WiltingPnt	0.05
SpecificLeafArea(m <sup>2</sup> g <sup>-1</sup> )	0.003
SpecificRootArea(m <sup>2</sup> kg <sup>-1</sup> )	0.022
Crown2StemDRat	0.25
TreeShapeParam	0.4
WoodDens(gCm <sup>-2</sup> )	220000
Fhdmax	15
Fhdmin	5
LeafTurnoverRate(s-1)	8.56E-09
MaxLeafTurnoverWaterStress(s-1)	0.000000018
LeafTurnoverWaterStressParam	0.2
MaxLeafTurnoverTempStress(s-1)	0.000000018
LeafTurnoverTempStressParam	0.2
ColdStressParam(degC)	1
RootTurnoverRate(s-1)	5.34E-09
MaxCanStorageParam(m)	0.0000624
albedo	0.1
emissivity	0.95
KBeers	0.55
CanopyWatEffic(gCm-1)	800
is_grass	0
DeadGrassLeafTurnoverRate(s-1)	0
DeadGrassLeafTurnoverTempAdjustment (degC)	0

The parameters are listed in the order they should appear in the vegetation configuration file. Make sure you include in the first line of the header the number of species in the file and the number of information items per species (2 43). For convenience, the information in Table [tab:exspecpars] is properly formatted in a parameter file named `SpeciesParams.tab`, which is provided in the folder of the case study.

The next step is providing information about the variables for vegetation. There are two ways to provide this information, through tables that provide constant variable values for each initial forest patch and through maps that provide spatially variable values.

The easiest way is to provide first the information using tables and spin-up the model to provide maps with the distributed variables. Then restart the model using the maps as initial forest conditions. If we are using tables we need to provide a map with the spatial distribution of the types of forest or *patches*. This spatial distribution is done using integer ID numbers for each patch. In this example we will assume that only one type of forest exist covering the entire area with ID 1. We can create the patch map using the `unit.map`:

```
pcrcalc patches.map = unit.map
```

The vegetation variables needed to run the model are the proportion of canopy coverage, the stem density, the leaf area index, the age, the total basal area, the species height and the root density of each species for each path. Each of these variables is contained in an individual file with the same format.

As mentioned earlier, we will assume that the canopy of vegetation type 1 (the only type) cover 60% of the basin. The canopy coverage file for this example would be

```
1 2
1 0.6
```

Where the first element in the first line indicate the number of patches (1), the second element is the number of covers in the patch (1 vegetation type + bare soil = 2). The second line indicates the patch ID for which this line is providing information (matching the appropriate ID in the file). the following numbers are the proportion of the patch covered by canopy for each vegetation type (only 1 in the case). The proportion of bare soil is calculated internally from this information. The information for each species must be entered in the same order that was provided in the table of vegetation parameters including 0.0 if there is no coverage of a specific species or vegetation type in a given patch. A file () with this information is included for convenience in the *example/Spatial* folder.

The same data structure is used in the files containing information for the other mandatory vegetation variables, for which files are conveniently provided: Stem density, leaf area index, stand age, total stand basal area, effective height and root density.

### Climate inputs

Navigate to the *Climate* folder of the case study. The maps generated in this section need to be placed in this folder. A climate zones map provides the information to spatially distribute the climate time series and should be created first. In this example we will partition our basin in ten climate zones following the elevation contours. The easiest way to create to do that is to reclassify the DEM in ten uniform elevation zones with unique integer IDs using a classification table (see PCRaster documentation for formats):

```
<,1430] 1
<1430,1580] 2
<1580,1730] 3
<1730,1880] 4
<1880,2030] 5
<2030,2180] 6
<2180,2330] 7
<2330,2480] 8
<2480,2630] 9
<2630,> 10
```

Assuming the previous table is saved under the name we can reclassify our climate zones map with the following command:

```
pcrcalc ClimZones.map = lookupnominal(ElevZonesClass.txt,
    ..\Spatial\DEM.map)
```

In the folder there are a set of example climate files providing climate information for the 10 time zones at a daily time step during 365 days starting at the beginning of the water year (October 1st). The temperature field has been



generated assuming a sinusoidal cycle of temperature. A standard environmental lapse rate has been used to distribute precipitation with elevation. Precipitation for the bottom-most zone has been generated randomly to simulate a typical semiarid climate with precipitation falling during fall and winter. A linear model has been used to simulate an increase of precipitation with altitude. The other climate variables have been generated using polynomial functions to simulate seasonality and are considered to be spatially uniform. You can import the files to a spreadsheet program like MSExcel and plot them to inspect the type of climate we are simulating.

In order to make these files usable for Ech2o we need to import them into binary format with the utility provided with Ech2o. This utility takes two arguments: the name of the properly formatted ascii file with the climate information and the desired name for the binary file to be written.

The following commands will import the climate files and generate the necessary binary files having the same name as the original text files but with a *.bin* extension.

```
asc2c Tavg.txt Tavg.bin
asc2c Tmin.txt Tmin.bin
asc2c Tmax.txt Tmax.bin
asc2c Precip.txt Precip.bin
asc2c Sdown.txt Sdown.bin
asc2c Ldown.txt Ldown.bin
asc2c RH.txt RH.bin
asc2c windspeed.txt windspeed.bin
```

We will introduce some random variability in the precipitation field using the isohyet map assuming no autocorrelation structure or directionality of the field. The random fluctuations are produced using a uniform distribution ranging with a range 0.5-1.5 to simulate precipitation fluctuations ranging from half to one and half times the incident local precipitation

```
pcrcalc isohyet.map = uniform(boolean(..\Spatial\unit.map))+0.5
```

### 2.2.3 Reporting results

The report time series section is another series of boolean switches that turn on or off the reporting (writing to the results folder) of time series for the specified variables. The spatial pixels for which the time series is produced are these indicated by the map. This map should contain the value zero or no data everywhere except for the pixels for which a time series is desired. These pixels should be marked with an integer ID that will be used to identify the time series in the resulting output file containing the time series information.

One way to create the map is by making use of a text file containing information on the location of the pixels to be monitored. This file should have one line per pixel to be monitored (probe). Each line contains the x and y coordinate of the pixel and the pixel identification number. The file located in the folder of this case study is an example of such a file with three probes.

Once this file is created you can import it to create a map using the PCRaster tool. Navigate back to the and transform the information contained in into a map using

```
col2map --clone base.map probes.txt Tsmask.map
```

### 2.2.4 Running the program

FIRST, MAKE SURE THE FOLDER WHERE ECH2O IS TOLD TO WRITE THE RESULTS EXISTS. ECH2O WILL NOT CREATE THE FOLDER IF IT DOES NOT EXIST AND WILL TERMINATE THE RUN WHEN IT ATTEMPTS TO WRITE TO THE NON-EXISTING FOLDER.

Open the configuration file in a text editor and replace the default input file names for the soil moisture keys with the correct filenames

```
Soil_moisture_1 = Soil_moisture_1.map
Soil_moisture_2 = Soil_moisture_2.map
Soil_moisture_3 = Soil_moisture_3.map
```

Once the database is complete and the configuration file correctly set we are ready to run Ech2o. This is simply done by navigating to the folder containing the ech2o configuration file and running the following command:

```
ech2o config.ini
```

Where stands for the name of the configuration file, typically `config.ini`, but this file can be named in any other way to differentiate different projects or runs.

After hitting enter you will see the splash screen with the version number and a report on the pre-processing steps (whether it was able to successfully read the files and create the components of the model run).

The screen reports information on the water mass (in  $m^3$ ) for the different components of the basin for each time step and information on the mass balance error (in % of the total input). The mass balance error should be a very small number (typically  $< 1.0e-10\%$ ). If the number is large or steadily increases as the simulation progresses it is an indication of some problem in the inputs.

Once the model has finished running you can inspect the results using or to display the timeseries files or the maps in the results folder.

For instance if you have reported discharge you can display it by typing

```
aguila OutletDisch.tab
```

Spatial time series can also be displayed. For instance if you have reported snow water equivalent maps series for one year at daily timesteps (365 time steps) you can inspect them with the command

```
aguila SWE00000.001+365
```

You can even drape them to the DEM. Assuming you are in the folder:

```
aguila -3 ../spatial\DEM.map + SWE00000.001+365
```

Also a file called is created in the root folder (where file is located). This file contains summary information on the water balance of the basin in total volumes of water ( $m^3$ ).

### 2.2.5 Spinning-up the model

As you see, the model diverges from the initial conditions provided and will finish with a very different spatial distribution of the state variables. Some of the variables will show a declining trend, others will show an increasing trend rather than a cycle. This indicates that the state of the model is not in equilibrium with the provided boundary conditions.

The process of running the model in order to allow it time to achieve a state of equilibrium is called 'spin-up'. The easiest way is to run the model long enough or multiple times in a loop with the same forcing until the different state variables show no significant change with respect to the previous run.

Probably the simplest way is to create a batch file that will run the model multiple times using the state-variables from the previous run as initial conditions for the following run. The first step is configure an initial run as explained in the previous example, using tables to initialize the vegetation parameters () and make sure the required state variables needed to initialize the model will be reported:

- Report\_Leaf\_Area\_Index = 1
- Report\_Veget\_frac = 1
- Report\_Stem\_Density = 1
- Report\_Stand\_Age = 1
- Report\_Root\_Mass = 1
- Report\_Tree\_Height = 1
- Report\_Basal\_Area = 1
- Report\_SWE = 1
- Report\_Soil\_Water\_Content = 1
- Report\_Soil\_Temperature = 1

The next step is tell the model that next time the model starts, the vegetation parameters will not be read from a table but that the parameters will be given as maps. For this set `Species_State_Variable_Input_Method = maps`.

When this variables is set to `maps`, the model expects to find a set of maps in the spatial information folder with the following names. There has to be one map per species:

- lai[n].map
- p[n].map
- ntr[n].map
- age[n].map
- root[n].map
- hgt[n].map
- bas[n].map

To initiate leaf area index, species proportion in each cell, tree density, age of stands, root mass, tree height and basal area, respectively. The value for inside the square brackets is the species id , where is the number of species being simulated.

To run the model in a loop we create a batch file that runs the model, takes the final state of the basin (as per the reported state variables), copies them with the right name in the spatial folder for initialization and runs the model again. Assuming you have set up the model as in the example of the next section, create a batch file named `spinup.bat` in the same folder where is located. Type the following contents into the file:

```
@echo off
set count = 1

:loop

echo Running iteration %COUNT%

start /w ech2o config.ini

ping -w 1000 1.1.1.1

echo finishing and copying files after iteration %COUNT%

copy /Y .\Results\root[0]0.365 .\Spatial\root[0].map
copy /Y .\Results\p[0]0000.365 .\Spatial\p[0].map
```

(continues on next page)

(continued from previous page)

```

copy /Y .\Results\nttr[0]00.365 .\Spatial\nttr[0].map
copy /Y .\Results\lai[0]00.365 .\Spatial\lai[0].map
copy /Y .\Results\hgt[0]00.365 .\Spatial\hgt[0].map
copy /Y .\Results\bas[0]00.365 .\Spatial\bas[0].map
copy /Y .\Results\age[0]00.365 .\Spatial\age[0].map

copy /Y .\Results\SWE00000.365 .\Spatial\SWE.map
copy /Y .\Results\SWC1_000.365 .\Spatial\Soil_moisture_1.map
copy /Y .\Results\SWC2_000.365 .\Spatial\Soil_moisture_2.map
copy /Y .\Results\SWC3_000.365 .\Spatial\Soil_moisture_3.map
copy /Y .\Results\Ts000000.365 .\Spatial\soiltemp.map
copy /Y .\Results\Q0000000.365 .\Spatial\streamflow.map

type .\Results\lai[0].tab >> .\Results\laiacum.txt
type .\Results\NPP[0].tab >> .\Results\NPPacum.txt
type .\Results\SoilMoistureAv.tab >> .\Results\SWCacum.txt

set /A COUNT=%COUNT%+1

goto loop

```

**Run the batch file by typing spinup.bat. This file will spinup the** model until you stop it pressing . Let the model spin for a period of 5 or 10 years.

If you are reporting time series of leaf area index, net primary production and soil moisture, the batch file will append the results in a file that contains the time series for the entire spinup period. Plotting this file in Excel will let us evaluate if the state variables are equilibrated at the end of the spinup period.

### 2.3 Configuration file keywords

Table 3: PATH DEFINITION SECTION

Keyword	Type	Units	Description
Maps_Folder	System Path	N/A	Path to folder with land surface information
Clim_Maps_Folder	System Path	N/A	Path to folder with Climate information
Output_Folder	System Path	N/A	Path to folder where outputs will be written

Table 4: **OPTIONS**

Keyword	Type	Units	Description
MapTypes	option	N/A	Format of maps. In this version the format is <i>csf</i> (PCRaster)
Species_State _Variable_Input_Method	option	N/A	Specifies the input format of the vegetation state variables. Options are <code>textit{table}</code> or <code>textit{maps}</code>
Reinfiltration	option	N/A	Switch 1/0 to turn reinfiltration on (1) or off (0)
Channel	option	N/A	Switch 1/0 to turn channel routing on (1) or off (0)
Vegetation_dynamics	option	N/A	Switch 1/0 to turn vegetation growth on (1) or off (0)
Aerodyn_resist_opt	option	N/A	Switches between different aerodynamic resistance formulations. 0: Penman; 1: Thom and Oliver (1977)
Soil_resistance_opt	option	N/A	Switches between different soil resistance formulations. 0: Passerat de Silans et al 1989; 1: Ivanov et al (2008)

Table 5: **TIME CONTROLS**

Keyword	Type	Units	Description
Simul_start	Integer	Seconds	Time of simulation start. In the current version this value must be 0
Simul_end	Integer	Seconds	Time when simulation ends in seconds. This value indicates the total simulated time
Simul_tstep	Integer	Seconds	Size of the integration time step
Clim_input_tstep	Integer	Seconds	Time step of climate forcing. Typically it is the same as <code>textit{Simul_tstep}</code> but can be larger (i.e. climate inputs are daily but we are using an hourly integration time step). <code>textit{Clim_input_tstep}</code> cannot be smaller than <code>textit{Simul_tstep}</code>
Report_interval	Integer	Seconds	Intervals between outputs. <code>textit{Report_interval}</code> cannot be smaller than <code>textit{Simul_tstep}</code> and typically it is equal to <code>textit{Simul_tstep}</code>

Table 6: Climate information (maps and binary files must be placed in ‘*Clim\_Maps\_Folder*‘)

Keyword	Type	Units	Description
Snow_rain_temp_threshold	scalar	$^{\circ}C$	Air temperature threshold for snow/rain transition
ClimateZones	Map file name	integer	Map identifying the climate zones
Isohyet_map	Map file name		This map allows to redistribute rainfall within a climate zone. It is a map with multiplication factors for rain in a given pixel. A map containing 1 over the domain has the effect of overriding this input (does not modify the precipitation input)
Precipitation	Binary cli- mate file	$ns^{-1}$	Precipitation input
AirTemperature	Binary cli- mate file	$^{\circ}C$	Average air temperature
MaxAirTemp	Binary cli- mate file	$^{\circ}C$	Maximum air temperature
MinAirTemp	Binary cli- mate file	$^{\circ}C$	Maximum air temperature
RelativeHumidity	Binary cli- mate file	$Pak$	Average air temperature
WindSpeed	Binary cli- mate file	$ns^{-1}$	Wind speed
IncomingLongWave	Binary cli- mate file	$Wm^{-2}$	Incoming long wave radiation
IncomingShortWave	Binary cli- mate file	$Wm^{-2}$	Incoming solar radiation

Table 7: **Drainage network** (Files must be located in Maps\_Folder)

Keyword	Type	Units	Description
local_drain_dirac	Map file name		D8 steepest descent ldd
channel_width	Map file name	m	mask with width of channel network. Pixels with no channel must be 0 or nodata. Positive numbers indicate the width of the channel in the pixel
channel_gw_transfer_param	Map file name 1}	m <sup>1</sup>	Coefficient controlling transfers of water from the subsurface system to the channel
mannings_n	Map file name 1/3}	m <sup>1/3</sup>	Manning's n roughness coefficient for channel

Table 8: *Initial conditions for soil states* (Files must be located in `Maps\Folder)

Keyword	Type	Units	Description
Streamflow	Map file name	m <sup>3</sup> s <sup>-1</sup>	Streamflow
snow_water_equivalent	Map file name	m	Snow water equivalent
Soil_moisture_1	Map file name	m <sup>3</sup> m <sup>-3</sup>	Volumetric soil water content for topmost soil layer
Soil_moisture_2	Map file name	m <sup>3</sup> m <sup>-3</sup>	Volumetric soil water content for layer 3
Soil_moisture_3	Map file name	m <sup>3</sup> m <sup>-3</sup>	Volumetric soil water content of bottommost layer
Soil_temperature	Map file name	°C	Soil temperature at boundary of thermal layer

Table 9: Soil parameters (Files must be located in Maps\_Folder)

Keyword	Type	Units	Description
DEM	Map file name	<i>m</i>	Digital elevation model
Slope	Map file name	<i>mm</i> <sup>-1</sup>	Local terrain slope. Rise over run
Horiz_Hydraulic_Conductivity	Map file name	<i>ms</i> <sup>-1</sup>	Effective soil hydraulic conductivity
Vert_Horz_Anis_ratio	Map file name	[-]	Ratio of vertical to horizontal hydraulic conductivity
Terrain_Random_Roughness	Map file name	<i>m</i>	Local surface roughness
Porosity	Map file name		Soil porosity
Air_entry_pressure	Map file name	<i>m</i>	Soil air entry pressure
Brooks_Corey_lambda	Map file name		Pore size distribution
Residual_soil_moisture	Map file name	<i>m</i> <sup>3</sup> / <i>m</i> <sup>3</sup>	Minimum allowed volumetric soil water content
Soil_depth	Map file name	<i>m</i>	Soil depth
Depth_soil_layer_1	Map file name	<i>m</i>	Depth of topmost soil layer
Depth_soil_layer_2	Map file name	<i>m</i>	Depth of second soil layer
Veget_water_use_param1	Map file name	<i>m</i>	Vegetation water use parameter as per Landsberg and Waring (1997)
Veget_water_use_param2	Map file name	<i>m</i>	Vegetation water use parameter as per Landsberg and Waring (1997)
Fraction_roots_soil_layer_1	Map file name		Fraction of roots in topmost soil layer. Sum of fraction of roots in soil layer 1 and 2 must be less than 1
Fraction_roots_soil_layer_2	Map file name		Fraction of roots in second soil layer. Sum of fraction of roots in soil layer 1 and 2 must be less than 1
Albedo	Map file name		Surface albedo
Surface_emissivity	Map file name		Surface emissivity/absorptivity



Table 10: **Forest parameters** (Files must be located in Maps\_Folder)

Keyword	Type	Units	Description
ForestPatches	Map file name	integer	Map identifying forest categories (patches)
Number_of_Species	Integer		Number of vegetation types included in the simulation
Species_Parameters	Parameter table		Table containing parameter information for each simulated vegetation type

Table 11: **Tables needed only if Species\_State\_Variable\_Input\_Method=tables**

Keyword	Type	Units	Description
Species_Proportion_Table	Variable table	$m^2$	Table with initial proportion of covered area (canopy cover) for each vegetation type with respect to cell area
Species_StemDensity_Table	Variable table	$trees/m^2$	Table with initial tree density per cell for each vegetation type
Species_LAI_Table	Variable table	$m^2$	Table with initial leaf area index for each vegetation type
Species_AGE_Table	Variable table	years	Table with initial average age each vegetation type
Species_BasalArea_Table	Variable table	$m^2$	Table with initial total basal area per vegetation type
Species_Height_table	Variable table		Table with initial effective height per vegetation type
Species_RootMass_table	Variable table	$g/m^3$	Table with initial root mass per volume of soil for each vegetation type

Table 12: **Map report switches**

Keyword	Type	Units	Description
Report_Long_Rad_Down	switch	$Wm^{-2}$	Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Ldown}
Report_Short_Rad_Down	switch	$Wm^{-2}$	Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Sdown}
Report_Precip	switch	$ms^{-1}$	Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Pp}
Report_Rel_Humidity	switch	$Pa/Pa$	Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{RH}

Continued on next page

Table 12 – continued from previous page

Keyword	Type	Units	Description
Report_Wind_Speed	switch	$m s^{-1}$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{WndSp}
Report_AvgAir_Temperature	switch	$^{\circ}C$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Tp}
Report_MinAir_Temperature	switch	$^{\circ}C$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{TpMin}
Report_MaxAir_Temperature	switch	$^{\circ}C$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{TpMax}
Report_SWE	switch	$m$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{SWE}
Report_Infilt_Cap	switch	$m s^{-1}$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{If}
Report_Infilt_Cap	switch	$m s^{-1}$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{If}
Report_Streamflow	switch	$m^3 s^{-1}$	1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Q}
Report_Soil_Water_Content_Average	switch	$m^3 m^{-3}$	1: Average volumetric water content for entire soil profile. Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Theta}
Report_Soil_Water_Content_L1	switch	$m^3 m^{-3}$	1: Volumetric water content for topmost soil layer. Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Theta}
Report_Soil_Water_Content_L2	switch	$m^3 m^{-3}$	1: Volumetric water content for second soil layer. Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Theta}
Report_Soil_Water_Content_L3	switch	$m^3 m^{-3}$	1: Volumetric water content for bottommost soil layer. Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Theta}
Report_Soil_Sat_Deficit	switch	$m$	Meters of water needed to saturate soil. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{SatDef}
Report_Soil_ETP	switch	$m s^{-1}$	Soil evaporation. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Evap}
Report_Soil_Net_RadReport	switch	$W m^{-2}$	Soil net radiation. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{NetR}
Report_Soil_LESoil	switch	$W m^{-2}$	Latent heat for surface layer. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{LE}
Report_Sens_Heat	switch	$W m^{-2}$	Sensible heat for surface layer. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{SensH}
Report_Grnd_Heat	switch	$W m^{-2}$	Ground heat. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{GrndH}
Report_Snow_Heat	switch	$W m^{-2}$	Turbulent heat exchange with snowpack. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{SnowH}
Report_Soil_Temperature	switch	$^{\circ}C$	Soil temperature at the bottom of first thermal layer. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Ts}

Continued on next page

Table 12 – continued from previous page

Keyword	Type	Units	Description
Report_Skin_Temperature	switch	$C$	Soil skin temperature. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Tskin}
Report_Veget_frac	switch	$m^2m^2$	Fraction of cell covered by canopy of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{p[textit{n}]}
Report_Stem_Density	switch	$stem^{-2}$	Density of individuals of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{ntr[textit{n}]}
Report_Leaf_Area_Index	switch	$m^2m^2$	Leaf area index of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{lai[textit{n}]}
Report_Stand_Age	switch	year	Age of stand of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{age[textit{n}]}
Report_Canopy_Conductance	switch	$ms^{-1}$	Canopy conductance for vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{gc[textit{n}]}
Report_GPP	switch	$Cm^2$	Gross primary production for vegetation type textit{n} during the time step. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{gpp[textit{n}]}
Report_NPP	switch	$C^{-1}$	Net primary production for vegetation type textit{n} during the time step. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{npp[textit{n}]}
Report_Basal_Area	switch	$m^2$	Total basal area of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{bas[textit{n}]}
Report_Tree_Height	switch	$m$	Height of stand of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{hgt[textit{n}]}
Report_Root_Mass	switch	$ym^{-3}$	Root mass per volume of soil vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{root[textit{n}]}
Report_Canopy_Temp	switch	$C$	Canopy temperature of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Tc[textit{n}]}
Report_Canopy_NetR	switch	$Wm^2$	Canopy temperature of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Tc[textit{n}]}
Report_Canopy_LE	switch	$Wm^2$	Latent heat for canopy layer of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{LEc[textit{n}]}
Report_Canopy_Sens_Heat	switch	$Wm^2$	Sensible heat for canopy layer of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Hc[end{appendix}textit{n}]}
Report_Canopy_Water_Stor	switch	$m$	Water storage in canopy layer of vegetation type textit{n}. 1: Report map for this variable - 0: Switch off reporting for this variable. Root name textsf{Cs[textit{n}]}

Continued on next page

Table 12 – continued from previous page

Keyword	Type	Units	Description
Report_Transpiration	switch	$m^3 s^{-1}$	Transpiration from vegetation type <code>textit{n}</code> . 1: Report map for this variable - 0: Switch off reporting for this variable. Root name <code>textsf{Trp[<code>textit{n}</code>]}</code>

Table 13: Time series report switches

Keyword	Type	Units	Description
Ts_OutletDischarge	switch	$m^3 s^{-1}$	1: Turns on reporting of discharge at cells with <code>textit{ldd}</code> value = 5 (outlets and sinks) - 0: Turns off report. File name <code>textsf{Discharge.tab}</code>
TS_mask	Map file name	nominal	Map identifying cells for which state variables will be reported. Map should be zero everywhere expect for target cells, which are identified with integer IDs ( <i>geq 1</i> ). A maximum of 32 cells can be reported.
Ts_Long_Rad_Down	Time series table	$W m^{-2}$	Time series of incoming long wave radiation to the surface layer for cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{Ldown.tab}</code>
Ts_Short_Rad_Down	Time series table	$W m^{-2}$	Time series of incoming short wave radiation to the surface layer for cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{Sdown.tab}</code>
Ts_Precip	Time series table	$ms^{-1}$	Time series of atmospheric long wave radiation for cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{Precip.tab}</code>
Ts_Rel_Humidity	Time series table	PaPa	Time series of relative humidity at the reference height for cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{RelHumid.tab}</code>
Ts_Wind_Speed	Time series table	$ms^{-1}$	Time series of wind speed at reference height for cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{WindSpeed.tab}</code>
Ts_AvgAir_Temperature	Time series table	$^{\circ}C$	Time series of average temperature at reference height for cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{AvgTemp.tab}</code>
Ts_MinAir_Temperature	Time series table	$^{\circ}C$	Time series of minimum temperature at reference height for cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{MinTemp.tab}</code>

Continued on next page

Table 13 – continued from previous page

Keyword	Type	Units	Description
Ts_MaxAir_Temperature	Time series table	$^{\circ}C$	Time series of maximum temperature at reference height for cells identified in textsf{TS_mask}. File name textsf{MaxTemp.tab}
Ts_SWE	Time series table	$m$	Time series of soil water equivalent at cells identified in textsf{TS_mask}. File name textsf{SWE.tab}
Ts_Infilt_Cap	Time series table	$m s^{-1}$	Time series of infiltration capacity at cells identified in textsf{TS_mask}. File name textsf{InfiltCap.tab}
Ts_Streamflow	Time series table	$m^3 s^{-1}$	Time series of streamflow at cells identified in textsf{TS_mask}. File name textsf{Streamflow.tab}
Ts_Soil_Water_Content_Average	Time series table	$m^3 m^{-3}$	Times series of Average volumetric water content for entire soil profile at cells identified in textsf{TS_mask}. File name textsf{Theta.tab}
Ts_Soil_Water_Content_L1	Time series table	$m^3 m^{-3}$	Times series of Average volumetric water content for topsoil at cells identified in textsf{TS_mask}. File name textsf{ThetaL1.tab}
Ts_Soil_Water_Content_L2	Time series table	$m^3 m^{-3}$	Times series of Average volumetric water content for second soil layer at cells identified in textsf{TS_mask}. File name textsf{ThetaL2.tab}
Ts_Soil_Water_Content_L3	Time series table	$m^3 m^{-3}$	Times series of Average volumetric water content for bottom-most soil layer at cells identified in textsf{TS_mask}. File name textsf{ThetaL3.tab}
Ts_Soil_Sat_Deficit	Time series table	$m$	Time series of soil water deficit defined as the water depth needed to saturate cells identified in textsf{TS_mask}. File name textsf{SoilSatDef.tab}
Ts_Soil_ETP	Time series table	$m s^{-1}$	Time series of soil evaporation at cells identified in textsf{TS_mask}. File name textsf{Evap.tab}

Continued on next page

Table 13 – continued from previous page

Keyword	Type	Units	Description
Ts_Soil_Net_Rad	Time series table	$Wm^{-2}$	Time series of net radiation for surface layer at cells identified in textsf{TS_mask}. File name textsf{NetRad.tab}
Ts_Soil_LE	Time series table	$Wm^{-2}$	Time series of latent heat for surface layer at cells identified in textsf{TS_mask}. File name textsf{LatHeat.tab}
Ts_Sens_Heat	Time series table	$Wm^{-2}$	Time series of sensible heat for surface layer at cells identified in textsf{TS_mask}. File name textsf{SensHeat.tab}
Ts_Grnd_Heat	Time series table	$Wm^{-2}$	Time series of ground heat at cells identified in textsf{TS_mask}. File name textsf{GrndHeat.tab}
Ts_Snow_Heat	Time series table	$Wm^{-2}$	Time series of heat exchanges with snowpack at cells identified in textsf{TS_mask}. File name textsf{SnowHeat.tab}
Ts_Soil_Temperature	Time series table	$^{\circ}C$	Time series of soil temperature at cells identified in textsf{TS_mask}. File name textsf{SoilTemp.tab}
Ts_Skin_Temperature	Time series table	$^{\circ}C$	Time series of soil skin temperature at cells identified in textsf{TS_mask}. File name textsf{SkinTemp.tab}
Ts_Veget_frac	Time series table	$m^2m^{-2}$	Time series of fractions occupied by vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{p[textit{n}].tab}
Ts_Stem_Density	Time series table	$stem^{-2}$	Time series of stem density of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{num_of_trees[textit{n}].tab}
Ts_Leaf_Area_Index	Time series table	$m^2m^{-2}$	Time series of leaf area index of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{lai[textit{n}].tab}

Continued on next page

Table 13 – continued from previous page

Keyword	Type	Units	Description
Ts_Canopy_Conductance	Time series table	$ms^{-1}$	Time series of canopy conductance of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{CanopyConduct[textit{n}].tab}
Ts_GPP	Time series table	$gCm^{-2}$	Time series of gross primary production of vegetation type textit{n} during the current time step at cells identified in textsf{TS_mask}. File name textsf{GPP[textit{n}].tab}
Ts_NPP	Time series table	$gCm^{-2}$	Time series of net primary production of vegetation type textit{n} during the current time step at cells identified in textsf{TS_mask}. File name textsf{NPP[textit{n}].tab}
Ts_Basal_Area	Time series table	$m^{-2}$	Time series of total basal area of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{BasalArea[textit{n}].tab}
Ts_Tree_Height	Time series table	$m$	Time series of effective tree height of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{TreeHeight[textit{n}].tab}
Ts_Root_Mass	Time series table	$gm^{-3}$	Time series of root density (mass per volume of soil) of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{RootMass[textit{n}].tab}
Ts_Canopy_Temp	Time series table	$^{\circ}C$	Time series of canopy temperature of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{CanopyTemp[textit{n}].tab}
Ts_Canopy_NetR	Time series table	$Wm^{-2}$	Time series of net radiation at canopy layer of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{CanopyNetRad[textit{n}].tab}
Ts_Canopy_LE	Time series table	$Wm^{-2}$	Time series of latent heat at canopy layer of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{CanopyLatHeat[textit{n}].tab}
Ts_Canopy_Sens_Heat	Time series table	$Wm^{-2}$	Time series of sensible heat at canopy layer of vegetation type textit{n} at cells identified in textsf{TS_mask}. File name textsf{CanopySensHeat[textit{n}].tab}

Continued on next page

Table 13 – continued from previous page

Keyword	Type	Units	Description
Ts_Canopy_Water_Stor	Time series table	$m$	Time series of water storage at canopy layer of vegetation type <code>textit{n}</code> at cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{CanopyWaterStor[<code>textit{n}</code>].tab}</code>
Ts_Transpiration	Time series table	$ms^{-1}$	Time series of transpiration for canopy layer of vegetation type <code>textit{n}</code> at cells identified in <code>textsf{TS_mask}</code> . File name <code>textsf{Transpiration[<code>textit{n}</code>].tab}</code>



## CHAPTER 3

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`