Atmospheric Model Evaluation Tool (AMET) User's Guide

Prepared under Work Assignment 3-04 of U.S. EPA Contract EP-W-05-045, "Operation of the Center for Community Air Quality Modeling and Analysis"

Prepared for:	Bill Benjey and Alice Gilliland U.S. EPA, ORD/NERL/AMD/APMB E243-04 USEPA Mailroom Research Triangle Park, NC 27711
Prepared by:	Institute for the Environment The University of North Carolina at Chapel Hill 137 E. Franklin St., CB 6116 Chapel Hill, NC 27599-6116

Date prepared: May 30, 2008

Contents

Tables .... **iv**

1. Overview **1**
- 1.1 Overall Objective and Basic Structure 1
- 1.2 Concept of an AMET "Project" 1
- 1.3 Organization of This User's Guide 2

2. Directory Structure **3**

3. Configuration **4**
- 3.1 Perl Configuration File (`amet-config.pl`) 4
- 3.2 R Configuration File (`amet-config.R`) 5

4. Datasets **6**
- 4.1 Model Data 6
- 4.2 Observational Data 7

5. Database Setup **11**
- 5.1 MET Setup 11
- 5.2 AQ Setup 12
- 5.3 Basic MySQL Commands 13

6. Database Population **14**
- 6.1 The wrfExample Project 14
- 6.2 The mm5Example Project 15
- 6.3 The mcipExample Project 16
- 6.4 The aqExample Project 17
- 6.5 Creating a New MET Project 18
- 6.6 Creating a New AQ Project 19

7. Analysis **20**
- 7.1 wrfExample 21
- 7.2 aqExample 23
- 7.3 Creating a New Analysis Project 27

8. CMAS Support for AMET **27**

References **28**

Appendix A: Overview Flow Diagram

Appendix B: Configuration and Input Files

Tables

Table 2-1. Directories under `$AMETBASE`. 3
Table 3-1. Most common variables that need to be changed in a`met-config.pl`. 4
Table 3-2. Most common variables that need to be changed in `amet-config.R`. 5
Table 7-1. MET analysis scripts. 22
Table 7-2. AQ analysis scripts. 24
Table B-1. amet-config.pl 32
Table B-2. amet-config.R 34
Table B-3. MET setup_project.pl 35
Table B-4. AQ setup_project.input 36
Table B-5. MET populate_project.input 37
Table B-6. AQ populate_project.input 40
Table B-7. MET analysis input variables 42
Table B-8. AQ analysis input variables 49
Table B-9. AQ network input variables 55
Table B-10. AQ networks, species, and frequency of observations 56

1. Overview

1.1 Overall Objective and Basic Structure

The Atmospheric Model Evaluation Tool (AMET) (Gilliam et al., 2005) is a suite of software designed to facilitate the analysis and evaluation of meteorological and air quality models. AMET matches the model output for particular locations to the corresponding observed values from one or more networks of monitors. These pairings of values (model and observation) are then used to statistically and graphically analyze the model's performance.

More specifically, AMET is currently designed to analyze outputs from the PSU/NCAR Mesoscale Model (MM5), the Weather Research and Forecasting (WRF) model, and the Community Multiscale Air Quality (CMAQ) model, as well as Meteorology-Chemistry Interface Processor (MCIP)-postprocessed meteorological data (surface only). The basic structure of AMET consists of two *fields* and two *processes*.

- The two fields (scientific topics) are **MET** and **AQ**, corresponding to meteorology and air quality data.

- The two processes (actions) are **database population** and **analysis**. Database population refers to the underlying structure of AMET; after the observations and model data are paired in space and time, the pairs are inserted into a database (MySQL). Analysis refers to the statistical evaluation of these pairings and their subsequent plotting.

Practically, a user may be interested in using only one of the fields (either MET or AQ), or may be interested in using both fields. That decision is based on the scope of the study. The three main software components of AMET are **`MySQL`** (an open-source database software system), **`R`** (a free software environment for statistical computing and graphics), and **`perl`** (an open-source cross-platform programming language).

1.2 Concept of an AMET "Project"

A central organizing structure for AMET applications is a *project*. A project groups a particular model simulation (specific model, physics-set, spatial domain, grid scale, etc.) with all of the AMET database tables that correspond to that simulation, the scripts necessary to populate that database, and the scripts required to analyze that project. For example, you might have one project for a 2002 36-km continental U.S. simulation, and another project for a 2002 12-km Eastern U.S. simulation. A project can be for either MET or AQ, not for both. It is essential that you uniquely name each project. It is recommended that you follow the directory structure when creating new projects, by copying one of the three example directories (`aqExample`, `mm5Example`, `wrfExample`) provided with the installation and then renaming it to the new project's name.

1.3 Organization of This User's Guide

The Community Modeling and Analysis System (CMAS) Center has created this user's guide to assist you in applying the AMET system in your work. CMAS obtained the MET and AQ portions of AMET separately from EPA, then integrated the two to create a consistent and integrated AMET package that uses the UNIX C-shell interface to perform both MET and AQ model evaluation and analyses. After this integration, we tested the integrated AMET package on multiple environments. After this integration, we tested the integrated AMET package on multiple environments.

Finally, we created this user's guide. The contents of the remaining sections are listed below.

- Section 2 describes the overall directory structure of the AMET installation.
- Section 3 gives instructions on how to configure the Perl and R configuration files.
- In Section 4 is an overview of the various model outputs and observed data provided with the AMET release.
- Section 5 provides instructions on how to create the AMET MySQL database, with specific instructions for each of the MET and AQ models. Sample MySQL commands are also shown for illustrative SQL queries.
- Section 6 gives instructions on how to populate the AMET MySQL database, with specific instructions for each of the MM5, WRF, MCIP, and CMAQ models, and also on how to create a new MET project and a new AQ project for subsequent analyses.
- In Section 7 are instructions on how to perform model evaluation for each of the WRF and CMAQ models, and includes an overview of the functionality of all the MET and AQ evaluation scripts provided.
  IMPORTANT NOTE: *The set of analyses/evaluation scripts provided in this release are strictly for illustration purposes on the functionality/design of AMET, and are not to be construed as a recommended suite of analyses scripts for model evaluation. We encourage the user community to use the scripts we have provided as examples as well as a basis to start developing other analyses scripts and contribute them to the modeling community to increase AMET functionality.*
- Section 8 discusses how to obtain support for AMET from the Community Modeling and Analysis System (CMAS) Center (http://www.cmascenter.org).
- In Appendix A is an overall flow diagram for AMET and its various components.
- Appendix B provides information on the various input files used in AMET. For each input file, a table lists brief descriptions of all user-defined variables that can be set by the user for a given evaluation.

Before using AMET and this user's guide, you must first install the AMET package on your system. For information on the installation process, please see the separate *Atmospheric Model Evaluation Tool (AMET) Installation Guide* that can be downloaded from the CMAS web site.

2. Directory Structure

In our discussions, we refer to the top of the AMET directory structure as "`$AMETBASE`". This environment variable is actually set in many of the scripts discussed below. For example, if you had untarred AMET's tarball in your home directory, then `AMETBASE` would be `~/AMET`.

Under `$AMETBASE` are the directories shown in Table 2-1.

Table 2-1. Directories under `$AMETBASE`.

Directory	Description
bin	External executables used by helper scripts.
configure	Configuration files for Perl and R.
model_data	Model output data (contains field-specific [i.e., `MET` and `AQ`] subdirectories).
obs	Observational data (e.g., MADIS, discussed in Section 4.2) (contains field-specific [i.e., `MET` and `AQ`] subdirectories).
output	Output of database population and analysis (contains project-specific subdirectories).
perl	Perl scripts used primarily for database population.
R	R scripts used for statistical analysis.
scripts_analysis	Project-specific wrapper scripts and inputs for analysis (contains project-specific subdirectories).
scripts_db	Project-specific wrapper scripts and inputs for database population (contains project-specific subdirectories).
src	Source code for third-party software.

Note: For large model outputs and for MADIS observations that cover a long period of time, it may be prudent to link these data within the appropriate AMET directories rather than moving or copying them.

3. Configuration

After untarring the AMET code and data and installing/building the required three tiers of software components (as discussed in the AMET installation guide referenced above), the next stage is to configure the AMET system. In the `$AMETBASE/configure` directory, you will find five files:

- A perl configuration file (`amet-config.pl`).
- An R configuration file (`amet-config.R`).
- Three Meteorological Assimilation Data Ingest System (MADIS) configuration files (`MET_matching.conf, MET_matching.MCIP.conf,` and `MET_matching_nonPX.conf`). These configuration files describe how MADIS variables "match" (map or translate to) various MET model variables. Most users will not need to change these variable mapping files, so they are not discussed further in this guide. For the AQ side, the matching of variables between model output and observations is project specific. See Section 6.5, "Creating a New AQ Project," for more details on changing the default mapping of AQ observations to model variables.

3.1 Perl Configuration File (`amet-config.pl`)

The Perl configuration file is used by the underlying Perl programs to populate the database with the data pairs created by AMET's matching of model output for particular locations to corresponding observed values (these data pairs are referred to in this document as "model-obs pairs"). Most users will need to modify only a few specific lines of this configuration file. The most common variables to change are shown in Table 3-1; a full description can be found in Appendix B.1.

Table 3-1. Most common variables that need to be changed in `amet-config.pl`.

Variable	Description
`$mysql_server`	The MySQL server location. Examples are **`localhost`** if MySQL is installed on the same machine on which you have installed AMET, or **`rama.cempd.unc.edu`** if you have installed the MySQL server on a remote host called **`rama`**.
`$root_pass`	Password for **`ametsecure`**, or your **`$root_login`** (if you changed it from "**`ametsecure`**"). This MySQL user will be created later when you are working through Section 5. To provide additional security, AMET is shipped with permissions that allow this file to be read only by the user.
`$perl_lib`	Location of additional Perl libraries. In our testing, we added additional libraries through the Comprehensive Perl Archive Network (CPAN) to **`/usr/local/lib/perl`**. Below that is the version number directory, and below that are the additional modules.

Variable	Description
$R_lib	All of the R library paths that should be searched by the R scripts. In our installation, they are **/usr/local/pkgs/Rpackages:$R_dir/lib/R/site-library**, since we installed additional R libraries under **/usr/local/pkgs/Rpackages**.
$fslftp_madis[1]	MADIS ftp site (MADIS is discussed in Section 4.2).
$login_madis[1]	MADIS-provided login.
$pass_madis[1]	MADIS-provided password.
$pass_ncep[1]	Anonymous password for the National Centers for Environmental Prediction (NCEP); this is usually your e-mail address. Note the necessary "\@".

[1] Needed only if using MET side of AMET.

3.2 R Configuration File (amet-config.R)

The R configuration file is used by the underlying R programs to perform statistical analysis on your model-obs pairs. Most users will need to modify only a few specific lines of this configuration file. The most common variables to change are shown in Table 3-2.

Table 3-2. Most common variables that need to be changed in amet-config.R.

Variable	Description
server	The MySQL server location. Examples are **localhost** if MySQL is installed on the same machine on which you have installed AMET, or **rama.cempd.unc.edu** if you have installed the MySQL server on a remote host called **rama**.
passwd	Password for **ametsecure**, or your **login** (if you changed it from "**ametsecure**"). This MySQL user will be created later when you are working through Section 5. To provide additional security, AMET is shipped with permissions that allow this file to be read only by the user. *Note*: The R and perl configuration files should have matching passwords and logins.

Variable	Description
newLibPath	Should include all R library paths where R packages were installed. In our installation, they are: **c("/usr/local/pkgs/Rpackages",oldLibPath)**, since we installed additional R libraries under **/usr/local/pkgs/Rpackages**. *Note:* The **oldLibPath** needs to be included in this variable.

4. Datasets

The AMET release includes both model and observational datasets provided as examples. You should have downloaded these into the proper directories during the installation process.

4.1 Model Data

For the model data, we have included both meteorological and air quality data. We have organized the data into four example projects: "mm5Example", "wrfExample", "mcipExample", and "aqExample". On the MET side, there is a 5-day MM5 simulation (July 04 2002 00:00 UTC to July 08 2002 23:00 UTC), a 5-day WRF simulation (July 05 2002 00:00 UTC to July 09 2002 23:00 UTC), and a 5-day MCIP-postprocessed MM5 simulation (July 05 2002 00:00 UTC to July 09 2002 23:00 UTC). The MM5 data file is

```
$AMETBASE/model_data/MET/mm5Example/MMOUT_DOMAIN1_02Jul04
```

Note that we have bolded "mm5Example" in the directory name above to highlight the fact that we are using the project name to organize the model output files into directories.

The WRF data consist of five WRF output files in netCDF format:

```
$AMETBASE/model_data/MET/wrfExample/
wrfout_d01_2002-07-05_00:00:00
wrfout_d01_2002-07-06_00:00:00
wrfout_d01_2002-07-07_00:00:00
wrfout_d01_2002-07-08_00:00:00
wrfout_d01_2002-07-09_00:00:00
```

The MCIP data consist of five METCRO2D files and one GRIDCRO2D file in netCDF format:

```
$AMETBASE/model_data/MET/mcipExample/
GRIDCRO2D_2002187
METCRO2D_2002186
METCRO2D_2002187
METCRO2D_2002188
```

```
METCRO2D_2002189
METCRO2D_2002190
```

On the AQ side, we have included two CMAQ output files for the period July 01 2002 0:00 UTC to July 11 2002 00:00 UTC. The two files:

```
$AMETBASE/model_data/AQ/aqExample/
test.36km.conc
test.36km.dep
```

correspond to the concentration and wet deposition output files from CMAQ, after they have been postprocessed with the `combine` utility.

All of the spatial domains cover the continental U.S. and have a 36-km grid resolution.

4.2 Observational Data

As with the model data, the observations directory structure is divided between MET and AQ fields. On the MET side, all of the observations come from the Meteorological Assimilation Data Ingest System (MADIS), provided by the National Oceanic and Atmospheric Administration (NOAA). If you are going to use AMET for MET analysis, you must first contact MADIS to obtain a MADIS account, login, and password (see **http://www-sdd.fsl.noaa.gov/MADIS** for details). You will use this login information in setting up the Perl configuration file (discussed in Section 3.1).

In the AMET structure, all of the MADIS data are stored under `$AMETBASE/obs/MET`. The file `stations.csv` is a metadata file that describes the station ID and location of each of the MADIS monitoring sites; this file is in comma-separated-value (.csv) format. The AMET distribution from CMAS will not include any MET observations from MADIS. When you run the database population scripts (Section 6), they will automatically call the MADIS system to download the available monitoring sites' observations for the specified time period via ftp. For example, the radiosonde data will be downloaded to

```
$AMETBASE/obs/MET/point/raob/netcdf
```

Each of the observation files is a netCDF file representing one hour's worth of data from all available monitoring sites. The netCDF file can be stored in a gzip compressed format to save space, and unzipped automatically as needed. Note that the default settings for the MADIS obs extraction are set in template files in `$AMETBASE/bin/madis_input`. You can modify these template files to extract mesonet data if desired. Also, you can adjust the quality control level from its current highest level. See the MADIS documentation available from their web site for more information on how to customize these template files.

On the AQ side, we have included the observational data for the following networks: Air Quality System (AQS) network, Clean Air Status and Trends Network (CASTNET), Interagency Monitoring of PROtected Visual Environments (IMPROVE) network, Mercury Deposition Network (MDN), National Atmospheric Deposition Program (NADP) network, SouthEastern

Aerosol Research and Characterization Study (SEARCH) network, and the Speciated Trends Network (STN). The observational datasets have been preprocessed and reformatted (in some instances from their original sources) for access by AMET. The temporal range is network dependent, and ranges from 2001 to 2006. The monitoring station locations are provided by a series of .csv files under the subdirectory `$AMETBASE/obs/AQ/site_lists`. A brief synopsis of each network, along with the steps taken to create these data for AMET, is given below. Note that in the species lists, each line is of the format "observed species name; model species name (units)".

4.2.1. Clean Air Status and Trends Network (CASTNet) Weekly

Source: CASTNet data are obtained through the CASTNet web site: **http://www.epa.gov/castnet/**. Weekly CASTNet data can be obtained by downloading the "drychem" file under the prepackaged datasets on the CASTNet web site. No postprocessing of the downloaded data is necessary in order for them to be compatible with the Site Compare (`sitecmp`) software packaged with the AMET system.

Species used with AMET:

tso4; ASO4T (μg/m^3)
tno3; ANO3T (μg/m^3)
tnh4; ANH4T (μg/m^3)
tno3+nhno3; ANO3T+HNO3_UGM3 (TNO3; μg/m^3)
nhno3; HNO3_UGM3 (μg/m^3)
wso2; SO2_UGM3 (μg/m^3)

4.2.2. Clean Air Status and Trends Network (CASTNet) Hourly

Source: CASTNet data are obtained through the CASTNet web site: **http://www.epa.gov/castnet/**. Hourly CASTNet ozone data can be obtained by downloading the files labeled "ozone_yyyy" under the prepackaged datasets on the CASTNet web site. Additionally, a "metdata_yyyy" file is also available on the CASTNet web site, which contains several meteorological variables in addition to ozone. No postprocessing of the downloaded data is necessary in order for them to be compatible with AMET's `sitecmp`.

Species used with AMET:

Ozone; ozone (ppb)

Additional species that could be used with AMET:

Surface Temperature	Precipitation
Relative Humidity	10m Wind Speed
Solar Radiation	10m Wind Direction

4.2.3. Interagency Monitoring of PROtected Visual Environments (IMPROVE)

Source: IMPROVE data can be through the IMPROVE web site: **http://vista.cira.colostate.edu/improve/**. The IMPROVE web site links to the Visibility Information Exchange Web System (VIEWS) web site, which is an interactive system for downloading various air-quality-related data. IMPROVE data obtained through the VIEWS system do not require any additional processing to work with AMET's `sitecmp`.

Species used with AMET:

SO4f_val; ASO4T (μg/m^3)
NO3f_val; ANO3T (μg/m^3)
NH4f_val; ANH4T (μg/m^3)
MF_val; PM25 (μg/m^3)
OCf_val; PM_OC (μg/m^3)
ECf_val; AECT (μg/m^3)
OCf_val+ECf_val; PM_OC+AECT (TC; μg/m^3)

4.2.4. Mercury Deposition Network (MDN)

Source: MDN data are obtained through the NADP/MDN network web site: **http://nadp.sws.uiuc.edu/mdn/**. Data are available for download as a comma-delimited file for all sites. No postprocessing of the downloaded data is necessary in order for them to be used with AMET's `sitecmp`.

Species used with AMET (from CMAQ deposition file):

HGconc; TWDEP_HG (ng/L)
HGdep; TWDEP_HG (μg/m^2)

4.2.5. National Atmospheric Deposition Program (NADP)

Source: NADP data are obtained through the NADP/NTN web site: **http://nadp.sws.uiuc.edu/**. Weekly wet concentration data are downloaded in comma-delimited format directly from the NADP web site. No postprocessing of the downloaded data is necessary in order for them to be used with AMET's `sitecmp`.

Species used with AMET (from CMAQ deposition file):

NH4; WDEP_NHX (mg/L or kg/ha)
NO3; WDEP_TNO3 (mg/L or kg/ha)
SO4; WDEP_ASO4T (mg/L or kg/ha)
precip; RT (mm)

4.2.6. SouthEastern Aerosol Research and Characterization (SEARCH) Study

Source: SEARCH data are obtained through the SEARCH web site: **http://www.atmospheric-research.com/public/index.html**. The SEARCH data can be downloaded as comma-delimited files for each SEARCH site. In order to be used with `sitecmp` and AMET, the individual site files must first be merged together into a single file. The example SEARCH data file provided with AMET should serve as an example of how the raw SEARCH data need to be combined and formatted in order to work with `sitecmp` and AMET.

Species used with AMET:

- o3; O3 (ppb)
- co; CO (ppb)
- so2; SO2 (ppb)
- no; NO (ppb)
- hno3; HNO3 (ppb)
- teom; PM25 (µg/m^3)
- no3; ANO3T (µg/m^3)
- so4; ASO4T (µg/m^3)
- nh4; ANH4T (µg/m^3)
- noy; NOY (ppb)

4.2.7. Speciation Trends Network (STN)

Source: STN data are obtained through the EPA's Air Quality System (AQS), located at **http://www.epa.gov/ttn/airs/airsaqs/**. The data provided with AMET are a sample of the STN data that can be obtained through the AQS. No postprocessing of the downloaded STN data is necessary in order for them to work with `sitecmp` and AMET.

Species used with AMET:

- m_so4; ASO4T (µg/m^3)
- m_no3; ANO3T (µg/m^3)
- m_nh4; ANH4T (µg/m^3)
- FRM $PM_{2.5}$ Mass; $PM_{2.5}$ (µg/m^3)
- oc_adj; PM_OC (µg/m^3)
- ec_niosh; AECT (µg/m^3)
- oc_adj+ec_niosh; PM_OC+AECT (TC; µg/m^3)

4.2.8. Air Quality System (AQS)

Source: AQS data are obtained through the EPA's Air Quality System (AQS), located at **http://www.epa.gov/ttn/airs/airsaqs/**. Various species of atmospheric gases are available for download through the AQS. Although only ozone data are provided with the AMET release, both **sitecmp** and AMET can process other gaseous species available through the AQS (some customization is required, however).

Species used with AMET:

ozone; ozone (ppb)

5. Database Setup

The next step in using AMET is to set up the MySQL database. Please refer to the flow diagram in Appendix A to understand the overall flow of data among the various modules within AMET. This section must be completed before you populate the database with your project-specific data (Section 6). This setup process is required only once for a given AMET installation. There are separate setup procedures for the two fields, MET and AQ. If you are using AMET for only one of those fields, you need to run only the corresponding setup. If you are running AMET for both fields, you will need to run both setups. In the following discussion, we assume the default name of the AMET database, "amet", and the default AMET MySQL user, "ametsecure". If you decide to change either of these, then you will need to update the appropriate variables in the Perl and R configuration files in the directory $AMETBASE/configure (see Section 3). Before you run the setup scripts, you will need to know the "root" password for the MySQL administrator. Note that this is not the same as the "ametsecure" password that will be created using the scripts discussed below.

5.1 MET Setup

Go to the setup directory

```
$ cd $AMETBASE/scripts_db/setupMET
```

Here, you will see a series of C-shell scripts (.csh) and an input file (.input). The input file, sites_meta.input, defines the metadata file for the MADIS stations (station ID, location, etc.).

To initialize the MET side of AMET, you will need to edit and run the initialize_met_db.csh script. Specifically, you should make sure the value of AMETBASE corresponds to your system. If you have already run the setup procedure for AQ, you should set the variable "new_db" to "N"; otherwise set it to "Y" to create a new amet database. Run the initialize script

```
$ ./initialize_met_db.csh
```

This will ask you for MySQL's "root" password. It will then set up the AMET database, "amet", and the AMET user, "ametsecure". It will also create two tables in your amet database: stations (observation sites metadata) and project_log (contains summary information for each MET project).

The setup directory also contains scripts for removing projects and removing the `amet` database. To delete a specific MET project, use

```
$./delete_project.csh
```

This script will ask you for the name of the project to delete. **CAUTION:** This will delete all of the tables in the database corresponding to that project.

To delete the `amet` database from MySQL, use

```
$./delete_db.csh
```

This script will ask you for the MySQL "`root`" password before proceeding. *Use this script with **EXTREME CAUTION** because this will delete all of the data in the database corresponding to **all** of the projects (**both MET and AQ**).*

5.2 AQ Setup

Go to the setup directory

```
$ cd $AMETBASE/scripts_db/setupAQ
```

As for the MET setup, you will see a series of C-shell scripts and an input file. The input file, `sites_meta.input`, defines the metadata files for the various AQ monitoring networks (station ID, location, etc.).

To initialize the AQ side of AMET, you will need to edit and run the `initialize_aq_db.csh` script. Specifically, you should make sure the value of `AMETBASE` corresponds to your system. If you have already run the setup procedure for MET, you should set the variable "`new_db`" to "`N`"; otherwise set it to "`Y`" to create a new `amet` database. Run the initialize script

```
$ ./initialize_aq_db.csh
```

This will ask you for MySQL's "`root`" password. It will then set up the AMET database, "`amet`", and the AMET user, "`ametsecure`". It will also create three tables in your `amet` database: `site_metadata` (observation sites metadata), `project_units` (contains species-specific units for each project), and `aq_project_log` (contains summary information for each AQ project).

The setup directory also contains scripts for removing projects and removing the `amet` database. To delete a specific MET project, use

```
$./delete_project.csh
```

This script will ask you for the name of the project to delete. **CAUTION:** This will delete all of the tables in the database corresponding to that project.

To delete the `amet` database from MySQL, use

```
$./delete_db.csh
```

This script will ask you for the MySQL "`root`" password before proceeding. *Use this script with **EXTREME CAUTION** because this will delete all of the data in the database corresponding to **all** of the projects (**both MET and AQ**).*

5.3 Basic MySQL Commands

As you begin to go through the `amet` database setup and the project-specific database populate process, you may want to query the database directly to see your progress. Here are a few commands to help you interact directly with the MySQL `amet` database. For more specifics, see one of the many MySQL books available, or look at the documentation under

http://dev.mysql.com/doc

To log onto the MySQL server from the command line, type

```
$ mysql -u ametsecure -D amet –p
```

This will give you a MySQL prompt ("`mysql>`"). Note that all MySQL commands are case insensitive, and they must end with a semicolon ("`;`").

To get a list of all the tables in your database, type

```
mysql> show tables;
```

After you have populated all of the example projects (end of Section 6), that command will yield a table like this:

```
+---------------------+
| Tables_in_amet      |
+---------------------+
| aqExample           |
| aq_project_log      |
| mm5Example_profiler |
| mm5Example_raob     |
| mm5Example_surface  |
| mcipExample_profiler|
| mcipExample_raob    |
| mcipExample_surface |
| project_log         |
| project_units       |
| site_metadata       |
| stations            |
| wrfExample_profiler |
| wrfExample_raob     |
| wrfExample_surface  |
+---------------------+
```

To select every column and row in your `project_log` table:

```
mysql> select * from project_log;
```

To select the latitude, longitude, and common name columns from the stations metadata table and limit the results to the first 20 rows:

```
mysql> select lat,lon,common_name from stations limit 20;
```

To select all station metadata where the monitor is from the CASTNET network:

```
mysql> select * from site_metadata where network=‘castnet’;
```

To determine which networks are included in the `aqExample` project:

```
mysql> select distinct network from aqExample;
```

6. Database Population

The database population phase of AMET must be performed for each new project. As discussed in Section 1.2, the *project* is the organizing structure that we use to group a particular model simulation with the scripts and data used to populate the `amet` tables. If you go to the database populate directory by typing

```
$ cd $AMETBASE/scripts_db
```

you will see four project directories, in addition to the setup directories described earlier. The projects are

(a) a MET example for the WRF model (`wrfExample`),
(b) a MET example for the MM5 model (`mm5Example`),
(c) a MET example for MCIP postprocessed surface data (`mcipExample`), and
(d) an AQ example for the CMAQ model (`aqExample`).

In the following subsections, we describe how to run each project.

6.1 The wrfExample Project

Go to the project directory

```
$ cd $AMETBASE/scripts_db/wrfExample
```

Here, you will see two input files and two C-shell scripts. The `setup_project.input` file is the input file for initializing the project's database tables (discussed later in Section 6). The only thing you should need to change in this file is the "`$email`" variable. Note that you should use the backslash "escape" character, "`\`", to prevent Perl from evaluating the "`@`" in your e-mail address. The `populate_project.input` file describes specific flags for the model outputs and observations that you want to process. See Appendix B for information on the specifics relating to each variable. You will likely not need to change anything in this second input file.

The C-shell file `metProject.csh` is a wrapper script for calling the Perl programs that actually populate the AMET database with the project data. You will likely only need to verify that the variable `AMETBASE` has been updated for this project. Run the script by typing

```
$ ./metProject.csh >& log.populate
```

This C-shell script will create three empty project tables in the AMET database: `wrfExample_profiler`, `wrfExample_raob`, and `wrfExample_surface.` These tables correspond to the matches between the model outputs and (1) the wind profiler observations, (2) the radiosonde observations, and (3) the surface observations. After creating these tables, the script then begins the matching process. This consists of auto-ftp-ing data from the MADIS web site for the model's temporal period, unzipping the downloaded data, finding the geographic location of each observation site on the model grid and interpolating to those locations, populating the appropriate table with the model-obs pairs for each variable, and optionally rezipping the data for compressed storage. Finally, the script updates the `project_log` with summary information for the `wrfExample` project.

The second C-shell file, `metFTP.csh`, is a wrapper script for calling Perl programs to download observational data from MADIS for a specific period of time. This allows you to download observational data without having the model output. Make sure the variable `auto_ftp` is set to 1 when running this script. Please note that the MADIS data need to be downloaded once for a given time period and will subsequently be available to all projects.

6.2 The mm5Example Project

Go to the project directory:

```
$ cd $AMETBASE/scripts_db/mm5Example
```

Here, you will see two input files and two C-shell scripts. The `setup_project.input` file is the input file for initializing the project's database tables (discussed later in Section 6). The only thing you should need to change in this file is the "`$email`" variable. Note that you should use the backslash "escape" character, "\", to prevent Perl from evaluating the "@" in your e-mail address. The `populate_project.input` file describes specific flags for the model outputs and observations that you want to process. See Appendix B for information on the specifics relating to each variable. You will likely not need to change anything in this second input file. Note that the MM5 model has been run with the Pleim-Xiu (PX) surface model and has the relevant flags for this configuration; see Appendix B for more information.

The C-shell file `metProject.csh` is a wrapper script for calling the Perl programs that actually populate the AMET database with the project data. You will likely only need to verify that the variable `AMETBASE` has been updated for this project. Run the script by typing

```
$ ./metProject.csh >& log.populate
```

This C-shell script will create three empty project tables in the AMET database: `mm5Example_profiler`, `mm5Example_raob`, and `mm5Example_surface.` These tables correspond to the matches between the model outputs and (1) the wind profiler observations, (2) the radiosonde observations, and (3) the surface observations. After creating these tables, the script then begins the matching process. This consists of converting the MM5 output file into a netCDF file, auto-ftp-ing data from the MADIS web site for the model's temporal period, unzipping the downloaded data, finding the geographic location of each observation site on the model grid and interpolating to those locations, populating the appropriate table with the model-obs pairs for each variable, and optionally rezipping the data for compressed storage. Finally, the script updates the `project_log` with summary information for the `mm5Example` project.

Note that if your MM5 simulation was configured to use a non-PX surface model, you will need to change the two variables `$diagnose_sfc` and `$match_config` in the `populate_project.input` file; see Appendix B for more information.

The second C-shell file, `metFTP.csh`, is a wrapper script for calling Perl programs to download observational data from MADIS for a specific period of time. This allows you to download observational data without having the model output. Make sure that the variable `auto_ftp` is set to 1 when running this script. Please note that the MADIS data need to be downloaded only once for a given time period and will subsequently be available to all projects.

6.3 The mcipExample Project

Go to the project directory

```
$ cd $AMETBASE/scripts_db/mcipExample
```

Here, you will see two input files and two C-shell scripts. The `setup_project.input` file is the input file for initializing the project's database tables (discussed later in Section 6). The only thing you should need to change in this file is the "`$email`" variable. Note that you should use the backslash "escape" character, "\", to prevent Perl from evaluating the "@" in your e-mail address. The `populate_project.input` file describes specific flags for the model outputs and observations that you want to process. See Appendix B for information on the specifics relating to each variable. You will likely not need to change anything in this second input file.

The C-shell file `metProject.csh` is a wrapper script for calling the Perl programs that actually populate the AMET database with the project data. You will likely only need to verify the variable `AMETBASE` has been updated for this project. Also note the variable `AMET_MCIP_GRID`, which should provide the full path to the GRIDCRO2D file used for this project. Run the script by typing

```
$ ./metProject.csh >& log.populate
```

This C-shell script will create three empty project tables in the AMET database: `mcipExample_profiler`, `mcipExample_raob`, and `mcipExample_surface.` These tables correspond to the matches between the model outputs and (1) the wind profiler observations, (2) the radiosonde observations, and (3) the surface observations. However, AMET currently works only on MCIP surface data, so the profiler and raob tables will be empty. After creating these tables, the script then begins the matching process. This consists of auto-ftp-ing data from the MADIS web site for the model's temporal period, unzipping the downloaded data, finding the geographic location of each observation site on the model grid and interpolating to those locations, populating the appropriate table with the model-obs pairs for each variable, and optionally rezipping the data for compressed storage. Finally, the script updates the `project_log` with summary information for the `mcipExample` project.

The second C-shell file, `metFTP.csh`, is a wrapper script for calling Perl programs to download observational data from MADIS for a specific period of time. This allows you to download observational data without having the model output. Make sure the variable `auto_ftp` is set to 1 when running this script. Please note that the MADIS data need to be downloaded only once for a given time period and will subsequently be available to all projects.

6.4 The aqExample Project

Go to the project directory:

```
$ cd $AMETBASE/scripts_db/aqExample
```

Here, you will see two input files, one C-shell script, and the `combine` subdirectory. The `setup_project.input` file is the input file for initializing the project's database table (discussed later in Section 6). The only thing you should need to change in this file is the "`$email`" variable. Note that you should use the backslash "escape" character, "`\`", to prevent Perl from evaluating the "`@`" in your e-mail address. The `populate_project.input` file describes specific flags for the model outputs and observations that you want to process. See Appendix B for information on the specifics relating to each variable. You will likely not need to change anything in this second input file. The `combine` subdirectory is not used for this example; it is discussed later in Section 6.5, "Creating a New AQ Project".

The C-shell file `aqProject.csh` is a wrapper script for calling the Perl programs that actually populate the AMET database with the project data. The variable that you will likely need to change for this project is "`AMETBASE`". Run the script by typing

```
$ ./aqProject.csh >& log.populate
```

Note that if you are overwriting an existing project (i.e., using the same project name), you cannot redirect the screen output to a file because the script will require input from you.

This C-shell script will create one empty project table in the AMET database: `aqExample`. After creating this table, the script then begins the matching process. This consists of calling a series of Perl programs and Fortran helper programs. The two Fortran helper programs are `$AMETBASE/bin/cmp_airs.exe` and `$AMETBASE/bin/sitecmp.exe`; the first one matches the AQS network's data to the nearest grid cell in the CMAQ model, and the second one does the same for the other networks. After each network has been matched to the model, the aqExample table is populated with the model-obs pairs. In addition to creating and populating the aqExample table, the script updates the project_units table with each network for that project. This table defines the physical units of the species variables for this network (e.g., ppb vs. µg/m^3). Finally, the script updates the aq_project_log with summary information for the aqExample project.

6.5 Creating a New MET Project

When you create your own projects, we recommend that you utilize the structure of naming your directories after your projects. If you choose not to do this, you will have to modify the provided run scripts.

To create a new project, follow these basic steps:

1. Copy the appropriate example project to a new directory.
2. Rename it after your new project (use the *exact* project name, as many scripts use the project name to navigate directories).
3. Create a new project directory under $AMETBASE/model_data/MET for the input model data.
4. Change the appropriate variables in the project C-shell script.
5. Run your new populate script.

For example, if we were creating a new WRF project called "wrfNC2007", we would use

```
$ cd $AMETBASE/scripts_db
$ cp -r wrfExample wrfNC2007
$ cd wrfNC2007
```

Create a new model data directory and move or link your model data into it, as follows:

```
$ cd $AMETBASE/model_data/MET
$ mkdir wrfNC2007
$ cd wrfNC2007
$ ln -s <model data> .
```

Here, you would replace "<model data>" with the path to your model data file(s). The metProject.csh script will perform the model-obs matching on all model outputs in this new directory.

Next, edit the $AMETBASE/script_db/wrfNC2007/metProject.csh variables AMET_PROJECT ("wrfNC2007") and AMET_PROJ_DESC (your description of the project).

Finally, run the populate script:

```
$ cd $AMETBASE/scripts_db/wrfNC2007
$ ./metProject.csh
```

This will create a new MET project in the `amet` database. Specifically, it will create a new row in your `project_log` table and three new tables: `wrfNC2007_profiler`, `wrfNC2007_raob`, and `wrfNC2007_surface`.

6.6 Creating a New AQ Project

Before describing the creation of a new AQ project, we need to clarify a potentially confusing issue: the relationship between model species and monitor species. In order for AQ database population to work, there must be a mapping between the model species and the various network species. This mapping is accomplished by postprocessing the CMAQ model data, and through species definitions in the `populate_project.input` file. The model data used in the `aqExample` section (Section 6.3) were already postprocessed, so we did not need to go through that step when running the example project. In a new project, you will likely need to postprocess your CMAQ data before they are ingested into the `amet` database. This postprocessing is accomplished in the third step of creating a new AQ project (described below), using the `combine` Fortran program.

Also, when you create your own projects, we recommend that you utilize the structure of naming your directories after your projects. If you choose not to do this, you will have to modify the provided run scripts.

To create a new project, follow these basic steps:

1. Copy the appropriate example project to a new directory.
2. Rename it after your new project (use the *exact* project name).
3. Postprocess the model data using the `combine` Fortran program.
4. Create a new project directory under `$AMETBASE/model_data/AQ` for the input model data.
5. Change the appropriate variables in the project C-shell script.
6. Run the new populate script.

For example, if we were creating a new AQ project called "`aqNC2007`", we would use

```
$ cd $AMETBASE/scripts_db
$ cp -r aqExample aqNC2007
```

Next you will need to postprocess the raw CMAQ concentration and wet deposition files to map the data to the appropriate species names. To do this, you will use the `combine` Fortran program. Go to the `combine` directory:

```
$ cd aqNC2007/combine
```

Edit the `combine_conc.csh` and `combine_dep.csh` scripts for your model data. For detailed instructions on `combine`, see **http://www.cmascenter.org/help/model_docs/cmaq/4.6/EVALUATION_TOOLS.txt**. Run the two combine scripts:

```
$ ./combine_conc.csh
$ ./combine_dep.csh
```

Regarding the `populate_project.input` file, you will not need to change it if you do not change the species definition files (`spec_def*`). If you do change the species definitions, you will need to change the species definitions and names in the `$AMETBASE/scripts_db/aqNC2007/populate_project.input`. See Appendix B for details.

Next, create a new model data directory and move or link your postprocessed model data into it, as follows:

```
$ cd $AMETBASE/model_data/AQ
$ mkdir aqNC2007
$ cd aqNC2007
$ ln -s <model data> .
```

Here, you would replace "<model data>" with the path to your postprocessed model data file(s).

Next, in the file `$AMETBASE/scripts_db/aqNC2007/aqProject.csh`, edit the variables `AMET_PROJECT` ("aqNC2007"), `START_DATE` (start date of the model data), `END_DATE` (end date of the model data), `CONC_1` (model concentration file(s)), and `DEP_1` (model deposition file(s)). Note that the model data files should point to your postprocessed model data.

Finally, run the populate script:

```
$ cd $AMETBASE/scripts_db/aqNC2007
$ ./aqProject.csh
```

This will create a new AQ project in the `amet` database. Specifically, it will create a new row in your `aq_project_log` table, a series of new rows (one for each network) in your `project_units` table, and a new project table: `aqNC2007`.

7. Analysis

The analysis phase of AMET consists of performing statistical analyses on the model-obs pairs and creating plots of the resulting statistics. The basic process is to query the project's database table(s) using a set of SQL criteria; to perform statistical analyses on the returned data; and to create plots, tables, and text file outputs. The current AMET package contains a series of preprogrammed statistical analysis and plotting routines, based on the R language. These scripts are provided strictly as a starting point and as illustrative examples. Because all the model-obs pairs are stored in a MySQL database, an advanced user can decide to access those data in any desired manner, including other software packages. All that is required is a MySQL interface and some exploration of the table structure. We encourage advanced users to extend these R scripts to create more specific or advanced plotting capabilities, to use other languages to expand AMET analysis capabilities, and to contribute these updates to the CMAS community.

As with the database populate phase, the project is the organizing structure that we use to group a particular model run (specific model, physics or chemistry, spatial domain, scale, etc.) with the scripts used to analyze the `amet` tables and with the output from this analysis (plots and data).

A second organizing structure is the grouping of three files for each type of analysis. In the project directories, you will see a C-shell script and an input file with similar names (e.g., `run_timeseries.csh` and `timeseries.input`). These two files set up everything that is necessary to run an underlying R script in `$AMETBASE/R` and then they run that script. The use of the C-shell interface allows users who are not very familiar with R to perform these predefined analyses, shielding them from the actual R code.

7.1 wrfExample

Go to the project directory:

```
$ cd $AMETBASE/scripts_analysis/wrfExample
```

Here, you will see a series of C-shell scripts and their accompanying input files. We will go through an analysis script in detail as an example for running each of the scripts in the project.

The `run_spatial_surface.csh` script creates a series of maps comparing the surface monitors to the model for one or more days. Each plot provides color-coded model performance metrics at each of the monitor locations. In each map, it plots the particular value at the monitor location.

First, edit the `run_spatial_surface.csh` file. Change the `AMETBASE` variable to correspond with your setup. The corresponding input file, `spatial_surface.input`, will likely not need to be changed.

Run the script:

```
$ ./run_spatial_surface.csh
```

This will run the script and print out the location of the plots. In addition to the script outputs, a detailed log file is produced and located in the directory `$AMETBASE/scripts_analysis/wrfExample`. After the script has completed, go to the output directory and view your maps:

```
$ cd $AMETBASE/output/wrfExample/spatial_surface
```

You should see a whole series of plots of the form:

```
wrfExample.<stats>.<variable>.20020705.20020709.pdf
```

A brief summary of each of the C-shell scripts is given in Table 7-1.

Table 7-1. MET analysis scripts.

C-shell Script	Input	Description
`run_daily_barplot.csh`	**`daily_barplot.input`**	Creates a series of daily barplots for all sites within your domain.
`run_met_aq.csh`	**`met_aq_coupled.input`**	Compares AQ and MET data. Creates a spatial plot of correlation and a time series plot of bias. *Note*: You need to setup both MET and AQ models and define the MET variables and the AQ species and network of interest.
`run_plot_prof.csh`	**`plot_prof.input`**	Creates time-height maps of wind direction and speed profiles statistics. Also can create spatial plot of site locations and names to help identify sites in the domain.
`run_plot_raob.csh`	**`plot_raob.input`**	Creates plots of rawinsonde sites. It creates a series of maps of all sites and vertical profile for user-specified site.
`run_spatial_surface.csh`	**`spatial_surface.input`**	Creates a series of maps comparing surface monitor and model values for one or more days.
`run_summary.csh`	**`summary.input`**	Creates a summary plot of a series of MET variables. This includes a time series plot and an "ametplot," which includes a scatterplot, time series, boxplot, and for wind direction, a histogram and directional scatterplot. Note that you may want to change the variable **`AMET_RUNID`** to change the file naming convention. Also, the **`AMET_CRITERIA`** is an additional SQL command that further limits the data used. This can be customized by the user.

C-shell Script	Input	Description
run_timeseries .csh	**timeseries .input**	Creates a time series plot for one or more monitoring stations. Creates a time series plot for temperature, mixing ratio, wind speed and direction. You control which monitors are plotted through the **AMET_SITEID** variable. You can also control date ranges and provide additional SQL criteria to further limit the data. *Note*: Multiple sites can either be averaged or used to create separate plots.
run_wind_prof .csh	**wind_prof .input**	Creates wind arrow profiles for a set of specific monitoring sites.

Under the `$AMETBASE/scripts_analysis` directory, you will notice that there is only a `wrfExample` project on the MET side. We did not include the `mm5Example` project nor the `mcipExample` project because the different inputs are basically treated the same in terms of analysis. You could easily add a new project directory for `mm5Example` or `mcipExample` by following the instructions in Section 7.3.

7.2 aqExample

Go to the project directory:

```
$ cd $AMETBASE/scripts_analysis/aqExample
```

Here, you will see a series of C-shell scripts and their accompanying input files. We will go through an analysis script in detail as a example for running each of the scripts in the project.

The `run_scatterplot.csh` script creates a scatterplot for one species from one or more monitoring networks. It compares the observed values to the corresponding model values. The AQ side has an added complication: not all monitoring networks monitor all species. Therefore, the user needs to know which network(s) has the species of interest; if that species is not available in the network(s) specified, the analysis scripts will likely fail. See Section 4.2 and Appendix B for more details on the various species that are monitored (or available) from each AQ network.

First, edit the `run_scatterplot.csh` file. Change the `AMETBASE` variable to correspond with your setup. Note that the species selected is SO4 and you are plotting two networks: IMPROVE and CASTNET. The corresponding input file, `scatterplot.input`, will likely not need to be changed.

Also note that all AQ analysis scripts make use of the `Network.input` input file. This file contains information about each observational network available to the project that is needed by the R scripts. More information about this file can be found in Appendix B, Table B-13.

Run the script:

```
$ ./run_scatterplot.csh
```

This will run the script and print out the location of the plots. In addition to the script outputs, a detailed log file is produced and located in the directory `$AMETBASE/scripts_analysis/aqExample`. After the script has completed, go to the output directory and view your plots:

```
$ cd $AMETBASE/output/aqExample/scatterplot
```

You should see files of the form:

```
aqExample_SO4_scatterplot.pdf
```

A brief summary of each of the C-shell scripts is given in Table 7-2.

Table 7-2. AQ analysis scripts.

C-shell Script	Input	Description	Scope
run_boxplot .csh	**boxplot .input**	Creates a box plot of model-obs quartiles. Designed for an entire year of data. Can be used with a period as short as 3 months.	single network; single species; single simulation
run_boxplot_hourly .csh	**boxplot_hourly .input**	Creates side-by-side boxplots to create a diurnal average curve. Hourly data only.	single network; hourly species only; single simulation
run_bugleplot .csh	**bugleplot .input**	Model performance criteria are adjusted as a function of the average concentration of the observed value for that species. As the average concentration of the species decreases, the acceptable performance criteria increase. Creates a bias and error plot.	multiple networks; single species; single simulation

C-shell Script	Input	Description	Scope
run_met_aq .csh	**met_aq_coupled .input**	Compares AQ and MET data. Creates a spatial plot of correlation and a time series plot of bias. Note that you need to setup both MET and AQ models and define the MET variables and the AQ species and network of interest.	AQS; O_3; single MET/AQ simulation pair
run_pave_overlay .csh	**pave_overlay .input**	Creates a data file that can be used by the program **bldoverlay** to create an overlay file. This file can be used in PAVE to overlay over CMAQ model output. Hourly data only.	single network; hourly species only; single simulation
run_plot_spatial .csh	**plot_spatial .input**	Plots the observed value, model value, and difference between the model and obs for each site. Multiple values for a site are averaged to a single value for plotting purposes.	multiple networks; single species; single simulation
run_scatterplot .csh	**scatterplot .input**	Creates a single model vs. obs scatterplot. This script will plot a single species from up to three networks on a single plot. Summary statistics are also included on the plot.	multiple networks; single species; multiple simulations
run_scatterplot_MtoM .csh	**scatterplot_MtoM .input**	Creates a single model-to-model scatterplot. *Note*: The model points correspond to network's site locations only.	multiple networks; single species; multiple simulations
run_scatterplot_single .csh	**scatterplot_single .input**	Creates a scatter plot for a single network that includes more specific statistics than run_scatterplot.csh.	single network; single species; multiple simulations

C-shell Script	Input	Description	Scope
`run_scatterplot_skill.csh`	**`scatterplot_skill.input`**	Creates a forecast skill scatter plot. The script is designed to work specifically with O_3.	all AQS networks; O_3; single simulation
`run_soccerplot.csh`	**`soccerplot.input`**	Creates a soccerplot for one or more species over one or more networks. Criteria and goal lines are plotted in such a way as to form a "soccer goal" on the plot area. Two statistics are then plotted: Bias [**`NMB`** (normalized mean), **`FB`** (fractional), or **`NMdnB`** (normalized median)] on the x-axis and Error [**`NME`** (normalized mean), **`FE`**(fractional), or **`NMdnE`**(normalized median)] on the y-axis. The better the performance of the model, the closer the plotted points will fall within the "goal" lines.	multiple network; multiple species; multiple simulations
`run_stacked_barplot.csh`	**`stacked_barplot.input`**	Data are averaged (mean or median) for SO_4, NO_3, NH_4, EC, OC, and $PM_{2.5}$ for the model and observed values. Averages are then plotted on a stacked bar plot, along with the percent of the total $PM_{2.5}$ that each species constitutes.	STN or SEARCH; species predefined; multiple simulations
`run_stats_plots.csh`	**`stats_plots.input`**	Generates a series of spatial plots of **`NMB, NME, FB, FE`**, and **`Correlation`**. CSV files with additional domain- and site-specific statistics are also included.	multiple networks; single species; single simulation

C-shell Script	Input	Description	Scope
run_timeseries .csh	**timeseries .input**	Creates a time series plot. With multiple sites; the sites are time averaged to create a single plot. Also plots the bias and error between the obs and model.	single network; single species; multiple simulations

7.3 Creating a New Analysis Project

Creating a new analysis project requires the same basic steps for both the MET and the AQ models. When you create your own analysis projects, we recommend that you utilize the structure of naming your directories after your projects (described earlier). To run analyses, you must already have populated the database with the new project (see Sections 6.4 and 6.5). To create a new analysis project, follow these basic steps:

1. Copy the appropriate example project to a new directory.
2. Rename it after your new project (use the *exact* project name, as many scripts use the project name to navigate directories).
3. Change the appropriate variables in the project C-shell scripts.
4. Run the new analysis scripts.

For example, if we were creating a new WRF project called "`wrfNC2007`", we would use

```
$ cd $AMETBASE/scripts_analysis
$ cp -r wrfExample wrfNC2007
$ cd wrfNC2007
```

In each of the C-shell scripts you want to run, make sure to change the `AMET_PROJECT` to `wrfNC2007`. You will also likely change dates and custom titles in many of the scripts.

Run the desired analysis scripts from your new project directory.

8. CMAS Support for AMET

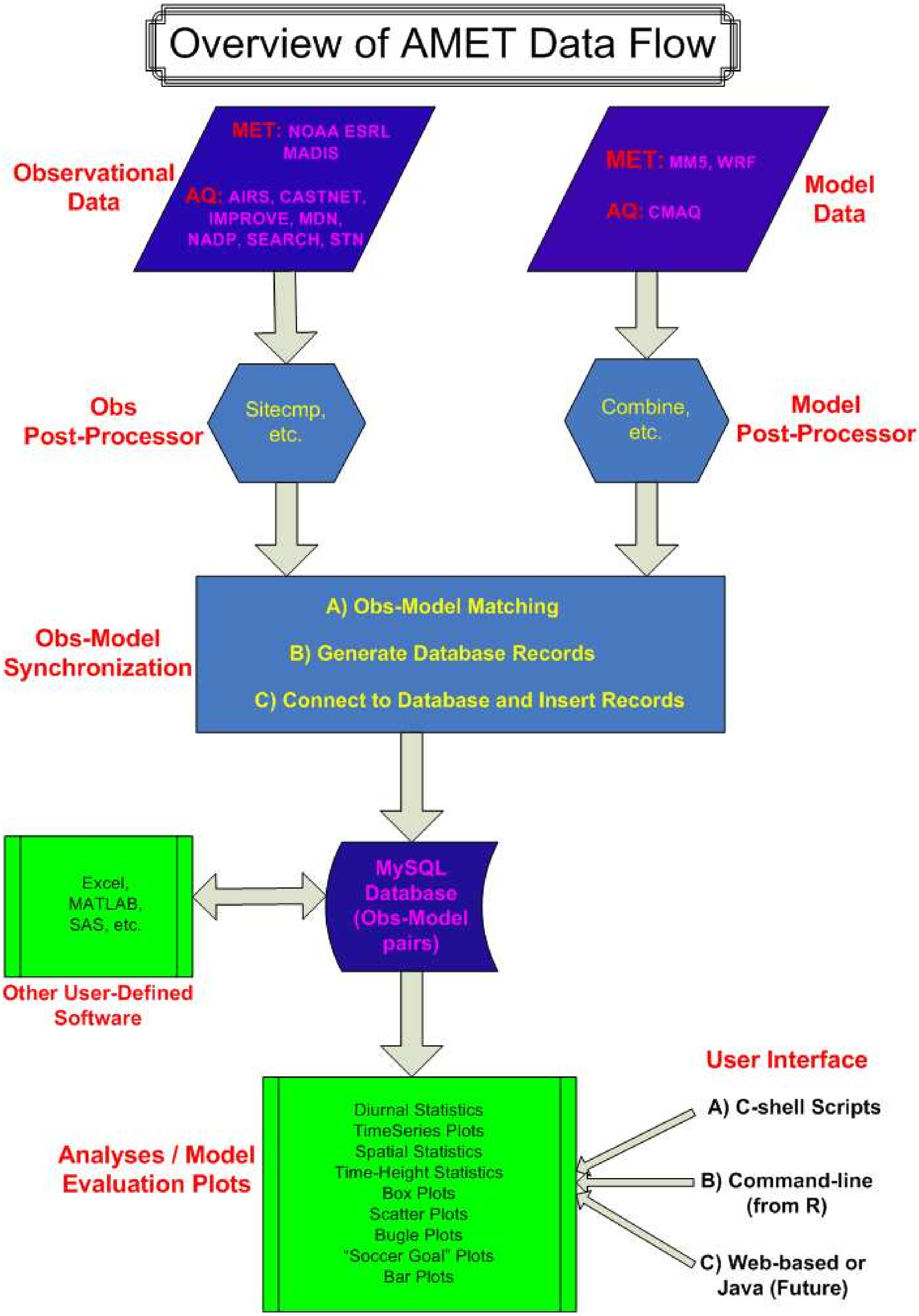
We have added AMET to Bugzilla, the CMAS bug-tracking and support request software system. You are encouraged to contact CMAS via bugzilla if you have bugs to report, or if you would like assistance with a specific component of AMET. The Bugzilla site for AMET is **http://bugz.unc.edu/enter_bug.cgi?product=AMET**. If you have never accessed this site before, a user account needs to be created by sending an email request to the CMAS administrator. We have created the following subsections on the Bugzilla AMET page:

- AQ Analyses
- AQ Database
- Installation
- Met Analyses
- Met Database
- Other

References

Gilliam, R, W. Appel, and S. Phillips, 2005. The Atmospheric Model Evaluation Tool: Meteorology Module. Presented at the 4th Annual CMAS Conference, Chapel Hill, NC, September 2005. Also available at:
http://www.cmascenter.org/conference/2005/abstracts/6_1.pdf

Appendix A:
Overview Flow Diagram


Overview of AMET Data Flow
Observational Data
MET: NOAA ESRL MADIS
AQ: AIRS, CASTNET, IMPROVE, MDN, NADP, SEARCH, STN
MET: MM5, WRF
AQ: CMAQ
Model Data
Obs Post-Processor
Sitecmp, etc.
Combine, etc.
Model Post-Processor
Obs-Model Synchronization
A) Obs-Model Matching
B) Generate Database Records
C) Connect to Database and Insert Records
Excel, MATLAB, SAS, etc.
Other User-Defined Software
MySQL Database (Obs-Model pairs)
User Interface
A) C-shell Scripts
B) Command-line (from R)
C) Web-based or Java (Future)
Analyses / Model Evaluation Plots
Diurnal Statistics
TimeSeries Plots
Spatial Statistics
Time-Height Statistics
Box Plots
Scatter Plots
Bugle Plots
"Soccer Goal" Plots
Bar Plots

Appendix B:
Configuration and Input Files

B.1 Perl Configuration File

This is the configuration file for all Perl scripts used in database population—for example, `$AMETBASE/configure/amet-config.pl`.

Table B-1. amet-config.pl

Variable	Description
`$amet_base`	Top of AMET directory tree. It is recommended that you keep this as the value of the environmental variable **`$AMETBASE`**. If you use something else, the provided scripts will need significant modification.
`$mysql_server`	MySQL server location. Examples are "**`localhost`**" for the same machine as AMET, or "**`rama.cempd.unc.edu`**" for a server on rama.
`$root_login`	MySQL user for adding data to the database and performing queries. "**`ametsecure`**" is recommended. This user will be created in the database setup. *Note:* To increase system security, users may want to restrict this configuration file to read and write only by user.
`$root_pass`	Password for "**`ametsecure`**", or your **`$root_login`** (if changed from "**`ametsecure`**"). This user will be created in the database setup.
`$dbase`	The name of the MySQL database created for all AMET projects. "**`amet`**" is recommended.
`$perl_lib`	Location of additional Perl libraries. In our example, we added additional libraries through CPAN to "**`/usr/local/lib/perl`**". Below **`$perl_lib`** is the version number directory and below that are the additional modules.
`$R_dir`	Base directory for R installation.
`$R_exe`	Full path to R executable.
`$R_lib`	All of the R library paths that should be searched. In our example, "**`/usr/local/pkgs/Rpackages:$R_dir/lib/R/site-library`**", we installed additional R libraries under **`/usr/local/pkgs/Rpackages`**.
`$EXEC_aqs`	Full path to **`cmp_airs`** executable.
`$EXEC_all`	Full path to **`sitecmp`** executable.
`$fslftp_madis`	MADIS ftp site (needed only if using MET side).

Variable	Description
$login_madis	MADIS-provided login (MET side only).
$pass_madis	MADIS-provided password (MET side only).
$fslftp_ncep	NCEP ftp site (needed only if using MET side).
$login_ncep	NCEP login; most likely "**anonymous**".
$pass_ncep	Anonymous password for NCEP, usually your email address. Note the necessary "\@" (MET side only).
$amet_verbose	Verbose `stdout`. "**yes**" for more verbose, "**no**" for less verbose.

B.2 R Configuration File

This is the configuration file for all R-based analyses scripts. Note that the values here should likely match the equivalent values in the Perl configuration file—for example, `$AMETBASE/configure/amet-config.R`.

Table B-2. amet-config.R

Variable	Description
`server`	MySQL server location. Examples are "**`localhost`**" for the same machine as AMET, or "**`rama.cempd.unc.edu`**" for a server on rama.
`login`	MySQL user for adding data to the database and performing queries. "**`ametsecure`**" is recommended. This user will be created in the database setup. *Note:* To increase system security, users may want to restrict this configuration file to read and write only by user.
`passwd`	Password for "**`ametsecure`**", or your login (if changed from "**`ametsecure`**"). This user will be created in the database setup, and should reflect the Perl configuration file.
`dbase`	The name of the MySQL database created for all AMET projects. "**`amet`**" is recommended.
`maxrec`	Maximum number of records to extract from the database for any one query. No maximum = **-1**.
`newLibPath`	Should include all R library paths where R packages were installed. In our installation, they are: **`c("/usr/local/pkgs/Rpackages",oldLibPath)`**, since we installed additional R libraries under **`/usr/local/pkgs/Rpackages`**. Note that the **`oldLibPath`** needs to be included in this variable.

B.3 MET Project Setup Input File

This is the MET input file for all new projects, and sets up an empty project—for example, `$AMETBASE/scripts_db/wrfExample/setup_project.input`.

Table B-3. MET setup_project.pl

Variable	Description
`$run_id`	Project name. Must be unique across both MET and AQ project names.
`$model`	MET model. Examples: "**`mm5`**", "**`wrf`**".
`$login`	Your username. Identifier used to populate **`user`** column in **`project_log`** table.
`$email`	Your email. Note: use the backslash character "****" to escape the "@" character for Perl.
`$description`	Brief project description.

B.4 AQ Project Setup Input File

This is the AQ input file for all new projects, and sets up an empty project—for example, `$AMETBASE/scripts_db/aqExample/setup_project.input`.

Table B-4. AQ setup_project.input

Variable	Description
$run_id	Project name. Must be unique across both MET and AQ project names.
$model	AQ model. Example: "**CMAQ**".
$login	Your username. Identifier used to populate **user** column in **aq_project_log** table.
$email	Your email. Note: use "****" to escape the "**@**" character for Perl.
$description	Brief project description.
$delete_id	Delete project if project already exists: "**y**" – yes, delete project, with prompt "**n**" – no, do not delete project
$exit_exists	Returned exit status of **AQ_create_project.pl** if project already exists: **1** – return error **0** – return success

B.5 MET Project Populate Input Files

This is the MET input file for populating new projects—for example, `$AMETBASE/scripts_db/wrfExample/populate_project.input`.

Table B-5. MET populate_project.input

Variable	Description
`$pid`	Project name. Must be unique across both MET and AQ project names.
`$model`	MET model. Examples: "**`mm5`**", "**`wrf`**", "**`mcip`**", or "**`eta`**".
`$obs_format`	Observation format: "**`madis`**" – MADIS observations "**`tdl`**" – preprocessed nonstandard obs
`$model_data_dir`	Directory of model output.
`$mcip_grid_file`	Path to MCIP `GRIDCRO2D` file.
`$obs_dir`	Top of observation directory. For MADIS, this is the equivalent of **`$TOP`** and **`$MADIS_DATA`**. For typical setup, this will be **`$AMETBASE/obs/MET`**.
`$tmp_dir`	Output directory for temporary MADIS files and MM5 netCDF files.
`$interp_method`	Interpolation technique from model grid to observation location: **`0`** – bilinear **`1`** – nearest neighbor
`$eval_class`	Observation types to include: "**`surface`**" – surface observations only "**`profiler`**" – wind profiler only "**`raob`**" – rawinsonde only "**`acars`**" – aircraft only "**`mesonet`**" – mesonet only "**`all`**" – all observation types

Variable	Description
`$diagnose_sfc`	Diagnose surface model: **`1`** – MM5 with PX surface model **`0`** – WRF or MM5 without PX surface model Note that if you use **`0`** for MM5, you need to change **`$match_config`** to non-PX configuration.
`$output_int`	Model output time step in hours.
`$eval_int`	Observations time step in hours.
`$forecast`	Model is a forecast, so observations may not be available for all hours: **`1`** – yes (forecast hour is set to time index in model output × output interval) **`0`** – no (forecast hour is set to zero)
`$auto_ftp`	Automatically ftp observations from MADIS: **`1`** – yes **`0`** – no
`$auto_unzip`	Automatically unzip MADIS data stored in gzip format: **`1`** – yes **`0`** – no
`$auto_zip`	Automatically zip MADIS data in gzip format for storage: **`1`** – yes **`0`** – no
`$time_index_start`	Starting index of model output. This can be used to skip a spin-up period. The starting point is index + 1. Example values: **`-1`** – start at 0th hour, i.e., include all time steps **`0`** – skip first time step, start at hour 1 **`11`** – start at hour 12

Variable	Description
`$process_npa`	Match model precip with National Precipitation Analysis (not supported at this time): **`1`** – yes **`0`** – no
`$real_time`	Real-time run, use current system date as time range in scripts: **`1`** – yes **`0`** – no
`$file_date`	For MM5 data, model data need to be in netCDF format or converted to netCDF. If they have been converted, they need to use the naming convention **`mmout.YYYY.MM.DD-NHRS`**. For example, **`mmout.2001.04.01.00-48`** is a model run starting at 4/1/2001 00 UTC and running for 48 hours. MM5 has already been converted to netCDF: **`1`** – yes **`0`** – no
`$rm_output_file`	Remove model output file after matching process: **`1`** – yes **`0`** – no
`$match_config`	Full path to configuration file for matching MADIS variables to model variables. Note that if you are using MM5 with a non-PX surface model, you will have to use **`MET_matching_nonPX.conf`** and change the **`$diagnose_sfc`** value to **`0.`** All MCIP output should use the MCIP configuration file.

B.6 AQ Project Populate Input Files

This is the AQ input file for populating new projects—for example, `$AMETBASE/scripts_db/aqExample/populate_project.input`.

Table B-6. AQ populate_project.input

Variable	Description
`$start_date`	Start date of model-observation matching.
`$end_date`	End date of model-observation matching.
`$m3_file[i]`	Array of model output file(s). Up to 6 model concentration files.
`$m3_dep_file[i]`	Array of model output file(s). Up to 6 model wet deposition files.
`$project_id`	Project name. Must be unique across both MET and AQ project names.
`$castnet_flag`	Example of monitoring network flag (CASTNET). Each network can be either included or excluded: "**`y`**" – include monitoring network's observations "**`n`**" – exclude monitoring network's observations
`$O3_units`	Concentration units for model O_3: "**`ppb`**" or "**`ppm`**"
`$precip_units`	Precipitation units in model: "**`cm`**" or "**`mm`**"
`$inc_rh`	Include relative humidity: "**`y`**" – include RH "**`n`**" – do not include RH
`$user_def_spec`	Additional or different definition for chemical species: "**`y`**" – user is providing new definitions "**`n`**" – user is not providing new definitions
`$ozone`	Example of specific species to be user-defined. For all species that use the default definitions, set them to "**`n`**". If user-defined, set that species to "**`y`**", and change the respective species name.

Variable	Description
$O3_conc_name	User-defined species. Used by **sitecmp** or **cmp_airs** to translate between model variables and monitor variables. Only used if respective species is set to user-defined ("**y**"). For example, total carbon (**TC_conc_name**) is being defined as AECT + PM_OC.

B.7 MET Analysis Input Files

The analysis input files are found in `$AMET/scripts_analysis/wrfExample`. The following is a partial list of variables. Not all of these variables are available in every input file.

Table B-7. MET analysis input variables

Variable	Description
`acars.only`	Either plot only ACARS airport sites on site location plot, or include all profile stations: **`TRUE`** – only ACARS sites **`FALSE`** – All sites
`ametp`	Flag to generate figures.
`aq_database`	Name of air quality database. This is typically the same as the meteorological database (e.g., "**`amet`**").
`aq_network`	Air quality network to be plotted. Default from **`csh`** script.
`aq_project`	Air quality project to be used. Default from **`csh`** script.
`aq_site_table`	Table in AMET database that contains metadata of air quality sites.
`aq_species_col`	Air quality variables to be included in analysis. Default value is from **`csh`** script.
`checksave`	Check to see if R datafile exists (i.e., if savefile is activated R datafile is saved, so there is no need to query database).
`colp`	Various color specifications for model evaluation metrics.
`colpRH`	Color specifications for model evaluation metrics involving relative humidity.
`colpT`	Color specifications for model evaluation metrics involving 2-m temperature.
`colpWS`	Color specifications for model evaluation metrics involving 10-m wind.

Variable	Description
`convert`	ImageMagick command. Used to crop margins of images. This variable has been deprecated.
`date`	Either one date used as both starting and ending date (default from **`csh`** script), or a vector containing starting and ending dates. Format: `YYYYMMDD`
`datee`	In **`met_aq_coupled.input`**, used to select ending date of analysis. Default uses value from **`csh`** script. In **`summary.input`**, used in the plot header to show the ending date used in the analysis. Does not change dates used in analysis. Format: `YYYYMMDD`
`dates`	In **`met_aq_coupled.input`**, used to select starting date of analysis. Default uses value from **`csh`** script. In **`summary.input`**, used in the plot header to show the starting date used in the analysis. Does not change dates used in analysis. Format: `YYYYMMDD`
`Daydelay`	Used in real-time mode to lag statistics by `daydelay` days.
`De`	Ending day of time series analysis. Default uses value from **`csh`** script.
`Diurnal`	Flag to partition and plot statistics as a function of time of day.
`Ds`	Starting day of time series analysis. Default uses value from **`csh`** script.
`elev`	Elevation criterion.
`extra`	Additional SQL criterion to use in database query.
`extra2`	Additional SQL criterion to use in database query. This is for the second site in time series plot.

Variable	Description
fcasthr	Forecast hour criteria used to isolate a particular segment of forecast data.
figdir	Directory where figures will be output. Default value comes from the **csh** script.
figid_sub	Secondary figure label.
figure	Full figure path and name.
fixed.legend	**FALSE** – Use default legend **TRUE** – Use custom legend from **legend.interval**
he	Ending hour of analysis.
histplot	Flag to plot histogram of statistics.
hs	Starting hour of analysis.
imageplot	Include contours on profile plots: **TRUE** – Contour and shade plot **FALSE** – Only shade plot
landuse	Land use classification criteria for MySQL query.
lat	Latitude specification for MySQL query.
layer	Layer specification for profiler statistics.
layerlab	Label for specified layer.
layerunit	Units for layer statistics.
layervar	Variable for layer statistics.
legend.div	Number of legend intervals.
legend.interval	Specific legend intervals.
legend.interval.bias	Legend intervals for mean bias.

Variable	Description
`level`	Level description.
`levsRH`	Levels for relative humidity statistics.
`levsT`	Levels for temperature statistics.
`levsWS`	Levels for wind speed statistics.
`lon`	Longitude specification for query.
`LT.offset`	Maximum and minimum UTC to local time offset in domain.
`max.dist`	Maximum distance allowed between AQ and MET sites in km.
`maxrec`	Maximum number of records to allow from database query (set to **`-1`** for unlimited).
`me`	Ending month of analysis; default comes from **`csh`** run script.
`met_database`	MySQL database name that holds meteorological project.
`met_network`	Meteorological network.
`met_project`	Meteorological project name to be used in analysis. Default uses value from **`csh`** script.
`met_site_table`	Name of the table that contains meteorological site information.
`met_variable_col`	Modeled and observational variables of interest. Should be column names from database.
`model`	AMET project name.
`model1`	Primary AMET project for time series plot.
`model2`	Secondary AMET project for time series plot.
`ms`	Starting month of analysis; default comes from **`csh`** run script.
`obnetwork`	Network used in analysis. This value is used only in the plot header, and does not affect the analysis.

Variable	Description
`obtime`	Times used in analysis. This value is used only in the plot header, and does not affect the analysis.
`pheight`	Plot height in pixels.
`pid`	A separate identification that is attached to the output. Can be used to distinguish between different output subsets for the same project. Default value is input from the **`csh`** script.
`player`	Logical to plot layer statistics.
`plotfmt`	File type of output. Default is taken from the **`csh`** script. Acceptable Values: **`png`**, **`pdf`**, **`jpg`**, or **`eps`**
`plotSingleProfile`	Logical (T or F) for generating hourly model-obs wind vector and potential temperature profile plots
`plotSiteMap`	Logical (T or F) for generating a site location plot that can aid in identifying site IDs and locations.
`plotsize`	Scale factor to increase or decrease the size of plots. **`1`** = 541 x 700 pixel (**`png`**) or 8.5 x 11 inch (**`pdf`**)
`processprof`	Logical to generate profile comparisons.
`prof`	Logical (**`T`** or **`F`**) for plotting raob-model profile comparison.
`proflim`	Lower and upper limit of profile plot.
`project`	AMET project name to be used in analysis. Default value is taken from **`csh`** run script.
`pwidth`	Width of plot.
`qcQ`	Quality control limits of moisture data. (*Note:* All data outside of this range are not considered.)
`qcT`	Quality control limits of temperature data.
`qcWS`	Quality control limits of wind speed data.

Variable	Description
query	MySQL query.
queryID	A separate identification that is attached to the output. Can be used to distinguish different output subsets for the same project. Default value is transferred from **pid**.
querystr	Additional SQL criteria that can be used to subset the data used by the analysis. Default value is transferred from the run script.
realtime	Run script in real-time automated mode.
savedir	Directory in which plots and other output will be saved. Default value is input from **csh** script.
savefile	Logical to generate an R data file that contains the data used in the statistics plots.
saveid	Name of R data file.
scex	Scale factor for statistics text size.
shadeplot	Flag to plot shaded statistics plot in addition to point statistics plot.
sres	Resolution of shaded plot in degrees.
statid	In **timeseries_plot**, the site ID to be used in the analysis. Default value is taken from the **csh** script. In **summary_plot**, the station label to be used in labeling the plot.
symb	Symbol shape to be used in plots. See R documentation for shape numbers.
symbo	Plot symbol.
symbsiz	Scale factor to adjust size of symbols on plots. **0.5** is very small while **1.5** is large.
syncond	This variable has been deprecated.

Variable	Description
t.test.flag	Logical to apply statistical significance test to the spatial statistics. If it is applied and the model and observation data are not different statistically, the values are not plotted.
textout	Logical to write text output of statistics and underlying data.
textstats	Logical to write text file of statistics.
thresh	Used in spatial surface to identify the minimum number of data points required at a particular site to compute the statistics.
time.of.day.utc	Range of time (UTC) to isolate met and AQ data (e.g., compare average temperature and average PM for hours between 6 and 12 UTC).
tserieslen	Length in days of time series if real-time mode is activated.
uniquepnum	Unique plot number (random).
wantfigs	Flag to generate figures.
wantsave	Flag to save station statistics data in R data file.
wdweightws	Logical to weight wind direction statistics by the wind speed (e.g., if wind speed falls below 3 m/s, the difference between model and observed wind direction is mitigated).
ye	Ending year of analysis default; comes from the **csh** script.
ys	Starting year of analysis default; comes from the **csh** script.
zlims	Specification of lower and upper vertical level of profile.

B.8 AQ Analysis Input Files

The analysis input files are found in `$AMET/scripts_analysis/aqExample`. The following is a partial list of variables in the AQ analysis input files. Not all of these variables are available in every input file.

Table B-8. AQ analysis input variables

Variable	Description
`add_query`	Additional query syntax to add to the MySQL query.
`aq_database`	AQ MySQL database. Most likely "**`amet`**".
`aq_network`	AQ monitoring network.
`aq_project`	AQ project name.
`aq_site_table`	AQ monitoring site MySQL table. Most likely "**`site_metadata`**".
`aq_species_col`	AQ variables, column names, from AQ project table.
`averaging`	Average across time: "**`n`**" – no averaging (default) "**`s`**" – seasonal averaging (DJF; MAM; JJA; SON) "**`m`**" – monthly averaging "**`a`**" – entire time period averaging
`axis_max_limit`	Axis (x and y) max limit: "**`NULL`**" – script-defined limit
`axis_min_limit`	Axis (x and y) min limit: "**`NULL`**" – script-defined limit
`bias_range_max`	Bias range max limit: "**`NULL`**" – script-defined limit

Variable	Description
bias_range_min	Bias range min limit: "**NULL**" – script-defined limit
Bldoverlay_exe	The location of the **bldoverlay** Fortran executable. Most likely **$AMETBASE/bin/bldoverlay**. (AQ only)
conf_line	Add confidence lines to scatterplots: "**y**" or "**n**".
coverage_limit	**%** necessary for data completeness (e.g., **75** means 75% data completeness).
custom_title	Custom title for plots: "" – no custom title
datee	End date of query, YYYYMMDD format (**met_aq_coupled.input** only).
dates	Start date of query, YYYYMMDD format (**met_aq_coupled.input** only).
diff_range_max	Difference range max limit: "**NULL**" – script defined limit
diff_range_min	Difference range min limit: "**NULL**" – script defined limit
end_date	End date of query, YYYYMMDD format.
end_hour	End hour of query, HH format.
error_range_max	Error range max limit: "**NULL**" – script defined limit
figdir	Output directory for plots.
fixed.legend	Fixed legend intervals (**met_aq_coupled.input**): "**F**" – False, use full range of data "**T**" – True, specify custom legend through **legend.interval**

Variable	Description
inc_ranges	Included quartile **ranges** on box plots**.**
legend.div	Number of levels if a fixed interval is not specified.
legend.interval	Custom legend interval (**met_aq_coupled.input**).
legend.interval.bias	Custom legend bias interval (**met_aq_coupled.input**).
LT.offset	Maximum and minimum UTC to LT offset in domain.
max.dist	Maximum distance allowed between MET and AQ "buddy" sites. These "buddy" sites are used to compare MET and AQ results in **met_aq_coupled**.
median	Statistical averaging method to use for stacked barplot: **TRUE** – median **FALSE** – mean
met_database	MET MySQL database. Most likely "**amet**".
met_network	MET monitoring network.
met_project	MET project name.
met_site_table	MET monitoring site MySQL table. Most likely "**stations**".
met_variable_col	MET variables, column names, from AQ project table.
num_ints	The number of color intervals to use for spatial plots. The script will ultimately determine the number of intervals, but **num_ints** can be set to increase or decrease the number of intervals.
overlay_opt	PAVE overlay times: **1** – hourly **2** – daily **3** – 1-hr daily max **4** – 8-hr daily max

Variable	Description
pid	Project name; must be unique across both MET and AQ.
plot_colors	Scatter plot symbol colors for primary simulation.
plot_colors2	Scatter plot symbol colors for secondary simulation.
plotfmt	Plot format, output type: "**PDF**" – pdf format "**PNG**" – png and pdf formats
plotsize	Scale factor to increase or decrease the size of a 541 x 700 pixel (**png**) or 8.5 x 11 inch (**pdf**) plot.
query	MySQL query to select data from database. In most cases, this is only part of the query. The complete query is constructed in the corresponding R script.
remove_negatives	Remove negative values: "**y**" or "**n**". default = "**y**".
remove_other	Remove "PM other" category from stacked bar plot analysis.
rmse_range_max	**RMS Error** range max limit: "**NULL**" – script defined limit
run_info_text	Include model run info as additional text to plots: "**y**" or "**n**"
run_name1	Project name; must be unique across AQ and MET.
run_name2	Second project name (allowed only in some scripts).
site	Plot label for when you are including only certain sites. Note that you will need to use an additional query to actually subset the data to these sites.
soccerplot_opt	Flag for soccer and bugle plot options: **1** – normalized mean bias/error **2** – fractional bias/error

Variable	Description
species	Chemical species to plot.
start_date	Start date of query, YYYYMMDD format.
start_hour	Start hour of query, HH format.
state	Plot label for indicating certain states. Note that you will need to use an additional query to actually subset the data to these states.
stat_file	File containing specific list of stations to analyze. User-defined.
stats_flags	Flags to determine which statistics are included on the **run_scatterplot.csh** script. Up to five statistics can be included, and are indicated by a '**y**'. Unused statistics are left blank. The order of the statistics flags is: **index of agreement (IA), correlation (r), RMSE, systematic RMSE, unsystematic RMSE, NMB, NME, Normalized Median Bias, Normalized Median Error, Mean Bias, Mean Error, Median Bias, Median Error, Fractional Bias, Fractional Error**
symb	Plot symbol: **15** – square **19** – circle
symbo	Plot symbol.
symbsiz	Plot symbol size: (**0.5** very small to **1.5** large). A value of **1** is recommended for most applications.
textstats	Produce text statistics file: **TRUE** or **FALSE**.
time.of.day.utc	Range of time (UTC) to isolate met and AQ data (e.g., compare average temperature and average PM for hours between 6 and 12 UTC).
use_avg_stats	Use time-averaged statistics: "**y**" or "**n**".
use_log	Use natural log transform: "**y**" or "**n**".

Variable	Description
zeroprecip	Include zero-precipitation obs: "**y**" or "**n**". (typically set to "**n**")

B.9 AQ Network Input File

In addition to the analysis input files which are script specific, AMET makes use of the input file **`$AMET/scripts_analysis/aqExample/Network.input`**. This file allows all of the network-specific processing to be handled in one location, and allows for easier addition of new networks into the analysis scripts.

Table B-9. AQ network input variables

Variable	Description
`inc_airmon_dep`	Include AIRMoN "**`network`**": "**`y`**" or "**`n`**".
`inc_aqs_1max`	Include AQS 1-hr max "**`network`**": "**`y`**" or "**`n`**".
`inc_aqs_1max_9cell`	Include AQS 1-hr max 9-cell average "**`network`**": "**`y`**" or "**`n`**".
`inc_aqs_8max`	Include AQS 8-hr max "**`network`**": "**`y`**" or "**`n`**".
`inc_aqs_8max_9cell`	Include AQS 8-hr max 9-cell average "**`network`**": "**`y`**" or "**`n`**".
`inc_aqs_hourly`	Include AQS hourly "**`network`**": "**`y`**" or "**`n`**".
`inc_castnet`	Include CASTNET "**`network`**": "**`y`**" or "**`n`**".
`inc_castnet_hr`	Include CASTNET hourly O_3 "**`network`**": "**`y`**" or "**`n`**".
`inc_improve`	Include IMPROVE "**`network`**": "**`y`**" or "**`n`**".
`inc_nadp`	Include NADP "**`network`**": "**`y`**" or "**`n`**".
`inc_search`	Include SEARCH "**`network`**": "**`y`**" or "**`n`**".
`inc_search_daily`	Include SEARCH daily "**`network`**": "**`y`**" or "**`n`**".
`inc_stn`	Include STN "**`network`**": "**`y`**" or "**`n`**".
`network_label`	Label of networks to be include in analysis.
`network_names`	Names of networks to be included in analysis.
`total_networks`	Total number of networks in analysis.

B.10 AQ Variables-Network Mapping

We provide below a mapping between monitoring networks, frequency (of observations), and chemical species that are included in the observed datasets, and that can be used in the model evaluation scripts. The units of all these variables were provided already in Section 4.2.

Table B-10. AQ networks, species, and frequency of observations

Network	Frequency	Chemical Species
AQS	hourly	O_3
CASTNET	weekly (hourly O_3)	SO_4, NO_3, NH_4, HNO_3, TNO_3, $S0_2$, O_3
IMPROVE	daily (every 3rd day)	SO_4, NO_3, NH_4, $PM_{2.5}$, EC, OC, TC
MDN	weekly	Wet deposition and concentration for Hg
NADP	weekly accumulated	Wet deposition and concentration for SO_4, NO_3, NH_4, and precipitation
SEARCH	hourly	SO_4, NO_3, NH_4, $PM_{2.5}$, HNO_3, NO, SO_2, O_3, CO, NO_Y
STN	daily (every 3rd day)	SO_4, NO_3, NH_4, $PM_{2.5}$, EC, OC, TC