Day 3: Analyzing Model Output

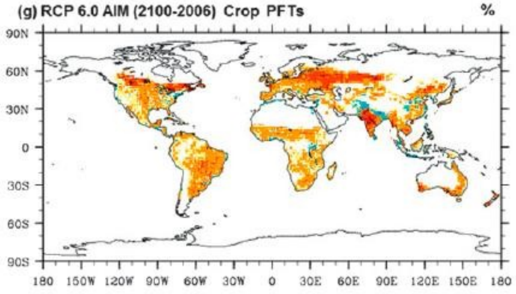
*Keith Oleson, *Quinn Thomas, Sanjiv Kumar, Sheri Mickelson, *Ahmed Tawfik, *Danica Lombardozzi

oleson at ucar.edu

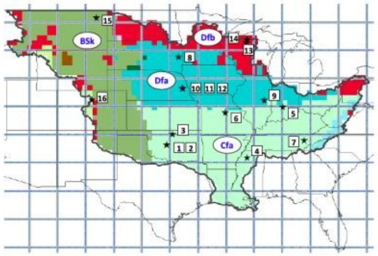
Thanks to Rosie Fisher, Will Weider, Kyla Dahlin, Adam Phillips, Dennis Shea, Susan Bates, Christine Shields, and Dave Bailey for their assistance.

NCAR
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH

U.S. DEPARTMENT OF ENERGY | Office of Science



Motivation

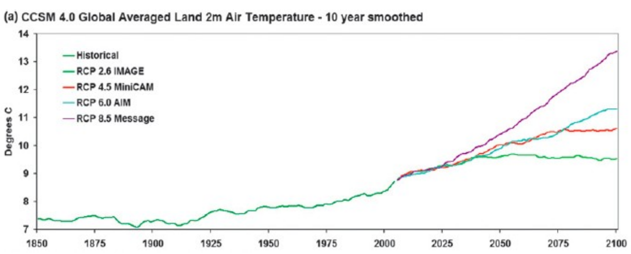


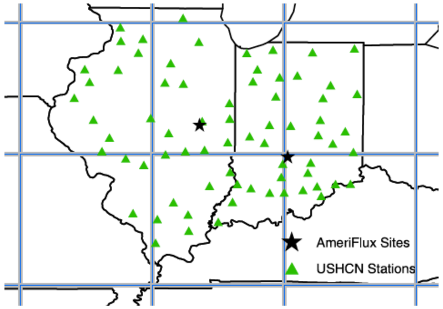
To find out one important thing from the sea of data (GB to TB model outputs)

To compare model outputs with the observations at a point (flux tower), regional (water basin), and global scales.

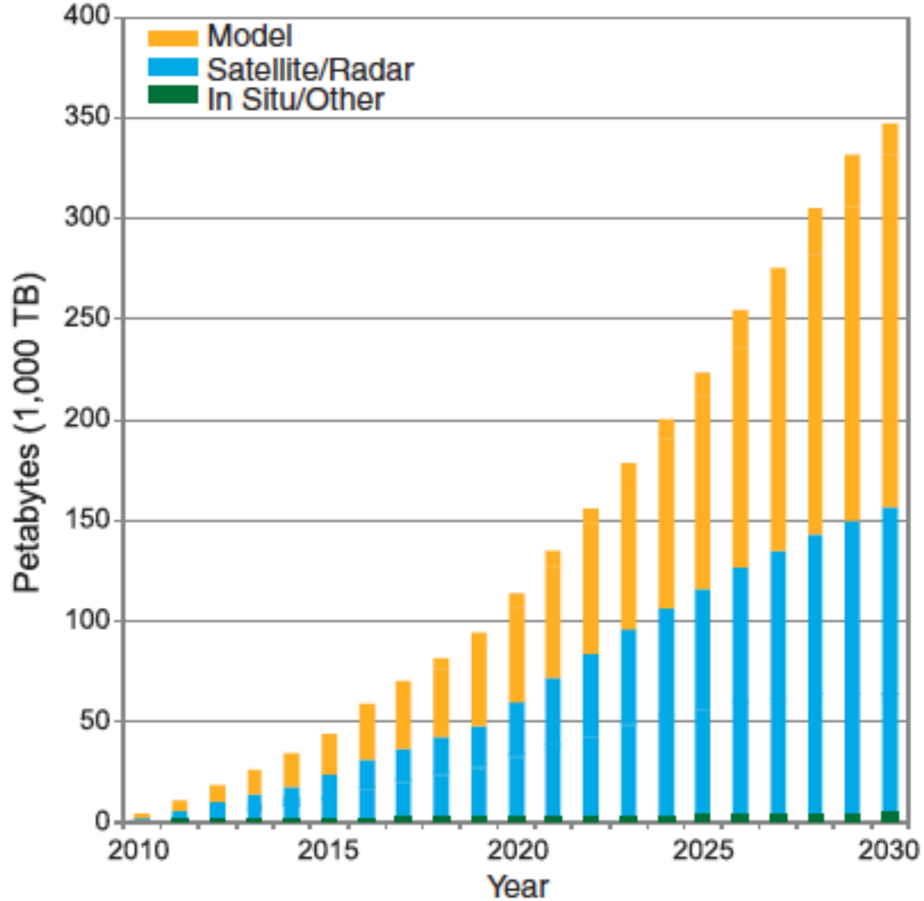
To make a sound scientific argument that is supported by the quantitative analysis of model results and the observations.

Publish your results





Climate Data Explosion



Climate Date Volume (worldwide) Projections

Overpeck et al. 2011, Science

Today's Goal

① Where are model outputs (Keith)?

② Understanding the data (netCDF, ncview, ncdump) (Keith)

③ Analysis in R (Quinn)

④ Introduction to NCL (**N**CAR **C**ommand **L**anguage) (Sanjiv)

⑤ The Land Diagnostic Package and ILAMB (Keith & Sheri)

Today's Goal

① **Where are model outputs (Keith)?**

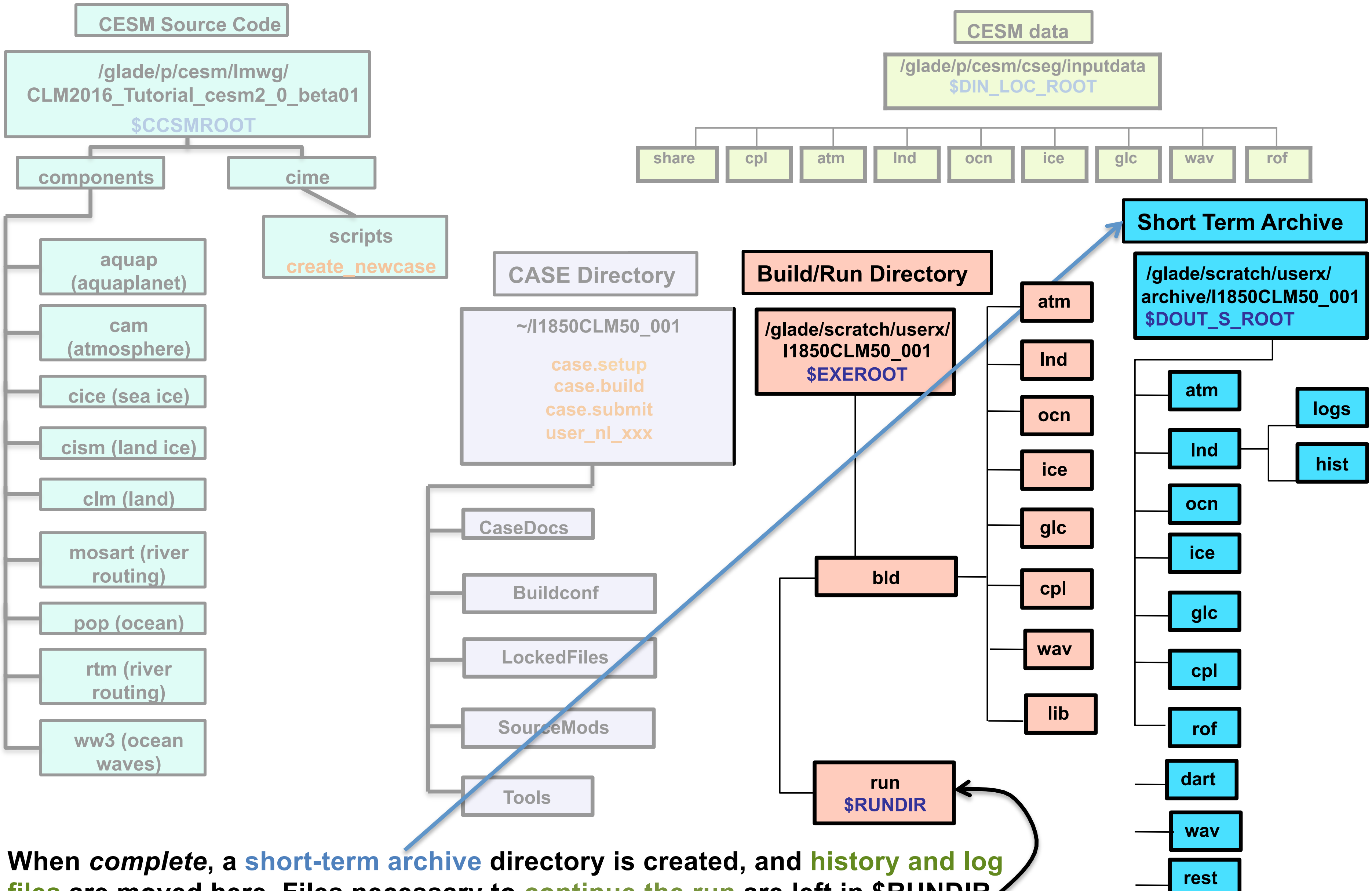
② Understanding the data (netCDF, ncview, ncdump) (Keith)

③ Analysis in R (Quinn)

④ Introduction to NCL (**N**CAR **C**ommand **L**anguage) (Sanjiv)

⑤ The Land Diagnostic Package and ILAMB (Keith & Sheri)

CESM Directory Structure



When *complete*, a short-term archive directory is created, and history and log files are moved here. Files necessary to continue the run are left in $RUNDIR

CESM History File Naming Conventions

All history output files are in “netCDF” format

Location of history files in short-term archive directory:
/glade/scratch/<logname>/archive/$case/<component>/hist
component = lnd, atm, ocn, etc.

CESM distinguishes between different time sampling frequencies by creating distinct history files for each frequency. *Sampling frequencies are set by the user within the namelist.*

Example history file names:

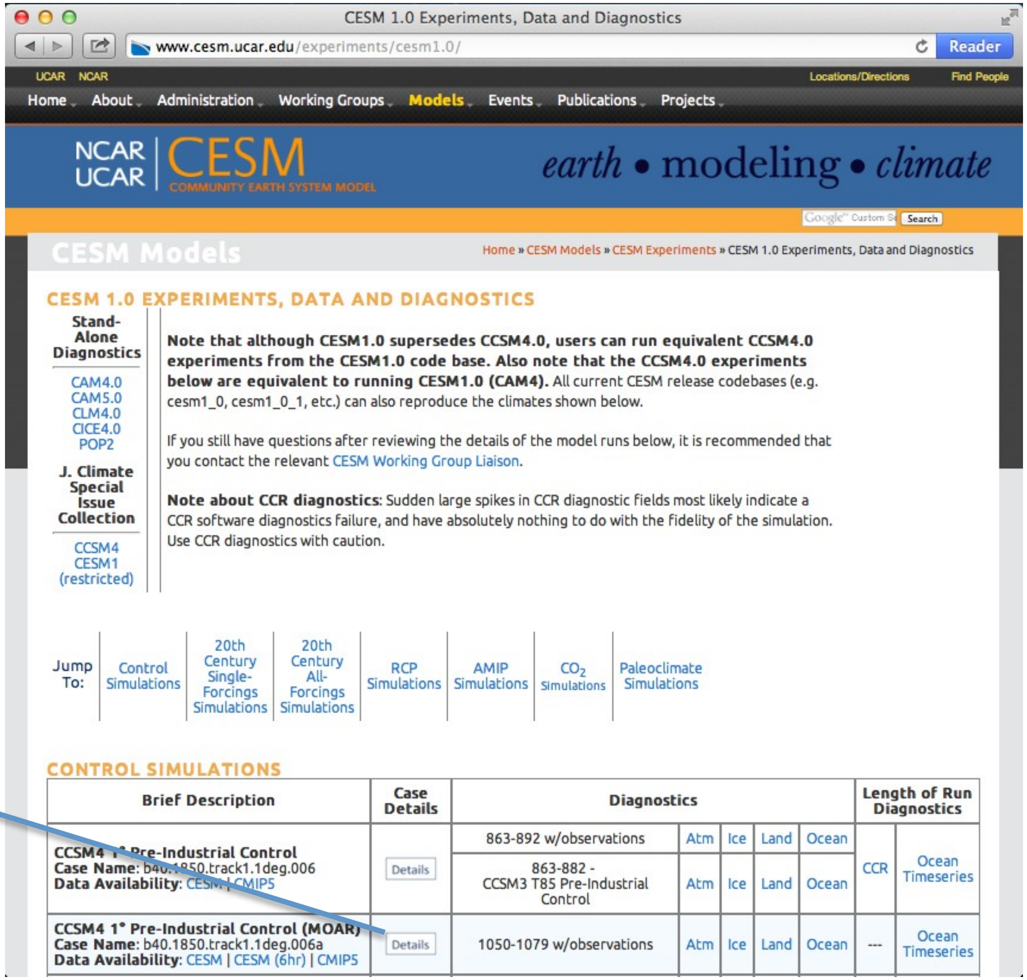
f40_test.cam2.h0.1993-11.nc
f40_test.clm2.h0.1993-11.nc
f40_test.pop.h.1993-11.nc
f40_test.cice.h.1993-11.nc

By default, h0/h denotes that the time sampling frequency is monthly.

Other frequencies are saved under the h1, h2, etc file names:
f40_test.cam2.h1.1993-11-02-00000.nc

CESM Experiments Pages

Case Name: b40.1850.track1.1deg.006a
Machine: NCAR:bluefire
CMIP5 ID: ---
Compset: B_1850_CN
Resolution: 0.9x1.25_gx1v6
Years: 953-1108
Initialization: 0953-01-01 of b40.1850.track1.1deg.006
HPSS Location: /CCSM/csm
/b40.1850.track1.1deg.006a
Case Details: MOAR control simulation. Extra output saved.
Start/End Dates: 12/29/10, 2/2/11
Data Release Date (Full): 7/15/11

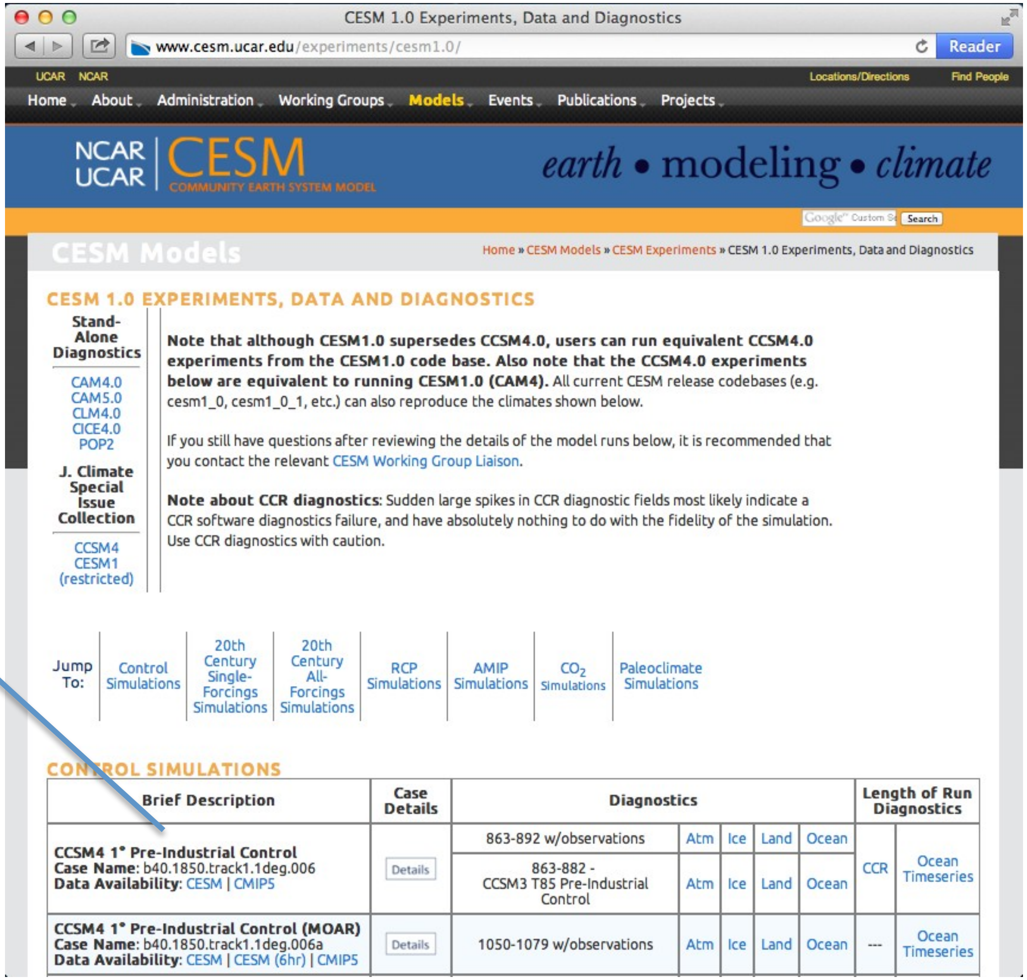


http://www.cesm.ucar.edu/experiments/cesm1.0/
http://www.cesm.ucar.edu/experiments/cesm2.0/

CESM Experiments Pages

CCSM4 1° Pre-Industrial Control
Case Name: b40.1850.track1.1deg.006
Data Availability: CESM | CMIP5

Links to the Earth System Grid



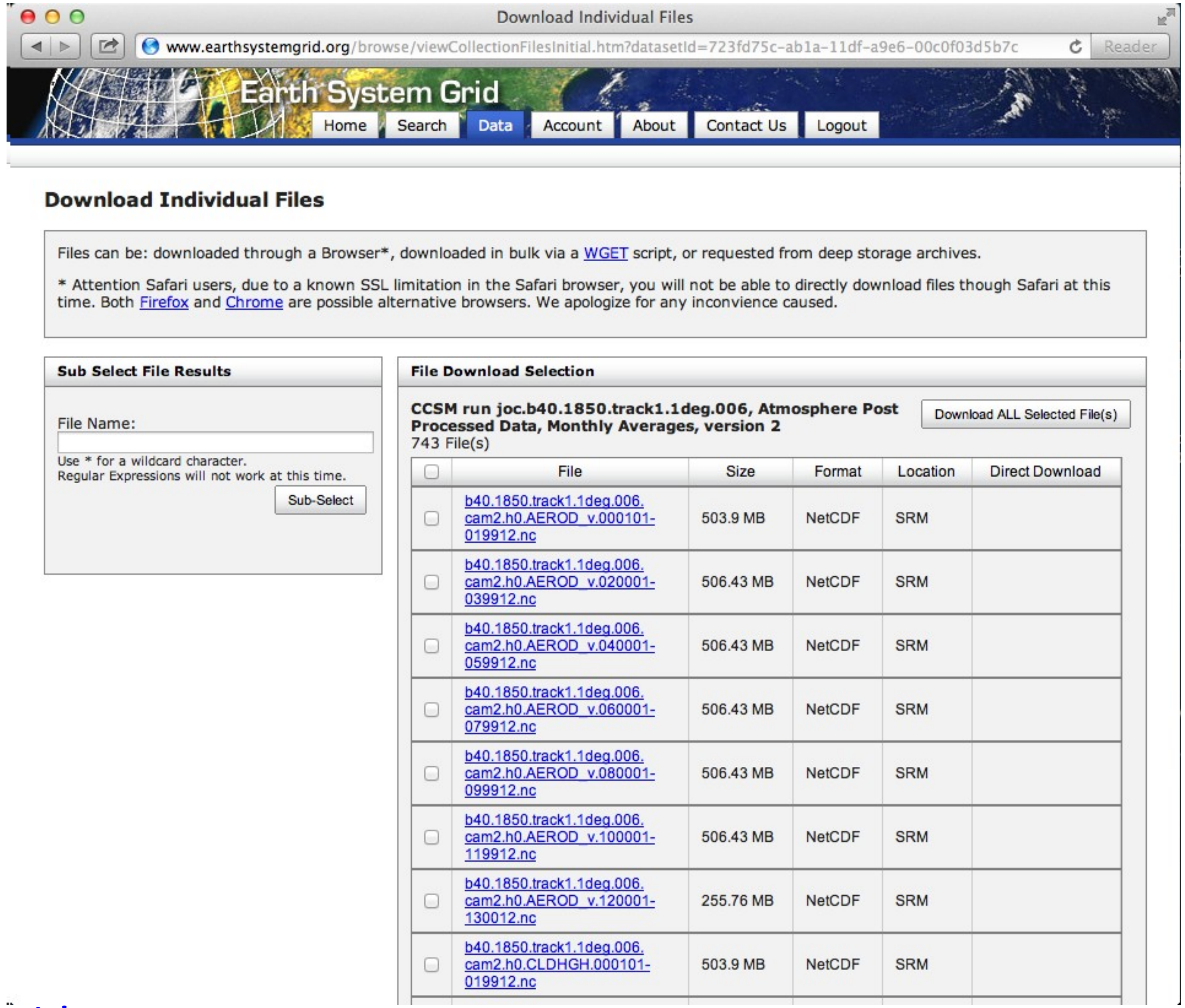
http://www.cesm.ucar.edu/experiments/cesm1.0/
http://www.cesm.ucar.edu/experiments/cesm2.0/

Earth System Grid

Publicly released CESM data is available via the ESG.

Registration is quick and easy. NCAR accounts are not required.

Post-processed data in CESM and CMIP formats along with raw history files are provided.



http://www.earthsystemgrid.org

Today's Goal

1. Where are model outputs (Keith)?
2. **Understanding the data (netCDF, ncview, ncdump) (Keith)**
3. Analysis in R (Quinn)
4. Introduction to NCL (**N**CAR **C**ommand **L**anguage) (Sanjiv)
5. The Land Diagnostic Package and ILAMB (Keith & Sheri)

Understanding the data

Change directory to your archive directory

cd /glade/scratch/$USER/archive/$CASE/lnd/hist

OR our data

cd /glade/scratch/dll/CLMTutorial2016_DataForAnalysis/I2000CLM50_001

All analysis should be done on 'geyser' machines and not on 'yellowstone'. Yellowstone nodes are for login and production/ development simulations, whereas 'geyser' and 'caldera' are for data analysis and visualizations. Read best practices at:

https://www2.cisl.ucar.edu/user-support/cisl-best-practices.

Login to geyser from your Yellowstone login node

bsub -Is -q geyser -W 4:00 -n 1 -P UCGD0002 xterm &

Introduction to netCDF

- netCDF stands for “network Common Data Form”
- All CESM outputs are in netCDF format (.nc)
- self-describing, portable, metadata friendly
- supported by many languages including fortran, C/C++, Matlab, **R**, ferret, GrADS, **NCL**, IDL; viewing tools like **ncview** / **ncdump**; and tool suites of file operators (NCO, CDO)
- Data can be stored in several dimensions: ensembles X time X levels X latitudes X longitudes
- Not easy to edit – maintains the original data (you may want to make copy before editing a netCDF file)

http://www.unidata.ucar.edu/software/netcdf

http://www.unidata.ucar.edu/software/netcdf/docs/BestPractices.html

A Detailed Look at a netCDF Variable

```
>module load ncl
>ncl
>f1 = addfile("./I2000CLM50_001.clm2.h0.0001-01.nc", "r")
>var1 = f1->TSA
>printVarSummary(var1)
```

```
Variable: var1
Type: float
Total Size: 55296 bytes
            13824 values
Number of Dimensions: 3
Dimensions and sizes:   [time | 1] x [lat | 96] x [lon | 144]
Coordinates:
            time: [31..31]
            lat: [-90..90]
            lon: [ 0..357.5]
Number Of Attributes: 5
  long_name :   2m air temperature
  units :       K
  cell_methods :        time: mean
  _FillValue :  1e+36
  missing_value :       1e+36
ncl 3>
```

Data type

Dimension and size

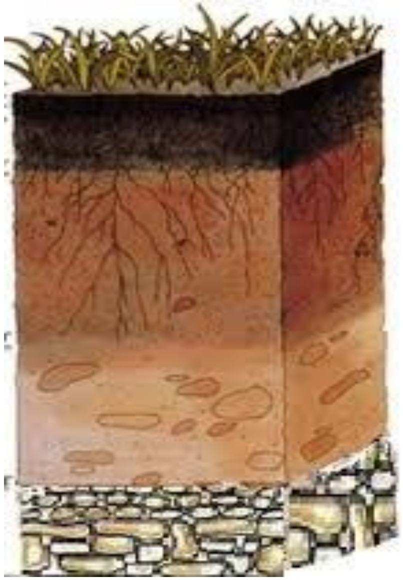
Coordinates

Attributes

```
>exit
```

Exercise in netCDF Variable

- Print Variable Summary for soil moisture (H2OSOI)
- Is there an extra dimension in H2OSOI?
- Explore this extra dimension (coordinates, size, etc.)

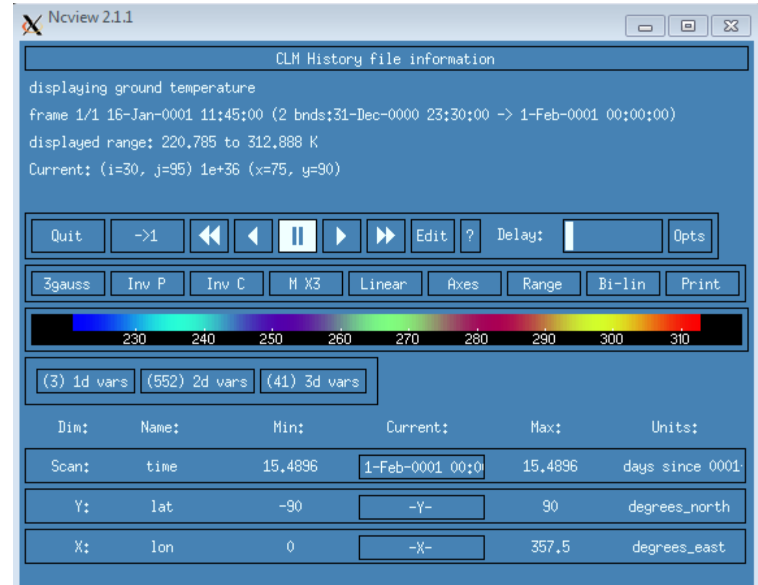


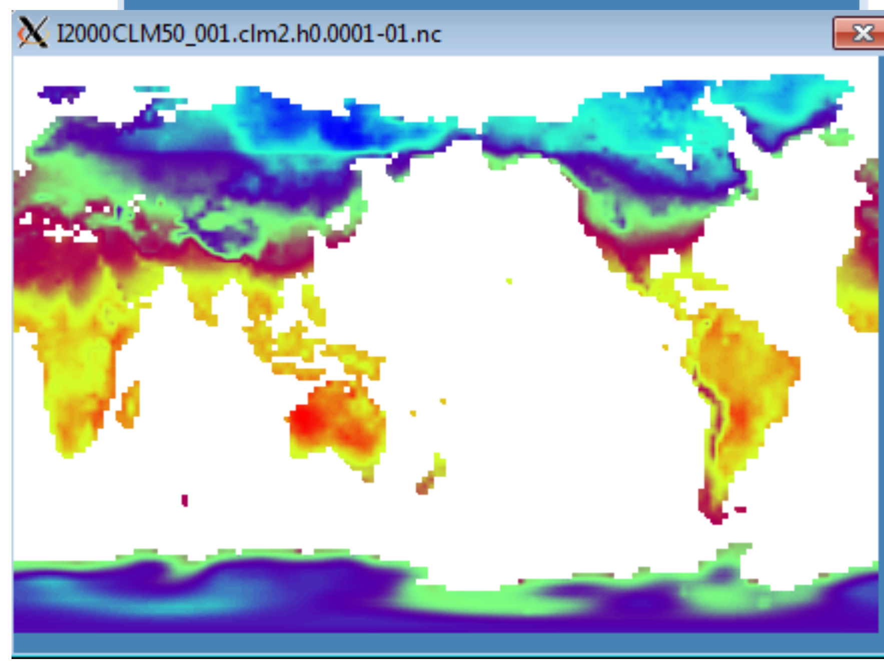
Introduction to ncview

ncview is a graphical interface which allows one to quickly view the variables inside a netCDF file.

ncview allows you to interactively visualize a selected variable across a selected range (time, spatial).

TG



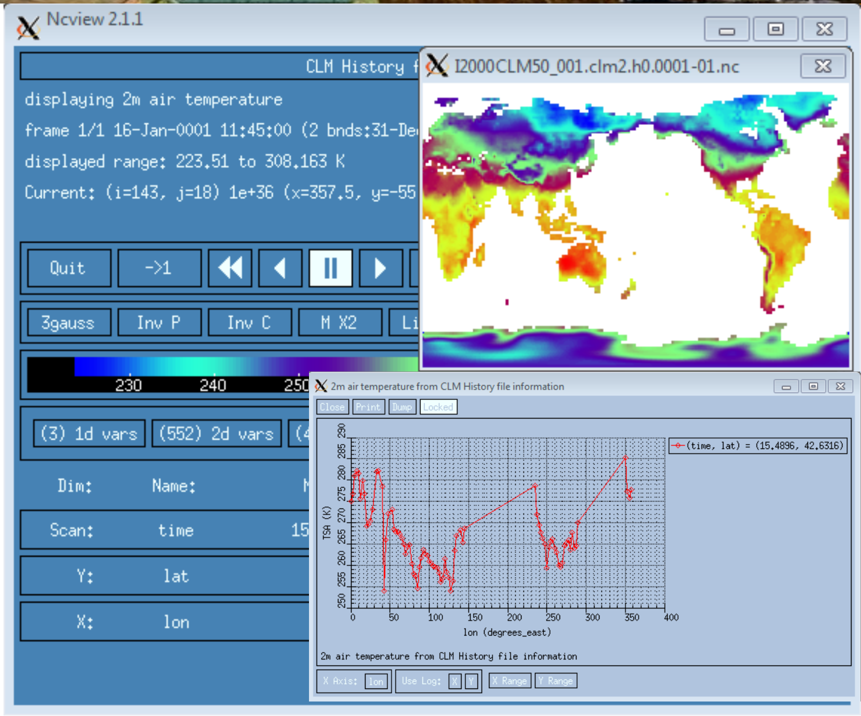



```
>module load ncview
>ncview I2000CLM50_001.clm2.h0.0001-01.nc &
```

You can also join several netCDF files

```
>ncview *0001-*.nc &
```

Exercise in ncview



- ◆Change the variable on the plot (e.g., TSA)
- ◆Click on any location on the plot
- ◆See XY plot
- ◆Change x – axis in the plot

Extra Credit Question

- ◆ Plot depth profile of soil moisture (H2OSOI)

Introduction to ncdump

ncdump is a netCDF utility that allows one to dump the contents of the netCDF file to screen or file.

Files are often too big to dump to screen, but one can look at subsets of the file using the different ncdump options.

Pipe the ncdump command through Less

>ncdump I2000CLM50_001.clm2.h0.0001-01.nc | less

Dump only header

>ncdump –h I2000CLM50_001.clm2.h0.0001-01.nc | less

Dump only one variable after the header; need to know the variable name

>ncdump –v TSA I2000CLM50_001.clm2.h0.0001-01.nc | less

oleson@geyser12:I2000CLM50_001

```
netcdf I2000CLM50_001.clm2.h0.0001-01 {
dimensions:
        lon = 144 ;
        lat = 96 ;
        gridcell = 5663 ;
        landunit = 21260 ;
        column = 69848 ;
        pft = 160456 ;
        levgrnd = 25 ;
        levurb = 5 ;
        levlak = 10 ;
        numrad = 2 ;
        levsno = 12 ;
        ltype = 9 ;
        nlevcan = 1 ;
        nvegwcs = 4 ;
        natpft = 17 ;
        string_length = 8 ;
        scale_type_string_length = 32 ;
        levdcmp = 25 ;
        hist_interval = 2 ;
        time = UNLIMITED ; // (1 currently)
variables:
        float levgrnd(levgrnd) ;
                levgrnd:long_name = "coordinate soil levels" ;
                levgrnd:units = "m" ;
        float levlak(levlak) ;
                levlak:long_name = "coordinate lake levels" ;
                levlak:units = "m" ;
        float levdcmp(levdcmp) ;
                levdcmp:long_name = "coordinate soil levels" ;
                levdcmp:units = "m" ;
        float time(time) ;
                time:long_name = "time" ;
```

http://www.unidata.ucar.edu/software/netcdf/docs/netcdf_utilities_guide.html#ncdump_guide

There are so much data/information in a single file. We need a better way to make sense out of all these files!

Motivation for writing analysis script

Today's Goal

① Where are model outputs (Keith)?

② Understanding the data (netCDF, ncview, ncdump) (Keith)

③ **Analysis in R (Quinn)**

④ Introduction to NCL (**N**CAR **C**ommand **L**anguage) (Sanjiv)

⑤ The Land Diagnostic Package and ILAMB (Keith & Sheri)

A Very Short Introduction to R

Quinn Thomas & Kyla Dahlin

CLM Workshop

Sept 2016

What is R?

- A free and open-source programming language and software environment for statistical computing and graphics
- Developed from another programming language, S.
- ‘Object oriented’ which essentially means it’s structured in a way that’s relatively intuitive

Good / Bad

- It's great because there are LOTS of user-submitted packages that expand its tools.
- Nice people sometimes publish their code in association with scientific papers.
- Almost anything you can imagine doing in R someone has already done – you just have to wade through the internet to figure out how to do it.
- Wading through the internet can be frustrating / exhausting. And the letter 'R' is not exactly google-able.
- Can be with slow with large datasets or numerous loops

Some Things to Know

- "#" delineates a comment
- Indexes start counting from **1** (ncl and python, among others, start with 0)
- When assigning variables, y <- 4 is the same as y = 4
- When you string things together in R use c(…)
- Logical operators :

  ==, !=, >, <, >=, <=, |, &, is.na(), !is.na()

Resources

- www.r-project.org
- www.rstudio.com
- www.google.com

For NetCDF in R:

www.image.ucar.edu/GSP/Software/Netcdf

Intro to Exercise

Where's the script?

/glade/p/cesm/lmwg/CLM2016_tutorial_space/Day3/

CLM_netcdf_R.R

Intro to Exercise (load R)

on yellowstone you can either run R by typing

>module load R/3.2.2

>R

(but only do this for very simple tasks)

or by opening an interactive job in ‘geyser’ (part of yellowstone)

>bsub -Ip -q geyser -W 2:00 -n 1 -P UCGD0002 **xterm**

then in that interactive window type

> module load R/3.2.2

>R

Install Packages

>install.packages(c("ncdf4", "raster", "rasterVis", "rgdal"))

you'll be asked to select a CRAN mirror (I like CA1, personally) then you'll see lots of text go by

then load the packages

>library(ncdf4)

>library(raster)

>library(rasterVis)

>library(rgdal)

Do Stuff

1. Open a .nc file
2. Extract a variable and convince R that it's actually a map of the whole globe
3. Plot it
4. Read in a variable from a time series (bunch of files)
5. Calculate point-wise mean and standard deviation
6. Calculate the highest annual value for vegetation for each gridcell
7. Clip out and look at a region (Australia)
8. Quit R

Today's Goal

① Where are model outputs (Keith)?

② Understanding the data (netCDF, ncview, ncdump) (Keith)

③ Analysis in R (Quinn)

④ **Introduction to NCL (NCAR Command Language) (Sanjiv)**

⑤ The Land Diagnostic Package and ILAMB (Keith & Sheri)

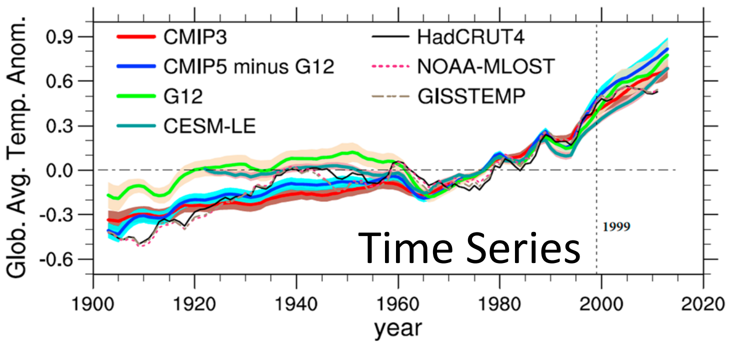
CLM5.0 Tutorial: Analyzing Model Outputs using NCL

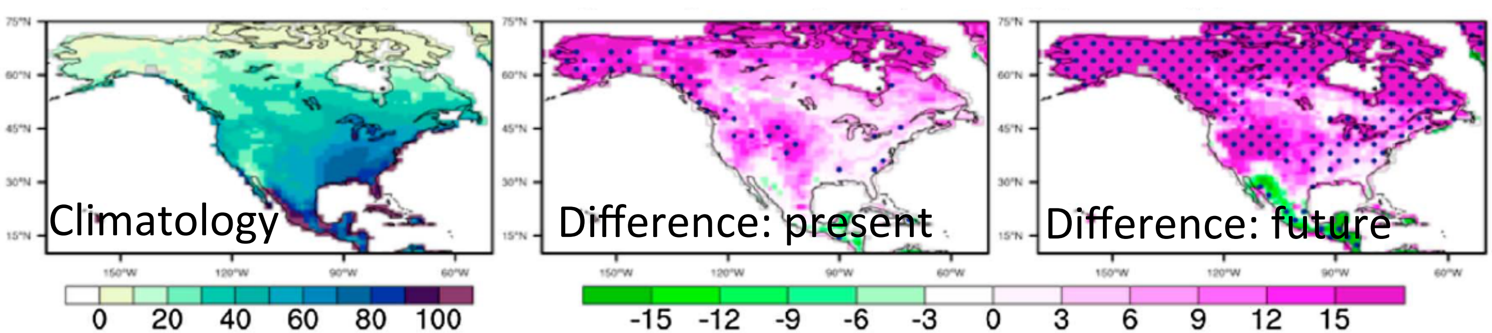
Sanjiv Kumar
sanjiv.kumar@noaa.gov

NATIONAL CENTER FOR ATMOSPHERIC RESEARCH
U.S. DEPARTMENT OF ENERGY | Office of Science
NCAR is sponsored by the National Science Foundation

Overview

- NCL – What, Why, and Where?
- NCL in interactive mode (command line)
- NCL in batch mode (scripting)
- Cool Graphics!
- Go for the hike!!

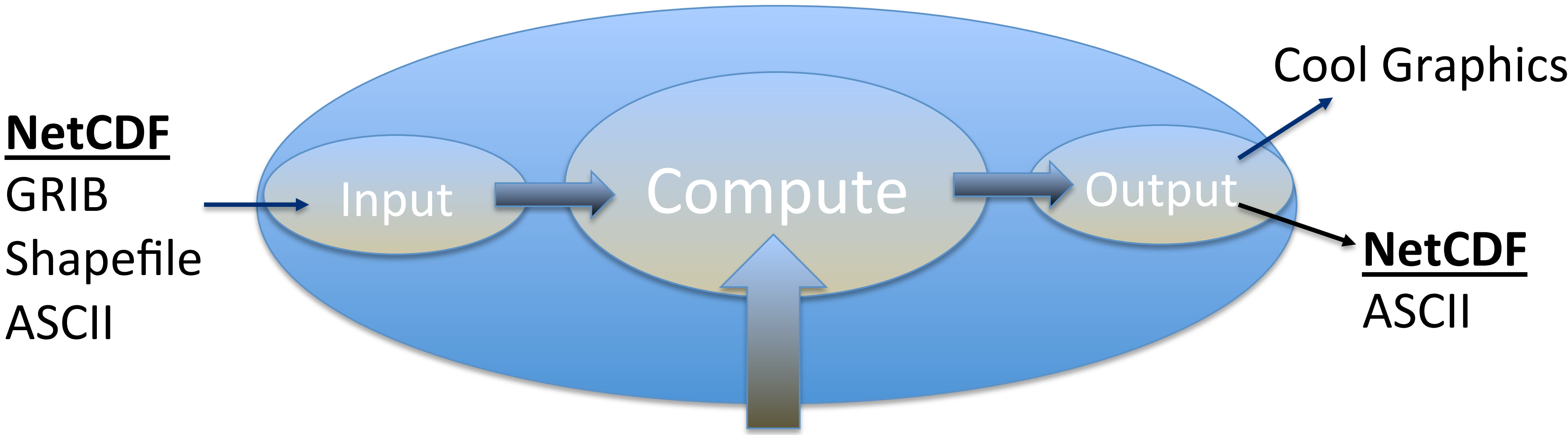




What is NCL

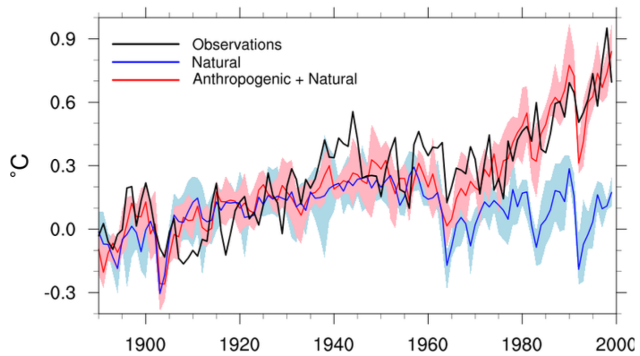
<u>N</u>CAR <u>C</u>ommand <u>L</u>anguage (NCL)

An Integrated Data Processing Environment

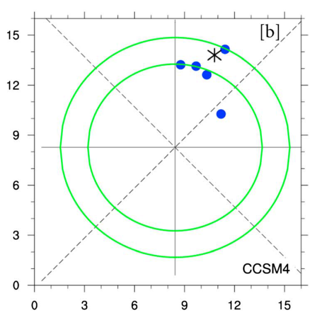


- Science part of data analysis
- Many built-in functions and procedures
- Interpolations from one grid to another

NCL: Why and Where?

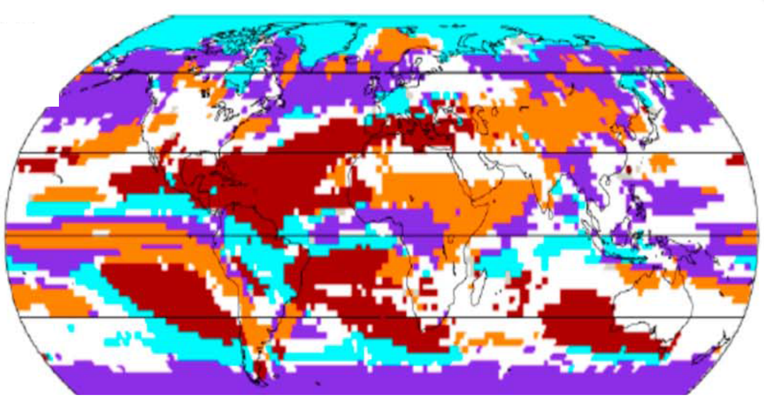


- In-house production
- Support available from experts: ncl-talk@ucar.edu
- Well documented and many application examples are available on the website: http://www.ncl.ucar.edu/
- NCL workshop/training is also offered (http://www.ncl.ucar.edu/Training/Workshops/)



Copyright (C) 1995-2015 - All Rights Reserved
University Corporation for Atmospheric Research
NCAR Command Language Version 6.3.0
The use of this software is governed by a License Agreement.
See http://www.ncl.ucar.edu/ for more details.
Ncl 0>

Actively updated and supported



- All details are here http://www.ncl.ucar.edu/
- Tour of the website
- You can also install and run NCL on your laptop (see installation instruction)

Free Software

Let's get NCL running

1. Open a secure shell window: Terminal, Cygwin, PuTTY, Mobasterm

2. Log on using your yubikey
   >ssh –XY <username>@yellowstone.ucar.edu

3. Log on to geyser from your Yellowstone login node
   >bsub -Is -q geyser -W 4:00 -n 1 -P UCGD0002 xterm &

When you see the token response prompt, enter your pin and then touch the yubikey button.

4. Make a data analysis directory
   >mkdir data_anal
   go to the new directory
   >cd data_anal
   Copy these scripts from /glade/p/cesm/lmwg/CLM2016_tutorial_space/Day3 into the directory you created:
   CLM_TUTORIAL16_NCL1.ncl
   CLM_TUTORIAL16_NCL2_Time_Series.ncl
   CLM_TUTORIAL16_NCL3_monthly_climo_centUSA.ncl

5. Load NCL module
   > module load ncl

6. Run NCL
   >ncl

```
Copyright (C) 1995-2015 - All Rights Reserved
University Corporation for Atmospheric Research
NCAR Command Language Version 6.3.0
The use of this software is governed by a License Agreement.
See http://www.ncl.ucar.edu/ for more details.
Ncl 0>
```

Let’s open a file and read a variable

Exercise 1

<YOU ARE STILL IN NCL, IF NOT THEN JUST TYPE NCL ON YOUR SCREEN>

```
> modInDir1 = "/glade/scratch/dll/CLMTutorial2016_DataForAnalysis/I1850CLM50_001”
> f0 = addfile(modInDir1+"/I1850CLM50_001.clm2.h0.0001-01.nc", "r")
```

- `addfile` → NCL function
- `modInDir1` → File location
- `"/I1850CLM50_001.clm2.h0.0001-01.nc"` → File Name
- `"r"` → Read only
- `f0` → **Local** reference for file access

```
>print(f0)
```

- It will print file header or metadata only (different from ncdump command)
- CLM history file header file is too long, too many variables
- It is a good idea to know in advance which variable you want to work on
- Use “q” to exit the print command

```
>var0 = f0->TSA
```

← Variable name in the file that you should know in advance

- `f0->` → ncl way of accessing a variable for a file which is locally added/available
- `var0` → arbitrary **local** variable name – that you will use in your ncl program

```
>printVarSummary(var0)
```

← Most dependable friend, anytime your program crashes first type this command

- `printVarSummary` → NCL procedure
- `var0` → **local** variable

Function and procedure names are case sensitive

Repeat Exercise 1 in a smarter

NCL script

- Use your favorite text editor, and open a new file
- Write a bunch of ncl commands in a text file
- Give the text file an extension of '.ncl'
- Run ncl script as follows: >ncl your_script_name.ncl

Go to your data_anal directory and open the first ncl script

> nedit CLM_TUTORIAL16_NCL1.ncl

- a long script, will walk through it together
- the script is segmented using 'exit'
- ; for commenting any line (;exit)
- If you want to run full script comment out all of the 'exit'

CLM_TUTORIAL16_NCL1.ncl

Add several files and concatenate it

flis1 = systemfunc("ls "+modInDir1+"/"+caseName1+".clm2.h0.*.nc")

- uses 'systsemfunc' to make a file list

f1 = addfiles(flis1, "r")

- uses 'addfiles' to add several files

ListSetType(f1, "cat")

- Uses 'ListSetType' to concatenate all files along the time dimension

Get the concatenated variable as a local variable

var1 = f1[:]->TSA

access variables from several files

Now run printVarSummary and see the difference

```
Variable: var0
Type: float
Total Size: 55296 bytes
            13824 values
Number of Dimensions: 3
Dimensions and sizes:   [time | 1] x [lat | 96] x [lon | 144]
………………………………………
…………………………
```

earlier

```
Variable: var1
Type: float
Total Size: 13271040 bytes
            3317760 values
Number of Dimensions: 3
Dimensions and sizes:   [time | 240] x [lat | 96] x [lon | 144]
………………………………………
…………………………
```

now

Let's do some math now!

NCL can do lot of math for you

- **-** but it is always a good idea to know what you are doing
- let's learn basics before we do the math

Variable: var1
Type: float
Total Size: 13271040 bytes
3317760 values
Number of Dimensions: 3
Dimensions and sizes: [time | 240] x [lat | 96] x [lon | 144]
Coordinates:
time: [31..7300]
lat: [-90..90]
lon: [0..357.5]
Number Of Attributes: 5
long_name : 2m air temperature
units : K
cell_methods : time: mean
_FillValue : 1e+36
missing_value : 1e+36

Do math on the same type
A (float) = B (float) * C (double) **NO**
A (float) = B (float) * C (integer) **YES**
Use type conversion function if necessary

You can access any dimension using '&'
var1&time

You can access any attribute using '@'
var1@units

Algebraic and Logical Operators

Algebraic – supports scalars and array operations

-	Subtraction / Negation
+	Addition / String concatenation
*	Multiplication
/	Divide
%	Modulus (integers only)
>	Greater than selection
<	Less-than selection
#	Matrix multiply
^	Exponentiation

Logical Operators

.le.	less than or equal
.lt.	less than
.ge.	greater than or equal to
.gt.	greater than
.ne.	not equal
.eq.	equal
.and.	and
.not.	not
.or.	or

Array Syntax and Operations

netCDF data: a multidimensional data format var1 (240 X 96 X 144)

Array syntax and operations are essential for efficiency purposes as well as a cleaner code

A given dimension's index is referred by *0 to n-1* where n is length of the given dimension e.g., 0 to239 for time dimension of var1

Array operation must conform: same size and shape

(1 X 96 X 144)

(240 X 96 X 144) → var1 = var0 **NO**

(240 X 96 X 144) → var2 = var1 **YES**

Scalars automatically conform to all array sizes

var1 = var1-273.15 **YES**

To change unit from Kelvin to degree Celsius

Arrays Subscripting

Standard subscripting – uses index of the dimension

```
var1_ann = var1(0:19, :, :)
```

A new variable

Time dimension -> 0:19

All lat

All lon

Named dimension – useful when you want to reorder dimension for some reason

```
var1_ann_re = var1(lat|:, lon|:, time|0:19)
```

Coordinate subscripting – useful for regional analsysis

```
var1_ann_reg = var1(:, {15.0:50.0}, {236.0:293.0})
```

It has to be in { }

You use actual coordinate values. NCL will figure it out which index it wants to bring in

Just few more basics!

Always create a new variable with metadata in it

Create a new variable as a sub-set or super-set of the existing variable that has metadata in it.

Sub-setting: var1_ann = var1(0:19, :, ;) … then do mathematics as needed

var3 = var1 copy metadata to the new variable var3
var 3 = (/var1 - 273.15/) assign new values without changing the metadata

Super-set: var4 = conform_dims((/2, 240, 96, 144/), var1, (/1, 2, 3/))
dimension 0 not defined, others dimensions ((/1,2,3/)) are copied from var1
Let us define dimension 0: var4!0 = "ens"
var4&ens = (/1850, 2000/)

Use functions that copies metadata to the new variable

var1_avg = dim_avg_n_Wrap(var1, 0) - will copy metadata to var1_avga
var1_avg_nr = dim_avg_n(var1, 0) - will not copy metadata in var_avg_nr

Do it the hard way, i.e. create a new variable and define and assign values to each dimension individually (not recommended)

Let NCL do some math for you!

Many many mathematical and earth science functions

Earth Science:
- Climatology
- CESM
- Crop
- Heat-stress
- Date
- Drought
- Lat/lon functions
- Metadata/missing values
- Meteorology
- Oceanography
- RIP functions
- WRF functions

Math and statistics:
- General applied math
- Bootstrap
- Cumulative distribution functions
- Empirical orthogonal functions
- ESMF regridding
- Extreme values
- Heat stress
- Interpolation
- Ngmath routines
- Random number generators
- Regridding
- Singular value decomposition
- Spherical harmonics
- Statistics

Climatology

calcDayAnomTLL	Calculates daily anomalies from a daily data climatology.
calcMonAnomLLLT	Calculates monthly anomalies by subtracting the long term mean from each point (lev,lat,lon,time version)
calcMonAnomLLT	Calculates monthly anomalies by subtracting the long term mean from each point (lat,lon,time version)
calcMonAnomTLL	Calculates monthly anomalies by subtracting the long term mean from each point (time,lat,lon version)
calcMonAnomTLLL	Calculates monthly anomalies by subtracting the long term mean from each point: (time,lev,lat,lon) version.
clmDayTLL	Calculates long term daily means (daily climatology) from daily data.
clmDayTLLL	Calculates long term daily means (daily climatology) from daily data.
clmMon2clmDay	Create a daily climatology from a monthly climatology.
clmMonLLLT	Calculates long term monthly means (monthly climatology) from monthly data: (lev,lat,lon,time) version.

Use of climatology function

```
var1_mon_mean = clmMonTLL(var1)
```

- this function (clmMonTLL) creates monthly climatology
- read about this function on ncl website
- copies the metadata
- depends on the order the order of the dimensions in input data (var1: time X lat X lon); if other please re-order it or use other function

```
var1_mon_std = stdMonTLL(var1)
```

- this function (stdMonTLL) calculates monthly standard deviation

Calculation for annual standard deviations

Calculate standard deviation of annual mean time series

- no direct ncl function
- let's do it by going through the basics
- first make annual time series then calculate standard deviation

DO Loop in NCL

optional

```
do n = start, end [, stride]
        statements
end do
```

- Useful but may not be efficient; try to minimize use of loop
- Use array arithmetic and/or built-in functions if available

Use mean and standard deviation functions to calculate it

- see CLM_TUTORIAL16_NCL1.ncl

Cool plots!

Three step process

1. Open a workstation

```
wks = gsn_open_wks("x11", "climatology_plot1")
```

use x11 until you finalize the plot
When finalized then you can use pdf, png, ps or others

2. Bring in resources needed

```
res = True
res@mpOutlineBoundarySets = "Geophysical”
.............
```

3. Draw the plot

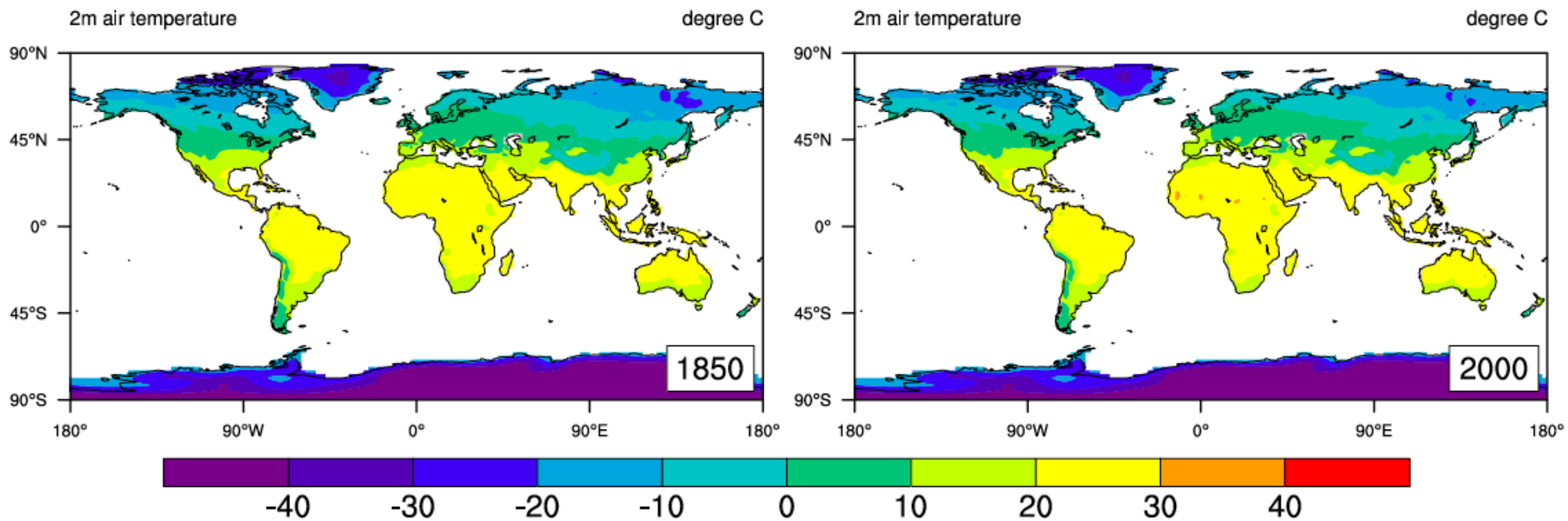
```
plot = new(2, graphic)
plot(0) = gsn_csm_contour_map(wks, var1_ann_avg, res)
```

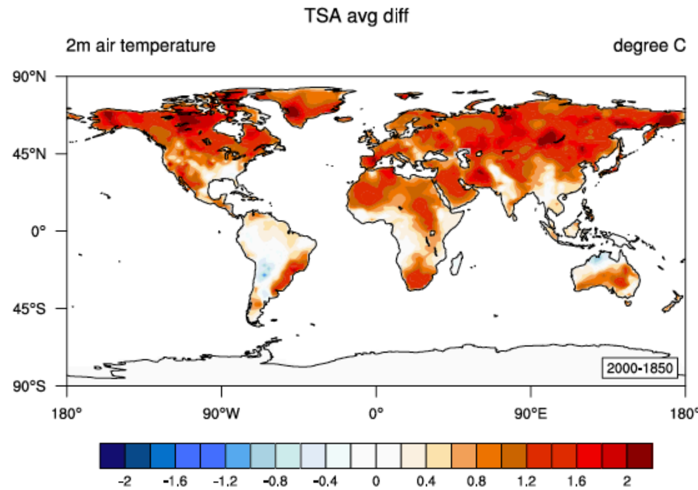
workstation — Two dimensional variable with lat, lon — resources

```
>ncl CLM_TUTORIAL16_NCL1.ncl
```

Cool plots!

> ncl CLM_TUTORIAL16_NCL1.ncl





Do you see global warming in the difference plot here!

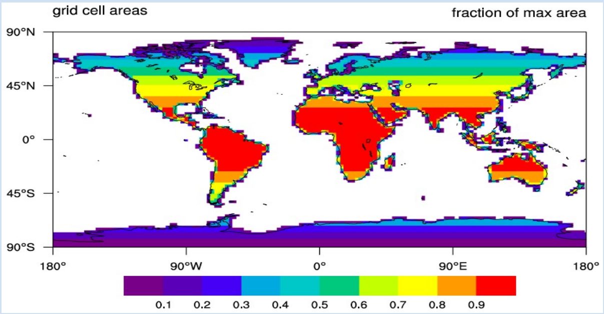
Time series plot

> ncl CLM_TUTORIAL16_NCL2_Time_Series.ncl

Just few extra steps!

```
ar = f0->area
lf = f0->landfrac
arwt = ar
arwt = ar*lf
```

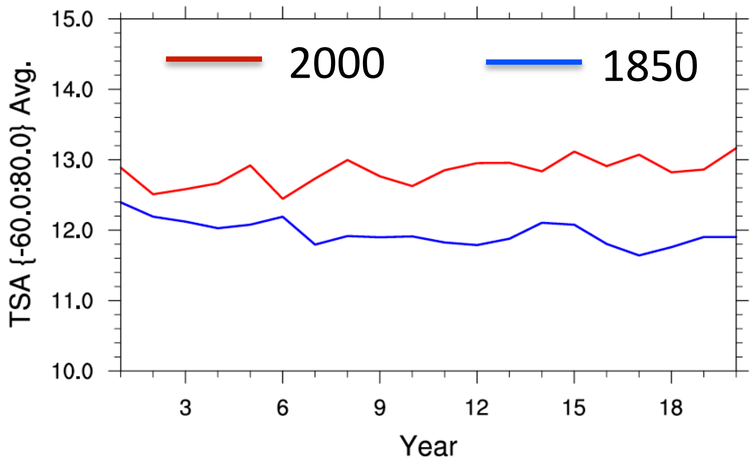
calculate area weight
This is required; particularly when you are doing global average



Calculate area weighted time series

```
P1 = new((/2, nyr/), "float", -99.0)
P1(0, :) = wgt_areaave2(var1_ann(:, {-60.0:80.0}, :), arwt({-60.0:80.0}, :), 0)
P1(1, :) = wgt_areaave2(var2_ann(:, {-60.0:80.0}, :), arwt({-60.0:80.0}, :), 0)
```

See XY Plot technique/ examples on ncl website



Uncertainty shading with transparency

> ncl CLM_TUTORIAL16_NCL3_monthly_climo_centUSA.ncl

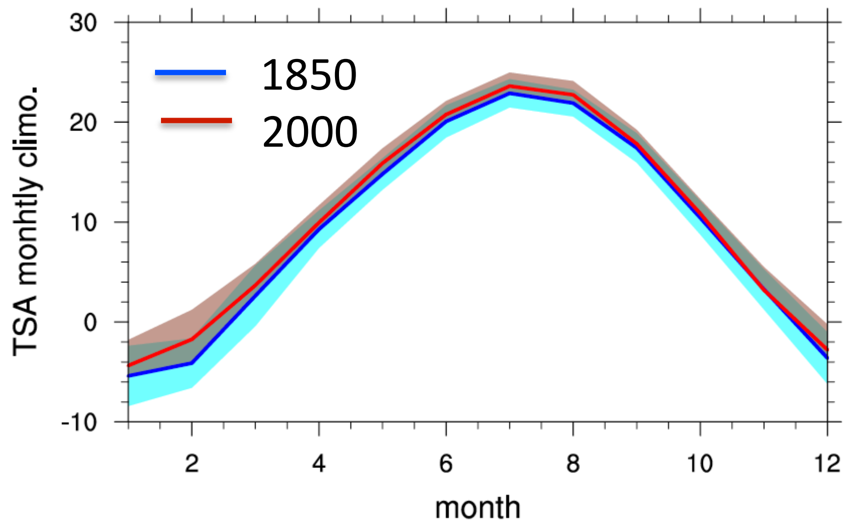
Shading and transparency stuffs in NCL

```
delete(res@xyLineColors)
res@xyLineColor = -1
res@gsnXYFillColors = (/"cyan1"/)
res@gsnXYFillOpacities = (/0.3/)
plotF1 = gsn_csm_xy(wks, time, ncurves(0:1, :), res)
overlay(plot, plotF1)
```

- delete(res@xyLineColors), res@xyLineColor = -1 ← For shading we do not need line
- res@gsnXYFillColors ← Shading color of your choice
- res@gsnXYFillOpacities ← transparency is set here
- plotF1 = gsn_csm_xy(...) ← Shading plot
- overlay(plot, plotF1) ← Overlay to the original XY plot

Monthly climatology of 2m air temperature in central USA (30N-50N, and 250E to 270E) in 1850 (blue) and 2000 (red).

Shading shows 1 standard deviation of inter-annual variability



Tired of writing the Matlab/R/NCL scripts!!
Help is on the way .....

Today's Goal

① Where are model outputs (Keith)?

② Understanding the data (netCDF, ncview, ncdump) (Keith)

③ Analysis in R (Quinn)

④ Introduction to NCL (**N**CAR **C**ommand **L**anguage) (Sanjiv)

⑤ **The Land Diagnostics Package and ILAMB (Keith & Sheri)**

Land Diagnostics Package

- The land diagnostics package post-processes monthly-average land history files into monthly, seasonal, and annual climatologies and means (These files alone can be useful for your own analysis.) and produces a series of plots using NCL displayed on a web page that gives you a comprehensive first-look at your simulation(s).

- The package used to be standalone, but will now be integrated into the CESM postprocessing system available with CESM2.0

http://www.cesm.ucar.edu/models/cesm1.2/clm/clm_diagpackage.html

Land Diagnostics Package

I2000CLM50_001_11_20 and I1850CLM50_001_11_20

National Center for Atmospheric Research
Boulder, Colorado

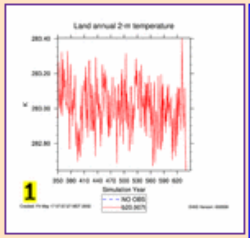
LND_DIAG Diagnostics Plots Source: /glade/p/cesm/lmwg/diag/lnd_diag4.2

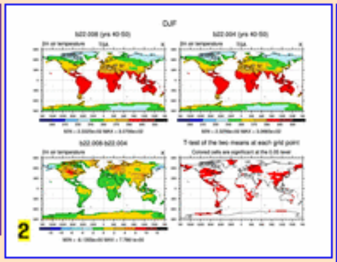
Set Description

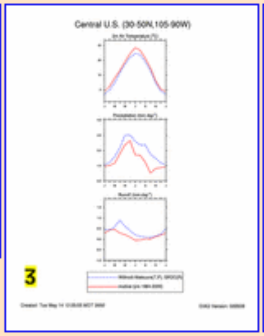
1 Line plots of annual trends in energy balance, soil water/ice and temperature, runoff, snow water/ice, photosynthesis
2 Horizontal contour plots of DJF, MAM, JJA, SON, and ANN means
3 Line plots of monthly climatology: regional air temperature, precipitation, runoff, snow depth, radiative fluxes, and turbulent fluxes
4 *(Inactive)* Vertical profiles at selected land raobs stations
5 Tables of annual means
6 Line plots of annual trends in regional soil water/ice and temperature, runoff, snow water/ice, photosynthesis
7 Line plots, tables, and maps of RTM river flow and discharge to oceans
8 *(Inactive)* Line and contour plots of Ocean/Land/Atmosphere CO2 exchange
9 Contour plots and statistics for precipitation and temperature. Statistics include DJF, JJA, and ANN biases, and RMSE, correlation

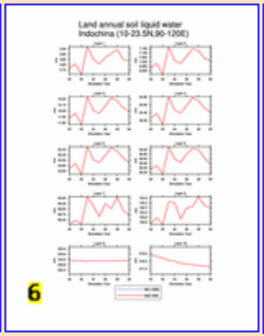
---

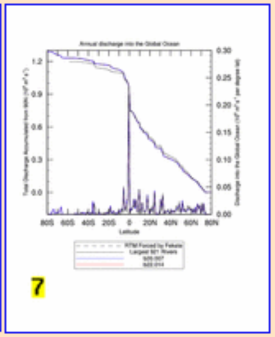
Click on Plot Type











Land Diagnostics Package

- For this tutorial, the package has been run to compare two of the tutorial simulations (I2000CLM50_001 and I1850CLM50_001) (this is called a model to model comparison). Netcdf files are created that contain the seasonal climatologies (*_climo.nc) and means (*_means.nc), and annual trends (*_ANN_ALL.nc). The netcdf files for the tutorial simulations can be found at:

/glade/p/cesm/lmwg/CLM2016_tutorial_space/Day3/Diag/OUTPUT/Tcase2_prefix
/glade/p/cesm/lmwg/CLM2016_tutorial_space/Day3/Diag/OUTPUT/Tcase1_prefix

Where:

Tcase2_prefix: I2000CLM50_001_11-20
Tcase1_prefix: I1850CLM50_001_11-20

LMWG Diagnostics Package

- An html file pointing to plots from the land diagnostics package for the tutorial simulations can be found at:

/glade/p/cesm/lmwg/CLM2016_tutorial_space/Day3/Diag/OUTPUT/Tcase2_prefix /Tcase2_prefix-Tcase1_prefix/setsIndex.html

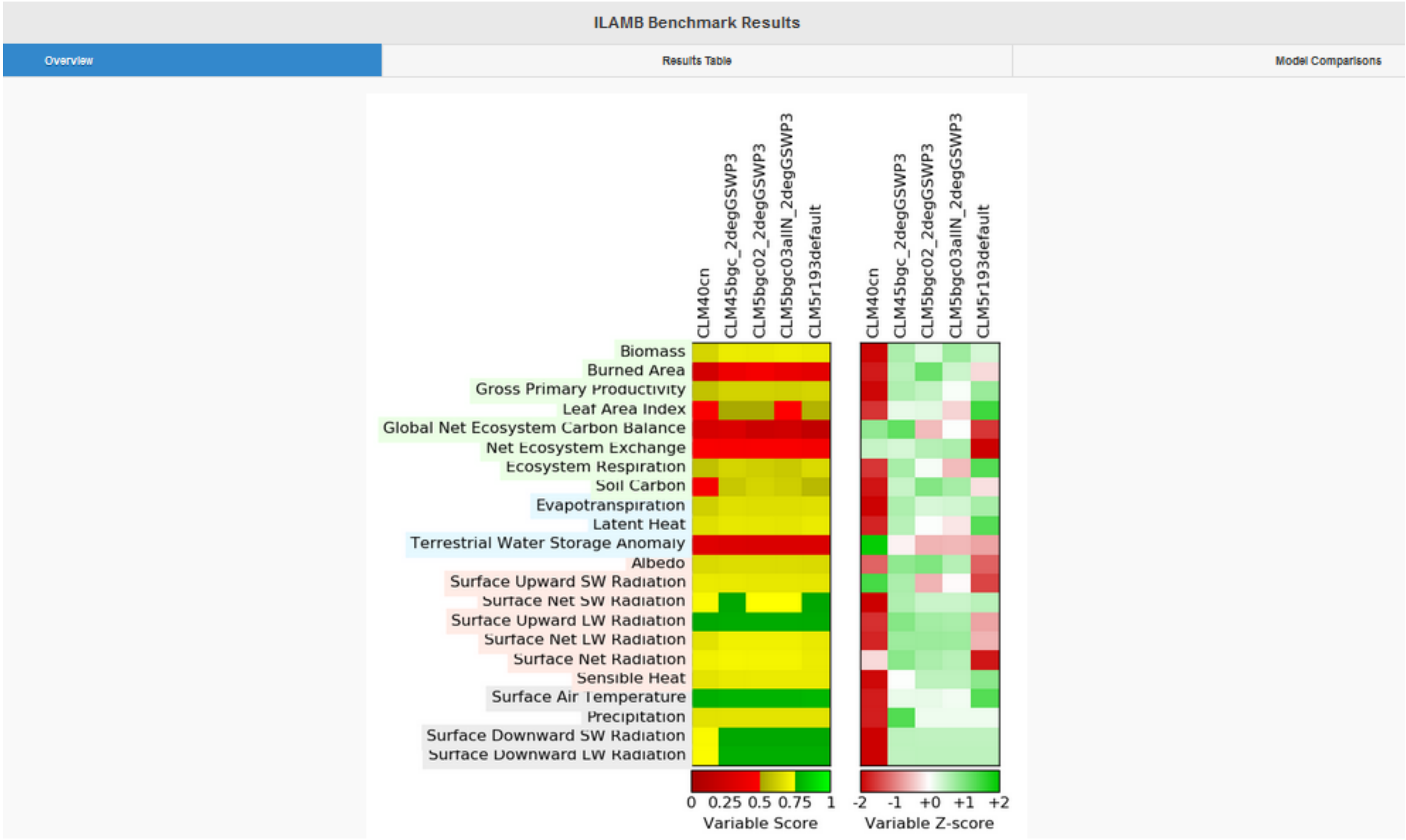
Tcase2_prefix: I2000CLM50_001_11-20
Tcase1_prefix: I1850CLM50_001_11-20

- You can view this by 'cd'-ing to the directory above, firing up a browser, and pointing to the setsIndex.html file

firefox &

In firefox: File->Open File...->setsIndex.html

ILAMB



https://bitbucket.org/ncollier/ilamb

ILAMB

- An html file pointing to output from ILAMB for various CLM4.0, CLM4.5 and CLM5 simulations can be found at:

/glade/p/cesm/lmwg/CLM2016_tutorial_space/Day3/Diag/OUTPUT/ILAMB/index.html

- You can view this by cd-ing to the directory above, firing up a browser, and pointing to the index.html file

firefox &

In firefox: File->Open File...->index.html

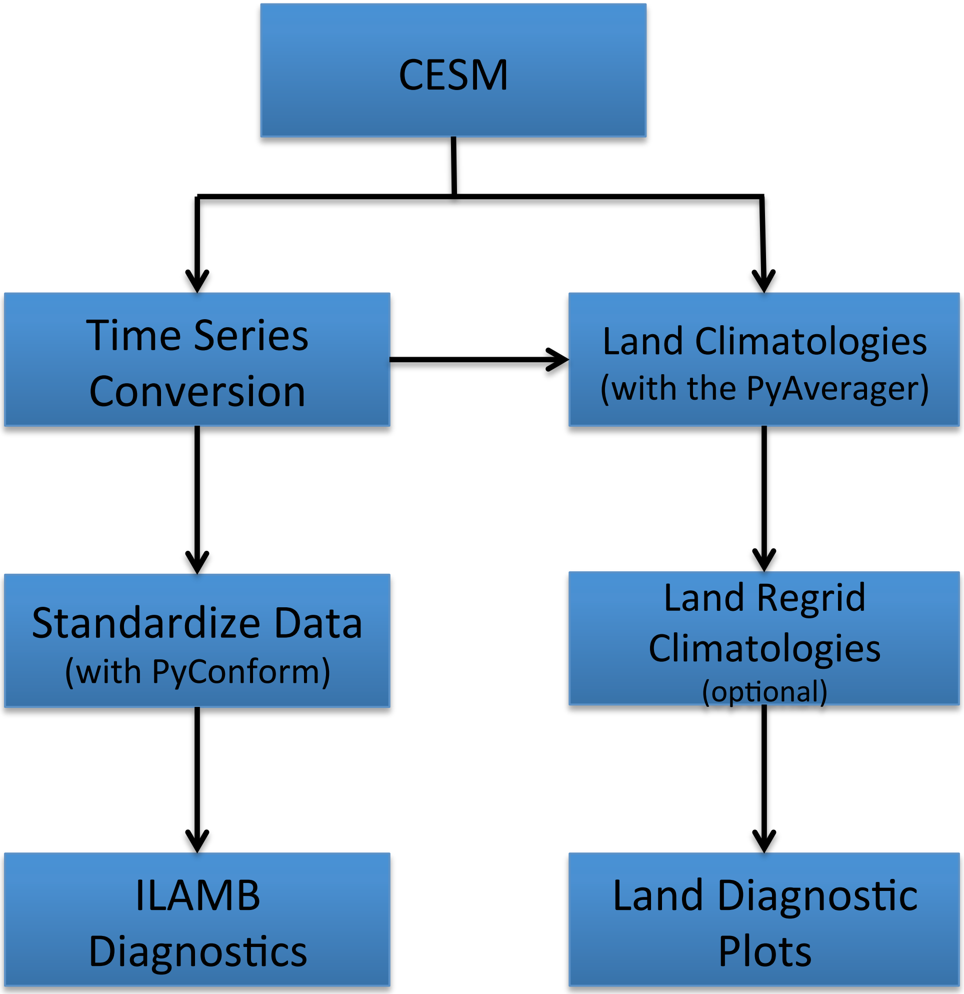
How to Run the Land Diagnostics Within the New Framework

For Further assistance, you can contact:

Sheri Mickelson
mickelso at ucar.edu

Alice Bertini
aliceb at ucar.edu

Land Diagnostic Workflow



#1 How to Install the Code

After you download the CESM code:

1. 'cd' into CESM's top level directory

   This is where you'll find the cime and components directories

2. Get the post processing code

   Follow the steps outlined in the github wiki documentation

   https://github.com/NCAR/CESM_postprocessing/wiki

1. 'cd' into the post processing directory and run:

   ./create_python_env -machine <cesm_supported_machine>

   ./create_ilamb_env

   These two commands create Python Virtual Environments

   - Similar to modules, these install all needed packages for you, but they stay out of your environment until you say load them and they go away when you unload them
   - The scripts within the post-processing suite automatically handle the load/unload for the users

Or on Yellowstone you can use the system copy:

For tcsh:
setenv POSTPROCESS_PATH /glade/p/cesm/postprocessing alias
cesm_pp_activate 'source $POSTPROCESS_PATH/cesm-env2/bin/activate.csh’

For bash:
export POSTPROCESS_PATH=/glade/p/cesm/postprocessing alias
cesm_pp_activate='. $POSTPROCESS_PATH/cesm-env2/bin/activate.sh’

#2 How to create a post processing case

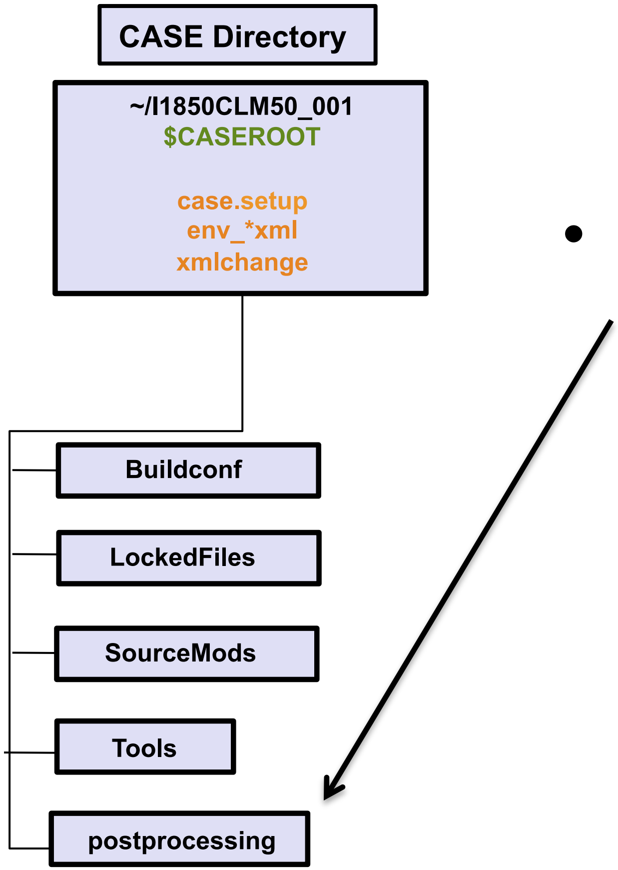
First you'll need to run …

1. If running with the copy on yellowstone, run 'cesm_pp_activate'

2. ./create_postprocess -caseroot CASEROOT
   - $CASEROOT can be an existing CESM case directory or a new directory you want the script to create

3. If running with the copy on yellowstone, run 'deactivate'

#3 How to setup a post processing case



- You'll need to cd into $CASEROOT/postprocess and edit the XML files
  - You can modify the values by running './pp_config -set key=value' or by editing the files by hand

Edit $CASENAME/postprocess/env_postprocess.xml

Most common variables (keys) to edit for land diagnostics:

LND_GRID (some valid values: "1.9x2.5”, "0.9x1.25”)
GENERATE_TIMESERIES
GENERATE_AVGS_LND
GENERATE_REGRID_LND
GENERATE_DIAGS_LND
GENERATE_ILAMB

Edit $CASENAME/postprocess/env_diags_lnd.xml

Some of the most common variables (keys) to edit:

LNDDIAG_DIAGOBSROOT
LNDDIAG_SOURCE_1(2)
LNDDIAG_OUTPUT_ROOT_PATH
LNDDIAG_MODEL_VS_OBS
LNDDIAG_MODEL_VS_MODEL
LNDDIAG_COMPUTE_CLIMO_CASE1(2)
LNDDIAG_caseid_1(2)
LNDDIAG_CASE1(2)_TIMESERIES
LNDDIAG_weightAvg
LNDDIAG_meansFlag
LNDDIAG_trends_lnd_1(2)
LNDDIAG_climo_lnd_1(2)
LNDDIAG_trends_atm_1(2)
LNDDIAG_climo_atm_1(2)
LNDDIAG_rmodel_1(2)
LNDDIAG_rtm_1(2)
LNDDIAG_trends_rtm_1(2)
LNDDIAG_climo_rtm_1(2)
LNDDIAG_clim_first_yr_1(2)
LNDDIAG_clim_num_yrs_1(2)
LNDDIAG_trends_first_yr_1(2)
LNDDIAG_trends_num_yrs_1(2)
LNDDIAG_set_(1-9)

Edit $CASENAME/postprocess/env_ilamb.xml

Some of the most common variables (keys) to edit:
(Still in development and might change slightly)

ILAMB_ROOT
NETCDF_FORMAT
BUILD_DIR
OBS_DATA__1*
FILE_PREFIX__1*
YEARS__1*
CASE_PATH__1*
TIMESERIES_OUTPUT_PATH__1*
MIP_OUTPUT_PATH__1*
EXPERIMENT__1*
MODEL__1*
CONVERT_TO_TS__1*
CONVERT_TO_CMIP__1*
TIME_VARIANT_VARS__1*

* Variable entry can be copied and the digit can be increased to add more data

Edit $CASENAME/postprocess/env_ilamb.xml

Some of the most common variables (keys) to edit:
(Still in development and might change slightly)

ILAMB_ROOT
NETCDF_FORMAT
BUILD_DIR
OBS_DATA__1*
FILE_PREFIX__1*
YEARS__1*
CASE_PATH__1*
TIMESERIES_OUTPUT_PATH__1*
MIP_OUTPUT_PATH__1*
EXPERIMENT__1*
MODEL__1*
CONVERT_TO_TS__1*
CONVERT_TO_CMIP__1*
TIME_VARIANT_VARS__1*

* Variable entry can be copied and the digit can be increased to add more data

#4 How to Run the Packages

Submit the run scripts in the $CASEROOT/postprocess directory

On yellowstone:

- bsub < timeseries
- bsub < lnd_averages
- bsub < lnd_regrid
- bsub < lnd_diagnostics
- bsub < ilamb (this will create time series, format the data, and create plots)
- ./copy_html (copy plots and webpages to a web server)

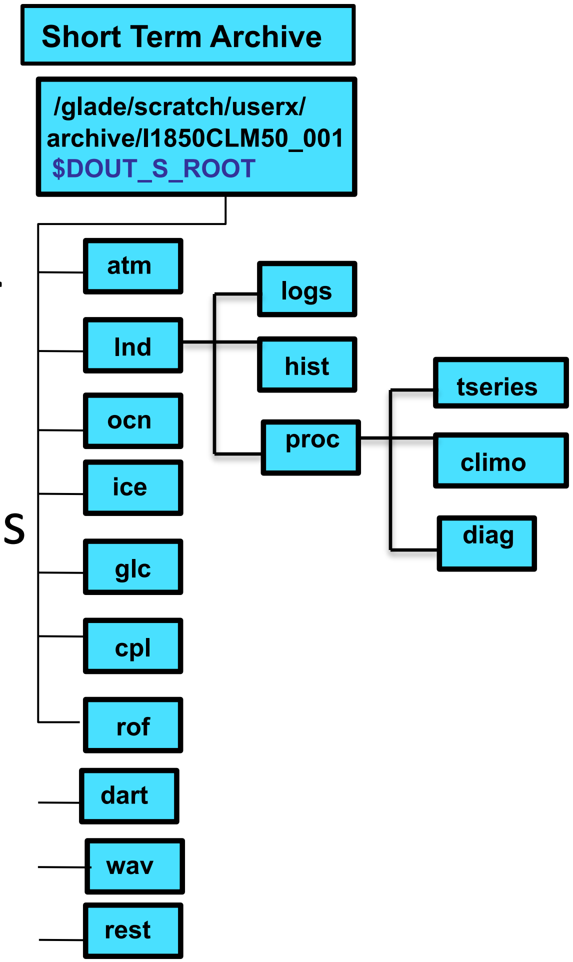
The order you submit these scripts matter to ensure the correct data exists.
Do not submit the next script until the previous script runs successfully.

Review $CASEROOT/postprocess/logs for errors

#5 Where does everything go?

The post processing XML variables define where the output files are placed. In general, all the post processing files are added to subdirectories under the short term archive location ($DOUT_S_ROOT) in:

lnd/proc/tseries – single variable time series files
lnd/proc/climo – climatologies
lnd/proc/diag – diagnostics plots and web pages



Might be needed

Survey of the data we want to analyze

> cd /glade/scratch/dll/CLMTutorial2016_DataForAnalysis/I1850CLM50_001

- Change directory to where output files have been stored

> ls

- See what's in there

> du -h

- See the size of the data

I1850CLM50_001.clm2.h0.0020-12.nc ← netCDF file type

case name | monthly output | year | month

20-years of monthly data: 17GB => 150-years: 1.3TB

Several GB to TB data
-difficult to visualize

→

One Figure
-easy to interpret