CONCEPT Emissions Modeling User's Guide
Introduction

Table of Contents

List of Tables and Figures

CONCEPT Emissions Modeling User's Guide

Introduction

This document is the User's Guide for the CONCEPT Emissions Modeling System. The first chapter contains an overview of the CONCEPT modeling system, along with system requirements and installation instructions. The following chapters each address one of the major emissions modeling components:

1.0 Overview of the CONCEPT System

The CONCEPT system is being developed as a suite of independent models that utilize common supporting routines and formats. The intent of this development effort is to create a modeling structure that allows the user maximum flexibility is selecting and applying all or part of the modeling system to meet individual modeling requirements.

1.1 Hardware and software requirements

The CONCEPT system requires a number of software packages to be installed prior to the installation of the CONCEPT model itself.

- **PostgreSQL** v.7.4.2 and v.7.4.3 have been tested: v.7.4.3 on a Solaris box; and v.7.4.2 on a Red Hat linux box. We recommend avoiding v.8.0.1 since the mail-list traffic indicates that problems exist in v.8.0.1. Further, only the PostGIS release candidates will work with v.8.0.1 and they are all still buggy (see PostGIS below). Source code can be downloaded from www.postgresql.org/ftp/source/. RPMS for Red Hat and Fedora can be downloaded from www.postgresql.org/ftp/binary/; SuSE from ftp.suse.com/pub/projects/postgresql/postgresql-7.4.7/ (note that this is a newer version, but we expect no problems).
- **PostGIS** v.0.9.1 (avoid, avoid, avoid any of the release candidates as they are still buggy. Further, the v.1.0.0-rcX candidates are only compatible with PostgreSQL v.8.0.1, which as stated above, is still not stable.). The source code can be downloaded from postgis.refractions.net/download.php. RPMS for Fedora, which is also supposed to be compatible with Red Hat, can be found at mappinghacks.com/rpm/fedora/.
- **PROJ.4** v.4.4.7 (v.4.4.8 under testing and v.4.4.9 is available). Although CONCEPT has been developed with v.4.4.7 and v.4.4.8, we have no reason to believe that v.4.4.9 will not work. The source can be downloaded from ftp.remotesensing.org/pub/proj/. Make sure to download ftp://ftp.remotesensing.org/pub/proj/proj-nad27-1.2.tar.gz (instructions for installation at www.remotesensing.org:16080/proj) to support datum conversions in North America. The RPMS for Fedora is available at mappinghacks.com/rpm/fedora/2/ (or /3/) – this supposedly will also work for Red Hat, and for SuSE at www.gdf-hannover.de/software/suse/9.1/RPMS/.
- **GEOS** v.2.1.0 (v.2.1.1 under testing). The source code can be downloaded at geos.refractions.net/. RPMS for Fedora is available at mappinghacks.com/rpm/fedora/2/ (or /3/) – this supposedly will also work for Red Hat, and for SuSE at www.gdf-hannover.de/software/suse/9.1/RPMS/ (though this is an older version of GEOS for SuSE).
- **ActivePerl** v.5.8 (many implementations exist). Perl source is available at aspn.activestate.com/ASPN/Downloads/ActivePerl/Source. Make sure that when you compile Perl that you also compile for the DBI. You will also need the DBD module from www.cpan.org/modules/by-module/DBD. ActveState also maintains RPMS for

multiple Linux systems (www.activestate.com/Products/ActivePerl/), but they also require you to register before you download the packages.

Be aware that if you compile these codes from source, you will likely need at least one of the following packages (also freeware and/or available under GPL): GNU C (ftp://mirrors.usc.edu/pub/gcc); Ant (ant.apache.org/srcdownload.cgi); GNU make (ftp://mirrors.usc.edu/pub/make); GNU readline ftp://mirrors.usc.edu/pub/readline); and GNU awk (ftp://ftp.gnu.org/pub/gnu/gawk).

In addition to these core packages CONCEPT may also require:

- **FORTRAN compiler**. The MOBILE6 and NONROAD models are FORTRAN based. The CONCEPT distribution contains compiled executables, but the models may need to be re-compiled on some systems. Also, the meteorological data converter and the CMAQ data output routines require FORTRAN modules. Th CAMQ output routines will require the code to be compiled and properly linked to the IO/API libraries, so a FORTRAN compiler is required to produce CMAQ formatted output.
- **IO/API with NCAR netCDF libraries.** The IO/API is required for production of CMAQ formatted output.

1.2 CONCEPT Installation Instructions

The CONCEPT installation file should be untarred in the directory where CONCEPT is to be installed.:

tar xvf concept_v1_042005.tar

This creates the concept and concept_projects directories. All system files are contained under the concept directory, while all input data is located under concept_project.

After the CONCEPT script, there area a number if steps that may need to be taken, depending on the system, and which options CONCEPT will be using:

1) Compile IO/API interface (required if creating CMAQ output formats)
The CMAQ creation routines require the users to compile with the IO/API libraries installed. The make files for this are located in:
/concept/src/lcmaq/ow2ncf and
/concept/src/cmaq/pts2ncf
These make files need to altered to reflect the location of the IO/API libraries.

2) Compile the MOBILE6 and ONOROAD models. The CONCEPT distribution contains executables for the MOBILE6 and NONROAD for the LINUX system.

If CONCEPT is installed on other platforms, re-compiling the models may be required. The make files to do this are located in:

concept/src/mv/mobile6/AIR/Source and
concept/src/nonroad/nonroad_model/src.unix

3) Compile the MM5 met data interface codes. The CONCEPT distribution contains executables for the MM5 met interface for the LINUX system. If CONCEPT is installed on other platforms, re-compiling the interface may be required. The make files to do this are located in:

/concept/mm5_met_interface/src

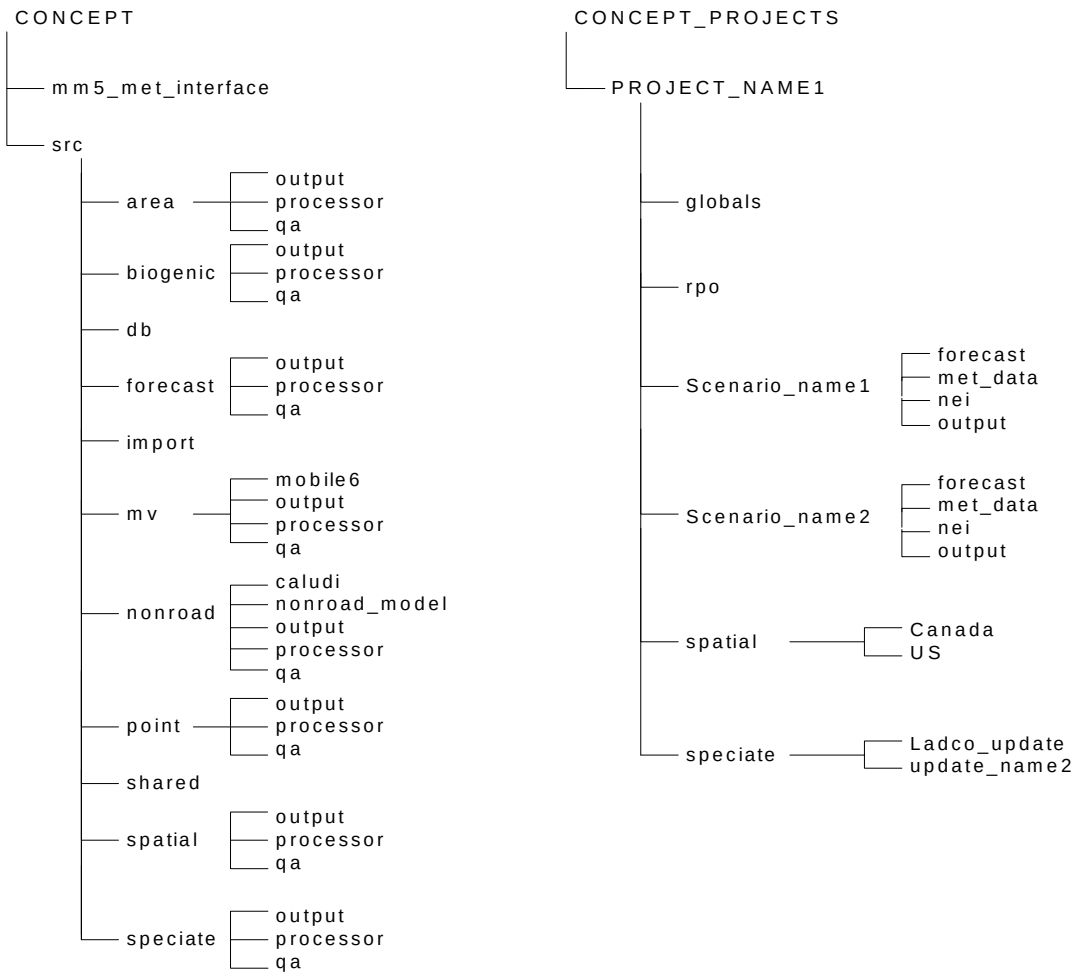
1.3 System Directory Hierarchy

The CONCEPT modeling system is divided into two separate directory structures: concept, which contains all of the source codes, executables, scripts, etc, and concept_project which contains all of the input data, output results and reports. Figure 1.3.1 shows the systems directory structure.

Execution of the CONCEPT system creates a POSTGRES database structure for the defined project name. This database is partitioned into a number of database schemas: scenario_name (may be multiple scenarios for each project), globals, xref, and spatial.

Figure 1.3.1

CONCEPT MODEL DIRECTORY STRUCTURES



1.4 Control File Requirements / Global Options

Episode specific options for each CONCEPT run are controlled through the CONCEPT run control file. The CONCEPT run_control file is located in the concept_projects/*project/scenario* directory. The file may contain blanks and comment lines (beginning with '#'). Notice that the path names for spatial surrogate shape coverages and MCIP meteorological data require an absolute path. Figure 1.1 is an example of a run_control.txt file.

FIGURE 1.4.1 Example Run_control .txt file

```
#
# Test run for Beta4
#
Description           = CONCEPT Kentucky test case
#
# General CONCEPT model parameters
#
DebugLevel            = 2
AreaQALevel           = 2
PointQALevel          = 2
StartDate             = 07/06/2002
NumDays               = 1
TimeZone              = EST
ChemicalMechanism     = CBIV
OutputFormat          = CMAQ
Pollutants            = NOX VOC CO SO2
#
# Grid definition
#
NumZCells             = 5
NumZCellsBelow        = 2
NumZCellsAbove        = 3
MbinMixingHeight       = 100
FMaxMixingHeight       = 1000
ElevatedPlumeCutoff   = 10
Gr`idName              = rpo36
Projection            = LAMBERT
UTMZone               = 12
GridDatum             = NAD83
Grid1Lat              = 33.0
Grid2Lat              = 45.0
GridLonOrigin         = -97.0
GridLatOrigin         = 40.0
OriginX               = -2736000
OriginY               = -2088000
NumXCells             = 146
NumYCells             = 112
XCellSize             = 36000
YCellSize             = 36000
#
#  Biogenic Model Parameters
#
BEISModel             = BEIS3
BEIS3Layers           = 5
#
#  Spatial Surrogate Input File Locations
#
SurrFileName=/otterp/disk3/concept_testing/beta_jgw/concept_projects/beta_jgw/scenario1/s
patial_coverages.txt
SurrDefiName=/otterp/disk3/concept_testing/beta_jgw/concept_projects/beta_jgw/scenario1/s
patial_surrogates.txt
#
#  MCIP Met data file locations (for CMAQ output format only)
```

```
#
MetCRO=/otterp/disk3/concept_testing/beta_v4_0311/concept/src/area/output/METCRO3D.v02_aa
a.36km
MetDOT=/otterp/disk3/concept_testing/beta_v4_0311/concept/src/area/output/METDOT3D.v02_aa
a.36km
#
# MV parameters - these are the spatial surrogates to use
# for spatially allocating starts and hpms vmt within
# each county.  They are currently set to the population
# surrogate.
#
starts_ssc            = 100
hpms_ssc              = 100
```

1.5 CONCEPT Script

CONCEPT is run entirely through the CONCEPT execution script. This script is located in the /concept directory, and contains commands to build the projects and scenarios and to run all of the model options. Also located in the /concept directory are script files (*.she) that demonstrate the calls required to run particular applications of the CONCEPT model. Figure 1.2 contains the CONCEPT command options available. This listing of options can also be obtained with the command: concept –help

Figure 1.5.1 CONCEPT commands

```
Project Set-up Commands:
  concept create_project   -n PROJECTNAME [-s SCENARIONAME]
  concept add_user         -n PROJECTNAME -u USERNAME
  concept init_project     -n PROJECTNAME
  concept drop_project     -n PROJECTNAME
  concept add_scenario     -n PROJECTNAME -s SCENARIONAME
  concept init_scenario    -n PROJECTNAME -s SCENARIONAME
  concept drop_scenario    -n PROJECTNAME -s SCENARIONAME
  concept import_globals   -n PROJECTNAME -d DIRECTORY [-t]
  concept import_rpo       -n PROJECTNAME -d DIRECTORY [-t]
  concept import_nei       -n PROJECTNAME -s SCENARIONAME -d DIRECTORY [-t]
  concept import_nei_area  -n PROJECTNAME -s SCENARIONAME -d DIRECTORY [-t]
  concept import_nei_point -n PROJECTNAME -s SCENARIONAME -d DIRECTORY [-t]
  concept import_mv        -n PROJECTNAME -s SCENARIONAME -d DIRECTORY [-t]
  concept import_control   -n PROJECTNAME -s SCENARIONAME -c CONTROLFILE
  concept import_sp        -n PROJECTNAME
  concept import_metdata   -n PROJECTNAME -s SCENARIONAME -d DIRECTORY [-t]
  concept run_grid_definition_model -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept import_spatial_coverages  -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept create_spatial_surrogates -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept build_speciation  -n PROJECTNAME -s SCENARIONAME -d DIRECTORY

Run Model Commands:
  concept qa_nei_area      -n PROJECTNAME -s SCENARIONAME
  concept qa_nei_point     -n PROJECTNAME -s SCENARIONAME
  concept qa_rpo_mv        -n PROJECTNAME -s SCENARIONAME
  concept run_area_model   -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_point_model  -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_biogenics_normal   -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_biogenics_temporal -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_mv_model      -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
                            -r REPCOUNTYFILE -m M6FILESDIR -x M6EXEDIR
```

```
  concept run_area_growth   -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_point_growth  -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_area_control  -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_point_control -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_point_cost    -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept run_nonroad_model -n PROJECTNAME -s SCENARIONAME -d DIRECTORY
  concept adj_nonroad_tmpr  -n PROJECTNAME -s SCENARIONAME
  concept run_nonroad_ouput -n PROJECTNAME -s SCENARIONAME -d DIRECTORY

Common options:
  -n PROJECTNAME    Project name (will be used as the database name so must
                    conform to PostgreSQL database naming requirements).
  -s SCENARIONAME   The name for the scenario (will be used as the schema name
                    for the scenario tables so must conform the PostgreSQL
                    schema naming requirements).
  -u USERNAME       The user name that needs access to the project database.
  -d DIRECTORY      The directory (full path) from which to load data or to
                    which output files should be written.
  -c CONTROLFILE    Initialize the project by loading the project control file,
                    the global reference files, and the RPO cross-reference
                    files.  The project control file is specified by
                    PROJECTCTLFILE.  The global reference data files and the
                    RPO data files are listed in the project control file.
  -r REPCOUNTYFILE  The file containing the specific inputs for MOBILE6 by
                    representative county.
  -m M6FILESDIR     The directory (full path) to which the MOBILE6 input file
                    should be written.
  -x M6EXEDIR       The directory (full path) to the mobile6 executable.
  -t                An optional parameter for most imports, tells CONCEPT to
                    use transactions to speed up imports.  Using transactions
                    is faster, but the import will abort if any errors occur.
  -h, -?, --help    Show this help, then exit.
  -V, --version     Output version information, then exit.
```

1.6 Setting Up a Project

Before running the CONCEPT models, the CONCEPT database and scenarios must be created, and the lookup and cross-reference data must be loaded into the system. The init_beta4.sh script contains the commands required to initialize the modeling system and load the lookup data. The following example initializes a project called beta4 with a scenario called scenario1. These commands can be run as a batch script, or individually from the command line.

Init_beta4.sh

```
#!/bin/sh
exec > log.init_beta4 2>&1
date

concept drop_project -n beta4
concept create_project -n beta4
concept init_project -n beta4
concept import_sp -n beta4
concept add_scenario -n beta4 -s scenario1
concept init_scenario -n beta4 -s scenario1
concept import_globals -n beta4 -d ../concept_projects/beta4/globals -t
```

```
concept import_rpo -n beta4 -d ../concept_projects/beta4/rpo -t
concept import_control -n beta4 -s scenario1 -c
../concept_projects/beta4/scenario/run_control.txt
concept run_grid_definition_model -n beta4 -s scenario1 –d
../concept_projects/beta4/scenario1/output
concept import_metdata -n beta4 -s scenario1 -d
../concept_projects/beta4/scenario1/met_data

date
```

1.7 Global Variable/Validation Tables

The import_globals command creates a number of tables used for data validation and error checking by many or all portions of the CONCEPT model. The ASCII versions of these input files are located in concept_projects/project_name/globals. Additional tables located in this directory are model specific, and will be described in the pertinent model sections.

control_device_code.txt
conversion_factors.txt
emission_types.txt
emission_units_codes.txt
fips.txt
pollutant_codes.txt
scc.txt
sic_codes.txt
time_zones.txt
tribal_codes.txt

Table 1.7.1 control_device_code.txt

Field Name	Description	Type	Size	Begin	End
CONTROL_DEVICE_CODE	Valid control codes	CHARACTER	4	1	4

Table 1.7.2 conversion_factors.txt

Field Name	Description	Type	Size	Begin	End
EMISSIONS_UNITS_CODES	Option for units of input emission values	CHARACTER	10	1	10
EMISSION_UNITS_DESC	Description	CHARACTER	10	11	20
FACTOR_TO_KG	Factor to convert to Kilograms	NUMBER	20	21	40

Table 1.7.3 emissions_types.txt

Field Name	Description	Type	Size	Begin	End
EMISSION_TYPE	Temporal emissions type code (from NEI definitions)	CHARACTER	5	1	5
EMISSION_TYPE_NAME	Description of code	CHARACTER	30	6	35

Table 1.7.4 emission_units_codes.txt

Field Name	Description	Type	Size	Begin	End
EMISSIONS_UNITS_CODE	Valid units for emission measures	CHARACTER	10	1	10

Table 1.7.5 fips.txt

Field Name	Description	Type	Size	Begin	End
COUNTRY_CODE	2-Letter country code (US)	CHARACTER	2	1	2
STATE_COUNTY_FIPS	State and county FIPS ID (01001)	CHARACTER	5	3	7
STATE_ABBR	2 Letter State Abbrev.	CHARACTER	4	8	11
COUNTY_NAME	County Name	CHARACTER	22	12	33
TIME_ZONE	Time zone for County	CHARACTER	3	34	36
IGNORE_DAYLT_SAVINGS	Flag “X” it ignore DLS	CHARACTER	1	37	37
COUNTY_CTR_LNG	Longitude of County center	NUMBER	11	38	48
COUNTY_CTR_LAT	Latitude of County center	NUMBER	10	49	58
COUNTY_AREA	County area (square miles)	NUMBER	13	59	71
COUNTY_WEST_LNG	Western most longitude of county	NUMBER	10	72	81
COUNRT_EAST_LNG	Eastern most longitude of county	NUMBER	10	82	91
COUNTY_NORTH_LAT	Northernmost latitude of county	NUMBER	11	92	102
COUNTY_SOUTH_LAT	Southernmost latitude of county	NUMBER	10	103	112

Table 1.7.6 pollutant_codes.txt

Field Name	Description	Type	Size	Begin	End
POLLUTANT_CODES	Valid pollutant codes for emission inputs.	CHARACTER	9	1	9
OUTPUT_UNITS	Required output units	CHARACTER	20	10	29

Table 1.7.7 scc.txt

Field Name	Description	Type	Size	Begin	End
SCC	Valid SCC codes	CHARACTER	10	1	10
SCC_DESC	Description of Code	CHARACTER	400	11	410

Table 1.7.8 sic_codes.txt

Field Name	Description	Type	Size	Begin	End
SIC_CODE	Valid SIC codes	CHARACTER	4	1	4

Table 1.7.9 time_zones.txt

Field Name	Description	Type	Size	Begin	End
TIME_ZONE	Valid Timezone Abbrev. (EST)	CHARACTER	6	1	6
GMT_OFFSET	# hours offset from GMT	NUMBER	4	7	10

Table 1.7.10 tribal_codes.txt

Field Name	Description	Type	Size	Begin	End
TRIBAL_CODE	Valid tribal codes	CHARACTER	4	1	4

1.8 Cross Reference and Lookup Tables

The CONCEPT model requires lookup and cross-reference tables for temporal, spatial, and species allocation. The RPO data exchange provides file formats for the input of these data types. The input files for these data tables is located in concept_projects/*project_name*/rpo. At this time, the CONCEPT program require these files to adhere to a strict naming convention:

RPO File Name	Description	Definition Table
rpo_cp.txt	Speciation Chemical conversion profiles	1.8.3.2
rpo_cr.txt	Speciation Profile Cross-reference	1.8.3.1
rpo_lp.txt	Speciation Profile definitions	1.8.3.3
rpo_sr_tribe.txt	Spatial Tribal/County overlay	1.8.2.3
rpo_sr.txt	Spatial Surrogate definitions	1.8.2.2
rpo_ss.txt	Spatial Surrogate assignments	1.8.2.1

rpo_td.txt	Temporal Profiles – 24 hour daily	1.8.1.4
rpo_tm.txt	Temporal Profiles – Monthly	1.8.1.2
rpo_ts.txt	Temporal Profile Assignments	1.8.1.1
rpo_tw.txt	Temporal Profile – Day of Week	1.8.1.3

1.8.1 Temporal Allocation Lookup and Cross-reference tables

To process the temporal allocation information, the cross-reference and profile tables need to be read from the RPO data exchange format.

Table 1.8.1.1 Temporal Profile Cross Reference (rpo.ts.txt)
RPO Input File Format – SCC to Temporal profile code Cross-reference table
Temporal Source (TS) Record format.

Field Name	Description	Type	Size	Begin	End
RECORD_TYPE	A code that identifies the type of record (TS)	CHARACTER	2	1	2
COUNTRY_ CODE	A code that identifies the country	CHARACTER	2	3	4
STATE_COUNTY_FIPS	The code for the state/province/tribe	CHARACTER	5	5	9
TRIBAL_CODE	FIPS county code	CHARACTER	5	10	14
STATE_FACILITY_ID	Facility id	CHARACTER	15	15	29
EMISSION_UNIT_ID	Source id	CHARACTER	6	30	35
EMISSION_POINT_ID	Stack id	CHARACTER	6	36	41
EMISSION_PROCESS_ID	Unique state/ local/ tribal id reported consistently over time	CHARACTER	6	52	47
SCC	Source category code	CHARACTER	10	48	57
PROFILE_TYPE	Type of profile number (monthly, weekly, daily, hourly)	CHARACTER	10	58	67
PROFILE_NUMBER	Profile key	NUMBER	4	68	71
WEEKDAY_TYPE	Weekend or weekday (use NIF emission type codes)	CHARACTER	2	72	73
START_DATE	Start date YYYYMMDD	NUMBER	8	74	81
END _DATE	End date YYYYMMDD	NUMBER	8	82	89

Table 1.8.1.2 Monthly Temporal Profile Lookup (rpo_tm.txt)
RPO Input File Format – Definition of Monthly temporal allocation profiles
Temporal Monthly (TM) Record Format.

Field Name	Description	Type	Size	Begin	End
RECORD_TYPE	A code that identifies the type of record (TM)	CHARACTER	2	1	2
PROFILE_NUMBER	Monthly TAF profile number	NUMBER	4	3	6
JAN	January allocation factor	NUMBER	5	7	11

FEB	February allocation factor	NUMBER	5	12	16
MAR	March allocation factor	NUMBER	5	17	21
APR	April allocation factor	NUMBER	5	22	26
MAY	May allocation factor	NUMBER	5	27	31
JUN	June allocation factor	NUMBER	5	32	36
JUL	July allocation factor	NUMBER	5	37	41
AUG	August allocation factor	NUMBER	5	42	46
SEP	September allocation factor	NUMBER	5	47	51
OCT	October allocation factor	NUMBER	5	52	56
NOV	November allocation factor	NUMBER	5	57	61
DEC	December allocation factor	NUMBER	5	62	66
NORM	Monthly normalization factor	NUMBER	8	67	74

Table 1.8.1.3 Weekly Temporal Profile Lookup (rpo_tw.txt)
Input File Format – Definition of Weekly temporal Allocation profiles
Temporal Weekly (TW) Record Format

Field Name	Description	Type	Size	Begin	End
RECORD TYPE	A code that identifies the type of record (TW)	CHARACTER	2	1	2
PROFILE_NUMBER	Weekly TAF profile number	NUMBER	4	3	6
MON	Monday allocation factor	NUMBER	5	7	11
TUE	Tuesday allocation factor	NUMBER	5	12	16
WED	Wednesday allocation factor	NUMBER	5	17	21
THU	Thursday allocation factor	NUMBER	5	22	26
FRI	Friday allocation factor	NUMBER	5	27	31
SAT	Saturday allocation factor	NUMBER	5	32	36
SUN	Sunday allocation factor	NUMBER	5	37	41
NORM	Weekly normalization factor	NUMBER	8	42	49

Table 1.8.1.4 Hourly Temporal Profile Lookup (rpo_td.txt)
Input File Format – Definition of Hourly Temporal Allocation Profiles
Temporal Daily (TD) Record Format

Field Name	Description	Type	Size	Begin	End
RECORD TYPE	A code that identifies the type of record (TD)	CHARACTER	2	1	2
PROFILE_NUMBER	Diurnal TAF profile number	NUMBER	4	3	6
HR01	Hour 1 allocation factor	NUMBER	5	7	11
HR02	Hour 2 allocation factor	NUMBER	5	12	16
HR03	Hour 3 allocation factor	NUMBER	5	17	21
HR04	Hour 4 allocation factor	NUMBER	5	22	26
HR05	Hour 5 allocation factor	NUMBER	5	27	31
HR06	Hour 6 allocation factor	NUMBER	5	32	36
HR07	Hour 7 allocation factor	NUMBER	5	37	41
HR08	Hour 8 allocation factor	NUMBER	5	42	46
HR09	Hour 9 allocation factor	NUMBER	5	47	51
HR10	Hour 10 allocation factor	NUMBER	5	52	56
HR11	Hour 11 allocation factor	NUMBER	5	57	61
HR12	Hour 12 allocation factor	NUMBER	5	62	66
HR13	Hour 13 allocation factor	NUMBER	5	67	71
HR14	Hour 14 allocation factor	NUMBER	5	72	76
HR15	Hour 15 allocation factor	NUMBER	5	77	81
HR16	Hour 16 allocation factor	NUMBER	5	82	86
HR17	Hour 17 allocation factor	NUMBER	5	87	91
HR18	Hour 18 allocation factor	NUMBER	5	92	96
HR19	Hour 19 allocation factor	NUMBER	5	97	101

HR20	Hour 20 allocation factor	NUMBER	5	102	106
HR21	Hour 21 allocation factor	NUMBER	5	107	111
HR22	Hour 22 allocation factor	NUMBER	5	112	116
HR23	Hour 23 allocation factor	NUMBER	5	117	121
HR24	Hour 24 allocation factor	NUMBER	5	122	126
NORM	Daily normalization factor	NUMBER	8	127	134

1.8.2 Spatial Allocation Lookup and Cross-reference tables

To process the spatial allocation information, the cross-reference and profile tables need to be available to the area source model. The spatial allocation data can be provide in two ways 1) through the spatial allocation model, 2) read from the RPO data exchange format. If using the RPO data import routines, the rpo_ss.txt and rpo_sr.txt files must be available in the rpo directory. The rpo_sr_tribe.txt files is independent of the grid or episode selections, and does not need to be regenerated

Table 1.8.2.1 SCC to Spatial profile code Cross-reference table (rpo_ss.txt)
RPO Input File Format – Spatial Source (AS) Record format.

Field Name	Description	Type	Size	Begin	End
RECORD_TYPE	A code that identifies the type of record (AS)	CHARACTER	2	1	2
COUNTRY_CODE	A code that identifies the country	CHARACTER	2	3	4
STATE_COUNTY_FIPS	The code for the state/province/tribe	CHARACTER	5	5	9
TRIBAL_CODE	The FIPS code for the county	CHARACTER	5	10	14
SCC	Source category code	CHARACTER	10	15	24
SSC	Spatial surrogate code	CHARACTER	10	25	34
WEIGHT	Normalized weighting fraction of SSC per SCC (In cases where these do not sum to one - the emissions model will normalize the values to one)	NUMBER	20	35	541

Table 1.8.2.2 Spatial Allocation profile definitions (rpo_sr.txt)
RPO Input File Format – Spatial profile definitions

Field Name	Description	Type	Size	Begin	End
RECORD_TYPE	A code that identifies the type of record (AR)	CHARACTER	2	1	2
COUNTRY_CODE	A code that identifies the country	CHARACTER	2	3	4
STATE_COUNTY_FIPS	The code for the state/province/tribe	CHARACTER	5	5	9
TRIBAL_CODE	The FIPS code for the county	CHARACTER	5	10	14
ICELL	I cell number (as defined by grid)	CHARACTER	3	15	17
JCELL	J cell number (as defined by grid)	CHARACTER	3	18	20
SSC	Spatial surrogate code	CHARACTER	10	21	30
RATIO	Ratio of the county/tribal land area for a given SSC in the cell. This is the fractional surrogate	NUMBER	10	31	40
VALUE	Actual value of spatial surrogate units (e.g. number of people for population, acres for forest land	NUMBER	10	41	50

Table 1.8.2.3 State/county/tribal cross-reference. (rpo_sr_tribal.txt)

Field Name	Description	Type	Size	Begin	End
RECORD TYPE	A code that identifies the type of record	CHARACTER	2	1	2
SSC		CHARACTER	10	3	12
COUNTRY CODE	A code that identifies the country	CHARACTER	2	13	14
STATE_COUNTY_FIPS	FIPS code for the state and county	CHARACTER	5	15	19
TRIBAL_CODE	Tribal identifier	CHARACTER	5	20	24
RATIO	Percentage of total County feature attributed to tribal area inside the county	NUMBER	12	25	36
BASIS		NUMBER	22	37	58
PARENT_BASIS		NUMBER	8	59	66

1.8.3 Speciation Allocation

To process the speciation allocation information, the cross-reference and profile tables need to be available to all of the CONCEPT models.

Table 1.8.3.1 SCC to Speciation profile code Cross-reference table (rpo_cr,txt)
RPO Input File Format – Speciation Profile Source (CR) Record format.

Field Name	Description	Type	Size	Begin	End
RECORD_TYPE	A code that identifies the type of record (CR)	CHARACTER	2	1	2
COUNTRY_CODE	A code that identifies the country	CHARACTER	2	3	4
STATE_COUNTY_FIPS	FIPS code for the state and county	CHARACTER	5	5	9
TRIBAL_C0DE	Tribal identifier	CHARACTER	5	10	14
SIC	4-digit sic, or 2 digit sic with remaining digits blank	CHARACTER	4	15	18
SCC	EPA source category code for point /area sources or a fraction of the code	CHARACTER	10	19	28
EM_MODE	Emissions mode for MV types		20	29	48
STATE_FACILITY_ID	Unique state/ local/ tribal id reported consistently over time	CHARACTER	15	49	63
EMISSION_UNIT_ID	Unique state/ local/ tribal id reported consistently over time	CHARACTER	6	64	69
EMISSION_POINT_ID	State/ local/ tribal id for point / location where emissions are released to ambient air	CHARACTER	6	70	75
POLLUTANT_CODE	Pollutant code	CHARACTER	15	76	90
PROFILE_CODE	Speciation profile code	CHARACTER	15	91	105

Table 1.8.3.2 Chemical conversion profiles (rpo_cp.txt)
(ie VOC-to-TOG)

Field Name	Description	Type	Size	Begin	End
RECORD_TYPE	A code that identifies the type of record (CP)	CHARACTER	2	1	2
COUNTRY_CODE	A code that identifies the country	CHARACTER	2	3	4
STATE_COUNTY_FIPS	FIPS code for the state and county	CHARACTER	4	5	8
TRIBAL_CODE	The FIPS code for the county	CHARACTER	3	9	11
SIC	4-digit sic, or 2 digit sic with remaining digits blank	CHARACTER	4	12	15
SCC	EPA source category code for point /area sources or a fraction of the code	CHARACTER	10	16	25
SITE_ID	Unique state/ local/ tribal id reported consistently	CHARACTER	15	26	40
EMISSION_UNIT_ID	Unique state/ local/ tribal id reported consistently	CHARACTER	6	41	46
EMISSION_POINT_ID	State/ local/ tribal id for point / location where emissions are released to ambient air	CHARACTER	6	47	52

PROCESS_ID	Unique state/ local/ tribal id reported consistently	CHARACTER	6	53	58
CONVERSION _FACTOR	Volatile conversion factor. This needs to reflect speciation profile at the time.	NUMBER	5	59	63
POLLUTANT_CODE_FROM	Name of pollutant or emission type converting from	CHARACTER	9	64	72
POLLUTANT_CODE_TO	Name of pollutant or emission type converting to	CHARACTER	9	73	81

Table 1.8.3.3 Lumped Speciation Profile Definition (rpo_lp.txt)
RPO Input File Format – Speciation Profile (LP) Record format.

Field Name	Description	Type	Size	Begin	End
RECORD_TYPE	A code that identifies the type of record (LP)	CHARACTER	2	1	2
PROFILE_CODE	Speciation profile code	CHARACTER	15	3	17
CHEMICAL_MECHANISM	Chemical mechanism code (indicate degree of lumping if variable , as is the case for SAPRC-99)	CHARACTER	10	18	27
POLLUTANT_CODE	Pollutant id	CHARACTER	9	28	36
CHEM_MECH_ GROUP	Chemical Mechanism Group	CHARACTER	10	37	46
DIVISOR	Divisor	NUMBER	10	47	56
SPLIT_FACTOR	Model chemical compound name used to represent the compound name	NUMBER	10	57	66

1.9 Meteorology

Two types of meteorology data is required for the CONCEPT system. MM5 format files are required of use in the calculations of the Mobile, Nonroad, and Biogenics models. To produce CMAQ-ready output formats, the point source model requires MCIP files in the horizontal and vertical grid structure required for output.

1.9.1 MM5 Files

Episode specific gridded, hourly meteorological data are required to calculate temperature sensitive emissions for on-road mobile source, nonroad emissions, and biogenics. CONCEPT requires that these data are available as tables in the CONCEPT scenario structures.

The importation of the MM5 files requires two phases:

1.9.1.1 Reformat the MM5 binary files into CONCEPT ready ASCII files.

Located under the concept directory, the mm5_met_interfac directory contains the scripts and source code to convert the MM5 binary, select the required met fields, and output an ASCII file for CONCEPT processing. By decoupling this converter from the concept structure, the user will be able to easily substitute in other meteorology file formats and processors, with the only requirements being that the processor output a text file in the prescribed format.

Figure 1.9.1.1 MM5 to CONCEPT conversion script (mm5v3_concept.sc)

```
# MM5, SATELITE
cat > control << -EOF-
PAR Technique      |MM5
Start/end date     |02070600 02070623
CAMx grid size     |147, 111
CAMx Dx, Dy        |36. 36.
CAMx Origin offset | 8,  8
Time Variant Output|../../concept_projects/beta3/scenario1/met_data/concept_met.txt
Time Invariant Outp|../../concept_projects/beta3/scenario1/met_data/met_invariant.txt
# MM5 files to proc| 3
MM5 filename       |mm5_test_data/MMOUT_DOMAIN1_00.jul05_02.v02_aaa
MM5 filename       |mm5_test_data/MMOUT_DOMAIN1_01.jul05_02.v02_aaa
MM5 filename       |mm5_test_data/MMOUT_DOMAIN1_02.jul05_02.v02_aaa
-EOF-

src/mm5concept.linux < control > log
```

The MM5 conversion script needs to be edited to supply the following information:

Start/end date - MMDDYYHH
CAMx grid size - Number of X, y cells, from the grid definition
CAMx Dx, Dy - Dimension of cells, in Kilometers
CAMx Origin offset - Modeling grid offset from meteorological modeling grid
Time Variant Output - Name/location for gridded hourly file (must be met_data subdirectory of modeling scenario1
Time Invariant Output – Name/location for temporally invariant parameters (latitude, longitude, ptop)
MM5 files to process - Number of MM5 files to read to access required data
MM5 filename - Name and locations of MM5 input files.

The output of this script is the met files, in the designated location, and a log file that summarizes the processing steps.

Table 1.9.1 Output of MM5 Conversion script – concept_met.txt

Field Name	Description	Type	Size	Begin	End
RUN_DATE	YYYYMMDD	CHARACTER	8	1	8
HOUR	HH – Always in GMT	NUMERIC	3	9	11
ICELL	XXXX	NUMERIC	5	12	16
JCELL	XXXX	NUMERIC	5	17	21
MET_PAR	Photosynthetically active radiation (umol/(m^2s))	NUMERIC	8	22	29
MET_PAR_BEIS2	Photosynthetically active radiation (uE/(m^2s))	NUMERIC	7	30	36
MET_TOTAL_RADIATION	Total solar radiation on a horizontal surface (watt/m^2)	NUMERIC	7	37	43
MET_TEMPERATURE	Dry bulb temperature (MM5 level 1) (K)	NUMERIC	6	44	49
MET_TEMP10M	Dry bulb temperature (10m) (K)	NUMERIC	6	50	55
MET_TEMP2M	Dry bulb temperature (2m) (K)	NUMERIC	6	56	61
MET_WS	Wind speed at surface (m/s)	NUMERIC	5	62	66
MET_WS10M	Wind speed at 10meters (m/s)	NUMERIC	5	67	71

MET_WD	Wind direction (degrees)	NUMERIC	6	72	77
MET_RUN_CONV	Convective rainfall rate (mm/hr)	NUMERIC	6	78	83
MET_RN_NONCONV	Nonconvective rainfall rate (mm/hr)	NUMERIC	6	84	89
MET_EVAPORATION_RATE	Rate of evaporation (mm/hr)	NUMERIC	6	90	95
MET_ABS_HUMIDITY	Absolute humidity (kg/kg)	NUMERIC	7	96	102
MET_BAROMETRIC_PRESS	barometric pressure (mbar)	NUMERIC	7	103	109
MET_CLOUD_COVER	Fraction cloud cover (%)	NUMERIC	6	110	115
MET_SOIL_MOIS	Soil moisture (m^3/m^3)	NUMERIC	7	116	122
MET_SOIL_TEMP	Soil Temperature (K)	NUMERIC	6	123	128
MET_USTAR	frictional velocity (ustar) (m/s)	NUMERIC	6	129	134
MET_ZZERO	surface roughness length (m)	NUMERIC	6	135	140

Table 1.9.2 Output of MM5 Conversion script – met_invariant.txt

Field Name	Description	Type	Size	Begin	End
ICELL	XXXX	NUMERIC	4	1	4
JCELL	XXXX	NUMERIC	5	5	9
LONGITUDE	Longitude of the grid cell center (cross-point)	NUMERIC	10	10	19
LATITUDE	Latitude of the grid cell center (cross-point)	NUMERIC	9	20	28
PTOP	Pressure at top of MM5 domain (mbar)	NUMERIC	9	29	37

1.9.1.2 Read the met test files into the CONCEPT database structures.

After the MM5 files have been translated into the concept_met.txt and met_invariant.txt files, the CONCEPT script can be used to create the data tables and populate them with the met data

Example:

```
concept import_metdata -n beta4 -s scenario1 –d
../concept_projects/beta4/scenario1/met_data
```

1.9.2 MCIP Meteorological Data