**3 CABLE: offline**

In this section we discuss building and running CABLE offline for single-site and regional/global applications.

We recommend the use of CABLE on UNIX/Linux platforms only. Building CABLE basically requires a Fortran compiler and a netcdf distribution. We recommend that you use Intel Fortran (at least version 10.1.20). On *vayu* we use version 11.1.074. The earliest version of netcdf we have used successfully is version 3.6.3. In addition, use of the provided build script will require that a ksh interpreter is installed. This will exist by default on all POSIX machines, unless you have a highly irregular setup.

In principle, CABLE is not restricted to UNIX/Linux platforms. However, whilst we have built and run CABLE on other platforms, it is generally not a straightforward endeavour. (The [CABLE sharepoint site](https://teams.csiro.au/sites/cable/default.aspx) has some notes about compiling CABLE1.4b on Windows machines.) We should note however that we have only used open source, free compilers on other platforms. If you have a Fortran compiler, and Fortran netcdf libraries, then CABLE should work, provided you can interpret the Makefiles and shell scripts. If you use CABLE with a different compiler, libraries or platform, then expect to check the impact that different compiler options might have on the model.

For single-site investigations, CABLE can be done within seconds using one single processor, the whole process including spinup would be finished within a minute usually (depending on the convergence threshold). Thus, the serial version will suffice. For global (or regional) offline runs, CABLE can still be run in serial mode (about 15 minutes/year for GSWP global run at 1x1 degree resolution), but would benefit from running on multiple processors to speed up the simulations (about 1 minute/year for GSWP global run).

Using the Open Message Passing Interface library (mpi), a wrapper has been developed so that it integrates with existing CABLE code. This mpi wrapper uses all the core code and serial offline code without modification, with the exception of the file *cable\_driver.F90* which is replaced by *cable\_mpimaster.F90* and *cable\_mpiworker.F90*. The control file *cable\_mpidrv.F90* calls the appropriate driver files in various CPUs and another file *cable\_mpicommon.F90* contains supporting routines for this mpi wrapper. Such a wrapper enables easy upgrading of the parallel CABLE code with future versions of CABLE. In case of model development requiring changes in *cable\_driver.F90* and *cable\_define\_types.F90*, changes may be required for the mpi codes as well; otherwise, there is no need to modify the wrapper.

(The addition to the User Guide based on the following code structure:

a) insert the mpi codes (plus Makefile\_mpi, build\_mpi.ksh) into the existing offline directory.

b) create a subdirectory CABLE-2.0a/offline/script\_global/ to hold the run scripts for the global runs, both serial and parallel versions:

serial\_gswp\_vayu.bash, mpi\_gswp\_vayu.bash, serial\_gswp\_burnet.bash, mpi\_gswp\_burnet.bash, serial\_gpcc\_vayu.bash, mpi\_ gpcc \_vayu.bash, serial\_ gpcc \_burnet.bash, mpi\_ gpcc \_burnet.bash.)

**3.1 Building CABLE**

3.1.1 *build.ksh* and *Makefile\_offline* for serial version of CABLE

……

3.1.2 *build\_mpi.ksh* and *Makefile\_mpi* for parallel version of CABLE:

Within the offline directory, there is the *Makefile\_mpi* and its corresponding script *build\_mpi.ksh*, both modified from their serial counterparts.

The build process here is similar to that for a serial run. Again, it is assumed that the *netcdf* module is pre-loaded. For parallel programming, the *openmpi* module also needs to be loaded. Here, the hidden directory used is called *.mpitmp* to distinguish from the serial run as the compiler used is now *mpif90* instead of *ifort*. After compilation, the executable named *cable-mpi\** is moved to the offline directory.

**3.2 Running CABLE**

……..

Finally, we describe the procedure for offline, global runs in serial mode (Sec. 3.2.3) and in parallel mode (Sec. 3.2.4).

**3.2.3 Global grid serial run**

Offline CABLE can be run for a single site or multiple points up to the whole globe. This is determined by reading the number of points in the met forcing file (Sec 5.1.5). Normally, single-site input files comprise a few years of all required met forcing data. As the met forcing file size gets larger with more points, there is an option to separate the met forcing by variable type and year. This is activated by setting the *cable.nml* variable *ncciy* to a non-zero year. The *cable.nml* variables gswpfile%variable\_name then become active. See the example *cable.nml* provided in CABLE-AUX/offline.

Global offline CABLE simulations have been run using met forcing downloaded from the GSWP-2 project (<http://www.iges.org/gswp/>). These met files have not been provided in the CABLE release, but anyone interested in such runs should contact [Bernard.Pak@csiro.au](mailto:Bernard.Pak@csiro.au) for help. The PALS website intends to support global offline simulations in future.

As the global (or regional) offline runs require huge storage to cater for the input meteorological forcing and the output files, it is recommended to run on the WORKDIR disk area instead of the HOME directory. (This is true for both the serial and mpi version.)

If you are using vayu.nci.org.au, ***your\_run\_directory*** would be */short/project\_name/user\_name/CABLE\_run\_dir/* with the respective names filled in. At the start, copy the executable and the run script to this directory. Unlike the single-site run, you would also need to copy the global met forcing files and other initialization files to some subdirectories here. Please note that because CABLE reads in the met forcing at every time step, leaving the met files on the HOME disk and linked to the WORKDIR disk will slow down the simulations. Finally, a copy of cable.nml is required; you need to check that the global offline switch is switched on and correct directories are listed for various input files. Examples of such *namelist* file can be found in *CABLE-AUX/offline/namelist\_dir/*.

To make the global offline serial run, copy the executable *cable\** to ***your\_run\_directory***. Submit a job using the script like the example serial\_gswp\_vayu.bash provided in ^branches/Users/NCIlogin/CABLE‑2.0a/offline/script\_global/.

The command for running it is *qsub serial\_gswp\_vayu.bash*, which is issued in ***your\_run\_directory*** with subdirectories ***namelistDir***, ***out\_gswp***, ***surface\_data*** and ***gswp***. The ***namelistDir*** subdirectory would be a copy of or linked to *CABLE-AUX/offline/namelist\_dir/*, which has the appropriate namelist files for each of the 10 years available. The ***out\_gswp*** subdirectory is created to hold the output files. The ***surface\_data*** subdirectory would hold copies of input files from *CABLE-AUX/ offline/*, *CABLE-AUX/core/biogeophys/* and *CABLE-AUX/core/biogeochem/* for model initialization. The ***gswp*** subdirectory would hold the 10 years of meteorological forcing from the GSWP2 experiment (you may have to download them directly from the GSWP2 site). Please note that the subdirectories ***out\_gswp***, ***surface\_data*** and ***gswp*** are mentioned in the example namelist files and you have to modify them according to your own directory set up and naming.

The time to simulate one model year on *Vayu* is about 15 minutes.

**3.2.4 Executing *cable-mpi\****

Subsequent to successfully building the parallel version of CABLE, an executable *cable-mpi\** is produced. After copying the executable to ***your\_run\_directory***, it is possible to run it with a script like mpi\_gswp\_vayu.bash provided in ^branches/Users/NCIlogin/CABLE‑2.0a/offline/script\_global/.

The command for running it is *qsub mpi\_gswp\_vayu.bash*, which is issued in ***your\_run\_directory*** with subdirectories ***namelistDir***, ***out\_gswp***, ***surface\_data*** and ***gswp***. The ***namelistDir*** subdirectory would be a copy of or linked to *CABLE-AUX/offline/namelist\_dir/*, which has the appropriate namelist files for each of the 10 years available. The ***out\_gswp*** subdirectory is created to hold the output files. The ***surface\_data*** subdirectory would hold copies of input files from *CABLE-AUX/ offline/*, *CABLE-AUX/core/biogeophys/* and *CABLE-AUX/core/biogeochem/* for model initialization. The ***gswp*** subdirectory would hold the 10 years of meteorological forcing from the GSWP2 experiment (you may have to download them directly from the GSWP2 site). Please note that the subdirectories ***out\_gswp***, ***surface\_data*** and ***gswp*** are mentioned in the example namelist files and you have to modify them according to your own directory set up and naming.

The time to simulate one model year on *Vayu* is about 2.2 minutes using 8 CPUs, 1.3 minutes using 16 CPUs and 1.2 minutes using 24 CPUs. Further code optimization is underway to improve the performance in using more processors.

**6.1 Fortran files in CABLE subdirectories**

Table 4 (addition):

cable\_mpicommon.F90 Supporting routines for the mpi wrapper

cable\_mpidrv.F90 Driver for the mpi wrapper of CABLE

cable\_mpimaster.F90 Driver for CABLE on the master CPU

cable\_mpiworker.F90 Driver for CABLE on the worker CPU