

CALPUFF (Non-USEPA) 2021

For Windows/Mac/Linux

- **NOTE:** This updated UBC guide is for installing the latest (at the time of writing) stable research versions of CALPUFF (V7.2.1), CALMET (V6.5.0), CALPOST (V7.1.0), and CALWRF (V2.0.3)
 - *****It is strongly recommended that you use Intel Fortran for compilation (ifort), and a Linux OS if possible*****
- If you would like instructions on how to install the USEPA version of CALPUFF (V5.8.5) and CALMET (V5.8.5), please refer to the instructions here:
https://www.eoas.ubc.ca/courses/atsc507/ADM/calpuff/calpuff_install-v2.pdf
- UBC is not affiliated with Exponent and CALPUFF; these instructions are meant for pedagogical purposes for the atmospheric dispersion modelling course **ATSC 595D**
- If you are reading these instructions outside of ATSC 595D, please note that these instructions may not apply to your specific systems, and UBC is under no obligation to provide support
 - Please contact Exponent instead:
https://www.exponent.com/services/practices/environmental-sciences/health-sciences/capabilities/atmospheric-sciences/calpuff-training-by-the-developers-of-the-calpuff_/?serviceId=2cf2375a-3964-4bc9-b372-27e654241da1&loadAllByPageSize=true&knowledgePageSize=7&knowledgePageNum=0&newseventPageSize=7&newseventPageNum=0&professionalsPageNum=1

- **Bolded entries are individual commands to be placed on the command line; they should be written and entered as a single line in the terminal**
- Main CALPUFF site:
<http://www.src.com/calpuff/download/download.htm#DOCUMENTATION>
- Full CALPUFF (V6) user's guide:
http://www.src.com/calpuff/download/CALPUFF_Version6_UserInstructions.pdf
 - V7 addendum:
http://www.src.com/calpuff/download/CALPUFF_v7_UserGuide_Addendum.pdf

Install Intel Fortran (Mac or Linux)

- Strongly recommend using Intel oneAPI suite to install CALPUFF; CALPUFF code does not compile or behave properly with GNU Fortran
- First, we need to download Intel oneAPI Base Toolkit
 - <https://www.intel.com/content/www/us/en/developer/tools/oneapi/base-toolkit-download.html>
 - Specify your OS (macos or linux); choose Online & Offline (recommended); choose Online installer
 - Download, and follow the instructions; open the bootstrapper

Get the Intel® oneAPI Base Toolkit

No Transistor Left Behind™ The Smart Path to Accelerated Computing without the Economic and Technical Models

Select options below to download

Operating System: Select operating system
macOS

Distribution: Select distribution
Online & Offline (recommended)

Installer Type: Select installer
Online

Online Installation

- Features customizable installation
- Downloads a small installer file
- Requires an internet connection on the host machine
- Downloads components at the same time as the installation
- Provides an option to create a local installer

What's Included in the Intel® oneAPI Base Toolkit for macOS*

Download size: 29.4 MB
Version: 2021.4
Date: September 28, 2021

Download

- Don't install everything (full Linux suite is like 24 GB once you have everything installed)! Choose Custom installation

intel. Intel® oneAPI Base Toolkit, v. 2021.4

Summary

What's Included

- Intel® oneAPI Threading Building Blocks (2021.4.0)
- Intel® Integrated Performance Primitives (2021.4.0)
- Intel® Integrated Performance Primitives Cryptography (2021.4.0)
- Intel® oneAPI Data Analytics Library (2021.4.0)
- Intel® oneAPI Math Kernel Library (2021.4.0)
- Intel® oneAPI Deep Neural Network Library (2021.4.0)
- Intel® Distribution for Python® (2021.4.0)
- Intel® Advisor (2021.4.0)
- Intel® VTune(TM) Profiler (2021.7.1)

Installation Location
/opt/intel/oneapi

Required Space
Download size: 310.2 MB
Installation size: 2.1 GB

Select your installation mode

I accept the terms of the license agreement

Recommended Installation
Install everything with commonly used features

Continue

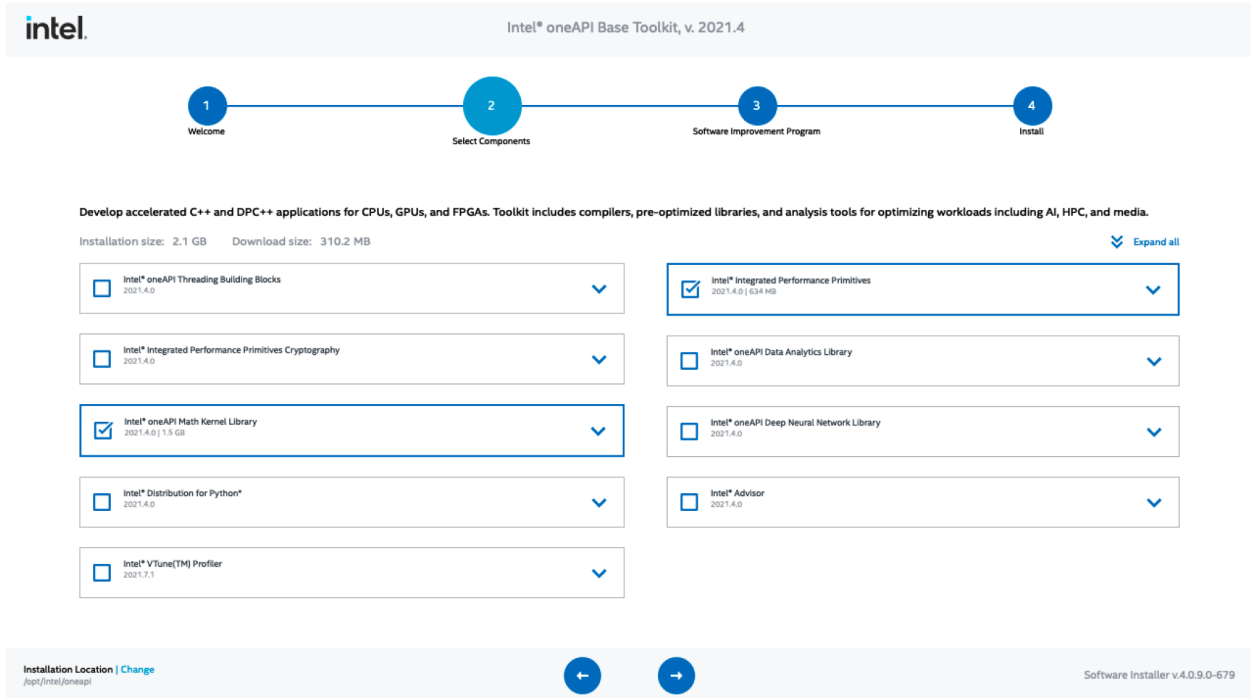
Custom Installation
Configure your installation

Customize

Create a customized offline package for future use

Software Installer v.4.0.9.0-679

- Select only Intel oneAPI Math Kernel Library and Intel Integrated Performance Primitives; deselect everything else, and continue with the install



- Then, install Intel oneAPI HPC Toolkit
 - <https://www.intel.com/content/www/us/en/developer/tools/oneapi/hpc-toolkit-download.html>
 - Specify your OS (macos or linux); choose Online & Offline (recommended); choose Online installer

Get the Intel[®] oneAPI HPC Toolkit

Select options below to download

Operating System: Select operating system
macOS

Distribution: Select distribution
Online & Offline (recommended)

Installer Type: Select installer
Online

Download the Toolkit

Online Installer

- Features customizable installation
- Downloads a small installer file
- Requires an internet connection on the host machine
- Downloads components at the same time as the installation
- Provides an option to create a local installer

What's Included in the Intel[®] oneAPI HPC Toolkit for macOS*

- Intel[®] C++ Compiler Classic
- Intel[®] Fortran Compiler Classic

- If you're on Mac, you only have C/C++ and Fortran to install; go ahead with that
 - If you're on Linux, go to Custom Install, and only select those two; deselect everything else

Intel[®] Software Installer

Intel[®] oneAPI HPC Toolkit, v. 2021.4

Summary

What's Included <ul style="list-style-type: none"> • Intel[®] C++ Compiler Classic (2021.4.0) • Intel[®] Fortran Compiler Classic (2021.4.0) 	Integrate with IDE • Xcode [®] .
Installation Location /opt/intel/oneapi	Required Space Download size: 48 MB Installation size: 247.6 MB

Select your installation mode

I accept the terms of the license agreement

Recommended Installation
Install everything with commonly used features Continue

Custom Installation
Configure your installation Customize

[Create a customized offline package for future use](#)

*Other names and brands may be claimed as the property of others

Software Installer v.4.0.9.0-679

- After that's done, your Intel installation is complete
 - The base directory should be in `/opt/intel/oneapi/`
 - NOT in your home directory
 - The executables (on Mac) should be in `/opt/intel/oneapi/compiler/2021.4.0/mac/bin/intel64/`
 - The image below shows the stuff you should have now; the important one is **ifort** (use this in place of gfortran)

```

codecov*      icc*          icpc.cfg      libcilkrts.5.dylib  profmerge*     xiar*
fortcom*      icc.cfg       ifort*        mpccom*          proforder*     xild*
ipp*          icpc*         ifort.cfg     profdcg*         tselect*       xilibtool*

```

- To set the required Intel executables and libraries to be in your path, you need to source `setvars.sh`
 - **source /opt/intel/oneapi/setvars.sh**
 - ****NOTE:** You'll need to do this every time you open a new terminal window, unless you put this command in your `.bashrc` or `.profile` or `.bash_profile` *******
- If you run **which ifort**, you should see the full path to the compiler
 - `/opt/intel/oneapi/compiler/2021.4.0/mac/bin/intel64/ifort`

Install CALPUFF

- Make a new directory in home, that will contain all CALPUFF-relevant software
 - `mkdir ~/calpuff`
 - `cd ~/calpuff`
- Download CALPUFF v7.2.1, unzip, and head into the unzipped folder
 - `wget`
http://www.src.com/calpuff/download/Mod7_Files/CALPUFF_V7.2.1_L150618.zip
 - `unzip CALPUFF_V7.2.1_L150618.zip`
 - `cd CALPUFF_v7.2.1_L150618`
 - **NOTE:** On Linux clusters, you may need to convert everything to lower case (applies to later unzips as well)
 - `unzip -LL CALPUFF_V7.2.1_L150618.zip`
 - `cd calpuff_v7.2.1_1150618`
- For Windows users, you should see a pre-compiled executable:
`./calpuff_v7.2.1.exe`
 - Sooooo...you should be good to go!
- For Mac and Linux users (using ifort → make sure you've sourced setvar.sh
---> `source /opt/intel/oneapi/setvars.sh`)
 - Compile (all on one line, of course)
 - `ifort -o calpuff.exe modules.for CALPUFF.FOR`

- If you run **ls** you should see **calpuff.exe** (this is the version you installed, and is different than the pre-compiled Windows executable `calpuff_v7.2.1.exe`)
- Run `calpuff.exe`
 - **`./calpuff.exe`**
 - (Windows: **`calpuff_v7.2.1.exe`**)
 - You should see the following error:

```
ERROR opening Control File
  File Name: calpuff.inp
  File Unit:          1
Problem reported from SETUP

The file may not exist in this location
Check the spelling of the name, and the location
```

Run CALPUFF

- The V7 source folders are very stripped down, and don't contain demo data files or INPs; V5 demos will NOT work, we have to download and unzip V6 files instead
- First, let's make a new tutorial subdirectory, and soft-link in a copy of your executable
 - **`mkdir tutorial`**
 - **`cd tutorial`**
 - **`ln -s ../calpuff.exe .`** (or just copy over the Windows executable)
- Come back to the main calpuff directory
 - **`cd ../..`**

- **pwd**
- You should be in ~/calpuff (or whatever your full path is)

- Make a new version 6 directory, to keep things clean
 - **mkdir CALPUFF_v6**
 - **cd CALPUFF_v6**

- Grab CALPUFF v6.42, unzip, and go into the new CALPUFF directory
 - **wget**
http://www.src.com/calpuff/download/Mod6_Files/CALPUFF_v6.42.zip
 - **unzip CALPUFF_v6.42.zip**
 - **cd CALPUFF**
 - **ls**

- You'll see a bunch of data files, an INP file, and some output files like CALPUFF.LST
 - We need CALPUFF.INP, OZONE.DAT, and calmet.dat to make our test run, so let's copy them to our tutorial directory
 - **cp CALPUFF.INP**
~/calpuff/CALPUFF_v7.2.1_L150618/tutorial/
 - **cp OZONE.DAT**
~/calpuff/CALPUFF_v7.2.1_L150618/tutorial/
 - **cp calmet.dat**
~/calpuff/CALPUFF_v7.2.1_L150618/tutorial/
 - **cd ~/calpuff/CALPUFF_v7.2.1_L150618/tutorial/**

- You should now be back in the tutorial directory, under the Version 7 build that you compiled

- Run calpuff, specifying the INP file
 - **./calpuff.exe CALPUFF.INP**
 - You should see:

```

SETUP PHASE

***** WARNING *****
Potential problem in Control file ---
Search for QA ALERT in List file ---

COMPUTATIONAL PHASE
--- Advection Step Starting:
--- YYYYJJJHH SSSS # Old # Emitted
+ 199000904 0 0 11
+ 199000905 0 11 11
+ 199000906 0 22 12
+ 199000907 0 34 12

TERMINATION PHASE

```

- Viewable text output is in CALPUFF.LST; doubles as a log file
- You'll also see other output like CALPUFF.CON and CALPUFF.DRY (you'll process CALPUFF.CON into a viewable format with CALPOST)

Install CALMET

- Analogous to AERMET from AERMOD
 - Processes meteorology files (obs + model) into a format useable by CALPUFF
 - For model data, you'd have to do an additional step to convert them into a readable .DAT format first (see CALWRF later on)
- Go back to your main installation directory
 - `cd ~/calpuff`
- Download, unzip, and enter the new directory
 - `wget`
http://www.src.com/calpuff/download/Mod7_Files/CALMET_v6.5.0_L150223.zip
 - `unzip CALMET_v6.5.0_L150223.zip`

- `cd CALMET_v6.5.0_L150223`
- For Windows users, you should see a pre-compiled executable:
 - `./calmet_v6.5.0.exe`
- For Mac and Linux users, compile CALMET
 - `ifort -o calmet.exe CALMET.FOR`
- If you run the new executable with `./calmet.exe` (Windows: `calmet_v6.5.0.exe`), you should see the following error

```
ERROR opening Control File
  File Name: calmet.inp
  File Unit:          15
Problem reported from SETUP

The file may not exist in this location
Check the spelling of the name, and the location
```

Run CALMET

- Make a new tutorial subdirectory, and soft-link in a copy of your executable
 - `mkdir tutorial`
 - `cd tutorial`
 - `ln -s ../calmet.exe .` (or just copy over the Windows executable)
- Go into the version 6 directory, and download the older version of CALMET
 - `cd ~/calpuff/CALPUFF_v6`
 - `cd CALPUFF_v6`

- **wget**
http://www.src.com/calpuff/download/Mod6_Files/CALMET_v6.4.0.zip
- **unzip CALMET_v6.4.0.zip**
- **cd CALMET**

- Copy over the .INP file, as well as the geographical data file (GEO1KM.DAT), the surface data file (SURF.DAT), and the precip data file (PRECIP.DAT), all to your tutorial directory
 - **cp CALMET.INP**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - **cp GEO1KM.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - **cp SURF.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - **cp PRECIP.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`

- You'll also need to copy in upper-air data (UPPWM.DAT, UPALBR.DAT, UPCHH.DAT), sea-surface data (4007.DAT), and model data (MM4.DAT)
 - **cp UPPWM.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - **cp UPALBR.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - **cp UPCHH.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - **cp 4007.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - **cp MM4.DAT**
`~/calpuff/CALMET_v6.5.0_L150223/tutorial/`

- Go back to the tutorial directory, and run CALMET
 - `cd ~/calpuff/CALMET_v6.5.0_L150223/tutorial/`
 - `./calmet.exe CALMET.INP`
- If all is well, you should see:

```

ENTERING SETUP PHASE
WARNINGS are found in the CONTROL file
Review messages written to the LIST file
ENTERING COMPUTATIONAL PHASE
+Processing Year, Day, Hour, Sec from: 1990  9  4  0 to:1990  9  4 3600
+Processing Year, Day, Hour, Sec from: 1990  9  5  0 to:1990  9  5 3600
+Processing Year, Day, Hour, Sec from: 1990  9  6  0 to:1990  9  6 3600
+Processing Year, Day, Hour, Sec from: 1990  9  7  0 to:1990  9  7 3600
ENTERING TERMINATION PHASE

```

- Text output is in CALMET.LST, but what you'd feed into CALPUFF is CALMET.DAT

Install CALPOST

- Post-processes CALPUFF output (i.e. CALPUFF.CON)
 - Turns CALPUFF.CON into text output
 - Computes averages, summaries, fluxes, etc.
- Go back to your main installation directory
 - `cd ~/calpuff`
- Download, unzip, and enter the new directory
 - `wget`
 - http://www.src.com/calpuff/download/Mod7_Files/CALPOST_v7.1.0_L141010.zip
 - `unzip CALPOST_v7.1.0_L141010`
 - `cd CALPOST_v7.1.0_L141010`
- For Windows users, you should see a pre-compiled executable:
 - `./calpost_v7.1.0.exe`
- For Mac and Linux users, compile CALPOST

- `ifort -o calpost.exe calpost.for`
- If you run the new executable with `./calpost.exe` (Windows: `calpost_v7.1.0.exe`), you should see the following error

```
ERROR: File not found -- see list file
```

Run CALPOST

- Make a new tutorial subdirectory, and soft-link in a copy of your executable
 - `mkdir tutorial`
 - `cd tutorial`
 - `ln -s ../calpost.exe .` (or just copy over the Windows executable)
- Go into the version 6 directory, and download the older version of CALPOST
 - `cd ~/calpuff/CALPUFF_v6`
 - `cd CALPUFF_v6`
 - `wget`
http://www.src.com/calpuff/download/Mod6_Files/CALPOST_v6.292.zip
 - `unzip CALPOST_v6.292.zip`
 - `cd CALPOST`

- Copy over the .INP file, as well as CALPUFF-produced .CON file
 - `cp calpost.inp`
`~/calpuff/CALPOST_v7.1.0_L141010/tutorial/`
 - `cp CALPUFF.CON`
`~/calpuff/CALPOST_v7.1.0_L141010/tutorial/`
- Go back to the tutorial directory, and run CALPOST
 - `cd ~/calpuff/CALPOST_v7.1.0_L141010/tutorial/`
 - `./calpost.exe calpost.inp`
- If successful, you should see:

```

CALPOST Application Completed
Last Period Processed ENDS at:
Year: 1990 Month: 1 Day: 9 Julian day: 9 Hour: 8 Second: 0

End of run -- Clock time: 10:22:46
Date: 28-Oct-2021

Elapsed clock time:      0 (seconds)
CPU time:                0 (seconds)

```

- Output can be found in calpost.lst

Install CALWRF (and NETCDF)

- Converts WRF output (V2 and V3) into .DAT files for use in CALMET
 - Analogous to arw2arl in Hysplit
- Go back to your main installation directory
 - `cd ~/calpuff`
- Before we grab CALWRF and install it (for Mac/Linux), we first have to install netcdf to work with WRF output files; you can use your package

manager, but it's better to install from source, with the same compiler you used for the others and for CALWRF later on

- ****NOTE:** for netcdf on Mac...Intel Fortran may NOT work. If you're struggling...open up a new terminal window, and install netcdf using your usual gcc/gfortran compilers
- Make a new directory, so we can install netcdf in there; grab netcdf-c, unzip it (note: v4.8+ may not work on Mac)
 - **mkdir netcdf**
 - **cd netcdf**
 - **wget**
<https://github.com/Unidata/netcdf-c/archive/refs/tags/v4.7.4.tar.gz>
 - **tar -xvzf v4.7.4.tar.gz**
- Specify your netcdf build directory (place in ~/calpuff); you'll produce ~/calpuff/netcdf/bin, ~/calpuff/netcdf/lib, ~/calpuff/netcdf/include, ~/calpuff/netcdf/share once you've built netcdf
 - **export NETCDF=~/calpuff/netcdf**
- Go into the netcdf-c source directory
 - **cd netcdf-c-4.7.4**
- Specify the compilers you're using
 - **export CC=icc (or gcc)**
 - **export FC=ifort (or gfortran)**
 - **export F90=ifort (or gfortran)**
 - **export F77=ifort (or gfortran)**
- Export CPPFLAGS and LDFLAGS (for netcdf to know later where to look for header files and lib files, respectively)
 - **export CPPFLAGS=-I\$NETCDF/include**
 - **export LDFLAGS=-L\$NETCDF/lib**

- Configure the compilation, specifying the build path with `--prefix`, disabling dap (directory access protocol), and disabling higher netcdf-4 functionality
 - `./configure --prefix=$NETCDF --disable-dap --disable-netcdf4`
- Build (make), move into the build directories (make install), and check that the installation was successful (make check)
 - `make`
 - `make install`
 - `make check`
 - (you should see all greens upon check, i.e. all tests passed)

- Grab netcdf-fortran
 - `cd ~/calpuff/netcdf`
 - `wget`
<https://www.unidata.ucar.edu/downloads/netcdf/ftp/netcdf-fortran-4.5.3.tar.gz>
 - `tar -xvzf netcdf-fortran-4.5.3.tar.gz`
 - `cd netcdf-fortran-4.5.3`
- Specify that we're also using netcdf-c libs
 - `export LIBS=-lnetcdf`
- Configure the compilation
 - `./configure --prefix=$NETCDF`
- Build (make), move into the build directories (make install), and check that the installation was successful (make check)
 - `make`
 - `make install`
 - `make check`

- (you may NOT see successful tests with Intel Fortran, at least on Mac; if so, you'll have to start over the netcdf install [with netcdf-c] using gfortran and gcc)
- Specify that we're also using netcdf-fortran libs too
 - **export LIBS="-lnetcdff -lnetcdf"**
- With netcdf properly installed, we can start working on CALWRF
 - **cd ~/calpuff**
- Download, unzip, and enter the new directory
 - **wget**
http://www.src.com/calpuff/download/Mod7_Files/CALWRF_v2.0.3_L190426.zip
 - **unzip CALWRF_v2.0.3_L190426.zip**
 - **cd CALWRF_v2.0.3_L190426**
- Directory structure's a bit different this time around
- Windows pre-compiled binary is in ./binary_windows/calwrf.exe
- For Mac and Linux: source code is in ./code
 - **cd code**
- Compile (switch out **gfortran** with **ifort** if the latter works)
 - **gfortran \$LDFLAGS \$LIBS \$CPPFLAGS -o calwrf.exe calwrf.f**
- Run
 - **./calwrf.exe**

- Error you should see:

```
Control inp file:calwrf.inp
At line 1399 of file calwrf.f (unit = 13)
Fortran runtime error: Cannot open file 'calwrf.inp': No such file or directory

Error termination. Backtrace:
#0  0x10464a700
#1  0x10464af78
#2  0x10464b982
#3  0x10477ad0e
#4  0x10477b565
#5  0x1043dd9c4
#6  0x1043dfbdf
#7  0x1043f8d14
```

Run CALWRF

- Make a new tutorial subdirectory, and soft-link in a copy of your executable
 - `mkdir tutorial`
 - `cd tutorial`
 - `ln -s ../calwrf.exe .` (or just copy over the Windows executable)
- Copy over the .inp file from two directories up
 - `cp ../../calwrf.inp .`
- Download and unzip the sample WRF output from the CALPUFF page on the course website (if you hadn't downloaded them previously); move them all into your current directory
 - `wget`
https://www.eoas.ubc.ca/courses/atasc507/ADM/calpuff/wrf_mini-outputFiles.zip
 - `unzip wrf_mini-outputFiles.zip`

- `mv wrf_mini-outputFiles/wrfout* .`
- (if you had previously downloaded this zip file, then move or link in the wrfout files: `mv <path_to_wrfouts>/wrfout* .` OR `ln -s <path_to_wrfouts>/wrfout* .`)
- Your tutorial directory should now look like this (don't worry about README and __MACOSX/)

```

README          calwrf.inp          wrfout_d01_2016-02-23_12:00:00 wrfout_d01_2016-02-23_15:00:00
__MACOSX/       wrf_mini-outputFiles/ wrfout_d01_2016-02-23_13:00:00 wrfout_d01_2016-02-23_16:00:00
calwrf.exe@    wrf_mini-outputFiles.zip wrfout_d01_2016-02-23_14:00:00

```

- Now we have to edit calwrf.inp to match with the wrfout files we have
- Using Vim (or your favourite editor):
 - `vi calwrf.inp`
- Change the name of the output to WRF.DAT; replace all the spatial data and times to “-9” (meaning all); set the number of WRF output files to 5; and list out the names of the wrf files (you can copy the text in the following)

```

1 create 3D.DAT file for WRF output
2 calwrf.lst          ! Log file name
3 WRF.DAT ! Output file name
4 -9,-9,-9,-9,-9,-9 ! Beg/End I/J/K ("- for all)
5 -9                ! Start datetime (UTC yyyyymmddhh, "-" for all)
6 -9                ! End   datetime (UTC yyyyymmddhh, "-" for all)
7 5                  ! Number of WRF output files
8 wrfout_d01_2016-02-23_12:00:00 ! File name of wrf output (Loop over files)
9 wrfout_d01_2016-02-23_13:00:00
10 wrfout_d01_2016-02-23_14:00:00
11 wrfout_d01_2016-02-23_15:00:00
12 wrfout_d01_2016-02-23_16:00:00
13
14
15 ***** Below are comments *****
16

```

```

Create 3D.DAT file for WRF output
calwrf.lst      ! Log file name
WRF.DAT ! Output file name
-9,-9,-9,-9,-9,-9 ! Beg/End I/J/K ("- " for all)
-9 ! Start datetime (UTC yyymmddhh, "- " for all)
-9 ! End  datetime (UTC yyymmddhh, "- " for all)
5 ! Number of WRF output files
wrfout_d01_2016-02-23_12:00:00 ! File name of wrf output (Loop over files)
wrfout_d01_2016-02-23_13:00:00
wrfout_d01_2016-02-23_14:00:00
wrfout_d01_2016-02-23_15:00:00
wrfout_d01_2016-02-23_16:00:00

```

```

***** Below are comments *****

```

- Save and quit (Vim: **Esc** > **:wq**), then run calwrf.exe
 - **./calwrf.exe calwrf.inp**

- If successful, you should see:

```
Processing GLOBAL ATTRIBUTES:

Warning: Attribute not exist:   3  DYN_OPT      -43
Check whether this att. is critical
Completed WRF file:           3  wrfout_d01_2016-02-23_14:00:00
Open WRF netcdf file         4  : wrfout_d01_2016-02-23_15:00:00
N_TIMES:                      1

Processing GLOBAL ATTRIBUTES:

Warning: Attribute not exist:   3  DYN_OPT      -43
Check whether this att. is critical
Completed WRF file:           4  wrfout_d01_2016-02-23_15:00:00
Open WRF netcdf file         5  : wrfout_d01_2016-02-23_16:00:00
N_TIMES:                      1

Processing GLOBAL ATTRIBUTES:

Warning: Attribute not exist:   3  DYN_OPT      -43
Check whether this att. is critical
Completed WRF file:           5  wrfout_d01_2016-02-23_16:00:00

CALWRF succeeded
STOP 99999
```

- If you run **ls**, you should see WRF.DAT (3D file to be used in calmet → analogous to MM4.DAT that we used before) alongside WRF.m2d (2D fields from WRF output)