# User Guide for Agro-IBIS Branch adamward-THMB

## Overview

This user guide contains specifications regarding an Agro-IBIS branch named “adamward-THMB” and other information that could be helpful for new Agro-IBIS user.

The branch adamward---THMB started from the trunk version of Agro---IBIS, featuring the following new functionalities:

- 2-D user-directive input of plant functional type (pft) in a yearly basis from NetCDF files.
- 2-D user-directive input of fertilizer application in a yearly basis from NetCDF files.
- 2-D user-directive input of soil with layered data from a NetCDF file (up to 11 layers).
- Hourly output (merged from branch <jpatton_hourly>).

Currently tested revision: Rev 335

## 1. Basic svn usage

This code uses svn for code management (download, update and distribution). To check the installation and version of svn in your machine, try <svn --version>.

1.1 Create a new branch from the trunk

```
svn copy -m 'Creating a new branch' https://trac.agron.iastate.edu/prepos/agro-ibis/trunk
https://trac.agron.iastate.edu/prepos/agro-ibis/branches/adamward-THMB
```

<svn copy>

purpose: Copy a file or directory while preserving history

usage: copy SRC[@REV]... DST (SRC: source; DST: destination)

<-m>

Option: Specify log message

Note: for more information, try <svn help copy> in the terminal.

Contact: Yuwei Li
Email: yuwei.lee03@gmail.com

Last updated: 2015-06-26

This copies the trunk version from the server to the specified directory in the server as a branch version. The specified directory will be created if not exist. The message from <-m> will be shown as the log message.

1.2 Download a version to your local working directory

```
svn checkout https://trac.agron.iastate.edu/prepos/agro-ibis/branches/adamward-THMB
```

```
<svn checkout>
purpose: Check out a working copy from a repository
usage: checkout URL[@REV]... [PATH]
```

This checks out a working copy of this branch from the server to the local working directory. [PATH] is omitted here, and therefore the copy will be checked out to the current directory of the terminal, with the name as the base name of the URL (in this case, adamward-THMB).

1.3 Merge with a branch

```
svn merge -r 217:330 vanloocke_energycrops adamward-THMB/
```

```
<svn merge>
purpose: Apply the differences between two sources to a working copy path
usage: merge [-c M[,N...] | -r N:M ...] SOURCE[@REV] [WCPATH]
```

where SOURCE can be either a URL or a working copy path, WCPATH is the working copy path that will receive the changes.

SOURCE (in revision REV) is compared as it existed between revisions N and M for each revision range provided. '-c M' is equivalent to '-r <M-1>:M'.

Note that, with the option [-r N:M ...], the revision range for merging is from rev <N+1> to rev <M>, i.e., rev <N> is not included.

In the example above, a local copy of the branch < vanloocke_energycrops> is going to be merged to the working copy of the branch < adamward-THMB>, with the revision range from rev 218 to rev 330.

1.4 Commit a version

```
svn commit -m “message for the commit”
```

This command commits your working copy to the svn server, which you already set up during the creation of this working copy using svn.

1.5 Add new file to the list of commit to the repository

```
svn add FILE
```

This command adds files, directories, or symbolic links in your working copy for addition to the repository. They will be uploaded and added to the repository on your next commit.

1.6 See what files a repository has without downloading a working copy

```
svn list
```

## 2. Branch usage

2.1 Modeling flags (ibis.infile)

| | | |
|---|---|---|
| isoil_layer: | 0: no (default)<br>1: yes | read-in soils with layered data from NetCDF file < *soil_layers.nc* >. |
| iyr_pft: | 0: no (default)<br>1: yes | read-in pft on a year-by-year basis from NetCDF files < *pft_<iyear>.nc* >. |
| iyr_fert: | 0: no (default)<br>1: yes | read-in fertilizer application on a year-by-year basis from NetCDF files < *fert_<iyear>.nc* >. |
| ihourlyout | 0: no (default)<br>1: yes | output hourly results to directory < *output/hourly* >.<br><br>(note: the code assumes existence of this directory, if error occurs, try run the script < *initial_setup* > or create the directory manually by using < mkdir output/hourly > |

Table 1 Sample ibis.infile input

```
0          ! irestart  0: not a restart run 1: restart run
2002       ! irstyear  actual calendar year of restart run
2001       ! iyear0    initial year of simulation (don't change for restart)
1          ! nrun      number of years in this simulation (change for restart)
```

| | |
|---|---|
| zedx | ! datasource: file in input_descriptions directory describing the climate data source (remember to also change comgrid.f if necessary) |
| .false. | ! weather_generator: if true, use weather generator prior to iyrdaily; if false, use random year of data |
| 1948 | ! iyranom year to start reading anomalies (don't chng for restart) (ignored if weather_generator=FALSE: then we read anomalies whenever we read daily data) |
| 1948 | ! iyrdaily year to start reading daily data (ditto) |
| 9999 | ! imetyear year to start using hourly met data from arlington field station (ditto) |
| 9999 | ! imetend year to end reading in hourly met data from arlington field station (ditto) |
| -1 | ! seed initial random number seed: should be < 0, or 0 to seed the random number generator using the current time |
| 0 | ! soilcspin 0: no soil spinup, 1: acceleration procedure used |
| 1 | ! iyearout 0: no yearly output, 1: yearly output |
| 0 | ! imonthout 0: no monthly output, 1: monthly output |
| 1 | ! idailyout 0: no daily output, 1: daily output |
| 0 | ! ihourlyout 0: no hourly output, 1: hourly output |
| 1 | ! isimveg 0: static veg, 1: dynamic veg, 2: dynamic veg-cold start |
| 0 | ! isimfire 0: wildfire off, 1: wildfire on |
| 0 | ! isimco2 0: fixed co2, 1: ramped co2 |
| 1 | ! irrigate_flag 0: no irrigation 1: irrigation strategy used everywhere 2: read map specifying where irrigation is used |
| 0 | ! nstress 0: no nitrogen stress, 1: apply nitrogen stress |
| 0 | ! isoybean 0: soybeans not grown 1: soybeans grown everywhere |
| 1 | ! imaize 0: maize not grown 1: maize grown everywhere |
| 0 | ! iwheat 0: no wheat grown 1: spring wheat grown everywhere 2: winter wheat |
| 0 | ! irotation 0: none 1: winter wheat/fallow 2: 2 corn/soy 3: corn/corn/soy 4: soy/winter wheat/corn |
| 0 | ! iholdsoiln 0: doesn't save soil inorganic N from restart 1: save inorganic soil N |
| 0 | ! isoil_layer 0(default): no; 1: yes Read-in soils with layered data from a NetCDF file. |
| 0 | ! iyr_pft 0(default): no; 1: yes Read-in pft on a year-by-year basis from NetCDF files. |
| 0 | ! iyr_fert 0(default): no; 1: yes Read-in fertilizer application on a year-by-year basis from NetCDF files. |
| 1.0 | ! ffact numeric multiplying factor applied to N fertilizer after end of fertilizer data (for all crops) |
| .false. | ! management_prescribed: If true, use prescribed planting dates & cultivars, as read from maps; if false, use prognostic planting dates & cultivars |
| transient | ! planting_input: Suffix on the planting date input file name (e.g., 'transient', 'detrended', 'linearized' or 'fixed'), for management_prescribed=true |
| transient | ! cultivar_input: Suffix on the cultivar input file name (e.g., 'transient' or 'fixed'), for management_prescribed=true |
| 1 | ! overveg 0: PFTs compete based on climate constraints 1: PFTs compete based on current land cover & climate |
| 9 | ! isoilay soil layer for which nitrate leaching/drainage is output |
| 0.000370 | ! co2init initial co2 concentration in mol/mol (real) |
| 0.209000 | ! o2init initial o2 concentration in mol/mol (real) |
| 3600. | ! dtime time step in seconds |
| 0 | ! idiag 0: no diagnostic output, 1-10 # of files to output |
| 41.1851 | ! snorth northern latitude for subsetting in/output |
| 41.1651 | ! ssouth southern latitude for subsetting in/output |
| -96.5222 | ! swest western longitude for subsetting in/output |
| -96.4222 | ! seast eastern longitude for subsetting in/output |

Color highlighted indicates new input in this version.

2.2 File I/O

Table 2 File I/O

| Directory | Components |
|---|---|
| input | General:<br>o **frate*.nc** files (corn, soy, wheat) and **frate.hdr** (text file): fertilizer rate and its specification for first year and number of years in freate*.nc file (*variables <fertmaize>, <fertsoy>, <fertwheat>*)<br>o **vegtype.nc**: fixed vegetation map (*variable <xinveg>, total vegetation type = 16 see* Table 5 Vegetation types)<br>o **soil_text.nc**: 3-d soil texture array (*variable <domtext>*)<br>o **surta.nc**: 2-d surface and vegetation arrays (*variable <lmask> 0=water; 1=land*)<br>o **topo.nc**: topography in [m] (*variable <xintopo>*)<br>o **deltat.nc**: absolute minimum temperature - temp on average of coldest month (C) (*variable <deltat>)*<br>Zedx (specified in input_descriptions):<br>o **RADS_60year_climo.nc**: climatological solar radiation (W m-2) (*variable < clmrads>*)<br>o **TMIN_60year_climo.nc**: climatological daily min temp. (C) (*variable < clmtmin>*)<br>o **TMAX_60year_climo.nc**: climatological daily max temp. (C) (*variable < clmtmax>*)<br>**PREC_60year_climo_mm_day.nc**: climatological precipitation (mm/day) (*variable <clmprec>*).<br>o **RELH_60year_climo.nc**: climatological relative humidity (%) (*variable <clmq>*).<br>o **WSPD_60year_climo.nc**: climatological wind speed (m s-1) (*variable <clmw>*).<br>o **TMIN_annual_60year_climo.nc**: annual average minimum air temperature (deg C) (*variable <tminavgann>*). |

| | |
|---|---|
| | ○ **daily/rads_*.nc**: daily solar radiation anomaly (*variable <xinradsd>*)<br>○ **daily/tmin_*.nc**: daily 2m temp minimum anomaly (C, additive anomaly) , or daily temp minimum (C) (variable < xintmind>)<br>○ **daily/tmax_*.nc**: daily temp maximum anomaly (C, additive anomaly), or daily temp maximum (C) (*variable <xintmaxd>*).<br>○ **daily/prec_*.nc**: daily precipitation anomaly (fraction, multiplicative anomaly), or daily precipitation (mm/day) (*variable <xinprecd>*).<br>○ **daily/relh_*.nc**: daily relative humidity anomaly (fraction, multiplicative anomaly), or daily relative humidity (%) (*variable <xinqd>*).<br>○ **daily/wspd_*.nc**: daily wind speed anomaly (fraction, multiplicative anomaly), or daily wind speed (m/s) (*variable <xinwindd>*). |
| output | yearly:<br>○ **npp.nc**: annual net primary productivity, by pft and total<br>○ **aet.nc**: annual average evapotranspiration<br>○ **runoff.nc**: annual runoff<br>○ **gdd.nc**: annual gdd info - climate and current year<br>○ **wsoi.nc**: annual average soil moisture, soil ice, volumetric water content, plant available water<br>○ **solar.nc**: annual average solar incident radiation<br>○ **ir.nc**: annual average solar infrared radiation<br>○ **sens.nc**: annual average sensible heat flux<br>○ **firefrac.nc**: fraction of grid cell lost to fire<br>○ **latent.nc**: annual average latent heat flux<br>○ **netrad.nc**: annual average net radiation<br>○ **plai.nc**: annual leaf area index<br>○ **biomass.nc**: annual biomass of carbon, by pft, upper canopy, lower canopy<br>○ **crops.nc**: annual crop growth variables<br>○ **csoi.nc**: annual total soil carbon<br>○ **litterfall.nc**: annual litterfall<br>○ **nsoi.nc**: annual total soil nitrogen |

| | |
|---|---|
| | ○ **totfall.nc**: annual total litterfall carbon<br>○ **csoislo.nc**: annual total soil carbon in slow pool<br>○ **csoipas.nc**: annual total soil carbon in pasive pool<br>○ **co2fluxes.nc**: annual total carbon from exchange of co2<br>○ **vegtype0.nc**: annual vegetation type - ibis classification<br>○ **fcover.nc**: annual fractional cover of canopies<br>○ **zcanopy.nc**: annual height of vegetation canopies<br>○ **exist.nc**: annual existence of each plant functional type<br>○ **frac.nc**: fraction of canopy occupied by each pft<br>monthly:<br>○ **rain.nc**: monthly average rainfall<br>○ **snow.nc**: monthly average snowfall<br>○ **aet.nc**: monthly average evapotranspiration<br>○ **transETratio.nc**: monthly average transpiration:ET ratio<br>○ **runoff.nc**: monthly average total runoff<br>○ **tsoi.nc**: monthly average soil temperature<br>○ **wsoi.nc**: monthly average soil moisture<br>○ **snod.nc**: monthly average snow depth<br>○ **snof.nc**: monthly average snow fraction<br>○ **solar.nc**: monthly average incident solar radiation<br>○ **netrad.nc**: monthly average net radiation<br>○ **ir.nc**: monthly average infrared radiation<br>○ **sens.nc**: monthly average sensible heat flux<br>○ **latent.nc**: monthly average latent heat flux<br>○ **lai.nc**: monthly average leaf area index for upper and lower canopies<br>○ **npptot.nc**: monthly total net primary productivity of carbon<br>daily:<br>○ **trunoff.nc**: daily average total runoff<br>○ **srunoff.nc**: daily average surface runoff<br>○ **drainage.nc**: daily average drainage |

| | |
|---|---|
| | ○ **snod.nc**: daily average snow depth<br>○ **snof.nc**: daily average snow fraction<br>○ **leachr.nc**: daily rate of nitrogen leached from profile (kg N m-2 y-1)<br>hourly:<br>○ **crop_properties.out**: Biomass, daily timesteps, biomass(npoi, npft), Leaf area index, daily timesteps plai(npoi, npft), Canopy height, daily timesteps, ztop(npoi,canopy), canopy = 1 for crops<br>○ **soil_properties.out**: Soil moisture, hourly timesteps, wsoi(npoi, nsoilay), Soil temperature, hourly timesteps, tsoi(npoi, nsoilay)<br>○ **gdd_properties.out**: GDD (plant), daily timesteps, gddplant(npoi, npft), GDD (soil), daily timesteps, gddsoi(npoi, npft) |
| restart | see subroutine <wrestart> in io.f for details |
| input_descriptions | Contains namelist file (text file) describing set up for climate dataset:<br>○ **cru_ncep_us**: using a combination of CRU (monthly values) and NCEP (daily anomalies) over the U.S. (for daily files: start year: 1948; number of years: 55).<br>○ **cru_ncep_us_2005**: using a combination of CRU (monthly values) and NCEP (daily anomalies) over the U.S. The climate files used here go through 2005 (for daily files: start year: 1948; number of years: 58).<br>○ **zedx**: climate data used when using zedx daily data (for daily files: start year: 1948; number of years: 60) |

**Appendix A Installation**

Important note: At this moment, it requires password for the subversion account in the server <https://trac.agron.iastate.edu:443> ISU Agron Trac. An alternative way to distribute the code is highly suggested to avoid the distribution of this info below.

user: adamward

password: adam

Step 1: check out the code from the server

```
svn checkout https://trac.agron.iastate.edu/prepos/agro-ibis/branches/adamward-THMB
```

(see section 1.2 for reference)

If a warning message shows up such as "Error validating server certificate for 'https://trac.agron.iastate.edu:443'), in the required action that followed, choose the option <accept (t)emporarily> or <accept (p)ermanently), then provide the password for 'adamward'. This should check out the code < adamward-THMB> from the server.

Step 2: compile the code

Follow steps in the document <README-first> to compile and set up the code. As an example,

(1) Run 'initial_setup' to start fresh

(2) Create an 'input' directory by linking

ln -s /Users/WardLab/Documents/agro-ibis/input_zedx input

(3) Configure makefile.config for your compiler, if needed

(4) Modify ibis.infile and comgrid.f, if necessary

(5) Run 'make'

If error occurred during compilation, you may want to check makefile or makefile.config (a common error is the incorrect path for the netcdf library in your machine).

In the above, steps (3) -(5) to compile the code, and steps (1) and (2) to set up the configuration to run the code. After this, the code should be readily compiled with the program name 'ibis' (some warning may occur due to the compiler or the code itself).

Step 3: testing

To test the code, simply run <./ibis> after setting up the < ibis.infile>.

## Appendix B Code Structure[1]

- <com*.f> are global modules for the main code to use (i.e. served as common utilities).
- <physiology.f> canopy physiology calculations.
    - <params.can> canopy parameters input
    - <vegetation.f> vegetation dynamics, soil temperature
    - <io.f> for I/O use
    - <params.*> Input parameters
    - <params.can> canopy parameters
    - <params.veg> vegetation PFT parameters
    - <params.soi> soil parameters
- <params.crp> crop parameters

Table 3 Common variables

| | |
|---|---|
| npts | number of hourly points of a year |

[1] Not intended to be comprehensive, for more detailed information, see < Agro-IBIS Quickstart Guide> page 2.

## Appendix C Common ibis classes

Table 4 Plant functional types (pft) [2]

| pft integer | pft |
|---|---|
| 1 | Tropical broadleaf evergreen trees |
| 2 | Tropical broadleaf drought---deciduous trees |
| 3 | Warm---temperate broadleaf evergreen trees |
| 4 | Temperate conifer evergreen trees |
| 5 | Temperate broadleaf cold---deciduous trees |
| 6 | Boreal conifer evergreen trees |
| 7 | Boreal conifer cold---deciduous trees |
| 8 | Boreal broadleaf cold---deciduous trees |
| 9 | Evergreen shrubs |
| 10 | Deciduous shrubs |
| 11 | Warm (c4) grasses |
| 12 | Cool (c3) grasses |
| 13 | Soybean |
| 14 | Maize |
| 15 | Wheat |

[2] See <io.f> for complete reference.

Table 5 Vegetation types[3]

| vegtype0 integer | vegtype0 |
|---|---|
| 1 | tropical evergreen forest / woodland |
| 2 | tropical deciduous forest / woodland |
| 3 | temperate evergreen broadleaf forest / woodland |
| 4 | temperate evergreen conifer forest / woodland |
| 5 | temperate deciduous forest / woodland |
| 6 | boreal evergreen forest / woodland |
| 7 | boreal deciduous forest / woodland |
| 8 | mixed forest / woodland |
| 9 | savanna |
| 10 | grassland / steppe |
| 11 | dense shrubland |
| 12 | open shrubland |
| 13 | tundra |
| 14 | desert |
| 15 | polar desert / rock / ice |
| 16 | croplands |

[3] See < vegetation.f> for complete reference.

**Appendix D Note on NetCDF**

Frequently used subroutines < vec2arr(vect, array) >, < arr2vec(array, vect) >:

variable <vect> is the source data, variable <array> is used to write to the NetCDF file. The key to use this subroutine is to keep in mind that <vect> is assumed to be the length of <npoi> (doesn't matter whether it is 1-D or 2-D vector), <array> is 2-D with length of nlonsub*nlatsub. After calling this function, the data from <vect> will be assigned to the corresponding location of <array>, while others (of 2-D form) will be assigned some dummy values. The index of the 1-D variable used to call this function (e.g. cdummy) will be filled accordingly.

Below is a standalone example to show how to write the NetCDF file (included in the source code as well).

**Appendix E THMB (Terrestrial Hydrology Model with Biogeochemistry)**

THMB: http://www.sage.wisc.edu/download/HYDRA/hydra.html

Installation

Note: the original source code has been modified. This installation guide refers to the modified code that should come with the distribution of the branch adamward-THMB. Also, a package as <Libs_tar> is provided along with this distribution for the inclusion of dependent libraries. You may use this provided package, or obtain the prerequisites per your preference. This procedure for compiling the code has been successfully tested in a linux machine and a Mac.

Note: for Mac users, it might be useful to use MacPorts (link) to install and organize the libraries, without having to do all the "dirty" work to building all the libraries of your own.

<Libs_tar> includes:

- NetCDF source code (netcdf-fortran-4.4.1.tar.gz)
- HDF4 source code (hdf-4.2.11.tar)
- ZLIB and JPEG source code (zlib-1.2.8.tar.gz, jpegsrc.v8d.tar.gz)

<Prerequisites>

1. THMB source code
2. NetCDF library
3. HDF4 library with ZLIB and JPEG (external library) (HDF5 is NOT supported in the code)

<Preparing libraries>

1. NetCDF library (if not installed)
2. HDF4 (if not installed)

The libs are built using <configure> (link). Procedure below uses libs provided in

<Libs_tar>.

1) Extract these individual compressed files to a directory of your preference, say, <Libs>, such that this directory contains 3 sub-directories as <hdf-4.2.11>, <jpeg-8d>, and <zlib-1.2.8>. You will then go to the individual sub-directory to build the corresponding lib.

2) We need to build libs <jpeg-8d> and <zlib-1.2.8> before building <hdf-4.2.11>. It doesn't matter which one to build first. The building process is very similar for all these libraries, except that some customized options may be needed.
3) For <jpeg-8d> and <zlib-1.2.8>, the default installation path for the lib is at '/usr/local/lib/'. If you don't have administrator right, you may want to install them into a different path.

   If you have administrator right, run the following without quote ' ' in the root directory of <jpeg-8d> and <zlib-1.2.8>:

   './configure'

   'make install'

   The installed libs will be located in /usr/local/lib (if not, check the configure file for the default setting).

   If you don't have administrator right, you will need to install them in a different path, run the following without quote ' ' in the root directory of <jpeg-8d> and <zlib-1.2.8>:

   './configure --prefix= YOUR_PATH' (where YOUR_PATH is your prefer path)

   'make install'

   The name of the libs should look like 'libjpeg.a' and some other libs for <jpeg-8d>, and 'libz.a' and some other libs for <zlib-1.2.8>.
4) For <hdf-4.2.11>, the built lib will be located at the root directory of <hdf-4.2.11>, as <./hdf-4.2.11/hdf4/lib>. Check the location of the libs for <jpeg-8d> and <zlib-1.2.8>, as you will need to specify their path to build <hdf-4.2.11>, by running the following:

   './configure --with-zlib=path_to_ZLIB_install_directory --with-jpeg= path_to_JPEG_install_directory'

   'make install'

Your compiled lib for HDF4 should be located at <./hdf-4.2.11/hdf4/lib> as 'libdf.a' and some other libs.

<Compiling THMB>

It becomes very simple to compile the code after having all these libraries built. In <makefile>, change path for the <*_INCLUDE> and <*_LIB> to their appropriate directory. Note that you only need to change the path for these variables, while the options in these variables shown as -l (like -lnetcdf, -ldf, -lz, -ljpeg), -I and -L (right before the path), should remain unchanged.

For NetCDF lib, if it is already installed in the machine, try ‘nc-config --all’, and use the information shown in --fflags and --flibs to specify $(CDF_INCLUDE) and $(CDF_LIB). With all options changed appropriately, simply run ‘make’. The compiled should be shown as ‘hydra.x’ by default.

<Test run>

The distribution provides sample input (climate, geomorphology) for testing in <sample_input>. Testing setup can be found in file <READMEhy.txt>.

File I/O

Read-in files

Table 6 THMB read-in files

| Name | Components |
|---|---|
| Geomorphology<br>(in hdf format) | Depending on the setup of <laket> in <hydra.inf>, different read-in files are needed by the code, as<br>❖ General (Needed by all setups)<br>o hdf.basin: basin mask file.<br>o tbase.hdf: digital elevation model.<br>o hdf.rivdir.mix: river direction file.<br>o hdf.mflac: sill elevation file.<br>❖ laket = 0 (lakes will be simulated)<br>o hdf.outnewi, hdf.outnewj: location of the sill for each lake.<br>❖ laket = 1 (lakes will be prescribed based on observed modern lakes areas of Cogley, 1989.<br>o hdf.coglak5<br>o hdf.outcogi, hdf.outcogj<br>❖ laket = 2 (lakes will be prescribed based on lake areas previously calculated by the code)<br>o hdf.larea<br>o hdf.outnewi, hdf.outnewj |
| Monthly input of runoff and drainage (from IBIS) , precipitation and evaporation (from OBS)<br>(in NetCDF format) | Path for this input files can be specified by variable <netcdf_input> in <hydra.inf><br>o srunoff.nc: runoff input in [mm/day] (*variable <runin>*).<br>o drainage.nc: drainage input in [mm/day] (*variable <drainin>*).<br>o rain.nc: precipitation input in [mm/day] (*variable <prcpi>*).<br>o evap.nc: evaporation input in [mm/day] (*variable <evapi>*). |

Table 7 THMB output files

| Format | Components |
|---|---|
| NetCDF | ○ sflux.wat.nc: time series of monthly river discharge in [$m^3/s$] (*variable* <*sflux*>).<br>○ watelv.wat.nc: time series of monthly water elevation in [m] (*variable* <*elevm*>).<br>○ deptm.wat.nc: time series of monthly water volume in [$m^3$] (*variable* <*deptm*>)<br>○ aream.wat.nc: time series of monthly water area in [$km^2$] (*variable* <*aream*>).<br>○ lakem.wat.nc: time series of monthly water indicator (*variable* <*lakem*>). |

**Appendix F Coupling Agro-IBIS and THMB**

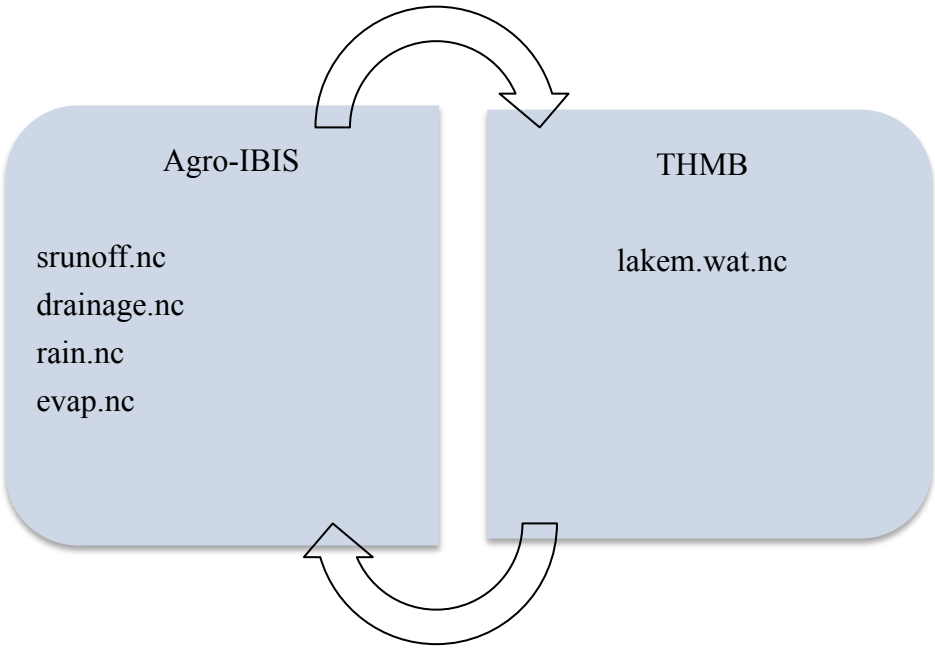


Figure 1 File communication for Agro-IBIS and THMB coupling

Notes on the coupling:

1. When using <Panoply> to view the IOs (*.nc) for Agro-IBIS and THMB, viewing error may occur but it will **NOT** affect the correctness of the data communication, currently known possibility include:

- Using 1 point grid, it won't show variable of latitude and longitude, also it shows error when trying to view the variable of interest.
- Variable at the 3rd dimension (shown as 'level') is for signaling whether communication is ready to make and for checking. It may show error when trying to view this variable in the netcdf viewer.

**Appendix G Major Development History**

| Revision | Note |
|---|---|
| Trunk | Starting the code from the trunk version |
| #330 | - read-in NetCDF capabilities<br>- hourly output by merging branch <jpatton_hourly> rev # 213 |
| | biofuel crops capability by merging branch <vanloocke_energycrops> rev #319 |

**Index**