CMAQ Installation Quick Start Guide

System Checks

The following support software are required for compiling and running CMAQ.

1. Fortran and C compilers, e.g., Intel, Portland Group, Gnu
2. Git
3. I/O API
4. netCDF
5. Message Passing Interface (MPI), e.g., OpenMPI or MVAPICH2.

Install CMAQ and Required Libraries

In the directory where you would like to install CMAQ, issue the following command to clone the CMAS CENTER GitHub repository for CMAQv5.2 Beta:

```
git clone -b 5.2Gamma https://github.com/CMASCenter/EPA-CMAQ.git C
```

For instructions on installing CMAQ from tarballs, see Chapter 5.

Configure the CMAQ build environment

Edit the file `CMAQv5.2/config.cmaq` to configure the CMAQ installation for the local computing architecture and compilers.

1. Set the script variable `CMAQ_HOME` to point the installation directory location of CMAQ on your Linux system.

2. Set the script variable `compiler` for one of the supported compilers.

3. Install the CMAQ libraries

The CMAQ build scripts require the following libraries and INCLUDE files to be available in the CMAQ_LIB directory (Note the CMAQ_LIB gets set automatically by the config.cmaq script, where `CMAQ_LIB =` `$CMAQ_HOME/lib`):

- netCDF library and INCLUDE files are located in the `$CMAQ_LIB/netcdf` directory
- I/O API library and module files are located in the `$CMAQ_LIB/ioapi` directory
- MPI library and INCLUDE files are located in the `$CMAQ_LIB/mpi` directory

The config.cmaq script will automatically link the required libraries into the CMAQ_LIB directory. Set the locations of the netCDF, I/O API, and MPI installations on your Linux system with the following config.cmaq environment variables:

- `setenv IOAPI`: the location of the I/O API installation on your system.
- `setenv NETCDF`: the location of the netCDF installation on your system.
- `setenv MPI`: the location of the MPI (OpenMPI or MVAPICH) on your system.

For example, if your netCDF libraries and includes files are installed in /usr/local/netcdf, set NETCDF to /usr/local/netcdf. Similarly, if your I/O API library is installed in /home/cmaq/ioapi/Linux2_x86_64ifort, set IOAPI to /home/cmaq/ioapi/Linux2_x86_64ifort.

4. Check the names of the I/O API and netCDF libraries using the `ioapi_lib` and `netcdf_lib` script variables.

5. Check the name of the MPI library using the `mpi` script variable. For MVAPICH use `-lmpich`; for openMPI use `-lmpi`.

6. Invoke the settings in the configuration script: `source` `$CMAQ_HOME/config.cmaq`

Compiling CMAQ

Compile the model builder, bldmake:

```
cd $CMAQ_HOME/UTIL/bldmake/src
make
```

Create the model executables for ICON, BCON, and CCTM:

```
cd $CMAQ_HOME/PREP/icon/scripts
./bldit.icon |& tee bldit.icon.log
```

```
cd $CMAQ_HOME/PREP/bcon/scripts
./bldit.bcon
```

```
cd $CMAQ_HOME/CCTM/scripts
./bldit.cctm
```

Install the CMAQ input reference/benchmark data

Dwnloaded the CMAQ reference data from the [CMAS Center Software Clearinghouse](#) and copy to `$CMAQ_HOME`. Navigate to the `$CMAQ_HOME` directory, unzip and untar the `CMAQv5.2.DATA.tar.gz` file:

```
cd $CMAQ_HOME
tar xvzf CMAQv5.2.DATA.tar.gz
```

Running the CMAQ Installation Test Simulation

To run the test simulation for the various CMAQ programs, change directories to the location of each program and execute the run script.

Run ICON to produce initial conditions:

```
cd $CMAQ_HOME/PREP/icon/scripts
./run.icon |& tee icon.log
```

Run BCON to produce boundary conditions:

```
cd $CMAQ_HOME/PREP/bcon/scripts
./run.bcon |& tee bcon.log
```

Check the ICON and BCON log file to ensure that the programs completed successfully. Note that CMAQ test simulation doesn't actually require that ICON and BCON be run; the test input data include CCTM-

ready initial and boundary conditions files.

Configure the CCTM script for MPI

For an MPI configuration with 6 processors,

```
cd $CMAQ_HOME/CCTM/scripts
```

Edit the CCTM run script (run.cctm) for the MPI configuration that you will use:

```
setenv NPROCS 6
setenv NPCOL_NPROW "3 2"
```

Most clustered multiprocessor systems require a command to start the MPI run-time environment. The default CCTM run script uses the *mpirun* command. Consult your system administrator to find out how to invoke MPI when running multiprocessor applications.

For single-processor computing, set NPROCS to 1 and NPCOL_NPROW to "1 1"

For single-processor computing,

```
setenv NPROCS 1
setenv NPCOL_NPROW to "1 1"
```

After configuring the MPI settings for your Linux system, using the following command to run the CCTM. Per the note above, different Linux systems have different requirements for submitting MPI jobs. The command below is an example of how to submit the CCTM run script

and may differ depending on the MPI requirements of your Linux system.

```
./run.cctm |& tee cctm.log
```

CMAQ Operational Guidance Document (c) 2016