

Chapter 15

PROGRAM CONTROL PROCESSING IN MODELS-3

Jeffrey Young*

Atmospheric Modeling Division
National Exposure Research Laboratory
U.S. Environmental Protection Agency
Research Triangle Park, NC 27711

ABSTRACT

Based on user choices and input, the Models-3 framework creates objects and include files that control the complete problem domain under which a CMAQ model is built and executed. The general concepts underlying the definition of these objects and the creation of the include files are discussed.

*On assignment from the National Oceanic and Atmospheric Administration, U.S. Department of Commerce. Corresponding author address: Jeffrey Young, MD-80, Research Triangle Park, NC 27711. E-mail: yoj@hpcc.epa.gov

15.0 PROGRAM CONTROL PROCESSING IN MODELS-3

Program control processing (PCP) refers to setting up internal arrays and mappings, global parameters, and data linkages to establish the complete problem domain in which the programs, or “models,” in the Community Multiscale Air Quality (CMAQ) system are built and executed. Controlled by user choices, the Models-3 framework carries out this processing automatically and provides the key to modularity and ensures uniform, consistent internal dimensioning, looping parameters and solver data in the compiled codes for all the CMAQ system models. Thus, modularity and consistency in establishing the computational domain and chemical mechanism for any particular application of the CMAQ system are assured by PCP. Modelers can easily and relatively safely carry out many different applications at different scales, or with different chemistry, or different solvers. Model developers can readily study the effects of different schemes and determine optimal implementations of the science and codes. PCP helps to establish a true “one-atmosphere” approach to modeling.

PCP involves both the Models-3 framework's graphical user interface and specialized processors that are launched from the framework. In the Models-3 framework, a user selects the computational grid and domain characteristics, the vertical layer structure, and the chemical mechanism to be used in building (compiling and linking) a model (an executable). How these choices get transformed automatically into compiled code is discussed in the following sections. This chapter focuses on the CMAQ Chemical Transport Model (CCTM) since the CCTM requires the most extensive use of PCP. However, some or all of the user choices made to set up a particular CCTM determine the particular application characteristics for all of the supporting processors (or “models”). In addition to the CCTM, CMAQ is comprised of eight additional processors as follows:

- The Models-3 emissions processing and projection system (MEPPS), described in Chapter 4.
- The emissions-chemistry interface processor (ECIP), described in Chapter 4.
- The plume-in-grid dynamics processor (PDM), described in Chapter 9.
- The meteorology-chemistry input processor (MCIP), described in Chapter 12.
- The land-use processor (LUPROC), described in Chapter 12.
- The initial and boundary conditions processors (ICON and BCON), described in Chapter 13.
- The photolytic J-value processor (JPROC), described in Chapter 14.
- The process analysis processor (PROCAN), described in Chapter 16. PROCAN is launched by the Models-3 framework and its execution is transparent to the user. The outputs from PROCAN are used to control CCTM diagnostic outputs (see Section 15.3.1 below).

Building a particular model in the Models-3/CMAQ system requires the creation of Fortran include files that populate the model's codes to set the specific computational grid and domain characteristics, the vertical layer structure, the chemical mechanism, and (for the CCTM) the process analysis to be used for the particular model. Once these model configuration objects are set by the user in the Models-3 Science Manager, the Models-3 framework automatically generates the required global Fortran include files. The use of global Fortran include files, the notion of "Global Name Table Data," and the design concepts that govern this implementation are discussed in Chapter 18, Sections 18.5 and 18.7.

15.1 Domain Configuration

The computational domain configuration data are contained in three include files:

- HGRD.EXT - The horizontal grid dimensions in terms of the number of grid cell columns and rows and the number of grid cells for which the computational domain boundary is extended for boundary data. The number of these perimeter boundary cells is also specified.
- VGRD.EXT - The vertical layer dimensions as the number of vertical layers. This include file also contains declarations for the layer surface and layer center arrays.
- COORD.EXT - The domain coordinate data:
 - ! The map projection type (Lambert, Mercator, Stereographic, UTM, or latitude-longitude).
 - ! The map projection parameters.
 - ! The center of the grid's coordinate system with respect to the "mother grid" or parent grid (usually the main grid from the meteorology pre-processing).
 - ! The horizontal grid cell sizes in meters.
 - ! The vertical layer type. There are 7 currently defined:
 1. σ_p coordinates.
 2. $\sigma_{p(0)}$ coordinates.
 3. σ_z coordinates.
 4. η_p coordinates.
 5. Pressure coordinates.
 6. Z coordinates (meters above ground).
 7. H coordinates (meters above mean sea level).
 - ! The units of the top vertical layer value.
 - ! The value of the top vertical layer variable.
 - ! The units of the vertical layer surface values.
 - ! The list of vertical layer surface values.
 - ! The list of vertical layer surface values converted to values monotonically increasing with altitude.
 - ! The grid name used for header description data in I/O API files (discussed in Section 15.2).

15.2 Input/Output Applications Programming Interface

Input and output data access for the CMAQ system is accomplished mainly through the Input/Output Applications Programmer's Interface (I/O API) sub-system [1]. The CMAQ system implementation of the I/O API is described in Chapter 18, Section 18.3. In order to access I/O API files and use other I/O API functions, it is necessary to specify the required declarations and parameters. Some or all of the following include files are needed in each subroutine that uses the I/O API:

- PARMS3.EXT - I/O API dimensioning parameters, file type and file characteristic parameters, coordinate system and map projection parameters, and vertical layer parameters.
- FDESC3.EXT - Fortran common blocks that contain a complete I/O API file description. The data in these common blocks are loaded from an I/O API file when opened and read. These data must be supplied from the user subroutine when an I/O API file is opened for writing.
- IODECL3.EXT - Declarations and usage comments for I/O API functions and routines.
- XSTAT3.EXT - Exit codes for the I/O API M3EXIT function. Generally, M3EXIT is called from any subroutine that produces a fatal error during I/O API access.

15.3 Other CCTM Configuration Control

15.3.1 CCTM Process Analysis

The Models-3/CMAQ system provides a diagnostic tool that allows a user to probe into the way the science processing is actually being executed in the CCTM. The process analysis tool optionally provides two types of information, which are termed Integrated Process Rates (IPR) and Integrated Reaction Rates (IRR). Note that these diagnostic tools apply only for the CCTM and not for the other models. The reader is referred to Chapter 16 for the details, but briefly the IPR processing captures the changes to the concentration field for different species, or groupings of species, for each individual science process modeled. The IRR processing focuses on the gas-phase chemistry and allows a detailed examination of various characteristics of the chemical mechanism implemented in the CCTM for the chosen scenario. The data produced by these tools are collected in I/O API output files, which can be further processed or examined using the visualization tools.

The CCTM requires three include files, even if neither IPR nor IRR processing is selected.¹ These include files are automatically generated from PROCAN, and the Models-3 framework then incorporates them in the CCTM build process. The three IPR/IRR include files are:

¹ If process analysis is not requested while building a CCTM model executable, the Models-3 framework automatically supplies the include files.

- PA_CTL.EXT - The declarations and values of the two flags that determine if IPR or IRR (or both) processing is to be performed.
- PA_CTM.EXT - Declarations and parameters that define the domain, output file descriptions, arrays and dimensions, IPR and IRR loop variables, and named common blocks for computational data.
- PA_DAT.EXT - Data statements containing values for the variables needed to fill in the common blocks in PA_CTM.EXT.

15.3.2 CCTM Fixed Data

The Models-3/CMAQ system can deal with other types of global include files and reference them from a user-supplied full path to build a model. The following three categories of global include files apply to the CCTM. The other processor models (MCIP, ECIP, etc.) also have include files in this category, but not as extensively.

- FILES_CTM.EXT - The set of logical file names used in the current CCTM implementation. The include file contains data statements for the file names as variables used in the codes and values for the variables. The values are character strings that are UNIX environment variable values set in the scripts launched by the Models-3 framework to run the models.
- CONST.EXT, CONST3_RADM.EXT - These include files contain parameter statements to define basic and frequently used air quality and meteorological modeling physical and mathematical constants. Inclusion of these files in the model codes helps to assure consistency in the science calculations across all the processes and modules.
- BLKPRM_500.EXT - Computational blocking: For computational efficiency and reduced memory requirements, operations may be performed on groups of grid cells (a block of cells) at a time. This include file provides convenient definitions to use 500-cell blocking.
- GRID_DECL.EXT - Declarations for the dimensions and species classes offsets in the main concentration array. See Chapter 18, Section 18.2.2 for details.
- EMISPRM.chem.EXT, EMISPRM.vdif.EXT - These two include files contain the declarations that control in which process the emissions sources are injected; either the vertical diffusion process of the gas-phase chemistry solver process. By using a C pre-processor (cpp) #ifdef directive, one or the other include file is actually included when the code is compiled. The cpp directive is set by the user during the model building phase. The reader is referred to Chapter 18, Section 18.5 for more details.

15.4 Generalized Chemistry

15.4.1 Design

In the past, a particular chemical mechanism was generally “hardwired” into a chemical transport model with mechanism parameters and variables embedded in the solver codes. Implementing even minor changes incurred a high potential for code and consistency errors. Additionally, some mechanism changes necessitated extensive coding changes, requiring careful tracing through all the mechanism dependencies. Worse, not finding these dependencies led to errors that were not necessarily detected. The Models-3/CMAQ system employs a generalized chemical mechanism processor (MP), also called the “mechanism reader,” which greatly simplifies the task of implementing or altering gas-phase chemical mechanisms and provides the capability of easily and safely using different mechanisms in the CMAQ system.

MP reads an ASCII file that contains a symbolic description of a gas-phase chemical mechanism, and generates three ASCII Fortran source files. The input ASCII file is called a *mechanism description*, or simply, “mechanism.”

The ASCII mechanism description file is formatted according to a simple set of rules in a free-form format similar to the approaches used by Jeffries et al. [2] and Gery and Crouse [3]. The types of reactions that are supported are described below in Section 15.4.3, and the specific formats are described in the Models-3 User Manual, Appendix M [4].

The Fortran files that are generated are incorporated as include files in the CMAQ Fortran codes. They consist of:

- Fortran named common blocks that declare parameters and explicitly dimensioned arrays associated with gas chemistry kinetic and photolytic reactions.
- Fortran data statements that fill in the common block arrays with actual values determined from the input ASCII mechanism file.

The common blocks and the data statements files are global and are used in all the codes that require gas chemistry information.

Another file that is generated from the mechanism description file by MP, which is transparent to the user, is a species list found in the mechanism description. This file is used by the framework to generate all the gas chemistry species global include files that are required in the model codes.

In order to implement the mechanism reader's capabilities, the CMAQ system requires generalized gas-phase chemistry solvers. At present, there are two such solvers available, SMVGEAR and QSSA, which are described in detail in Chapter 8. The use of generalized solvers precludes some code optimizations that can increase performance, but significantly facilitates implementing new or altered chemical mechanisms.

15.4.2 Operation

The CMAQ system accounts for chemistry in three forms: gas phase, aerosols (liquid and solid phase), and the aqueous phase. Therefore, the Models-3 framework controls certain aspects of how the model species are treated in a model simulation and provides linkages between species in different phases. Setting up the linkages is accomplished by means of a series of four species tables in a spread sheet format that are filled out by the user. These tables also allow the user to control some aspects of the simulation and generated outputs and provide links to data such as emissions, initial and boundary concentrations, and dry deposition velocities that are generated by the CMAQ input processors. The mechanism definition and species tables become objects that the Models-3 framework stores in its database. Therefore if a required chemical mechanism already exists in the database as a predefined mechanism, the user does not have to enter any mechanism data into the framework.

Since different gas-phase chemical mechanisms can be used in the CMAQ system and these mechanisms may employ different species names, it is necessary to supply the linkages among the gas-phase species, the fixed aerosol species, and the species that participate in aqueous-phase chemistry. Similarly, it is also necessary to link the gas-phase species to emissions, deposition velocities, and to aqueous-phase scavenging. In addition, the species tables allow the user to select which species concentrations are written to the output files and whether or not they are to be advected or diffused. In the same manner, most of these linkages must be established for aerosol, non-reactive, and possibly tracer species.

The methodology used to establish links between species names involves the concept of surrogate names. Surrogate names are used to provide linkages between model species within the CCTM and to link the model species to those that represent data provided by other CMAQ processors. For example the emissions-chemistry interface processor, ECIP, that links the Models-3 emissions processing and projection system (MEPPS) with the CCTM could write a species named XXX that represents xxx emissions rates. However, the CCTM that uses xxx may have the corresponding model species named YYY. Using the surrogate name concept, emissions species XXX gets mapped to model species YYY. The reader is referred to the Models-3 User Manual [4] for more details on the implementation of the set of predefined surrogate names used in the current version.

An additional feature that can be applied to any of the surrogate species linkages is the application of a multiplicative, or scale factor. The user can set the factor to be other than unity (the default) to easily modify some model input data related to the selected species and process linkage. For example, this could be used to test the effect of changing the deposition velocity for a certain species, or to modify initial or boundary conditions data. Also, the factors can be set to modify certain data between model species groups within the CCTM, such as in the gas-phase to aerosol linkage.

These linkages are set by the user and are distributed among the four tables that establish the linkages among gas-phase, aerosols, non-reactive, and (possibly) tracer species. Non-reactive species are gas-phase, but do not participate in gas-phase reactions. Optional tracer species are inert and may have special, user-defined initial and boundary concentrations as well as emissions

sources. The reader is referred to the Models-3 User Manual [4] for more details and a complete description of the species tables.

The following is a list of the linkages set up in the current implementation of the Models-3/CMAQ system. The linkages take place with respect to the CCTM or model species name; the surrogate names and flags point to this name.

1. Model species molecular weight.
2. Emissions species surrogate name.
3. Emissions species scale factor.
4. Initial and boundary concentrations species surrogate name.
5. Initial and boundary concentrations species scale factor.
6. Deposition velocity species surrogate name.
7. Deposition velocity species scale factor.
8. Gas-phase to aerosol linkage species surrogate name.
9. Gas-phase to aqueous-phase linkage species surrogate name.
10. Gas-phase aqueous scavenging linkage species surrogate name.
11. Gas-phase aqueous scavenging linkage species scale factor.
12. Aerosol to aqueous-phase linkage species surrogate name.
13. Aerosol aqueous scavenging linkage species surrogate name.
14. Aerosol aqueous scavenging linkage species scale factor.
15. Non-reactive to aerosol linkage species surrogate name.
16. Non-reactive to aqueous-phase linkage species surrogate name.
17. Non-reactive aqueous scavenging linkage species surrogate name.
18. Non-reactive aqueous scavenging linkage species scale factor.
19. Model species flag for participation in the advection processes.
20. Model species flag for participation in the diffusion processes.
21. Model species flag for inclusion in the concentration output file.
22. Model species flag for inclusion in the dry deposition output file.
23. Model species flag for inclusion in the wet deposition output file.
24. Tracer species - similar linkages to any of the above.

It should be noted that items 1 - 7 and 19 - 23 apply to all of the four groups of species (gas phase, aqueous phase, solid and liquid aerosols).

15.4.3 Supported Reaction Types

In this section we describe the types of gas-phase chemical reactions supported by the generalized mechanism processor, MP, in the current version of the Models-3/CMAQ system. The reader is referred to Chapter 8 for a description of the available mechanisms.

The gas-phase mechanism description file is a list of mechanism reactions. The mechanism reactions list consists of lines of symbolic descriptions of elementary reactions among modeled chemical species, followed by a rule-based symbolic description of the reaction rate expression for that reaction. The reader is referred to Chapter 8 for mechanism description examples.

These mechanism reaction lists are generally “free-form,” in that spacing of the symbols on the page is not important (except, perhaps for readability), and reaction descriptions may span many lines if necessary. Liberal use of comments is facilitated by allowing for comment lines and comments embedded within the reaction lines. In addition, each reaction may be labeled for subsequent reference. The photolysis reactions are given labels for a specific photolytic table reference. These tables are generated by the J-value processor and are used in the CCTM for the gas-phase chemistry processing. Currently, there are 10 different rate constant expressions available for thermal reactions, including Arrhenius-type expressions and the class of so-called fall-off reactions (**Type f**, below). New expressions can be added, as necessary.

The type of reaction rate expressions that can be calculated are:

- Type 1 :** $k = A$
- Type 2 :** $k = A(T/300)^B$, where $T =$ temperature in deg K.
- Type 3 :** $k = Ae^{(-C/T)}$, where $C = E_a / R$,
 E_a is the activation energy, and
 R is the gas constant.
- Type 4 :** $k = A(T/300)^B e^{(-C/T)}$
- Type 5 :** $k = k_f = k_f / Ae^{(-C/T)}$, where k_f is any previously defined forward reaction.
- Type 6 :** $k = Ak_n$, where k_n refers to the n^{th} reaction.
- Type 7 :** $k = A(1.0 + 0.6P)$, where $P =$ pressure in atmospheres.
- Type 8:** $k = k_0 + \frac{k_3[M]}{1 + \frac{k_3[M]}{k_2}}$, where $k_0, k_2,$ and k_3 are **Type 3**.
- Type 9:** $k = k_1 + k_2[M]$, where k_1 and k_2 are **Type 3**.
- Type f:** $k = k = \frac{k_0[M]}{1 + k_0[M] / k_i} F^G$
 where k_0 and k_i may be **Type 1, 2, 3,** or **4**,
 $G = \{ 1 + [\ln(k_0 [M] / k_i) / n]^2 \}^{-1}$,
 $F = 0.6$ (usually), and
 $n = 1.0$ (usually).

15.4.4 Mechanism Parsing Rules

The mechanism description file that MP reads must follow a set of format rules in order to describe the reactions listed above. Generally, reserved ASCII characters and keywords on one line (which may wrap around the page) represent parts of a reaction, including species reactants and products, stoichiometric coefficients and rate expressions. In addition, labels are supported, which function to allow one reaction to refer to another and to fix photolytic reaction table names for photolytic reactions. As mentioned above, the reader is referred to Chapter 8 for mechanism description examples. Also, Appendix M of the Models-3 User Manual contains a complete description of these formats.

15.4.5 Chemical Species Include Files

As described above, MP reads and processes the mechanism description file, and one of the outputs it generates is a file that contains the list of chemical species found in the mechanism. The Models-3 framework reads this species file to initialize the gas-phase species table in a format similar to a spreadsheet, which the user completes by inserting data such as molecular weights, surrogate names and factors, and processing flags. The initial set-up presented to the user consists of all blank entries in each spreadsheet cell, but with the first column filled in with the species found in the MP-generated species file. Thus all the species in the mechanism are listed in the order in which they were found in the mechanism description file. The top row is also filled in with category headers that are related to the linkages described above and that determine the framework-generated Fortran include files to be used in the codes.

There are three additional tables that are initialized for aerosols, non-reactive and tracer species. They do not have the first column set up, but the first row is set as in the gas-phase table. It is up to the user to fill in the first column, as with all the other spreadsheet cells. Appendix M of the Models-3 User Manual describes the details of how to enter data into these tables.

The following sections describe all the standard include files that are generated from the species tables and the behavior of the models when accessing the data declared in the include files.

15.4.5.1 Gas-Phase Reactions

The gas-phase reactions data are required in the CCTM by the gas-phase chemistry solvers and the subroutine PHOT, which calculates the photolysis rate constants. These data are also required in JPROC, the processor model that calculates the table-based photolysis rates (see Chapter 14). There are two include files associated with this data:

- RXCM.EXT - contains all the declarations, parameter statements and array definitions associated with the gas-phase chemistry reactions. This file also contains three named common blocks for memory allocation of the data from RXDT.EXT.
- RXDT.EXT - contains data statements associated with the parameters and arrays declared in RXCM.EXT. These data fill the arrays in the named common blocks in RXCM.EXT.

The CCTM driver program must include both of these files in its declaration section. The subroutines in the gas-phase chemistry solver process module must include only the common block file, RXCM.EXT. That goes for the subroutine PHOT, as well. PHOT is called from the gas-phase chemistry “class-driver,” the top-level subroutine (see Section 18.2.2 in Chapter 18). The CCTM driver, by including both files, causes the common blocks to be loaded with the necessary data used by all the modules that need the chemistry data, including the process analysis module, PROCAN.

The remaining sections describe the global include files associated with the four species tables. As mentioned above, these files are automatically generated by the Models-3 framework, according to the user-specified linkages and flags, and are targeted for inclusion in all model routines that require the specific chemistry data in them. For additional details the reader is referred to the Models-3 User Manual [4]. Each of the include files contains declarations and parameter statements for the loop index and array dimensioning for each particular species group. If a particular species group is not used (e.g., aerosols or tracers), the include file contains declarations and parameter statements that set the species loop counter to zero and its array dimensioning parameter to one, thus maintaining complete generality and modularity within the codes. In addition, by setting the dimensioning value to one (Fortran does not allow zero), the compiled codes do not waste memory, a typical problem when coding in Fortran with some pre-determined maximum dimension to hopefully account for all cases.

15.4.5.2 Model Species

The framework generates four model species include files that together contain the names and molecular weights of all the chemical species available globally to the model.

The model species include files are:

- GC_SPC.EXT - gas-phase model species names and molecular weights.
- AE_SPC.EXT - aerosol species names and molecular weights.
- NR_SPC.EXT - non-reactive species names and molecular weights.
- TR_SPC.EXT - tracer species names and molecular weights.

The next three sections describe include files that are associated with data access by the CCTM from external files. The data are referenced from I/O API files by means of file variable names that are contained in headers in each of the files.

15.4.5.3 Emissions

The CCTM reads the emissions data from an I/O API file produced by ECIP using the surrogate name concept discussed above. The data are read by file variable names (surrogates) and stored in arrays that are linked to the corresponding model species names.

The CCTM has the option of emissions sources injected as part of the vertical diffusion process or as part of the chemistry solver process, where the injected sources are treated as part of the chemical production terms. To read source emissions data into the CCTM for processing either in the vertical diffusion or in the gas-phase chemistry modules, the following include files are required:

- GC_EMIS.EXT - gas-phase emissions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
- AE_EMIS.EXT - aerosol emissions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
- NR_EMIS.EXT - non-reactive emissions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
- TR_EMIS.EXT - tracer emissions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).

15.4.5.4 Initial and Boundary Conditions

For both the initial and boundary concentrations the CCTM reads I/O API input data by variable name reference. The processing first checks to see if the model species name is on the file and reads the data associated with that name. If the model species name is not on the file, it checks for the surrogate name. If that, too is not available, the input value is set to a minimum value (“model zero”). Additional information on initial and boundary conditions and how they are implemented in the Models-3/CMAQ system can be found in Chapter 14.

- GC_ICBC.EXT - gas-phase initial and boundary conditions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
- AE_ICBC.EXT - aerosol initial and boundary conditions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
- NR_ICBC.EXT - non-reactive initial and boundary conditions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
- TR_ICBC.EXT - tracer initial and boundary conditions surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT)

15.4.5.5 Dry Deposition

The following include files are used to read in dry deposition velocities from an I/O API file produced by the meteorology-chemistry input processor, MCIP. Using the surrogate name concept discussed above, the data are read by file variable names (surrogates) and stored in arrays that are linked to the corresponding model species names.

- GC_DEPV.EXT - gas-phase deposition velocity surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
- AE_DEPV.EXT - aerosol deposition velocity surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
- NR_DEPV.EXT - non-reactive deposition velocity surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
- TR_DEPV.EXT - tracer deposition velocity surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).

15.4.5.6 Wet Scavenging

In both the scavenging and the cross-phase linkages it is necessary to set up a mapping from the gas-phase species to the generic species names that are associated with aerosols and aqueous-phase chemistry and to the names of species that are absorbed by cloud and rain water. Because of the generality with respect to the gas-phase species names, these generic names are set in the subroutines associated with aerosols, aqueous chemistry, and removal by in-cloud and precipitation scavenging. They are linked to the model species names by employing the surrogate name concept described above.

- GC_SCAV.EXT - gas-phase scavenging surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
- AE_SCAV.EXT - aerosol scavenging surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
- NR_SCAV.EXT - non-reactive scavenging surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).

- TR_SCAV.EXT - tracer scavenging surrogate species names with scale factors and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).

15.4.5.7 Cross-Phase Linkage

- GC_G2AE.EXT - surrogate names for gas-phase model species that participate in aerosol chemistry with scale factors and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
- GC_G2AQ.EXT - surrogate names for gas-phase model species that participate in aqueous-phase chemistry with scale factors and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
- AE_A2AQ.EXT - surrogate names for aerosol species that participate in aqueous-phase chemistry with scale factors and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
- NR_N2AE.EXT - surrogate names for non-reactive species that participate in aerosol chemistry with scale factors and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
- NR_N2AQ.EXT - surrogate names for non-reactive species that participate in aqueous-phase chemistry with scale factors and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
- TR_T2AE.EXT - surrogate names for tracer species that participate in aerosol chemistry with scale factors and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).
- TR_T2AQ.EXT - surrogate names for tracer species that participate in aqueous-phase chemistry with scale factors and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).

15.4.5.8 Operational Choices

The Models-3 framework provides the capability of allowing the user to control how some of the processing is carried out with respect to the chemical species in the CCTM. The following lists describe the include files that the framework generates for the various processes:

1. **Advected species:** In order to save memory storage or to minimize computation, the user may choose not to advect certain modeled species such as some radicals. In general, the special “counter species” used in a mechanism should not be advected.
 - GC_ADV.EXT - names of gas-phase model species that are advected and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).

- AE_ADV.EXT - names of aerosol species that are advected and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
 - NR_ADV.EXT - names of non-reactive species that are advected and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
 - TR_ADV.EXT - names of tracer species that are advected and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).
2. **Species that undergo diffusive processes:** The comment above, for advected species, applies also to diffused species.
- GC_DIFF.EXT - names of gas-phase model species that are diffused and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
 - AE_DIFF.EXT - names of aerosol species that are diffused and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
 - NR_DIFF.EXT - names of non-reactive species that are diffused and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
 - TR_DIFF.EXT - names of tracer species that are diffused and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).
3. **Species that are saved to the dry deposition output file:** Generally, all the dry deposition species that are modeled would be saved to the output file, but a user might elect to reduce the number and save only a subset.
- GC_DDEP.EXT - names of gas-phase model species that are written to the dry deposition output file and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
 - AE_DDEP.EXT - names of aerosol species that are written to the dry deposition output file and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
 - NR_DDEP.EXT - names of non-reactive species that are written to the dry deposition output file and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
 - TR_DDEP.EXT - names of tracer species that are written to the dry deposition output file and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).
4. **Species that are saved to the wet deposition output file:** See the comment in item 3.
- GC_WDEP.EXT - names of gas-phase model species that are written to the wet deposition output file and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
 - AE_WDEP.EXT - names of aerosol species that are written to the wet deposition output file and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).

- NR_WDEP.EXT - names of non-reactive species that are written to the wet deposition output file and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
- TR_WDEP.EXT - names of tracer species that are written to the wet deposition output file and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).

5. Species that are saved to the concentration output file: See the comment in item 3.

- GC_CONC.EXT - names of gas-phase model species that are written to the concentration output file and indices that point to the positions of the species in the model species name table (in GC_SPC.EXT).
- AE_CONC.EXT - names of aerosol species that are written to the concentration output file and indices that point to the positions of the species in the model species name table (in AE_SPC.EXT).
- NR_CONC.EXT - names of non-reactive species that are written to the concentration output file and indices that point to the positions of the species in the model species name table (in NR_SPC.EXT).
- TR_CONC.EXT - names of tracer species that are written to the concentration output file and indices that point to the positions of the species in the model species name table (in TR_SPC.EXT).

15.4.5.9 Tracer Species

The use of tracer species is purely user-determined. An application with tracers can provide the modeler with insights into how the model is simulating various physical processes, like advection or diffusion. If tracer species are to be modeled, the user must have created the special table entries appropriate to the application. In addition special data, such as emissions or initial and boundary tracer concentrations, must be created in the corresponding I/O API files. The specialized data creation is outside the scope of the Models-3 framework and must be carried out by the user in conjunction with the application that is being modeled.

15.6 Conclusion

In this chapter we have described how PCP helps to establish modularity and consistency for particular applications of models in the CMAQ system. The Models-3 framework processing, through PCP, enables model developers and model users to study the effects of different implementations of the science and codes or to execute different applications at various scales with different chemical mechanisms or with different numerical solvers. The design and implementation of the concepts in the program control processing helps to establish a true “one-atmosphere” approach to modeling.

15.7 References

- [1] “The EDSS/Models-3 I/O API” http://sage.mcnc.org/products/I/O_API/

- [2] Jeffries H. E. (1990), "User's Guide to Photochemical Kinetics Simulation System PC-PKSS Software Version 3," Chapel Hill, N.C. 27514
- [3] Gery M. W. and R. R. Crouse (1990), "User's Guide for Executing OZIPR," EPA/600/8-90/069, U. S. Environmental protection Agency, Research Triangle Park, NC 27711
- [4] EPA Third-Generation Air Quality Modeling System Models-3 Volume 9b, User Manual, Appendices, June 1998, EPA-600/R-98/069(b)

This chapter is taken from *Science Algorithms of the EPA Models-3 Community Multiscale Air Quality (CMAQ) Modeling System*, edited by D. W. Byun and J. K. S. Ching, 1999.