

Content of ORCHIDEE forcing files

Jan Polcher

May 2014

This document describes the NetCDF file and its content so that it can be used to force ORCHIDEE. It lays the basis for the code to be used in the driver which reads this information, interpolates it to the time step desired by the user and hands it down to ORCHIDEE. For the moment these specifications are implemented in the module “forcing_tools.f90” in the driver which feeds OASIS (driver2oasis.f90).

We deal here with forcing at the sub-diurnal frequency which is better than 6 hourly. For a forcing with a lower frequency other information might be needed in the NetCDF file and more importantly the simple interpolations used today might not be enough any more. It is left to future discussions if daily forcing files should be covered by this document or not and if it should be implemented in another module.

At the end of the document, the impact of an incorrect specification of time intervals of the forcing data on an ORCHIDEE simulation are documented. This is illustrated with the WFDEI forcing which poses the problem of scalar variables lagging in time the fluxes.

General considerations

This section provides some general consideration the files used to force ORCHIDEE should follow. We have chosen to put all variables into one file and preferably use one file per year. These files should follow the following conventions :

- The NetCDF file has to comply with the CF convention (<http://www.cgd.ucar.edu/cms/eaton/netcdf/CF-20010629.htm>) for all the meta data provided.
- The variables needed and their names follow the ALMA convention (http://www.lmd.jussieu.fr/~polcher/ALMA/convention_input_3.html).
- List of variables :

<u>Variable</u>	<u>Description</u>	<u>Definition</u>	<u>Units</u>	<u>Sign</u> (direction of positive values)
Wind_N	Near surface northward wind component	Northward wind measured at reference levels near the surface (3D variable).	m/s	Northward
Wind_E	Near surface eastward wind component	Eastward wind measured at reference levels near the surface (3D variable).	m/s	Eastward
	A simplification is possible if only the module of the wind is available.			
Rainf	Rainfall rate	Average total rainfall over a time step of the	kg/m ² s	downward

		forcing data.		
	Sub-grid rainfall information			
Snowf	Snowfall rate	Average total snowfall over a time step of forcing data.	kg/m ² s	downward
	Sub-grid snowfall information			
Tair	Near surface air temperature	Temperature measured at reference levels near the surface (3D variable)	K	
Qair	Near surface specific humidity	Specific humidity measured at reference levels near the surface (3D variable)	kg/kg	
PSurf	Surface pressure	Pressure measured at the surface	Pa	
SWdown	Surface incident shortwave radiation	Incident radiation in the shortwave part of the spectrum averaged over the time step of the forcing data	W/m ²	downward
LWdown	Surface incident longwave radiation	Incident longwave radiation averaged over the time step of the forcing data	W/m ²	downward
CO2air	Near surface CO ₂ concentration	The partial pressure of CO ₂ concentration at the atmospheric reference level (3D variable).	ppmv	

- As the temporal interpolations might need an extra forcing time step before or after the first and last date to be simulated by ORCHIDEE, it is convenient to be able to provide a list of forcing files. This means that the forcing files can contain any period of time and the user has to specify to the driver a list of file names which covers a time period longer than the one to be simulated. This avoids having to add extra time steps in the files or repeat a forcing time step in one or more files.

Description of geographical domain

The geographical domain of the forcing data has to be provided on a standard latitude and longitude coordinate system. It can be any spatial domain ranging from a single point up to the entire globe. It is for the moment a limitation of ORCHIDEE (because of the interpolation of the surface description files) that only regular lat/lon grids can be provided. In principle the forcing data can be provided on

any grid on the sphere, as long as it is time invariant.

- The latitude and longitude information of the grid have to be specified according to the CF convention (<http://www.cgd.ucar.edu/cms/eaton/netcdf/CF-20010629.htm#ctype>). The bounds informations for the cells of the horizontal grid are not needed for the moment. As soon as irregular or unstructured grids will be supported by ORCHIDEE this will have to change.
- In order to reduce the disk space used by the forcing data and as ORCHIDEE only runs over land points, the horizontal grid can be gathered. This means that only the forcing data for land points are stored in the NetCDF file together with an indexing table which allows to scatter the data on the full grid while reading it. This is supported by the CF convention : <http://www.cgd.ucar.edu/cms/eaton/netcdf/CF-20010629.htm#gath>

Vertical axis description

Depending on the origin of the forcing data, the vertical coordinates of some atmospheric variables might have to be specified in different ways:

- Sigma levels for data from certain atmospheric models
- Hybrid sigma levels also from atmospheric models
- Time varying heights in meters can also be obtained from certain atmospheric models.
- Time invariants heights in meters mostly from in-situ observations.

This information matters for the following variables : Wind_E, Wind_N, Tair, Qair and CO2air. In order to determine the appropriate height information the following variables in the forcing file will be examined. They are used to compute for ORCHIDEE the height in meters above ground of air temperature, humidity and CO₂ concentration (zlev_vec) and wind (zlevuv_vec).

Sigma and Sigma_uv

Expected is one real value which allows to compute the height of the first level with the following formula :

$$zlev_vec(i,j) = rau(i,j) * cte_grav * (psurf(i,j) - Sigma * psurf(i,j))$$

$$zlevuv_vec() = rau(i,j) * cte_grav * (psurf(i,j) - Sigma_uv * psurf(i,j))$$

Should only Sigma exist, Sigma=Sigma_uv will be assumed.

HybSigA, HybSigB, HybSigA_uv and HybSigB_uv

Expected is one real value for each variable. They allow to compute the height of the first level with the following formula :

$$zlev_vec(i,j) = rau(i,j) * cte_grav * (psurf(i,j) - (HybSigA + HybSigB * psurf(i,j)))$$

$$zlevuv_vec(i,j) = rau(i,j) * cte_grav * (psurf(i,j) - (HybSigA_uv + HybSigB_uv * psurf(i,j)))$$

If the UV values do not exist we assume that they are the same as for T and Q.

Sample CDF file is provided :

https://forge.ipsl.jussieu.fr/orchidee/attachment/wiki/Branches/Driver_Atmos_Lev/levels_Hybrid.cdf

Levels and Levels_uv

This is the case when the levels are provided as 2D time evolving fields. The height is supposed to be in meters.

The following assignation will be done for the values read at each forcing time step :

$$\text{zlev_vec}(i,j) = \text{Levles}(i,j)$$

If levels_uv is present :

$$\text{zlevuv_vec}(i,j) = \text{levels_uv}(i,j)$$

Else :

$$\text{zlevuv_vec}(i,j) = \text{levels}(i,j)$$

Height_Lev1 and Height_Levuv

This comes back to the same case as when these variables are provided through the run.def file.

The following assignations will be done :

$$\text{zlev_vec}(i,j) = \text{Height_Lev1}$$

If Height_Levuv is present :

$$\text{zlevuv_vec}(i,j) = \text{Height_Levuv}$$

Else :

$$\text{zlevuv_vec}(i,j) = \text{Height_Lev1}$$

Sample CDF file is provided :

https://forge.ipsl.jussieu.fr/orchidee/attachment/wiki/Branches/Driver_At看_Lev/levels_Height.cdf

Time axis

Describing the time axis in the forcing files used by ORCHIDEE has been a headache since the first off-line driver for the model because depending on the source of the data some variables might be instantaneous values, others averages and to complicate further the matter they are not always centred on the time interval of validity. The interpretation made by the driver of the timing of each variable will affect directly the diurnal cycle of the atmospheric data provided to ORCHIDEE.

In order for the driver to be able to recognize this and act accordingly some meta information on the time axis will be read.

- Any variable with the units “seconds since” will be read as a time axis and used by the driver. The axis names 'time' will be used as a reference and a default for further operations (**not yet fully implemented in forcing_tools.f90**).
- For each of the time variables read, the bounds will be computed using the hypothesis that we have a constant time step. The time step length will be computed as the average over the entire length of the axis.
- Three types of bounds will be computed using the following hypothesis :

1. The date provided by the axis is the start of the validity interval
 2. The date is the centre of the time interval
 3. The date is the end of time interval over which the variable is valid.
- Based on the CF method (<http://www.cgd.ucar.edu/cms/eaton/netcdf/CF-20010629.htm#subg>), for each forcing variable read we will look for the attribute “cell_methods” in order to determine which time axis needs to be used. The cell method is expected to contain the following information : “name: method(position). This information will be interpreted as follows : “name” is the name of the variable to be used as a time axis, “method” will determine if the variable is an instantaneous value or an average, finally “position” will determine if the date associated to each time step is either the start, the centre or the end of the time interval over which it is valid. For instance : "time: mean(end)" means that the variable “time” will be used as time axis and interpreted as the date of the end of the interval over which values were averaged in order to obtain the data to be read from the forcing file.
 - If no cell_method attribute is found for a forcing variable, then it is assumed to use the “time” variable as an axis and it is interpreted as instantaneous values centred on the dates provided.

To illustrate here are some metadata for 2 typical values in a forcing file : Air temperature as instantaneous/centred values and Downward solar radiation as averages provided at the end of the interval.

```

float Tair(time, land) ;
    Tair:long_name = "Near surface air temperature at 2 m at time stamp" ;
    Tair:units = "K" ;
    Tair:title = "Tair" ;
    Tair:actual_max = 322.7556f ;
    Tair:actual_min = 205.3198f ;
    Tair:Fill_value = 1.e+20f ;
    Tair:missing_value = 1.e+20f ;
    Tair:_FillValue = 1.e+20f ;
    Tair:cell_methods = "time: instantaneous" ;

float SWdown(time, land) ;
    SWdown:long_name = "Mean surface incident shortwave \n",
                        "radiation over the previous 3 hours" ;
    SWdown:units = "W/m2" ;
    SWdown:title = "SWdown" ;
    SWdown:actual_max = 1240.201f ;
    SWdown:actual_min = 0.f ;
    SWdown:Fill_value = 1.e+20f ;
    SWdown:missing_value = 1.e+20f ;
    SWdown:_FillValue = 1.e+20f ;
    SWdown:cell_methods = "time: mean(end)" ;

```

Impact of time axis specification on ORCHIDEE

In order to test the importance of the time stepping information in the forcing file, we have used two driver for one and the same version of ORCHIDEE. They both have different hypothesis on the meaning of the forcing fields on the given time axis :

- `dim2driver.f90` : It assumes that all fluxes are averaged over the time step from t to $t+1$ and scalar variables are instantaneous at time t . Thus none of the meta information on the time axis present in the forcing file are used.
- `driver2oasis.f90` : uses all the meta-information for the forcing variables available in the file as described above (`cell_methods` for instance) and can use thus different positions on the time axis for all variables.

It has to be noted that the time interpolation of the downward short-wave in both drivers uses also the zenith angle following the method proposed for GSWP-1. Thus any time shift in the the downward

solar radiation can in part be hidden by the time interpolation.

The forcing file used for this test is WFDEI, which like many corrected versions of re-analysis has a time shift between fluxes and scalar variables. The time step of the forcing is 3 hourly. In the present case we have the following validity intervals :

- fluxes : SWdown, LWdown, Rainf and Snowf available at time t correspond to averages over the time t-1 and t.
- scalar : Tair, Qair, CO2air, Psurf, Wind_N and Wind_E are instantaneous at time t.

The WFDEI file start at 0:00UTZ on 1st of January ($t=1$). This means that the fluxes cover the period 21:00 to 24:00 of the 31st of December of the previous year. As this was noted early on as a problem with dim2driver.f90, a version of the WFDEI forcing was produced where the first time step has been deleted (WFDEI_ben forcing) but the time axis kept as is. Thus for WFDEI_ben the first value of the fluxes correspond to the interval 0:00 to 3:00 UTC of the 1st of January and the scalars are considered to be instantaneous at 1:30UTC.

In the following analysis, we will look at one diurnal cycle (21 June 2002) to see how well the forcing variables are interpreted. Over a longer period we will examine the impact it has on the energy balance of a river basin.

The diurnal cycle

We will examine here three forcing variables over the point 49.25 North and 0.25 East on the 21st June 2002. This ensure that the downward solar radiation should peaks at noon exactly.

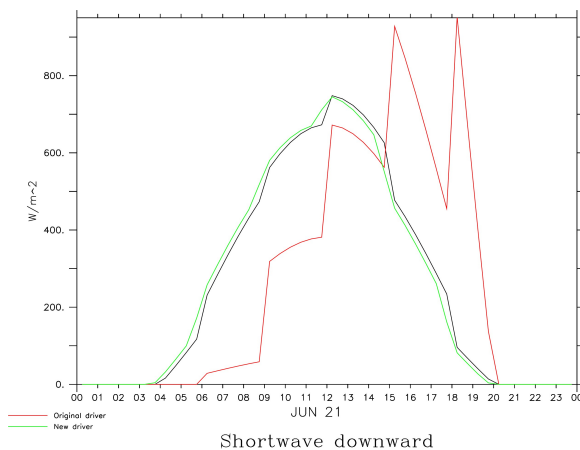


Figure 1: Downward short wave radiation. Black line is the WFDEI_ben forcing and dim2driver.f90, the red line the original WFDEI forcing and the dim2driver.f90 and finally the green line is the driver2oasis.f90 driver.

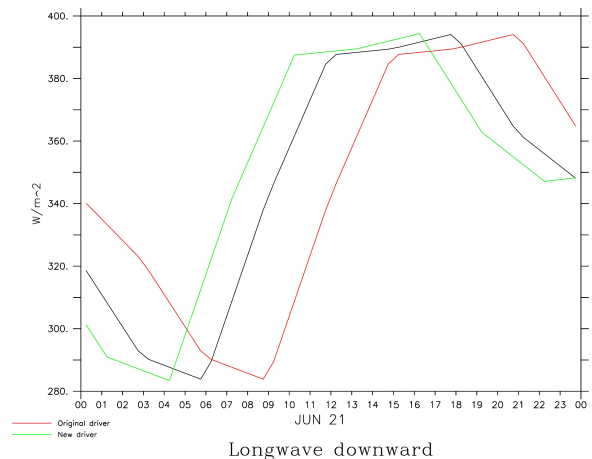


Figure 2: As figure 1 for downward long wave radiation

Figure 1 shows that when the diurnal cycle of the downward solar radiation for WFDEI in dim2driver.f90 is inconsistent with the zenith angle. The temporal interpolation to the 30 minutes time step produces numerical errors and thus very noisy fluxes. Deleting the first values in the WFDEI forcing (WFDEI_ben) solves this issues as now indeed the first time step corresponds to the

assumption of the dim2driver.f90 code. The driver2oasis.f90 interpolates correctly the short wave forcing producing a diurnal cycle which peaks at noon as expected.

Figure 2 demonstrates with the long wave flux that WFDEI in dim2driver produces a 4.5 hours shift (compared to the driver2oasis.f90 usage of the forcing data) resulting from the first 3 hourly step and the assumption that the values are valid in the middle of the interval. The WFDEI_ben forcing in which the first time_step was deleted reduces the problem by 3 hours as expected.

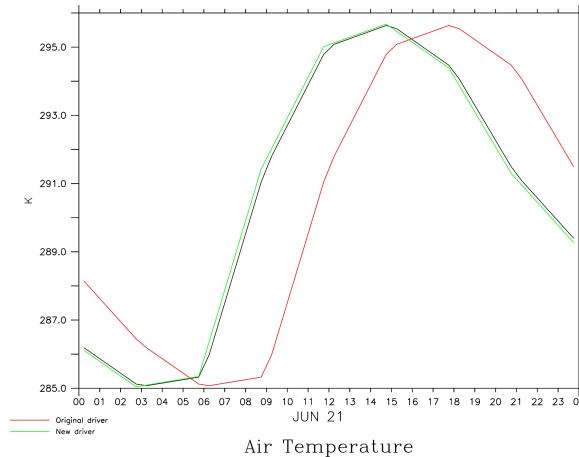


Figure 3: As figure 1 for air temperature

Figure 3 shows that dim2driver.f90 uses correctly the scalar variables for the WFDEI_ben while the original forcing data produces a 3 hourly shift. It is unclear why the scalar variables show a correct diurnal cycle in the WATCH_ben forcing.

The basin average energy balance

For the comparison of long simulations (from 1982 to 2012) we have not compared with the WFDEI_ben forcing as in this one the rainfall of the ERA-Int re-analysis was corrected with GPCC instead of CRU. The Nile and Danub basins will be analysed in order to sample a wet and an arid region.

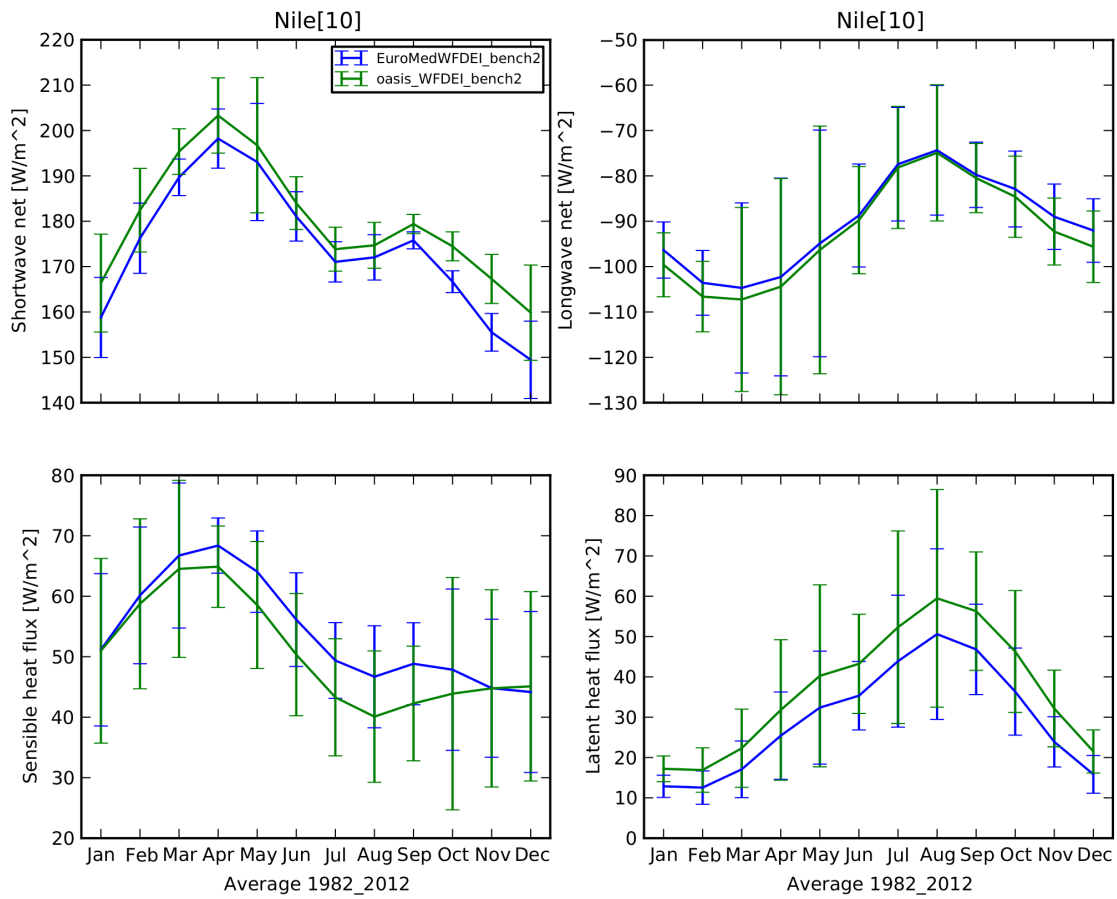


Figure 4: The four main energy fluxes averaged over the Nile basin

Over both regions the better description of the diurnal cycle of the forcing file results in a higher evaporation. In other words it affect the average Bowen ration in favour of the latent heat flux.

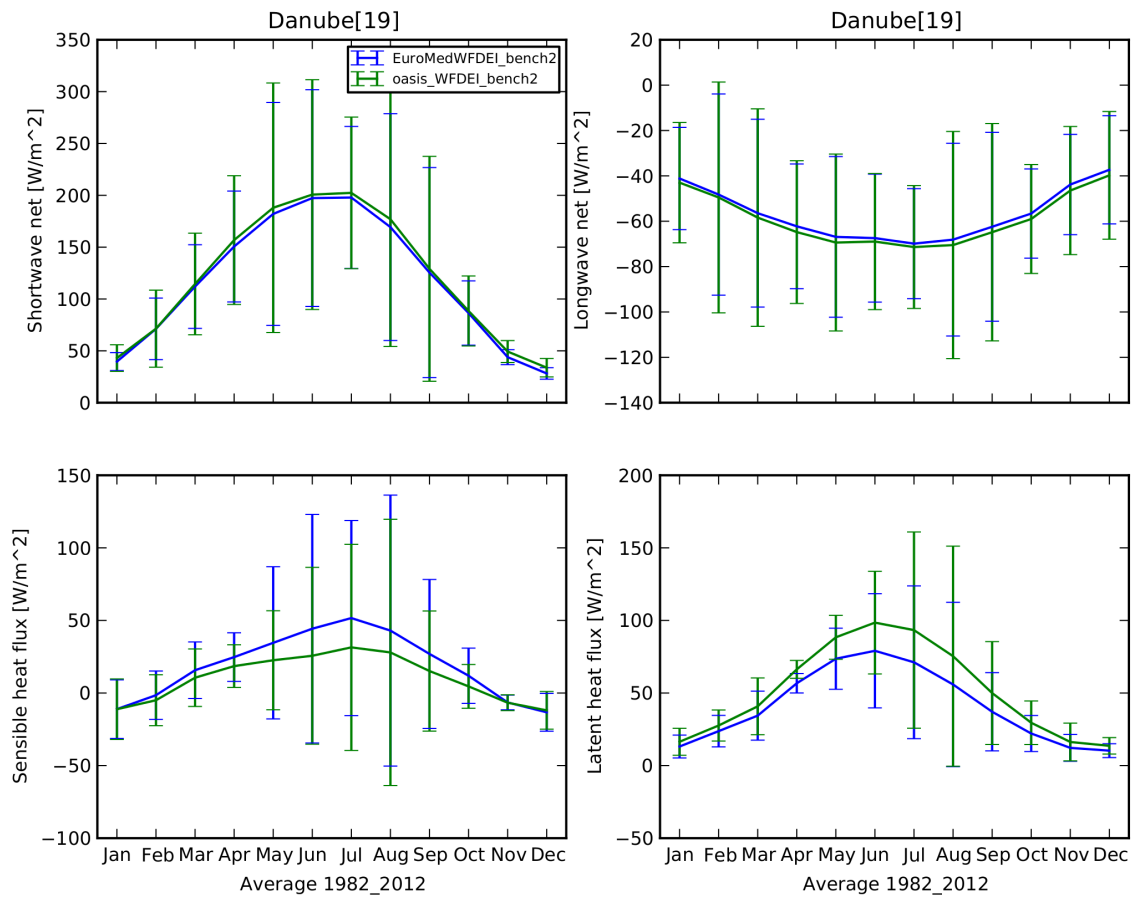


Figure 5: As figure 4 but for the Danube basin.