Compiling MET

Julie Prestopnik

Overview of Options

- NOAA and NCAR users
  - use existing MET builds on various machines
- MET available in a Docker container
- NCO
  - Download, configure, compile MET and its dependencies for official releases

Existing MET Builds (NOAA machines)

- MET User's Page shows all existing MET builds
  - Download -> Existing MET Builds
    - Select desired MET Version

met-8.0	met-7.0
tide	gyre
luna	surge
mars	venus
theia	gaea
	jet

Docker Container

- MET User's Page
  - Download -> Docker Container
    - Select desired MET Version
  - Direct Link
    - https://dtcenter.org/met/users/downloads/docker_container/index.php

- GitHub Repo
  - Instructions for building and using MET
    - https://github.com/NCAR/container-dtc-met

Downloading MET

- Download MET release and compile locally.
  - Register and download:
    - https://dtcenter.org/met/users/downloads/index.php
  - Be sure to get latest patches for version

- Supported Platforms and Compilers:
  1. Linux with GNU compilers
  2. Linux with Intel compilers
  3. Linux with Portland Group (PGI) compilers

Dependencies

- REQUIRED:
  - C++/Fortran Compilers (GNU, PGI, Intel)
  - GNU Make Utility
  - Unidata's NetCDF4 library (both NetCDF-C and NetCDF-CXX)
  - HDF5 library (required to support NetCDF4)
  - NCEP's BUFRLIB Library v10.2.3
  - GNU Scientific Library (GSL)
  - Z Library (zlib)
- OPTIONAL:
  - GRIB2 C-Library with JASPER and PNG libraries
  - HDF4 and HDF-EOS2 libraries for MODIS-Regrid tool
  - Cairo and FreeType libraries for MODE-Graphics tool
  - Python 2.7 for Python embedding to prepare 2D gridded data fields for reading by MET tools
- RECOMMENDED:
  - Unified Post-Processor
  - COPYGB (included with Unified Post-Processor)
  - wgrib and wgrib2
  - R statistics and graphics package

Building MET

- Steps for building MET:
  1. Build required/optional libraries, using the same family of compilers
  2. Download and patch the latest MET patches
  3. autoconf determines available compilers, but can be explicitly set by the user
  4. Set necessary environment variables
     1. Paths for HDF5, NetCDF, BUFRLIB, and GSL libraries
  5. Configure the installation for the system and run configure
     1. Compilation of various tools can be turned on/off
  6. Run make install and make test and check for runtime errors
     1. Tests scripts run each of the desired tools at least once
     2. Uses sample data distributed with the tarball

Directory Structure

File or Directory	Contents
README	Installation instructions and release notes
configure (plus supporting files and subdir)	Used by the autoconf build process; Configures the MET package for installation on a system
bin/	Built MET executables
data/	Contains map data, colortables, sample input data, GRIB and GRIB2 table files, and default configuration files
doc/	MET User's Guide
out/	Output generated by the test scripts
scripts/	Test scripts to be run after building MET
src/	MET Source Code

Nuances of Compiling MET on WCOSS machines, Theia, Gaea, and Jet

Script for MET and its Dependent Libraries

- Bash script (compile_all.sh)
- Expects **tar_files** directory with library and MET tar files
- Requires environment variables to be set
  - TEST_BASE – installation directory
  - COMPILER – format is compiler_version (e.g. ics_15.0.3)
  - MET_SUBDIR – location for top level MET subdir (e.g. met-8.0)
  - MET_TARBALL – name of MET tarball (e.g. met-8.0.20180927.tar.gz)
  - MET_PYTHON_CC – -I<directory containing Python includes>
  - MET_PYTHON_LD – -L<directory containing Python library file> -l<necessary libraries to link to>
  - PYTHON_MODULE_USE – modulefile location for Python
- Set of constants inside script for dependent libraries and MET
  - Set to 1 to install, 0 not to install

How to find Python values

- Find default Python
  - On tide, for example, default Python is 2.6.6, MET requires 2.7
    - ***which python*** returns /usr/bin/python
    - ***python –version*** returns Python 2.6.6
  - Find Python 2.7
    - Run ***module avail python*** or ***module spider python***
    - On tide, for example, ***module avail python***, produces:
      - ---------- /usrx/local/dev/modulefiles ----------
      - python/2.7.13 python/2.7.14 python/3.6.3(default)
      - ---------- /usrx/local/modulefiles ----------
      - python/2.7.12 python/2.7.13(default)
    - Run ***module use*** on selected modulefiles path and ***module load*** for desired version
    - ***echo $LD_LIBRARY_PATH*** returns …:/usrx/local/dev/python/2.7.14/lib:…
    - Check /usrx/local/dev/python/2.7.14/include for include files

Tide / Gyre (IBM)

- export TEST_BASE=/global/noscrub/Julie.Prestopnik/met/8.0/
- export COMPILER=ics_15.0.3
- export MET_SUBDIR=${TEST_BASE}/
- export MET_TARBALL=met-8.0.20180927.tar.gz
- export MET_PYTHON_CC=-I/usrx/local/dev/python/2.7.14/include/python2.7
- export MET_PYTHON_LD=-L/usrx/local/dev/python/2.7.14/lib\ -lpython2.7\ -lutil
- export PYTHON_MODULE_USE=/usrx/local/dev/modulefiles

Luna / Surge (Cray)

- export TEST_BASE=/gpfs/hps3/emc/global/noscrub/Julie.Prestopnik/met/8.0
- export COMPILER=PrgEnv-intel_5.2.56
- export MET_SUBDIR=${TEST_BASE}/
- export MET_TARBALL=met-8.0.20180927.tar.gz
- export MET_PYTHON_CC=-I/usrx/local/prod/python/2.7.13/include/python2.7
- export MET_PYTHON_LD=-L/usrx/local/prod/python/2.7.13/lib\ -lpython2.7\ -lutil
- export PYTHON_MODULE_USE=/usrx/local/prod/modulefiles

Mars / Venus (Dell)

- export TEST_BASE=/gpfs/dell2/emc/verification/noscrub/Julie.Prestopnik/met/8.0
- export COMPILER=ips_18.0.1.163
- export MET_SUBDIR=${TEST_BASE}/
- export MET_TARBALL=met-8.0.20180927.tar.gz
- export MET_PYTHON_CC=-I/usrx/local/prod/packages/python/2.7.13/include/python2.7
- export MET_PYTHON_LD=-L/usrx/local/prod/packages/python/2.7.13/lib\ -lpython2.7\ -lutil\ -lpthread
- export PYTHON_MODULE_USE=/usrx/local/prod/modulefiles/core_third/python

Theia

- setenv TEST_BASE /contrib/met/8.0/
- setenv COMPILER intel_16.1.150
- setenv MET_SUBDIR ${TEST_BASE}
- setenv MET_TARBALL met-8.0.20180927.tar.gz
- setenv MET_PYTHON_CC -I/contrib/anaconda/anaconda2/4.4.0/include/python2.7
- setenv MET_PYTHON_LD -L/contrib/anaconda/anaconda2/4.4.0/lib/\ -lpython2.7\ -lutil
- setenv PYTHON_MODULE_USE /contrib/modulefiles

Gaea

- setenv TEST_BASE /ncrc/usw/met/8.0/
- setenv COMPILER PrgEnv-intel_6.0.3
- setenv MET_SUBDIR ${TEST_BASE}
- setenv MET_TARBALL met-8.0.20180927.tar.gz
- setenv MET_PYTHON_CC -I?
- setenv MET_PYTHON_LD -L? -l?
- setenv PYTHON_MODULE_USE ?

- Have not yet installed met-8.0

Jet

- setenv TEST_BASE /lfs1/projects/dtc-hurr/MET/MET_releases/8.0/
- setenv COMPILER intel_14.0.3
- setenv MET_SUBDIR ${TEST_BASE}/
- setenv MET_TARBALL met-8.0.20180927.tar.gz
- setenv MET_PYTHON_CC -I?
- setenv MET_PYTHON_LD -L? -l?
- setenv PYTHON_MODULE_USE ?

- Have not yet installed met-8.0
- Trouble finding Python 2.7

Compiling MET on Tide

- COMPILE_GSL=0
- COMPILE_BUFRLIB=0
- COMPILE_ZLIB=0
- COMPILE_LIBPNG=0
- COMPILE_JASPER=0
- COMPILE_G2CLIB=0
- COMPILE_HDF=0
- COMPILE_HDFEOS=0
- COMPILE_NETCDF=0
- COMPILE_FREETYPE=0
- COMPILE_CAIRO=0
- COMPILE_MET=1
- COMPILE_MET_PATCHES=0

```
[Julie.Prestopnik@t21a3 8.0]$ ./compile_all.sh
TEST_BASE = /global/noscrub/Julie.Prestopnik/met/8.0/
COMPILER  = ics_15.0.3
MET_SUBDIR = /global/noscrub/Julie.Prestopnik/met/8.0//
MET_TARBALL = met-8.0.20180927.tar.gz
LD_LIBRARY_PATH = /global/noscrub/Julie.Prestopnik/met/8.0//external_libs/lib:/usrx/local/lsf/9.1/linux2.6-glibc2.3-x86_64/lib:/usrx/local/intel/2015/impi_latest/intel64/l
ib:/usrx/local/intel/2015/itac_latest/mic/slib:/usrx/local/intel/2015/itac_latest/intel64/slib:/usrx/local/intel/2015/composerxe/lib/intel64:/usrx/local/intel/2015/mpirt/l
ib/intel64:/usrx/local/intel/2015/ipp/lib/intel64:/usrx/local/intel/2015/mkl/lib/intel64:/usrx/local/intel/2015/tbb/lib/intel64/gcc4.4

Compiling libraries into: /global/noscrub/Julie.Prestopnik/met/8.0//external_libs
module load ics/15.0.3
ics/15.0.3
export  CC=/usrx/local/intel/2015/composer_xe_2015.3.187/bin/intel64/icc
export CXX=/usrx/local/intel/2015/composer_xe_2015.3.187/bin/intel64/icc
export  FC=/usrx/local/intel/2015/composer_xe_2015.3.187/bin/intel64/ifort
export F77=/usrx/local/intel/2015/composer_xe_2015.3.187/bin/intel64/ifort
export F90=/usrx/local/intel/2015/composer_xe_2015.3.187/bin/intel64/ifort
module use /usrx/local/dev/modulefiles
module load python

Compiling MET at Thu Sep 27 21:33:02 UTC 2018
MET Configuration settings...
export MET_FONT_DIR=/global/noscrub/Julie.Prestopnik/met/8.0//fonts
export MET_FREETYPEINC=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs/include/freetype2
export MET_CAIROINC=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs/include/cairo
export MET_GSL=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs
export MET_PYTHON_CC=-I/usrx/local/dev/python/2.7.14/include/python2.7
export MET_HDF=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs
export MET_TARBALL=/global/noscrub/Julie.Prestopnik/met/8.0//tar_files/met-8.0.20180927.tar.gz
export MET_HDFEOS=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs
export MET_NETCDF=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs
export MET_FREETYPELIB=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs/lib
export MET_CAIROLIB=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs/lib
export MET_PYTHON_LD=-L/usrx/local/dev/python/2.7.14/lib -lpython2.7 -lutil
export MET_GRIB2C=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs
export MET_SUBDIR=/global/noscrub/Julie.Prestopnik/met/8.0//
export MET_BUFRLIB=/global/noscrub/Julie.Prestopnik/met/8.0//external_libs/lib
cd /global/noscrub/Julie.Prestopnik/met/8.0/met-8.0
./configure --prefix=/global/noscrub/Julie.Prestopnik/met/8.0// --enable-grib2 --enable-modis --enable-mode_graphics --enable-lidar2nc --enable-python  > configure.log 2>&
1
make > make.log 2>&1
make install > make_install.log 2>&1
make test > make_test.log 2>&1
Finished compiling at Thu Sep 27 21:40:31 UTC 2018
[Julie.Prestopnik@t21a3 8.0]$
```

```
[Julie.Prestopnik@t21a3 met]$ more 8.0
#%Module#####################################################################
##
##      Model Evaluation Tools
##
proc ModulesHelp { } {
  puts stderr "Sets up the paths and environment variables to use the Model Evaluation Tools v8.0
     *** For help see the official MET webpage at http://www.dtcenter.org/met/users ***"
}

# The intel compiler is required to run MET
module load ics/15.0.3
module use /usrx/local/dev/modulefiles
module load python/2.7.14

set base /global/noscrub/Julie.Prestopnik/met/8.0/
set lib_base /global/noscrub/Julie.Prestopnik/met/8.0/external_libs

prepend-path PATH $base/bin:$lib_base/bin
prepend-path LD_LIBRARY_PATH $lib_base/lib
```

```
[Julie.Prestopnik@t21a3 8.0]$ module use /global/noscrub/Julie.Prestopnik/modulefiles
[Julie.Prestopnik@t21a3 8.0]$ module load met/8.0
[Julie.Prestopnik@t21a3 8.0]$ ensemble_stat --version

MET Version:    V8.0
Repository:     https://svn-met-dev.cgd.ucar.edu/tags/met/met-8.0
Revision:       5860
Change Date:    2018-09-27 13:46:59 -0600 (Thu, 27 Sep 2018)

[Julie.Prestopnik@t21a3 8.0]$ ensemble_stat

*** Model Evaluation Tools (METV8.0) ***

Usage: ensemble_stat
        n_ens ens_file_1 ... ens_file_n | ens_file_list
        config_file
        [-grid_obs file]
        [-point_obs file]
        [-ens_mean file]
        [-obs_valid_beg time]
        [-obs_valid_end time]
        [-outdir path]
        [-log file]
        [-v level]
        [-compress level]

        where   "n_ens ens_file_1 ... ens_file_n" is the number of ensemble members followed by a list of ensemble member file names (required).
                "ens_file_list" is an ASCII file containing a list of ensemble member file names (required).
                "config_file" is an EnsembleStatConfig file containing the desired configuration settings (required).
                "-grid_obs file" specifies a gridded observation file. May be used multiple times (optional).
                "-point_obs file" specifies a NetCDF point observation file. May be used multiple times (optional).
                "-ens_mean file" specifies an ensemble mean model data file (optional).
                "-obs_valid_beg time" in YYYYMMDD[_HH[MMSS]] sets the beginning of the matching time window (optional).
                "-obs_valid_end time" in YYYYMMDD[_HH[MMSS]] sets the end of the matching time window (optional).
                "-outdir path" overrides the default output directory (.) (optional).
                "-log file" outputs log messages to the specified file (optional).
                "-v level" overrides the default level of logging (2) (optional).
                "-compress level" overrides the compression level of NetCDF variable (0) (optional).

[Julie.Prestopnik@t21a3 8.0]$
```

Running MET on NOAA Machines

- Existing MET Builds webpage
  - https://dtcenter.org/met/users/downloads/existing_met_builds/index.php
- Theia
  - module use /contrib/modulefiles
  - module load met/8.0
- Gaea (currently only 7.0)
  - module use /usw/ldtn/modulefiles
  - module load met/7.0
- Jet (currently only 7.0)
  - module use /lfs1/projects/dtc-hurr/MET/MET_releases/modulefiles
  - module load met/7.0

Running MET on NOAA Machines (cont'd)

- WCOSS machines
  - tide/gyre (currently tide only)
    - module /global/noscrub/Julie.Prestopnik/modulefiles
    - module load met/8.0
  - luna/surge (currently luna only)
    - module use /gpfs/hps3/emc/global/noscrub/Julie.Prestopnik/modulefiles
    - module load met/8.0
  - mars/venus (currently mars only)
    - module use /gpfs/dell2/emc/verification/noscrub/Julie.Prestopnik/modulefiles
    - module load met/8.0

Getting Help

- MET User's Page – Documentation and User Support
  - MET User's Guide - https://dtcenter.org/met/users/docs/overview.php
  - FAQ - https://dtcenter.org/met/users/support/faqs/index.php
  - Known Issues and Fixes - https://dtcenter.org/met/users/support/known_issues/index.php
  - MET Online Tutorial - https://dtcenter.org/met/users/support/online_tutorial/index.php

- MET-Help - email assistance service providing support for registered MET users. ***met_help@ucar.edu***

Questions?