United States Environmental Protection Agency
Office of Research and Development Washington, DC 20460
EPA-600/R-98/069(b) June 1998

EPA Third-Generation Air Quality Modeling System

Models-3 Volume 9b User Manual

APPENDICES

APPENDIX A

Glossary

Accumulation Mode. A convenient term that describes the size range of the mass of aerosol particles with diameters nominally between 0.1 and 1.0 micrometers. The term is based upon the concept introduced by Kenneth Whitby that particles in this size range accumulate in the atmosphere from the growth of much smaller particles (See Aitken mode). The growth may be from new mass condensing on the smaller particles or from the coagulation of the smaller particles.

Adaptive Grid. A grid structure that varies during model execution according to the value(s) of some model parameter(s). For example, in a photochemistry model, grid resolution may automatically increase in areas of high spatial gradients of NO_x. This allows more accurate determination of plume-to-background concentration ratios, which greatly influence photochemical ozone production. In a meteorological model, an adaptive grid may automatically increase the grid resolution in an area of modeling where there is a large atmospheric pressure change across a grid cell.

Air Quality Modeling System. A computational system environment that combines a set of physical and chemical models which describe atmospheric processes that are important to trace-gas and particulate matter distributions in the atmosphere. These systems typically include meteorological models, emissions models, chemistry transport models, and the analysis and visualization tools necessary for supporting decisions related to air quality.

Aitken Mode. The term describes the size range of the mass of aerosol particles with diameters nominally between 0.01 and 0.1 micrometers. Such particles are formed in the atmosphere by nucleation of gas-phase precursors or by direct emissions from sources. The most common source is combustion .

Alpha. Digital Equipment Company workstation.

Arakawa-B. Horizontal grid staggering system described by Arakawa-B and Lamb (1977). Mass variables and momentum variables are defined on separate horizontal grids of spacing equal to delta-x. The two grids are offset by 0.5 * delta-x in the north-south and east-west directions. A sample grid is shown in Figure 5-2.

ARC/INFO. A high-end Geographical Information System with capabilities for the automation, modification, management, analysis, and display of geographical information.

ArcPlot. Plot subsystem of ARCInfo.

Automatic QC. Quality control correction is accomplished automatically without user intervention.

Bookmark. In on-line help, a bookmark marks an entry of a help document so it can be quickly accessed later. A list of bookmarks appears in the Bookmarks Menu on the help browser window and also in the bookmark's window. Each item on the list is a hypertext link to a marked entry.

Check. Read a file or a set of data, compare the read values against specific criteria defined by the user and/or the software.

Chemistry-Transport Model (CTM). A model that simulates various physical and chemical processes that are important for understanding atmospheric trace-gas distributions. The processes include atmospheric transport; vertical mixing; atmospheric chemistry in the gas phase, in the aqueous phase, and in aerosols; cloud mixing and precipitation scavenging; and surface removal processes. Generally, a chemistry-transport model relies on a meteorological model for the description of atmospheric states and motions, and depends on an emissions model for the description of anthropogenic and biogenic emissions that are injected into the atmosphere.

Class. A collection of software modules associated with a science process.

Community Multi-Scale Air Quality (CMAQ). This type of model simulates various physical and chemical processes for understanding atmospheric trace-gas distributions. The processes include atmospheric transport; vertical mixing; atmospheric chemistry in the gas phase, in the aqueous phase, and in aerosols; cloud mixing and precipitation scavenging; and surface removal processes. Generally, a CMAQ model relies on a meteorological model for the description of atmospheric states and motions, and depends on an emissions model for the description of anthropogenic and biogenic emissions that are injected into the atmosphere. CMAQ models will typically include a variety of interchangeable modules.

Conforming Datasets. Conforming datasets are in I/O API format. Most programs (models visualization and analysis routines) in the Models-3 system are able to read and write conforming datasets. This capability eliminates the need for data conversion even within a distributed computing environment.

Conforming Programs. Conforming programs generally use the I/O API library routines for reading and writing data. Key data structures are defined by globally shared information. You define this critical data structure information once, and it is automatically made available for use in conforming code throughout the system. This globally shared information is permanently stored as objects in an object database. In Models-3, these objects are a stored collection of related information, such as grid dimensions and resolution, coordinate system definitions, chemical mechanism species, and reactions.

Control Equipment Efficiency. The overall collection efficiency in weight percentage for all control equipment applicable to pollutants at the projected pollutant segment level. Valid values are 0.001 through 99.999.

Cray. A powerful vector supercomputer.

Daemon. A process that runs in the background independently and performs a function. An example is a printer daemon which controls the job queue for the printer.

Dataset Manager. Use Dataset Manager to manipulate files for use with models and analysis programs within Models-3. You can access any data file on your network as long as you have appropriate permissions and provide a full pathname. Registering data will simplify and speed up your search for and access to a particular file, by enabling the use of browsers and keyword selections. Some metadata is required to initially register a dataset, but once it is registered you can easily access a dataset anywhere on the network through use of your mouse.

Decision Support System. An automated system that provides information for decision making.

Emission Model. This type of model simulates the emissions of trace gas and particulate matter into the atmosphere under ambient meteorological conditions and socioeconomic activities. Emissions from both natural and man-made sources are modeled. Emissions modeling is an important component of air quality modeling systems because it is the chief forcing function of CMAQ models. Emissions models will typically include a variety of interchangeable modules.

Environmental Modeling System. A set of computational models that mathematically represent a simplified version of real-world phenomena, along with the necessary mechanisms for managing the processing, the data produced by it, and the analysis and visualization tools necessary for making related decisions. (This could be the creation and behavior of environmental pollutants as a function of weather and chemical interactions.) Researchers use these "scaled-down" versions of the world to perform experiments that would be too dangerous, too costly, or simply impossible in the real world.

Eulerian. Fluid motion specification where one concentrates on what happens at a spatial point, x, so that independent variables are taken as x and *t*.

Exploratory Models. Test bed models for developing scientific algorithms, which are not thoroughly reviewed and evaluated. (See operational models and screening models.)

FORTRAN. Formula translator (computer programming language).

Framework. A system of mechanisms to manage the scheduling and execution of computational models, the data produced by them, the analysis and visualization tools necessary for understanding their results for decision making, and the interfaces to all these capabilities.

Framework Administrator. A function of Models-3 which centralizes the administrative tasks of each Models-3 framework subsystem and provides a uniform interface. Administrative tasks include monitoring, tuning, and maintaining. Refer to the Models-3 Installation Manual for information on using the Framework Administrator subsystem.

FSVIEW. Full screen editor used with the SAS programming language (one of the optional pieces of SAS, but required by Models-3).

Generalized Coordinate System. A scheme for constructing coordinate systems that allows for choices of horizontal map projection type (e.g., Lat./Lon., Lambert, Mercator, Stereographic Universal Transverse Mercator, or Matrix) and projection parameters, as well as for choice of various vertical coordinate type (e.g., Pressure Coordinate, Height above Ground, Height above Sea Level, Sigma-P Hydrostatic, Sigma-P Non-Hydrostatic, and Sigma-Z). The advantage of a generalized coordinate system is that a CMAQ model can adapt to a variety of different possibilities in horizontal and vertical parameters.

Generic Grid System. A scheme for constructing grid systems that allows for choices of origin, orientation, and spatial resolution. It allows a model to be expressed in a grid system that is optimal for representing the governing equations. For a regular, rectangular grid system, mapping gridded data to the earth's surface can be achieved by defining the number of rows and columns, cell size, and location and extent. For an irregular grid system, grid intersections (nodes) are described by coordinates from a reference position.

Geocoded. An entity with an identifier associated with geographic boundaries (i.e, state or county code).

Geographic Information System. A computer-based system for managing, analyzing, manipulating, and displaying geographic information in both numeric and graphical ways.

Geospatial. Refers to the spatial extent of a geographic boundary.

GIS. Geographic Information System. A computer-based system for the management, analysis, manipulation, and display of geographic information in both numeric and graphical ways.

Grid. A rectangle defining the extent of an area (domain) considered for a study. A grid is subdivided into smaller rectangles, usually squares, called grid cells.

Grid Cell. The smallest subdivision of a grid.

Grid Resolution. Length of shortest side of a rectangular grid cell.

Gridspec. A MEPPS directory level. It contains the files that have been temporalized (i.e., the emissions have been processed from data extracted over an expanse of time) and gridded (or are to be gridded). If a Gridspec directory is not predefined, then use the MEPPS Main Menu/Session to create one.

Growth Factor. An adjustment factor used to estimate the growth in a source's activity level between the inventory base year and a projected year. Valid values are 0.00 - 99.99

Help Browser Window. The help server displays the help browser window when providing on-line help for an application. The browser displays entries of a help document; you can step to adjacent entries in the document, or jump to other entries in the help document set by using hypertext links.

Heterogeneous, Distributed Computing Environment. A heterogeneous computing environment consists of multiple machines of different types (for example, supercomputers, graphics workstations, and workstation file servers). A distributed computing environment permits the execution of a large computational task to be shared by multiple machines linked together by a network. Thus, a heterogeneous, distributed computing environment consists of many different kinds of machines networked together.

Hydrostatic. Used to indicate that model assumes that the atmosphere is in hydrostatic equilibrium, i.e. surfaces of constant pressure and constant mass coincide and are horizontal throughout. Complete balance exists between the force of gravity and the pressure force.

Hypertext Link. A hypertext link is a specially designated word or phrase displayed in text that has been saved in hypertext (html) format. A hypertext link provides nonsequential access to other entries in a document set. In Models-3 the "Help" facility is done using hypertext. Hypertext linking is done to all help entries.

IDA. The Inventory Data Analyzer used for input and quality control checks of emission inventories.

Input/Output Application Programmer Interface (I/O API). A software library that reads and writes files. It uses an internal data format that is machine independent and that conforms to the widely used University Corporation for Atmospheric Research Network Common Data Format (netCDF). The I/O API files contain self-describing headers with complete information that is necessary to use and interpret the data contained in the file. The Models-3 I/O API format

is slightly more restrictive than the standard netCDF format, on how the header information must be written. The I/O API library provides a variety of data structure types and a set of reusable access routines that offer selective direct access to the data in terms that are meaningful to the environmental modeler. Supported data types include gridded, boundary, vertical profile, grid nest, time series, and event-driven.

Internal. With respect to Models-3 data, the data are available within the software. The user does not have to provide them (look-up tables, ranges, list of state/county names).

Inventory. With respect to the Models-3 emission system, inventory refers to a file or a database containing emission data for a specific set of pollutants for a specific time period (typically for the entirety of a specific year) for an area (country, states, counties).

Irix. Operating system for SGI computers.

Keyword. A keyword is a word or phrase, up to 40 characters long, that can be used to locate an entity in help text or to locate a Models- object (e.g., dataset, program, study, or model) using a "find" screen.

Linux. A UNIX-like operating system for IBM PCs and compatibles that use Intel 386, 486, or Pentium chips. It is available and re-distributable under the terms of the GNU Public License.

Massively Parallel Processing. Computer processing employing multiple CPUs to execute multiple computational tasks simultaneously. Massively parallel systems employ a large number of CPUs simultaneously on the same task. In contrast, conventional computer design uses a single CPU to perform computational tasks in a strictly linear, sequential order.

Mesoscale. In meteorology, phenomena or processes that have a temporal and horizontal spatial scale smaller than the conventional rawinsonde network, but significantly smaller than individual cumulus clouds. This implies that the horizontal to several hundred kilometers, with a time scale of approximately 1 to 12 h. The vertical scale extends from tens of meters to the depth of the troposphere. From Pielke (1984).

Metadata. Information about data and about the processes that produced them. In particular, information about the data's structure, internal characteristics and history, and location within a data storage environment, as well as information derived from the data.

Meteorological Model. This type of model provides descriptions of atmospheric motions, momentum, moisture, heat fluxes, turbulence characteristics, clouds and precipitation, and atmospheric radiative characteristics. Most meteorological models currently in use for air quality modeling were originally developed for the prediction of weather. CMAQ models require

information from a meteorological model that is designed to address specific issues relevant to air quality modeling (such as planetary boundary layer information, cloud distribution and mixing characteristics, precipitation, and surface fluxes) rather than the specific problems of weather forecasting.

Mie scattering. A generalized particulate light-scattering mechanism that follows from the laws of electro-magnetism applied to particulate matter.

Mixed-media. Simultaneously involving more than one environmental pollutant medium, such as air and water.

Mobile 5a. A computer application that calculates emission factors for on-road mobile vehicles.

Model. A representation of a planned or existing object or condition.

Model Builder (person). Applied scientists and software developers who construct and study theories to explain physical and chemical processes, and use computer models to study their theories.

Model Builder (Models-3 component). Model Builder allows a user to select grid location and resolution, and chemistry mechanism to build an executable model suited to user needs. It also allows a model developer to select alternative formulations of individual science components to build customized models for exploratory studies and assists with the development of configuration files for creating new model executables from existing, modified, or new science processes.

Model User. Research, production, or quality assurance person (i.e., applied scientists and engineers) who generates valid input for the modeling systems, runs the modeling systems, maintains audit trails, analyzes input and output for correctness, and produces analyses of the modeling results.

Modeling Structure. A design specification that provides the paradigm of operation and the interface specifications for the modules used to construct a particular family of models. In a CMAQ model, for example, the paradigm is that modules act as operators upon a shared concentration field, and four types of interfaces (call interfaces, INCLUDE-file interfaces, I/O interfaces, and UNIX-environment interfaces) must be specified.

Modeling System. A set of computational models and associated data processors that together provide for the simulation of processes of interest.

Models-3. The third generation air quality modeling system. It is a flexible system that addresses multiple air quality issues, such as regional and urban scale oxidant and acid deposition.

Models-3 Components. The different subsystem within the Models-3 framework. Each component is represented by its own icon. The available components are: Dataset Manager, Model Builder, Program Manager, Science Manager, Source Code Manager, Strategy Manager, Study Planner, Tools Manager, and Framework Administrator.

Module. A subset that is part of a larger program (such as meteorological, emissions, or CMAQ). In a modular program, all modules of a particular type (e.g., those that compute dry deposition) are interchangeable, allowing you to replace one module with another to determine, for example, how each module affects modeling results. Examples of modules include science modules and analysis and visualization modules.

Monotonic. A quality of strictly increasing or decreasing within some interval.

Multi-level Nesting. Multi-level nesting refers to employing *nested grids* within nested grids, possibly several levels deep.

National Inventories. Refers to standard inventories such as the 1990 Interim and the 1995-NET.

Nested Grids. Nesting refers to fitting a finer-resolution grid over part of a coarser-resolution grid. The finer-resolution grid receives information (such as boundary conditions) from the coarser-grid simulation.

Non-conforming Datasets. Non-conforming datasets are ones that are not in I/O API format. They can be used in the Models-3 framework by programs that are specifically designed to read those datasets. When non-conforming datasets and programs are used, however, you must know how to match programs and datasets, and which data formats and programs are transportable to different machine architectures. Those considerations are automatically managed by the Models-3 framework for those using conforming datasets and conforming programs.

Non-hydrostatic. Used to indicate that the model does not assume that the atmosphere is in hydrostatic equilibrium. Air is not assumed to have only horizontal motion relative to the earth.

NQS. The Network Queuing System provides a means to submit batch jobs to local and remote UNIX machines.

Operational Models. These models offer fully functional modeling of relevant science processes such as atmospheric, emissions, and chemical transport processes. They represent the current state-of-the-art that has undergone significant quality assurance review, peer review, and evaluation. The turnaround time for these models is generally much longer than for screening models but short enough to allow episodic studies.

Orbix. Registered software product of IONA Technologies, Ltd. used as the object request broker within the Models-3 framework system.

Other Inventories. Refers to any non-National Inventory.

Parameterize. To create an algorithm that describes the average large-scale behavior of a physical phenomenon, rather than describing the subgrid-scale behavior in terms of the underlying physics and chemistry. For example, a parameterized cloud algorithm might describe average cloud behavior over 80-km-square cells, although the individual clouds are much smaller that 80 km across.

Planetary Boundary Layer. The portion of the atmosphere adjacent to the earth's surface. This layer generally ranges from 100 m to 2 km in height and is significantly influenced by surface effects (e.g., heating, friction). These effects cause the layer to be well-mixed, which affects the distribution of pollutants in the air.

Pop-up Window. A pop-up window is a special window for displaying an on-line help entry. The window opens when you select a specially designated hypertext link. Pop-up windows are useful for quickly displaying short, concise units of information. You cannot jump or step to other entries from a pop-up window.

Prepare. Read and process a file or a set of data.

Process. Read a file or a set of data, perform the desired functionality (QC, reformatting, algebraic operations, etc.) and submit the processed data to the next set of actions.

Process Analysis. Traces the source(s) of a chemical species within a simulation. One example of process analysis is determining whether a species concentration originated within the cell in which it is found or within some other cell(s). Another example is determining what chemical constituents were involved in creating a species produced by reaction (rather than transported from somewhere else).

Program Manager. Use Program Manager to manipulate executable programs. During registration, enter characteristics of programs into the framework. These programs are used in the Study Planner to define studies.

QC. The act of reading data (inventories, files), checking them for correctness, and/or completeness, and/or consistency. QC may involve automatic correction, substitution, or the filling of the missing data. All QC processes are followed by QC reports.

Query Mode. Query mode designates how context-sensitive help is displayed for an application. Help for an application in query mode is displayed in the help browser when you make a request. The browser is not updated until you make another request. In query mode, request help by pressing the **<F1>** key.

Register Data. When you register data, you are making something that already exists (e.g., a file) known to the system.

Rule Effectiveness Percent. An adjustment to projected estimated emissions data to account for emissions underestimates due to compliance failures and the inability of most inventory techniques to include these failures in an emission estimate. The adjustment accounts for known underestimates due to noncompliance with existing rules, control equipment downtime, or operational problems and process upsets. Valid values: 0 to 100.

Rule Penetration Percent. An adjustment to projected estimated emissions data to account for emissions underestimates due to failure to implement rules throughout the area of intent. Valid values: 0 to 100.

Scalable. In the context of parallel computer architectures and algorithms, a parallel architecture or algorithm is termed scalable if its performance behaves linearly in terms of the number of processors employed. For example, doubling the number of processors does not cause new communications bottlenecks to appear, but doubles the performance achieved.

Scale Flexibility. The modeling system's ability to accurately simulate physical and chemical processes that describe atmospheric pollutants at different spatial and temporal scales. A modeling system with scalable algorithms for physical and chemical processes and with a generic grid system has this quality.

Science Manager. Use the Science Manager to register the basic scientific components (i.e., spatial, temporal, and chemical) that are involved in the simulation. Thereafter, other parts of the system will use what you have registered here. This saves you from having to enter spatial, temporal, or chemical specifications more than once.

Science Module. A component that is part of a modeling program (such as a meteorological, emissions, or CMAQ) and that computes data for a discrete category of environmental phenomena (e.g., dry deposition, photochemistry, vertical advection).

Screening Models. These models have simplified science processes designed for quick assessment studies. Use these models when you are willing to sacrifice some accuracy for faster turnaround time. A typical screening study involves making a large number of runs in order to identify episodes or situations that warrant more detailed modeling.

Source. A MEPPS directory level subordinate to the Study directory. It contains ASCII-format files such as INPRO-processed emission files. If the Source directory is not predefined, then you can define it via the MEPPS Main Menu/Session.

Source. With respect to air pollution, a point, area, or mobile source that produces and/or emits air pollutants.

Source Classification Code (SCC). The SCC system is used for classifying emissions sources at the level of individual processes (e.g., automobiles, boilers, solvent sources) within an emissions data inventory.

Source Code Manager. Use Source Code Manager to store or retrieve source code for conforming scientific models. It allows you to retrieve a version of a source code file, change it, and return it to the code archive after the change has been tested. Once this file has been returned to the archive, all other users who check out the file will see your version of the file. It tracks historical information on the source code. Source code is changed by scientists, not by the regulators. It is distributed to the regulators because they must compile a model with their own grid. The compilation program uses Source Code Manager.

Speciation. In Models-3, speciation refers to using one of the chemical mechanisms available with Models-3 to disaggregate a chemical substance (pollutant) into simpler compounds.

Species. Typically, a chemical substance or group of related substances whose behavior is modeled during environmental simulation modeling.

Strategy Manager. Use the Strategy Manager for air quality problems to project point-, area-, and mobile-source emissions contributions, and to estimate the relative effectiveness of control scenarios. You may adjust pollutant growth factors and emissions control factors to perform "what if" analyses. These factors are applied to EPA 1990 (base year) inventories to project future-year emissions. Other run-defining parameters you may select are: Geographic coverage (EPA-region, state, county, or user-defined area); Pollutant type (CO, NO_x, PM10, SO_2, or VOC); Emission source type (Point, Area, and/or Mobile); Source Classification Codes (SCCs); Projection year(s) (1991 through 2010).

Study. A MEPPS directory level. If the Study directory is not predefined, then you can define it via the MEPPS Main Menu/Session.

Study Planner. User Study Planner to manipulate (save, edit, delete, etc.) a study and to execute its processes. The Study Planner allows you to graphically enter each program and dataset, and to enter how they connect to make a plan. Multiple plans are allowed in each study to allow for different approaches and/or different segments. Once a plan is entered with its graphical diagram, the plan can be executed.

Sub-grid-scale Phenomena. Modeling phenomena that occur on a scale smaller than the grid resolution of the modeling system, such as point-source plumes and convective clouds.

Summary report. Generally refers to an automatic short report generated after the execution of a process.

Surface Fluxes. The exchange of material, energy, or momentum between the surface of the earth and the atmosphere.

Table of Contents Window. With respect to the Help facility in Models-3, the table of contents is a window available from the help browser window that lists the help documents in a help document set. The table of contents window also lists the entries for each help document. The entries listed are hypertext links. Selecting an entry from the list displays it in the help browser window.

Televisualization. Visualization across distances. For example, a scientist in Boston may use network technologies to display images on the graphics workstation of a scientist in Atlanta, thus permitting the two scientists to conduct a real-time discussion (e.g., via telephone or workstation) of the scientific processes being illustrated without requiring physical proximity.

Time step. A time step is a fixed unit of time. A model may have one or more internal time steps for various processors. In the Models-3 framework, "time step" is used to indicate the length of time between output of variables from the model. Another term might be "output time interval."

Tools Manager. Use the Tools Manager to access third-party applications (tools) that are registered with the Models-3 framework. The third-party applications that are available are ARC/INFO, Vis5D, MEPPS, PAVE, SAS, and VisDriver. The Tools Manager allows you to add tools to the system that will help you facilitate your work.

Unicos. The name of the Cray supercomputer operating system.

Vised. A system for interactive visualization of large 5-D gridded meteorological or air quality data developed by the University of Wisconsin Space Science Center. It is widely used by

scientists to visualize the output from numerical simulations of the earth's atmosphere and oceans.

VisDriver. A graphical user interface for finding and selecting Models-3 data files on local or remote hosts and launching various visualization applications on the data.

Visualization. An important aspect of scientific computing that provides a method for presenting data quickly and compactly that is easy to understand.

APPENDIX B

Acronyms

2-D	Two-dimensional
3-D	Three-dimensional
5-D	Five-dimensional
ADE	Atmospheric Diffusion Equation
ADT	Atlantic Daylight Savings Time
AE	Aerosol
AIRS	Aerometric Information Retrieval System
AML	ARC/INFO Macro Language
AREAL	Atmospheric Research and Exposure Assessment Laboratory
ASCII	American Standard Code for Information Interchange
AVHRR	Advanced Very High Resolution Radiometer from satellite platforms
AVS	Application Visualization System
BCON	The boundary conditions processor
BEIS2	Biogenic Emissions Inventory System
CAA	Clean Air Act
CAAA	Clean Air Act Amendments
CASE	Computer-Aided Software Engineering
CB-IV	Carbon Bond-IV
CCM2	Community Climate Model Version 2
CCTM	CMAQ Chemistry-Transport Model processor
CE	Control Efficiency
CEM	Continuous Emission Monitoring
CENCODE	Census Track Code
CHIEF	Clearinghouse for Inventories and Emission Factors
CLIM	Climatological tables

CMAQ	Community Multi-Scale Air Quality
CO	Carbon Monoxide
CORBA	Common Object Request Broker Architecture
CPU	Central Processing Unit of a computer
CTM	Chemistry-Transport Model
CVS	Concurrent Versions System
CYID	FIPS County Code
DAG	Directed Acyclic Graph
DATAGRID	Background fields pre-processor for MM5
DX	Data Explorer
E-GAS	Economic Growth and Analysis System
ECIP	Emissions-Chemistry Interface Processor
ECMTOGA	European Centre for Medium-range Tropical Ocean and Global Atmospheric Program
ECMWF	European Centre for Medium-Range Weather Forecasts
EDS	Eastern Daylight Savings Time
EIP	Emission Inventory Improvement Program
EMEFS	Eulerian Model Evaluation Field Study
EMPRO	MEPPS Emissions Processor
EMS	Emissions Modeling System
EMF_NO	Emission Factors for NO
EMF_OVOC	Emission Factors for Other VOCs
EMP_ISOP	Isoprene Emission Factor
EMP_TERP	Monoterpene Emission Factor
EPA	Environmental Protection Agency

EPM	Emissions Projection Module
ERCAM	Emission Reduction and Cost Analysis Model
ESDLS	EPA Spatial Data Library Systems
ESRI	Environmental Systems Research Institute
EST	Eastern Standard Time
ETSD	Enterprise Technology Service Division
FDDA	Four-Dimensional Data Assimilation
FHA	Federal Highway Administration
FIPS	Federal Information Processing Standard
FIRE	Factor Information Retrieval System
FPE	Future Projected Emissions
FREDS	Flexible Regional Emission Data System
FTP	File Transport Protocol
GB	Gigabyte
GEMAP	Geocoded Emissions Modeling and Projection
GEMPAK	The General Meteorology Package
GF	Growth Factor
GIP	GEMAP Input Processor
GIS	Geographic Information System
GISDB	Geographical Information System Data Base
GNU	General Public License
GOP	GEMAP Output Processor
GrADS	The Grid Analysis and Display System
GUI	Graphical User Interface
HPCC	High Performance Computing and Communication

HTML	HyperText Markup Language
I/O API	Input/Output Application Programming Interface
IC/BC	Initial Conditions/Boundary Conditions
ICON	Initial Conditions Processor
IDA	Inventory Data Analyzer
IEEE	Institute of Electrical and Electronics Engineers, Inc.
INEGI	Instituto Nacional De Estadistica Geografia E Informatica
INPRO	MEPPS Input Processor
INTERP	Interpolation pre-processor for MM5
IOV	Initial Operating Version
IRR/MB	Integrated Reaction Rates/Mass Balance Analysis
JPROC	Photolysis Rate Processor
LAN	Local Area Network
LRPM	Lagrangian Reactive Plume Module
LUCODE	Land-use/Landcover Code
LUPROC	Land-use Module in MCIP
MCIP	Meteorology-Chemistry Interface Processor
MDS	Mountain Daylight Savings Time
MECH	Chemical Mechanism Reader
MEDA	Model Evaluation Data Archive
MEIIIP	Models-3 Emission Inventory Independent Input Processor
MEPPS	Models-3 Emission Processing and Projection System
MEPRO	Models-3 Emissions Projection processor
MEPSE	Major Elevated Point Source Emission
MM4	Fourth-generation Mesoscale Meteorological Model

MM5	Fifth-generation Mesoscale Meteorological Model
MM5v2	Fifth-generation Penn State/NCAR Mesoscale Meteorological Model Version 2
MOLWT	Molecular Weight
MoVEM	Motor Vehicle Emission Model
MP	Mechanism Processor
MPP	Massively Parallel Processing (or Processor)
MPS	Multiple Projection System
MS	Modeling System
MSS	Mass Storage System
MST	Mountain Standard Time
NAAQS	National Ambient Air Quality Standards
NAPAP	National Acid Precipitation Assessment Program
NASA	National Aeronautics and Space Administration
NCAR	National Center for Atmospheric Research
NCEP	National Centers for Environmental Prediction
NDDN	National Dry Deposition Networks
NDPD	National Data Processing Division (now Enterprise Technology Service Division)
NET	National Emission Trends
NetCDF	Network Common Data Format
NHPN	National Highway Planning Network
NMC	National Meteorological Center
NO_x	Oxides of Nitrogen
NOAA	National Oceanic and Atmospheric Administration
NQS	Network Queuing system

NR	Non-reactive
NWS	National Weather Service
OAQPS	Office of Air Quality Planning and Standards
OO	Object-Oriented
OTAG	Ozone Transport Advisory Group
OUTPRO	Output Processor
PACP	Process Analysis Control Process
PAVE	Package for Analysis and Visualization of Environmental Data
PBL	Planetary Boundary Layer
PC	Personal Computer
PDM	Plume Dynamics Model
PDS	Pacific Daylight Savings Time
PinG	Plume-in-Grid Modeling
PM2.5	Particulate matter up to 2.5 microns
PM10	Particulate matter up to 10 microns
POLID	Pollutant Identifier
PST	Pacific Standard Time
PSU	The Pennsylvania State University
QA	Quality Assurance
QC	Quality Control
QSSA	Quasi-Steady State Approximation
RADM	Regional Acid Deposition Model
RAMS	Regional Atmospheric Modeling System
RAOB	Rawinsonde Upper Air Observations
RAWINS	Objective Analysis Pre-processor for MM5

RDBMS	Relational Database Management System
RE	Rule Effectiveness
RFP	Reasonable Further Progress
ROG	Reactive Organic Gases
ROM	Regional Oxidant Model
ROMNET	Regional Oxidant Model for Northeast Transport
ROP	Rate of Progress
RP	Rule Penetration
RTS	Run Time System
SAPRC	State Air Pollution Research Center
SAROAD	Storage and Retrieval of Aerometric Data
SAS	Statistical Analysis System
SCCS	Source Code Control System
SCION	Southeastern Consortium: Intermediate Oxidant Network
SDC	Systems Development Center
SDF	Software Development File
SENIOR	Southeastern Network for Intensive Oxidant Research
SERON	Southeast Regional Oxident Network
SGA	Selected Geographical Area
SGI	Silicon Graphics Incorporated
SIC	Standard Industrial Classification
SIP	State Implementation Plan
SMVGEAR	Sparse Matrix Vectorized Gear
SO_2	Sulfur Dioxide
SOC	Semivolatile Organic Compounds

SON	Spatial Ozone Network
SORPONA	Southern Oxidant Research Program for Ozone Nonattainment
SOS	Southern Oxidants Study
SODA	Southern Oxidant Study Data Archive
SPC	Species
SRS	System Requirement Specification
SST	Sea Surface Temperature
STID	FIPS State Code
StP/OMT	Software Through Pictures/Object Modeling Techniques
TAF	Temporal Allocation Factors
TERRAIN	Terrain and land use pre-processor for MM5
TIGER	Topologically Integrated Geographic Encoding Referencing
TOG	Total Organic Gases
TOMS	Total Ozone Mapping Spectrometer
TR	Tracer
TTN	Technology Transfer Network
UAM	Urban Airshed Model
UCAR	University Corporation for Atmospheric Research
URL	Uniform Resource Locator
USEPA	United States Environmental Protection Agency
USGS	United States Geological Survey
UTC	Universal Time Coordinate
UTM	Univeral Transverse Mercator
VMT	Vehicle Miles Traveled
VOC	Volatile Organic Compounds

APPENDIX C

User Interface Field Descriptions

Address Type: Describes the information provided by the address. Click on the pull-down menu button to view the address types. Values are:

- Mailing.
- Physical.
- Both.

Adjust Invent to TOG: Adjust Inventory to Total Organic Gases.

Angle from True North to Y-Axis: A parameter of the Stereo graphic Map Projection.

Angle between Cylinder Axis and North Polar Axis: A parameter of the Mercator Map projection in degrees.

Animation Rate: Rate of displaying a sequence of similar screens in milliseconds.

Area: Vis5D parameter for visualization area.

- Argument Name.
- Argument Value.

Author: The person who created the study.

Auto Contour: Selection button for PAVE to do automatic contouring.

Auto Register: Selection button to invoke auto-registration of dataset files into the Models-3 database.

Calculate: Click on these buttons to have the system automatically generate dates.

Calculate Date of Year: If you have entered the calendar date, click the *Day of Year* button for the system to calculate the Julian day.

Calculate End: Once you have entered the start dates, click the *End* button for the system to calculate the end date for the simulation. Used in conjunction with the number of time steps and the duration of the time steps.

Cal Origin: Button on the Grid Definition page of the Horizontal Grid Detail Window. Used to invoke calculation of a grid's X and Y origin based on any parent grid and row and column offsets.

Calculate Start: Once you have entered the end dates, click the *Start* button for the system to calculate the start date for the simulation.

Case: Scenario (Metadata) specifications for a simulation run.

Case ID: The unique identifier for cases generated by the Models-3 system. This ID is guaranteed to be unique across all Models-3 systems.

Case Name: A 16-character name for a case.

Case Status: The status of a case:

- Unprotected.
- Protected.

Cell Size: The grid cell dimensions:

- Delta X. Grid cell size in X-direction (units depend on the horizontal coordinate system).
- Delta Y. Grid cell size in Y-direction (units depend on the horizontal coordinate system).

Central Meridian: A parameter for the longitude of the central meridian in the Lambert Map Projection.

Chem Mechanism or Chem Mech: A named entity that contains chemical reactions and reaction rate constants for the chemistry transport model. It includes four species tables: gas phase, aerosol, non-reactive, and tracer.

City: The city name of an organization or individual.

Column Offset: Offset (in cells) of the Southwest (SW) corner of a grid in the X direction (+ to the right, - to the left) relative to the SW corner of a parent grid. If no parent grid exists, then the offset is zero.

Columns: The number of grid-cell columns in a grid. Specified on the Grid Definition page of the Science Manager Grid Detail Window.

Command Line: The command along with flags and options that is used to initiate a process.

Commands Archive: The command used to archive the dataset. Based on your selection of a hardware device, an archive command line is generated and displayed in this field. It can be modified.

Commands Restore: The command used to restore the archived dataset. Based on your selection of a hardware device, a restore command line is generated and displayed in this field.

Comments: The field for additional comments that provide greater detail than the short description field, relating to the latitude, longitude, and vertical component.

Compiler: The compiler, which includes the version used for creating the executable. Click on the pop-up selection button to view options.

Compiler Command: The compiler invocation command line including flags and options:

- f77 ... for the FORTRAN 77 compiler.
- cc ... for the C/C++ compiler.

Compiler Flags: Options given to a run of the compiler when creating an executable program.

Compute Program: The program or command that was executed against the dataset. Enter a program, or click on the pop-up selection button to navigate to the Select Program pick list to select one.

Config File: A file of model configuration information used by CVS to assembled the pieces needed to build an executable model.

Config ID: A unique identifier assigned to each configuration file generated by Models-3. This ID is guaranteed to be unique across all Models-3 systems.

Config Name: A 16-character name for a configuration file accepted by the bldmod processor.

Config Status:

- Protected.
- Unprotected.

Contact Name: The name of the person responsible for the dataset or an event.

Contact Organization: The name of the organization responsible for the dataset or an event.

Contact Position: The name of the position responsible for the dataset or an event.

Contact Type: The contact type. Click on the pull-down menu button to view contact types, either meta data or dataset.

Contact User ID: The User ID of the person responsible for the dataset or an event. Click on the pop-up selection button to view the Contact List.

Contour Range: PAVE parameters. See PAVE documentation.

- min.
- max.

Coordinate System ID: The unique identifier for coordinate systems generated by Models-3. This ID is guaranteed to be unique across all Models-3 systems (See also Horizontal Coordinate System).

Coordinate System Name: A 16-character name for a (map) coordinate system (See also Horizontal Coordinate System Name).

Coordinate System Type: See Horizontal Coordinate System Type.

Country: The country code for the organization or individual.
In MEPPS: US (00), Canada (01) and Mexico (02).

CPP Flags: The compile time flags passed to the C language preprocessor.

CPU Usage: The estimated CPU usage for running a 1-day simulation (hours/days).

Creation Date: The date the (science) component was created (See Version Creation Date).

Created By: The identifier for the creator of a (science) component.

Cross Section: PAVE pop-up menu select button for type cross section to visualize.

- X.
- Y.
- Z.

Data File: A file containing information.

Data File Host: Compiler host that houses a given data file.

Dataset Defaults Register: A toggle button that indicates whether the datasets created by the plan should be saved. Click the box to select. Click again to unselect.

Dataset Defaults Save Duration: The length of time, in days, that the datasets should be saved.

Dataset ID: See ID.

Dataset Name: The short name given to a dataset. This does not have to be a unique name.

Dataset Status: Indicates the status of the dataset. Click on the pull-down menu button to view status options. Values include:

- Archived.
- Online.
- Removed.

Dataset Type: A predefined list of dataset categories. New dataset types may be added by the System Administrator. Enter a dataset type, or click on the pop-up selection button to navigate to the Select Dataset Type pick list to select one. Predefined dataset types include:

- Air quality.
- Air quality AIRS.
- Air quality PAMS.
- Air quality simulated.
- Boundary condition.
- Chemistry.
- Demographic.
- Demographic census.
- Demographic tiger.
- Emissions.
- Emissions FIRE.
- Emissions area source.
- Emissions inventory.
- Emissions mobile.
- Emissions natural.
- Emissions point source.
- Emissions processed.
- Initial condition.
- Land use.
- Land use BEIS.
- Meteorology.

- Meteorology NWS.
- Meteorology processed.
- Meteorology satellite.
- Meteorology surface.
- Meteorology upper air.
- Satellite AVHRR.
- Soil.
- Solar radiation.
- Terrain height.
- Terrain height global.
- Terrain height USGS.
- Weather.

Date: The date when an action was taken against a dataset. Default is the current date if nothing is entered.

Date of Collection: The date when the latitude and longitude were researched or collected. The format is yyyy/mm/dd.

Day of Year: The date in the Julian calendar day, which can be a starting or ending date. Enter a Julian date, or if you have entered a calendar date in the date field, click the Calculate *Day of Year* button for the system to calculate the Julian day.

Debug: A toggle button that indicates if the execution is in debug mode. Click the box to select. Click again to unselect.

Definition: The definition of the entity or attribute.

Definition Source: The source of the entity or attribute definition.

Delete Existing Files: Toggle switch on Study Planner's Annotate Node Properties window to delete or not delete existing files of the same name when a study plan is executed.

Delimited Method: A selection for the File Converter tool (under Tools Manager) relating to if records of an ASCII file are fixed length or use a delimiter at the end of each record.

Description: A brief, 80 character long, textual description sufficient to identify an item from among a list of similar items.

Description Category: Describes the category of the feature referenced by the latitude and longitude. Click on the selection button to view description category values:

- PG Plant Entrance (General).
- PP Plant Entrance (Personnel).
- PF Plant Entrance (Freight).
- AS Air Release Stack.
- AV Air Release Vent.
- ST Storage Tank.
- WR Water Release Pipe.
- SP Lagoon or Settling Pond.
- LW Liquid Waste Treatment Unit.
- AE Atmosphere Emissions Treatment Unit.
- SD Solid Waste Treatment/Disposal Unit.
- SS Solid Waste Storage Area.
- LF Loading Facility.
- LC Loading Area Centroid.
- PU Process Unit.
- PC Process Unit Area Centroid.
- AB Administrative Building.
- FC Facility Centroid.
- NE NE Corner of Land Parcel.
- SE SE Corner of Land Parcel.
- NW NW Corner of Land Parcel.
- SW SW Corner of Land Parcel.
- CE Center of Facility.
- WL Well.
- WA Well Protection Area.
- WM Water Monitoring Station.
- AM Air Monitoring Station.
- OT Other (name in comments).
- UN Unknown.

Device Name: The name by which a device is accessed on the computer. A device name might be the name of a printer (e.g., lc890).

Device Type: The type of hardware device where the dataset is stored. Enter a device type, or click on the pop-up selection button to navigate to the Select Device Type pick list to select one.

Domain Codeset: By selecting Codeset, you access the Codeset Domain. Click on the *Insert* button to access the Codeset Domain window. Enter information in the following fields:

- **Name**: Title of the codeset (required).
- **Source**: The authority for the codeset.

Domain Enumerated: By selecting Enumerated, you access the Enumerated Domain. Click on the *Insert* button to access the Enumerated Domain window. Enter information in the following fields:

- **Value**: Name of label of a member of the enumeration set.
- **Definition**: The definition of the enumeration set.
- **Definition Source**: The source of the enumeration set.

Domain Range: By selecting Range, you access the Range Domain. Enter information in the following fields:

- **Minimum**: The smallest value that an attribute may be assigned.
- **Maximum**: The largest value that an attribute may be assigned.

E-mail: The electronic mail address of the contact individual or position.

Edit: Selection button for editing a file.

Effective Creation Date: The creation date for this horizontal grid, which identifies the earliest date this version could be used in Models-3. The format is yyyy/mm/dd.

Effective Expiration Date: The date on which a horizontal grid version was superseded by a later version. The format is yyyy/mm/dd.

Emission Layers: Vertical layers for which emissions values are calculated. The bottom of the bottom layer can be relative to sea level or ground level. The top of the top layer is taken to be the top of the atmosphere.

End: Selection from the Study Planner Plan Editing tool on the Plan Layout page. Used if the output from the plan is to be passed to another file server.

End Date: The end date in calendar (yyyy/mm/dd) format. Enter a date, or if you have entered a date in the *Day of the Year* field, click the Calculate *Calendar* button for the system to calculate the calendar end date. Also, if you know the start date information and have entered it in the appropriate fields, you may click on the Calculate *End* button for the system to calculate the end date.

End Time: The end time of the readings in the dataset. The format is in hh:mm:ss. Click on the pull-down menu button next to the time field to select A.M., P.M., or Military time.

Environment Host: The computer on which the dataset is located. Enter a host, or click on the pop-up selection button to navigate to the Select Host pick list to select one.

Environment Medium Label: This identifies the offline medium of an archived dataset.

Environment Variables: The environment variables applicable to the active node.

Executable Full Path: The directory path and name of the program to be executed.

Executable Host: The computer host (name of the machine) that stores the physical executable program file.

Execute: Selection button on the Study Planner's Plan Layout page used to initiate execution of the plan.

Execution Host: The name used in conjunction with the path name to determine where to execute the plan. Enter an execution host name or click on the pop-up selection button to navigate to the Select Execution Host pick list to select one. This can be a read-only field if it was created by Models-3.

Execution Path: The path, relative to the current directory, of the execute location. Enter an execution full path or click on the pop-up selection button to navigate to the File Selection pick list to select one. This can be a read-only field if it was created by Models-3.

Expiration Date: The date when a (science) component was/is to be rendered obsolete. Set by the Models-3 System Administrator, or the last date before a superseding version takes effect. (See Version Expiration Date.)

External Boundary Thickness: The number of cells along the perimeter of the grid that are considered to be external to the simulation domain, but are still used for computational purposes.

File Format: The format of the file. Click on the pull-down menu button to view the format options. Values include:

- ASCII.
- IOAPI/NETCDF.
- OSF1 Binary.
- SAS.
- Sun Solaris Binary.
- Others.
- Unknown.

File Full Path: The path, relative to the current directory, to where the dataset will be copied or moved. Enter a file path, or click on the pop-up selection button to navigate to the File Selection pick list to select one.

File Host: The name used in conjunction with the path name to determine where to find the support file. Enter a host name, or click on the pop-up selection button to navigate to the File Host pick list to select one.

File Size: The size of the file in KB.

Filter: One or more entries on Find screens used to screen or "filter" entities from a larger pool (e.g., by a version number or the initials of the individual who last modified the entity).

Flag Name: The name of a command line argument parameter.

Flag Value: The value assigned to a Flag Name.

Formula: An expression for obtaining a result based on symbols and figures.

Full Domain: PAVE selection button to use the full domain.

Generate: The Generate button has various behaviors. Usually, it will trigger the generation of FORTRAN include files for use in building a model.

Global Flag:

- verbose.
- parse_only.
- compile_all.
- one_step.
- clean_up.

Graph Type: Graph types vary with respect to where they are used. They may take the form of pie charts, bar charts, line graphs, multiple line graphs (for comparison), etc.

Grid: The grid under which the executable was built. Click on the pop-up selection button to view options.

Grid Family: An association among related spatial specifications, including Horizontal Grids and Vertical Layerings. A many-to-one association.

Grid ID: See ID.

Grid Name: A 16-character maximum name for a grid.

Grid Size: Information on the size of the grid, which includes:

- Columns. The number of grid cells in the X-direction.
- Rows. The number of grid cells in the Y-direction.
- External Boundaries Thickness (in cells).

Grid Status:

- Protected.
- Unprotected.

Horizontal Coordinate System Name: A 16-character name that identifies a coordinate system.

Horizontal Coordinate System Type: A list of valid horizontal coordinate system types. Click on the pull-down menu button to view values:

- Latitude/Longitude.
- Lambert.
- Mercator.
- Stereo graphic.
- Universal Transverse Mercator.
- Matrix.

Horizontal Datum: Describes the reference datum for horizontal measure. Click on the pull-down menu button to view horizontal datum values. Values are:

- 1 NAD27.
- 2 NAD83.
- O Other.
- U Unknown.

Horizontal Grid: In Dataset Manager, a pick-list selectable value used for filtering through dataset metadata to find those that match the horizontal grid type.

Host: The computer on which the program is executed, or on which a program or dataset resides.

ID: The unique identifier generated by the Models-3 system. This ID is guaranteed to be unique across all Models-3 systems. The ID can be for a plan, study, dataset, program, vertical layering mechanism, grid, coordinate system, horizontal coordinate system.

Include: A file of C or Fortran statements that is included during compilation of a model being built. Include files are used to tailor models to specific studies.

Inventory Data Analyzer Path: Path to Inventory Data Analyzer execute module. Field will contain a default path.

Key Words: The key words by discipline or subject area to facilitate future searching. Click on the pop-up menu button to navigate to the Key Word Selection list to select or add a new key word.

Label: The name of the entity or attribute.

Larger Secant Angle: A parameter (in degrees latitude) of the Lambert Map Projection.

Last Updated By: Identifies the last user to modify a grid, coordinate system, vertical layering, or case.

Latitude Coordinate Origin (within tangent circle): A parameter (in degrees latitude) of a Mercator projection.

Layer Number: The number of the vertical layer.

Layer Value: The value of the vertical layer.

Legend: Selection button on Study Planner Plan Layout page to display a legend of plan symbols and execution statuses.

Level: PAVE parameter. See PAVE documentation.

Level Range: PAVE parameters. See PAVE documentation.

- min.
- max.

Library: The location of the library files.

Line Width: Optional Vis5D parameter to specify the line width of the contour lines.

Linear (button): Within the vertical layering, given the number of layers and the top of the specification, this button will trigger a linearly spaced distribution of the layers.

Link: Selection from the Study Planner Plan Editing tool on the Plan Layout page. Used to create links between nodes.

Link Flags: See compiler flags.

Link Type: (From the "Link as:" menu selection on the Define Link Information window)

- Environment Variable.
- Command Line Argument.
- User Input File.

Log (button): Within the vertical layering, given the number of layers and the top of the specification, this button will trigger a logarithmically spaced distribution of the layers.

Logical Name: The name used to associate an input dataset with a program.

Longitude Coordinate Origin (within tangent circle): A parameter (in degrees longitude) of a Mercator projection.

Mandatory: Indicates whether this input file is required for running the program. Click the box to select. Click again to unselect.

Map File: Full map file path for use in Vis5D and PAVE.

Map File Host: Host computer where the Map File is located.

Map Projection Units: The unit values for the map projections. Latitude/Longitude projection units are in degrees. Lambert, Mercator, Stereo graphic, and Universal Transverse Mercator (UTM) projection units are in meters.

Max Length: On the Describe Variable window, the maximum length of the field for the variable in an ASCII record.

Measurement Accuracy: Describes the accuracy of the measurement in meters.

Measurement Method: Describes the means used to determine the vertical measure. Click on the selection button to view measurement values. Values include:

- A1 Altimetry.
- G1 GPS Carrier Phase Static Relative Positioning Technique.
- G2 GPS Carrier Phase Kinematic Relative Positioning Technique.
- G3 GPS Code Measurements - Differential.
- G4 GPS Code Measurements - Precise Positioning Service.
- G5 GPS Code Measurements - Standard Positioning Service SA Off.
- G6 GPS Code Measurements - Standard Positioning Service SA On.
- L1 Precise Leveling from a Bench Mark.
- L2 Leveling between Non Bench Mark Control Points.
- L3 Trigonometric Leveling.
- P1 Photogrammetric.
- S1 Classical Surveying Techniques.
- T1 Topographic Map Interpolation.
- OT Other.

Mechanism ID: See ID. For chemical mechanisms.

Mechanism Name: A 16-character name for a chemical mechanism.

Mechanism Status:

- Protected.
- Unprotected.

Mechanism Table: The file from the chemical reaction table.

Medium Label: Identifies the offline medium of an archived dataset.

Memory: The physical memory (in MB) of a host. Also a Vis5D parameter.

MEPPS Path: Full path, including file name, to MEPPS executable.

Model Name: The name of a model being built by a model builder.

Module: Code that is compiled and included in a Models-3 conforming model. A conforming model is comprised of many modules, one module of each class.

Modules Configuration File: File containing a list of submodule selection information used by the Models-3 Builder to construct a tailored executable model.

Modules Configuration Name: The name of a configuration object used to build a model by bldmod.

Multitime: PAVE parameter. See PAVE documentation.

Name: The name of the dataset, coordinate system, device, program, spatial grid, study, variable, plan, node, variable, grid, horizontal coordinate, coordinate system, vertical layering, case, or mechanism.

Node: Selection from the Study Planner Plan Editing tool on the Plan Layout page. Used to create nodes when creating a study plan.

Number of Layers: Within the vertical layering, the quantity of layers for that named specification.

Number of Time Steps: The number of steps in the simulation.

Official: Flags the dataset or science manager objects as official and therefore unchangeable. This may be set only by the System Administrator in the System Administration component.

Operating System: The Operating system under which the program or executable was built and updated. Click on the pop-up selector button to view operation system values. Values include:

- DIGITAL UNIX 4.0
- DIGITAL UNIX 4.1
- IRIX 5.3
- IRIX 6
- NT 4.0
- NT 5.0
- SUN 4.1.3
- SUNOS 5.6
- UNICOS 9.0.

Organic Gas Type:

- VOC (Volatile Organic Compound)
- TOG (Total Organic Gases)
- ROG (Reactive Organic Gases)
- THC (Total Hydro-Carbon)
- NMHC (Non-Methane Hydro-Carbon)
- NMOG (Non-Methane Organic Gases)

Organization: Organization for which a dataset contact works on Define Contact Information window.

Origin (X, Y) SW Corner: X is in the east-west direction or longitude; Y is in the north-south direction or latitude. Units of measure depend upon the selected coordinate system type. You must enter a:

- Column Offset.
- Row Offset.

Output Disk Space: The estimated disk space (in KB) needed for output.

Owner ID: The ID of owner of the dataset.

Parent Dataset: Optional name of a dataset to which another dataset is generated from.

Parent Grid: A grid that is "nested" inside a coarser grid and positioned relative to the origin of the coarse grid. The Southwest corner of a nested grid is relative to the Southwest corner of the parent grid.

Parent ID: The ID of the dataset from which the new dataset was copied. You may manually input the ID of a closely related dataset.

Password: The password for a host (only needed if the host is not Orbix enabled).

Plan Acronym: A short (up to five characters) name for the plan, which is used in the default file path.

Plan ID: See ID.

Plan Name: A string of characters (80 max.) for the plan name. You may name the plan whatever you wish. but this must be unique within the study.

Platform: The hardware platform on which the program was built or operated. Click on the pop-up selector button to view hardware platform values. Values include:

- ALPHA. (Digital Equipment Corporation platform)
- CRAY. (Cray Super Computer)
- OTHERHW.
- PC - Personal Computer.
- SGI. (Silicon Graphics Inc. UNIX-based workstation)

- SUN. (Sun UNIX-based workstation

Point of Tangency (Latitude): A parameter (in degrees) for the Stereo graphic Map Projection.

Point of Tangency (Longitude): A parameter (in degrees) for the Stereo graphic Map Projection.

Point-Line-Area: A flag that indicates the type of latitude/longitude data. Click on the pull-down menu button to view the value options. Values are:

- P Point (default).
- L Line.
- A Area.

Position: Position within an organization for which a dataset contact works (Define Contact Information window).

Process Analysis Control Grid: A grid incorporated to build the process analysis executable.

Process Analysis Control Vertical Layer: A vertical layer incorporated to build the process analysis executable.

Program: A program is an executable or a script file used for Models-3 framework to formulate a study in study planner. Enter an existing program, or click on the pop-up selection button to navigate to the Select Program pick list to select one. This name does not need to be unique.

Program Acronym: A short (up to five characters) name to describe the program, which is used in the default file path.

Program ID: See ID.

Program Name: See Name.

Program Status: The status of the program. The status may be one of the following:

- Official (Protected - saved in a Models-3 directory with write protection)
- Unofficial (Unprotected - stored in a user directory where the owner or possibly a user can change it.)

Program User ID: The ID of the person who created the program.

Project: The name of the project.

Projection: Click this button to view windows for the different map projection types for the Coordinate System.

Propagate Changes to All Existing Nodes: Toggle button on the Define Plan Level Variables screen in Study Planner. Used to propagate changes in execution defaults such as the execution host.

Protected: Indicator that an entity is not changeable by the user.

Purpose: The summary of the intentions with which the dataset was developed. It explains why the dataset was created, and what it was intended to provide or prove.

Quality Indicator: Indicates the status of quality control of the dataset. Click on the pull-down menu button to view the quality indicator options. Values include:

- Failed.
- Not QC'ed.
- Passed.

Register: Toggle switch on Define Plan Level Variables window to invoke automatic registration of datasets generated by a study plans execution.

Reset: On the Study Planner Plan Layout page, this button resets a study plan back to a ready-to-run condition. (Useful after a plan termination.)

Rows: The number of grid-cell rows in a grid. Specified on the Grid Definition page of the Science Manager Grid Detail Window.

Row Offset: Offset (in cells) of SW corner of a grid in the Y direction (+ up, - down) relative to the SW corner of a parent grid. If no parent grid exists, then the offset is zero.

Save Duration: Length of time (in days) to save files that are auto-registered as a result of a study plan execution. This is currently metadata only and does not result in files being automatically deleted when the save duration time has expired.

Scale: Describes the scale of the source used to determine the latitude and longitude. Click on the selection button to view scale values. Values are:

- 1 >=1:500.

- 2 1:501 - 1:5,000.
- 3 1:5,001 - 1:10,000.
- 4 1:10,001 - 1:15,000.
- 5 1:15,001 - 1:20, 000.
- 6 1:20,001 - 1:25,000.
- 7 1:25,001 - 1:50,000.
- 8 1:50,001 - 1:100,000.
- 9 < 1:100,000.
- A 1:10,000.
- B 1:12,000.
- C 1:15,840.
- D 1:20,000.
- E 1:24,000.
- F 1:25,000.
- G 1:50,000.
- H 1:62,500.
- I 1:63,360.
- J 1:100,000.
- K 1:125,000.
- L 1:250,000.
- M 1:500,000.
- N None.
- O Other.
- Un Unknown.

Scatter: PAVE parameter. See PAVE documentation.

Science Chemical Mechanism: See Chemical Mechanism.

Science Grid Family: See Grid Family.

Science Vertical Layer: See Vertical Layer.

Select (button): Used to select grid numbers in a given grid family for processing.

Skip Lines: The number of header lines (records) to skip on an ASCII-format input file to the File Converter tool. This allows one to strip off comments or other header information off the front of an ASCII file.

Smaller Secant Angle: A parameter in degrees latitude for the Lambert Map Projection.

Source: Describes the party responsible for collecting or otherwise providing the latitude and longitude. Click on the selection button to view source options. Options are: .

- R1 EPA Region 1.
- R2 EPA Region 2.
- R3 EPA Region 3.
- R4 EPA Region 4.
- R5 EPA Region 5.
- R6 EPA Region 6.
- R7 EPA Region 7.
- R8 EPA Region 8.
- R9 EPA Region 9.
- R0 EPA Region 10.
- << 01-55 Use FIPS Code For States >>.
- RE Regulated Entity.
- CR Contractor.
- TR Tribe.
- HQ EPA Headquarters.
- DB Dun & Bradstreet.
- PV Private.
- FA Other Federal Agency.
- OT Other.
- UN Unknown.

Specification: Button on the File Converter tool main window to display a Formulate File Description window for specifying the format for a file.

Start Column: When specifying an ASCII file format to the File Converter tool, selecting the Variables *Insert* button invokes a Describe Variable window where the user specifies the start column in the record for the variable.

Start Date: The start date in calendar (yyyy/mm/dd) format for a CASE (or temporal specification). Enter a date, or if you have entered a date in the Julian Day field, click the Calculate *Calendar* button for the system to calculate the calendar start date. Also, if you know the end date information and have entered it in the appropriate fields, you may click on the Calculate *Start* button for the system to calculate the start date.

Start Time: The start time of the readings in the dataset. The format is in hh:mm:ss. Click on the pull-down menu button next to the time field to select A.M., P.M., or Military time. Also the start time of a temporal specification or CASE.

State: The state or province of the organization or individual.

Status: Indicates the status of the dataset. Click on the pull-down menu button to view status options. Values include:

- Archived.
- Online (default).
- Removed.

Street: The address for the organization or individual.

Study: A study is a composition of plans. Enter an existing study, or click on the pop-up selection button to navigate to the Select Study pick list to select one.

Study Acronym: A short (up to five characters) name for a study, which is used in the default file path for the study.

Study Global Environment Variables: Global environment variables are a means to provide run-time input to the study and all its plans.

Study ID: See ID.

Study Name: The short name given to a study. This may not be unique across sites, but must be unique within one autonomous site.

Study Status: The status for a study:

- Protected.
- Unprotected.

Subdomain: A domain within a larger domain. A domain has both a horizontal component and a vertical component. PAVE parameters for a subdomain (x-min., x-max., y- min., y-max.).

Submit User ID: The User ID of the user submitting a program or model for execution. User's login ID.

Supplemental: This captures any additional information about the dataset. It is used for information that is not specifically requested in any other part of the meta data.

Suspend: Toggle button.

Telephone: The telephone number of the organization or individual.

Time: The time when an action was taken against a dataset. Defaults to current time if nothing is entered. Click on the pull-down menu button to select A.M., P.M., or Military. Also, the start time or end time of a temporal specification or CASE.

Time Convention: The time zone in which a temporal specification (CASE) is defined. Click on the selection button to view the time convention values. Values include:

- Atlantic Standard.
- Atlantic Daylight Saving.
- Central Standard.
- Central Daylight Saving.
- Eastern Standard.
- Eastern Daylight Saving.
- Mountain Standard.
- Mountain Daylight Saving.
- Pacific Standard.
- Pacific Daylight Saving.
- GMT.
- GMT+1.
- GMT-1.
- GMT+2.
- GMT-2.
- GMT+3.
- GMT-3.
- GMT+4.
- GMT-4.
- GMT+5.
- GMT-5.
- GMT+6.
- GMT-6.
- GMT+7.
- GMT-7.
- GMT+8.
- GMT-8.
- GMT+9.
- GMT-9.
- GMT+10.
- GMT-10.
- GMT+11.

- GMT-11.
- GMT+12.
- GMT-12.
- None.

Time Step: The interval into which the time period is divided. Click on the pull-down menu button to select years, months, weeks, hours, minutes, or seconds. Click on the selection button to view the time step options. If the time value you wish to use in not included in the list, you may type in your own time value. Time step values include:

- 60.
- 30.
- 15.
- 10.
- 5.
- 4.
- 3.
- 2.
- 1.
- Not Applicable.

Time Step On: The point at which the data is generated or collected during a time step. Values are:

- Beginning.
- Ending.
- Centered.
- Irregular.

Title: Title for a graphic using the visualization tools.

Topography File: File defining terrain characteristics for a specified area. Used by the Terrain program in the MM5 application system of programs.

Topography File Host: Host computer where a topography file may be accessed.

Type: A predefined set of dataset types (land use, terrain, etc.). Enter a dataset type or click on the pop-up selection button to navigate to the Select Data Type pick list to select one.

Unit: The units that describe the attribute. Enter a unit or click on the pop-up selection button to navigate to the Select Unit pick list to select one.

Units: The unit that accompanies the accuracy value. Click on the pull-down menu button to view unit values. Values are:

- 1 Degrees.
- 2 Minutes.
- 3 Seconds.
- 4 Meters.
- 5 Feet.
- 6 Kilometers.
- 7 Miles.

Unit Category: For the File Converter tool, when specifying a file description and inserting variables into the description, you can optionally click the pick-list button to select a unit category (e.g., area, length, pressure, etc.).

Universal Transverse Mercator (UTM) Zone: Range of +/-1 to +/-60.

Update Physical: On the Define Link Information window for a Study Plan link you can click on the *Update Physical* button to display a Define Physical File Location window whereby you can assign a new physical file name to a logical file input to or output from a study plan node.

User Input File Template: A template that sets up the user input file. Enter a user input file template, or click on the pop-up selection button to navigate to the User Input File Template Pick list to select one.

User ID: User's ID as registered in the Models-3 database and required for accessing various host systems.

UTM: Universal Transverse Mercator map projection.

Variable Description: A brief, textual description sufficient to identify this item from among a list of similar items.

Variable Name: The name of the environment variable (one word).

Variable Value: The value of the environment variable, which may be changed at the study plan level or when a program is annotated to a node.

Vector: PAVE data file parameter. See PAVE documentation.

Variable Name: The name of the environment variable (one word).

Variable Value: The value of the environment variable, which may be changed at the study plan level or when a program is annotated to a node.

Vector: PAVE data file parameter. See PAVE documentation.

Verify: On the Study Planner Plan Layout page you can use the *Verify* button to have Study Planner check that all the files and programs are available prior to executing the plan.

Verification: Indicates that the latitude and longitude have been verified by EPA staff, grantees, or contractors through a given process. Click on the selection button to view verification values. Values include:

- A Proximity to Polygon Centroid (Zip Code).
- B Proximity to Polygon Centroid (County).
- C Proximity to Polygon Centroid (Other).
- D Proximity to Alternative Facility Coordinate.
- E Point in Polygon (Zip).
- F Point in Polygon (County).
- G Point in Polygon (Other).
- H Verified Relative to Map Features (1:100K or Tiger).
- I Verified Relative to Map Features (1:24K).
- J Verified Relative to Map Features (Others).
- K Ground Truth Conducted.
- L Verified, Unknown Method.
- M Not Verified.

Version: Indicates the major and minor version in the form of X.X.

Version (button): An indicator to denote the status of a science (CAMS) module. Values are:

- Release.
- Development.

Version Creation Date: The creation date for the science component version, that is the earliest date this version could be used in Models-3.

Version Expiration Date: The date on which the science component version was superseded by a later version. The format is mm/dd/yyyy.

- H - Height above Mean Sea Level (meters).
- Sigma-P Hydrostatic (Pascals).
- Sigma-P Nonhydrostatic (Pascals).
- Sigma-Z (Pascals).
- Not Applicable.

Vertical Datum: Describes the reference datum for vertical measure. Click on the pull-down menu button to view vertical datum values. Values are:

- 1 NAVD88.
- 2 NAVD29.
- 3 Elevation from Mean Sea-Level.
- 4 Local Tidal Datum.
- O Other.
- U Unknown.
- N Not Applicable.

Vertical Layer Status: The status of the vertical layer:

- Official.
- Unofficial.

Vertical Layering: The name of the vertical layer associated with the grid. Enter an existing vertical layer, or navigate to the Select Vertical Layering pick list to select one.

Vertical Layering Generation: The generation of FORTRAN include files for a model that provide the vertical layer data at compile time.

Vertical Layering ID: See ID.

Vertical Layering Name: A 16-character name for a vertical layering.

VGTop: The value for the top of the vertical layering.

View Grid: On the Science Manager Grid Definition page, selecting the *View Grid* button will invoke a grid view program to display the grid on the screen. (Click *Exit* on the Grid Detail window to terminate the grid view. Attempting to close the grid view window by clicking on the "" in the upper left corner of the window can cause problems because of the grid view software used.)

Working Directory: The directory where to send the files.

Working Host: The machine on which the files will reside.

Work Space Path: Specify a directory for MEPPS work space.

X Center (East-West): The east-west parameter to define the origin in the UTM Map Projection.

X Center (-180 < =X <= 180, Longitude): The longitude parameter of the origin for the Mercator Map Projection.

X Origin: The position of a grid's SW corner relative to the X-Center specified for a Lambert projection.

X Value: The value of the X coordinate.

Y Center (-90 <= Y <= 90, Latitude): The latitude parameter of the origin for the Mercator Map Projection.

Y Center (North-South): The north-south parameter to define the origin in the UTM Map Projection.

Y Origin: The position of a grid's SW corner relative to the Y-center specified for a Lambert projection.

Y Value: The value of the Y coordinate.

Zip Code: The postal code of the organization or individual.

APPENDIX D

Data Sources

1.0 INTRODUCTION

Data for meteorology, emissions, chemistry transport, and air quality observations are identified and discussed in this appendix. Some of the data are optional.

2.0 METEOROLOGICAL DATA

For the initial public release of Models-3, all meteorological data will be generated by MM5 Version 2 as described in Chapter 5. MM5 requires static data and case-specific meteorological data. Pre-formatted meteorological data used as input to MM5 are available from the National Center for Atmospheric Research (NCAR) and from Unidata, a National Science Foundation-sponsored program that manages and distributes atmospheric and related data. Other sources are available that have archives of MM5 simulation outputs that may be used, however, they are not in the business of providing data so arrangements would need to be made on an individual basis. One such source is Nelson Seaman at Penn State University.

Much of the meteorological data used in Models-3 resides at EPA's National Environmental Supercomputing Center in Bay City, MI.

For information on how to obtain data from NCAR see **http://www.mmm.ucar.edu/mm5/scd-acct.html** .

For information on how to obtain data from Unidata see **http://www.unidata.ucar.edu** .

2.1 Static MM5 Data

All the static data that are needed to run MM5 are provided on the Models-3 installation tapes. The static data includes:

- Land-use tables at 60, 30, and 10 minute resolutions.
- Terrain height tables at 60, 30, 10, 5, and 0.5 minute resolutions.
- Climatological sea-surface temperature database (if observed SST data are not available).

In addition, there are tables which are only needed for generating plots using NCAR Graphics in TERRAIN, RAWINS, and GRAPH. They are:

- Map table (map.tbl).
- Map area table (maparea.tbl).
- Map contour table (con.tbl).
- Land-use color table (luco.tbl).
- Map color filling table (mapfi.tbl).

- Color filling table (confi.tbl).
- Global Rawinsonde upper air observations (RAOB) station table.
- EZMAP area identification table (ezids).

2.2 Meteorological Data

Files of meteorological data archived on the NCAR Mass Storage System (MSS) are referred to as MSS files. These data come from five different sources:

- European Centre for Medium-Range Weather Forecasts (ECMWF) Northern-Hemisphere analyses (for the years 1980 through 1989), with 2.5 degree latitude-longitude resolution.

- Navy Daily Northern Hemisphere Sea Surface Temperature (SST).

- National Meteorological Center (NMC) Global Analyses, with 2.5 degree latitude-longitude resolution which includes the Operational RAOB and Surface (SFC) Data.

- ECMWF Basic Level III Consolidated dataset global analyses for the Tropical Ocean and Global Atmospheric (ECMTOGA) program (beginning with 1985) with 2.5 degree latitude-longitude resolution. (Requires special authorization to use.)

- Near real-time global NMC medium range forecast analyses, available from Unidata, with 2.5 degree latitude and 5.0 degree longitude resolution. For any particular level at which data are not available, fields are either derived from other existing fields or generated by vertical or temporal interpolation.

Periodically updated catalogs of these files are available from NCAR via anonymous file transfer protocol (ftp). To access the files, an account is required for the NCAR CRAY YMP (host name is Shavano).

Table D-1 summarizes the meteorological data files acquired from NCAR for testing.

Meteorological Data	NCAR Catalog #	Dates
ECMWF Global Grids 1980 - 1989	Y02398*	880701-881231 00Z
	Y02396*	880701-881231 12Z
Navy Daily Northern Hemisphere SST	K8616K*	1983FEB02-1990JUN30
NMC Global Analyses	K4496K	1987JUL01-1987JUL15
NMC Operational RAOB	K6487K	1987JUL05-1987JUL23
	K7477K*	1988 JUL Timeframe
	K7478K*	1988 JUL Timeframe
	K7481K	1988 JUL Timeframe
NMC SFC Data	K6643K	1987JUL05-1987JUL11
	K6644K	1987JUL12-1987JUL18
	K7465K*	1988JUL31-1988AUG06
	K7466K*	1988AUG07-1988AUG13
	K7469K	1988AUG28-1988SEP03
	K6677K	1987JUL05-1987JUL11
	K6678K	1987JUL12-1987JUL18
	K7471K*	1988JUL31-1988AUG06
	K7472K*	1988AUG07-1988AUG13
	K7475K	1988AUG2 -1988SEP03
Unidata		None

Table D-1. Meteorological Data Used for Testing

The meteorological data files used by MM5 are as follows:

- Analysis files (may be NMC, ECMTOGA, ECMWF, or Unidata):
  - NMC (1 or 2 MSS files), depending on simulation period.
  - ECMTOGA (1 or 2 MSS files), depending on simulation period.
  - ECMWF data at 0000 Universal Time Coordinate (UTC) and at 1200 UTC (1 or 2 MSS files each), depending on simulation period.

- Unidata files in NetCDF format for each of the following:

  - Pressure data.
  - Geopotential height.
  - U wind speeds.
  - V wind speeds.
  - Temperatures.
  - Relative Humidities.

- Snow-cover data (if episode is during winter/early spring).

- Sea Surface Temperature data (may be NMC, NAVY, CLIM, or Unidata):

  - NMC (for only ECMTOGA analysis; MSS file).
  - NAVY (MSS file).
  - CLIM (Global climatological tables; MSS file).
  - Unidata.

The following data are required if the RAWINS script variable INOPS = "ARCHIVE":

- Upper air RAOB sounding data (1 or more MSS files).
- 6-hour surface observation data.
- 3-hour surface analysis data.

The following data are required if the RAWINS script variable INOPS = "UNIOBS":

- Upper air analysis data (1 or more Unidata files in NetCDF format).
- Surface analysis data (1 or more Unidata files in NetCDF format).

3.0 EMISSION DATA

Appendix K (Emission System Software and Data Files) contains more detailed information on emission data. For the initial public release of Models-3, the emission data will be generated by MEPPS. The data needed to execute MEPPS consists of the following types:

- Biogenic Emissions Inventory System (BEIS2) Land Use/Land Cover and MCIP-output meteorology data (i.e., temperatures at 1.5 meters above ground and solar radiation).

- Census and Political Boundary Data.

- Road Network Data.

- Raw Emission Inventory Data.

- Other System Input Files and Databases.

3.1 Political Boundary and Census Tract Data

The political boundary and population/housing census tract data (48 states) used by MEPPS are provided on the Models-3 installation tapes. This information is from the 1990 census database. This data are used by ARC/INFO to generate input data used by MEPPS.

Land Use / Land Cover data is used by MEPPS for spatial surrogates. The census tract data was generated from Topologically Integrated Geographic Encoding Referencing (TIGER)/line files.

The Motor Vehicle Emission Model requires national ARC/INFO coverage of major transport routes and land use (taken from the Federal Highway Administration and TIGER/line files). This data is included on the Models-3 installation tapes.

3.2 Raw Emission Data

Raw emission data is converted into the required MEPPS format by the Input Processor (INPRO). The following National emission inventory and temporal factor files are provided on the Models-3 installation tapes:

- 1985 Inventory.
- 1988 Interim Inventory.
- 1990 Interim Inventory.
- 1990 Updated Inventory.
- 1990 National Emission Trend (NET) Inventory.
- 1995 NET Inventory.
- 1995 Inventory for southern provinces of Canada.
- Default Temporal Allocation Factor (TAF) files needed to generate the day-specific emission files.

(No emission data is available for Mexico.)

3.3 Other System Input Files and Databases

The MEPPS system input files, lookup table files, etc. are provided for the area source, point source, and biogenic models. Also, the required input files and lookup table files are provided for mobile source and the Speciation Model Mechanism. The Mobile Source Vehicle miles

traveled inventory data and the Mobile 5a files are also provided. (The input control file for Mobile 5a must be provided by the user.)

4.0 CHEMISTRY-TRANSPORT DATA

Chemistry-transport data needed for the initial release of includes:

- Absorption Cross Section and Quantum Yield.
- Tropospheric Ozone and Total Column Ozone.

4.1 Absorption Cross Section and Quantum Yield

Absorption cross section and quantum yield data are provided on the Models-3 installation tapes. They are required for calculating the photolysis rates in the J-value processor (JPROC). These data are molecular properties that are functions of wavelength and temperature and are unique to each species.

4.2 Tropospheric Ozone and Total Column Ozone

The absorption by ozone is an important radiative process that is calculated in CMAQ using NASA recommendations. Ozone data for the tropospheric layer (surface to about 12 kilometers) from the Total Ozone Mapping Spectrometer (TOMS) may be used with JPROC, but is not required. This data is at a 1.0 degree by 1.25 degree resolution. The data is in dobson units and covers from approximately 50 degrees north to 50 degrees south. The tropospheric ozone data is available from November 1978 through May 1993. The Total Column Ozone data is available from NASA on a CD. The CD contains daily gridded ozone data and statistical data for the TOMS instrument on the Nimbus 7 spacecraft. Nimbus 7 is in a South-North sun synchronous polar orbit, measuring ozone over the entire world every 24 hours.

5.0 OBSERVATION DATA FOR AIR QUALITY MODEL EVALUATION

Data for the evaluation of an air quality model may be obtained from many sources. The ones included below are those that have been used by the Models-3 developers.

5.1 Air Quality Data

The air quality observation data provided on the Models-3 installation tapes is for July 11-15, 1995, and is the data used in the tutorial study. Multiple data sources were used and the data collected into a single file. The data is average hourly surface ozone concentrations from the following networks: Aerometric Information Retrieval System (AIRS); Clean Air Status and

Trends Network (CASTNet); Southeastern Consortium Intermediate Oxidant Network (SCION); and North American Research Strategy for Tropospheric Ozone (NARSTO) - Northeast.

5.2 Satellite Data (GOES Clouds)

This data was acquired from the National Climatic Data Center in Asheville, NC. (The data is archived at the University of Wisconsin.) The data is not required, but is very useful for establishing the credibility of air quality estimates when there is convection activity.

5.3 Composite Rainfall Derived from NWS Radar

The use of this data, like the GOES cloud data, is not required but is useful for confirmatory purposes. The rainfall rates are at 4 km resolution and are for every 15 minutes. The data is available from the Marshall Space Flight Center:

User Services Office
Global Hydrology and Climate Center
977 Explorer Blvd.
Huntsville, AL 35806

Phone: 256/922-5932
E-mail: ghrc@eos.nasa.gov

APPENDIX E

Vis5D Version 4.2

1.0 OVERVIEW OF Vis5D

Vis5D is a software system for visualizing data made by numerical weather models and similar sources. Vis5D works on data in the form of a five-dimensional rectangle. That is, the data are real numbers at each point of a "grid" which spans three space dimensions, one time dimension and a dimension for enumerating multiple physical variables. Of course, Vis5D works perfectly well on data sets with only one variable or one timestep (i.e. no time dynamics). However, your data should have some depth in all three spatial dimensions.

The Vis5D system includes the Vis5D visualization program, several programs for managing and analyzing five-dimensional data grids, and instructions and sample source code for converting your data into its file format. The Vis5D source code is included so you can modify it or write new programs. Sample data sets from the LAMPS model are also included from Bob Schlesinger's thunderstorm model, so you can work through the examples.

Vis5D version 1.0 was written by Bill Hibbard and Dave Santek of the University of Wisconsin Space Science and Engineering Center, supported by the NASA Marshall Space Flight Center, and by Marie-Francoise Voidrot-Martinez of the French Meteorology Office. Later version enhancements were written by Bill Hibbard, Brian Paul, and Andre Battaiola. Dave Kamins and Jeff Vroom of Stellar Computer, Inc. provided substantial help and advice in using the Stellar software libraries. Simon Baas and Hans de Jong of the Netherlands ported Vis5D to HP workstations. Pratish Shah of Kubota Inc. ported Vis5D to the Kubota Alpha/Denali workstation. Mike Stroyan of Hewlett Packard added PEX support.

Vis5D is offered under the terms of the GNU General Public License, which you can find in the file "NOTICE". As the notice states, there is no warranty for the Vis5D system, but we would be interested in hearing about your questions and problems. Also, if you would like to be added to the Vis5D mailing list, send e-mail to:

Bill Hibbard (email: whibbard@macc.wisc.edu) at:

Space Science and Engineering Center
University of Wisconsin - Madison
1225 West Dayton Street
Madison, WI 53706

This document is the complete guide for using Vis5D. It is available in PostScript or ASCII.

Contents:

1.1 Vis5D Documentation on the World Wide Web

The Vis5D Home Page is available on the World Wide Web page at URL:

http://www.ssec.wisc.edu/~billh/vis5d.html

This is linked to another Web document that describes how to use Vis5D files as a World Wide Web medium for exchanging model output, at URL:

http://www.ssec.wisc.edu/~billh/view5d.html

There is a Web document describing the Vis5D API (Application Programmer Interface), intended to help system developers use Vis5D as a visualization subsystem of other systems, at URL:

http://www.ssec.wisc.edu/~billh/api.html

There is a Web document describing the Vis5D Tcl scripting interface at URL:

http://www.ssec.wisc.edu/~billh/script.html

Copies of the api.html and script.html files are included with the Vis5D ftp distribution.

2.0 SYSTEM REQUIREMENTS AND INSTALLATION

In the following sections we describe the hardware and software required to run Vis5D and detail how to install Vis5D on your system.

2.1 System Requirements

Vis5D currently works with the following systems. In all cases, at least 32 MB of RAM is recommended and at least an 8-bit color display are required:

1. Silicon Graphics workstations:
   IRIX version 4.0.1 or higher.
   Multiple processors are used when present.

2. IBM RS/6000 workstations:
   Model 320H or higher.
   AIX version 3 or later.
   3-D graphics hardware is supported through OpenGL.

3. HP series 7000 or 9000 workstations:
   HP-UX A.09.01 or later.
   PEX optional.

4. Sun Sparc workstations:
   SunOS 5.x or later.

5. DEC Alpha workstations:
   OSF/1 V1.3 or later.
   Kubota Denali Graphics hardware supported with KWS V1.3.3 or later and NPGL Run-time license.

6. IBM PC compatibles with Linux:
   75MHz Pentium CPU or faster recommended.
   Linux 1.0 or later.
   XFree86 (X window system) must be installed.

Note that on systems which don't have 3-D graphics hardware or OpenGL, all 3-D rendering is done in software using Mesa (an OpenGL work-alike). Be aware that software rendering is rather slow. 3-D graphics hardware is recommended.

If you would like to port Vis5D to a new graphics system or workstation read the "PORTING" file which gives more information. If you succeed, please inform us so that we may add your work to the distribution.

2.2 Installing Vis5D

Vis5D is obtained via anonymous ftp. If you don't have Internet access, you can obtain Vis5D on tape by sending us a blank QIC or DAT and a note explaining what you need.

Here are the installation instructions:

1. Go to the directory in which you want Vis5D installed:
   % cd /usr/mydir

   NOTE: The installation of Vis5D will result in a new subdirectory named "vis5D-4.2/" being created in the current directory.

   NOTE: Be sure that you have write permission in this directory. If you do not, you should become superuser before proceeding. When finished installing Vis5D be sure to set the file ownership and permissions accordingly.

2. Start ftp:
   % ftp iris.ssec.wisc.edu
   or
   % ftp 144.92.108.63

3. Login as anonymous and send your e-mail address as the password:
   Name: anonymous
   Password: e-mail-address

4. Go to the pub/vis5d directory:
   ftp> cd pub/vis5d

5. Transfer files in binary mode:
   ftp> binary

6. Get the Vis5D archive file:
   ftp> get vis5d-4.2.tar.Z

7. Get the optional sample data archive file:
   The vis5d-data.tar.Z file contains topography, map outlines and sample data sets. If you've used Vis5D in the past you can should already have these files and can move them into your new "vis5d-4.2" directory.

   ftp> get vis5d-data.tar.Z

8. Exit ftp:
   ftp> bye

9. Uncompress and un-tar the archive file:
   % uncompress vis5d-4.2.tar.Z
   % tar -xvf vis5d-4.2.tar

10. Change to the newly created vis5d directory:
   % cd vis5d-4.2

11. Optionally uncompress and untar the data file:
   % mv ../vis5d-data.tar.Z
   % uncompress vis5d-data.tar.Z
   % tar -xvf vis5d-data.tar

12. Run make:
   % make

   Make will print a list of systems supported for Vis5D. Look for yours on the list and type the appropriate make command. For example, suppose you have an IBM RS/6000 without OpenGL and 3-D graphics hardware. You should type:

   % make ibm-x

   Vis5D and its utility programs will now be compiled. If you do not have C and/or FORTRAN compilers on your system, this step will fail with an error message such as "cc: Command not found." or "f77: Command not found." In this case you will have to get the appropriate archive of executable programs:

   a. List the "viewer" files. These files are archives which contain vis5d executables for common UNIX systems:
      ftp> dir *viewer*
   b. Make sure you're still in the vis5d-4.2 directory and download the viewer file for your operating system (xxx):
      ftp> get vis5d.xxx.viewer.tar.Z
   c. Exit ftp:
      ftp> bye
   d. Uncompress and un-tar the archive:
      % uncompress vis5d.xxx.viewer.tar.Z
      % tar -xvf vis5d.xxx.viewer.tar

13. Test Vis5D:
    % ./vis5d LAMPS.v5d

    NOTE: To quit, click on the "EXIT" widget button.

14. You may delete the .tar files if desired.

2.3 Manifest

When you are finished installing Vis5D you should have a directory named "vis5d-4.2" which contains the following files and subdirectories:

README	this documentation file in ASCII
README.ps	this documentation file in PostScript
NOTICE	the GNU general public license (copyright)
PORTING	an ASCII document with notes on porting Vis5D
LAMPS.v5d	Sample LAMPS data set
SCHL.v5d	Sample data set: Bob Schlesinger's thunderstorm model
OUTLSUPW	World continental map lines file
OUTLUSAL	Low resolution map of US with state boundaries
OUTLUSAM	Medium resolution map of US with state boundaries
EARTH.TOPO	Earth topography file
api.html	documentation file for the API between Vis5D and its user interface
script.html	documentation file for Vis5D scripting language
*.tcl	example scripts
vis5d	this is the vis5d visualization program
v5dappend	utility to join v5d files together
v5dinfo	utility to see summary of a v5d file
v5dstats	utility to see statistics of a v5d file
v5dedit	utility to edit the header of a v5d file
v5dimport	utility to convert, resample, and reduce v5d files
gr3d_to_v5d	utility to convert a McIDAS GR3D file to v5d format
comp_to_v5d	utility to convert (a) comp5d file(s) to v5d format
listfonts	utility to list fonts available on SGI systems for IRIS GL
src	source code for vis5d
util	source code for the Vis5D utilities
lui5/	source code for LUI user interface library
Mesa/	source code for the Mesa 3-D graphics library
import/	source code for the v5dimport program
userfuncs/	directory of user-written analysis functions

contrib/	software contributed by Vis5D users
convert/	source code for sample data conversion programs

2.4 Customizing

After installation and testing you may want to customize the vis5d program by editing the src/vis5d.h file:

1. The visualization program vis5d assumes your system has 32 megabytes of memory. Although you can override this when you invoke vis5d, it may be convenient to change the default if your system has more than 32 MB. The default number of megabytes is defined by the value of MBS in the src/vis5d.h include file.

2. If you want to specify a different default topography or map file, you can edit src/vis5d.h and change the values for TOPOFILE and/or MAPFILE. For example, if you move the map and topography files to /usr/local/bin, you would specify "/usr/local/data/EARTH.TOPO" and "/usr/local/data/OUTLUSAM" respectively.

When finished changing the src/vis5d.h file you must recompile the programs by repeating installation step 12 above.

If Vis5D is going to be used by multiple users on your system you may want to move the vis5d executables and data files to a common directory tree such as /usr/local:

```
% mv vis5d /usr/local/bin
% mv v5d* /usr/local/bin
% mv OUTL* /usr/local/data
% mv EARTH.TOPO /usr/local/data
```

then change the vis5d.h file as described above to indicate where the map and topography files are stored.

3.0 PUTTING YOUR DATA INTO Vis5D

Vis5D works with data organized as a 5-D rectangle. The first 3 dimensions are spatial: rows, columns, and levels (or latitude, longitude, and height). The fourth dimension is time. The fifth dimension is the enumeration of multiple physical variables such as temperature, pressure, water content, etc.

In addition to the data itself, there are a number of parameters needed to describe a Vis5D dataset: the sizes of the five dimensions (number of rows, columns, levels, timesteps, and

variables), geographic position and orientation of the data (map projection), the names of the variables, the actual times and dates associated with each timestep, etc.

The vis5d visualization program accepts two file formats: v5d files and comp5d files. Both store 3-D data in a compressed format which Vis5D can use quickly and efficiently. Comp5d files are those which were produced by the comp5d program in previous versions of Vis5D. The v5d file format is the new, and preferred, file format used in version 4.0 and later of Vis5D. It is intended to be a replacement for the comp5d format because it more flexible and may be extended in the future.

To view your data with Vis5D you will typically write a conversion program to convert your data files to v5d format. To help you do this we've included four sample conversion programs to guide you. Basically, you just add the instructions to read your file format, we provide the instructions to write the v5d file. See section 3.1 below.

If you have used Vis5D in the past, you may continue to convert your data to McIDAS format and use comp5d to make a compressed file. However, to take full advantage of the new map projections and vertical coordinate system in version 4.0 and higher, you should write a new conversion program to make v5d files.

Another option for getting your data into Vis5D is the v5dimport utility. v5dimport is a new program for file conversion, combining, and resampling. It reads a number of different file formats and can be extended to read new formats. See section 7 for more details.

3.1 Converting Your Data to v5d Format

Files in the v5d format are created with functions from the v5d library. We've included four sample conversion programs which outline how to make a v5d file. They are located in the convert subdirectory. You can choose which one to use as a template for your data converter:

foo_to_v5d.f	A Fortran program which assumes a rectangular lat/long map projection and equally spaced linear vertical coordinate system.
foo2_to_v5d.f	A Fortran program which allows any map projection and vertical coordinate system as well as a different number of vertical levels for each variable.
foo_to_v5d.c	A C program which assumes a rectangular lat/long map projection and equally spaced linear vertical coordinate system.

foo2_to_v5d.c A C program which allows any map projection and vertical coordinate system as well as a different number of vertical levels for each variable.

In any case, each conversion program uses three functions to write the v5d file: v5dCreate (or v5dCreateSimple), v5dWrite, and v5dClose. v5dCreateSimple is used to create v5d files which only specify the most basic parameters. v5dCreate allows more complicated parameters. There are versions of these functions for C and Fortran programs.

Here are the descriptions of the v5dCreate and v5dCreateSimple functions in a format similar to man page documentation. C programmers should note that in the argument descriptions we describe arrays by FORTRAN convention, i.e. A(1) is the first element of A whereas in C this would be A[0].

Fortran-callable functions:

```
integer function v5dcreatesimple (name, numtimes, numvars,nr, nc, nl, varname,
      timestamp, datestamp, northlat, latinc, westlon, loninc, bottomhgt, hgtinc)
character* (*) name
integer numtimes
integer numvars
integer nr
integer nc
integer nl
character*10 varname(MAXVARS)
integer timestamp(*)
integer datestamp(*)
real northlat
real latinc
real westlon
real loninc
real bottomhgt
real hgtinc

integer function v5dcreate (name, numtimes, numvars, nr, nc, nl, varname, timestamp,
      datestamp, compress, projection, proj_args, vertical, vert_args)
character* (*) name
integer numtimes, numvars
integer nr
integer nc
integer nl(*)
character*10 varname(MAXVARS)
```

```
integer timestamp(*)
integer datestamp(*)
integer compress
integer projection
real proj_args(*)
integer vertical
real vert_args(*)
```

C-callable functions:

```
int v5dCreateSimple (name, numtimes, numvars, nr, nc, nl, varname, timestamp,
        datestamp, northlat, latinc, westlon, loninc, bottomhgt, hgtinc)
char *name;
int numtimes;
int numvars;
int nr, nc, nl;
char varname[MAXVARS][10];
int timestamp[], datestamp[];
float northlat, latinc;
float westlon, loninc;
float bottomhgt, hgtinc;

int v5dCreate (name, numtimes, numvars, nr, nc, nl, varname, timestamp, datestamp,
        compress, projection, proj_args, vertical, vert_args)
char *name;
int numtimes, numvars;
int nr, nc, nl[];
char varname[MAXVARS][10];
int timestamp[], datestamp[];
int compress;
int projection;
float proj_args[];
int vertical;
float vert_args[];
```

Arguments used by v5dCreate and v5dCreateSimple:

name	The name of the v5d file to create
numtimes	Number of timesteps (at least 1)
numvars	Number of variables (at least 1)
nr	Number of rows in all 3-D grids (at least 2)

nc	Number of columns in all 3-D grids (at least 2)
varname	Array of variable names: varname(1) = name of first variable varname(2) = name of second variable ... varname(numvars) = name of last variable
timestamp	Array of time labels for the timesteps in HHMMSS format: timestamp(1) = time of first timestep timestamp(2) = time of second timestep ... timestamp(numtimes) = time of last timestep
datestamp	Array of date labels for the timesteps in YYDDD format datestamp(1) = date of first timestep datestamp(2) = date of second timestep ... datestamp(numtimes) = date of last timestep

Arguments used only by v5dCreateSimple:

nl	Number of levels in all 3-D grids (at least 1)
northlat	Latitude of northern edge of box in degrees
latinc	Increment between rows in degrees (positive)
westlon	Longitude of western edge of box in degrees (positive West longitude)
loninc	Increment between columns in degrees (positive)
bottomhgt	Bottom boundary of box in km
hgtinc	Increment between levels in km (positive)

Arguments used only by v5dCreate:

nl	Number of levels in the 3-D grids per variable: nl(1) = number of levels for first variable nl(2) = number of levels for second variable ... = ... nl(numvars) = number of levels for last variable
compress	Compression mode (1, 2 or 4 bytes per grid point)
projection	Indicates type of map projection: 0 = linear, rectangular, generic units 1 = linear, rectangular, cylindrical-equidistant 2 = Lambert Conformal 3 = Stereographic 4 = Rotated

proj_args Projection arguments:
if projection=0 then
proj_args(1) = North boundary of 3-D box
proj_args(2) = West boundary of 3-D box
proj_args(3) = Increment between rows
proj_args(4) = Increment between columns
else if projection=1 then
proj_args(1) = North Latitude bound of 3-D box
proj_args(2) = West Longitude bound of 3-D box
proj_args(3) = Increment between rows in degrees
proj_args(4) = Increment between cols in degrees
else if projection=2 then
proj_args(1) = Standard Latitude 1
proj_args(2) = Standard Latitude 2
proj_args(3) = Row of North/South pole
proj_args(4) = Column of North/South pole
proj_args(5) = Longitude parallel to columns
proj_args(6) = Increment between columns in km
else if projection=3 then
proj_args(1) = Latitude of center (degrees)
proj_args(2) = Longitude of center (degrees)
proj_args(3) = Row of center of projection
proj_args(4) = Column of center of projection
proj_args(5) = Spacing between columns at center
else if projection=4 then
proj_args(1) = North boundary on rotated sphere
proj_args(2) = West boundary on rotated sphere
proj_args(3) = Increment between rows
proj_args(4) = Increment between columns
proj_args(5) = Earth Latitude corresponding to (0,0)
proj_args(6) = Earth Longitude corresponding to (0,0)
proj_args(7) = Rotation angle
endif

vertical Indicates type of vertical coordinate system:
0 = equally spaced levels in generic units
1 = equally spaced levels in km
2 = unequally spaced levels in km

vert_args Vertical coordinate system arguments:
if vertical=0 then
vert_args(1) = height of bottom level
vert_args(2) = spacing between levels

```
else if vertical=1 then
        vert_args(1) = height of bottom level in km
        vert_args(2) = spacing between levels in km
else if vertical=2 then
        vert_args(1) = height (km) of grid level 1 (bottom)
        vert_args(2) = height (km) of grid level 2
        ...
        vert_args(N) = height (km) of grid level N (top) where N is the
        maximum value in the nl array.
else if vertical=3 then
        vert_args(1) = pressure (mb) of grid level 1 (bottom)
        vert_args(2) = pressure (mb) of grid level 2
        ...
        vert_args(N) = pressure (mb) of grid level N (top)
where N is the maximum value in the nl array.
endif
```

The v5dWrite function is used to write a single 3-D grid of data to a v5d file. The grid is identified by a timestep and physical variable number. Here is the synopsis of v5dWrite:

Fortran-callable function:

```
integer function v5dwrite( time, var, data )
integer time
integer var
real data(*)
```

C-callable function:

```
int v5dWrite( time, var, data )
int time;
int var;
float data[];
```

Arguments descriptions:

time	A timestep number in the range [1..numtimes]
var	A variable number in the range [1..numvars]
data	3-D array of grid values; number of values = nr*nc*nl(var) ordered as data[row+nr*(col+nc*lev)] where row increases from North to South, col increases from West to East, and lev increases from bottom to top

The v5dClose function closes the v5d file after the last grid has been written. No arguments are needed. Here is the synopsis of v5dClose:

Fortran-callable function:

integer function v5dclose

C-callable function:

int v5dClose()

Each of the create functions returns 1 when successful and 0 when an error occurs.

Looking at any of the example data conversion programs, you'll see that there are variables which directly correspond to the arguments to v5dCreate/v5dCreateSimple. It is up to you to initialize these variables. For example, you'll have to assign to numtimes the number of timesteps in your dataset, assign to numvars the number of variables in your dataset, etc. After you've initialized all these variables, the v5dCreate (or v5dCreateSimple) call will create the v5d file. If you've failed to initialize any of the variables you will see an appropriate error message.

Next, the conversion program will enter a nested loop inside of which you must insert the code to read your data for the appropriate timestep and physical variable number. Read your data into the array specified. The v5dWrite call will then compress and write the data to the v5d file. Finally, the v5dClose function will be called after all the data has been written.

After you've written and compiled your file converter, you should test it with one of your data files then check that it worked by running the v5dinfo and v5dstats utility programs on the v5d file. If everything looks OK, try running vis5d.

Here is an example of typical values that might be assigned to each variable if one were using the too_v5d.f program:

Assignment	Comments
numtimes = 5	5 timesteps
numvars = 4	4 physical variables
nr = 30	30 rows in each 3-D grid
nc = 40	40 columns in each 3-D grid
nl = 20	20 levels in each 3-D grid
varname(1) = "U"	U (east/west) wind component
varname(2) = "V"	V (north/south) wind component
varname(3) = "T"	Temperature
varname(4) = "P"	Pressure
timestamp(1) = 140000	2:00:00 pm
timestamp(2) = 141500	2:15:00 pm
timestamp(3) = 143000	2:30:00 pm
timestamp(4) = 144500	2:45:00 pm

timestamp(5) = 150000	3:00:00 pm
datestamp(1) = 94036	36th day of 1994 (February 5)
datestamp(2) = 94036	"
datestamp(3) = 94036	"
datestamp(4) = 94036	"
datestamp(5) = 94036	"
northlat = 60.0	Northern boundary of box is at 30 degrees latitude
latinc = 1.0	There is 1 degree of latitude between each of the 30 rows
westlon = 100.0	Western boundary of 3-D box is at 100 degrees longitude
loninc = 0.5	0.5 degree of longitude between each of the 40 columns
bottomhgt = 0.0	Bottom of box is at 0km (sea level)
hgtinc = 1.0	1 km between each of the 20 grid levels (top at 19.0km)

The product of the number of rows, columns, levels, timesteps, and variables is the total number of data points. In this example: 30*40*20*5*4 = 480,000. A real dataset may be 100 rows by 100 columns by 20 levels, have 50 timesteps, and 10 variables for a total of 100,000,000 data points.

The difference between the foo_to_v5d program (which uses v5dCreateSimple), and the foo2_to_v5d program (which uses v5dCreate), is the later allows you to specify any map projection, vertical coordinate system, a different number of grid levels for each physical variable, and to control data compression. To specify a map projection, you must set the value of projection to 0,1,2 or 3 to indicate which projection, then specify the projection-dependent parameters in the proj_args array. Specifying the vertical coordinate system is done similarly.

It is sometimes useful to specify a different number of grid levels for each variable. For example, suppose most of your variables have 30 grid levels but a some variables have fewer grid levels, perhaps only one. Prior to version 4.0 of Vis5D, you would have had to fill in the extra levels with redundant, missing or dummy data values. With the v5dCreate function you can specify how many grid levels are present for each individual physical variable with the nl array parameter. Be aware that the amount of data passed to the v5dWrite call will depend on which variable you're writing. For example, if your grid has C columns and R rows then the number of values in the data array passed to v5dWrite for variable V must equal C*R*nl(V).

By default, the bottom-most grid level of each variable is displayed at the bottom of the 3-D box; each grid extends upward for how ever many levels are present. Sometimes, however, the bottom-most grid level of a particular variable should be positioned higher up. An example of this is a combined ocean/atmosphere dataset. There may be a total of 18 grid levels: the bottom 8 grid levels being ocean data and the top 10 grid levels being atmospheric data. In this case, the bottom of the atmospheric data should be offset or shifted upward by 8 grid levels.

Elaborating on the ocean/atmosphere example, suppose we have 2 ocean variables named S (salinity) and T (temperature) and 2 atmosphere variables named P (pressure) and T1 (temperature). There are 8 layers of ocean data and 10 layers of atmospheric data. Here is a summary showing how the lowlev array is the solution to this situation:

varnum	varname(varnum)	nl(varnum)	lowlev(varnum)
1	S	8	0
2	T	8	0
3	P	10	8
4	T1	10	8

The lowlev array is not specified in the v5dCreate function because it was developed after the v5dCreate function was well established. Instead, the new v5dSetLowLev function is called with the lowlev array. This separate function was added to extend the functionality of v5dCreate without changing its calling sequence. Here is the synopsis of v5dSetLowLev:

Fortran-callable function:

```
integer function v5dsetlowlev( lowlev )
integer lowlev(*)
```

C-callable function:

```
int v5dSetLowLev( lowlev )
int lowlev[];
```

Argument description:

lowlev Specifies the vertical offset, in grid levels, for each variable.

lowlev(1) = offset for first variable
lowlev(2) = offset for second variable
... = ...
lowlev(numvars) = offset for last variable

v5dSetLowLev may be called at any point between v5dCreate and v5dClose.

The v5dCreate and v5dcreate functions allow you to control how the grid data are compressed. The default is for grid values to be linearly scaled to one byte integers. This works very well for most data sets, since the scaling factors are chosen independently for each combination of timestep, variable and vertical level. Furthermore, the compression to one byte per grid point enables Vis5D's high degree of interactivity, since compression allows entire data sets to be resident in memory. However, the compress argument of the v5dCreate and v5dcreate functions lets you pick whether grid point values are scaled to 1-byte integers, scaled to 2-byte integers, or

left as 4-byte floating point values (no compression). We recommend that you try compression to 1-byte integers first, and only use 2 or 4 bytes if you have precision problems at 1-byte.

Vis5D version 4.2 and later allow you to specify the physical units for each variable in your dataset. The v5dSetUnits() function takes two arguments: a variable number and a units character string. If the first variable in your file is P and the units are millibars then you can specify that with:

```
C: v5dSetUnits( 1, "millibars" )
Fortran: call v5dsetunits( 1, "millibars" )
```

The units will be displayed by the v5dinfo program and in Vis5D when using the probe.

To compile your program which uses v5dCreate, v5dWrite, and v5dClose you must link with the src/v5d.o and src/binio.o files. See the makefiles in the convert/ directory for examples.

Finally, if your data is generated by an atmospheric or oceanic model, you may want to consider modifying your model to generate v5d files directly using the v5dCreate, v5dWrite, and v5dClose functions. Look at the sample data conversion programs for ideas.

3.2 Map Projections and Vertical Coordinate Systems

Version 4.0 of Vis5D added support for new map projections and vertical coordinate systems. When we use the term map projection, we're referring to the relationship between the rows and columns of data in the 3-D grid to the latitude/longitude of the earth. The term vertical coordinate system refers to the relationship between the vertical levels of data in the 3-D grid to altitude in the atmosphere (or depth in the ocean).

Vis5D 4.2 supports the following map projections:

(0) Generic rectilinear: this is a linear, regularly-spaced coordinate system with no implied units. This system is useful when your data is not related to earth science (computational fluid dynamics for example.) North/south coordinates increase upward and east/west coordinates increase to the left. The projection is defined by four parameters:

NorthBound	Northern boundary of 3-D box
WestBound	Western boundary of 3-D box
RowInc	Increment (spacing) between grid columns
ColInc	Increment (spacing) between grid rows

Example: Suppose your 3-D grid has 80 rows and 60 columns and NorthBound = 100.0 meters, WestBound = 50.0 meters, RowInc = 0.5 meters, and ColInc = 0.5 meters, then:

the south boundary will be at 60.5 meters. i.e. southbound = NorthBound - (RowInc * (rows-1))
and the east boundary will be at 79.5 meters. i.e. eastbound = WestBound + (ColInc * (columns-1))

(1) Rectilinear lat/long (cylindrical equidistant): this is the rectangular latitude/longitude coordinate system used in previous versions of Vis5D. Latitude increases to the North (upward in the graphical display) and longitude increases to the West (leftward in the graphical display; positive west latitude). The projection is defined by four parameters:

NorthBound	Northern boundary of 3-D box in degrees of latitude in the range [-90S,90N].
WestBound	Western boundary of 3-D box in degrees of longitude in the range [-180E,180W].
RowInc	Increment (spacing) between grid rows in degrees of latitude greater than zero.
ColInc	Increment (spacing) between grid columns in degrees of longitude greater than zero.

Example: If your 3-D grid has 30 rows and 60 columns and if NorthBound = 70.0, WestBound = 140.0, RowInc = 1.0, and ColInc = 0.5, then:

the south boundary will be at 41 degrees latitude. i.e. (NorthBound - RowInc * (rows-1))
and the east boundary will be at 110.5 degrees longitude. i.e. (WestBound - ColInc * (columns-1))

(2) Lambert conformal: a conic projection defined by the following six parameters:

Lat1, Lat2	First and second standard latitudes in the range [-90S,90N]. Lat1 and Lat2 define where the imaginary cone intersects the sphere of the Earth. Lat1 and Lat2 must have the same sign, that is, they must both be positive or both negative. Also, Lat1 must be greater than or equal to Lat2.
PoleRow, PoleCol	These parameters indicate the position of the north or south pole with respect to the 3-D grid coordinate system. These values may

be outside the 3-D grid. If Lat1 and Lat2 are positive, the north pole is assumed, else, the south pole is assumed.

CentLon — Central longitude: this parameter indicates which Earth longitude is to be parallel to the 3-D grid columns.

ColInc — Increment (spacing) between grid columns at the central longitude and standard latitudes, in km. This parameter controls the scale of the projection.

Example 1: Suppose your 3-D grid has 35 rows and 40 columns and you want a Lambert conformal projection of the United States centered over Wisconsin:

Lat1 = 70.0
Lat2 = 20.0
PoleRow = -35.0
PoleCol = 20.0
Central Longitude = 90.0
ColInc = 100.0

Example 2: Suppose your 3-D grid has 35 rows and 40 columns and you want a Lambert conformal projection over Australia:

Lat1 = -20.0
Lat2 = -70.0
PoleRow = 60.0
PoleCol = 20.0
Central Longitude = -130.0
ColInc = 200.0

Note: Beware that when the pole is visible in a Lambert conformal projection, there is usually a wedge-shaped region (with its apex at the pole) which is undefined (i.e. Longitude is >180 AND <-180). In this region, there will be no map lines and the topography will be incorrect.

(3) Azimuthal Stereographic: an azimuthal stereographic projection defined by five parameters:

CentLat, CentLon — Latitude and longitude of the center of projection. The apex of the imaginary cone will be over this coordinate.

CentRow, CentCol	Row and column of the center of projection. The grid row and column indicated will be at the center of the projection. These values may be outside the 3-D box.
ColInc	Increment (spacing) between grid columns in km at the center of the projection. This parameter controls the scale of the projection.

Example: Suppose your 3-D grid has 40 rows and 40 columns and want an azimuthal stereographic projection centered over of the north pole:

CentLat = 90.0
CentLon = 0.0
CentRow = 20.0
CentCol = 20.0
ColInc = 200.0

(4) Rotated rectilinear lat/long: this is the rectangular latitude/longitude coordinate system on a sphere rotated with respect to the Earth's natural latitude/longitude. North/south coordinates increase upward on the rotated sphere and east/west coordinates increase leftward on the rotated sphere. The projection is defined by seven parameters:

NorthBound	Northern boundary of 3-D box in degrees of latitude in the range [-90S,90N].
WestBound	Western boundary of 3-D box in degrees of longitude in the range [-180E,180W].
RowInc	Increment (spacing) between grid rows in degrees of latitude greater than zero.
ColInc	Increment (spacing) between grid columns in degrees of longitude greater than zero.
CentLat, CentLon	Latitude and longitude on Earth corresponding to Latitude/Longitude = (0,0) on the rotated sphere.
Rotation	Clockwise angle of rotation of rotated sphere about its (0,0) point.

Example: Over small regions the Earth is nearly flat and we can exploit this to create nearly square grids for small scale models. We can generate a nearly square grid of 41 rows by 41 columns over a small region over Wisconsin with:

NorthBound = 2.0
WestBound = 2.0
RowInc = 0.1
ColInc = 0.1
CentLat = 43.0

CentLon = 90.0
Rotation = 0.0

Vis5D 4.2 supports the following vertical coordinate systems:

(0) Equally spaced, generic units: this is a linear vertical coordinate system in which levels in the 3-D grid are equally spaced. No specific units are implied. The coordinate system is defined by two parameters:

BottomBound Bottom boundary of 3-D box.
LevInc Increment(spacing) between grid levels.

Example: Suppose your 3-D grid has 20 levels and you want the bottom boundary to be 0.0 meters and you want .1 meters between levels. Then:

BottomBound = 0.0
LevInc = 0.1

(1) Equally spaced, kilometers: this is a linear vertical coordinate system used in previous versions of Vis5D. Grid levels are equally spaced. The coordinate system is defined by two parameters:

BottomBound Bottom boundary of 3-D box in km.
LevInc Increment (spacing) between grid levels in km greater than zero.

Example: Suppose your 3-D grid has 20 levels and you want .5 kilometers between grid levels. Then:

BottomBound = 0.0
LevInc = 0.5

(2) Unequally spaced, kilometers: this is a linear vertical coordinate system in which grid levels can be unequally spaced. The coordinate system is defined by an array of N height parameters where N is the number of levels in the 3-D grids. If the number of grid levels is different for each variable, N is the maximum number of grid levels.

Height(1) Height of first (bottom) grid level in km
Height(2) Height of second grid level in km

Height(N) Height of Nth (top) grid level in km

Note that the Height values must increase with N.

Example: Suppose your 3-D grids have 10 levels and you want the grid levels to be more closely spaced near the bottom than near the top. Then:

Height(1) = 0.0
Height(2) = 0.1
Height(3) = 0.2
Height(4) = 0.3
Height(5) = 0.4
Height(6) = 0.6
Height(7) = 0.8
Height(8) = 1.0
Height(9) = 1.3
Height(10) = 1.6

It is also possible to display the vertical axis on a logarithmic scale. This is done with the -log command line option when you start Vis5D. In this case, the vertical axis is logarithmic with respect to height but linear with respect to pressure. The relationship between height (H) and pressure (P) is:

P = 1012.5 * e^(H / -7.2) (^ denotes exponentiation)

H = -7.2 * Ln(P / 1012.5) (Ln denotes natural log)

The constants 1012.5 and -7.2 are just defaults which can be overridden when you specify the -log option. See section 6.1 for details.

(3) Unequally spaced, millibars: this is a linear vertical coordinate system in which grid levels can be unequally spaced. The coordinate system is defined by an array of N pressure parameters where N is the number of levels in the 3-D grids. If the number of grid levels is different for each variable, N is the maximum number of grid levels.

Pressure(1)	Pressure of first (bottom) grid level in km
Pressure(2)	Pressure of second grid level in km
...	...
Pressure(N)	Pressure of Nth (top) grid level in km

Note that the Pressure values must decrease with N.

For the purposes of calculating wind trajectories, Vis5D assumes the relationship between height (H) and pressure (P) is:

P = 1012.5 * e^(H / -7.2) (^ denotes exponentiation)

H = -7.2 * Ln(P / 1012.5) (Ln denotes natural log)

Only the v5d file format is capable of storing the new map projection and vertical coordinate system information. When a v5d file is read into Vis5D, this information is used to setup the topography, map lines, and compute wind trajectories.

The Vis5D program also supports two other display projections: spherical and cylindrical. Instead of drawing a rectangular 3-D box, these projections will actually warp the 3-D box into a spherical or cylindrical shape. These projections are used by specifying the -projection option with the value spherical or cylindrical; they are not specified in the v5d file. The spherical option can be used to display your data on a 3-D globe. The cylindrical option can be used to display your data on a flat, round topography. It's probably best to just experiment with these options using the LAMPS dataset for example. See the section on Vis5D's command line options for more information.

3.3 Special Variables and Data Values

Analysis and visualization of wind information is an important part of Vis5D. Specifically, the Vis5D program looks to see if your data set contains variables named U, V and W. If present, they are assumed to be the three components of wind vectors and are used to display trajectory tracings and wind slices.

Positive U values are eastward, negative U is westward. Positive V values are northward, negative V is southward. Positive W values are upward, negative W is downward. The units for U, V and W are assumed to be meters per second except when a generic map projection or vertical coordinate system is used. In that case, the units are in X per second where X is the units used to specify the NorthBound, WestBound, RowInc, and colinc parameters.

If you do not like to use U, V, and W for wind vector components you can either specify other wind variable names on the vis5d command line or enter them while running vis5d.

Strictly speaking, U, V and W do not have to represent wind motion. They can be used to represent any flow field such as ocean currents. However, you may want to scale U, V, and W by some constant for visualization purposes.

Vis5D allows any grid data value to be undefined or 'missing'. For example, datasets based on observations are often incomplete or contain erroneous values. In your data conversion program you can indicate a grid value is missing by assigning it a value greater than 1.0e30. Missing data in vis5d will show up as holes in isosurfaces and contour slices and as black regions in colored slices. The data probe will report missing values as 'Missing'.

4.0 McIDAS 3-D GRID DATA FILES

In previous versions of Vis5D, it was standard practice to put one's data into a McIDAS GR3D file, then compress it with comp5d prior to using vis5d. While directly converting to the v5d format is preferred, we still include this information on the McIDAS format. If you don't want to put your data into McIDAS files, you may skip to section 5 now.

WE RECOMMEND AGAINST THIS WAY OF GETTING YOUR DATA INTO VIS5D - INSTEAD USE THE TECHNIQUES DESCRIBED IN SECTION 3 OF THIS DOCUMENT.

A McIDAS GR3D file contains a sequence of 3-D grids of data. The three-dimensional grids are organized into short sequences to enumerate the values of multiple physical variables at a single time. The short sequences of physical variables are repeated into a longer sequence which steps through many timesteps. These files have a names of the form GR3Dnnnn where nnnn is a 4-digit number between 0001 and 9999. The McIDAS utility programs then refer to files only by a number (1 through 9999).

A 3-D grid file contains a directory entry for each 3-D grid, which describes the size and geographic location of the grid, and the date, time and name of physical variable of the data in the grid array. A five-dimensional data set consists of a sequence of 3-D grids in a 3-D grid file, all with the same size and geographic locations. The grid sequence repeats the same short sequence of physical variables stepping forward through time. For example, the grid sequence from a weather model could be:

GRID NUMBER	DATE	TIME	PHYSICAL VARIABLE NAME
1	88035	000000	U
2	88035	000000	V
3	88035	000000	W
4	88035	000000	T
5	88035	000000	P
6	88035	010000	U
7	88035	010000	V
8	88035	010000	W
9	88035	010000	T

10	88035	010000	P
11	88035	020000	U
12	88035	020000	V
13	88035	020000	W
14	88035	020000	T
15	88035	020000	P

This data set consists of 3 timesteps of 5 physical variables. The physical variables are the U, V and W components of the wind vector, the temperature T and the pressure P. The date is February 4, 1988 and the timesteps are midnight, 1 AM and 2 AM. Dates are in YYDDD format and times are in HHMMSS format as described earlier.

4.1 Putting Your Data Into a McIDAS 3-D Grid File

The following sample program creates a 3-D grid file and fills its 3-D grids with data for a five-dimensional data set. This program can be found in the file sample.F, its makefile is sample.m. The easiest way to read your data into a 3-D grid file is to alter the sample.F program. The subroutines it calls are all in the libmain.a library, and their source is in the src subdirectory. Here is a listing of sample.F:

```
1   C THE MAIN PROGRAM OF YOUR CONVERSION PROGRAM MUST
2   C BE NAMED SUBROUTINE MAIN0
3   C
4   SUBROUTINE MAIN0
5   C
6   C THE NEXT TWO COMMENTS ARE PRINTED BY THE 'help sample'
    COMMAND
7   C ? SAMPLE program to convert data to 3-D grid files
8   C ? sample gridf#
9   C
10  C DIMENSIONS OF 3-D GRID
11  C NOTE NLATS AND NLONS MUST BOTH BE LESS THAN OR EQUAL TO
    150
12  C NLATS, NLONS AND NHGTS MUST ALL BE AT LEAST 2
13  PARAMETER (NLATS=31,NLONS=51,NHGTS=16)
14  C
15  C NUMBER OF PHYSICAL VARIABLES AND NUMBER OF TIMESTEPS
16  C NOTE EITHER OR BOTH MAY BE EQUAL TO 1. THAT IS, Vis5D DOES
17  C NOT FORCE YOU TO HAVE MULTIPLE VARIABLES OR TIME
    DYNAMICS.
18  PARAMETER (NVARS=5,NTIMES=100)
```

```
C
C ARRAY FOR 3-D GRID DATA
REAL*4 G(NLATS, NLONS, NHGTS)
C ARRAYS FOR GRID FILE ID AND GRID DIRECTORY
INTEGER ID(8), IDIR(64)
C ARRAY FOR VARIABLE NAMES
CHARACTER*4 CNAME(5)
C
C LATITUDE, LONGITUDE AND HEIGHT BOUNDS FOR SPATIAL GRID
DATA XLATS/20.0/,XLATN/50.0/
DATA XLONE/70.0/,XLONW/120.0/
DATA XHGTB/0.0/,XHGTT/15.0/
C
C STARTING DATE IN YYDDD AND TIME IN HHMMSS
DATA JDAY/88035/,JTIME/020000/
C TIMESTEP IN HHMMSS
DATA JSTEP/000100/
C
C NAMES OF THE FIVE PHYSICAL VARIABLES
DATA CNAME/'U   ','V   ','W   ','T   ','P   '/
C INITIALIZE GRID DIRECTORY TO ZEROS
DATA IDIR/64*0/
C
C READ GRID FILE NUMBER FROM COMMAND LINE.  IPP WILL
C CONVERT THE PARAMETER # 1 TO AN INTEGER, WITH A DEFAULT
C VALUE OF 0.
IGRIDF=IPP(1,0)
C IF ILLEGAL GRID FILE NUMBER, PRINT ERROR MESSAGE AND
RETURN
IF(IGRIDF .LT. 1 .OR. IGRIDF .GT. 9999) THEN
CALL EDEST('BAD GRID FILE NUMBER ',IGRIDF)
CALL EDEST('MUST BE BETWEEN 1 AND 9999 ',0)
RETURN
ENDIF
C
C CALCULATE GRID INTERVALS
XLATIN=(XLATN-XLATS)/(NLATS-1)
XLONIN=(XLONW-XLONE)/(NLONS-1)
XHGTIN=(XHGTT-XHGTB)/(NHGTS-1)
C
C DATE AND TIME FOR FIRST TIMESTEP
```

```
C IDAYS CONVERTS YYDDD FORMAT TO DAYS SINCE JAN. 1, 1900
IDAY=IDAYS(JDAY)
C ISECS CONVERTS HHMMSS FORMAT TO SECONDS SINCE MIDNIGHT
ISEC=ISECS(JTIME)
C
C INITIALIZE GRID IDENTIFIER TEXT TO BLANKS
C NOTE LIT CONVERTS A CHARACTER*4 TO AN INTEGER*4
DO 10 I=1,8
10   ID(I)=LIT('    ')
C
C SET UP DIRECTORY ENTRY
C
C DIMENSIONS OF GRID
IDIR(1)=NLATS*NLONS*NHGTS
IDIR(2)=NLATS
IDIR(3)=NLONS
IDIR(4)=NHGTS
C
C LATITUDES AND LONGITUDES IN DEGREES * 10000
IDIR(22)=4
IDIR(23)=NINT(XLATN*10000.)
IDIR(24)=NINT(XLONW*10000.)
IDIR(25)=NINT(XLATIN*10000.0)
IDIR(26)=NINT(XLONIN*10000.0)
C
C HEIGHTS IN METERS
IDIR(31)=1
IDIR(32)=NINT(XHGTT*1000.)
IDIR(33)=NINT(XHGTIN*1000.)
C
C CREATE THE GRID FILE
CALL IGMK3-D(IGRIDF, ID, NLATS*NLONS*NHGTS)
C
C LOOP FOR TIMESTEPS
DO 200 IT=1,NTIMES
C
C SET DATE AND TIME IN DIRECTORY ENTRY
C IYYDDD CONVERTS DAYS SINCE JAN. 1, 1900 TO OUR YYDDD
FORMAT
IDIR(6)=IYYDDD(IDAY)
```

```
98  C IHMS CONVERTS SECONDS SINCE MIDNIGHT TO OUR HHMMSS
    FORMAT
99  IDIR(7)=IHMS(ISEC)
100 C
101 C LOOP FOR PHYSICAL VARIABLES
102 DO 190 IV=1,NVARS
103 C
104 C SET VARIABLE NAME IN DIRECTORY ENTRY
105 IDIR(9)=LIT(CNAME(IV))
106 C
107 C ************************************************************
108 C READ YOUR DATA FOR TIMESTEP NUMBER IT AND VARIABLE
    NUMBER IV
109 C INTO  THE ARRAY G HERE.
110 C NOTE THAT G(1,1,1) IS THE NORTH WEST BOTTOM CORNER AND
111 C G(NLATS,NLONS,NHGTS) IS THE SOUTH EAST TOP CORNER.
112 C MARK A GRID POINT AS 'MISSING DATA' BY SETTING IT = 1.0E35
113 C ************************************************************
114 C
115 C CALCULATE 3-D GRID NUMBER
116 IGRID=IV+NVARS*(IT-1)
117 C WRITE DATA IN G AND DIRECTORY IN IDIR TO 3-D GRID
118 C NOTE WE PASS THE NEGATIVE OF THE GRID NUMBER (I.E. -IGRID)
119 CALL IGPT3-D(IGRIDF,-IGRID, G, NLATS, NLONS, NHGTS. IDIR, IGNO)
120 C
121 C END OF PHYSICAL VARIABLE LOOP
122 190  CONTINUE
123 C
124 C INCREMENT DATE AND TIME. CONVERT JSTEP FROM HHMMSS TO
    SECONDS
125 ISEC=ISEC+ISECS(JSTEP)
126 C IF SECONDS CARRY PAST ONE DAY. ADJUST SECONDS AND DAYS
127 IDAY=IDAY+ISEC/(24*3600)
128 ISEC=MOD(ISEC,24*3600)
129 C
130 C END OF TIMESTEP LOOP
131 200  CONTINUE
132 C
133 RETURN
134 END
```

The routines IGMK3D and IGPT3D are the interface to the 3-D grid structures. The call to IGMK3D at line 90 creates a 3-D grid file. Its parameters are:

1 INTEGER*4 - number of 3-D grid file to create
2 array of 8 INTEGER*4 - a 32 byte text ID for the file
3 INTEGER*4 - maximum number of grid points in any 3-D grid.

After the 3-D grid file is created, IGPT3D is called in line 119 once for each combination of timestep and physical variable to put 3-D grids into the file. Its parameters are:

1 INTEGER*4 - number of 3-D grid file to write to
2 INTEGER*4 - minus the number of the 3-D grid to write. This is 0 or positive to indicate write to next empty grid.
3 array of REAL*4 - array of grid points to write
4 INTEGER*4 - first dimension of grid array, # of latitudes
5 INTEGER*4 - second dimension of grid array, # of longitudes
6 INTEGER*4 - third dimension of grid array, # of heights
7 array of 64 INTEGER*4 - directory for 3-D grid
8 INTEGER*4 - number of 3-D grid actually written, returned by IGPT3D.

Vis5D allows data sets which span more than one 3-D grid file. In this case the grid sequence of repeating variables and repeating timesteps continues across grid file boundaries. A single 3-D grid file is limited to 100,000,000 grid points (400 megabytes). If your data set contains more than this number of grid points, then you should alter sample.F to create a new 3-D grid file (by incrementing IGRIDF and calling IGMK3D) on every Nth timestep, where N timesteps will fit in one 3-D grid file. Note that the comp5d command described in section 4 references data sets as sequences of 3-D grid files.

The Vis5D system processes the gridded data based on the information in the grid directories, which is contained in the IDIR array in the sample.F program. It is a good idea to initialize IDIR to all zeros, as in line 40. The size of the 3-D grid is set in entries 1 to 4 of IDIR (lines 72 to 75). Note the restrictions on data set size described in section 4 of this document.

The date and time of the 3-D grid are set in entries 6 and 7 of IDIR, as in lines 97 and 99. Note that they are represented in our YYDDD and HHMMSS formats described above. Four functions are available in libmain.a for converting between these formats and a format which makes date and time calculations easy. The IDAYS function converts YYDDD format to days since January 1, 1900, as in line 60. The ISECS function converts HHMMSS format to seconds since midnight, as in lines 62 and 125. This makes it easy to do calculations with dates and times, as in lines 125, 127 and 128. Then the IYYDDD function converts days back to YYDDD and the IHMS function converts back to HHMMSS, as in lines 97 and 99.

The physical variable name is 4 ASCII characters packed into entry 9 of IDIR, as in line 105. The LIT function in libmain.a converts a CHARACTER*4 to an INTEGER*4.

The spatial location of the grid is described in terms of latitude and longitude in ten-thousandths of a degree, and in terms of height (altitude) in meters. The grid element G(1,1,1) is in the north west bottom corner of the grid, and the grid element G(NLATS,NLONS,NHGTS) is in the south east top corner. The grid latitude and longitude are described in entries 21 to 25 of IDIR, as in lines 78 to 82. The grid heights are described in entries 31 to 33, as in lines 85 to 87. The NINT function is a FORTRAN intrinsic for converting a REAL to the nearest INTEGER. The latitude, longitude and height spacings are simply the distances between successive grid points. Latitudes are positive in the northern hemisphere, longitudes are positive in the western hemisphere, and of course heights are positive above sea level.

The real work in modifying the sample.F program is writing code for getting your data into the G array, in lines 107 to 113. For some data you may want to fake the latitude, longitude and height coordinates. However, if your data is geographical and large scale, then you may want to describe its location accurately, and it may be necessary to resample your data to a regularly spaced grid in latitude, longitude and height from some other map projection. It may also be necessary to transpose your data array to get the index order to be LAT, LONG and HGT, and to invert your data array in some index to make sure G(1,1,1) is the north west bottom corner. Even in faked coordinates, you may need to transpose or invert your data array to get the right 'handedness' in the display. The Vis5D system allows grid points marked as missing, indicated by array values greater than 1.0E30. If you do fake the latitude, longitude and height coordinates, then the topography and map display of the Vis5D program will be meaningless. If you calculate trajectories for your data set, either use accurate coordinates, or take great care to get relative time, distance and velocity scales consistent in the faked coordinates. Otherwise trajectory paths will not be realistic.

The IPP function in libmain.a returns the value of a command parameter as INTEGER*4, as in line 45. There are similar functions CPP and DPP in libmain.a which return CHARACTER*12 (converted to upper case) and REAL*8 values for command parameters. They get command parameters based on their sequential position in the command line. They all have similar function parameters:

1 INTEGER*4 - sequence number of command parameter
2 (IPP) INTEGER*4 - default value of command parameter
or
2 (CPP) CHARACTER*12 - default value of command parameter
or
2 (DPP) REAL*8 - default value of command parameter.

There is also a mechanism for picking up command parameters based on keywords. This is done with the functions IKWP, CKWP and DKWP in libmain.a. They get command parameters based on position after a keyword of the form '-keyword'. IKWP returns an INTEGER*4, CKWP returns a CHARACTER*12 (converted to upper case) and DKWP returns a REAL*8. They all have similar function parameters:

1 CHARACTER*12 - keyword string in command line
2 INTEGER*4 - sequence number of command parameter after keyword
3 (IKWP) INTEGER*4 - default value of command parameter
or
3 (CKWP) CHARACTER*12 - default value of command parameter
or
3 (DKWP) REAL*8 - default value of command parameter.

The NKWP function in libmain.a returns the number of sequential parameters after a keyword. Its function parameter is:

1 CHARACTER*12 - keyword string in command line.

On the most machines the REAL*4 format is not a subset of the REAL*8 format, so make sure to declare DPP and DKWP as REAL*8, as well as their third function parameters (for default values of command parameters).

If you would rather write your grid conversion program in C instead of FORTRAN, look at the file 'sample.c'. It contains examples of how to easily read and write grid files using C structures and routines in studio.

4.2 Using the McIDAS Utilities

Once your data set is in a 3-D grid file, you can list directory information about the grids using the command:

igg3d list I J -gr3df N

where N is the 3-D grid file number, and I and J give the range of grid numbers to list. You can get a quick idea of the data values using the command:

igg3d info I J -gr3df N

which will list the minimum and maximum values, the mean, the standard deviation and the number of grid points marked for missing data, for grid numbers I to J in 3-D grid file number N.

There are restrictions on the dimensions of data sets which can be visualized using the Vis5D program. Currently, you are limited to a maximum of 30 physical variables and 400 times steps. The Vis5D program will also fail if there is a trivial spatial dimension:

NLATS < 2
NLONS < 2
NHGTS < 2

The Vis5D program will perform badly, possibly making errors, if the total 5-D size:

NLATS * NLONS * NHGTS * NTIMES * NVARS

is too large. The limit depends on the amount of memory in your system. For a 64 MB system, the limit is around 25,000,000, with performance degrading as the data set size exceeds the limit.

Vis5D provides the gg3d and igg3d programs which can be used to reduce the resolution and scale of a data set to meet these limits. The gg3d program resamples a 3-D grid to new array dimensions and new extents in latitude, longitude and height, using the command:

gg3d samp N I M J
gg3d ave N I M J

where N and I are the numbers of the source 3-D grid file and grid, and M and J are the numbers of the destination 3-D grid file and grid. The 'samp' version calculates destination grid point values by linearly interpolating between source grid point values, and is appropriate for increasing resolution. The 'ave' version calculates destination grid points by averaging multiple source grid point values, and is appropriate for decreasing resolution. Without any keywords gg3d will do a straight copy operation. Invoke the gg3d command with the keyword:

-size NLATS NLONS NHGTS

to set the grid dimensions for the destination grid as different from the dimensions for the source grid. Invoke gg3d with the keywords:

-lat XLATS XLATN
-long XLONE XLONW
-hgt XHGTB XHGTT

to set extents (range bounds) for the latitude, longitude and height for the destination grid as different from the extents for the source grid. The -lat, - long and -hgt keywords take real arguments.

The igg3d program provides options for copying and deleting 3-D grids and for interpolating between 3-D grids in time. Sequences of 3-D grids are copied using the command:

igg3d get N I J M K

where N is the source 3-D grid file number, I and J are the range of source grid numbers, M is the destination grid file number, and K is the starting destination grid number. A single grid may be copied within a 3-D grid file using the command:

igg3d copy I J -gr3df N

where N is the 3-D grid file number, I is the number of the source grid and J is the number of the destination grid. A range of grids may be deleted with the command:

igg3d del I J -gr3df N

where N is the 3-D grid file number and grid numbers between I and J are to be deleted.

The igg3d command provides two different options for time interpolation. The first is:

igg3d ave K I J D T -gr3df N

where grid number K is produced by interpolating between grid numbers I and J, all in 3-D grid file number N. Grid number K will be assigned day D (in YYDDD format) and time T (in HHMMSS format). The relative weighting of grids I and J is calculated from this date and time, assuming linear time interpolation. If grid K is not between grids I and J in date and time, igg3d prints an error message. The igg3d command also provides a more complex time interpolation option:

igg3d int I D T -setdel S M -lag U V -gr3df N

This will put a grid in the next empty slot of 3-D grid file number N, assigned to day D (in YYDDD format) and time T (in HHMMSS format). This grid will be interpolated from a sequence of grids, all in file number N, at grid numbers I, I+S, I+2S, ... , I+(M-1)S. This sequence of grids should be ascending in date and time. igg3d will search the sequence and linearly interpolate between the two consecutive grids from the sequence which bracket day D and time T. Furthermore, the interpolation will be done in a coordinate system moving at constant velocity (U, V), where U and V are in meters per second, with V positive for motion from south to north and U positive for motion from west to east. The two bracketing grids from the sequence will be shifted in latitude and longitude to their positions at day D and time T, and the result interpolated between these two spatially shifted grids. Furthermore, if the grids in the

sequence are identified in their directory entries with variable name 'U ' or 'V ', then the corresponding component of the velocity (U, V) will be subtracted from the grid values.

The 'int' option of igg3d may seem complex, but it is just what you need if you want to write a script to re-interpolate a five-dimensional data set to a new sequence of timesteps. It is particularly useful if the source sequence does not have uniform timesteps, or if the physics are moving through the spatial grid and you want to avoid blurring in the time re-interpolation. You would set M equal to the number of timesteps and S equal to the number of physical variables in the source five-dimensional data set. The I parameter would be set equal to the grid number in the first timestep of the variable being interpolated. Note that this igg3d option will put the new grid at the end of the grid file containing the source data set, but you can use 'igg3d get' to move it to another grid.

You can use the command:

igu3d make N M

to create 3-D grid file number N, which allows 3-D grids of up to M points each. The names of 3-D grid files have the form:

GR3Dnnnn

where nnnn is the four digit decimal grid file number, padded with leading zeros if needed to make four digits.

5. Vis5D UTILITIES

Vis5D includes a number of utility programs. This section describes each one. The new v5dimport program is described separately in Section 7.

v5dinfo

Usage: v5dinfo file

Description: v5dinfo prints information about the given v5d file such as the size of the 3-D grid, the number of timesteps, the names of the variables, etc.

This program will also work on comp5d files. Therefore, the old compinfo program has been removed.

v5dstats

Usage: v5dstats file

Description: v5dstats prints simple statistical information about the grid data in the named v5d file. Again, comp5d files are also accepted.

v5dedit

Usage: v5dedit file.v5d

Description: v5dedit allows you to change header information such as the map projection, vertical coordinate system and variables names in the named file. It is an interactive, menu-driven program and is intended to be self explanatory. This program does NOT work with comp5d files.

v5dappend

Usage: v5dappend [-var] [...] file.v5d [...] target.v5d

Description: v5dappend allows you to append a number of v5d files together to make one larger file. This might be useful if your weather model generates a separate .v5d file for each timestep because you'll want to join those files together to view the data in Vis5D.

The arguments are, in order:

- An optional list of variables to omit from the output file. For example, if you want to omit the variables U and THETA you would use the arguments -U and -THETA.
- The list of v5d files to append onto the target file.
- The name of the target v5d file to create (if it doesn't exit) or append onto (if the target file already exists).

Note that the dimensions of the grids (rows, columns and levels) must be the same in each file to append them together. The map projection and vertical coordinate system information will be taken from the first input file and ignored the remaining files.

gr3d_to_v5d

Usage: gr3d_to_v5d N M file.v5d C

Description: gr3d_to_v5d converts (a) McIDAS GR3D file(s) to a v5d file. N is a number which indicates the name of the first grid file, M is the number of grid files to convert, file.v5d is the name of the file to produce, and C is 1, 2 or 4 to indicate how many bytes per grid point to use for compression (the default is 1). Example: if N=20 and M=4 then the files GR3D0020, GR3D0021, GR3D0022, and GR3D0023 will be read an converted to the named file.v5d.

igg3d

Usage: igg3d ...

Description: igg3d is used to perform a variety of manipulations on McIDAS GR3D files. See section 4.2 for more details.

igu3d

Usage: igu3d ...

Description: igu3d is a utility to perform a variety of manipulations on McIDAS GR3D files. See section 4.2 for more details.

gg3d

Usage: gg3d ...

Description: gg3d is a utility for resampling McIDAS GR3D files. See section 4.2 for more details.

listfonts

Usage: listfonts

Description: listfonts, used on SGI systems only, lists the IRIS GL fonts available for use in Vis5D's 3-D window. After listing the fonts you may use one in Vis5D by specifying it with the -font option. For non-SGI systems or systems using OpenGL, use the xlsfonts or xfontsel program to select a font.

comp5d

Usage: comp5d N M filename

Description: comp5d converts one or more McIDAS GR3D files to the comp5d format used in previous (and the current) versions of Vis5D.

N is the first 3-D grid file number and M is the number of grid files in the data set. The M parameter allows data sets which span multiple grid files and should not be confused with the total number of 3-D grids in the data set.

filename is the name of the compressed grid file. You can choose whatever name you want, but note that comp5d will convert the name to all upper case characters.

If your data set contains wind vector components you can use the -wind keyword to select a subset of wind components or calculate horizontal wind speed, named 'SPD ', for the compressed file. The longitude, latitude, and vertical components of the wind vector must be named 'U ', 'V ' and 'W ' respectively. If you use the -wind keyword, then only those wind-relevant variables (i.e. U, V, W & SPD) whose names are listed after -wind will be included in the compressed file. For example, to include SPD and W in the compressed file, from a 3-D grid file containing U, V and W components, use the command:

comp5d N M F -wind SPD W

help

Usage: help utilityname

Description: The help command will list a quick reference to the parameter formats for the named utility such as igg3d, igu3d, gg3d, and comp5d utilities. Example: help igg3d

maketopo.c

This program, found in the util directory, is a template program for generating your own new topography (*.TOPO) files. Read the information at the top of the file for instructions. To compile maketopo see the makefile named maketopo.m.

makemap.c

This program, found in the util directory, is a template program for generating your own new McIDAS map outline (OUTL*) files. Read the information at the top of the file for instructions. To compile makemap see the makefile named makemap.m. If you create a map with lots too many line segments, it will be displayed with some line segments

missing and some extra crazy line segments. You can fix this by increasing MAXMAPVERT and MAXMAPSEG in src/globals.h, then re-making vis5d.

newmap.c

This program and mapfunc.f, found in the util directory, is used to transform the vertices of an existing map outline file to make a new map outline file. This might be useful if you need to transform a map to a new coordinate system. Read the newmap.c and newmap.m files for more information.

6. USING Vis5D TO VISUALIZE YOUR DATA

This section describes how to use the Vis5D visualization program, Vis5D. It is almost completely controlled using the mouse with a graphical user interface. The best way to learn to use it is to experiment. There is no way to harm your data from within the program.

6.1 Starting Vis5D

After you have made a v5d file, you can interactively visualize it with the command:

vis5d file.v5d [options]

[options] may be any combination of the following (though none are usually needed):

-alpha
: Use alpha blending instead of "screen door" transparency.

-area N
: [SGI only] Specifies the first of a sequence of McIDAS area files to read and then display inside the 3-D box. See section 6.14 for more information.

-box x y z
: This lets you specify the aspect ratio or proportions of the 3-D box. Default values are 2 2 1.

-barbs
: Use wind barbs in place of wind vectors.

-date
: Use 'dd month yy' in place of 'yyddd' on the clock.

-font xfontname

-font glfontname height
: Set the font used for the clock and text labels in the 3-D window. You can determine which form of the font option is used on your system by typing 'vis5d' alone and examining the options. The first form expects the name of an X

window system font. Use the xlsfonts command to see a list of X fonts on your system. The second form expects the name and size (72=1 inch) of an IRIS GL font. Use the listfonts command included with Vis5D to see a list of GL fonts on your system.

Example 1: vis5d LAMPS.v5d -font fg-30

Example 2: vis5d LAMPS.v5d -font Helvetica 30

-full

Open the 3-D window as a borderless, full-screen size window.

-funcpath pathname

Specify the directory to search for user Fortran functions.

Example vis5d LAMPS.v5d -funcpath /usr/local/vis5d/userfuncs

-geometry WxH+X+Y (or WxH or +X+Y)

Specify the geometry of the 3-D window.

Example vis5d LAMPS.v5d -geometry 640x480-10+10

-hirestopo

Display a high-resolution topography. This is only recommended on systems with fast graphics hardware.

-legend position size

Set color legend position and size. Position values are 1 (bottom, the default), 2 (top), 3 (left) and 4 (right). Size is the height of the legend bar and is between 10 and 1000 (default=128).

-log [a] [b]

Display height on a logarithmic axis instead of linear. This is discussed in section 3.2. The optional arguments a and b are the scale and exponent factors in the height/pressure equation. The defaults are 1012.5 and -7.2, respectively.

-map file

Use a map file other than the default of OUTLSUPW. See section 2.3 to setup a different default.

Example: vis5d LAMPS.v5d -map OUTLUSAL

-mbs n

Override the assumed system memory size of 32 megabytes. See section 2.3 to setup a different default value.

-path pathname

Use a different path for map and topo files instead of the current.

Example: vis5d LAMPS.v5d -path /usr3/data

-projection p

Set the display map projection, default is to display data in its natural projection (obtained from the data file). p may be one of:

cylindrical-display data on a cylindrical Earth

spherical-display data on a spherical Earth

Only the first 3 characters are significant/needed. You will be prompted for additional parameters.

Example: vis5d LAMPS.v5d-projection spherical

-quickstart

Don't load any grids when starting Vis5D, even if the whole file will fit into memory. The grids will be read as needed. This option is useful when reading a file via NFS.

-rate ms

Change the default animation rate. ms is the minimum delay in milliseconds between frames. Default is 100 ms.

-script script.tcl

Specifies a Vis5D/Tcl script to execute automatically.

-sequence filename

[not available on all systems] Specifies a file containing a sequence of images to texture map over the topography. See section 6.14 for more information.

-texture rgbfile

[not available on all systems] Specify an SGI .rgb file to texture map over the topography. See section 6.14 for more information.

-topo file

Use a topography file other than the default of EARTH.TOPO. See section 2.3 to setup a different default.

-trajvars uvar vvar [wvar]

Specify which variables are to be used for trajectory tracing.

Defaults are U, V, and W.

Example: vis5d LAMPS.v5d -trajvars U2 V2 W2

-vertical v

Set the vertical coordinate system, default is obtained from datafile. v may be one of:

generic - linear, equally spaced levels in generic units
equal - linear, equally spaced levels in km
nonequal - linear, unequally spaced levels in km

Only the first 3 characters of v are significant/needed. You will be prompted for additional parameters.

Example: vis5d LAMPS.v5d -vertical nonequal

-wdpy xdisplay

Put the widgets on a different X display. Useful in combination with -full for making slides and videos.

Example: vis5d LAMPS.v5d -full -wdpy pluto:0

-wide w

Set width of line segments in pixels (default is 1.0). Again, useful for making videos.

Example: vis5d LAMPS.v5d -wide 3.0

-wind2 uvar vvar [wvar]

Specify the names of a secondary set of U, V, and (optionally) W wind component variables to use when drawing the Hwind2, Vwind2 and Strm2 vector slices. Useful when you have two sets of wind vector components that you want to visualize simultaneously.

Example: vis5d MYDATA -wind2 U2 V2 W2

If you start Vis5D without arguments you will get a list of all the command line options and keyboard functions. Otherwise, Vis5D will begin by reading the data file.

Previous versions of Vis5D required that the entire file be read into main memory; if you didn't have sufficient memory you couldn't visualize the file. In version 4.0 and higher, this restriction is lifted; you may visualize files which are larger than main memory. This is implemented with a grid cache: Vis5D reads data only when needed and discards it on a least-recently-used basis. Small files will be read in their entirety as in previous versions.

For the user, this means Vis5D will allow you to visualize large files even with only 32 MB of main memory. However, performance will degrade as the ratio of file size to main memory size increases. If you observe sluggish performance and a lot of disk activity while running Vis5D you should get more memory.

6.2 The Control Panel

After Vis5D has opened/read your file, two windows will appear: a 3-D window on the right and a control panel on the left of the screen. The 3-D window is used to view and interact with the data. In its upper-left corner is a combination analog/digital clock which indicates the current timestep. The control panel contains several groups of buttons.

Starting at the top, the first button group contains the following buttons:

[ANIMATE]	[STEP]	NEW VAR	EXIT
[TEXTURE]	TOP	SOUTH	WEST
[TOPO]	[MAP]	BOX	CLOCK
SAVE	RESTORE	GRID #'s	CONT #'s
[ANIM-REC]	REVERSE	[SAVE PIC]	[PERSPEC]
SCRIPT	INTERP	UVW VARS	

These buttons are used to control the primary functions of Vis5D. Some of the above buttons are enclosed in brackets [] to indicates that they may be blank upon starting Vis5D. This will

happen when the button does not apply to the current data set, because the button would conflict with a command line option, or because the feature is not available on your hardware.

The next group of radio buttons control the viewing mode which determines how the mouse is used in the 3-D window:

Normal	Normal mouse mode is used to rotate, zoom, and pan the graphics in the 3-D window. See section 6.3.
Trajectory	This mode is used for creating and displaying wind trajectories. See section 6.7.
Slice	This mode is used to reposition horizontal and vertical slices. See section 6.5.
Label	This mode is used to create and edit text labels in the 3-D window. See section 6.8.
Probe	Used to inspect individual grid values by moving a 3-D cursor through the 3-D grid. See section 6.9.

These modes are mutually exclusive; only one may be selected at a time. To the immediate right of these buttons is the mouse button legend. It is there to remind you of the use of each mouse button in the 3-D window for the currently selected mode.

Next are buttons labeled:

Hwind1 Vwind1 HStrm Hwind2 Vwind2 VStrm

A wind vector slice (Hwind or Vwind) depicts wind values by drawing small arrows which point in the direction of the wind. The length of each line segment indicates its magnitude. The tails of the line segments are all anchored within a horizontal or vertical plane through the 3-D box. The horizontal wind streamline slice (HStrm) depicts wind values by drawing streamlines on a horizontal plane. The vertical wind streamline slice (VStrm) depicts wind values by drawing streamlines on a vertical plane. The location of slice planes can be changed with the mouse while in "Slice" mode. See section 6.5 for more details.

The bottom part of the control panel window contains a 2-D matrix of buttons. Each row corresponds to a physical variable in your dataset. Each column corresponds to one type of graphical representation. By selecting the correct row and column you can view any variable as a 3-D isosurface, horizontal contour slice, vertical contour slice, horizontal colored slice, vertical

colored slice, or volume rendering. This matrix of button is scrollable if there are more rows of buttons than will fit in the window. You can use the mouse to drag the scrollbar or press the up/down arrow keys on your keyboard to scroll the button matrix.

The display of any graphic is controlled by clicking on its widget button with the left mouse button. Each type of graphic also has a small pop-up control window which appears when turned on. The control windows are different for each type of graphic and are explained below. To bring up a graphic's control window without toggling its display, use the middle mouse button. When the graphic is displayed it will be the same color as the widget button, making it easy to distinguish and identify different variables in the display. To change the color of the graphic, click on its widget button with the right mouse button and a small window with four slider widgets will appear. By changing the levels of red, green, and blue you can make any color.

If the control panel window becomes obscured by other windows, you can bring it to the top by pressing the **<F1>** key while the mouse pointer is in the 3-D window. This is especially useful when using the '-full' option.

6.3 Controlling Vis5D

The topmost group of buttons in the control panel operate the main functions of Vis5D. Some will be discussed in more detail later.

ANIMATE — This toggle button turns animation on or off. Use the left or middle mouse buttons for forward animation and the right mouse button for reverse animation. Does not appear when viewing data sets with one timestep. To make the animation slower or faster, hit the S and F key on the keyboard while the mouse cursor is inside the 3-D viewing window.

STEP — This button has three possible uses depending on which mouse button is pressed:

- Left Button - Step ahead one timestep
- Middle Button - Go to first timestep.
- Right Button - Backward one timestep.

This button does not appear when viewing data sets with one timestep.

NEW VAR — Used to duplicate physical variables or invoke external analysis functions. This is explained further in section 6.11

EXIT — Exit the program. A window will appear to ask you to verify your decision.

TEXTURE Toggles display of texture maps on/off if they are loaded. See section 6.14 for more information.

TOP Depending on which mouse button is pressed:
Left or Middle: Reset the 3-D window to the default top-view.
Right: Set the 3-D window to a bottom-view.

SOUTH Depending on which mouse button is pressed:
Left or Middle: Set 3-D window to a south-view.
Right: Set 3-D window to a north-view.

WEST Depending on which mouse button is pressed:
Left or Middle: Set 3-D window to a west-view.
Right: Set 3-D window to an east-view.

TOPO Toggle the display of topography. This button will not appear if the topography file was not found. Click on TOPO with the right mouse button to edit the topography color.

MAP Toggle the display of map lines. This button will not appear if the map file was not found. Click on MAP with the right mouse button to edit the color of the map lines.

BOX Toggle the display of the 3-D box.

CLOCK Toggle the display of the clock.

SAVE Save current graphics and colors. After you've setup a variety of isosurfaces, slice, wind trajectories and colors it is useful to be able to save them and restore them the next time the data set is visualized. You'll be prompted for a filename. The file format, as of Vis5D 4.2 is a Tcl script. See section 6.14 for more information.

RESTORE Restore the information save with the SAVE button. See section 6.14 for more information.

GRID #s Normally the bounds of the data set in latitude, longitude and kilometers are displayed along the edges of the box. Use this button to display the numbers in grid coordinates instead.

CONT #s The numbers which are drawn on contour line slices can be toggled on or off with this button.

[ANIM-REC] This button works just like ANIMATE but allows fast animations on system with slow 3-D rendering. After each timestep is rendered the image is saved in memory. When the animation loop repeats the images are quickly copied from memory to the 3-D window resulting in a faster animation.

REVERSE Normally, the 3-D box and clock are drawn in white on a black background. This option reverses that and draws a black box and clock on a white background. This is useful for making paper print outs.

SAVE PIC Used to save the image in the 3-D window to a file. Depending on what system you're using a number of different picture file formats are supported. On SGI systems be sure you have the 'tops', 'frombin', and 'togif' program installed from your IRIX CD-ROM. When using OpenGL on SGIs the 'fromxwd' program is also needed. Unfortunately there is a bug in this program which often causes it to fail. Included with Vis5D however is a patched version of fromxwd.

PERSPEC Toggle between perspective and orthogonal viewing projections.

SCRIPT Used to run Vis5D Tcl scripts. When you click on this button a file request will appear in which you can select the Tcl script to run. For more information see section 6.14.

INTERP Starts the Vis5D interactive interpreter. In your shell window you may then enter Tcl commands. Vis5D will be suspended while the interpreter is active. Type 'exit' to exit the interpreter. For more information see section 6.14.

UVW VARS Opens a window in which you can specify the names of the variables to use for computing trajectories and wind slices.

LEGENDS Toggles the display of colorbar legends in the 3-D window.

6.4 Viewing Modes

In 'Normal' mouse mode the mouse is used to view the data in the 3-D window. By pressing the left mouse button and moving the mouse while the cursor is in the 3-D window, the 3-D image

can be rotated. At any instant you can only control two of the three degrees of freedom of box rotations. However, by releasing and re-pressing the left mouse button you can change your "grip" on the box. With practice you will learn to control the box through a series of mouse moves, releasing and re-pressing the left button between moves.

The center button controls two very different things depending on how the mouse is moved. Holding the center button down and sliding the mouse away from yourself zooms in, making the box get bigger. Sliding the mouse towards yourself zooms out and makes the box get smaller. Holding the center button down and sliding the mouse right moves a plane of invisibility (i.e. a clipping plane) into the box, creating a cut away view of the box contents. Sliding the mouse left brings the clipping plane toward yourself, eventually out of the box altogether.

The right mouse button is pressed to translate the box in the window. This is useful if you want to zoom in to something that is not in the center of the box. Note that the center of rotation for box rotations stays at the center of the screen rather than in the center of the box.

The other four viewing modes will be discussed in detail in following sections.

6.5 Isosurfaces

An isosurface (3-D contour surface) shows the 3-D volume bounded by a particular isovalue. The isosurface has the specified isolevel, the volume inside contains values greater (or less) than the isovalue. The volume outside contains values less (or greater) than the isovalue.

The first column of buttons in the control panel's button matrix controls isosurfaces. Clicking on one of these buttons with the left mouse button causes a pop-up window with a slider and OK button to appear below. Select an isovalue on the slider and click on the OK button to generate an isosurface for all timesteps.

Toggling ANIMATE on will let you watch the time dynamics of the isolevel contour surfaces. Note that the surfaces are generated asynchronously with the animation, so you may not see the surfaces for all the timesteps as the clock hand makes it revolution. The new surfaces will appear on successive clock revolutions.

Clicking on an isosurface button with the middle mouse button will summon the pop-up window without toggling the surface on or off.

6.5.1 Isosurface Color

An isosurface may either be drawn entirely in one color or colored according to the values of another physical variable.

To change the color of an isosurface, click on the appropriate isosurface button with the right mouse button. A window will appear with a column of variable names (first button labeled "monocolor") and four sliders labeled red, green, blue, and transparency.

By default, monocoloring is used. To change the isosurfaces's color just move the red, green, and blue sliders.

If you click on a button other than "monocolor" you will tell Vis5D to draw the isosurface according to another physical variable. The red, green, blue sliders will be replaced with a color table editor. You can change the color table (which maps data values to colors) by drawing new curves with the mouse or by pressing the up, down, left, and right cursor keys on your keyboard.

As an example, suppose you're viewing the LAMPS.v5d data set. Make an isosurface of wind speed at 40 m/s. The isosurface should be blue. Click on the SPD isosurface button with your right mouse button. The color window appears. Click on the T button in that window and the isosurface will now be colored according to temperature. You can modify the mapping from temperature values to colors by "drawing" the red, green, and blue curves in the color table window with the mouse buttons or by pressing the cursor keys. Changing the color table is explained more below in the section about colored slices.

6.6 Slices

Slices allow you to look at planar cross sections of data in the 3-D box. These slices can be oriented either horizontally or vertically and may depict either contour lines, colored slices, wind vectors, or wind stream lines.

As described in section 6.1, the last group of buttons on the control panel is a matrix of buttons, the second through fifth columns of which control slices. There is a column of buttons for horizontal contour slices, vertical contour slices, horizontal colored slices and vertical colored slices, respectively. If your data set contains U, V, and W variables, there will also be a row of wind vector slice buttons as described in 6.2. There are two buttons for horizontal wind slices and two buttons for vertical wind slices.

To activate/turn on a slice, click on the appropriate widget button with the left mouse button. The initial position for slices is the middle of the box. The exact slice location in terms of latitude, longitude or elevation is given by a small numeric labels near the one corner of each slice. To print the numbers as grid coordinates instead of geographic coordinates, toggle the "GRID #s" widget button on the control panel.

The position of slices can be changed interactively using the mouse. To do so you must first be in SLICE mode by selecting the SLICE radio button. To move any slice, simply point at the

slice's corner with the mouse, press the right mouse button and drag it to a new position. Vertical slices may also be moved in a perpendicular motion by "grabbing" the middle of the top or bottom edge and dragging it. A slice may be moved while in animation mode, however, some jumpiness may occur because new slices are computed asynchronously.

6.6.1 Contour Line Slices

When viewing a horizontal or vertical contour line slice (button columns two and three) a small control window will appear as well. In this pop-up window you can enter the interval to use between contour lines. Just type in a new number to change the interval. Decreasing the interval will cause denser contour lines to be generated, increasing the interval will result in sparser lines.

If you enter a negative interval then all contour lines with a negative value will be drawn with dashed lines while positive values will be drawn with solid lines.

Optionally, after the interval value you may specify a range of values (a,b) which will cause only contour values between a and b to be drawn. For example, suppose you enter

-10 (-30,20)

This will result in contour lines for values between -30 and 20 at intervals of 10 with negative lines drawn as dashed lines.

The "CONT #s" button on the control panel toggles the display of the contour numbers within the slice.

6.6.2 Colored Slices

When a viewing a horizontal or vertical colored slice (button columns four and five) a color table window will appear. In this pop-up window you can change the mapping from data values to colors. If the LEGENDS control panel button is selected the color table will also be displayed in the 3-D viewing window.

The window shows graphs of red, green, and blue over the range of data values. To change the red, green, or blue function press the left, middle, or right mouse button, respectively, and drag the mouse to draw a new function. By default, low data values are mapped to blue and high data values are mapped to red.

Instead of using the mouse you can use the keyboard cursor (arrow) keys to modify the shape and position of the default function curves. Press the left/right keys to move the curves left or right. Press the up/down keys to change the shape of the curves.

You may also change the transparency of the slice as a function of the data values. Press and hold the SHIFT key while using the mouse or up/down keys to change the transparency.

There are a number of other keyboard controls for the color table window:

r	reset red, green and blue values
R	reset transparency values
c	copy color to an off-screen clipboard
p	paste colors from the off-screen clipboard
s	save color values to a file, enter filename in your shell window
l	load color values from a file, enter filename in your shell window

6.6.3 Wind Vector Slices

Wind vector slices are displayed with the buttons near the center of the control panel labeled HWIND-1, VWIND-1, HWIND-2 and VWIND-2. The pop-up window for these graphics contains two type-in fields to control the density and scaling of the wind vectors. The scale parameter is used to multiply the length of vectors drawn. If you want to double the length of all vectors, enter 2.0. If you want to halve the lengths, enter 0.5. The density parameter controls how many wind vectors are displayed. This value can only be between zero and one. To make one-half the number of vectors, enter 0.5, for one-fourth enter 0.25, etc. The default values for both parameters is 1.0.

6.6.4 Wind Stream Slices

Wind stream slices show the path of wind as connected line segments. The pop-up control window contains a type-in widget to control the density of streamlines (note that the scale parameter is not used). The density parameter controls how many streamlines are displayed. This value can only be between 0.5 and 2.0. To make one-half the number of streamlines, enter 0.5, to make twice the number of streamlines, enter 2.0, etc. The default density is 1.0.

6.6.5 Slice colors

The color of a slice's control button matches that of the slice itself (except for colored slices for which the slice's tick mark matches the slice's button.) To change the color of a slice click on the slice's button with the right mouse button. A window with red, green, and blue sliders will appear. Move the sliders to change the color.

6.7 Volume Rendering

Volume rendering is a technique for displaying a 3-dimensional field as a semi-transparent colored fog. Though volume renderings of some physical variables don't look, others can be displayed very effectively with the right color mapping.

The volume rendering feature is available in Vis5D on almost all systems. One exception is older SGI computers using IRIS GL which don't support alpha blending. Be warned that systems without 3-D graphics hardware (i.e. those using Mesa) will render volumes very slowly.

The sixth column of buttons on the control panel are the volume buttons. Only one may be displayed at a time. When a volume rendering is activated a pop-up window with a color table appears. This color table is used in exactly the same way as described for colored slices above. That is, using the mouse or keyboard you can change the function which maps data values to color and transparency. Again, the transparency can be changed while holding down the SHIFT key and drawing a curve with the mouse or pressing the up/down keys.

For those who are curious about the implementation of this feature, the volume rendering is made as follows:

1. Examine the current viewing transformation to determine which axis of the 3-D box is most nearly parallel to the view direction.
2. Create a number of colored slices perpendicular to that axis which map data values to colors and opacity.
3. Render the colored slices in back to front order. The alpha values at vertices are interpolated and blended to make smooth transitions between and within slices.

Despite the simplicity of the algorithm, most fields are rendered acceptably. Those that aren't can be improved by adjusting the color and opacity mappings. While more attractive volume rendering techniques are known, this technique can be implemented quickly on many systems.

6.8 Wind Trajectories

Wind trajectories trace the motion of air through the 3-D volume much line smoke trails in a wind tunnel. To enter trajectory mode select the TRAJECTORY radio button on the control panel. A pop-up window will appear near the bottom of the screen and a 3-D cursor will appear inside the 3-D view box. This 3-D cursor is used to specify where a new wind trajectory should

be made. The STEP button on the main control panel is also important because it is used to select the timestep at which to create the trajectory.

Wind trajectories are dealt with in sets. Currently, eight sets are available. Each set is represented in the trajectory window with a button labeled Set1, Set2, ..., Set8. Each set can be individually displayed, colored, or deleted. As you create new trajectories you may want to group them in sets corresponding to location, time, etc.

The first step in creating a trajectory is to select a position with the 3-D cursor. Use the right mouse button to drag the 3-D cursor around inside the 3-D box. The 3-D cursor will move in 2-D in a plane parallel to the plane of projection. That is, the cursor will stay at a constant distance of depth. By alternately rotating the view box with the left mouse button and placing the cursor with the right mouse button, the 3-D cursor can be placed anywhere inside the view box. The TOP, SOUTH, and WEST buttons as explained in section 6.2 can also be useful when making trajectories.

Second you should select a timestep with the STEP button on the control panel. When the trajectory is made, it will be traced forward from the current timestep to the last timestep and will be traced backward through time to the first timestep.

Finally, to make a trajectory at the current cursor location and current timestep, press the middle mouse button when pointing inside the 3-D window. The trajectory will appear as a line segment. By turning on the ANIMATE button, you can observe how the trajectory travels through time and space. Typically, you will repeat the process of positioning the 3-D cursor and clicking the middle mouse button to create a set of trajectories.

Interesting results can be seen by making a trajectory when the ANIMATE button is turned on: a trajectory will be created for every timestep instead of just one. This will show you the path of every air parcel which passes through a single point in space.

Here is a summary of the various trajectory functions:

1. To position the 3-D cursor, use a combination of rotating the view box with the left mouse button and dragging the 3-D cursor with the right mouse button.

2. Use the STEP button or ANIMATE option to select a timestep.

3. Press the middle mouse button to create a trajectory at the current cursor location and timestep.

4. To toggle the display of a trajectory set on or off, click on the set button with the left mouse button.

5. Select the current trajectory set by clicking on the set button with the middle mouse button.

6. A trajectory set may be deleted with the 'Delete Set' button in the trajectories window. You will asked to verify your decision.

7. You can delete the last trajectory made by clicking on the 'Delete Last' button in the trajectories window.

Wind trajectories can be depicted in two ways: as line segments or as ribbons. You can select ribbons by clicking on the RIBBON button in the trajectory window. Toggling the RIBBON button will not effect trajectories you have already made; it only controls how new trajectories will be displayed.

The trajectory window also contains two type-in widgets labeled STEP and LENGTH. The STEP value is used to control the step size used in the trajectory tracing algorithm. The LENGTH value is used to control the length of trajectories. 1.0 is the default value for each. Each acts as a multiplier. If you want the trajectory tracer to integrate in steps 1/2 the default size, enter a step value of 0.5. If you want trajectories to be twice as long as the default length, enter a length value of 2.0.

The color of trajectories is controlled in the same way as for isosurfaces. That is, a trajectory set may either be mono-colored or colored according to another physical variable. Click on the trajectory set button with the right mouse button to bring up its color window. See section 6.5.1 for details on using the color window.

When viewing color-mapped trajectories be aware that the color of a trajectory is time dependent. Only the head of the trajectory is colored according to the value of another variable for the current timestep. The tail of the trajectory is colored according to the color of the other variable when the head was at that location.

6.9 Wind Variables

By default, wind trajectories and the first set of wind slices are computed from the variables named U, V, and W while the second set of wind slices are computed from the variables named U2, V2, and W2. Other variables can be specified through the "UVW VARS" button on the control panel. When you click on this button a pop-up window appears in which you can specify the names of the variables to use for computing trajectories, the first set of wind slices, and the

second set of wind slices. Just type in the new variables names. Be aware that uppercase and lowercase are significant. Be sure you enter valid names otherwise Vis5D may not compute the graphic you select.

After you've entered new wind component variable names click on APPLY to use the new values but keep the window visible. Click on OK to use the new values and close the window. Click on CANCEL to discard your changes and close the window.

You can also specify the wind component variables on the command line when you start Vis5D. See section 6.1.

6.10 Text Labels

Text labels are used to annotate the image in the 3-D viewing window. Typically this is used for making presentation graphics. You could add a title, your name, the date, highlight a particular feature of the data, or document the meaning of the data seen in the window.

To enter text labeling mode select the LABEL radio button on the control panel.

To create a text label position the mouse pointer somewhere in the 3-D window and press the left mouse button. A vertical bar cursor will appear at that location and you can now type in the text. The <Backspace> key can be used to correct errors. When you are finished, press <Return>.

To move a text label to a new position, point at it with the mouse, hold down the middle mouse button and drag the mouse. As you move the mouse an outline of the text will be dragged with the pointer until you release the mouse button.

To delete a text label, pointing at it with the mouse and pressing the right mouse button. Be careful, you will not be asked for verification before deleting a label. Once it's deleted you can only restore it by retyping it.

The SAVE button on the control panel will save any text labels you have made.

Use the '-font' command option to select a different font.

6.11 Data Probe

Sometimes it's useful to be able to inspect individual data values at various locations in the 3-D volume. You can do this with the data probe. Click on the PROBE radio button on the control panel. A 3-D cursor appears in the 3-D box which you can move around using the right mouse button. For each physical variable the value for the current timestep is printed along the left edge

of the 3-D window. If physical units are specified for the variable they will be printed next to the value. Units can be assigned with the v5dSetUnits() function in your data conversion program as described in section 3.1.

If you turn on the GRID #'s button, the probe will be constrained to integral grid coordinates. That is, the cursor will 'snap' to the nearest discrete grid coordinate.

6.12 Making New Variables

The NEW VAR button on the control panel is used to add new physical variables to the button matrix. There are three kinds of new variables you can add:

1. Cloned variables: these are copies of existing variables. You can use a cloned variable to make two different isosurfaces of the same variable simultaneously, for example.

2. External function variables: you can invoke an external function (which you write) to compute a new variable as a function of existing variables.

3. Computed variables: you can compute a new variable by typing in a formula involving values of existing variables.

When you click on the NEW VAR button a window appears which lists the variables that you can clone, lists the external functions that you can invoke, and lets you type in a formula for computing a new variable. After a new variable has been created anew row of buttons will be added to the control panel for the new variable. You can use then make isosurfaces, contour slices, etc. of the that variable like any other.

6.12.1 Cloned Variables

Suppose you want to clone the U wind component variable so that you can make both +20 and -20 isosurfaces of it. First, click on NEW VAR and then select U from the pop-up window. The cloned variable will be named U'. You can then treat U' as any other variable and make an isosurface of it.

6.12.2 Type-in Formulas

Type-in formulas let you type in mathematical expressions to compute new variables as a function of existing variables. For example, to compute wind speed from U, V, and W you would enter the formula:

SPD3D = SQRT(U*U + V*V + W*W)

To compute the ration of the dew point (TD) to the temperature you would enter the formula:

RATIO = TD / T

Formulas may use the names of existing variables, numbers, the arithmetic operations +, -, *, / and ** (exponentiation), and the functions SQRT, EXP, LOG, SIN, COS, TAN, ATAN (arc tangent), ABS (absolute value), MIN and MAX. MIN and MAX take two arguments, while the other functions all take one argument.

Click on the OK button to compute the new variable or CANCEL to discard the formula. You can edit the formula later by selecting it again from the NEW VAR pop-up window.

6.12.3 External Analysis Functions

External analysis functions are an advanced feature, so new Vis5D users may want to skip this section for now.

An external analysis function is a function written by you in FORTRAN which is called by Vis5D to produce a new variable as a function of the existing variables. As an example, there is included a function SPD3D which computes wind velocity as: SPD3D = SQRT(U*U + V*V + W*W). Be aware that the external function feature is intended for experienced Vis5D users who are also proficient FORTRAN programmers.

All external functions must be placed in a directory named "userfuncs" (this may be changed in the vis5d.h file). This is relative to the current directory when you run Vis5D. For example, suppose you always run Vis5D while in "/usr/jones/data", then your analysis functions must be in "/usr/jones/data/userfuncs". Also, this directory contains a script "externf" which is used to compile your function.

To write an external function it's best to copy one of the supplied examples and then modify it. The included "userfuncs/example.f" is fully commented for this purpose. Later, when you call your function from within Vis5D, the function will be invoked once for each timestep. The arguments passed to the function include:

1. the number of physical variables in the data set
2. the name of each variable
3. the size of the 3-D grid
4. the date and time of the timestep
5. map projection and vertical coordinate system information

6. the actual 3-D grids of data for each physical variable

Your function will have to scan the list of variable names to find the ones it needs for the computation. Then it must do the actual computation, generating a new grid of data to return to Vis5D. The examples we've included demonstrate how to do this. Specifically, you should look at example.f which has detailed documentation of the function arguments. The map projection and vertical coordinate system arguments work in exactly the same way as the v5dCreate library call discussed in Section 3.1.

Suppose you want your function to be named "delta". Then the name of the FORTRAN program must be "delta.f". You would compile the function by typing "externf delta". If there are no errors, an executable file "delta" will be written. Then in Vis5D when you select NEW VAR, "delta" should appear in the list of functions in the pop-up window.

There are two places for Vis5D to get the grid data which it passes to your external function: from the original, uncompressed McIDAS file or the compressed v5d/comp5d file. The uncompressed McIDAS data is better because it has more precision. If the McIDAS file can't be found, then the compressed data which Vis5D has in memory will be passed to your external function. Note that this has no bearing whatsoever on the construction of your external function.

You can retrieve the position and values of the data probe from within your function. To get the position of the probe use:

```
CALL PROBEPOS( ROW, COL, LEV, LAT, LON, HGT )
```

The position in grid coordinates will be returned in ROW, COLumn, and LEVel. The position in geographic coordinates will be returned in LATitude, LONgitude, and HeiGhT.

To get the value of any physical variable at the current probe position and current timestep use:

```
VALUE = PROBEVAL( VAR )
```

where VAR specifies which physical variable you want.

6.13 Saving Image Files and Printing

The SAVE PIC button on the control panel can be used to save the image in the 3-D window to a file. When you click on SAVE PIC a pop-up window appears in which you can select the file format and filename. The choices of file formats depends on the computer you're using. The formats supported by Vis5D are:

XWD - X Window Dump, displayable with xwud or xv.
RGB - SGI image file format, displayable with ipaste or xv.
GIF - Standard GIF format, displayable with xv and many other programs.
PostScript - may be printed or viewed on-screen with a program like ghostview.
Color PostScript - may be printed or viewed with a ghostview-like program.

The irix4 and irix5 configurations of Vis5D (using GL) directly write RGB files. To make a GIF file the togif program must be available. To make a PostScript file requires the tops program. togif and tops and many other RGB file converters are shipped standard with IRIX. If they're not found in /usr/sbin install them from your IRIX CD-ROM.

All other configurations of Vis5D (using OpenGL) directly write XWD files. To make an RGB file the fromxwd program is used. Unfortunately, the fromxwd program shipped by SGI has a bug which causes it to fail. Since source code for fromxwd is shipped with IRIX we include a patched version which works correctly. To make a gif file requires both fromxwd and togif (only available on SGI systems). To make a gray scale PostScript file requires the xpr utility (standard with X11). To make a color PostScript file the tops program is needed (only available on SGI systems).

If you don't have any of the utilities mentioned above you should try using xv to convert your image files. xv is available by ftp from export.lcs.mit.edu in contrib/ and from ftp.cis.upenn.edu in the pub/ directory.

To print a Vis5D image, position the mouse pointer over the 3-D window and press the P key. You'll be asked to verify your action. Vis5D uses lpr to send a PostScript image file to the default printer or the printer specified by the PRINTER environment variable. To generate the PostScript file Vis5D uses the utilities described above. If you have problems printing you should try to first save your image as a PostScript file then try to print it manually using lpr or lp. Another option is to save your image as an XWD file then use xpr (a standard X11 utility) to convert it to PostScript and print it.

To learn more about the xwud, xpr, fromxwd, tops and togif program read the man pages. Many of these programs have options which you may find useful.

6.14 Texture mapping

Texture mapping is a term from computer graphics which means to display a 2-D image over a surface in 3-D. In Vis5D you can display images over the topography (or bottom of the 3-D box when topography is turned off) such as satellite or map images. Texture mapping is only available on SGI systems and those using the Mesa library. Hardware support for texture mapping is highly recommended.

There are three types of texture/image mapping in Vis5D which can specified on the command line:

-area N — N is the number of the first of a sequence of McIDAS area files. The number of files read equals the number of timesteps in your data file. Images should all be of the same size. You must use McIDAS to do remapping if necessary.

Example: Suppose your datafile has 4 timesteps and you specify -area 100, then AREA0100, AREA0101, AREA0102 and AREA0103 will be loaded and displayed.

This option needs the McIDAS library which is only available on SGI systems.

-sequence file — This works like the -area option, except that the data come from a very simple file format rather than from McIDAS area files. The file starts with 3 int's that contain the number of images in the sequence, the number of lines per image, and the number of pixels per line. The rest of the file contains the images, one byte per pixel. The function read_texture_sequence in the image.c file of the src directory reads this file and serves as a file format reference for those wishing to create such image sequence files.

-texture file — This options specifies a single image to display over the topography for all timesteps. The file format is the SGI RGB format. The free XV program can be used to convert your image to RGB format.

When a texture map is available the TEXTURE button on the control panel is used to toggle the display of the imagery on or off.

6.15 TCL scripting

Vis5D 4.2 features a scripting facility. That is, you can control Vis5D with a text file of commands using the Tcl language. Scripting is an advanced subject and documented separately in the Vis5D scripting document at http://www.ssec.wisc.edu/~billh/script.html.

Note that the SAVE and RESTORE buttons on the control panel write and read Tcl files. You may want to use bits of these files as a basis a new Tcl scripts.

6.16 Keyboard Functions

The following keyboard functions can be invoked while the mouse pointer is inside the 3-D viewing window:

Key	Function
F1	Raise or lower the control panel window. This is useful with the - full option.
F2	Toggle display of system information including memory used and number of graphics to be computed.
P	Print the current window image. A PostScript printer must be available. Set the PRINTER environment variable from your shell to specify which printer to use.
S	Slower animation - increases the minimum time between frames by 10 msec.
F	Faster animation - decreases the minimum time between frames by 10 msec.

If you want to program your own keyboard functions look the in the file src/gui.c for the func1(), func2(), func3(), etc functions. They are called when the corresponding function key is pressed.

6.17 Final Notes

The SGI version of Vis5D uses multiple CPUs if available to compute graphics in the background thereby increasing Vis5D's speed. On other systems, Vis5D tries to interleave the computation of graphics with user interaction. This results in the user interface being a bit sluggish until all pending graphics computations are completed.

The Vis5D user interface may be complex to describe in words, but we have tried hard to make it simple in reality. After a little practice using the sample data sets we hope it feels natural.

Since version 3.2 of Vis5D there is a user-contributed software directory: contrib/. See the README file in that directory for a description of current contributions.

7.0 THE v5dimport UTILITY

The v5dimport utility is a new program for converting grid files to v5d format, combining multiple source files, resampling to new coordinate systems and culling variables and timesteps. It has both a graphical and command line user interface.

For example, you may use v5dimport to read 2 McIDAS GR3D files and a 2-D McIDAS GRID file, resample all the data to a Lambert Conformal projection, omit the CWAT and VORT variables and then write the data to a Vis5D file called lambert1.v5d.

The basic order of events when using v5dimport is:

1. Read the input file(s).
2. Select grids for output according to timestep, physical variable, map projection or vertical coordinate system.
3. Setup a map projection and vertical coordinate system for the output file.
4. Write the output file. Resampling is done at this time.
5. Optionally, start Vis5D on the output file.

Currently, v5dimport can read the following file formats:

McIDAS GR3D and GRID files
Vis5D v5d and comp5d files
GRADS files
"UW vis" files (used at the University of Wisconsin)
EPA MM4 and RADM files (on Crays only)

7.1 Using v5dimport's graphical interface

Start v5dimport from your shell with

v5dimport [-path pathname] [files]

where [files] is an optional list of input files and [-path pathname] specifies that the directory named "pathname" is to be used as the default, in place of the current directory, for the input file browser and for making output files.

When v5dimport has started you'll see its main window appear. It consists of:

1. a scrollable list of all grids scanned from the input files
2. buttons used for selecting/culling grids according to variable name, timestep, projection or vertical coordinate system.
3. buttons and type-in fields for describing and creating the output file.

7.1.1 Reading input grids

You may read additional grid files into v5dimport at any time by clicking on the "Read file..." button. Use the file selector to locate your file and click on OK or CANCEL. It's best to read all input files right at the beginning because whenever a new file is read all grids are selected for output, overriding any selections you may have previously made.

The button labeled "Discard all grids" does exactly what it says. It's equivalent to exiting v5dimport and restarting it. After reading each input file, the list of grids shown in the top half of the window, will be resorted by time then variable name.

The columns in this list are:

Grid - grid number (no significant meaning)
YYDDD - the year and date of the grid
HHMMSS - the time of the grid in hours, minutes, and seconds
Variable - the variable name
Nr - number of grid rows
Nc - number of grid columns
Nl - number of grid levels
Proj# - the projection number (see "Select by projection..." window)
VCS# - the vertical coordinate system number (see "Select by VCS...")
Filename - name of file the grid was found in

7.1.2 Selecting grids for output

It's often the case that one wants to discard some physical variables or timesteps from the input file so they aren't written to the output file. By default, all variables are selected for output.

To select/cull variables, click on the "Select by variable..." button. A pop-up window will appear which lists all the variables. The ones that are high lighted are selected for output. Click on variables names to select or deselect them.

Similarly, you can select timesteps via the "Select by time..." button. A pop-up window listing all timesteps will appear. Use the mouse to select the timesteps you want and unselect the timesteps you wish to omit. Note that you can select/deselect a number of timesteps by just dragging the mouse while holding down the button.

Finally, grids may be selected or discarded according to their map projection or vertical coordinate system (VCS) via the "Select by projection..." and "Select by VCS... buttons.

Note that as you select/deselect timesteps, variables, projections, or VCSs the effected grids will be high-lighted/unhigh-lighted in the main grid list.

The "Select All" and "Select None" buttons do just what they imply.

7.1.3 Defining the output file

The default parameters for the output file (grid size, projection, etc) are taken from the first file read in. You should always review these parameters before making your output file. It will often be necessary to change these values.

The number of rows, columns, and levels for the output file is specified by the type-in fields on the main window labeled "Rows", "Columns" and "Max Levels". Type in new values if the defaults are incorrect.

The map projection for the output file can be viewed and changed by clicking on the "Map projection..." button. In this pop-up window you'll be able to choose a map projection type then enter the specific projection parameters. There is also a "Guess" button which will attempt to find a reasonable output projection given the currently selected grid list. It's often helpful to have the "Select by Projection" pop-up window on-screen to compare the output projection to the input projections.

The vertical coordinate system for the output file can be viewed and changed by clicking on the "Vertical Coord System..." button. In this pop-up window you'll be able to choose a vertical coordinate system type and enter the specific parameters. This window also has a "Guess" button to try to find a reasonable default. Similarly, it's often helpful to have the "Select by VCS" pop-up window on-screen to compare the output VCS to the input VCSs.

7.1.4 Making the output file

Enter a filename for the output file in the type-in field at the bottom of the main window then click on "Make". Messages will be printed as the file conversion takes place. If there are any errors the process will halt. Note that generating the output file can be time-consuming if data must be resampled from the input grid's coordinate system to a new coordinate system for the output file.

If you click on "Visualize" this will make the file and then automatically start up Vis5D on that file (i.e., you don't need to click on "Make" first). If you type a filename in the type-in field, it will use that name. Otherwise, it will use your login name followed by ".v5d". If you want command line options on the Vis5D command, put them in a file named "vis5d_options". For example, "-mbs 64".

7.1.5 Miscellaneous

An options window is available by clicking on the "Options..." button.

The first item controls the "combining of co-located data". It may be the case that several 3-D grids, selected for output, are co-located in space and time. When computing the value to put in the output file you can either choose the data value from the higher resolution grid at that location, or take the average of all grid values at that grid location.

The second item controls how grid data is compressed in the output file. By default, grid values are scaled down to 1-byte integers. Alternately, you can scale down to 2-byte integers for better resolution, or perform no compression/scaling by selecting 4-byte floating point values. This option represents a tradeoff in file size and precision.

7.2 Using v5dimport's text interface

The text/type-in interface to v5dimport is useful when X is not available or when you want to run v5dimport with a script. To start v5dimport in text mode enter:

```
v5dimport -t [-path pathname] [files]
```

where [files] is an optional list of input files and [-path pathname] specifies that the directory named "pathname" is to be used as the default, in place of the current directory, for the input file browser and for making output files. Through the text interface it's possible to run v5dimport with a script by using your shell's import redirection feature:

```
v5dimport -t <script
```

After you've invoked v5dimport with the -t option you'll see a >> prompt at which you can issue any of these commands:

exit	exit v5dimport.
help	online help.
list	show lists of grids, timesteps, variables, map projections, or vertical coordinate systems.
read	read an input file.
keep/omit	used to select which grids, according to timestep, variable, map projection or vcs, are to be included in or omitted from the output file.
info	display parameters of output file.
rows	specify number of grid rows for output file.
columns	specify number of grid columns for output file.
levels	specify max number of grid levels for output file.
projection	specify the output file's map projection.
vertical	specify the output file's vertical coordinate system.
make	make the output file.

visualize make the output file and start Vis5D.

Using the text interface to v5dimport is similar in strategy to the graphical interface:

1. Read input files.
2. Select grids by timestep, variable, projection, and/or VCS. This is typically done by a series of list, omit, and keep commands.
3. Set/adjust output file parameters. Typically a series of info, rows, columns, levels, projection, and vertical commands.
4. Make the output file, or make the output file and start Vis5D.

Use the help command to learn the exact syntax for each command.

A v5dimport script is simply an ASCII file of v5dimport commands and their arguments. In the simplest case it may contain only a few commands such as:

```
# read my file, omit two vars, write v5d file
read mydata.dat
omit var CW
omit var RW
make outdata.v5d
exit
```

As v5dimport executes a script it prints each command and its result. Lines which start with a # are considered comments and ignored.

7.3 Adding support for new file format

v5dimport was written so that adding code to read new file formats should be easy. The source code for v5dimport is in the import/ subdirectory. Look for the comment /*** ADD NEW FORMATS HERE ***/ to see where code has to be added to support a new file format.

Basically, you need to write two new functions. One which scans your file format to build a list of grid_info structs. The other reads the actual grid data from your file given a grid_info struct. These functions should be put in a new file named read_foo.c where foo is the name of your file format. Then, update the file.c file to use your functions. Use the existing read_*.c files as a guide.

7.4 Notes on specific file formats

The symbol EPA is defined on the cc command line with -DEPA only on systems which can read EPA files. Currently, only Cray systems can read EPA files because the EPA-provided file reading functions only work on Cray computers.

The symbol MCIDAS is defined on the cc command line with -DMCIDAS only on systems which can use the libmcidas.a file. Only SGI's in 32-bit mode are supported now.

8. 0 SAMPLE DATA SETS

To demonstrate or experiment with Vis5D we provide two sample datasets.

8.1 Bob Schlesinger's thunderstorm simulation

To visualize the Schlensinger thunderstorm file enter the command

```
vis5d SCHL.v5d
```

To view an isosurface of QL (moisture content):

1. Click on the QL button in the left column of the button matrix.
2. On the slider, select a value near 1.0, then click on the OK button.
3. Turn on animation with the ANIMATE button.

To view a vertical contour line slice of QL:

1. Turn off animation by clicking on ANIMATE again.
2. Click on the QL button in the third column.
3. Move the slice by first selecting the SLICE radio button. Then use the right mouse button to drag any corner of the slice along the edges of the 3-D box.

8.2 LAMPS model

To visualize a LAMPS (Limited Area Meso-Scale Prediction System) model simulation of an extratropical cyclone, enter the command:

```
vis5d LAMPS.v5d
```

To view an isosurface of wind speed over a topography with map lines:

1. Click on the TOPO and MAP buttons.
2. Click on the SPD button in the first column. Then select a value near 45.0 on the slider and click on OK.
3. Turn on ANIMATE and you will see an animation of the 45 m/s wind isosurface.

To make some interactive wind trajectories:

1. Turn off the wind speed isosurface by clicking on the SPD button again
2. Select the TRAJECTORY button.
3. Move the mouse pointer into the 3-D window and press the middle mouse button. You will get a series of white wind trajectory lines passing through the 3-D cursor location.
4. Move the 3-D cursor by dragging it with the right mouse button then click the middle button to make more trajectories.
5. Select RIBBON and then the SET 2 button and try making some yellow ribbon trajectories.

8.3 Example McIDAS files and utilities

The Schlesinger and LAMPS data sets are also available as the 3-D McIDAS grid files named GR3D0001 and GR3D0002. They are available on the Vis5D ftp site. See section 2 for more information.

To list the grids in GR3D0001 and to see statistics about them, enter the commands:

```
igg3d list 1 190 -gr3df 1
igg3d info 1 190 -gr3df 1
```

The SCHL.v5d file was made from the GR3D0001 file with the command:

```
gr3d_to_v5d 1 1 SCHL.v5d
```

To list the grids in GR3D0002 and to see statistics about them, enter the commands:

```
igg3d list 1 189 -gr3df 2
igg3d info 1 189 -gr3df 2
```

The LAMPS.v5d file was made from the GR3D0002 file with the command:

gr3d_to_v5d 2 1 LAMPS.v5d

A variety of other sample datasets are available on the ftp site or upon request.

9.0 VERSION HISTORY

This is a summary of the versions of Vis5D.

1.0 (December 1988)

This was the original version of Vis5D for the Stellar GS-1000. It was used to give demonstrations at the ECMWF in December 1988 and at the AMS conference in Anaheim in January 1989. It had the following features:

Depict time series of multivariate 3-D grids by animated isosurfaces and horizontal contour line slices.
World topography map with map boundaries.
Wind trajectory tracing with the traj5d program.

2.0 (Fall 1991)

This version was only available for the Stellar GS-1000/2000 and introduced the following features:

Faster isosurface generation.
Horizontal and vertical slices moved interactively with the mouse.
Colored slices.
Interactive wind trajectory creation.
Ribbon trajectories.
Label / text annotations.
"Pretty" rendering option.

The format of the compressed grid file was changed slightly with version 2.0. Specifically, the trajectory files of version 1.0 were eliminated, trajectories are now stored in the compressed grid file itself. Also, the internal storage representation for surfaces and slices has been changed.

2.1 (February 1992)

This is the first version of Vis5D available for the SGI and IBM workstations. It was also modified to use less memory during isosurface generation.

2.2 (April 1992)

This version of Vis5D runs on the base SGI Indigo with 8-bit color though some features not available. It also has the following improvements:

The -box option for changing the proportions of the 3-D box (SGI and Stardent only).
User topography files. Vis5D now uses the EARTH.TOPO file instead of TOPOHRES to make the map. The maketopo.c program shows how to make new .TOPO files. (SGI and Stardent only)

3.0 (August 1992)

This version features the following improvements:

Horizontal and vertical wind vector slices.
Improved SAVE and RESTORE functionality.
New trajectory widget options.
Separate map and topography controls.
CLONE option added.
Simultaneous colored and contour line slices.
Improved transparency, PRETTY option on SGI.
Same source code for SGI, Stardent, and IBM.
Improved portability and porting guide added.
New video and hard-copy convenience features.

3.1 (July 1993)

New features:

User-written analysis functions.
SAVE PIC button to save window image to a file.
Perspective viewing mode.
New contour line options to draw dashed negative lines and restrict contouring to a specific range of values.
Data Probe mode.
Topography color editing.
Grid compression done layer-by-layer.

3.2 (August 1993)

New features and changes:

Volumetric rendering on SGI systems with VGX, VGXT, VTX, RE, or RE2 graphics hardware.
User-contributed software directory.
2-D contour function rewritten in C.

3.3 (January 1994)

New features:

Vis5D ported to HP, DEC, Sun, and Kubota (DEC Alpha) workstations.
The most important part of this work was the enhancement and integration of the VOGL library. This work was done by Simon Baas and Hans de Jong for the Dutch Meteorological Institute, KNMI. Porting to the Kubota Denali graphics system was done by Pratish Shah of Kubota Inc.

-wdpy option now creates a window on the widget display which can be used to move and interact with the 3-D view using the widget display's mouse.

SAVEPIC button let's you save the window image in PostScript or color PostScript formats (SGI only).

-wind2 option added to specify a second set of U, V, W variables for the second set of wind vector slices.

-texture option added for a texture mapping an image onto the topography (SGI only).
user functions are computed faster on SGI multi-processor systems by computing time steps in parallel.

4.0 (December 1994)

New features:

Map projections and new vertical coordinate systems.
Type-in formulas for computing new variables.
Time sequences of satellite images can be texture mapped onto the topography for visual comparison with model data.
Data may be displayed over a spherical Earth.

File caching: compressed grid files which are too large to read into memory in their entirety are read in piece-by-piece as needed, a least-recently-used replacement policy is used to purge data when memory is full.

New compressed grid format. New format allows new header information to be added in the future, currently stores additional projection information. Also allows control of data compression.

New command line options: -geometry, -trajvars, -projection, -vertical, -area, -sequence

External functions can query the probe position and values with PROBEPOS and PROBEVAL functions.

Interactive control over animation rate (using F and S keys)

When the "GRID #'s" button is turned on, the probe/trajectory cursor snaps to discrete grid points.

New utilities for .v5d files: v5dinfo, v5dstats, v5dedit, comp_to_v5d, and gr3d_to_v5d.

4.1 (May 1995)

New features:

Rotated map projection.

Improved widgets.

Stored-frame animation.

Better 3-D rendering in software using Mesa instead of VOGL.

Vis5D files defined as a World Wide Web medium for exchanging model output.

4.2 (April 1996)

New features:

Wind streamlines.

Colored isosurfaces and trajectories.

Scripting with Tcl.

UVW variable widget.

Pressure vertical coordinate system.

programmer's API between Vis5D and its user interface.

v5dSetLowLev function allows fields to occupy any sub-interval of vertical levels.

physical units can be specified for each variable in a v5d file.

v5dimport program.

APPENDIX F

PAVE User Guide - Version 1.4.1.

PAVE User Guide - Version 1.4.1

Table of Contents

1. Introduction

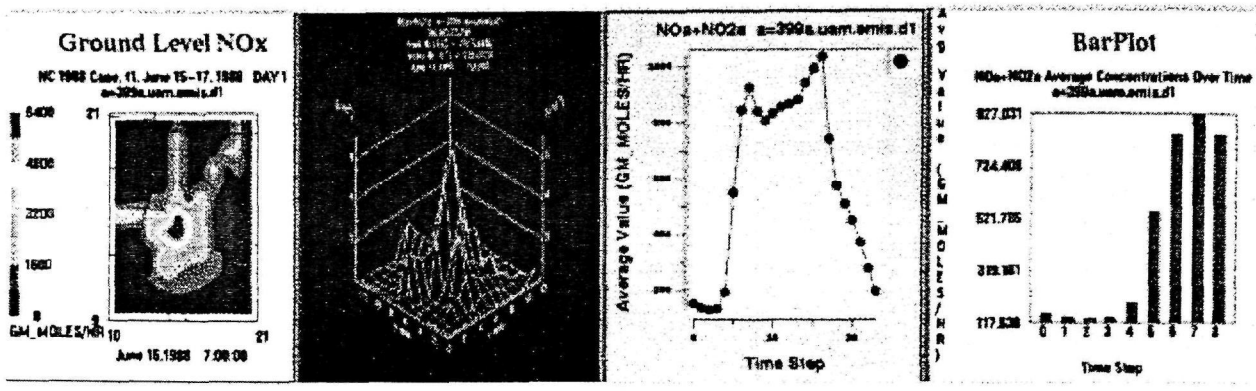
This document describes how to use MCNC's Package for Analysis and Visualization of Environmental data (PAVE), a flexible and distributed application to visualize multivariate gridded environmental datasets. Features include (1) baseline graphics with the option to export data to high-end commercial packages, (2) access and manipulation of datasets located on remote machines, (3) support for multiple simultaneous visualizations, (4) an architecture that allows PAVE to be controlled by external processes, (5) low computational overhead, and (6) no software distribution cost.

PAVE version 1.4.1 source code was frozen on 3/17/97, and binary executables for AIX4 (IBM), IRIX 5.x (SGI), Solaris 2.5 (Sun), OSF1 (DEC Alpha), and HP-UX 9.x (HP) are available for the use of interested visualizing suspects. Instructions for obtaining and installing PAVE are available from http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.download.html.

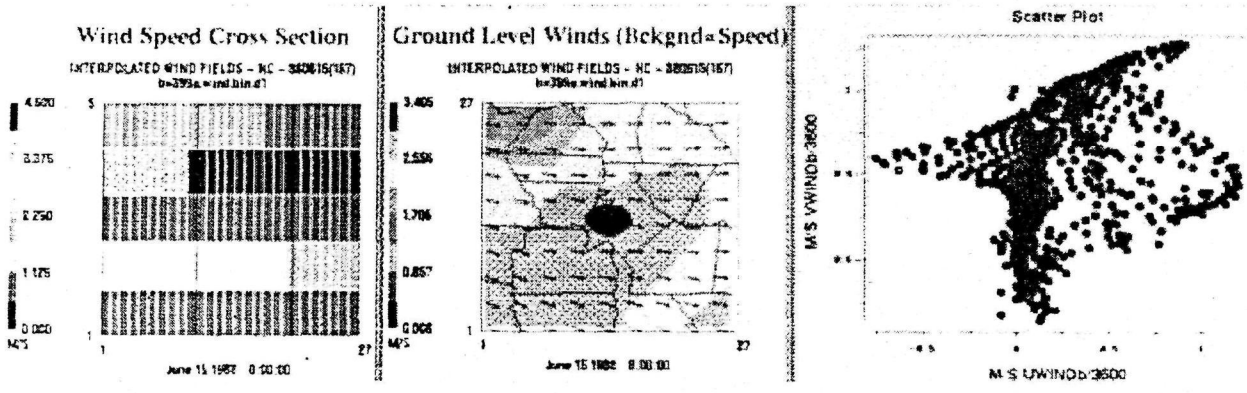
If you are reading this document for the first time, we recommend that you first briefly scan the entire document. Then you may wish to try the suggestions in the Quick PAVE Jumpstart section to help you in your first hands-on PAVE session.

2. Types of plots produced by PAVE

From left to right, these are examples of a smoothed tile plot, a 3D mesh plot, a time series line plot, and a time series bar plot.



From left to right, these are examples of a tile plot of a vertical cross section, a wind vector plot, and a scatter plot.



3. Getting your data into PAVE

PAVE currently supports four input file formats: IO/API (netCDF), UAM-IV, UAM-V, and "chain files" (which contain a list files of one of the other three types). Files of these types are loaded into PAVE from the Add/Delete/Select Dataset_popup window. This window comes up automatically when you start PAVE, and also appears when you choose Edit/Select From Dataset List from PAVE's Datasets menu. See Quick PAVE Jumpstart for more details on loading datasets.

I/O API Format

The Models-3/ EDSS Input/Output Applications Programming Interface (see http://www.iceis.mcnc.org/EDSS/ioapi/H.AA.html for further info) provides an easy-to-learn, easy-to-use programming interface to files for the air quality model developer and model-related-tool developer, in both FORTRAN and C/C++. I/O API files are portable across computing platforms. This means that the same file can be read on a Sun workstation, a DEC Alpha workstation, and on a Cray supercomputer. If your data is not already in I/O API, UAM-IV, or UAM-V format, we recommend that you write a translation program to convert your data into I/O API format.

There is source code for two example conversion programs is included with the PAVE distribution. One of them converts an ASCII ROM dataset with a Lat/Lon map projection, and the other converts a binary SAQM dataset with a Lambert Conformal map projection. Note that you will need additional libraries, and possibly header files, in order to successfully compile this code. These libraries and header files can be found in:

<top level PAVE dir>/<platform type>/lib/*

The example conversion code is in:

<top level PAVE dir>/doc/pave/convert_to_m3ioapi/rom_to_m3ioapi/*
<top level PAVE dir>/doc/pave/convert_to_m3ioapi/saqm_to_m3ioapi/*

NOTE: these example codes will probably need to be adapted for use with your data files. They were written for specific data files being used for testing purposes, and were NOT originally intended to be cleanly written examples of how to easily convert generic gridded data into PAVE format. However, they are being included here with the hope that you may find them useful.

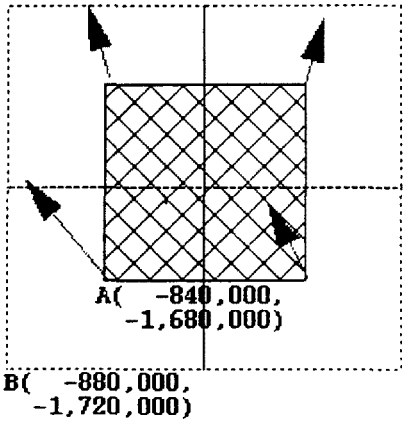
One sometimes confusing concept is the precise meaning of the XORIG, YORIG, XCELL, and YCELL parameters in an IO/API data file. Below is an image of an example of using these parameters with two different data files that need to be co-registered within a PAVE plot. Note that units are in meters.

```
Cross: NROWS = 1, NCOLS = 1,
       XORIG = -840,000, YORIG = -1,680,000
       XCELL =   80,000, YCELL =     80,000

Dot:   NROWS = 2, NCOLS = 2,
       XORIG = -880,000, YORIG = -1,720,000,
       XCELL =   80,000, YCELL =     80,000
```

The cross-hatched square represents the single cross-grid cell colored by some variable, say, O3. Note that the lower-left corner of the sqaure is at the cross-grid origin (-840,000, -1,680,000).

The four arrows are 2D wind vectors from the dot-grid. They are rooted at the centers of the cells of the dot-grid (dotted line). For example, the first arrow is rooted at:
dot-grid(xorig + xcell / 2, yorig + ycell / 2) =
(-880,000 + 80,000 / 2, -1,720,000 + 80,000 / 2) = (-840,000, -1,680,000).



UAM-IV Format

Most geographically oriented UAM-IV input and output files are directly readable by PAVE. These include DIFFBREAK, REGIONTOP, TEMPERATURE, WIND, AIRQUALITY, BOUNDARY, TOPCONC, EMISSIONS, PTSOURCE, AVERAGE, and INSTANT.

UAM-V Format

Some of the UAM-V input and output files are compatible with the UAM-IV file formats. These files are readily visualized by PAVE because it can obtain all the information it needs from inside the files (i.e. they are "self-describing"). The files in this category are: EMISSIONS,

PTSOURCE, BOUNDARY, AIRQUALITY, and the coarse grid AVERAGE and INSTANT files.

However, many UAM-V files have new formats that are not self-describing. PAVE needs additional information in order to read these files, such as the kind of data in the file, the number of rows, columns, and layers in the data, the geographic region covered, etc. The file types that fall into this category are wind, temperature, cloud, water vapor, rain, vertical diffusion, height, fine grid average, and fine grid instant.

In order to display a file of one of these latter types with PAVE, a UAM-V "metafile" must be used. A metafile is an ASCII file that contains the additional information PAVE needs to read and correctly interpret the data. To visualize data in a file that needs a metafile, select the metafile from the PAVE file browser, instead of the file that contains the data. A description of the contents of UAM-V metafiles follows.

The very first line of the metafile MUST be the following:

#! UAMV DESCRIPTION FILE

If PAVE does not find the above string, the file is considered to be of unknown type and an error is returned.

Several keywords must be present in the metafile. Each keyword should be on a separate line and followed by a value. Blank lines are permitted. The following keywords are required (and are listed in the recommended order):

UAMV_FILE
UAMV_TYPE
NCOLS
NROWS
LEVELS
XORG
YORG
UTM_ZONE (for UTM-based domains only)
DX or DLON
DY or DLAT
FINE_GRID
TITLE

Keyword Descriptions

UAMV_FILE - the name of the file that contains the actual data to be displayed by PAVE. It can be either a full or a relative pathname to the current working directory.

UAMV_TYPE - the type of data in the UAMV_FILE. Valid choices are: Wind, Temp, Cloud, H2O, Rain, Vdif, Height, FineGridAverage, and FineGridInstant.

NCOLS - number of horizontal columns in the grid.

NROWS - number of horizontal rows in the grid.

LEVELS - number of vertical layers in the grid.

XORG - the x-coordinate of the lower left corner of the grid, in km for UTM-based grids and degrees for latitude-longitude grids.

YORG - the y-coordinate of the lower left corner of the grid, in km for UTM-based grids and degrees for latitude-longitude grids.

DX - size of the horizontal cell in the x-direction for UTM-based grids (km).

DLON - size of the horizontal cell in x-direction for lat-lon grids (degrees).

DY - size of the horizontal cell in y-direction for UTM-based grids (km).

DLAT - size of the horizontal cell in y-direction for lat-lon grids (degrees).

FINE_GRID - logical variable. Possible values: 0 and 1. If the value is 1, the file represents data on a fine grid, otherwise it is on a coarse grid. For meteorological input files, this value should be set to one only if there is a layer of cells outside the domain specified by { XORG, YORG, NROWS, NCOLS, DX, and DY]}.

TITLE - A title for tile plots. Spaces are allowed.

Example of a UAM-V temperature metafile on the OTAG coarse grid:

```
#! UAMV DESCRIPTION FILE
UAMV_FILE       /home/user/tmpr.cc.20jul93.ld.rams1
UAMV_TYPE       Temp
NCOL            64
```

NROWS	63
LEVELS	5
XORG	-99.0
YORG	26.0
DLON	0.5
DLAT	0.3333333
FINE_GRID	0
TITLE	Coarse Grid Temperature: July 20, 1993

Example of a UAM-V fine grid average metafile on the OTAG fine grid:

#! UAMV DESCRIPTION FILE	
UAMV_FILE	/home/trayanov/testpave/avrg.ff.20jul93-93.mc.93basA1
UAMV_TYPE	FineGridAverage
NCOLS	137
NROWS	110
LEVELS	7
XORG	-92.0
YORG	32.0
DLON	0.16666667
DLAT	0.11111111
FINE_GRID	1
TITLE	Base Case Fine Grid Average

Creating metafiles with scripts

It is relatively straightforward to use a script to create metafiles. Examples of scripts that do this follow. However, the specifics (such as parsing file names to determine file type) depend on the particular application. The two examples given below are for the OTAG project. Note that these are hypertext links which are only available if you are reading this document using a Web browser.

Script to create OTAG input metafiles

Script to create OTAG output metafiles

The above scripts are provided with your PAVE installation. They are called make_in_metas and make_out_metas and are located in the scripts/subdirectory of the top level directory where PAVE is installed. Before using them, be sure to edit the upper portion of the scripts to match your episode and directory structure.

Viewing multiple sequential files as one dataset using chain files

Data from multiple files can be concatenated and displayed as if they were from a single file. This is useful if you wish to animate over sequential data that is stored in multiple files. In order to do this you need to supply a "chain file" - an ASCII file that contains a list of the data files to be concatenated.

The first line of a chain file must be

#! LIST_OF_CHAINED_FILES

If this is not found, PAVE will not recognize that the file is a chain file. A list of full path names to the individual files to be concatenated should follow this line with one filename per line. Note that no blank lines or comment lines are permitted and the files MUST be in the order in which you wish the data to be displayed. The individual files can be one of the following types: netCDF, UAM-IV or UAM-V (regular or meta files).

A script is provided with PAVE to simplify the creation of chain files. The script will currently work only in cases where the files to be chained reside in a single directory. To create a chain file, cd to the directory that contains the files you wish to chain, and type

chain_files name_of_chain_file list_of_files_to_chain

For example:

chain_files avrg.cc.20-30jul93-93.mc.basB avrg.cc.*

Limitations: The current version of PAVE does not check the files for consistency (i.e. whether they are of the same type, whether the grid is the same in all the files, or whether the files are listed in the proper sequential order).

Here is an example of a chain file:

```
#! LIST_OF_CHAINED_FILES
/home/trayanov/testpave/rain.cc.20jul93.ai.meta
/home/trayanov/testpave/rain.cc.21jul93.ai.meta
/home/trayanov/testpave/rain.cc.22jul93.ai.meta
```

4. Using formulas

One of PAVE's most powerful features is its formula capability, which enables you to calculate and visualize derived variables from your datasets "on the fly". For example, you can calculate the ratio of a variable from one file to a variable from another file, and then visualize the ratio. It is easy to load formulas into PAVE using the Add/Delete/Select Formula_popup window, which appears automatically when you start PAVE. The window can also be brought up manually by choosing Edit/Select From Formula List from PAVE's Formulas menu. See Quick PAVE Jumpstart for more details on loading formulas.

All PAVE visualizations are generated using one or more formulas. A formula may be very simple. For example, the formula "O3a" refers to the variable "O3" in data set "a" - which is the first dataset that was loaded into PAVE. (Note that data sets are given sequential letters as they are loaded into PAVE, and are referred to by those letters in PAVE formulas.) An example of a formula to calculate the percent difference in O3 between datasets a and b is: "(O3a-O3b)*100/(O3b+0.001)".

Formulas must be in infix notation, and can contain the following operators, listed in their order of precedence:

Highest Precedence	1) abs, sqr, sqrt, exp, log, ln, sin, cos, tan, sind, cosd, tand, minx, miny, minz, maxx, maxy, maxz, mint, maxt, mean, sum, min, max,
	2) **
	3) /, *
	4) +, -
	5) <, <=, > >=
	6) ==, !=
	7) &&
Lowest Precedence	8) ||

Explanations of these operations are given below. If you wish to override the default operator precedence, or are uncertain as to which operator will take precedence, you can feel free to use parentheses in your formulas. This will force expressions within the parentheses to be evaluated first.

PAVE also has an occasionally used feature that allows you to specify a time step index after a variable name. For example, O3a:1 is the first hour of ozone. So, if you wanted to plot each cell averaged in time over the first six hours of your data, you could enter and plot the following formula:

(O3a:1+O3a:2+O3a:3+O3a:4+O3a:5+O3a:6)/6

This is cumbersome and it also uses a lot of memory, but it may be useful for you.

There is another useful feature of the parser that not many people know about, that enables you to compute and visualize the rate of change of a variable. For example, the formula d[O3a]/dt calculates the change in ozone concentration over time. A limitation of this feature is that the variable between the brackets must be an atomic variable, that is to say, it can not be a formula other than a basic variable from one of your datasets.

Formulas may also contain integer or floating point constants, or the following operands which are replaced by PAVE's formula parser to be the constant values noted:

E	2.7182818284590452354
PI	3.14159265358979323846
NROWS	number of rows in the formula's currently selected domain
NCOLS	number of rows in the formula's currently selected domain
NLEVELS	number of levels in the formula's currently selected domain

The following operators are binary (they have an operand on both sides of the operator), and usually return an array of data by performing that operation on each cell of the operands' arrays. The only time these operators return a single number is when both operands (ops) are themselves a single number.

+	Returns the sum of the ops
-	Returns the difference of the ops
*	Returns the product of the ops
/	Returns ratio of the ops
**	Returns the left op raised to the power of the right op, calculated using the C math library's pow() function

The following operators are boolean binary operators. Boolean operators return either 0 or 1 for each cell of the resulting array of data, or in the case of two operands (ops) that are single numbers, just the single number 0 or 1. You may find these operators useful to "screen out" ranges of your data that are of particular interest. For example, if you are only concerned about the variable O3a when its value exceeds 0.080, you might look at the formula (O3a>0.080)*O3a. If O3a is less than or equal to 0.080, the result of the formula will be set to 0 in that cell. Otherwise, the value of O3a will be used in that cell.

<	Returns 1 if the left op is less than the right op, else 0
<=	Returns 1 if the left op is less than or equal to the right op, else 0

>	Returns 1 if the left op is greater than the right op, else 0
>=	Returns 1 if the left op is greater than or equal to the right op, else 0
!=	Returns 1 if the left op is not equal to the right op, else 0
==	Returns 1 if the left op is equal to the right op, else 0
&&	Returns 1 if both ops are non-zero, else 0
||	Returns 1 if either op is non-zero, else 0

The following operators are unary (they have a single operand on the right side of the operator), and usually return a time-stepped matrix of data by performing that operation on each cell of the operand's array. The only time these operators return a single number is when the operand is itself a single number. The C math library routines called are listed with most of these operators. For further information on these routines, please check your man pages.

abs	fabs(op)
sqrt	sqrt(op)
sqr	Returns the square of the op
log	log10(op)
exp	exp(op)
ln	log(op)
sin	sin(op)
cos	cos(op)
tan	tan(op)
sind	sin(op*(PI/180.0))
cosd	cos(op*(PI/180.0))
tand	tan(op*(PI/180.0))

The following unary operators return a single number in all cases. Their single operand must follow on the right hand side of the operator. The functionality is listed beside each operator name.

mean	average cell value for all cells in currently selected domain
sum	sum of all cell values in currently selected domain
mint	time step index with minimum value in currently selected domain
maxt	time step index with maximum value in currently selected domain
minx	x index with minimum value in currently selected domain
maxx	x index with maximum value in currently selected domain
miny	y index with minimum value in currently selected domain
maxy	y index with maximum value in currently selected domain
minz	z index with minimum value in currently selected domain
maxz	z index with maximum value in currently selected domain

where "currently selected domain" includes the currently selected rows, columns, layers, and time steps for the currently selected formula. Thus, the minimum and maximum values in the currently selected domain occur at the locations (minx,miny,minz,mint) and (maxx,maxy,maxz,maxt), respectively. An interesting use of the sum operator is to calculate the sum of the result of a boolean expression (e.g. sum(O3a>120)) to find the number of cell-hours that meet the boolean condition.

The unary min and max operators behave a little differently:

min For each cell (i,j,k) in the currently selected domain, this calculates the minimum value for that cell over the currently selected time steps. In other words, the minimum value in cells (i,j,k,tmin..tmax).

max For each cell (i,j,k) in the currently selected domain, this calculates the maximum value for that cell over the currently selected time steps. In other words, the maximum value in cells (i,j,k,tmin..tmax).

NOTE: currently the unary + and - operators [as in -1 or -(x+y)] are not supported, but hopefully these will be added later.

5. Spatial and temporal data subsetting

PAVE allows you to easily subset your data by geographic region, layer range, and time range. This section explains the concepts of how PAVE manages this information in its memory space. An understanding of this section should help you with your PAVE usage. To adjust layer ranges, geographic regions of interest (domains), or time step ranges for datasets and formulas, you can use menu items in the Datasets and Formulas menus on PAVE's main interface window.

Each dataset has some number of layers greater than or equal to one, some number of time steps greater than or equal to one, and some geographic region onto which its grid maps. A geographic region, often referred to as a domain, is defined by the map projection type (Lat/Lon, UTM, Lambert Conformal, etc.), the number of rows and columns in the dataset, and the geographic boundaries of the area on which the grid falls.

For each unique geographic region (or domain) that PAVE learns about by examining the dataset(s) chosen by the user, PAVE keeps a single domain object. Domain objects are where the currently selected geographic region for formulas and datasets are stored in memory. When a domain object is created by PAVE, it defaults to having all of its cells selected, until the user chooses a subregion within that domain. Because PAVE only stores a single domain object for each unique domain that it learns of, there may be numerous datasets and formulas that make use of the same domain object. Therefore, when a user chooses to modify a formula's or a dataset's

selected region of interest, the region of interest for all formulas and datasets that refer to the same region and have the same grid dimensions will be affected. Any subsequent plots of any formulas who rely on that domain object will reflect its newly selected geographic region of interest.

Similarly, PAVE uses layers objects to store information about currently selected layer ranges. Each dataset i has some Ni number of layers associated with it, and each formula j has some Nj number of layers associated with it. For each unique number of layers that PAVE learns about, PAVE keeps a single layers object in memory. Therefore, when a user chooses to modify a formula's or a dataset's selected layer range, there may be more than one dataset and more than one formula on which the modified layer range has an effect. When PAVE learns about a new unique number of layers, the layers object that it creates by default has only the first layer selected, until the user modifies the layer range for this layers object.

Time ranges are handled differently than layer ranges and geographic regions of interest. Each dataset has exactly one unique time range object associated with it. That time range object is associated only with that one dataset. Each formula also has exactly one unique time range object associated with it. That time range object is associated only with that one formula. Because a formula may rely on more than one dataset to derive its data, a formula's time step range is always bounded by the time ranges of the datasets it depends on. For example, a formula could have three datasets from which it derives data. Those datasets could have 72, 48, and 24 time steps of data in their currently selected time range objects. In this case, the maximum number of time steps in that formula's time range object could ever have would be 24. Should the user reduce the number of steps in the currently selected time range of any of the three datasets to less than 24, then the maximum number of time steps in that formula's time range object would immediately be reduced accordingly.

6. Navigating through PAVE's menu items

PAVE's main graphical user interface is broken into six menus: File, Datasets, Formulas, Graphics, Export, and Help. This section briefly describes each of the items on these menus.

File Menu

The file menu has six menu items.

Choose Configuration File For New Tile Plots brings up a file browser that allows you to select a file that contains configuration settings for tile plots. Settings can include information such as colors to use, number of colors to use, min and max values for the legend, etc. These settings will be used for all subsequent tile plots, until another configuration file is chosen. The

ASCII configuration files can be edited manually, or saved directly from a tile plot. For further information, please see the section on Configuring plots.

Load Dataset List From File brings up a file browser that allows you to select an ASCII file that contains a list of datasets that you would like to visualize with PAVE. Loading in a dataset list from a file will first cause PAVE to remove any currently loaded datasets from PAVE's memory. In this way, dataset names loaded in will then be referenced by letters a, b, c, etc.

Load Formula List From File brings up a file browser that allows you to select an ASCII file that contains a list of formulas that you would like to visualize with PAVE. Note that any formulas loaded in will be verified against the currently loaded datasets. As long as the variables in the formulas match variables in the respective datasets, they will be loaded into PAVE. Otherwise they will be ignored.

Save Dataset List To File allows you to save a list of the currently loaded datasets to an ASCII file. If you want to look at the same set of datasets in many different PAVE sessions, you may find this feature useful. Dataset lists can subsequently be retrieved using the Load Dataset List From File menu item described above.

Save Formula List To File allows you to save a list of the currently loaded formulas to an ASCII file. If you want to look at the same set of formulas in many different PAVE sessions, you may find it useful to use this feature. Formula lists can subsequently be retrieved using the Load Formula List From File menu item described above.

Exit PAVE causes your PAVE session to end.

Datasets Menu

The datasets menu allows you to modify the information PAVE stores regarding available datasets.

Edit/Select From Dataset List brings up an "Add/Delete/Select Dataset" window that allows to you load and delete datasets from PAVE's memory. Click on the "Add" button, and an EDSS File Browser should appear (this may take a couple of seconds). You can then browse for datasets on your local machine and on remote machines if your system is properly configured. For more information on using the EDSS File Browser, see Quick Pave Jumpstart.

If you want to browse for files on remote machines, first verify that you've met the requirements discussed in the Requirements for use section of this document. Then click on the large button in the center of the EDSS File Browser that contains information about "Host:", "User:", and "Owner Module:". This brings up a "host selector" window in which you can enter the remote

host name (e.g. sequoia.nesc.epa.gov). Clicking on "Select" will cause the file browser to go to your home directory on the remote machine if your system is configured properly. You should then be able to browse for files on that machine.

Datasets selected with the EDSS File Browser are loaded into PAVE sequentially, and are referenced using the letters a, b, c, etc. When a new dataset it loaded into PAVE, it becomes the currently selected dataset. You can always determine which is the currently selected dataset by looking at the "Dataset:" line on the main PAVE window. You can change the currently selected dataset by simply single-clicking on the desired dataset name in the "Add/Delete/Select Dataset" window. PAVE's "Species List" window will then contain the names of the variables contained in the dataset, each followed by the letter corresponding to the dataset.

Clicking on a dataset name and then clicking on the "Delete" button will remove that dataset from PAVE's memory. Clicking on the "Close" button will close the window.

Select Time Range of Current Dataset brings up a window with two sliders that can be used to crop the currently selected dataset's time range to a smaller time range, which by default is set to the maximum range in the dataset. Cropping, or subselecting, a dataset's time range will affect any plots made using that dataset's variables. The maximum time range that can be used for any of the variables in a plot is specified by these sliders. Please see the section on Spatial and temporal data subsetting for further information on how PAVE stores time ranges.

Select Layer Ranges Matching Current Dataset brings up a window with two sliders that can be used to change the currently selected dataset's layer range to a different layer range. By default, a dataset's layer range is set to be layer 1 (ground level) only. Changing a dataset's layer range will affect any plots made using that dataset's variables, and may also affect plots made using other datasets. The maximum layer range that can be used for any of the variables in a plot is specified by these sliders. Note that there may be multiple datasets and formulas that share the same layer range information. Please see the section on Spatial and temporal data subsetting for further information on how PAVE stores layer ranges.

Select Regions Of Interest Matching Current Dataset brings up a window that can be used to change the currently selected dataset's geographic region of interest. By default, a dataset's geographic region of interest is the entire region referenced by the dataset. This window can be used to select part of this region. Changing a dataset's geographic region of interest will affect any plots made using that dataset's variables, and possibly those of other data sets because there may be multiple datasets and formulas that refer to the same region of interest.

The geographic region of interest window has its own menu. The File menu can be used to save a geographic region of interest to a file, retrieve a region of interest from a file, or to close the window. The Edit menu can be used to easily select all of the grid cells on the window, or to

select none of them. Cells can explicitly be toggled on or off by dragging the mouse over the region to be toggled. You may find that selecting a particular region is easier if you resize the window to make it larger. Please see the section on Spatial and temporal data subsetting for further information on how PAVE stores geographic regions of interest.

View Variables In Current Dataset will cause a window to appear that lists the variables in the currently selected dataset, followed by the letter that corresponds to the dataset. You can click on any of the variables to cause that variable to be added to the formula list and become the currently selected formula. The formula can then be plotted using selections from the Graphics menu without any further clicks of the mouse.

Formulas Menu

The formulas menu allows you to modify the information PAVE stores regarding available formulas.

Edit/Select From Formula List brings up an "Add/Delete/Select Formula" window that allows to you add and delete formulas from PAVE's memory. You can enter a formula using the "Enter New Formula:" type in widget according to the rules described in the Using formulas section, then click on the "Add" button to load it into PAVE's memory. When a new formula is loaded into PAVE, it becomes the currently selected formula. You can always determine which is the currently selected formula by looking at the "Formula:" line on the main PAVE control window. You can change the currently selected formula by simply clicking on the desired formula name in the "Add/Delete/Select Formula" window.

Clicking on a formula and then clicking on the "Delete" button will remove that formula from PAVE's memory. Clicking on the "Close" button will close the window.

Select Time Range of Current Formula brings up a window with two sliders that can be used to crop the currently selected formula's time range to a smaller time range. Cropping, or subselecting, a formula's time range will affect any plots made using that formula. The maximum time range that can be used for the formula in a plot is specified by these sliders. Please see the section on Spatial and temporal data subsetting for further information on how PAVE stores time ranges.

Select Layer Ranges Matching Current Formula brings up a window with two sliders that can be used to change the currently selected formula's layer range to a different layer range. By default, a formula's layer range is set to be layer 1 (ground level) only. Changing a formula's layer range will affect any plots made using that formula, and possibly that of other plots too since there may be multiple datasets and formulas that share this same layer range information. Please see the

section on Spatial and temporal data subsetting for further information on how PAVE stores layer ranges.

Select Regions Of Interest Matching Current Formula brings up a window that can be used to change the currently selected formula's geographic region of interest. By default, a formula's geographic region of interest is set to be the entire region referenced by the formula. This window can be used to select part of this region. Changing a formula's geographic region of interest will affect any plots made using that formula's variables, and possibly other plots since there may be multiple datasets and formulas that share this same geographic region of interest information.

The geographic region of interest window has its own menu. The File menu can be used to save a geographic region of interest to a file, retrieve a region of interest from a file, or to close the window. The Edit menu can be used to easily select all of the grid cells on the window, or to select none of them. Cells can explicitly be toggled on or off by dragging the mouse over the region to be toggled. You may find that selecting a particular region is easier if you resize the window to make it larger. Please see the section on Spatial and temporal data subsetting for further information on how PAVE stores geographic regions of interest.

Graphics Menu

The graphics menu is used to create plots.

Create Tile Plot makes a tile plot of data for the currently selected formula using that formula's selected geographic region of interest, layer range, and time step range. A tile plot has its own menus that can be used to operate on that tile plot and/or its data.

- The tile plot's File menu can be used to print and save images and animations; these are described more thoroughly in the Printing and exporting images, animations, and data section. Also this menu has a "Save Configuration Settings" item that saves the tile plots legend settings to a file, as discussed in the Saving configuration settings for later use subsection of the Configuring plots section.

- The tile plot's Interact menu can be used to change the effect of a dragging the mouse pointer to select a portion of the tile plot's cells. Interact..Probe causes dragging to bring up a window with the raw data values for selected region. Interact..Zoom causes dragging to zoom in on the selected region. Interact..Time Series causes dragging to create a time series line plot of the average values over the selected region. WARNING: using Interact..Time Series in PAVE 1.4.1 from a tile plot that was generated for a subdomain of data instead of the whole domain can cause incorrect results. (see Known Bugs and workarounds for more information).

- The tile plot's Control menu can be used to bring up several windows. The Animate.. item displays a window that can be used to animate through the tile plot's time steps. The Zoom.. item displays a window that is used to switch between all the different Zoom regions that have been selected for this plot. The Configure.. item brings up the tile plot's configuration window, discussed in detail in the Configuring plots section of this document.

- The tile plot's Map menu is used to toggle between five different map overlays. The following maps are built into PAVE: Counties, Medium-res States, High-res States, Rivers, Roads/transport routes, and a world map. If the world map is used for a region that is not within the North American continent, you will probably need to adjust PAVE's "preclip" region using the preclip command line argument or its corresponding environment variables. Otherwise your map will be limited to lines within North America. Note that for a given map projection, PAVE may be quite slow the first time it displays a given map within a session. This is because PAVE calculates the map on-the-fly with respect to the map-projection of the dataset. Subsequent tile plots that use the same map and projection will be faster, since PAVE caches map projections in memory for future use.

- The tile plot's Plot menu can be used to create a time series line plot at the cell with the maximum or minimum value in that tile plot, over the entire time range of that tile plot.

Create 3D Mesh Plot uses data for the currently selected formula at that formula's selected geographic region of interest, layer range, and time step range, to make a 3D mesh plot.

Create Time Series Line Plot makes a time series line plot using data for the currently selected formula at that formula's selected geographic region of interest, layer range, and time step range. Each time step's data is averaged linearly to produce that time step's data point. WARNING: If multiple layers are selected, PAVE 1.4.1 will show data for only the topmost layer instead of averaging the data over the selected layers.

Create Time Series Bar Plot makes a time series bar plot using data for the currently selected formula at that formula's selected geographic region of interest, layer range, and time step range. Each time step's data is averaged linearly to produce that time step's data point.

Animate Tile Plots Synchronously.. animates all currently displayed tile plots in a round-robin fashion.

Set Minimum Frame Time.. brings up a window that allows you to adjust the minimum time between animation frames. This is useful if you have an extremely fast machine on which you

want to slow down the animation. The minimum time can be set in 0.1 second increments to any value between 0 and 5 seconds.

Set Tile Plot Cross Section Type brings up a sub-menu that can be used to choose the axis on which PAVE slices the data to make a plot. By default, PAVE does a "Z Cross Section" plot, where every data point on the plot is on the same Z layer. "X Cross Section" and "Y Cross Section" plot types can also be chosen. Subsequent plots will reflect the chosen plot type. Choose X Cross Section when you want to plot a cross section in which the x coordinate is constant, and Y Cross Section when you want the y coordinate to be constant. Note that you will have to adjust your geographic region of interest and the layer range appropriately for the plot type, as PAVE needs to have the geographic region of interest sufficiently cropped for it to be able to resolve which individual plane it is plotting. The other sub menu items are currently grayed out, as PAVE doesn't make use of them. Future versions of PAVE may have these enabled.

Default slices.. is grayed out as PAVE doesn't use this feature.

Export Menu

The export menu can be used to save data to disk for use by other programs.

AVS5 saves the currently selected formula's data to an AVS5 field file, that can then be visualized with the Application Visualization System from Advanced Visual Systems, Inc.

netCDF saves the currently selected formula's data to a Models-3 I/O API data file. Data in this format can be visualized later using PAVE, or any other program that reads netCDF/Models-3 I/O API data files. You may find it helpful to save your data this way to perform an extraction from your data, or save the result of a complex formula to prevent having to recalculate the result.

Tabbed ASCII saves the currently selected formula's data to a tab delimited data file suitable for reading into a spreadsheet application such as Excel or Lotus.

Data Explorer and Iris Explorer are grayed out, as this functionality has not yet been added to PAVE.

Help Menu

The help menu brings up Mosaic loaded with web pages to help you with your PAVE usage.

User Guide first looks on the local disk for the PAVE user guide, for <top level PAVE directory>/doc/pave/Pave.html. If this file does not exist, it attempts to connect to the user guide installed at MCNC, which is at http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.html.

Frequently Asked Questions first looks on the local disk for the PAVE FAQ, for <top level PAVE directory>/doc/pave/Pave.FAQ.html. If this file does not exist, it attempts to connect to the FAQ installed at MCNC, which is at http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.FAQ.html.

Models-3 I/O API attempts to use Mosaic to browse the Models-3 I/O API web pages located at http://www.iceis.mcnc.org/EDSS/ioapi/H.AA.html.

Please try to answer your questions using these resources before sending mail to pave@mcnc.org.

7. Configuring plots

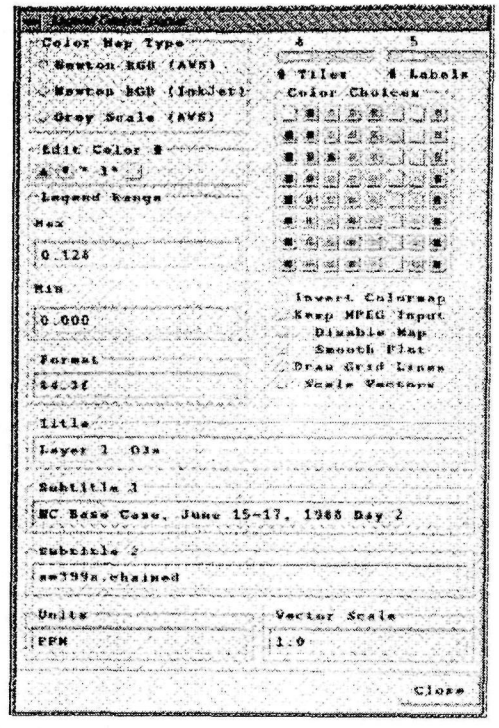
Configuring PAVE Time Series Line Plots

Many aspects of PAVE time series line plots can be configured by the user. To access the main configure menu, click on the configure button at the bottom of the time series line plot. Many users like to edit the minimum and maximum of the axes, and the spacing between and labeling of the ticks. To configure these attributes, first click on the Axis button on the configure menu. An Axis Configuration window will appear, and at the top are buttons that let you choose the axis to edit - select one of these. To set the maximum and minimum values displayed for the axis, enter the desired values in the -max and -min fields. To set the frequency of the tick labels, enter a value for the stepsize. There are also a number of other features that can be configured.

Configuring PAVE Tile Plots

Many aspects of PAVE tile plots can be configured by the user. Configuration settings can also be saved to a file for use in later PAVE plots.

To adjust the configuration of a tile plot, choose the "Control..Configure..." menu item on a tile plot to see a window similar to the following:



Note that with any of the type in widgets on this window, changes only take effect when you press the enter key in that widget.

The Color Map Type radio buttons allow you to choose from three different color maps.

The Edit Color # box allows you to individually edit any of the colors in that tile plot's legend. The arrows enable you to scroll through the legend's colors until you find the one you wish to change. You can then modify that color by clicking on one of the Color Choices color squares to the right. Alternatively, you can modify that color by clicking on the small color square in the Edit Color # box, that will then bring up a window that allows you to specify the individual RGB values for that color.

The Legend Range box allows you to specify the minimum and maximum cutoffs for the color legend used by the tile plot. Note that any cells with values higher than the maximum are plotted with the same color as the maximum, and cells with values lower than the minimum are plotted with the same color as the minimum.

The Format typing specifies a C language sprintf format that is used to print the labels on the legend. "%4.3f" causes the numbers to print like "0.128". There are many ways of printing floating point numbers using sprintf(). Please check your local sprintf() man pages or your favorite C language text if you need further information on format characters.

The Invert Colormap toggle reverses the order of the colors in the given color legend.

The Keep MPEG Input toggle allows the user to save the parameter file and the screen capture (.xwd) files used by mpeg_encode to create an MPEG animation. When this button is selected, files with the following names will remain in the same directory as any MPEGs saved from that tile plot:

[MPEG FILE NAME].param
[MPEG FILE NAME].mpg.0000.xwd
[MPEG FILE NAME].mpg.0001.xwd
:
[MPEG FILE NAME].mpg.N.xwd [where there are N+1 frames in the MPEG]

Note that the MPEG animation will be generated, regardless of the setting of the "Keep MPEG Input" button.

Parameter files are used by mpeg_encode to generate the MPEG files from a stream of XWD images saved by PAVE. If you want to edit a parameter file and then generate an MPEG "manually" using mpeg_encode, the command line format is:

mpeg_encode <parameter file name>

Using the parameter file below, /home/thorpe/UWINDa2.mpg would be generated from /home/thorpe/UWINDa2.mpg.[0000-0005].xwd.

```
IQSCALE                 6
PQSCALE                 6
BQSCALE                 6
PSEARCH_ALG             LOGARITHMIC
BSEARCH_ALG             CROSS2
GOP_SIZE                10
SLICES_PER_FRAME        1
PIXEL                   HALF
RANGE                   10
PATTERN                 IBBPBBPBBPBBPBB
FORCE_ENCODE_LAST_FRAME
BASE_FILE_FORMAT        PPM
OUTPUT                  /home/thorpe/UWINDa2.mpg
INPUT_CONVERT           /pub/storage/edss/framework/EDSSv0.x/IRIX5_mips/bin/
                        OPTIMIZE/publi
c_domain/xwdtopnm * |   /pub/storage/edss/framework/EDSSv0.x/IRIX5_mips/bin/
```

OPTIMIZE/public_ domain/pnmdepth 255
INPUT_DIR /home/thorpe
INPUT UWIND a2.mpg.*.xwd [0000-0005]
END_INPUT REFERENCE_FRAME ORIGINAL
FORCE_ENCODE_LAST_FRAME

The Disable Map toggle allows you to show the data for that plot without a map.

The Smooth Plot toggle can be used to do bilinear interpolation on that tile plot, which provides a smoothed out look at your data. This slows down the drawing of the plot because PAVE then needs to do calculations for each pixel of the plot rather than for each grid cell.

The Draw Grid Lines toggle can be used to turn on grid lines for your plot, in case this makes it easier for you to see your data's grid cell size and locations.

The Scale Vectors toggle is only relevant when the tile plot has vectors on it. By default, the vectors shown are all the same length. If you want them to be scaled by vector magnitude, turn on this toggle.

The Vector Scale type is only relevant when the tile plot has vectors on it. If you want to lengthen the vectors, use a scale greater than one. If you want to shorten them, use a scale less than one.

The rest of the items on this window are pretty much self explanatory.

Saving configuration settings for later use

Tile plot configuration settings can be saved to an ASCII file, edited if desired, and reused in future PAVE sessions. Using a tile plot's a "File...Save Configuration Settings" menu item brings up a file browser that allows you to save the current configuration of the tile plot to an ASCII file suitable for editing. The information saved includes the legend range, the format of the legend labels, the number of labels on the legend, the number of tiles/colors used, the colors themselves, and all of the individual toggle and radio buttons on the Configure.. window. In subsequent PAVE sessions, these configuration settings can be retrieved to affect future tile plots.

An example of the file format used is as follows:

ColorMapType NEWTON_COLORMAP
Legend_Max 0.128
Legend_Min 0.000

```
Legend_Format       %4.3f
Number_Labels       5
Invert_Colormap     0
Number_Tiles        8
255         0       0       ColorNumber8
255         147     0       ColorNumber7
218         255     0       ColorNumber6
70          255     0       ColorNumber5
0           255     221     ColorNumber4
0           143     255     ColorNumber3
0           95      255     ColorNumber2
223         223     223     ColorNumber1
Save_MPEG_Files     0
Disable_Map         0
Smooth_Plot         1
Draw_Grid_Lines     1
Scale_Vectors       0
```

Note the following about the format of the configuration file:

- Any of the entries can be deleted from the configuration file. PAVE uses its default setting for any entry not supplied.

- ColorMapType should be one of the following:

  NEWTON_COLORMAP [for Newton RGB (AVS)]
  JET_COLORMAP [for Newton RGB (InkJet)]
  GREY_COLORMAP [for Grey Scale (AVS)]

- The colors are denoted by values for the red, green, and blue components of each color, respectively, in a 0..255 (inclusive) scale. The fourth entry in each color line should be of the form "ColorNumberI", where I is the number of the color (1..Number_Tiles) being described on that line.

- The Number_Tiles value should exactly match the number of colors supplied.

- The Legend_Format line expects its argument to be that usable by the C language's printf() routine for formatting a float for output.

- Invert_Colormap, Save_MPEG_Files, Disable_Map, Smooth_Plot, Draw_Grid_Lines, and Scale_Vectors are boolean values - they should be set to either 0 or 1.

Configuration files can be loaded into PAVE sessions in two ways - using the user interface or from the command line. PAVE's main window now has a "File... Choose Configuration File For New Tile Plots" menu item, which can be used to select the file. Alternatively, PAVE's command line arguments or standard input stream can accept input of the form "-configFile ". Once PAVE loads a configuration file, that information will be used as the default for all subsequent PAVE plots, until another configuration file is specified to PAVE.

8. Printing and exporting images, animations, and data

PAVE provides facilities to print plots, export images and animations, and export subsets of data. In cases where PAVE's built in capabilities do not allow you to capture the images you want (e.g. saving multiple windows together), the tools xv and snapshot may allow you to accomplish your goals.

Printing plots

You probably need to set your PRINTER environment variable prior to launching PAVE for printing to work correctly. (See the Quick PAVE Jump start section for further information.)

To print tile plots, if you are printing to a black and white printer, you should probably choose "Control/Configure" from the plot's menu bar and then experiment with various settings for the colors. You may want to reduce the number of colors in the plot by choosing a smaller number of tiles, or alternatively you may want to choose to use the gray scale Colormap. Once you are ready to print the image, just choose the File..Print menu item on the tile plot, and it will be printed on the printer specified by the PRINTER environment variable. If for some reason this doesn't work, then see the sections below on using snapshot and/or xv. These tools will enable you to save a screen captured image to a PostScript file that can then be printed.

To print 3D mesh plots, first decide whether you want to print using black and white or color PostScript. You can then select either the "Black & White PostScript" or the "Color PostScript" button on the 3D mesh plot's "ANIMATING SURFACE" window. Then type in the filename, complete with path name, in the resulting window. Your PostScript will be saved to the chosen file, and then you can issue an lpr command from the UNIX command line to print that file.

To print time series line plots or scatter plots, select the "print" button on the plot's window. Then type in the filename, complete with path name, in the resulting window. Your PostScript will be saved to the chosen file, and then you can issue an lpr command from the UNIX command line to print that file.

To print time series bar plots, see the sections below on using snapshot and/or xv. These tools will enable you to save an screen captured image to a PostScript file that can then be printed.

Exporting images and animations

Tile plots can be saved as GIF, Postscript, SGI's RGB, XWD, TIFF, and several other image file formats using file menu items on the tile plot.

Tile plots can be saved to an MPEG animation using the File..Save MPEG Animation menu item on the tile plot. Several points to note regarding MPEG animations:

1. They can take a while to generate.

2. They use up a fair amount of disk space, typically 0.4MB per frame to generate, and 0.2-0.4 MB for the MPEG file for each 24 hours of animation. This varies proportionally with the size of the image you are animating. It also varies according to the number of changes in the image during animation.

3. The MPEG animations can be viewed using mpeg_play, which is distributed with PAVE as <pave installation dir> //$EDSS_ENV/bin/OPTIMIZE/public_domain/mpeg_play. Use "mpeg_play -help" for a description of its options. Here is a useful example:

   mpeg_play test.mpeg -loop -framerate 5 -quiet

   "-loop" keeps it looping. "-quiet" keeps it from printing to stdout (which makes the transition from the last frame to the first much faster). "-framerate 5" slows it down to print 5 frames per second.

To print other types of PAVE plots, see the sections below on using snapshot and/or xv. These tools will enable you to save an screen captured image to an image file that can then be printed.

Exporting PAVE data

PAVE's Export menu enables you to save the currently selected formula's data in AVS5 field format, netCDF Models 3 I/O API format, and as tabbed ASCII data suitable for use in spreadsheet applications. (Note that the currently selected formula's data is saved when using the export menu items.) Of particular interest may be the option to export netCDF data. You might find a very large dataset cumbersome to keep around when you are only interested in a small number of variables, perhaps in a small time range or spatial region. PAVE can be used to subselect the variable, time, levels and region of interest and then save this data to a smaller, more manageable dataset. Also, you can save the result of a complex formula to a netCDF file and therefore prevent having to wait for it to compute again.

Using snapshot to create PostScript files from screen images

On an SGI, type snapshot. To view the snapshot menu, place the mouse pointer over snapshot and press the right mouse button. Choose the New file name option to name the file that will be output from snapshot. You should give it a name with a ".rgb" extension. While the mouse pointer is over snapshot, press and hold down the shift key. Then move the mouse pointer to the top left corner of the part of the screen you wish to capture. Then press the left mouse button and drag the mouse pointer until you reach the bottom right corner of the section you want to capture. You should see a red box surrounding the part of the screen that will be captured. If you need to modify the size or location of the box, place the mouse pointer over snapshot and press and hold down the shift button. To move the box, place the mouse pointer inside the box and then drag the box using the middle mouse button. To resize the box, place the mouse pointer near one of the edges of the box and drag the edge (or corner) using the left mouse button.

Once you have selected the portion of the screen to be captured, use the Save as ... option on the snapshot menu to create an RGB file. To convert the RGB file to black and white PostScript, type:

tops your_file.rgb > your_file.ps

If you want a color PostScript file, type:

tops your_file.rgb -rgb > your_file.ps

You can then use xpsview your_file.ps to view the new PostScript file. You can then capture another image by placing the mouse pointer over snapshot, pressing and holding the shift button, and then creating a new box with your left mouse button.

For more information on snapshot and tops, see their man pages.

Sun also has a version of snapshot that can capture images. The images can be loaded and viewed with image tool and then saved to a variety of formats, including postscript. Both snapshot and image tool have nice user interfaces to help you through the process. See the man pages for more information.

Using xv to create PostScript files from screen images

xv is a very useful tool for capturing and manipulating images - especially GIF files. Type "xv" to start the program. To view the xv menu, place the mouse pointer over the initial window and press the right mouse button. Click on the grab button with the left mouse button, then use the

middle button to drag over the screen region you wish to save to a PostScript file. A copy of these pixels will appear in a separate window.

Now left-click the "xv controls"'s Save button. Choose PostScript from the "Format" menu, and "Full Color" or your preferred color choice from the "Colors" Menu. Next use the browser to save the PostScript to a file. It is recommended that you give the file a .ps extension, to indicate its format.

You can then use xpsview your_file.ps to view the new PostScript file.

xv is available for a variety of UNIX Platforms via anonymous ftp to ftp://xm.com/pub/xv-3.10a.tar.Z. For more information on xv, see the www page at http://www.sun.com/sunsoft/catlink/xv/xv.html

NOTE: xv is a shareware program that you will need to pay $25 to license if you decide to use it. This can be remitted to

US Mail: John Bradley
1053 Floyd Terrace
Bryn Mawr, PA 19010
FAX: (610) 520-2042

Electronic Mail regarding xv should be sent to one of these three addresses:

xv@devo.dccs.upenn.edu - general xv questions
xvbiz@devo.dccs.upenn.edu - all xv licensing questions
xvtech@devo.dccs.upenn.edu - bug reports, technical questions

9. Driving PAVE using scripts

PAVE has a large number of command line arguments. These can be typed into the PAVE standard input stream (the window where PAVE was launched), or supplied via command line arguments when PAVE is invoked. The command line argument method is often used within scripts written to automate plots for the user. Several example scripts are provided below. The format for PAVE command line arguments can be determined by typing "pave -usage" at the command line, which produces the following:

```
usage: pave
       [ -f [<host>:]<pathname/filename> ]
       [ -subdomain <xmin> <ymin> <xmax> <ymax> ]
       [ -fulldomain ]
```

[-s "<formula>"]
[-level <level>]
[-levelRange <levelMax> <levelMin>]
[-crossSectionType X|Y|Z]
[-titleString "<title string>"]
[-subTitle1 "<sub title 1 string>"]
[-subTitle2 "<sub title 2 string>"]
[-height <tile plot height in pixels>]
[-width <tile plot width in pixels>]
[-unitString "<unit string>]
[-barplotYformat "<format string>"]
[-tileYlabelsOnRight]
[-mapName "<pathname>/<mapFileName>"]
[-contourRange <minCut> <maxCut>]
[-saveImage <image type> <file name>]
[-autoContourRange]
[-configFile <configFileName>]
[-gtype <tile|line|mesh|bar>]
[-vectorTile "<formula>" "<U>" "<V>"]
[-vector "<U>" "<V>"]
[-scatter "<formula1>" "<formula2>"]
[-multitime <Nformulas> "<formula1> .. "<formulaN>"]
[-ts <time step>]
[-tinit <initial time step>]
[-tfinal <final time step>]
[-system "<UNIX command>"]
[-help|fullhelp|usage]
[-quit|-exit]
[-version]

-f [<host>:]<pathname/filename> tells PAVE to load in this dataset and make it the currently selected dataset. Each time you enter a new sequence of script commands with -f options in it, PAVE will remove any previously loaded datasets from its memory, and begin denoting the new dataset(s) with the letters a, b, etc. So if you want to load in a number of datasets using the -f option with PAVE's standard input, be sure to have all the -f pairs on one line. If you are using a script, it is fine to leave -f commands on separate lines that end with backslashes.

-subdomain <xmin> <ymin> <xmax> <ymax> sets the PAVE domain matching the currently selected dataset to the bounding box specified by its arguments. The currently selected dataset is usually the most recently added dataset, unless you have modified it by selecting another dataset using the user interface. It is often handy to type -subdomain commands into PAVE's standard

input if you are trying to select a very precise subdomain (such as that needed for a vertical cross-section plot).

-fulldomain sets the PAVE domain matching the currently selected dataset to be completely selected. The currently selected dataset is usually the most recently added dataset, unless you have modified it by selecting another dataset using the user interface.

-s "<formula>" loads the specified formula into PAVE's memory, and makes it the currently selected formula.

-level <level> sets the level range of all formulas to the single level specified.

-levelRange <levelMax> <levelMin> sets the level range of all formulas to the range specified.

-crossSectionType X|Y|Z sets the graphics cross section type for subsequent plots to the slice type specified.

-titleString "<title string>" sets the title for the next plot made to the specified title. Subsequent plots will use the default PAVE title, unless this argument is used again.

-subTitle1 and -subTitle2 allow the user to control a tile or vector plot's subtitles if desired. Subsequent plots will use the default subtitles, unless these arguments are used again.

-width and -height allow the user to control a tile plot's width and height in pixels. All subsequent plots will use the supplied value for height/width, until a non-positive value is supplied as a subsequent argument. At that point PAVE's default height/width will be used.

-unitString can be used to override the default unit label used for tile plots. The default value comes from the dataset(s) themselves.

-barplotYformat <format string> can be used to adjust the format used by sprintf() to draw the y axis labels on bar plots. The default format is %g. Please see a C reference book or the man pages on sprintf() for specifics on usable formats for float variables. NOTE: This option does not work on the IBM platform.

-tileYlabelsOnRight causes the tile plot's Y axis labels to appear on the right hand side of the plot, rather than the default of the left hand side.

-mapName "<pathname>/<mapFileName>" causes PAVE to use the supplied map name instead of the default map for tile plots. The default map is <top level PAVE

directory>/$EDSS_ENV/bin/OPTIMIZE/maps/OUTLUSAM, which is a medium resolution state outline map.

-contourRange <minCut> <maxCut> sets the contour minimum and maximum cutoffs to use for tile plots. By default, the contour range is set by the range of the data for each plot.

-saveImage <image type> <file name> saves the most recently created tile plot, The image type can be RGB, XWD, GIF, MPEG, or PS. Note: this may cause PAVE to crash if the most recently created tile plot has been closed.

This option can also be used to generate a stream of images from a tile plot, one for each time step of data associated with that tile plot. This can be done either through a command line option (-saveImage) or through the standard Motif user interface that comes with a tile plot's window. By supplying a file name with % format characters suitable by the C Language's printf() routine, PAVE is notified that a stream of images should be saved rather than a single image. The printf % format characters are used to generate the individual file name for each time step of data.

For example, if a user needs to save Gif images of variable UWIND for the first 13 hours from dataset /home/thorpe/example_data/399test.wind.bin.d1 to files /tmp/test00.gif .. /tmp/test12.gif with smoothly colored background (as opposed to tiled) with a contour range of -4000 to +4000 from a script, quitting PAVE upon completion, the following script would do the job:

```
#!/bin/csh
setenv SMOOTH_PLOTS 1
pave \
-f /home/thorpe/example_data/399test.wind.bin.d1 \
-s UWINDa \
-contourRange -4000 4000 \
-tinit 0 -tfinal 12 -gtype tile -saveImage GIF /tmp/test%02d.gif \
-quit
```

-autoContourRange tells PAVE to no longer use the contour range supplied by a previous -contourRange command, but rather use the default range that is set by the range of the data for each plot.

-configFile <configFileName> specifies a configuration file for PAVE to use for configuring subsequent tile plots. Please see the Configuring plots section for more information on configuration files.

-gtype <tile|line|mesh|bar> instructs PAVE to create a plot using the specified type and the currently selected formula's data.

-preClip <llLat> <llLon> <urLat> <urLon> will cause PAVE to use a "pre-clip" map region bounded by the supplied lat/lon coordinates. These arguments can be used to access PAVE's world map over a region outside of North America. For further information on how to use the world map, please see the PAVE FAQ located at http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.FAQ.html#WorldMap.

-vectorTile "<formula>" "<U>" "<V>" creates a vector plot with the result of "formula" as the background tiles, U as the left to right vector component, and V as the down to up vector component. Note that the formulas for the three components should have already been loaded into PAVE, and they are case sensitive.

-vector "<U>" "<V>" creates a vector plot with U as the left to right vector component and V as the down to up vector component. There are no background colors used for this type of plot. Note that the formulas for the two components should have already been loaded into PAVE, and they are case sensitive.

-scatter "<formula1>" "<formula2>" creates a scatter plot using the two formulas specified. Note that the formulas for the two components should have already been loaded into PAVE, and they are case sensitive.

-multitime <Nformulas> "<formula1> .. "<formulaN>" creates a time series line plot showing each of the Nformulas formulas with its own line. Note that Nformulas must be between 1 and 8, and that all the formulas for the plot should have already been loaded into PAVE, and they are case sensitive.

-ts <time step> sets the selected time step for each formula in PAVE's memory to the specified step number, where the first step number is denoted by 0. This is a little used option that has no effect in today's PAVE, but may be used in future versions.

-tinit <initial time step> sets the first time step for each formula's time step range to the specified step number, where the first step number is denoted by 0.

-tfinal <final time step> sets the last time step for each formula's time step range to the specified step number, where the first step number is denoted by 0.

-system "<UNIX command>" sends the specified command to the UNIX command line using C language's system() routine.

-help | -fullhelp | -usage display the information on all the command line arguments available. Each of these three versions perform the identical function.

-copyright prints out copyright information on PAVE itself and all of the third party public domain applications that it uses.

-quit | -exit ends the PAVE session.

-version prints out information about the PAVE version being used on the standard output stream, that is usually sent to the window PAVE was launched in.

Here are a few notes to keep in mind regarding the above scripting commands:

- default host will be local.
- default formula will be most recently added.
- default level will be level 1.
- default time step will be 0 (0 based).
- default initial time step will be 0 (0 based).
- default final time step will be Nsteps in file-1 (0 based).
- host can be expressed as either a file name or an ip number.
- files must be either UAM, UAM-V, or netCDF format.
- datasets (filenames) will be tagged (by the PAVE subsystem) "a", "b", ..., in the order PAVE learns of them.
- default domain will be all on.

Here is an example PAVE script:

compare.pave:

```
#!/bin/csh -f
tenv DATA_DIR /ep/otag/jul93/output
setenv SCEN1 07bas1B
setenv SCEN2 93snsDuc2
pave \
-f $DATA_DIR/$SCEN1/bin/avrg.cc.22jul93-07.mc.$SCEN1 \
-f $DATA_DIR/$SCEN2/bin/avrg.cc.22jul93-93.mc.$SCEN2 \
-configFile /ep/otag/jul93/output/base_o3.cfg \
-unitString "PPM" \
-titleString "$SCEN1 Ozone" \
-s O3a -tinit 12 -tfinal 17 -gtype tile \
-titleString "$SCEN2 Max Ozone" \
```

-s max\(O3b\) -tinit 12 -tfinal 18 -gtype tile \
-configFile /ep/otag/jul93/output/compare_o3.cfg \
-titleString "$SCEN1 Ozone - $SCEN2 Ozone" \
-s O3a-O3b -gtype tile

This script compares ozone concentrations in coarse grid average files for two scenarios and draws three plots. In the first few lines some environment variables are set to help find data in a generic fashion.

pave \
-f $DATA_DIR/$SCEN1/bin/avrg.cc.22jul93-07.mc.$SCEN1 \
-f $DATA_DIR/$SCEN2/bin/avrg.cc.22jul93-93.mc.$SCEN2 \

Here pave is started and two files are loaded (coarse grid average files for a day of interest).

-configFile /ep/otag/jul93/output/base_o3.cfg \

In the next line a configuration file is defined. This contains preferences for "base ozone plots" (i.e. plots of ozone concentrations, not differences). The contents of the file are:

base_o3.cfg:

ColorMapType	NEWTON_COLORMAP
Legend_Max	0.160
Legend_Min	0.000
Legend_Format	%4.3f
Number_Labels	7
Invert_Colormap	0
Number_Tiles	8
Save_MPEG_Files	0
Disable_Map	0
Smooth_Plot	0
Draw_Grid_Lines	0
Scale_Vectors	0

Here the standard Colormap is being used, max and min values are set, and the number of labels is changed to 7. The rest of the values are defaults - they do not need to be in the configuration file. You can create a configuration file by configuring a plot as you want it and choosing Save Configuration Settings from the file menu of the plot.

-unitString "PPM" \

```
-titleString "$SCEN1 Ozone" \
-s O3a -tinit 12 -tfinal 17 -gtype tile \
```

In these lines, setup is being done up to plot ozone for the first data set. The units and title are defined, the formula to plot is chosen, and in this case only hours 12-17 will be plotted. Finally, a tile plot is created due to the keywords "-gtype tile". NOTE: the \'s at the ends of the lines are continuation characters - don't forget these when you're using PAVE from a C-shell script!

```
-titleString "$SCEN2 Max Ozone" \
-s max\(O3b\) -tinit 12 -tfinal 18 -gtype tile \
```

In these lines a plot of the maximum ozone in dataset b is created by using the max operator. The \'s around the ()s are required because C-shell will try to interpret the ()s otherwise. Note that by using tinit and tfinal, the maximum is for those hours only, but the resulting file will contain multiple time steps with the same value for each one.

```
-configFile /ep/otag/jul93/output/compare_o3.cfg \
-titleString "$SCEN1 Ozone - $SCEN2 Ozone" \
-s O3a-O3b -gtype tile
```

In the last three lines a difference plot is drawn with different preferences as defined by a new configuration file. The rest is similar to the other plots. Here is the configuration file for the difference plot:

compare_o3.cfg:

```
ColorMapType      JET_COLORMAP
Legend_Max        0.032
Legend_Min        -0.032
Legend_Format     %4.3f
Number_Labels     9
Invert_Colormap   0
Number_Tiles      9
```

Huge example of command line usage:

Here is a huge example of invoking PAVE with command line arguments. Almost all available command line arguments are used by this example. These types of invocations typically are used from within a script that can be run over and over. Of particular note are the vectorTile, vector, scatter, and multitime options, that are only available through command line or via stdin (but NOT via the PAVE user interface).

```
pave \
-f /pub/storage/edss/framework/EDSSv0.2/example_data/399a.uam.emis.d1 \
-f /200p_scratch/olerud/399a/uamdata/399a.wind.bin.d1 \
-s "sqrt(UWINDb*UWINDb+VWINDb*VWINDb)" \
-s UWINDb \
-contourRange -3.0 3.0 \
-titleString "Ground Level U Wind Speed" \
-gtype tile \
-s VWINDb \
-contourRange -3.0 3.0 \
-titleString "Ground Level V Wind Speed" \
-gtype tile \
-autoContourRange \
-titleString "Ground Level Winds (Bckgnd=Speed)" \
-vectorTile "sqrt(UWINDb*UWINDb+VWINDb*VWINDb)" "UWINDb" "VWINDb" \
-vector "UWINDb" "VWINDb" \
-s "NO2a+NOa" \
-titleString "Ground Level NOx Emissions" \
-gtype tile \
-titleString "Ground Level NOx Emissions" \
-gtype mesh \
-titleString "Ground Level NOx Emissions" \
-gtype line \
-titleString "Ground Level NOx Emissions" \
-gtype bar \
-scatter "UWINDb" "VWINDb" \
-f /200p_scratch/rmm/eastUS_em1_g0 \
-s "NOc+NO2c" \
-contourRange 0 200 \
-subdomain 26 24 37 34 \
-level 1 \
-titleString "Ground Level NOx" \
-gtype tile \
-level 5 \
-titleString "Level 5 NOx" \
-gtype tile \
-levelRange 1 5 \
-contourRange 0 200 \
-subdomain 31 24 31 34 \
-crossSectionType X \
-titleString "First 15 Levels NOx at Col 31" \
```

```
-gtype tile \
-contourRange 0 200 \
-subdomain 26 30 37 30 \
-crossSectionType Y \
-titleString "First 15 Levels NOx at Row 30" \
-gtype tile \
-s "NOc" \
-s "NO2c" \
-multitime 2 NOc NO2c
```

10. Optional Environment Variables

The following environment variable settings can be made prior to invoking PAVE. They are completely optional, and most of them can also be set from within PAVE's user interface. You may find these useful if you frequently want smooth plots, grid lines, no maps, etc.

setenv PAVE_EXE <some pathname/executable name> will cause the pave script to use that version of the binary executable when launching PAVE.

setenv DISABLE_MAPS 1 will cause PAVE to disable map drawing by default on tile plots.

setenv DRAW_GRID_LINES 1 will cause grid lines to be drawn by default on tile plots.

setenv PRINTER 200p_lw will cause tile plots to be sent to 200p_lw when printed.

setenv SBUS_EXEC_RC <rc file name> can be used to override the default file ~/.edss_exec_rc, which is used to locate various EDSS subsystems.

setenv SBUS_RLOGIN_RC <rc file name> can be used to override the default file ~/.edss_rlogin_rc, which is used for the user IDs to use for remote data access. You will only need to worry about this if your remote account's login name is not the same as your local login name.

setenv SCALE_VECTORS 1 will cause vector length to be scaled by their magnitude by default on vector plots.

setenv SMOOTH_PLOTS 1 will cause PAVE to smoothly interpolate pixels on tile plots by default.

setenv TENTHS_SECS_BETWEEN_FRAMES 10 will cause a minimum of 1 wall clock second between animation frames. Units are tenths of a second, allowable range is 0..50.

setenv PRECLIP_LLLAT <0>

setenv PRECLIP_LLLON <-180>

setenv PRECLIP_URLAT <85>

setenv PRECLIP_URLON <70> will cause PAVE to use a "pre-clip" map region bounded by the lat/lon coordinate pairs (-180,0) and (70,85). These environment variables can be used to access PAVE's world map over a region outside of North America. For further information on how to use the world map, please see the PAVE FAQ located at http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.FAQ.html#WorldMap.

setenv EDSS_MAPDIR <directory_name> will override PAVE's <top level PAVE directory>/$EDSS_ENV/bin/OPTIMIZE/maps dir, which is where PAVE looks for its maps by default.

setenv [P_ALP|P_BET|P_GAM|XCENT|YCENT|XORIG|YORIG|XCELL|YCELL] <some real number> can be used to supersede any of the respective netCDF header values. Before creating a PAVE plot, the environment variables will be checked and appropriate fields replaced.

setenv DENOMINATOR_CUTOFF <some floating point number> will cause PAVE to work around divide by zero conditions. If this environment variable is set, then for each division, if the denominator is less than or equal to DENOMINATOR_CUTOFF's value, then the result of the divide is set to 0

11. Requirements for use

You must have access to a Sun running Solaris 2.x, an SGI running IRIX 5.x, a DEC Alpha running OSF1, an IBM RS6000 running AIX 4.x, or an HP running HP-UX 9.x with PAVE installed on it.

You must have your input datasets in Models 3 I/O API (netCDF), UAM-IV, or UAM-V formatted data files.

PAVE can be displayed on most X displays with at least 8 bits of color. Using X windows software on a PC or Macintosh, you should be able to display PAVE output from any of the above UNIX platforms.

(OPTIONAL - READ THE REST OF THIS SECTION THIS ONLY IF YOU WANT TO ACCESS REMOTE DATA)

PAVE uses several optional files in your home directory when starting up, and writes over them with each PAVE session termination. These are used to maintain a snapshot of the current formulas and datasets being used within PAVE:

```
~/.edss_rlogin_rc      # you may need to create if using remote data
~/.edss_exec_rc        # PAVE will create if not already there
~/.pave_history_rc     # PAVE will set this up automatically
~/.pave.AA.cases       # PAVE will set this up automatically
~/.pave.AA.formula     # PAVE will set this up automatically
```

The only one that really matters is ~/.edss_rlogin_rc, which is used for setting up remote data accessibility. Here are the contents of an example ~/.edss_rlogin_rc file, which you can borrow from to create your own. Also note the instructions for setting up remote .rhosts files and paths to the visd and busd daemons.

```
# ~/.edss_rlogin_rc
#
# ----------------------------------------
# If you want to use PAVE to read remote data:
# ----------------------------------------
# This file will need to be used by PAVE to launch a
# daemon on any remote machine(s). The daemon(s) actually
# read the data and ship it back to your local machine.
#
# Copy this file to *your* ~/.edss_rlogin_rc and modify
# it appropriately
#
# On each machine you will need to set up a ~/.rhosts
# file that allows THIS machine to rsh to it. On most machines
# a .rhosts file is a list of machine and login name pairs
# found in your home directory (e.g. sirrocco smith). Make sure
# that this file is readable only by you for security reasons.
#
# Suppose you are running PAVE on sirocco, and you want to use it
# to read data that is sitting on sequoia. Test to see that your
# .rhosts file on sequoia works by making sure the following
# commands on sirocco:
#
# rsh sequoia.nesc.epa.gov -l <YOUR sequoia USERID> which visd
# rsh sequoia.nesc.epa.gov -l <YOUR sequoia USERID> which busd
#
```

```
# successfully execute and tell you the paths to the visd and busd
# daemons on the remote sirocco machine. The visd and busd daemons
# are in <pave installation dir>/<platform type>/bin/OPTIMIZE/
# for each platform type; on remote machines you may just wish to
# copy them to your home directory if it makes it easier for you.
# These daemons are used to read remote data; until they are in your
# remote path, you won't be able to read any data on that remote
# machine.
#
# Any line in this file with a # in it is ignored.
#
#########################################################################
# $HOME/.edss_rlogin_rc file contains the userids for remote machine
# names. If remote userid is same as local, you don't need to list it.
# Lines preceded by a '#' are ignored.
# The format of the contents in the file are:
# <machine-name><space><userid>
# The machine name can be the entire name or without the domain name
# (e.g., nox, rain)
#########################################################################
#mary.jane.doe doe
sequoia.nesc.epa.gov tsr
t90.ncsc.org demo_t90
```

The other files will be set up for you automatically whenever PAVE executes.

12. Quick PAVE Jumpstart

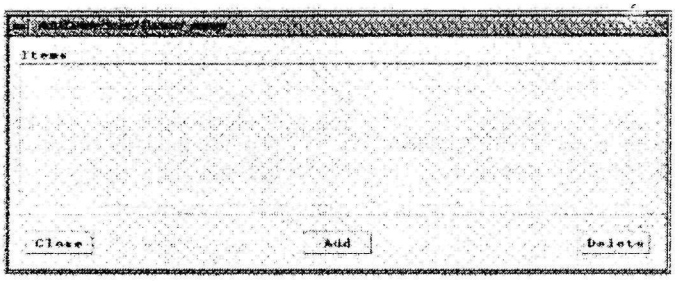
This section can be used to guide you through an example PAVE session. If you are a new user, it can help you become familiar with most of PAVE's features. Once you have satisfied the Requirements for use section above, then:

- Login to the machine where PAVE is installed, then setenv your DISPLAY environment variable to wherever you are sitting. (as in "setenv DISPLAY mymachine.wherever.gov:0"). If you will be printing tile plots directly from PAVE, you may also want to set your PRINTER environment variable appropriately. For example, if you "setenv PRINTER 200p_qms", then PAVE will use "lpr -P200p_qms" to print tile plots. Otherwise PAVE will use just "lpr", which defaults to the printer "lp", and may or may not be accessible to your system.

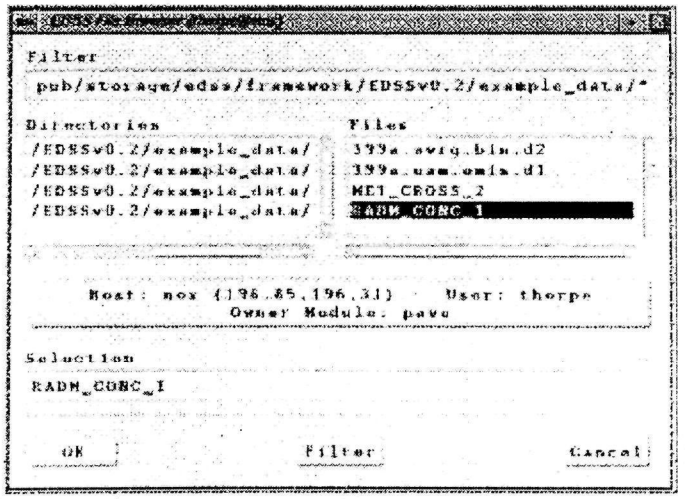
- Launch PAVE from the command line by typing "pave" at the UNIX prompt. You will probably need to make sure the scripts subdirectory of the top level PAVE installation directory into your path in order for your shell to be able to resolve the pave location. The README file that comes with PAVE has information on how to do this.

  Typing "pave" launches a "wrapper script", which in turn fires up the software bus and the PAVE executable itself. The PAVE executable itself should never be launched directly from the command line, as this wrapper script is required to set up PAVE's environment.

- You should see an "Add/Delete/Select Dataset" pop-up window:



Click on its "Add" button, and a file browser should appear:



The file browser has a "Filter" widget at the top. You can see the files in a particular directory by typing or pasting in the directory name and adding a "/*" at the end and hitting return. Alternatively, you can navigate through the directory structure by double clicking on directories in the Directories list (or single clicking and hitting return). Once

you get to the directory that contains the file you want, you can select the file by clicking on it in the Files list.

Use the file browser to add the file

"<top level PAVE directory>/example_data/RADM_CONC_1"

(If RADM_CONC_1 is not available, choose any dataset in the directory.) This will become PAVE's dataset "a", and this will be the currently selected dataset. Do the same with

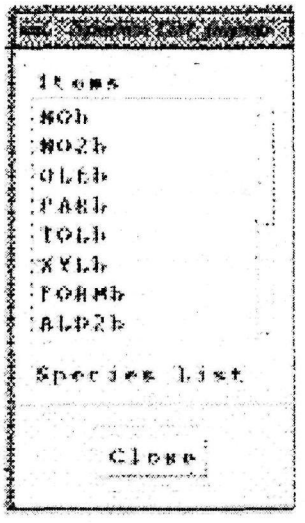
"<top level PAVE directory>/example_data/399a.uam.emis.d1"

which will become dataset b.

There may be several other example datasets in <top level PAVE directory>/example_data that you may want to play around with, although some may be missing due to disk space considerations:

```
-rw-r--r--  1 thorpe  edss    8549788 Aug 17 09:26 399a.avrg.bin.d2
-rw-r--r--  1 thorpe  edss    1070068 Aug 17 09:27 399a.uam.emis.d1
-rw-r--r--  1 thorpe  edss   11870588 Aug 17 09:29 MET_CROSS_2
-rw-r--r--  1 thorpe  edss   11870584 Aug 17 09:30 RADM_CONC_1
-rw-rw-r--  1 thorpe  edss    7871036 Aug 17 09:32 bot_west_chem_ec1_g0
-rw-r--r--  1 thorpe  edss     201048 Aug 17 09:32 exact1
```

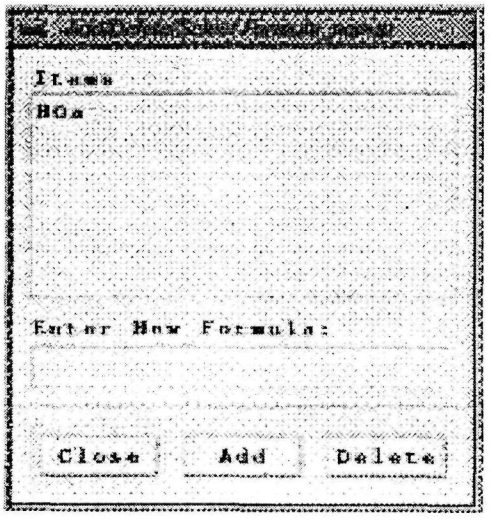
- After adding the dataset, you should see a "Species List" window:



This should now be displaying the species available in dataset b, the currently selected dataset.

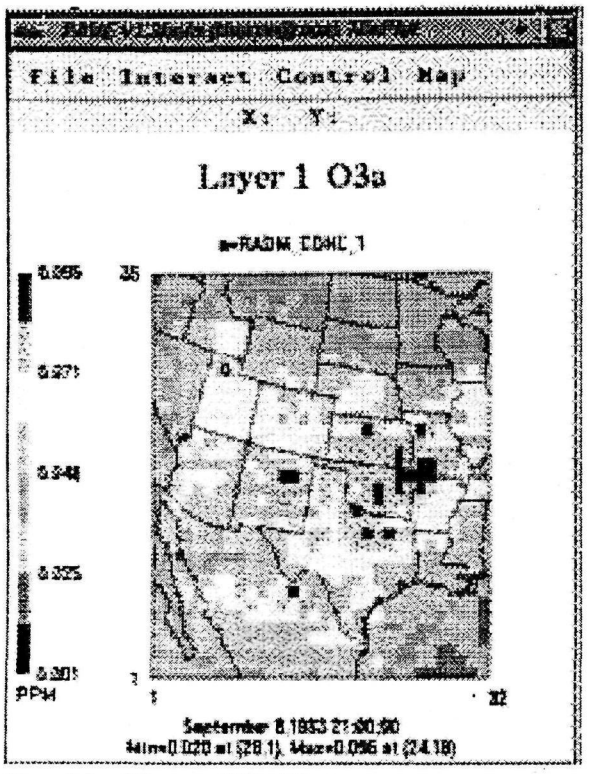
- Go back to the "Add/Delete/Select Dataset" window and "Select" the first dataset (denoted by "a") by clicking on it. This now becomes the currently selected dataset, and you should now see that dataset's available variables in the "Species List" pop up. Click on "HOa" to add that variable to the available formulas list.

- You should see an "Add/Delete/Select Formula" window:

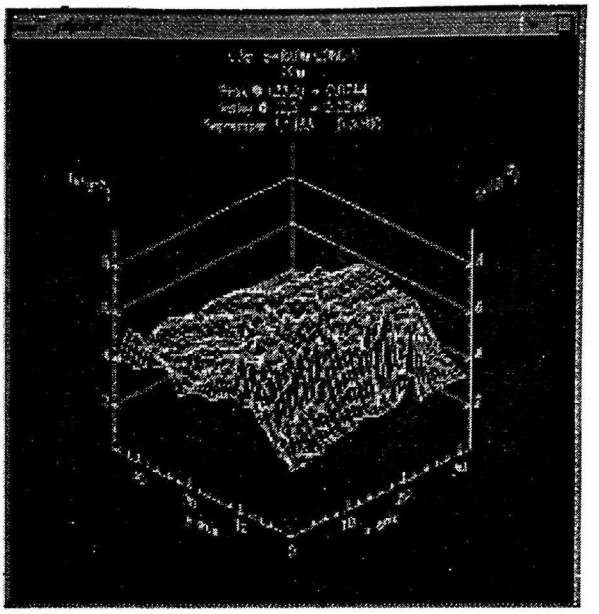


Add item "HOa+HO2a" in the "Enter New Formula:" type in box and click on "Add". Add "NOb" and "O3a" as well. You now have a total of four formulas added to your formula list, and O3a is the currently selected formula.

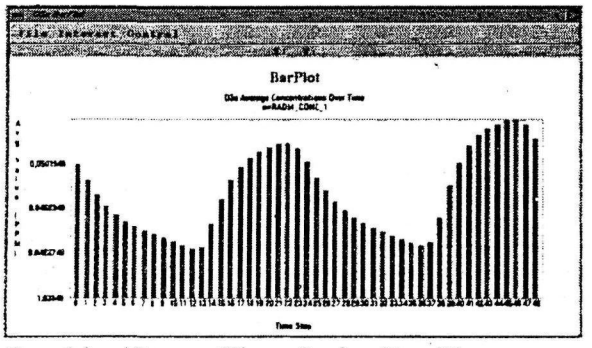
- When the PAVE window is up, select the following sequences of plots from the menu bar at the top of the PAVE window:



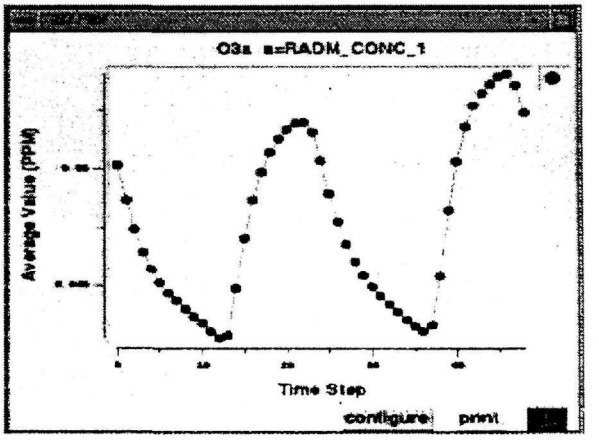
Graphics/Create Tile Plot



Graphics/Create 3d Mesh Plot



Graphics/Create Time Series Bar Plot



Graphics/Create Time Series Plot

NOTE: All the plots above are associated with the selected formula, which is probably O3a. The selected formula can be modified with the "Formulas menu/ Edit/Select from Formula List" window.

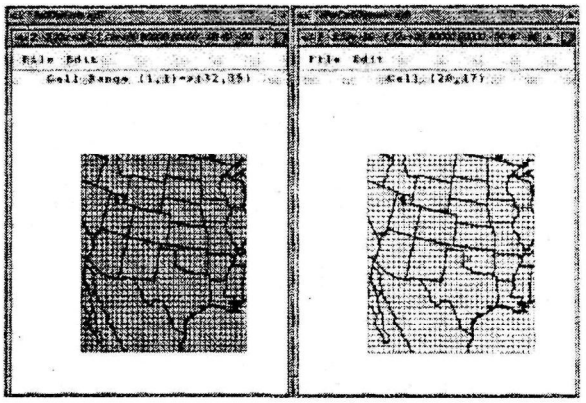
The tile plot and mesh plot show data for the selected region at a particular time step, and can be animated to show other time steps. The time series line and bar plots show data

averaged over the selected region at each time step. Here are some things to try with the plots:

- If you want to adjust a tile plot's legend min/max cutoffs, choose "Control/Configure" from the plot's menu bar.

- If you want to animate a tile plot over time, select the "Control/Animate" menu item on a tile plot. This brings up a window from which you can start the animation.

- If you want to animate all existing tile plots synchronously, select the "Graphics/Animate Tile Plots Synchronously" menu item from the main PAVE window. This presents a window that you can use to start the animations.



- When using a slider (such as in the Animate window for Tile Plots), to scroll by one time step in either direction, click on the slider itself and then use the left and right arrow buttons on the keyboard.

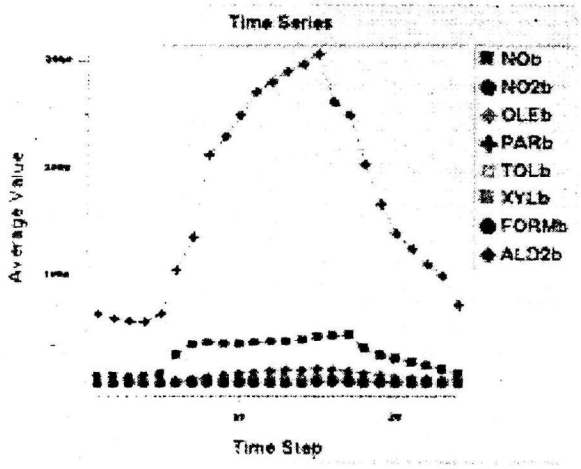
- Select "Formulas/Select Region of Interest Matching Current Formula" from the menu bar. The domain window associated with the selected formula will be displayed.

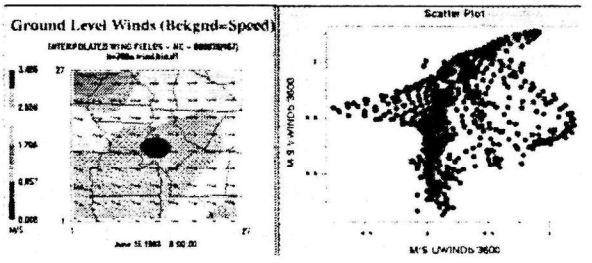
By default, the entire domain is selected for any dataset/formula loaded into PAVE. This is indicated by a light blue background on all of the cells in the domain window. You may want to do a time series of a single cell, or perhaps a tile plot of a smaller region of cells. To subselect a region, just left-click-and-drag over any blue areas to deselect cells, and conversely over any white areas to select cells. Also, you can use the Edit menu's "Select All" and "Select None" items to turn all the cells in the domain on or off. Subsequent plots will show data only from the subselected region.

This window can be closed using the File menu's "Close" item. Also note the File menu's "Save Domain To File" and "Load Domain From File" menu items that allow you to save and later retrieve frequently used domain subselections.

- How to add a remote dataset: If you have followed the optional instructions under point 3. of the Requirements to use PAVE section of this document, you can add remote datasets using the file browser provided to choose datasets. Select "Datasets/Edit/Select from Dataset List" from the menu bar. An Add/Delete/Select Dataset window will appear. Click on the "Add" button, and an EDSS file browser should appear (this may take a couple seconds). Now click on the large button in the center of the file browser that contains "Host:", "User:", and "Owner Module:" information. This brings up a "host selector" window in which you can enter the remote host name (e.g. sequoia.nesc.epa.gov). Clicking on Select should then enable you to browse for files on the remote machine, starting from your home directory.

- NOTE: If you want to do a vector plot, a scatter plot of two formulas, or a time series line plot with multiple (2-8) variables plotted on it, then you must send commands to PAVE via stdin or command line arguments, which are described in the "Driving PAVE using scripts" section of this document. Unfortunately there are currently no graphical user interface methods to access these features. There are examples of invoking PAVE with command line arguments in the same section.





- Explore the menu items available to you from tile plot windows.

  – The background map can be toggled between county boundaries, medium (the default map) or high resolution state boundaries, rivers, or roads. Tile plots can

be printed, or saved as PostScript, XWD, RGB, or Gif images. You can zoom in on, probe the data values in, and generate time series plots for specific regions of tile plots.

- To view a vertical cross section of data using the Tile Plot, first click on the dataset for which you would like to view the cross section. Then under the datasets menu, choose "Select Layer Ranges Matching Current Dataset".

- You should see a window with sliders for the highest and lowest layers to use. Set these appropriately depending on which layers you want to see.

- To see a vertical cross section, the lowest and highest should be different layers (as opposed to just plotting a horizontal cross section - in that case the min and max layers are set to the same value). Note that you can also select a layer range for a formula using the corresponding options under the Formulas menu.

  After selecting the layers you want to see, type a statement of the following form into PAVE's standard input:

  -subdomain 80 20 80 90

  Here the arguments to -subdomain are: x1 y1 x2 y2 to define the region to be selected. This will cause the data where x = 80, and y is between 20 and 90 to be selected. To see this region on the screen (and/or to set this region using the User Interface rather than a command line option), you can choose "Select Regions of Interest Matching Current Dataset". A window showing the domain will come up. Alternatively, you can select the cells with the mouse, but this gets difficult when grid cells are small and you want to select a very precise area.

  Now that the layers and region to be plotted are selected, you need to set the cross section type. Under the Graphics menu, choose "Set Tile Plot Cross Section Type" and you will see a submenu of X, Y, or Z cross sections. To make a plot for the above example choose "X Cross Section" (remember this by noting that x is constant). Next, draw the plot for a formula using that dataset by choosing "Create Tile Plot" from the Graphics menu, and you will see the plot of that cross section.

- Explore the rest of the menu items not documented above & enjoy!

13. Known bugs and workarounds

If you use the tile plot Control..Time Series menu item to generate a time series plot from a tile plot that was generated by first selecting a subdomain, PAVE may not show correct data. Work around: Use this menu item only from plots that are not on a subdomain - zoom in on the subdomain instead and then generate the plot; or select the subdomain for the time series plot you want and choose Create Time Series Line Plot from the Graphics menu to create the plot.

Creating a time series plot from the Graphics menu when there are multiple layers of data selected generates a plot for the top data layer only, instead of for the data averaged over the selected layers.

PAVE sometimes gives a message like "Can't allocate ramp color 214" fprintf()'ed to stderr and runs out of colors to use. Work around: exit PAVE, exit out of any other applications running on your X-display that are using lots of colors (AVS, drawing/painting programs, etc), and then get back into PAVE. If you were running netscape, try restarting it with the option -ncols 64 to limit the number of colors it uses. Hopefully future versions of PAVE will address this problem in a more robust fashion.

Mesh plots don't display to a DEC 5000 X-Server (possibly all 8-bit servers?).

Occasionally when saving tile plot images to data files, the menus "disappear" from the tile plot. They are still active however they are "invisible". Work around: iconify the tile plot, de-iconify it, and you should be able to see the menus again.

Printing tile plots may not work reliably on the Sun platform at this time. A work- around is to save your images as Gif or RGB files, convert them to PostScript (see the Saving PAVE Plots to PostScript section for more information on this), then print them.

The new maps in PAVE are a bit slow to generate, especially the first time you create a plot with a given map projection. (Speed up of the map generation process is a consideration for future versions of PAVE.)

If you choose "File/Exit" and PAVE does not shut down, there may be a mesh plot or time series line plot still running that prevents PAVE from exiting. Work around: locate any line or mesh plot windows that you have created, and quit out of them before exiting PAVE.

Miscellaneous memory leaks. Work around: If you run PAVE for a large number of plots and you notice severe performance degradation, exit out of PAVE and then restart it.

In general, PAVE needs a lot more QA.

14. Run Time Errors

Malloc failure: If PAVE prints or pops up a message of the form "malloc failure" or "VIS_DATA_Dup() failed" this means that PAVE has run out of usable memory while trying to perform the requested operation. To make more memory available, try closing some plots or quit and restart PAVE.

Can't allocate ramp color xxx: If PAVE prints a message about not being able to allocate a color that means that all available colors for the current session on your system have been used up. Applications like netscape will use as many colors as possible. To keep more colors available to applications like PAVE, try running netscape with the option -ncols 64 to prevent it from getting all the colors. This problem may also occur if you've selected a lot of different color schemes during your current PAVE session. To get all colors in the desired color scheme try quitting and restarting PAVE after you've chosen the scheme that you like. Also, logging out of your current session and logging back in should make the maximum number of colors available.

15. History of new features

New Features in PAVE v1.4.1

PAVE now can display a world map for data which falls outside the North America region. For further information on how to use the world map, please see the PAVE FAQ located at . http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.FAQ.html#WorldMap.

New Features in PAVE v1.4

- The following operators have been added to PAVE's formula parser:

Op	Name	Function
a < b	Less than	Returns 1 if a < b, else 0
a <= b	Less than or equal	Returns 1 if a <= b, else 0
a > b	Greater than	Returns 1 if a > b, else 0
a >= b	Greater than or equal	Returns 1 if a >= b, else 0
a == b	Equal	Returns 1 if a equals b, else 0
a != b	Not equal	Returns 1 if a does not equal b, else 0
a && b	And	Returns 1 if a != 0 and b != 0, else 0
a || b	Or	Returns 1 if a != 0 or b != 0, else 0

Please note that PAVE's operator precedence (highest to lowest) is as follows. If you wish to override the precedence given below, or are uncertain as to which operator will

take precedence, you can feel free to use parentheses in your formulas. This will force expressions within the parentheses to be evaluated first.

Highest Precedence	1) abs, log, sqr, sqrt, exp, ln, sin, cos, tan, sind, cosd, tand, minx, miny, minz, max, maxy, maxz, mean, min, max, sum, mint, maxt
	2) **
	3) /, *
	4) +, -
	5) <, <=, > >=
	6) ==, !=
	7) &&
Lowest Precedence	8) ||

- Variables from UAM and netCDF formatted data files can now be mixed within the same formula, and a suitable map will be produced, provided the variables lie on the same domain and have the same map projection. In addition, minor discrepancies in the precision of the map information associated with the variables will not prevent PAVE from drawing a map in the plots.

- PAVE now "plots" formulas that result in a single number (for example, "mean(O3a)") by sending that single number to the text window, along with information about the time and domain currently selected.

- The data is now labeled much more clearly on all time series plots. The annotations include information on the spatial and temporal subdomain used to get the data for the plot, as well as the datasets used.

- There is now a black frame drawn around a tile plot's color legend. This makes the legend much more readable when white is chosen as one of the colors.

- PAVE now displays "PAVE by MCNC" in tiny letters on tile plots and bar plots.

- There are now several new command line arguments that can be given to PAVE. They are:

  - barplotYformat <format string>
  - tileYlabelsOnRight
  - subTitle1 "<sub title 1 string>"
  - subTitle2 "<sub title 2 string>"
  - height <tile plot height in pixels>

- width <tile plot width in pixels>
- copyright
- system "<System Command>"

<format string> is a format character string that will be used by sprintf() to draw the y axis labels on bar plots. The default format is %g. NOTE: This option does not work on the IBM platform.

The -tileYlabelsOnRight causes the tile plot's Y axis labels to appear on the right hand side of the plot, rather than the default of the left hand side.

The -width and -height command line arguments have been added to PAVE, to allow the user to control a tile plot's width and height in pixels. All subsequent plots will use the supplied value for height/width, until a non-positive value is supplied as a subsequent argument. At that point PAVE's default height/width will be used.

The -subTitle1 and -subTitle2 arguments allow the user to control a tile or vector plot's subtitles if desired.

The -copyright option prints out copyright information on PAVE itself and all of the third party public domain applications that it uses.

The -system option allows the user to supply a command to be fed to the UNIX command line using a call to the system() library routine.

- There are several user interface changes in this version.

The "close" and "quit" menu items have been removed from the Motif menu accessed from the upper left hand corner of a tile plot window or a time series bar plot window.

A Tile Plots "Configure..." window has been reorganized to be smaller, in order to fit on smaller displays.

Several other minor user interface improvements have been made.

- There are several bug fixes in this version.

A bug with colors and tile plot configuration files has been resolved. Users can now select a configuration file with color information in it, make plots, and then select a configuration file without color information, and the original Colormaps will correctly be used.

Closing a tile plot while it is animating no longer causes PAVE to crash.

files with greater than one layer no longer cause PAVE to crash.

Double clicking on the upper left hand corner of the main PAVE window, a tile plot window, or time series bar plot window will now have no effect . Previously, this caused PAVE to immediately crash.

nan (bogus not-a-number) data in a tile plot no longer causes PAVE to crash. Instead, a warning message is printed to the stderr stream, and no color is plotted in that part of the image.

PAVE no longer kills all processes in its process group when exiting. This means that parent processes (such as VisDriver or the EDSS Console) will remain alive when PAVE exits. However it also means that any mesh plots will prevent PAVE's completion, until they are quit out of individually.

Several other minor bugs have been fixed.

New Features in PAVE v1.4 beta

Version 1.4 beta fixes an X-Window related bug that caused PAVE to crash occasionally with certain selected domains of interest.

New Features in PAVE v1.4 alpha

There are a number of new features in this version that you may find useful, including:

- There is now a Help menu in PAVE, from which the user can bring up the PAVE User Guide, the PAVE FAQ, and the Models-3 I/O API Web pages using Mosaic.

- PAVE makes better use of memory. Now, if you close a window, PAVE releases all memory associated with that window. Thus, if you see error messages like "malloc failed" or "VIS_DATA_Dup failed" you can try closing windows to free up memory. Deleting formulas and datasets will also cause some memory to be freed (and reduce PAVE's start up time). In addition to freeing memory upon closing windows, PAVE uses memory more efficiently in general.

- PAVE now allows the user to edit individual colors used in tile plots, using a Color Chooser window that enables setting of the individual red, green, and blue components on a 0..255 scale using sliders. This window is accessible from a tile plot's

"Control...Configure.." window, by clicking on the color shown in the "Edit Color #" box. For ideas on colors to use, see the file

/usr/lib/X11/rgb.txt

for non-Suns or

/usr/openwin/lib/X11/rgb.txt

on Suns.

- Vector plots now work with vertical cross sections.

- Vector plots with no tile backgrounds no longer show a color legend.

- There are now two additional titles that are user controllable on the PAVE tile plots. These can be changed using a tile plot's "Control...Configure.." window. They are called "Subtitle 1" and "Subtitle 2". Subtitle 1 is usually initialized to an empty string (and therefore not shown on the plot), but if a UAM data file is used to create the plot, then the file_id character string will be used. Subtitle 2 is initialized to a list of datasets used to make that plot.

- There is now a vector scale that is user controllable. Using a vector plot's "Control...Configure.." window, the user can select the "Scale Vectors" radio button to scale them by vector magnitude, as opposed to the default which is to use a unit vector length for all vectors. Once "Scale Vectors" is selected, then the value contained in the "Vector Scale" type in widget will be used to scale the length. The default vector scale is 1. A Tile Plot's File menu now has the following menu items: "Time Series At Max Point" and "Time Series At Min Point", which find the location of the min or max value in that tile plot's data, and plots a time series of the data at that point.

- A Tile Plot's Interact menu now has a "Time Series" menu item. This interact mode causes a time series to be generated for a tile plot's data over a point or region that is dragged or clicked on.

- The PAVE formula parser now accepts formulas to calculate the change in a variable per time step, using the syntax d[<variable name>]/dt. For example, d[O3a]/dt would calculate the change in concentration of ozone from dataset A per time step. Given dataset A with N time steps, this formula will actually give you only N-1 time steps. This is because each hour I's calculation is computed using O3a:I+1 - O3a:I Note that only "atomic" variables may be used in formulas using this operator. You cannot make such

formulas any more complex than the above example, i.e., the following would be ILLEGAL: d[O3a+O3a]/dt and d[O3a]/dt+O3a.

- PAVE's min and max operators have been modified to calculate the minimum/maximum value over the selected time period at each cell. For example, if the user has the formula max(O3a), then each cell's data for that formula would be the maximum value of ozone occurring during the time period currently selected.

- While MPEG frames are being generated and written to disk, the user does not have control of PAVE. However, MPEG encoding is now done by a background process, in order to return PAVE's control to the user as soon as possible. When the MPEG file has been created, the user will be notified by a beep from their terminal.

- If the user wishes to save the parameter file used by mpeg_encode to create an MPEG animation, there is an option under a tile plot's Control...Configure window to enable this. Checking on the "Keep MPEG Input" button will cause the following files to remain in the same directory as any MPEGs saved from that tile plot:

  [MPEG FILE NAME].param
  [MPEG FILE NAME].mpg.0000.xwd
  [MPEG FILE NAME].mpg.0001.xwd
  :
  [MPEG FILE NAME].mpg.N.xwd [where there are N+1 frames in the MPEG]

  Note that the MPEG animation will be generated regardless of the setting of the "Keep MPEG Input" button.

  Parameter files are used by mpeg_encode to generate the MPEG files from a stream of XWD images saved by PAVE. If you want to generate an MPEG file "manually" using mpeg_encode, the command line format is:

  mpeg_encode

  For the parameter file below, /home/thorpe/UWINDa2.mpg would be generated from home/thorpe/UWINDa2.mpg.[0000-0005].xwd.

```
IQSCALE         6
PQSCALE         6
BQSCALE         6
PSEARCH_ALG     LOGARITHMIC
BSEARCH_ALG     CROSS2
```

```
GOP_SIZE                10
SLICES_PER_FRAME        1
PIXEL                   HALF
RANGE                   10
PATTERN                 IBBPBBPBBPBBPBB
FORCE_ENCODE_LAST_FRAME
BASE_FILE_FORMAT        PPM
OUTPUT                  /home/thorpe/UWINDa2.mpg
INPUT_CONVERT           /pub/storage/edss/framework/EDSSv0.x/IRIX5_mips/bin/O
                        PTIMIZE/public_domain/xwdtopnm * |
                        /pub/storage/edss/framework/EDSSv0.x/IRIX5_mips/bin/O
                        PTIMIZE/public_domain/pnmdepth 255
INPUT_DIR               /home/thorpe
INPUT
UWINDa2.mpg.*.xwd       [0000-0005]
END_INPUT
REFERENCE_FRAME ORIGINAL
FORCE_ENCODE_LAST_FRAME
```

- Tile plot configuration settings can now be saved to an ASCII file, edited if desired, and reused in future PAVE sessions. PAVE tile plots now have a "File...Save Configuration Settings" menu item that brings up a browser to save the current configuration of the tile plot to an ASCII file suitable for editing and modification by the end user. Information saved includes the colors, number of tiles, number of labels, legend range, legend format, and all of the individual radio buttons on the Configure.. window. In subsequent PAVE sessions, these configuration settings can be retrieved to affect future tile plots.

  An example of the file format used is as follows:

```
ColorMapType      NEWTON_COLORMAP
Legend_Max        0.128
Legend_Min        0.000
Legend_Format     %4.3f
Number_Labels     5
Invert_Colormap   0
Number_Tiles      8
255       0       0       ColorNumber8
255       147     0       ColorNumber7
218       255     0       ColorNumber6
70        255     0       ColorNumber5
0         255     221     ColorNumber4
```

```
0               143     255     ColorNumber3
0               95      255     ColorNumber2
223             223     223     ColorNumber1
Save_MPEG_Files 0
Disable_Map     0
Smooth_Plot     1
Draw_Grid_Lines 1
Scale_Vectors   0
```

Note the following about the format of the configuration file:

- Any of the entries can be deleted from the configuration file. PAVE would then use its default setting for any entry not supplied.

- ColorMapType should be one of the following:

  NEWTON_COLORMAP[for Newton RGB (AVS)]
  JET_COLORMAP [for Newton RGB (InkJet)]
  GREY_COLORMAP [for Grey Scale (AVS)]

- The colors are denoted by values for the red, green, and blue components of each color, respectively, in a 0..255 (inclusive) scale. The fourth entry in each color line should be of the form "ColorNumberI", where I is the number of the color (1..Number_Tiles) being described on that line.

- The Number_Tiles value should exactly match the number of colors supplied.

- The Legend_Format line expects its argument to be that usable by the C language's printf() routine for formatting a float for output.

- Invert_Colormap, Save_MPEG_Files, Disable_Map, Smooth_Plot, Draw_Grid_Lines, and Scale_Vectors are boolean values - they should be set to either 0 or 1.

Configuration files can be loaded into PAVE sessions in two ways - using the user interface or from the command line. PAVE's main window now has a "File... Choose Configuration File For New Tile Plots" menu item that can be used to select the file. Alternatively, PAVE's command line arguments or standard input stream can accept input of the form "-configFile ". Once PAVE loads a configuration file, that information will be used as the default for all subsequent PAVE plots, until another configuration file is specified to PAVE.

- The "Formulas...Edit/Select From Formula List" window has been modified somewhat to facilitate easier additions of a number of formulas. From left to right, the buttons on now read "Add", "Close", and "Delete". The first formula typed in by a user should be followed by a click on the "Add" button. Once that occurs, the default button will be "Add". Each subsequent formula need only be followed by a carriage return, which will then perform the same function as clicking on the "Add" button.

- PAVE now can generate a stream of images from a tile plot, one for each time step of data associated with that tile plot. This can be done either through a command line option (-saveImage) or through the standard Motif user interface that comes with a tile plot's window. By supplying a file name with % format characters suitable by the C Language's printf() routine, PAVE is notified that a stream of images should be saved rather than a single image. The printf % format characters are used to generate the individual file name for each time step of data.

  For example, if a user needs to save Gif image for the first 13 hours saved to files /tmp/test00.gif .. /tmp/test12.gif of variable UWIND from dataset /home/thorpe/example_data/399test.wind.bin.d1 with smoothly colored background (as opposed to tiled) with a contour range of -4000 to +4000 from a script, quitting PAVE upon completion, the following script would do the job:

```
#!/bin/csh
setenv SMOOTH_PLOTS 1
pave \
-f /home/thorpe/example_data/399test.wind.bin.d1 \
-s UWINDa \
-contourRange -4000 4000 \
-tinit 0 -tfinal 12 -gtype tile -saveImage GIF /tmp/test%02d.gif \
-quit
```

- Several small changes have been made to make the user interface easier to use. For example, reselecting a menu item that causes a window to appear will bring the window to the front, even if it is buried.

- Several bugs that caused crashes have been fixed.

New Features in PAVE v1.3

There are a number of new features in this version that you may find useful, including:

- User controllable unit labels.

- The default color map for tile plots has been improved.
- MPEG animations can now be saved under Solaris 2.5.
- The IBM version has been upgraded from AIX3 to AIX4.
- Several bugs that caused crashes have been fixed.

New Features in PAVE v1.3beta

- Saving RGB, PostScript, XWD, and Gif images.
- Saving MPEG animations.
- On-the-fly map projections, which enables any netCDF dataset conforming to the Models-3 I/O API standard to be handled by PAVE.
- Several maps can be toggled between, including low and high resolution state boundaries, rivers, roads, and county boundaries.
- Reading datasets that are spread across multiple files.
- Exportation of netCDF Models-3 I/O API data files.
- Saving/retrieval of selected geographic domains.
- Select All/Select None features in the region of interest selection boxes.
- Printing directly from a PAVE tile plot.
- Multiple time series (1-8) can now be overlaid on the same plot.
- The min/max for the displayed data is shown on tile plots.
- Command line arguments for many of the above features.

16. Future PAVE Development

We would love to add a number of features to PAVE, but unfortunately at the current time we are out of funds to use for PAVE development. If you or someone you know is interested in some of these features and has ideas about how to obtain funding to continue development, please contact us via email to pave_support@ncsc.org.

Some of the features we would like to add are listed below.

- Plot observational data & compare with model data.

  There are many possibilities here - we can plot explicit points, or gridded observations. There are a several potential types of rendering including colored circles or textual numbers at observation sites, and contour lines showing the observations (with the first two being easier to incorporate). It is likely that once we have the ability to plot explicit points, we can use the software to view point source emissions data of various types.

- Improve scripting capabilities.

  All PAVE capabilities (as are practical) should be available via scripts and standard input.

- Improved statistical capabilities.

  People often want to use simple statistics to assist with there data analysis (e.g. mean, std. dev.). It would be good if PAVE could produce these. In addition, people may want to slice the data in different ways to obtain these statistics (e.g. mean for each time step or for each vertical layer). There are also a number of commonly used statistics that deal with comparing model data with observations. We may want to examine the "tables" generated for OTAG for ideas for other typically used statistics.

- Optimize PAVE performance and memory usage.

  PAVE performance, both CPU and memory related, can be improved in a number of ways. For example, loops can be reordered to reduce paging to disk, memory leaks can be plugged, and memory can be reused in some formulas.

- User interface improvements.

  The PAVE user interface could use a number of enhancements. Probably the most important of these is to make all plots accessible via the user interface. For example, vector plots, scatter plots, and time series plots with multiple lines are not currently available via the user interface. In addition, small usability improvements could include: an hourglass cursor when PAVE is busy, an indicator of which "mode" a PAVE tile plot is in (e.g. probing, zooming, time series), making changes to the min and max take effect without hitting return, giving windows more meaningful names, and many more.

- Handle missing data appropriately.

  Currently, it is not possible to ignore values that represent missing data (e.g. -999 in some extractions of UAM-V data). This will become more important as observations are included. PAVE readers and renderers need to be updated to handle missing data effectively. Similar techniques might be employed for data outside the user specified legend range.

- Add new types of plots and options for existing ones.

  Additional kinds of plots have been requested by users. Some examples include: vertical cross section plots where the size of the layers is proportional to actual layer size, vertical

profiles, contour plots, flux plots, observational data plots described above, PIG plots, nested grid plots, adaptive grid plots, box & whisker plots. Also, making scatter plots internal to PAVE would help reduce the problems associated with transferring data to BLT and would provide additional flexibility.

- Improve usability of long formulas.

  Currently, long formulas are often difficult to use. This might be improved by providing some defaults for commonly used long formulas (e.g., VOC, TOC, wind speed). Allowing user-defined aliases for long formulas and hierarchical formulas (e.g., NOy - NOx) would be very useful. Allowing the user to specify that multiple parts of a formula refer to the same data set would reduce the complexity involved with long formulas (e.g. [NO + NO_2]a where the a is applied to both NO and NO_2).

- Improve use of map backgrounds.

  It would be useful to be able to plot multiple maps (e.g. state + counties, or counties + roads) using different colors and thicknesses as needed. Another possibility would be to allow backgrounds generated by packages such as GISs to be displayed - these might show roads, water, cities, etc.

- Increase configurability of plots.

  It would be useful in some cases to allow users to define the cutoffs between the colors plotted (currently PAVE makes each bin of equal size from the min to the max). Another alternative might include assigning bins according to percentiles. Additional configuration options could be added to vector plots with the density of the vectors plotted.

- Rework color usage.

  We might use the colors allocated by netscape so the two packages do not fight for colors, we could prevent allocation of colors that are not actually being used, and provide the user with the option of a private color table so that PAVE is not at the mercy of other applications when allocating colors. Also, we could improve the options for black and white plots by adding patterns to the existing shades of gray.

- Improve plot labeling & layout.

  The density and location of grid cell numbering on tile plots and in the domain selector could be improved. Also, the legend could be outlined in black to set off any white or

light values and be positioned relative to the size of the legend labels. The size of tile plots could be made user configurable, and the aspect ratio preserved when the plot is manually resized. Data plotted on time series plots could be labeled more clearly.

- Improve Data Import / Export capabilities.

  It may be possible to include additional readers for PAVE. Also, enhancements of the data export facility would be very useful. For example, one might want to create a new data set out of components several other data sets, or from the result of several complex formulas. For this, PAVE would need to allow multiple formulas to be selected for export, and allow the renaming of variables in the output dataset (e.g. call a variable NOy instead of the long formula used to calculate NOy).

17. For further information

- check the latest version of this document at **http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.html**,

- the latest PAVE Frequently Asked Questions list (FAQ) at **http://www.iceis.mcnc.org/EDSS/pave_doc/Pave.FAQ.html**,

- or send email to pave_support@ncsc.org for bug reports, suggestions for new or improved features, and questions that are unanswered by the above resources.

Steve Thorpe
NCSC Environmental Programs
MCNC
3021 Cornwallis Road
Research Triangle Park, NC 27709
Email: pave_support@ncsc.org

Last Update: 17 Mar 1997 20:33:43
File: @(#)Pave.html 1.86
Pathname: /tmp_mnt/pub/storage/edss/framework/src/pave/pave_doc/SCCS/s.Pave.html

APPENDIX G

CVS

CVS

The source is available at **http://www.polycon.fi:81/CVS-manual/cvs_5.html#SEC5**.

Concurrent Versions System (CVS) is a version control system that enables you to record the history of your source files. Bugs, for example, may be introduced into your files when software is modified. You may not detect these bugs until long after you have made the modification. With CVS, you can easily retrieve old versions to see exactly which change caused the bug.

You could, of course, save every version of every file you have ever created and waste a great deal of disk space in the process. CVS solves this problem by storing only the differences between versions in a single file.

With multiple developers working on the same project, it is all too easy to overwrite the changes of others who are working on the project unless you are extremely careful. Some editors, like GNU Emacs, try to prevent the same file from being modified by two people at the same time. Unfortunately, if someone is using another editor, that safeguard will not work. CVS solves this problem by insulating different developers from one another. Every developer works in his own directory, and CVS merges the work when each developer is done.

Revision Numbers

Each version of a file has a unique *revision number*. Revision numbers look like '1.1,' '1.2,' '1.3.2.2' or even '1.3.2.2.4.5'. A revision number always has an even number of period-separated decimal integers. By default revision 1.1 is the first revision of a file. Each successive revision is given a new number by increasing the rightmost number by one. The following figure displays a few revisions, with newer revisions to the right.

CVS is not limited to linear development. The revision tree can be split into branches, where each branch is a self-maintained line of development. Changes made on one branch can easily be moved back to the main trunk.

Each branch has a branch number, consisting of an odd number of period-separated decimal integers. The branch number is created by appending an integer to the revision number where the corresponding branch forked off. Having branch members allows more than one branch to be forked off from a certain revision.

A file can have several versions, as described above. Likewise, a software product can have several versions. A software product is often given a version number such as '4.1.1'.

In a typical work-session using CVS as in Models-3, a repository has been set up. Suppose you are working on a simple transport chemistry module. The source consists of a handful of F files and a 'Makefile'. The module is called 'tc' (Transport Chemistry), and the repository is set up as a module called 'tc'.

The first thing you must do is to get your own working copy of the source for 'tc'. For this, you use the checkout command:

```
$ cvs checkout tc
```

This will create a new directory called 'tc' and populate it with the source files.

```
$ cd tc
$ ls
CVS          Makefile     backend.f    driver.f     frontend.f   parser.f
```

The 'CVS' directory is used internally by CVS. Normally, you should not modify or remove any of the files in it.

You start your favorite editor, hack away at 'backend.f', and a couple of hours later you have added an optimization pass to the compiler. A note to RCS and SCCS users: There is no need to lock the files that you want to edit.

When you have checked that the compiler is still compilable, you decide to make a new version of 'backend.f.

```
$ CVS commit backend.f
```

The Repository

The CVS repository stores a complete copy of all the files and directories which are under version control. Normally, you never access any of the files in the repository directly. Instead, you use CVS commands to get your own copy of the files, and then work on that copy. When you've finished a set of changes, you check (or commit) them back into the repository. The repository records exactly what you changed, when you changed it, and other such information.

CVS can access a repository by a variety of means. It may be on the local computer, across the room, or across the world. To distinguish various ways to access a repository, the repository

name can start with an access method; for example, the access method :local: means to access a repository directory, so the repository :local:/usr/cvsroot means that the repository is in '/usr/local/cvsroot' on the computer running CVS.

APPENDIX H

Control Equipment Codes
Used in Emission Data Processing

Control Equipment Codes

000 NO EQUIPMENT
001 WET SCRUBBER - HIGH EFFICIENCY
002 WET SCRUBBER - MEDIUM EFFICIENCY
003 WET SCRUBBER - LOW EFFICIENCY
004 GRAVITY COLLECTOR - HIGH EFFICIENCY
005 GRAVITY COLLECTOR - MEDIUM EFFICIENCY
006 GRAVITY COLLECTOR - LOW EFFICIENCY
007 CENTRIFUGAL COLLECTOR - HIGH EFFICIENCY
008 CENTRIFUGAL COLLECTOR - MEDIUM EFFICIENCY
009 CENTRIFUGAL COLLECTOR - LOW EFFICIENCY
010 ELECTROSTATIC PRECIPITATOR - HIGH EFFICIENCY
011 ELECTROSTATIC PRECIPITATOR - MEDIUM EFFICIENCY
012 ELECTROSTATIC PRECIPITATOR - LOW EFFICIENCY
013 GAS SCRUBBER (GENERAL, NOT CLASSIFIED)
014 MIST ELIMINATOR - HIGH VELOCITY I.E. V>250 FT/MIN
015 MIST ELIMINATOR - LOW VELOCITY I.E. V<250 FT/MIN
016 FABRIC FILTER - HIGH TEMPERATURE I.E. T>250F
017 FABRIC FILTER - MEDIUM TEMPERATURE I.E. 180F<T<250 F
018 FABRIC FILTER - LOW TEMPERATURE I.E. T<180F
019 CATALYTIC AFTERBURNER
020 CATALYTIC AFTERBURNER WITH HEAT EXCHANGER
021 DIRECT FLAME AFTERBURNER
022 DIRECT FLAME AFTERBURNER WITH HEAT EXCHANGER
023 FLARING
024 MODIFIED FURNACE OR BURNER DESIGN
025 STAGED COMBUSTION
026 FLUE GAS RECIRCULATION
027 REDUCED COMBUSTION - AIR PREHEATING
028 STEAM OR WATER INJECTION
029 LOW EXCESS - AIR FIRING
030 USE OF FUEL WITH LOW NITROGEN CONTENT
031 AIR INJECTION
032 AMMONIA INJECTION
033 CONTROL OF % O_2 IN COMBUSTION AIR (OFF-STOICHIOMETRIC FIRING)
034 WELLMAN-LORD/SODIUM SULFITE SCRUBBING
035 MAGNESIUM OXIDE SCRUBBING
036 DUAL ALKALI SCRUBBING
037 CITRATE PROCESS SCRUBBING
038 AMMONIA SCRUBBING

039 CATALYTIC OXIDATION - FLUE GAS DESULFURIZATION
040 ALKALIZED ALUMINA
041 DRY LIMESTONE INJECTION
042 WET LIMESTONE INJECTION
043 SULFURIC ACID PLANT - CONTACT PROCESS
044 SULFURIC ACID PLANT - DOUBLE CONTACT PROCESS
045 SULFUR PLANT
046 PROCESS CHANGE
047 VAPOR RECOVERY SYS (INCL. CONDENSERS, HOODING, OTHER ENCLOSURES)
048 ACTIVATED CARBON ADSORPTION
049 LIQUID FILTRATION SYSTEM
050 PACKED-GAS ABSORPTION COLUMN
051 TRAY-TYPE GAS ABSORPTION COLUMN
052 SPRAY TOWER
053 VENTURI SCRUBBER
054 PROCESS ENCLOSED
055 IMPINGEMENT PLATE SCRUBBER
056 DYNAMIC SEPARATOR (DRY)
057 DYNAMIC SEPARATOR (WET)
058 MAT OR PANEL FILTER
059 METAL FABRIC FILTER SCREEN (COTTON GINS)
060 PROCESS GAS RECOVERY
061 DUST SUPPRESSION BY WATER SPRAYS
062 DUST SUPPRESSION BY CHEMICAL STABILIZERS OR WETTING AGENTS
063 GRAVEL BED FILTER
064 ANNULAR RING FILTER
065 CATALYTIC REDUCTION
066 MOLECULAR SIEVE
067 WET LIME SLURRY SCRUBBING
068 ALKALINE FLY ASH SCRUBBING
069 SODIUM CARBONATE SCRUBBING
070 SODIUM-ALKALI SCRUBBING
071 FLUID BED DRY SCRUBBER
072 TUBE AND SHELL CONDENSER
073 REFRIGERATED CONDENSER
074 BAROMETRIC CONDENSER
075 SINGLE CYCLONE
076 MULTIPLE CYCLONE W/O FLY ASH REINJECTION
077 MULTIPLE CYCLONE W/FLY ASH REINJECTION
078 BAFFLE

079 DRY ELECTROSTATIC GRANULAR FILTER
080 CHEMICAL OXIDATION
081 CHEMICAL REDUCTION
082 OZONATION
083 CHEMICAL NEUTRALIZATION
084 ACTIVATED CLAY ADSORPTION
085 WET CYCLONIC SEPARATOR
086 WATER CURTAIN
087 NITROGEN BLANKET
088 CONSERVATION VENT
089 BOTTOM FILLING
090 CONVERSION TO VARIABLE VAPOR SPACE TANK
091 CONVERSION TO FLOATING ROOF TANK
092 CONVERSION TO PRESSURIZED TANK
093 SUBMERGED FILLING
094 UNDERGROUND TANK
095 WHITE PAINT
096 VAPOR LOCK BALANCE RECOVERY SYSTEM
097 INSTALLATION OF SECONDARY SEAL FOR EXTERNAL FLOATING ROOF TANK
098 MOVING BED DRY SCRUBBER
099 MISCELLANEOUS CONTROL DEVICES
101 HIGH EFFICIENCY PARTICULATE AIR FILTER
102 LOW SOLVENT COATINGS
103 POWDER COATINGS
104 WATERBORNE COATINGS
105 PROCESS MODIFICATION - ELECTROSTATIC SPRAYING
106 DUST SUPPRESSION BY PHYSICAL STABILIZATION
107 SELECTIVE NONCATALYTIC REDUCTION FOR NOX
108 DUST SUPPRESSION - TRAFFIC CONTROL

APPENDIX I

Environment Variables

Models-3 System Environment Variables Defined by System Administrator

Environment Variable	Variable Description	Example Value
M3HOME	Models-3 home directory	/home/models3
M3DATA	Models-3 data directory	/home/models3/datasets
M3MODEL	Models-3 models directory	/home/models3/models
M3TOOLS	Models-3 tools directory	/home/models3/tools
M3MMEPPS	Models-3 MEPPS directory	/home/models3/models/mepps
M3MEMPRO	Models-3 EMPRO directory	/home/models3/models/empro
M3MMEPRO	Models-3 MEPRO directory	/home/models3/models/mepro
M3MMIDPRO	Models-3 Inventory Data Analyzer	/home/models3/models/midpro

Global Environment Variables Defined by User on Science Manager Case Page

Environment Variable	Variable Description	Example Value
G_STDATE	Start date (yyyyddd)	1995192
G_STTIME	Start time (hhmmss)	000000
G_RUNLEN	Length of run (hhhmmss) *hours is not restricted to 2 digits	240000
G_TSTEP	Output timestep (hhmmss)	010000

CMAQ System Program Environment Variables

PROGRAM	ENVIRONMENT VARIABLE	VARIABLE DESCRIPTION	DEFAULT VALUE
BCON	SDATE	Start date (yyyyddd)	G_STDATE
	STIME	Start time (hhmmss)	G_STTIME
	RUNLEN	Length of Run (hhhmmss)	G_RUNLEN
ICON	SDATE	Date (yyyyddd)	G_STDATE
	STIME	Start time (hhmmss)	G_STTIME
	PROF. UNITS	Units of profile data	ppb
	RATIONAL. FUNC		False
JPROC	CSQY	Directory path to the absorption cross-section/quantum yield files	M3DATA/nostudies/phot
	JPROC_STDATE	Start date (yyyyddd)	G_STDATE
	JPROC_TOMSEXIT	Does TOMS data file exist? .TRUE. or .FALSE.	.FALSE.
	JVALUES	Full path/output file name	na
	ET	Extraterrestrial radiation file name	M3DATA/nostudies/phot/ Etirradiance.dat
	PROFILES	Vertical profile data file name	M3DATA/nostudies/phot/PROFILES.dat

PROGRAM	ENVIRONMENT VARIABLE	VARIABLE DESCRIPTION	DEFAULT VALUE
	TOMS	TOMS data full path/file name	na
	02ABS	O_2 absorption cross section file name	M3DATA/nostudies/phot/02_NASA94
	03ABS	O_3 absorption cross section file name	M3DATA/nostudies/phot/03_NASA94
ECIP	EXECUTION_ID	Processor Name	ECIP_TGT
	IDPLUM	Plume dispersion method 0 - Plume depth equals plume rise 1 - Empirical method	0
	IDTSEC	Time interval(s)	3600
	IHRM	Mid-point of time period (hhmmss)	003000
	IMAJPS	Include major point source emissions - .TRUE. or .FALSE.	.TRUE.
	IMEPS	Point source handling method .TRUE. - Process MEPSE file .FALSE. - Run Plume in Grid	.TRUE.
	IMETHD	Interpolation method 0 - PBL Similarity 1 - Linear interpolation	1
	IPRFLG	Print diagnostic output ? .TRUE. or .FALSE.	.TRUE.
	JJDATE	Start date (yyyyddd)	G_STDATE
	JJTIME	Start time (hhmmss)	G_STTIME

PROGRAM	ENVIRONMENT VARIABLE	VARIABLE DESCRIPTION	DEFAULT VALUE
	MAXTIM	Number of timesteps in CTM simulation plus one. (i.e. For a 24-hour CTM simulation, value should be 25.)	na
LUPROC	GRDNAM	Grid Name	na
	SDATE	Year of land-use data (yyyyddd) where day is 000	1995660
	STIME	Start time (hhmmss)	000000
MCIP	EXECUTION_ID	User defined execution ID	MCIP_TGT
	GRDNAM	CTM Grid/domain name for output files. Enter selected CTM grid name. See Science Manager.	na
	HSTRT	Start hour in GMT (hh); depends on MM5 data. (00 - for daily runs)	00
	HTORUN	Number of hours to run MUFT(hhh); depends on MM5 data. Typically, the number of hours in the CTM simulation plus one. (i.e. For a 24-hour CTM run, values should be 25 *HTORUN controls run length, not G_RUNLEN.)	na
	HTOSKIP	Number of hours to skip in processing (hh); depends on MM5 data. (00 - for daily runs)	00
	IO	Location of CTM domain origin offset in row direction : Enter ROW_OFFSET value from parent MM5 domain in Science Manager.	na

PROGRAM	ENVIRONMENT VARIABLE	VARIABLE DESCRIPTION	DEFAULT VALUE
	IOAPI_CHECK_HEADERS	Check I/O API file headers TRUE - Check headers FALSE - Do not check headers	TRUE
	J0	Location of CTM domain origin offset in column direction: Enter COL_OFFSET value from parent MM5 domain in Science Manager.	na
	JUDATE	Start date (yyyyddd)	G_STDATE
	LCALCCLD	Cloud algorithm TRUE - compute cloud parameters FALSE - cloud parameters from MM5	TRUE
	LCALCPBL	Estimation method for PBL parameters of note .TRUE. - recompute PBL parameters . FALSE. - PBL parameters from MM5	TRUE
	LSANITY	Perform range checks on output parameters .TRUE. - Check range .FALSE. - Do not check range	TRUE
	LSLFLUX	Similarity algorithm .TRUE. - Use surface layer similarity .FALSE. - Use PBL similarity	FALSE
	LUTYPE	Enter file type of land-use data 1: TERRAIN binary 13 category 2: Preprocessed ACSII 11 category 3: MM5 dominant land-use category 4: use land-use in GRIDCRO2D 5: use USGS I/O API land-use file	5

PROGRAM	ENVIRONMENT VARIABLE	VARIABLE DESCRIPTION	DEFAULT VALUE
	METCOL	Column (east-west) direction cell dimension in input meteorology data: Enter number of columns for selected MM5 grid from Science Manager. Should be larger than CTM NCOLS.	na
	METLAY	Number of layers in input meteorology data: Enter number of vertical layers for selected MM5 grid from Science Manager. Should be larger than or equal to CTM NLAYS.	na
	METROW	Row (south-north) direction cell dimension in input meteorology data: Enter number of rows for selected MM5 grid from Science Manager. Should be larger than CTM NROWS.	na
	NDEP	Number of deposition layers collapsed into lower layer 1 - no collapsing 2 -lowest 2 layers are collapsed 3 -lowest 3 layers are collapsed	1
CCTM	CLD_DIAG	*Write to cloud diagnostic file* Y or N	Y
	CTM_MINSYNC	Min sync time step (sec)	300
	CTM_PROGNAME	Program Name	na
	CTM_RUNLEN	Length of run (hhhmmss)	G_RUNLEN
	CTM_STDATE	Start date (yyyyddd)	G_STDATE
	CTM_STTIME	Start time (hhmmss)	G_STTIME
	CTM_SYMPROC	Symmetric Processing Y or N	.N.

PROGRAM	ENVIRONMENT VARIABLE	VARIABLE DESCRIPTION	DEFAULT VALUE
	CTM_TSTEP	Output Timestep (hhmmss)	G_TSTEP
	FL_ERR_STOP	Stop on inconsistent input file T or F	FALSE
PDM	JJDATE	Start date (yyyyddd)	G_STDATE
	JJTIME	Start time (hhmmss)	G_STTIME
	NHRS	Length of run (hhhmmss)	G_RUNLEN
	ICHUST	Method to compute Lagrangian time scale and sigma y: ICHUST = 1 Apply Hanna et al. (1982) formulas ICHUST = 2 Apply other semi-empirical equation	1
	ICHUSY	Method to determine turbulent lateral dispersion parameter (sigma y) ICHUSY =1 Distance dependent approach for turbulent sigma y term ICHUSY = 2 Time dependent formula for turbulent sigma y term ICHUSY = 3 Alternative semi-empirical form ICHUSY = 4 Another alternative semi-empirical form for all distances	2
	ISHEAR	Direction shear term for sigma y ISHEAR = 0 Omit shear term ISHEAR = 1 Include shear term	1
	IMETHD	Interpolation method for meteorological parameters in vertical IMETHD = 0 PBL similarity IMETHD = 1 Linear interpolation approach	1

	IDPLUM	Method for initial vertical plume spread IDPLUM = 0 let plume depth = plume rise IDPLUM = apply temperature gradient empirical approach	1
	IPARTFLG	Determine plume fractional amounts in layers IPARTFLG = 0 Omit computation IPARTFLG = 1 Turn-on computation	1
	FACTC	Plume width factor relative to grid size to trigger transfer of CONC to CCTM grid	1
	INITC	Plume width (m) criterion for PING initialization	2000.
	DDIRC	Wind direction shear criterion across plume (radians)	1.8
	DSPDC	Wind speed shear criterion across plume (m/s)	3.5
	SPRFA	Initial plume spread factor	10.
	SZ0FA	Plume depth factor	2.00

MEPPS Environment Variables

Environment Variable	Location	Function
EMS	EMS_HOME/project/EMS_PROJECT/gridspec/ EMS_GRID/common/sas	SAS® library reference holding ungridded emission files.
EMS_BIN	EMPRO_HOME/bin	EMPRO executable directory.
EMS_CAT	EMS_HOME/project/EMS_PROJECT/lookup	Directory with study lookup tables.
EMS_CVRT	EMS_HOME/project/EMS_PROJECT/gridspec/ EMS_GRID/common/sas	SAS® file reference holding ungridded emission files.
EMS_DOMAIN	Not a location.	Source name.
EMS_FILE	EMS_HOME/project/EMS_PROJECT/gridspec/ EMS_GRID/scenario/ EMS_SCENARIO/sas	Contains ASCII files pertaining to specific day scenarios. Same as EMS_RUN
EMS_GRID	Does not have a location.	Grid name.
EMS_GRD	EMS_HOME/project/ EMS_PROJECT/ gridspec/ EMS_GRID/sas	SAS® library reference GIS grid description and gridded emission datasets.
EMS_HOME	System administrator defined.	Base directory for EMPRO software and system files.
EMS_LOC	EMS_HOME/project/ EMS_PROJECT/ raw_data/ EMS_DOMAIN/common	Directory containing raw data files for a data source.
EMS_PROJECT	Does not have a location.	Study name.
EMS_RUN	EMS_HOME/project/ EMS_PROJECT/ gridspec/ EMS_GRID/scenario/ EMS_SCENARIO/sas	Contains SAS® datasets for a specific case. Default directory for I/O API files.
EMS_SCENARIO	Not a location.	EMPRO case name.

Environment Variable	Location	Function
EMS_SYS	EMPRO_HOME/lib/sysdata	EMPRO system lookup tables.
EMS_SYSF	SAS_HOME/sas/sasxxx/saslib	SAS library and file references.
EMSG_GRD	EMS_HOME/project/ EMS_PROJECT/ gridspec/ EMS_GRID/sas	SAS® file reference for GIS coverages overlaid with the grid and loaded into datasets.
EMSG	EMS_HOME/project/ EMS_PROJECT/ gridspec/ EMS_GRID/common/gis	Contains ungridded ARC/INFO® coverage files overlaid on the grid.
EMSG_GRD	EMS_HOME/project/ EMS_PROJECT/ gridspec/ EMS_GRID/gis	SAS® file reference for gridded ARC/INFO® coverage files.
GIP_BIN	MEPPS_HOME/bin	MEPPS executable directory.
GIP_INV	MEPPS_HOME/inv	Base directory of raw national or international emission inventories.
GIP_SYS	MEPPS_HOME/lib	MEPPS system lookup tables.
GISDB	GSDB	Assigned location of national or international ARC/INFO® Coverages.
MAPS	SAS_HOME/sas/sasxxx/maps	SASGRAPH® map data directory.
MEPPS_HOME	System administrator defined.	Base directory for MEPPS software and system data.
RAW_DATA	EMS_HOME/project/ EMS_PROJECT/ raw_data/EMS_DOMAIN/common	Contains ASCII emission and input files.
SASHELP	SAS_HOME/sas/sasxxx/sashelp	Contain SAS® help programs.
SASUSER	SAS_HOME/sas/sasxxx/sasuser	Contains user-specific programs.

Environment Variable	Location	Function
SASROOT	System administrator defined.	Root directory for SAS® software.
SPEC_LIB	MEPPS_HOME/lib/ sysdata	As for EMS_SYS, except for focus on speciation tables.
WORK	System administrator defined.	Work area intended for files of processing underway.

Environment Variables Used in the Study Planner for MEPPS

Environment Variable	Variable Description	Example Value
MEPPS_HOME	Base directory for MEPPS software.	normally this should be M3MMEPPS.
EMPRO_HOME	Base directory for EMPRO software.	normally this is M3MEMPRO.
EMS_HOME	Environment Variables.	/home/User Name/
EMS_PROJECT	Project Directory.	.../project/ProjectName/ ex: mepps_4
EMS_GRID	Grid Directory.	/project/Proj Name/gridspec/Grid Name/ ex.: c_ctmt_36_ne
EMS_DOMAIN	Source, defines raw data directory.	/project/Proj Name/raw_data/Source Name/ common/ ex: proj_4
EMS_LOC	Source, defines raw data directory. If specified, overrides EMS_DOMAIN.	/project/Proj Name/raw_data/Source Name/ common/ ex: base_4
MET_CRO_2D MET_CRO_3D MET_DO_3D GRID_DOT_2D MET_DATE	Names of the 4 MCIP Met Files required. Met Data Date of the First Day.	MET_CRO_2D_G1 mm/dd/yyyy

Environment Variable	Variable Description	Example Value
For the Stack Files: MOLES GRAMS STAK_MAJOR CRO_SP_MAJOR CRO_SP_MAJG STAK_MEPSE CRO_SP_M_MEPSE CRO_SP_MEPG LOGFILE	Used in Generate Stack. I/O API Files. They Generate Moles/Grams Emission Files. Major Stack File. Major Stack Emission File (in Moles). Major Stack Emission File (in grams). MEPSE Stack File. MEPSE Stack Emission File (in Moles). MEPSE Stack Emission File (in grams). I/O API Log File Name.	na
For the 2-D Speciated Files: SPEMIS2D SPEMIS2D_G Point Area Mobile BIO	 2D Speciated Emission File (in Moles). 2D Speciated Emission File (in Grams). Names of 2-D Speciated emission Files to be included.	na

Note 1: A typical MEPPS Study contains several MEPPS Plans. Examples are:

area	Grid level area source calculations (US data)
area_can	Grid level area source calculations (US and Canadian data
mobile	Grid level mobile source calculations (US data)
mobile_can	Grid level mobile source calculations (US and Canadian data)
point	Grid level point source calculations (US data)
point_can	Grid level point source calculations (US and Canadian data)
case	Case level calculations (point, area, mobile, biogenic)

Note 2: The **Plan Node** has the following node level environment variable:

EMS_SCENARIO Case name is used by case level calculations if there are multiple scenarios such as a Base case vs. Projection case. The user can specify the Met data day to use for the first case day in environment variable **MET_DATE.** The environment variable contains a date of the form yyyy/mm/dd.

Note 3: The user should set the following environment variables for node **calc_mv**:

EF_YEAR Specifies the year of emission factors used in the motor vehicle emission calculations It defaults to the case year if not specified.

EF_TYPE Specifies the one of the following two ways to assign emission factors; HR by hour; AF by areatype and roadtype.

APPENDIX J

Input and Output File Descriptions

1.0 File Formats and Naming Conventions

Standard files created and used in the Models-3/CMAQ system are written and read using the Models-3 Input/Output Applications Programming Interface (I/O API) software library developed by MCNC. The internal data format used by this I/O API library is an IEEE machine independent format that conforms to the widely used University Corporation for Atmospheric Research network Common Data Format (netCDF) © 1993. I/O API files include self-describing headers containing the necessary information (i.e. variable names, descriptions, and units) to use and interpret the data contained in the file. The I/O API library provides a variety of data structure types and a set of reusable access routines that offer selective direct access to the data in terms that are meaningful to the environmental modeler. Supported data types include: gridded, boundary, ID referenced, vertical profile, grid nest, time series, and event-driven (See **http://sage.mcnc.org/products/ioapi/index.html** for details).

The user does not need to know the format details of these standard data files since all file I/O is performed via calls to the I/O API library routines. Therefore in this appendix, descriptions of I/O API data files are presented in tables containing three columns with the variable name, description, and units just as they appear on the file headers. Since the users can define their own variable names, the names and descriptions in these files can vary with your application. The examples presented here are for the RADM2_AE_AQ chemistry mechanism. You can find additional descriptions of the chemical species names in Appendix M. The variable names used in the emissions processors can be different from the variable names used in the CCTM as long as they are properly linked using the Models-3 Chemical Mechanism Manager (See Section 7.5.2.3 Species Table Data). Any file described in this appendix that does not adhere to this I/O API netCDF standard format are considered to be non-standard files and will be designated with an acronym for their data type. Non-standard data types currently used in Models-3/CMAQ include ASCII and IEEE binary.

MCIP often uses a file suffix _G1 to designate a grid family association. This type of designation is typically used when preparing for multiple CCTM executions in a nested grid sequence. The file structure and content are the same whether or not this suffix appears.

Models-3/CMAQ file names contain abbreviations that can be interpreted as follows:

IC	Initial Conditions
INIT	Initial Conditions
AERO	Aerosol
BC	Boundary Conditions
BDY	Boundary Cells
BNDY	Boundary Cells
CONC	Concentrations

CTM	Chemistry Transport Model
CRO	Cross Points
DIAM	Stack Diameters
DEP	Deposition
DOT	Dot points
EMIS	Emissions
GASC	Gas phase chemistry
GRID	Time independent data
IPR	Integrated Process Rates
IRR	Integrated Reaction Rates
MEPSE	Major Elevated Point Source Emitters
MET	Time dependent data
NONR	Non-reactive species
PA	Process Analysis
PING	Plume-in-Grid
PROFILE	Vertical profile data
TRAC	Tracer species
VIS	Visibility

2.0 Boundary Condition Processor (BCON)

a. Input Files

i. BC_PROFILE

Near rural clean tropospheric air quality concentration vertical profiles for long-lived RADM species. If you are using a CB-4 mechanism, be sure to define a new BCON configuration template that specifies RADM2 to CB4 conversion.

Label	Description	Definition	Units
SO2	SO2 gas concentration	(Real, E9.3)	ppmV
SULF	SULF concentration	(Real, E9.3)	ppmV
NO2	NO2 gas concentration	(Real, E9.3)	ppmV
NO	NO gas concentration	(Real, E9.3)	ppmV
O3	O3 gas concentration	(Real, E9.3)	ppmV
HNO3	HNO3 gas concentration	(Real, E9.3)	ppmV
H2O2	H2O2 gas concentration	(Real, E9.3)	ppmV
ALD	ALD gas concentration	(Real, E9.3)	ppmV
HCHO	HCHO gas concentration	(Real, E9.3)	ppmV
OP1	OP1 gas concentration	(Real, E9.3)	ppmV
OP2	OP2 gas concentration	(Real, E9.3)	ppmV
PAA	PAA gas concentration	(Real, E9.3)	ppmV
ORA1	ORA1 gas concentration	(Real, E9.3)	ppmV
ORA2	ORA2 gas concentration	(Real, E9.3)	ppmV

NH3	NH3 gas concentration	(Real, E9.3)	ppmV
N2O5	N2O5 gas concentration	(Real, E9.3)	ppmV
NO3	NO3 gas concentration	(Real, E9.3)	ppmV
PAN	PAN gas concentration	(Real, E9.3)	ppmV
HC3	HC3 gas concentration	(Real, E9.3)	ppmV
HC5	HC5 gas concentration	(Real, E9.3)	ppmV
HC8	HC8 gas concentration	(Real, E9.3)	ppmV
ETH	ETH gas concentration	(Real, E9.3)	ppmV
CO	CO gas concentration	(Real, E9.3)	ppmV
OL2	OL2 gas concentration	(Real, E9.3)	ppmV
OLT	OLT gas concentration	(Real, E9.3)	ppmV
OLI	OLI gas concentration	(Real, E9.3)	ppmV
TOL	TOL gas concentration	(Real, E9.3)	ppmV
XYL	XYL gas concentration	(Real, E9.3)	ppmV
ACO3	ACO3 gas concentration	(Real, E9.3)	ppmV
TPAN	TPAN gas concentration	(Real, E9.3)	ppmV
HONO	HONO gas concentration	(Real, E9.3)	ppmV
HNO4	HNO4 gas concentration	(Real, E9.3)	ppmV
KET	KET gas concentration	(Real, E9.3)	ppmV
GLY	GLY gas concentration	(Real, E9.3)	ppmV
MGLY	MGLY gas concentration	(Real, E9.3)	ppmV
DCB	DCB gas concentration	(Real, E9.3)	ppmV
ONIT	ONIT gas concentration	(Real, E9.3)	ppmV
CSL	CSL gas concentration	(Real, E9.3)	ppmV
ISO	ISO gas concentration	(Real, E9.3)	ppmV
HO	HO gas concentration	(Real, E9.3)	ppmV
HO2	HO2 gas concentration	(Real, E9.3)	ppmV
ASO4I	SO4 i-mode aerosol conc.	(Real, E9.3)	u/m^3
ASO4J	SO4 j-mode aerosol conc.	(Real, E9.3)	u/m^3
NUMATKN	i-mode aerosol number conc.	(Real, E9.3)	number/m^3
NUMACC	j-mode aerosol number conc.	(Real, E9.3)	number/m^3
ASOIL	coarse dust aerosol conc.	(Real, E9.3)	u/m^3
NUMCOR	coarse aerosol number conc.	(Real, E9.3)	number/m^3

ii. CTM_CONC_1 See 4.b.i CMAQ output file

b. Output Files

i. BNDY_CONC_1

Boundary Conditions

VARIABLE NAME	DESCRIPTION	UNITS
NO2	Variable NO2	ppmV
NO	Variable NO	ppmV
O3	Variable O3	ppmV
HONO	Variable HONO	ppmV

VARIABLE NAME	DESCRIPTION	UNITS
HO	Variable HO	ppmV
HNO3	Variable HNO3	ppmV
HNO4	Variable HNO4	ppmV
HO2	Variable HO2	ppmV
NO3	Variable NO3	ppmV
H2O2	Variable H2O2	ppmV
HCHO	Variable HCHO	ppmV
CO	Variable CO	ppmV
ALD	Variable ALD	ppmV
OP1	Variable OP1	ppmV
OP2	Variable OP2	ppmV
PAA	Variable PAA	ppmV
KET	Variable KET	ppmV
ACO3	Variable ACO3	ppmV
GLY	Variable GLY	ppmV
MGLY	Variable MGLY	ppmV
DCB	Variable DCB	ppmV
ONIT	Variable ONIT	ppmV
N2O5	Variable N2O5	ppmV
SO2	Variable SO2	ppmV
SULF	Variable SULF	ppmV
ETH	Variable ETH	ppmV
HC3	Variable HC3	ppmV
HC5	Variable HC5	ppmV

VARIABLE NAME	DESCRIPTION	UNITS
HC8	Variable HC8	ppmV
OL2	Variable OL2	ppmV
OLT	Variable OLT	ppmV
OLI	Variable OLI	ppmV
TOL	Variable TOL	ppmV
CSL	Variable CSL	ppmV
XYL	Variable XYL	ppmV
PAN	Variable PAN	ppmV
TPAN	Variable TPAN	ppmV
ORA1	Variable ORA1	ppmV
ORA2	Variable ORA2	ppmV
ISO	Variable ISO	ppmV
NH3	Variable NH3	ppmV

3.0 Initial Condition Processor (ICON)

a. Input Files

i. IC_PROFILE See 2.a .i

BCON Input file. Currently the same clean tropospheric vertical profile is used for both ICON and BCON input.

ii. CTM_CONC_1 See 4.b.i CMAQ output file

b. Output Files

i. INIT_CONC_1

Initial Conditions

VARIABLE NAME	DESCRIPTION	UNITS
NO2	Variable NO2	ppmV
NO	Variable NO	ppmV
O3	Variable O3	ppmV
HONO	Variable HONO	ppmV
HO	Variable HO	ppmV
HNO3	Variable HNO3	ppmV
HNO4	Variable HNO4	ppmV
HO2	Variable HO2	ppmV
NO3	Variable NO3	ppmV
H2O2	Variable H2O2	ppmV
HCHO	Variable HCHO	ppmV
CO	Variable CO	ppmV
ALD	Variable ALD	ppmV
OP1	Variable OP1	ppmV
OP2	Variable OP2	ppmV
PAA	Variable PAA	ppmV
KET	Variable KET	ppmV
ACO3	Variable ACO3	ppmV
GLY	Variable GLY	ppmV
MGLY	Variable MGLY	ppmV
DCB	Variable DCB	ppmV
ONIT	Variable ONIT	ppmV
N2O5	Variable N2O5	ppmV
SO2	Variable SO2	ppmV
SULF	Variable SULF	ppmV

VARIABLE NAME	DESCRIPTION	UNITS
ETH	Variable ETH	ppmV
HC3	Variable HC3	ppmV
HC5	Variable HC5	ppmV
HC8	Variable HC8	ppmV
OL2	Variable OL2	ppmV
OLT	Variable OLT	ppmV
OLI	Variable OLI	ppmV
TOL	Variable TOL	ppmV
CSL	Variable CSL	ppmV
XYL	Variable XYL	ppmV
PAN	Variable PAN	ppmV
TPAN	Variable TPAN	ppmV
ORA1	Variable ORA1	ppmV
ORA2	Variable ORA2	ppmV
ISO	Variable ISO	ppmV
NH3	Variable NH3	ppmV

4.0 CMAQ Chemistry-Transport Model (CCTM)

a. Input Files

Currently the initial conditions files (INIT_ ...) all default to use the INIT_GASC_1 file which defaults to the BCON output file BNDY_CONC_1. The CCTM boundary conditions files (BNDY_ ...) all default to use the BNDY_GASC_1 file which defaults to the BCON output file BNDY_CONC_1. The user can substitute other files of similar format to modify for specific conditions relating to aerosols (AERO), gas phase chemistry (GASC), non-reactive species (NONR), or tracer species (TRAC).

i.	**BNDY_AERO_1**	See 2.b.i	BCON Output file BNDY_CONC_1
ii.	**BNDY_GASC_1**	See 2.b.i	BCON Ouput file BNDY_CONC_1
iii.	**BNDY_NONR_1**	See 2.b.i	BCON Ouput file BNDY_CONC_1
iv.	**BNDY_TRAC_1**	See 2.b.i	BCON Ouput file BNDY_CONC_1
v.	**DEPV_TRAC_1**	See 8.b.viii	MCIP Output file MET_CRO_2D
vi.	**EMIS_1**	See 5.b.i	ECIP Output file EMIS_1
vii.	**EMIS_TRAC_1**	See 5.b.i	ECIP Output file EMIS_1
viii.	**GRID_BDY_2D**	See 8.b.1	MCIP Output file GRID_BDY_2D
ix.	**GRID_CRO_2D**	See 8.b.iii	MCIP Output file GRID_CRO_2D
x.	**GRID_DOT_2D**	See 8.b.v	MCIP Output file GRID_DOT_2D
xi.	**INIT_AERO_1**	See 3.b.i or 4.b.i	

Use INIT_CONC_1 for first run of a case. If more than one run needed to complete the case use CTM_CONC_1 output file from previous run of the case.

xii. **INIT_GASC_1** See 3.b.i or 4.b.i

Use INIT_CONC_1 for first run of a case. If more than one run needed to complete the case use CTM_CONC_1 output file from previous run of the case.

xiii. **INIT_NONR_1** See 3.b.i or 4.b.i

Use INIT_CONC_1 for first run of a case. If more than one run needed to complete the case use CTM_CONC_1 output file from previous run of the case.

xiv. **INIT_TRAC_1** See 3.b.i or 4.b.i

Use INIT_CONC_1 for first run of a case. If more than one run needed to complete the case use CTM_CONC_1 output file from previous run of the case.

xv. **MEPSE_1** See OUTPRO output file CRO_SP_MEPSE.

Optional file only used when PING is included in execution. Must match with selected chemistry mechanism.

xvi. **MET_BDY_2D** See 8.b.vi MCIP Output file MET_BDY_2D

xvii. **MET_BDY_3D** See 8.b.vii MCIP Output file MET_BDY_3D

xviii. **MET_CRO_2D** See 8.b.viii MCIP Output file MET_CRO_2D

xix. **MET_CRO_3D** See 8.b.ix MCIP Output file MET_CRO_3D

xx. **MET_DOT_3D** See 8.b.x MCIP Output file MET_DOT_3D

xxi. **PDM_PING_1** See 10.b.i PDM Output file

Optional file only used when PING is included for execution. Must match with selected chemistry mechanism.

xxii. **XJ_DATA** See 6.b.i JPROC Output file (same as JVALUES)

b. Output Files

i. CTM_CONC_1

Simulated Air Quality Concentrations

VARIABLE NAME	DESCRIPTION	UNITS
NO2	Variable NO2	ppmV
O3P	Variable O3P	ppmV
NO	Variable NO	ppmV
O3	Variable O3	ppmV
01D	Variable 01D	ppmV
HONO	Variable HONO	ppmV
HO	Variable HO	ppmV

VARIABLE NAME	DESCRIPTION	UNITS
HNO3	Variable HNO3	ppmV
HNO4	Variable HNO4	ppmV
HO2	Variable HO2	ppmV
NO3	Variable NO3	ppmV
H2O2	Variable H2O2	ppmV
HCHO	Variable HCHO	ppmV
CO	Variable CO	ppmV
ALD	Variable ALD	ppmV
MO2	Variable MO2	ppmV
OP1	Variable OP1	ppmV
OP2	Variable OP2	ppmV
PAA	Variable PAA	ppmV
KET	Variable KET	ppmV
ACO3	Variable ACO3	ppmV
ETHP	Variable ETHP	ppmV
GLY	Variable GLY	ppmV
MGLY	Variable MGLY	ppmV
DCB	Variable DCB	ppmV
TCO3	Variable TCO3	ppmV
ONIT	Variable ONIT	ppmV
N2O5	Variable N2O5	ppmV
SO2	Variable SO2	ppmV
SULF	Variable SULF	ppmV
ETH	Variable ETH	ppmV

VARIABLE NAME	DESCRIPTION	UNITS
HC3	Variable HC3	ppmV
HC3P	Variable HC3P	ppmV
HC5	Variable HC5	ppmV
HC5P	Variable HC5P	ppmV
XO2	Variable XO2	ppmV
HC8	Variable HC8	ppmV
HC8P	Variable HC8P	ppmV
OL2	Variable OL2	ppmV
OL2P	Variable OL2P	ppmV
OLT	Variable OLT	ppmV
OLTP	Variable OLTP	ppmV
OLI	Variable OLI	ppmV
OLIP	Variable OLIP	ppmV
TOL	Variable TOL	ppmV
TOLP	Variable TOLP	ppmV
CSL	Variable CSL	ppmV
XYL	Variable XYL	ppmV
XYL	Variable XYL	ppmV
KETP	Variable KETP	ppmV
PAN	Variable PAN	ppmV
ISO	Variable ISO	ppmV
TPAN	Variable TPAN	ppmV
OLN	Variable OLN	ppmV
XNO2	Variable XNO2	ppmV

VARIABLE NAME	DESCRIPTION	UNITS
ORA1	Variable ORA1	ppmV
ORA2	Variable ORA2	ppmV
TERP	Variable TERP	ppmV
ISO_RO2	Variable ISO_RO2	ppmV
ISOPROD	Variable ISOPROD	ppmV
ISON_RO2	Variable ISON_RO2	ppmV
IP_RO2	Variable IP_RO2	ppmV
ASO4J	Variable ASO4J	micrograms/m**3
ASO4I	Variable ASO4I	micrograms/m**3
ANH4J	Variable ANH4J	micrograms/m**3
ANH4I	Variable ANH4I	micrograms/m**3
ANO3J	Variable ANO3J	micrograms/m**3
ANO3I	Variable ANO3I	micrograms/m**3
AORGAJ	Variable AORGAJ	micrograms/m**3
AORGAI	Variable AORGAI	micrograms/m**3
AORGPAJ	Variable AORGPAJ	micrograms/m**3
AORGPAI	Variable AORGPAI	micrograms/m**3
AORGBJ	Variable AORGBJ	micrograms/m**3
AORGBI	Variable AORGBI	micrograms/m**3
AECJ	Variable AECJ	micrograms/m**3
AECI	Variable AECI	micrograms/m**3
A25J	Variable A25J	micrograms/m**3
A25I	Variable A25I	micrograms/m**3
ACORS	Variable ACORS	micrograms/m**3

VARIABLE NAME	DESCRIPTION	UNITS
ASEAS	Variable ASEAS	micrograms/m**3
ASOIL	Variable ASOIL	micrograms/m**3
NUMATKN	Variable NUMATKN	number/m**3
NUMACC	Variable NUMACC	number/m**3
NUMCOR	Variable NUMCOR	number/m**3
AH2OJ	Variable AH2OJ	micrograms/m**3
AH2OI	Variable AH2OI	micrograms/m**3
NH3	Variable NH3	ppmV

ii. CTM_DRY_DEP_1

Simulated Dry Deposition

VARIABLE NAME	DESCRIPTION	UNITS
SO2	hourly dry deposition values	kg/hectare
SULF	hourly dry deposition values	kg/hectare
NO2	hourly dry deposition values	kg/hectare
NO	hourly dry deposition values	kg/hectare
O3	hourly dry deposition values	kg/hectare
HNO3	hourly dry deposition values	kg/hectare
H2O2	hourly dry deposition values	kg/hectare
ALD	hourly dry deposition values	kg/hectare
HCHO	hourly dry deposition values	kg/hectare
OP1	hourly dry deposition values	kg/hectare
OP2	hourly dry deposition values	kg/hectare
PAA	hourly dry deposition values	kg/hectare

VARIABLE NAME	DESCRIPTION	UNITS
ORA1	hourly dry deposition values	kg/hectare
ORA2	hourly dry deposition values	kg/hectare
ASO4J	hourly dry deposition values	kg/hectare
ASO4I	hourly dry deposition values	kg/hectare
ANH4J	hourly dry deposition values	kg/hectare
ANH4I	hourly dry deposition values	kg/hectare
ANO3J	hourly dry deposition values	kg/hectare
ANO3I	hourly dry deposition values	kg/hectare
AORGAI	hourly dry deposition values	kg/hectare
AORGPAJ	hourly dry deposition values	kg/hectare
AORGPAI	hourly dry deposition values	kg/hectare
AORGBJ	hourly dry deposition values	kg/hectare
AORGBI	hourly dry deposition values	kg/hectare
AECJ	hourly dry deposition values	kg/hectare
AECI	hourly dry deposition values	kg/hectare
A25J	hourly dry deposition values	kg/hectare
A25I	hourly dry deposition values	kg/hectare
ACORS	hourly dry deposition values	kg/hectare
ASEAS	hourly dry deposition values	kg/hectare
ASOIL	hourly dry deposition values	kg/hectare
NUMATKN	hourly dry deposition values	kg/hectare
NUMACC	hourly dry deposition values	kg/hectare
NUMCOR	hourly dry deposition values	kg/hectare
NH3	hourly dry deposition values	kg/hectare

iii. CTM_IPR_1

Optional; Integrated Process Rates file contents depend on the number of species to be output as past of Process Analysis. See Appendix M for interpretation.

Simulated Process Rate Contribution to Species Concentrations. (Only 120 species can be written to an individual file. You need to assign a separate output file for each 120 species that will be output.)

VARIABLE NAME	DESCRIPTION	UNIT
XADV_NO	X-Advection of NO	ppm
YADV_NO	Y-Advection of NO	ppm
ZADV_NO	Z-Advection of NO	ppm
ADJC_NO	Advection mass adjustment for NO	ppm
HDIF_NO	Horizontal diffusion for NO	ppm
VDIF_NO	Vertical diffusion for NO	ppm
EMIS_NO	Emissions of NO	ppm
DDEP_NO	Dry deposition for NO	ppm
CLDS_NO	Clouds for NO	ppm
CHEM_NO	Chemistry of NO	ppm
AERO_NO	Aerosols for NO	ppm
XADV_NO2	X-Advection of NO2	ppm
YADV_NO2	Y-Advection of NO2	ppm
ZADV_NO2	Z-Advection of NO2	ppm
ADJC_NO2	Advection mass adjustment for NO2	ppm
HDIF_NO2	Horizontal diffusion for NO2	ppm
VDIF_NO2	Vertical diffusion for NO2	ppm

VARIABLE NAME	DESCRIPTION	UNIT
EMIS_NO2	Emissions of NO2	ppm
DDEP_NO2	Dry deposition for NO2	ppm
CLDS_NO2	Clouds for NO2	ppm
CHEM_NO2	Chemistry of NO2	ppm
AERO_NO2	Aerosols for NO2	ppm
XADV_O3	X-Advection of O3	ppm
YADV_O3	Y-Advection of O3	ppm
ZADV_O3	Z-Advection of O3	ppm
ADJC_O3	Advection mass adjustment for O3	ppm
HDIF_O3	Horizontal diffusion for O3	ppm
VDIF_O3	Vertical diffusion for O3	ppm
EMIS_O3	Emissions of O3	ppm
DDEP_O3	Dry deposition for O3	ppm
CLDS_O3	Clouds for O3	ppm
CHEM_O3	Chemistry of O3	ppm
AERO_O3	Aerosols for O3	ppm
XADV_CO	X-Advection of CO	ppm
YADV_CO	Y-Advection of CO	ppm
ZADV_CO	Z-Advection of CO	ppm
ADJC_CO	Advection mass adjustment for CO	ppm
HDIF_CO	Horizontal diffusion for CO	ppm
VDIF_CO	Vertical diffusion for CO	ppm
EMIS_CO	Emissions of CO	ppm
DDEP_CO	Dry deposition for CO	ppm

VARIABLE NAME	DESCRIPTION	UNIT
CLDS_CO	Clouds for CO	ppm
CHEM_CO	Chemistry of CO	ppm
AERO_CO	Aerosols for CO	ppm
XADV_VOC	X-Advection of VOC	ppm
YADV_VOC	Y-Advection of VOC	ppm
ZADV_VOC	Z-Advection of VOC	ppm
ADJC_VOC	Advection mass adjustment for VOC	ppm
HDIF_VOC	Horizontal diffusion for VOC	ppm
VDIF_VOC	Vertical diffusion for VOC	ppm
EMIS_VOC	Emissions of VOC	ppm
DDEP_VOC	Dry deposition for VOC	ppm
CLDS_VOC	CLDS_VOC	ppm
CHEM_VO	Chemistry of VOC	ppm
AERO_VOC	Aerosols for VOC	ppm
XADV_HNO3	X-Advection of HNO3	ppm
YADV_HNO3	Y-Advection of HNO3	ppm
ZADV_HNO3	Z-Advection of HNO3	ppm
ADJC_HNO3	Advection mass adjustment for HNO3	ppm
HDIF_HNO3	Horizontal diffusion for HNO3	ppm
VDIF_HNO3	Vertical diffusion for HNO3	ppm
EMIS_HNO3	Emissions of HNO3	ppm
DDEP_HNO3	Dry deposition for HNO3	ppm
CLDS_HNO3	Clouds for HNO3	ppm
CHEM_HNO3	Chemistry of HNO3	ppm

VARIABLE NAME	DESCRIPTION	UNIT
AERO_HNO3	Aerosols for HNO3	ppm
XADV_H2O2	X-Advection of H2O2	ppm
YADV_H2O2	Y-Advection of H2O2	ppm
ZADV_H2O2	Z-Advection of H2O2	ppm
ADJC_H2O2	Advection mass adjustment for H2O2	ppm
HDIF_H2O2	Horizontal diffusion for H2O2	ppm
VDIF_H2O2	Vertical diffusion for H2O2	ppm
EMIS_H2O2	Emissions of H2O2	ppm
DDEP_H2O2	Dry deposition for H2O2	ppm
CLDS_H2O2	Clouds for H2O2	ppm
CHEM_H2O2	Chemistry of H2O2	ppm
AERO_H2O2	Aerosols for H2O2	ppm
XADV_OX	X-Advection of OX	ppm
YADV_OX	Y-Advection of OX	ppm
ZADV_OX	Z-Advection of OX	ppm
ADJC_OX	Advection mass adjustment for OX	ppm
HDIF_OX	Horizontal diffusion for OX	ppm
VDIF_OX	Vertical diffusion for OX	ppm
EMIS_OX	Emissions of OX	ppm
DDEP_O	Dry deposition for OX	ppm
CLDS_OX	Clouds for OX	ppm
CHEM_OX	Chemistry of OX	ppm
AERO_OX	Aerosols for OX	ppm
XADV_NOZ	X-Advection of NOZ	ppm

VARIABLE NAME	DESCRIPTION	UNIT
YADV_NOZ	Y-Advection of NOZ	ppm
ZADV_NOZ	Z-Advection of NOZ	ppm
ADJC_NOZ	Advection mass adjustment for NOZ	ppm
HDIF_NOZ	Horizontal diffusion for NOZ	ppm
VDIF_NOZ	Vertical diffusion for NOZ	ppm
EMIS_NOZ	Emissions of NOZ	ppm
DDEP_NOZ	Dry deposition for NOZ	ppm
CLDS_NOZ	Clouds for NOZ	ppm
CHEM_NOZ	Chemistry of NOZ	ppm
AERO_NOZ	Aerosols for NOZ	ppm

iv. CTM_IPR_2

Optional; Integrated Process Rates file contents depend on the number of species output as part of the process analysis. See 4.b.iii for format and Appendix M for interpretation.

v. CTM_IRR_1

Optional; Integrated Reaction Rates are needed only when requested in process analysis. See table below and Appendix M for details.

vi. CTM_IRR_2

Optional; Integrated Reaction Rates are needed only when requested in process analysis. See table below and Appendix M for details.

vii. CTM_IRR_3

Optional; Integrated Reaction Rates are needed only when requested in process analysis. See table below and Appendix M for details.

IRR_1	IRR_1:<P1>	ppm

IRR_2	IRR_2:<P2>	ppm
IRR_3	IRR_3:<P3>	ppm
⋮		
IRR_N	IRR_N:<PN>	ppm

viii. CTM_PING_1

Optional; needed only when PING is included in execution. See 10.b.i PDM Output file PDM_PING_1.

ix. CTM_VIS_1

Optional; needed only when visibility module is included in execution.

Simulated Visibility

VARIABLE NAME	DESCRIPTION	UNITS
DCV_Mie	hourly visual range	deciview
EXT_Mie	hourly extinction coefficient	1/km
DCV_Recon	hourly visual range	deciview
EXT_Recon	hourly extinction coefficient	1/km

x. CTM_WET_DEP_1

Simulated Wet Deposition

VARIABLE NAME	DESCRIPTION	UNITS
SO2	hourly wet deposition values	kg/hectare
SULF	hourly wet deposition values	kg/hectare
NO2	hourly wet deposition values	kg/hectare
NO	hourly wet deposition values	kg/hectare
O3	hourly wet deposition values	kg/hectare

VARIABLE NAME	DESCRIPTION	UNITS
HNO3	hourly wet deposition values	kg/hectare
H2O2	hourly wet deposition values	kg/hectare
ALD	hourly wet deposition values	kg/hectare
HCHO	hourly wet deposition values	kg/hectare
OP1	hourly wet deposition values	kg/hectare
OP2	hourly wet deposition values	kg/hectare
PAA	hourly wet deposition values	kg/hectare
ORA1	hourly wet deposition values	kg/hectare
ORA2	hourly wet deposition values	kg/hectare
ASO4J	hourly wet deposition values	kg/hectare
ASO4I	hourly wet deposition values	kg/hectare
ANH4J	hourly wet deposition values	kg/hectare
ANH4I	hourly wet deposition values	kg/hectare
ANO3J	hourly wet deposition values	kg/hectare
ANO3I	hourly wet deposition values	kg/hectare
AORGAJ	hourly wet deposition values	kg/hectare
AORGAI	hourly wet deposition values	kg/hectare
AORGPAJ	hourly wet deposition values	kg/hectare
AORGPAI	hourly wet deposition values	kg/hectare
AORGBJ	hourly wet deposition values	kg/hectare
AORGBI	hourly wet deposition values	kg/hectare
AECJ	hourly wet deposition values	kg/hectare
AECI	hourly wet deposition values	kg/hectare
A25J	hourly wet deposition values	kg/hectare

VARIABLE NAME	DESCRIPTION	UNITS
A25I	hourly wet deposition values	kg/hectare
ACORS	hourly wet deposition values	kg/hectare
ASEAS	hourly wet deposition values	kg/hectare
ASOIL	hourly wet deposition values	kg/hectare
NUMATKN	hourly wet deposition values	number/hectare
NUMACC	hourly wet deposition values	number/hectare
NUMCOR	hourly wet deposition values	number/hectare
NH3	hourly wet deposition values	kg/hectare
HPLUS	hourly wet deposition values	kg/hectare
PLCL	pressure at lifting condensation level, or MISSING = -9999.9	Pascal
CLOUD_BOTTOM	layer containing bottom of cloud, or MISSING = -9999.9	layer-number
NP_CLOUDTOP	layer containing top of NP cloud, or MISSING = -9999.9	layer-number
RAIN_FLAG	Rain-event flag	1 or 0

xi. CTM_WET_DEP_2

Same as above, but with additional species.

xii. FLOOR_FILE

The subroutine SCIPROC, called from DRIVER, calls each physical process in the time-splitting paradigm used in the CCTM. After each physical process call, SCIPROC calls a routine named CKSUMMER that provides a crude diagnostic measure of the concentration field for this run log file.

Due to round-off or other numerical errors, concentration values my fall below zero in some cases, although this should be extremely rare. For those cases, CKSUMMER resets the concentration values to zero, and records pertinent information in the FLOOR_FILE; namely, for which process the event occurred, the scenario date and time, the grid cell and species, and the value before it was reset. This information may be useful for debugging problems.

An example FLOOR_FILE:

Concentrations less than, but reset to 0.000E+00 in ZADV.

date:time	Col	Row	Layer	Species	Value before reset
1995192:020000	33	13	4	65	-2.457E-35

5.0 Emissions-Chemistry Interface Processor (ECIP)

a. Input Files

i. EMIS_AREA

See OUTPRO output file SPEMIS2D. This contains 2D speciated emissions (moles).

ii. EMIS_MAJOR

Optional; See OUTPRO output file CRO_SP_MAJOR. Used frequently when major point sources are being considered. Otherwise all sources are included in the area source file .

iii. EMIS_MEPSE

Optional; See OUTPRO output file CRO_SP_MEPSE. Used when PING will be used with CCTM.

iv.	**GRID_CRO_2D**	See 8.b.iii	MCIP Output file GRID_CRO_2D
v.	**GRID_DOT_2D**	See 8.b.v	MCIP Output file GRID_DOT_2D
vi.	**MET_CRO_2D**	See 8.b.viii	MCIP Output file MET_CRO_2D
vii.	**MET_CRO_3D**	See 8.b.ix	MCIP Output file MET_DOT_3D

viii. **MET_DOT_3D** See 8.b.x MCIP Output file MET_DOT_3D

ix. **STACK_MAJOR**

Optional; See OUTPRO output file STAK_MAJOR. Used when separate major stack emissions are input to CCTM.

x. **STACK_MEPSE**

Optional; See OUTPRO output file STAK_MEPSE. Used when PING is used within CCTM.

b. Output Files

i. EMIS_1

Estimated emissions rates formatted for CCTM (optionally units can be specified in grams/second for use with other models)

VARIABLE NAME	DESCRIPTION	UNITS
ALD	Emission Rate for ALD	moles/s
CO	Emission Rate for CO	moles/s
CSL	Emission Rate for CSL	moles/s
ETH	Emission Rate for ETH	moles/s
HC3	Emission Rate for HC3	moles/s
HC5	Emission Rate for HC5	moles/s
HC8	Emission Rate for HC8	moles/s
HCHO	Emission Rate for HCHO	moles/s
ISO	Emission Rate for ISO	moles/s
KET	Emission Rate for KET	moles/s
NO	Emission Rate for NO	moles/s
NO2	Emission Rate for NO2	moles/s
OL2	Emission Rate for OL2	moles/s

VARIABLE NAME	DESCRIPTION	UNITS
OLI	Emission Rate for OLI	moles/s
OLT	Emission Rate for OLT	moles/s
ORA2	Emission Rate for ORA2	moles/s
SO2	Emission Rate for SO2	moles/s
SULF	Emission Rate for SULF	moles/s
TOL	Emission Rate for TOL	moles/s
XYL	Emission Rate for XYL	moles/s
TERPB	Emission Rate for TERPB	moles/s
PM2_5	Emission Rate for PM2_5	g/s
PM10	Emission Rate for PM10	g/s
NH3	Emission Rate for NH3	moles/s

ii. NFOUT1

This is one of two diagnostic Output files generated during an ECIP simulation if the user selects the print option flag, IPRFLG = .TRUE. , in the run script. This file contain selected results of the computations performed for each point source. These output files may become large, so as a default the user may wish not to produce these files.

The NFOUT1 diagnostic print file is generated with unit 25 containing various parameters computed in the plume rise subroutines PR and PLUMER. A sample of the output in this file is provide below.

This print is from PLUMER in the NFOUT1 file;

```
STABLE: 1 3   65.0643   8.13666E-02  0.519198   1.00000E-05
NEUTRAL SELECTED; 2   2.09363
 0 1 2  -1.17608E-02   6.09600   2.09363   8.18963   7.14282   9.23645
STABLE: 1 3   81.1974  0.379495   1.24594   1.00000E-05
NEUTRAL SELECTED; 2   2.51280
 0 1 2  -1.18217E-02   13.1064   2.51280   15.6192   14.3628   16.8756
```

```
STABLE: 1 3   183.797   4.40143   1.24594   1.00000E-05
NEUTRAL SELECTED; 2   13.1601
 0 1 2  -1.18217E-02   13.1064   13.1601   26.2665   19.6865   32.8466
STABLE: 1 3   160.228   2.61260   1.11627   1.00000E-05
NEUTRAL SELECTED; 2   18.0713
 0 1 2  -1.17608E-02   13.1064   18.0713   31.1777   22.1420   40.2133
```

The following print is also part of the NFOUT1 file and is written from subroutine PR at each hour interval. It shows the number of plumes, the total fractional amounts of plumes, and the normalized fractional amount in each layer.

```
 NOTE: STATS AT END OF PR
LAYER=   1 FREQ= 1304 TOTAL=  940.92773   0.57832
LAYER=   2 FREQ=  824 TOTAL=  655.50891   0.40289
LAYER=   3 FREQ=   39 TOTAL=   30.56367   0.01879
LAYER=   4 FREQ=    0 TOTAL=    0.00000   0.00000
LAYER=   5 FREQ=    0 TOTAL=    0.00000   0.00000
LAYER=   6 FREQ=    0 TOTAL=    0.00000   0.00000
```

iii. NFOUT2

This diagnostic print file created on unit 11 with logical name NFOUT2 contains the following information from each hour;

Record 1: JJDATE, JJTIME where
JJDATE = current date (YYYYJJJ)
JJTIME = current model hour (HHMMSS)

Record 2: MMS IR, MEPS, TOP, BOT, DPTH, ZCN, IPLUM, JPLUM, ISTKFLG where
IR = counter of simulation hours
MEPS = point source number from stack parameter file
TOP = height of the plume top after plume rise (meters)
BOT = height of plume bottom after plume rise (meters)
DPTH = thickness of the plume (meters)
ZCN = effective plume rise height of plume centerline (meters)
IPLUM = grid column of point source location
JPLUM = grid row of point source location
ISTKFLG = status flag of point source inside (0) or outside domain(other)

An example of a few lines of print from this file generated from subroutine OUTP are given below.

```
1995192     0
 1   18   9.2   7.1   2.1   8.2 23 17  0
 1  334  16.9  14.4   2.5  15.6 22 16  0
 1  336  32.8  19.7  13.2  26.3 22 16  0
 1  419  40.2  22.1  18.1  31.2 23 17  0
 1  420  40.2  22.1  18.1  31.2 23 17  0
 1  471 363.8 192.4 171.4 278.1 23 17  0
 1  472 363.8 192.4 171.4 278.1 23 17  0
 1  491 346.2 216.6 129.6 281.4 23 17  0
 1  566 213.2 105.0 108.2 159.1 22 16  0
 1  567 239.5 113.8 125.7 176.6 22 16  0
 1 1009 254.0 118.6 135.4 186.3 24 19  0
 1 1010 254.0 118.6 135.4 186.3 24 19  0
```

6.0 Photolysis Processor (JPROC)

a. Input Files

i. ET(ASCII)

The ET file contains wavelength dependent extra-terrestrial irradiance values (or the radiation incident at the top of the atmosphere). Irradiance values can be specified on any wavelength grid. The wavelength spacing between values can be irregular. The file is made up of several record types, including comments, data type, factor, and data records. At the beginning of the file, there can be any number of comment records. A comment record is distinguished by "!" in column 1. The data type record follows the initial comment records. The data type record provides information about how the data values are registered in relation to the wavelength. Currently, there are only four recognized data types: 1) point values (P) where irradiance data are given at an actual point; 2) centered values (C) where the irradiance value represents an interval average centered on the specified wavelength; 3) beginning values (B) where the irradiance values is also an interval average for an interval beginning at the specified wavelength; and 4) ending values (E) where the irradiance values are interval averages for the interval ending at the specified wavelength. Another record type is the factor record, which simply gives a factor that should be used to multiply the irradiance value to get it into proper units. The last record type is the data record, which contains the wavelength and irradiance values.

Sample FORTRAN code to read file:

```
101   CONTINUE
      READ( ETUNIT, 1003, IOSTAT = IOST ) TYPE
1003  FORMAT( A1 )
      IF ( TYPE .EQ. '!' ) GO TO 101
      READ( ETUNIT, 1005, IOSTAT = IOST ) FACTOR
1005  FORMAT( /, 4X, F10.1 )
      IWL = 0
201   CONTINUE
        IWL = IWL + 1
        READ( ETUNIT, *, IOSTAT = IOST ) WLIN( IWL ), F( IWL )
        F( IWL ) = F( IWL ) * FACTOR
      IF ( IOST .EQ. 0 ) GO TO 201
```

ET irradiance.data Variables

Parm No.	Variable Name	Units	Data Type	Description
1	TYPE		Character*1	type of data registration (P,C,B,E)
2	FACTOR		Real	Factor to be applied to the data
3	WLIN(iwl)	nm	Real	wavelength
4	F(iwl)	Photons cm-2 s-1	Real	Irradiance for the given wavelength

Sample file:

```
! Extra Terrestrial Irradiance
! taken from the RADM data---derived from the WMO 1985 report Table 7-4
! format:  wl, et_irrad
Beginning
! With FAC, units are (Photons cm-2 s-1)
FAC=1.0
 185.185    3.620E+11
 186.916    4.730E+11
 188.679    5.610E+11
```

```
 190.476    6.630E+11
 192.308    6.900E+11
 194.175    9.560E+11
...
 705.000    5.010E+15
 715.000    4.930E+15
 725.000    4.910E+15
 735.000    4.810E+15
```

ii. PROFILES

This file contains several vertical Profiles that are needed by the program JPROC. These profiles were originally used in the photolysis rate calculations in RADM. The file itself can be subdivided into three sections. The first section contains seasonal profiles of ozone, temperature, and air concentration for 19 different latitudinal bands (latitudinally-dependent data are indexed 1 through 19 such that 1 = 90N, 10 = Equator, 19 = 90S. The temperature and air concentration profiles are from Louis (Ph.D. thesis 1974 University of Colorado), and the ozone profiles are from Isaksen et al. diabatic 2D model. The second section of the file contains monthly ozone concentrations in Dobson units for the 19 latitudinal bands. These data were taken from the WMO (1981) report, which originally came from Deutch 1971. The third section of the file contains standard profiles for air concentration, temperature, ozone (Nicolet et al. 1982 Planet.Space Sci. v30 p935), and aerosols (Elterman 1968 AFCRL-68-0153 report Air Force Cambridge Labs). Only the standard aerosol profile is used in the current implementation since local profiles are interpolated from the latitudinal and seasonal profiles. However, these standard profiles can be used if subroutine INTERP is not invoked. Seasonal averages are centered around: ISEA=1- winter February 4; ISEA=2 - spring May 6; ISEA=3 - summer August 6; ISEA=4 - fall November 6. MXLEV = 51.

Sample FORTRAN code to read file:

```
      DO 70 ISEA = 1, 4
        DO ILAT = 1, 19
          READ( PFUNIT, 1001, IOSTAT = IOST )
     &        ( XO3  ( ISEA, ILAT, ILEV ), ILEV = 1, MXLEV )
1001      FORMAT( 8E10.3 )
        END DO
        DO ILAT = 1, 19
          READ( PFUNIT, 1001, IOSTAT = IOST )
     &        ( XT   ( ISEA, ILAT, ILEV ), ILEV = 1, MXLEV )
        END DO
        DO ILAT = 1, 19
          READ( PFUNIT, 1001, IOSTAT = IOST )
     &        ( XAIR( ISEA, ILAT, ILEV ), ILEV = 1, MXLEV )
        END DO
70    CONTINUE
```

```
 DO IMON = 1, 12
   READ( PFUNIT, *, IOSTAT = IOST ) ( XDOBS( ILAT, IMON ),
&                                                 ILAT = 1, 19 )
 END DO
 DO ILEV = 1, MXLEV
   READ( PFUNIT, *, IOSTAT = IOST ) T ( ILEV ), AIR( ILEV ),
&                                   O3( ILEV ), AER( ILEV )
 END DO
```

PROFILES.data Variables

Parm No.	Variable Name	Units	Type	Description
1	XO3	molecules/cm3	real	seasonal, latitudinal, vertical ozone profiles
2	XT	K	real	seasonal, latitudinal, vertical temperature profiles
3	XAIR	molecules/cm3	real	seasonal, latitudinal, vertical air concentration profiles
4	XDOBS	Dobson Units	real	latitudinal, seasonal ozone values
5	T	K	real	temperature standard profile
6	AIR	molecules/cm3	real	air concentration (number density) profile
7	O3	molecules/cm3	real	ozone standard profile
8	AER		real	aerosol attenuation profile

Sample file

```
6.212E+11 5.395E+11 4.577E+11 3.752E+11 3.552E+11 4.943E+11 6.921E+11 7.150E+11
1.060E+12 2.424E+12 3.976E+12 5.058E+12 5.843E+12 6.207E+12 6.388E+12 6.629E+12
6.792E+12 6.839E+12 6.704E+12 6.202E+12 5.645E+12 5.389E+12 5.106E+12 4.484E+12
3.823E+12 3.329E+12 2.967E+12 2.825E+12 2.745E+12 2.597E+12 2.391E+12 2.062E+12
1.712E+12 1.433E+12 1.192E+12 9.912E+11 8.233E+11 6.832E+11 5.658E+11 4.645E+11
3.792E+11 3.081E+11 2.499E+11 2.039E+11 1.664E+11 1.341E+11 1.056E+11 7.645E+10
5.293E+10 4.513E+10 3.734E+10
6.103E+11 5.330E+11 4.557E+11 3.833E+11 3.552E+11 4.279E+11 5.548E+11 6.060E+11
8.875E+11 1.806E+12 2.930E+12 3.992E+12 4.902E+12 5.378E+12 5.723E+12 6.236E+12
6.644E+12 6.757E+12 6.610E+12 6.035E+12 5.438E+12 5.334E+12 5.228E+12 4.675E+12
4.036E+12 3.534E+12 3.128E+12 2.886E+12 2.704E+12 2.503E+12 2.276E+12 1.967E+12
1.649E+12 1.387E+12 1.159E+12 9.699E+11 8.106E+11 6.759E+11 5.625E+11 4.658E+11
3.834E+11 3.101E+11 2.488E+11 2.028E+11 1.662E+11 1.346E+11 1.071E+11 7.930E+10
5.662E+10 4.765E+10 3.867E+10
5.899E+11 5.194E+11 4.490E+11 3.891E+11 3.546E+11 3.613E+11 4.173E+11 4.982E+11
7.171E+11 1.188E+12 1.883E+12 2.930E+12 3.970E+12 4.563E+12 5.078E+12 5.874E+12
6.535E+12 6.721E+12 6.561E+12 5.901E+12 5.254E+12 5.304E+12 5.380E+12 4.901E+12
4.287E+12 3.778E+12 3.325E+12 2.979E+12 2.690E+12 2.432E+12 2.181E+12 1.891E+12
1.606E+12 1.361E+12 1.146E+12 9.650E+11 8.117E+11 6.814E+11 5.709E+11 4.774E+11
3.960E+11 3.188E+11 2.529E+11 2.057E+11 1.690E+11 1.377E+11 1.107E+11 8.358E+10
6.117E+10 5.077E+10 4.037E+10
.
360   360.  360.  370.  360.  320.  290.  260.  250.  250. .255.  265.  285.  285.  310.  320.
340   320.  300.
390.  390.  390.  360.  375.  340.  305.  270.  255.  250.  255.  265.  275.  285.  290.  295.
315.  300.  295.

288     2.55E19 8.00E11 2.40E-1
282     2.31E19 7.39E11 1.06E-1
275     2.09E19 6.90E11 4.56E-2
...
```

iii. O2ABS

The O2ABS file contains wavelength dependent Absorption cross section data for molecular oxygen (O2). These data can be specified on any wavelength grid. The wavelength spacing between values can be irregular. The file is made up of several record types, including identification comments, data type, factor, and data records. At the beginning of the file, there is an identification record. Next, there can be any number of comment records. A comment record is distinguished by "!" in column 1. The data type record follows the comment records. The data type record provides information about how the data values are registered in relation to the wavelength. Currently, there are only four recognized data types: 1) point values (P) where data are given at an actual point; 2) centered values (C) where the value represents an interval average centered on the specified wavelength; 3) beginning values (B) where the values is also an interval average for an interval beginning at the specified wavelength; and 4) ending values (E) where the values are interval averages for the interval ending at the specified wavelength. Another record type is the factor record, which simply gives a factor that should be used to multiply the

absorption cross section values by to get it into the proper units. The last record type is the data record, which contains the wavelength, absorption cross section, and quantum yield values.

Sample FORTRAN code to read file:

```
      READ( O2UNIT, 1001, IOSTAT = IOST ) PHOTID
1001  FORMAT( A16 )
101   CONTINUE
      READ( O2UNIT, 1003, IOSTAT = IOST ) TYPE
1003  FORMAT( A1 )
      IF ( TYPE .EQ. '!' ) GO TO 101
      READ( O2UNIT, 1005, IOSTAT = IOST ) FACTOR
1005  FORMAT( /, 4X, F10.1 )
      IWL = 0
201   CONTINUE
        IWL = IWL + 1
        READ( O2UNIT, *, IOSTAT = IOST ) WLIN( IWL ), CSIN( IWL )
        CSIN( IWL ) = CSIN( IWL ) * FACTOR
      IF ( IOST .EQ. 0 ) GO TO 201
```

Parm num	Variable Name	Units	Type	Description
1	PHOTID		Character*16	identifier for the absorbing molecule
2	TYPE		Character*1	type of data registration (P,C,B,E)
3	FACTOR		Real	Factor to be applied to the data
4	WLIN	nm	Real	wavelength
5	CSIN	cm2/molecule	Real	absorption cross sections

Sample file

```
O2_NASA94
! Molecular Oxygen photolysis (O2)
```

```
! O2 + hv -> O + O
! Taken from NASA (1994); QY was set to 1.0 and is not used in the
! photolysis rate program.
! format:  wl, abs_cs, qy
Centered
! With FAC, units are (cm^2/molecule)
FAC=1.0E-24
  205    7.35    1.0
  206    7.13    1.0
...
  238    1.22    1.0
  239    1.10    1.0
  240    1.01    1.0
```

iv. O3ABS

The O3ABS file contains wavelength dependent Absorption cross section data for ozone (O3). These data can be specified on any wavelength grid. The wavelength spacing between values can be irregular. The file is made up of several record types, including identification comments, data type, factor, and data records. At the beginning of the file, there is an identification record. Next, there can be any number of comment records. A comment record is distinguished by "!" in column 1. The data type record follows the comment records. The data type record provides information about how the data values are registered in relation to the wavelength. Currently, there are only four recognized data types: 1) point values (P) where data are given at an actual point; 2) centered values (C) where the value represents an interval average centered on the specified wavelength; 3) beginning values (B) where the values is also an interval average for an interval beginning at the specified wavelength; and 4) ending values (E) where the values are interval averages for the interval ending at the specified wavelength. Another record type is the factor record, which simply gives a factor that should be used to multiply the absorption cross section values by to get it into the proper units. The last record type is the data record, which contains the wavelength, absorption cross section, and quantum yield values.

Sample FORTRAN code to read file:

```
      READ( O2UNIT, 1001, IOSTAT = IOST ) PHOTID
1001  FORMAT( A16 )
101   CONTINUE
      READ( O2UNIT, 1003, IOSTAT = IOST ) TYPE
1003  FORMAT( A1 )
      IF ( TYPE .EQ. '!' ) GO TO 101
      READ( O2UNIT, 1005, IOSTAT = IOST ) FACTOR
1005  FORMAT( /, 4X, F10.1 )
      IWL = 0
201   CONTINUE
        IWL = IWL + 1
```

```
      READ( O2UNIT, *, IOSTAT = IOST ) WLIN( IWL ), CSIN( IWL )
      CSIN( IWL ) = CSIN( IWL ) * FACTOR
    IF ( IOST .EQ. 0 ) GO TO 201
```

Parm num	Variable Name	Units	Type	Description
1	PHOTID		Character*16	indentifier for the absorbing molecule
2	TYPE		Character*1	type of data registration (P,C,B,E)
3	FACTOR		Real	Factor to be applied to the data
4	WLIN	nm	Real	wavelength
5	CSIN	cm2/molecule	Real	absorption cross sections

Sample file

```
O3O1D_NASA94
! Ozone Photolysis to O1D (O3O1D)
! O3 + hv -> O2 + O(1D)
! Taken from NASA (1994);  Absorption CS is at 273 K;  QY is at 298 K;
! WMO (1988) data used for wl .ge. 362.5 nm
Beginning
! With FAC, units are (cm^2/molecule)
FAC=1.0E-20
 175.439    81.1       0.9
 176.991    79.9       0.9
 178.571    78.6       0.9
...
 727.500     0.0525    0.0
 732.500     0.0475    0.0
 737.500     0.0447    0.0
```

v. TOMS

TOMS Data File Format

Ozone data from the NASA Total Ozone Mapping Spectrometer (TOMS) for each day are gridded into 1 degree latitude zones by 1.25 degree longitude zones. Latitudes go from -90 degrees (the south pole) to 0 degrees (the equator) to +90 degrees (the north pole) in 1 degree steps, so there are 180 latitude zones. The first zone extends from -90 to -89, so the grid cell is considered to be centered on -89.5. Similarly, longitudes go from -180 (west longitude) to 0 (Greenwich, England) to +180 (east longitude) in 1.25 degree steps, so there are 288 longitude zones (360/1.25). The first longitude zone extends from -180 to -178.75, so the grid cell is considered to be centered on -179.375.

As an example, the first few lines of a data file may look like the following:

```
 Day:   22 Jan 22, 1997     EP/TOMS     NRT OZONE     GEN:97.091 Asc LECT: 11:17
AM
  Longitudes:   288 bins  centered on 179.375 W to 179.375 E   (1.25 degree steps)
  Latitudes  :  180 bins  centered on   89.5   S to   89.5   N  (1.00 degree steps)
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    0   0   0   0   0   0   0   0   0   0   0   0   0     lat =   -89.5
 2902902902902902902902902902902902902902902902902902912912912912912912912290
 2902902902902902902902892892892892892892892892892902902902902902902902862 86
 2862862862862862862862862862862892892892892892892902902902902902902902902 90
```

The first three lines are header information which includes the date the data were taken, the instrument, the type of processing (e.g. production, near real time, etc), the generation date of the file, and the local equator crossing time of the satellite.

Then, the 288 longitude values for one latitude zone, centered at -89.5, are given. The next 288 longitude values are given for the latitude zone centered at -88.5, and so on. The zeros denote flagged data, i.e. data that could not be collected due to lack of sunlight or other problems. All measurements are given in Dobson units and are integers with 3 significant figures. A Dobson

Unit represents the physical thickness of the ozone layer if it were brought to the Earth's surface. A value of 300 Dobson units equals three millimeters or 1/10th of an inch.

This format is the same as the format used for the Nimbus-7 and Meteor-3 TOMS CD-ROM's. Code written to read those files may therefore, with a few minor alterations, be used to read these files.

```
      READ( TMUNIT, 1001 ) DDD, YYYY
1001  FORMAT( 6X, I3, 9X, I4 )
      READ( TMUNIT, 1003 ) HEADER
1003  FORMAT( A80 )
      READ( TMUNIT, 1003 ) HEADER
      DO ILAT = 1, NLAT
        READ( TMUNIT, 1005, IOSTAT = IOST ) ( OZONE( ILON, ILAT ),
     &                                        ILON = 1, NLON )
1005    FORMAT( 1X, 25I3 )
      END DO
```

Parm No.	Variable Name	Units	Type	Description
1	DDD		Integer	Julian Date
2	YYYY		Integer	Year
3	HEADER		Character*80	Header Records
4	OZONE	Dobson Units	Integer	Total Ozone Column Values

vi. CSQY Files

The CSQY files contains wavelength dependent absorption Cross Section and Quantum Yield (CSQY) data for various photolysis reactions. These data can be specified on any wavelength grid. The wavelength spacing between values can be irregular. The file is made up of several record types, including identification comments, data type, factor, and data records. At the beginning of the file, there is an identification record. Next, there can be any number of comment records. A comment record is distinguished by "!" in column 1. The data type record follows the comment records. The data type record provides information about how the data values are registered in relation to the wavelength. Currently, there are only four recognized data types: 1) point values (P) where data are given at an actual point; 2) centered values (C) where the value represents an interval average centered on the specified wavelength; 3) beginning values (B) where the values is also an interval average for an interval beginning at the specified

wavelength; and 4) ending values (E) where the values are interval averages for the interval ending at the specified wavelength. Another record type is the factor record, which simply gives a factor that should be used to multiply the absorption cross section values by to get it into the proper units. The last record type is the data record, which contains the wavelength, absorption cross section, and quantum yield values.

Sample FORTRAN code to read file:

```
      READ( O2UNIT, 1001, IOSTAT = IOST ) PHOTID
1001  FORMAT( A16 )
101   CONTINUE
      READ( O2UNIT, 1003, IOSTAT = IOST ) TYPE
1003  FORMAT( A1 )
      IF ( TYPE .EQ. '!' ) GO TO 101
      READ( O2UNIT, 1005, IOSTAT = IOST ) FACTOR
1005  FORMAT( /, 4X, F10.1 )
      IWL = 0
201   CONTINUE
        IWL = IWL + 1
        READ( O2UNIT, *, IOSTAT = IOST ) WLIN( IWL ), CSIN( IWL ), QYIN( IWL )
        CSIN( IWL ) = CSIN( IWL ) * FACTOR
      IF ( IOST .EQ. 0 ) GO TO 201
```

Parm num	Variable Name	Units	Type	Description
1	PHOTID		Character*16	Photolysis reaction identifier
2	TYPE		Character*1	type of data registration (P,C,B,E)
3	FACTOR		Real	Factor to be applied to the data
4	WLIN	nm	Real	Wavelength
5	CSIN	cm2/molecule	Real	Absorption Cross Section
6	QYIN	molecules/photon	Real	Quantum Yield

Sample file

```
HCHOmol_CBIV88
! Formaldehyde Photolysis to molecular hydrogen (HCHOmol)
! HCHO + hv -> H2 + CO
! Taken from Gery et al. (1988); Absorption CS from Bass (1980); QY from
! Calvert (1980) fit to his data and Moortgat and coworkers (1978, 1979).
! format:  wl, abs_cs, qy
Centered
! With FAC, units are (cm^2/molecule)
FAC=1.0E-20
 280  2.34  0.440
 281  1.65  0.420
 282  0.76  0.400
 283  0.46  0.380
...
 355  0.26  0.230
 356  0.05  0.200
 357  0.03  0.160
 358  0.04  0.130
 359  0.03  0.100
```

b. Output Files

i. JVALUES

There is a single output file from processor JPROC. This file contains calculated photolysis rates for various altitudes, latitudes, and hour angles. The file contains several header records for the date, number of levels, level values, number of latitudes, latitude values, number of hour angles, hour angles values, number of photolysis reactions, photolysis reaction ID, and cloud enhancement factor. Following the header records are the data records. The data records give the photolysis rates at different heights, latitudes, and hour angles for each photolysis reaction.

Sample FORTRAN code to read file:

```
      WRITE( JVUNIT, 2013 ) JDT
2013  FORMAT( 3X, I7, 2X, '(yyyyddd) Julian Date for the file' )
      WRITE( JVUNIT, 2015 ) JVHT
2015  FORMAT( 3X, I2, 2X, 'LEVELS (m)' )
      WRITE( JVUNIT, 2017 ) ( XZJV( IHT ), IHT=1, JVHT )
2017  FORMAT( 3X, 30( F7.1, 1X ) )
      WRITE( JVUNIT, 2019 ) JVLAT
2019  FORMAT( 3X, I2, 2X, 'LATITUDES (deg)' )
      WRITE( JVUNIT, 2021 ) ( XLATJV( ILAT ), ILAT=1, JVLAT )
2021  FORMAT( 3X, 30( F5.1, 1X ) )
      WRITE( JVUNIT, 2023 ) JVTMAX
```

```
2023  FORMAT( 3X, I2, 2X, 'HOUR ANGLES (from noon)' )
      WRITE( JVUNIT, 2025 ) ( XHAJV( ITIME ), ITIME=1, JVTMAX )
2025  FORMAT( 3X, 12( F5.1, 1X ) )
      WRITE( JVUNIT, 2027) NPHOTAB
2027  FORMAT( 3X, I2, 2X, 'PHOTOLYTIC REACTIONS' )
      DO IPHOT = 1, NPHOTAB
        WRITE( JVUNIT, 2029 ) PHOTAB(IPHOT), ACLD( IPHOT )
2029    FORMAT( 6X, '''', A16, ''',', 5X, F3.1 )
      END DO
      DO IHT = 1, JVHT
        DO ILAT = 1, JVLAT
          DO IPHOT = 1, NPHOTAB
            WRITE( JVUNIT, 2031 ) IHT, ILAT, IPHOT
2031        FORMAT( 1X, 3( I3, 1X ) )
            WRITE( JVUNIT, 2033 ) ( XJVAL( IPHOT, ITIME, ILAT, IHT ),
     &                              ITIME = 1, JVTMAX )
2033        FORMAT( 1X, 1P, 5( E13.7, 2X ) )
            END DO
          END DO
        END DO
```

Parm No.	Variable Name	Units	Type	Description
1	JDT		Integer	Julian Date (yyyyddd)
2	JVHT		Integer	number of vertical levels on file
3	XZJV	m	Real	level heights
4	JVLAT		Integer	number of latitude bands on file
5	XLATJV	degrees	Real	latitude bands
6	JVTMAX		Integer	number of hour angles on file
7	XHAJV	hours	Real	hour angles

Parm No.	Variable Name	Units	Type	Description
8	NPHOTAB		Integer	number of photolysis reactions on file
9	PHOTAB		Character*16	photolysis reaction identifier
10	ACLD		Real	reaction dependent above cloud enhancement factor
11	IHT		Integer	level index
12	ILAT		Integer	latitude index
13	IPHOT		Integer	photolysis reaction index
14	XJVAL	min-1	Real	Photolysis rates

7.0 Land-Use Processor (LUPROC)

a. Input Files

i. LANDUSE

High resolution USGS North America land cover characteristics data base adapted to CMAQ's base map projection which uses Lambert conformal projection with origins at latitude 40 degrees North and longitude 90 degrees West. The original database has 1-km nominal spatial resolution and is based on 1-km AVHRR data spanning April 1992 through March 1993. (See **http://edcwww.cr.usgs.gov/landdaac/glcc/nadoc1_1.html**). The original data base has been converted to 4 km resolution for use in CMAQ.

Soil Texture Area Data for 4km MM5 Grid

Rows: 1080
Columns: 1620
Grid Lower-Left Corner X and Y: -3240000.000 -2160000.000

Grid Upper-Right Corner X and Y: 3240000.000 2160000.000

Coordinate System Description:
Projection: Lambert Conformal Conic Sphere with R = 6370997m
Units: Meters
1st standard parallel: 30 0 0.000
2nd standard parallel: 60 0 0.000
central meridian: -90 0 0.000
false easting (meters): 0.00000
false northing (meters): 0.00000

Data formats (FORTRAN) for each cell:

I5, 1X, I5	for COL number and ROW number
F13.3, 1X, I6	for TOTAREA (hectare) of cell and NTYPES # of types in cell
12X, A4, 1X, F13.3	for Soil Texture Name and AREA (hectare)

where possible Soil Texture names (up to 4 characters) and descriptions are:

Urbn	Urban or Built-up Land
DryC	Dryland Cropland and Pasture
IrrC	Irrigated Cropland and Pasture
MxDI	Mixed Dryland/Irrigated Cropland and Pasture
GrMo	Grassland/Cropland Mosaic
WdMo	Woodland/Cropland Mosaic
Grss	Grassland
Shrb	Shrubland
MxGr	Mixed Shrubland/Grassland
Chap	Chaparral
Svna	Savanna
BrFr	Broadleaf Deciduous Forest
EvFr	Evergreen Coniferous Forest (SE removed)
SbFr	Subalpine Forest
MxFr	Mixed forest
DCFr	Deciduous Coniferous Forest
EBFr	Evergreen Broadleaf Forest (Palm?)
H2O	Water
HbWt	Herbaceous Wetland (none in east)
FrWt	Forested Wetlands (e.g. Everglades)
Brrn	Barren or Sparsely Vegetated
BrTu	Shrub and Brush Tundra
HrTu	Herbaceous Tundra

BGTu	Bare Ground Tundra
WtTu	Wet Tundra
MxTu	Mixed Tundra
Glac	Perennial Snowfields or Glaciers

Example file:

```
  1    1
   1600.000     1
 H2O      1600.000
  2    1
   1600.000     1
 H2O      1600.000
  3    1
   1600.000     1
 H2O      1600.000
  4    1
   1600.000     1
 H2O      1600.000
  5    1
   1600.000     1
 H2O      1600.000
  6    1
   1600.000     1
 H2O      1600.000
  7    1
   1600.000     1
 H2O      1600.000
  8    1
   1600.000     1
 H2O      1600.000
  9    1
```

b. Output Files

i. LAND_CRO_2D

LUPROC windows out the land-use data for the user-defined domain and converts percentages of 27 vegetation categories in the LANDUSE database into RADM's 11-category fractions (1-urban land, 2-agricultural land, 3-range land, 4- deciduous forest, 5-coniferous forest, 6-mixed

forest including wetland, 7-water, both salt and fresh, 8-barren land, mostly desert, 9-nonforested wetland, 10- mixed agricultural and range land, 11-rocky open areas with low growing shrubs). The output file has the same resolution as the CCTM domain and the LUPROC domain should include the boundary cells in addition to the CCTM domain.

Gridded Land-use Fractions

VARIABLE NAME	DESCRIPTION	UNITS
LUSE_URBA	land-use fraction for category_URBAN_LAND	fraction
LUSE_AGRI	land-use fraction for category_AGRICULTURE	fraction
LUSE_RANG	land-use fraction for category_RANGE	fraction
LUSE_DECI	land-use fraction for category_DECIDUOUS_ FOREST	fraction
LUSE_CONI	land-use fraction for category_CONIFEROUS_ FORES	fraction
LUSE_MIXF	land-use fraction for category_MIXFOREST_ WETLND	fraction
LUSE_WATE	land-use fraction for category_WATER	fraction
LUSE_BARR	land-use fraction for category_BARREN_LAND	fraction
LUSE_NUNF	land-use fraction for category_NONFOREST_WETLND	fraction
LUSE_MIXA	land-use fraction for category_MIXAGRI_RANGE	fraction
LUSE_ROCK	land-use fraction for category_ROCKY_OPENSHRUB	fraction

8.0 Meteorological-Chemistry Interface Processor

a. Input Files

i. LAND_CRO_2D See 7.b.i LUPROC output file LAND_CRO_2D

ii. MM51 (IEEE binary) See 9.b.i MM5v2 output file.

b. Output Files

i. GRID_BDY_2D_G1

Calculated time independent 2D meteorological and geographic data in boundary cells of grid

VARIABLE NAME	DESCRIPTION	UNITS
LAT	latitude (south negative)	DEGREES
LON	longitude (west negative)	DEGREES
MSFX2	squared map-scale factor (CROSS)	(M/M)**2
HT	terrain elevation	M
ZZERO	roughness length	M
PSTAR0	pressure difference between top and surface	Pascal
JACOBS0	total Jacobian at surface	M
LUSE_URBA	land-use fraction for category_URBAN_LAND	fraction
LUSE_AGRI	land-use fraction for category_AGRICULTURE	fraction
LUSE_RANG	land-use fraction for category_RANGE	fraction
LUSE_DECI	land-use fraction for category_DECIDUOUS_FOREST	fraction

VARIABLE NAME	DESCRIPTION	UNITS
LUSE_CONI	land-use fraction for category_CONIFEROUS_FORES	fraction
LUSE_MIXF	land-use fraction for category_MIXFOREST_WETLND	fraction
LUSE_WATE	land-use fraction for category_WATER	fraction
LUSE_BARR	land-use fraction for category_BARREN_LAND	fraction
LUSE_NONF	land-use fraction for category_NONFOREST_WETLND	fraction
LUSE_MIXA	land-use fraction for category_MIXAGRI·RANGE	fraction
LUSE_ROCK	land-use fraction for category_ROCKY_OPENSHRUB	fraction

ii. GRID_BDY_3D_G1

Calculated time independent 3D meteorological and geographic data in boundary cells of grid

VARIABLE NAME	DESCRIPTION	UNITS
DENS0	density of reference atmosphere	KG/M**3
ENTRP0	entropy density of reference atmosphere	KG/M**3
JACOB0F	total Jacobian at surface	M
JACOB0M	total Jacobian at layer middle	M
TEMP0	temperature of reference atmosphere	K
PRES0	pressure of reference atmosphere	Pascal
X3HT0F	height of layer face (top) above ground	M
X3HT0M	height of layer middle above ground	M

iii. GRID_CRO_2D_G1

Calculated time independent 2D meteorological and geographic data at cross points of grid

VARIABLE NAME	DESCRIPTION	UNITS
LAT	latitude (south negative)	DEGREES
LON	longitude (west negative)	DEGREES
MSFX2	squared map-scale factor (CROSS)	(M/M)**2
HT	terrain elevation	M
ZZERO	roughness length	M
PSTAR0	pressure difference between top and surface	Pascal
JACOBS0	total Jacobian at surface	M
LUSE_URBA	land-use fraction for category_URBAN_LAND	fraction
LUSE_AGRI	land-use fraction for category_AGRICULTURE	fraction
LUSE_RANG	land-use fraction for category_RANGE	fraction
LUSE_DECI	land-use fraction for category_DECIDUOUS_FOREST	fraction
LUSE_CONI	land-use fraction for category_CONIFEROUS_FORES	fraction
LUSE_MIXF	land-use fraction for category_MIXFOREST_WETLND	fraction
LUSE_WATE	land-use fraction for category_WATER	fraction
LUSE_BARR	land-use fraction for category_BARREN_LAND	fraction
LUSE_NONF	land-use fraction for category_NONFOREST_WETLND	fraction

VARIABLE NAME	DESCRIPTION	UNITS
LUSE_MIXA	land-use fraction for category_MIXAGRI_RANGE	fraction
LUSE_ROCK	land-use fraction for category_ROCKY_OPENSHRUB	fraction

iv. GRID_CRO_3D_G1

Calculated time independent 3D meteorological and geographic data at cross points of grid

VARIABLE NAME	DESCRIPTION	UNITS
DENS0	density of reference atmosphere	KG/M**3
ENTRP0	entropy density of reference atmosphere	KG/M**3
JACOB0F	total Jacobian at surface	M
JACOB0M	total Jacobian at layer middle	M
TEMP0	temperature of reference atmosphere	K
PRES0	pressure of reference atmosphere	Pascal
X3HT0F	height of layer face (top) above ground	M
X3HT0M	height of layer middle above ground	M

v. GRID_DOT_2D_G1

Calculated time independent 2D meteorological and geographic data at dot points of grid

VARIABLE NAME	DESCRIPTION	UNITS
PSTAR	pressure difference (PSURF - PTOP)	Pascal
JACOBS	total Jacobian at surface	M
DENSAS	air density at surface	KG/M**3
USTAR	cell averaged friction velocity	M/S
WSTAR	convective velocity scale	M/S

VARIABLE NAME	DESCRIPTION	UNITS
RIB	bulk Richardson number	M
PBL	PBL height	M
ZRUF	surface roughness length	M
MOLI	inverse of Monin-Obukhov length	1/M
HFX	sensible heat flux	WATTS/M**2
QFX	latent heat flux	WATTS/M**2
RADYNI	inverse of aerodynamic resistance	M/S
RBNDYI	inverse laminar boundary layer resistance	M/S
RSTOMI	bulk stomatal resistance for water	M/S
TEMPG	skin temperature at ground	K
TEMP10	air temperature at 10 m	K
TEMP1P5	air temperature at 1.5 m	K
SURF2	surface parameter 1 currently not used	
ALBEDO	surface albedo	NODIM
FSOIL	heat flux in soil layers	WATTS/M**2
GLW	longwave radiation at ground	WATTS/M**2
GSW	solar radiation absorbed at ground	WATTS/M**2
RGRND	solar radiation reaching surface	WATTS/M**2
RNET	net radiation	WATTS/M**2
RN	nonconvective precipitation per met TSTEP	CM
RC	convective precipitation per met TSTEP	CM
CFRACH	fraction of high cloud	FRACTION
CFRACM	fraction of middle cloud	FRACTION

VARIABLE NAME	DESCRIPTION	UNITS
CFRACL	fraction of low cloud	FRACTION
CFRAC	total cloud fraction	FRACTION
CLDT	cloud top layer height (m)	M
CLDB	cloud bottom layer height (m)	M
WBAR	average liquid water content of cloud	G/M**3
VD_SO2	deposition velocity for species SO2	M/S
VD_SULF	deposition velocity for species SULF	M/S
VD_NO2	deposition velocity for species NO2	M/S
VD_NO	deposition velocity for species NO	M/S
VD_O3	deposition velocity for species O3	M/S
VD_HNO3	deposition velocity for species HNO3	M/S
VD_H2O2	deposition velocity for species H2O2	M/S
VD_ALD	deposition velocity for species ALD	M/S
VD_HCHO	deposition velocity for species HCHO	M/S
VD_OP	deposition velocity for species OP	M/S
VD_PAA	deposition velocity for species PAA	M/S
VD_ORA	deposition velocity for species ORA	M/S
VD_NH3	deposition velocity for species NH3	M/S
VD_PAN	deposition velocity for species PAN	M/S
VD_HONO	deposition velocity for species HONO	M/S
VD_CO	deposition velocity for species CO	M/S

vi. MET_BDY_2D_G1

Calculated time dependent 2D meteorological and geographic data in boundary grid cells

VARIABLE NAME	DESCRIPTION	UNITS
PSTAR	pressure difference (PSURF - PTOP)	Pascal
JACOBS	total Jacobian at surface	M
DENSAS	air density at surface	KG/M**3
USTAR	cell averaged friction velocity	M/S
WSTAR	convective velocity scale	M/S
RIB	bulk Richardson number	M
PBL	PBL height	M
ZRUF	surface roughness length	M
MOLI	inverse of Monin-Obukhov length	1/M
HFX	sensible heat flux	WATTS/M**2
QFX	latent heat flux	WATTS/M**2
RADYNI	inverse of aerodynamic resistance	M/S
RBNDYI	inverse laminar boundary layer resistance	M/S
RSTOMI	bulk stomatal resistance for water	M/S
TEMPG	skin temperature at ground	K
TEMP10	air temperature at 10 m	K
TEMP1P5	air temperature at 1.5 m	K
SURF2	surface parameter 1 currently not used	
ALBEDO	surface albedo	NODIM
FSOIL	heat flux in soil layers	WATTS/M**2
GLW	longwave radiation at ground	WATTS/M**2

GSW	solar radiation absorbed at ground	WATTS/M**2
RGRND	solar radiation reaching surface	WATTS/M**2
RNET	net radiation	WATTS/M**2
RN	nonconvective precipitation per met TSTEP	CM
RC	convective precipitation per met TSTEP	CM
CFRACH	fraction of high cloud	FRACTION
CFRACM	fraction of middle cloud	FRACTION
CFRACL	fraction of low cloud	FRACTION
CFRAC	total cloud fraction	FRACTION
CLDT	cloud top layer height (m)	M
CLDB	cloud bottom layer height (m)	M
WBAR	average liquid water content of cloud	G/M**3
VD_SO2	deposition velocity for species SO2	M/S
VD_SULF	deposition velocity for species SULF	M/S
VD_NO2	deposition velocity for species NO2	M/S
VD_NO	deposition velocity for species NO	M/S
VD_O3	deposition velocity for species O3	M/S
VD_HNO3	deposition velocity for species HNO3	M/S
VD_H2O2	deposition velocity for species H2O2	M/S
VD_ALD	deposition velocity for species ALD	M/S
VD_HCHO	deposition velocity for species HCHO	M/S
VD_OP	deposition velocity for species OP	M/S
VD_PAA	deposition velocity for species PAA	M/S
VD_ORA	deposition velocity for species ORA	M/S

VD_NH3	deposition velocity for species NH3	M/S
VD_PAN	deposition velocity for species PAN	M/S
VD_HONO	deposition velocity for species HONO	M/S
VD_CO	deposition velocity for species CO	M/S

vii. MET_BDY_3D_G1

Calculated time dependent 3D meteorological and geographic data in boundary grid cells

VARIABLE NAME	DESCCRIPTION	UNITS
JACOBF	total Jacobian at layer face	M
JACOBM	total Jacobian at middle layer	M
DENSA_J	J weighted total air density	KG/M**2
DENSW_J	J weighted density of vapor	KG/M**2
ENTRP_J	J weighted entropy of moist air	KG/M**2
WHAT_JD	J & Density weighted vertical contra-W	KG/(M*S)
QC	cloud water mixing ratio	KG/KG
QR	rain water mixing ratio	KG/KG
TA	air temperature	K
QV	water vapor mixing ratio	KG/KG
PRES	pressure	Pa
DENS	total density of air	KG/M**3
WWIND	true W component of wind	M/S
ZH	mid-layer height above ground	M
ZF	full-layer height above ground	M
JDRATE	time rate change of Jacob*Density	KG/M**2/S
LAMDA	Lagrangian multiplier	None
MCONERR	Mass consistency error	KG/M**2/

viii. MET_CRO_2D_G1

Calculated time dependent 2D meteorological and geographic data at cross points in grid

VARIABLE NAME	DESCCRIPTION	UNITS
PSTAR	pressure difference (PSURF - PTOP)	Pascal
JACOBS	total Jacobian at surface	M
DENSAS	air density at surface	KG/M**3
USTAR	cell averaged friction velocity	M/S
WSTAR	convective velocity scale	M/S
RIB	bulk Richardson number	NODIM
PBL	PBL height	M
ZRUF	surface rouhness length	M
MOLI	inverse of Monin-Obukhov length	1/M
HFX	sensible heat flux	WATTS/M**2
QFX	latent heat flux	WATTS/M**2
RADYNI	inverse of aerodynamic resistance	M/S
RBNDYI	inverse laminar boundary layer resistance	M/S
RSTOMI	bulk stomatal resistance for water	M/S
TEMPG	skin temperature at ground	K
TEMP10	air temperature at 10 m	K
TEMP1P5	air temperature at 1.5 m	K
SURF2	surface parameter 1 currently not used	
ALBEDO	surface albedo	NODIM
FSOIL	heat flux in soil layers	WATTS/M**2
GLW	longwave radiation at ground	WATTS/M**2

VARIABLE NAME	DESCCRIPTION	UNITS
GSW	solar radiation absorbed at ground	WATTS/M**2
RGRND	solar radiation reaching surface	WATTS/M**2
RNET	net radiation	WATTS/M**2
RN	nonconvective precipitation per met TSTEP	CM
RC	convective precipitation per met TSTEP	CM
CFRACH	fraction of high cloud	FRACTION
CFRACM	fraction of middle cloud	FRACTION
CFRACL	fraction of low cloud	FRACTION
CFRAC	total cloud fraction	FRACTION
CLDT	cloud top layer height (m)	M
CLDB	cloud bottom layer height (m)	M
WBAR	average liquid water content of cloud	G/M**3
VD_SO2	deposition velocity for species SO2	M/S
VD_SULF	deposition velocity for species SULF	M/S
VD_NO2	deposition velocity for species NO2	M/S
VD_NO	deposition velocity for species NO	M/S
VD_O3	deposition velocity for species O3	M/S
VD_HNO3	deposition velocity for species HNO3	M/S
VD_H2O2	deposition velocity for species H2O2	M/S
VD_ALD	deposition velocity for species ALD	M/S
VD_HCHO	deposition velocity for species HCHO	M/S
VD_OP	deposition velocity for species OP	M/S
VD_PAA	deposition velocity for species PAA	M/S

VARIABLE NAME	DESCCRIPTION	UNITS
VD_ORA	deposition velocity for species ORA	M/S
VD_NH3	deposition velocity for species NH3	M/S
VD_PAN	deposition velocity for species PAN	M/S
VD_HONO	deposition velocity for species HONO	M/S
VD_CO	deposition velocity for species CO	M/S

ix. MET_CRO_3D_G1

Calculated time dependent 3D meteorological and geographic data at cross points in grid

VARIABLE NAME	DESCCRIPTION	UNITS
JACOBF	total Jacobian at layer face	M
JACOBM	total Jacobian at layer middle	M
DENSA_J	J weighted total air density	KG/M**2
DENSW_J	J weighted density of vapor	KG/M**2
ENTRP_J	J weighted entropy of moist air	KG/M**2
WHAT_JD	J & Density weighted vertical contra-W	KG/(M*S)
QC	cloud water mixing ratio	KG/KG
QR	rain water mixing ratio	KG/KG
TA	air temperature	K
QV	water vapor mixing ratio	KG/KG
PRES	pressure	Pa
DENS	total density of air	KG/M**3
WWIND	true W component of wind	M/S
ZH	mid-layer height above ground	M
ZF	full-layer height above ground	M

VARIABLE NAME	DESCCRIPTION	UNITS
JDRATE	time rate change of Jacob*Density	KG/M**2/S
LAMDA	currently not used	NONE
MCONERR	currently not used	KG/M**2/S

x. MET_DOT_3D_G1

Calculated time dependent 3D meteorological and geographic data at dot points of grid

VARIABLE NAME	DESCCRIPTION	UNITS
UWIND	U-comp. of true wind at dot point	M/S
VWIND	V-comp. of true wind at dot point	M/S
UHAT_JD	(contra_U*Jacobian*Density) at square pt	KG/(M*S)
VHAT_JD	(contra_V*Jacobian*Density) at triangle pt	KG/(M*S)

9.0 Mesoscale Meteorological Model (MM5v2)

a. Input Files

See MM5 User Documentation (References are at the end of Chapter 5).

b. Output Files

i. MM51 (IEEE binary)

Overview

The raw MM5 output files contain model-generated prognostic variables. Those files, MMOUT_DOMAIN*n* (where *n* is a one-digit domain identifier), are in a non-standard binary format that is explained in the chart in this appendix. The output fields listed are repeated for each time period that output is generated. (For example, in a 24-hour simulation where output is generated every hour, the information in this chart would be repeated 25 times in MMOUT_DOMAIN*n*.) The fields that are generated in the output files often depend on the

model physics options that are selected by the user. Sample FORTRAN code to read the MM5 output follows the table.

File Size

Output files from MM5 can tend to get fairly large, sometimes larger than 1-2 gigabytes. Below is a formula that can be used to predict the size of the output file that will be generated from MM5 when run on the EPA Cray ("sequoia"). The inputs to the formula are: number of gridpoints in north-south dimension (*imax*), number of gridpoints in east-west dimension (*jmax*), number of vertical layers (*ksig*), number of output times (*ntimes*), number of 3-D variables (*n3d*), and number of 2-D variables (*n2d*). The output of the formula, *size*, will be the approximate size of the file in bytes.

size = { *ntimes* * [*header* + (*n2d* * *imax* * *jmax* * 8) + (*n3d* * *imax* * *jmax* * *ksig* * 8)] } * 1.002

If the user selects the same physics options as in the Models-3 tutorial, there will be 8 3-D variables and 14 2-D variables. The size of *header* is 3,520,000 bytes. The factor 1.002 is used to account for control information that is included in the output file.

Example 1: User is running a domain with dimensions 82 x 88 x 30, and output is generated hourly for a 120-hour simulation. Thus, *imax* is 82, *jmax* is 88, *ksig* is 30, and *ntimes* is 121 (initial time, plus 120 simulation hours). The user selects the same physics options as in the tutorial, so *n3d* is 8 and *n2d* is 14.

size = { 121 * [3,520,000 + (14 * 82 * 88 * 8) + (8 * 82 * 88 * 30 * 8)] } * 1.002
size = { 121 * [3,520,000 + 808,192 + 13,854,720] } * 1.002
size = { 121 * 18,182,912 } * 1.002
size = 2,200,132,352 * 1.002
size = 2,204,532,617 bytes (or 2.2 Gb). Actual file size is 2,204,463,104 bytes.

Example 2: User is running a 37 x 37 x 30 domain, with 121 output times.

size = { 121 * [3,520,000 + (14 * 37 * 37 * 8) + (8 * 37 * 37 * 30 * 8)] } * 1.002
size = { 121 * [3,520,000 + 153,328 + 2,628,480] } * 1.002
size = { 121 * 6,301,808 } * 1.002
size = 762,518,768 * 1.002
size = 764,043,806 bytes (or 764 Mb). Actual file size is 764,035,072 bytes.

Output Variables

Field_Variable Name	Data Type	Size of Array	Units	Cross or Dot?	Output When...
Header	MIF, MRF, MIFC, MRFC	Integer Real Char*80 Char*80	1000 x 20 1000 x 20 1000 x 20 1000 x 20	N/A	N/A
U-Component Wind coupled with P-Star	U	Real IMAX x JMAX x KSIG	kPa m s^{-1}	Dot	
V-Component Wind coupled with P-Star	V	Real IMAX x JMAX x KSIG	kPa m s^{-1}	Dot	
Temperature coupled with P-Star	T	Real IMAX x JMAX x KSIG	kPa K	Cross	
Specific Humidity coupled with P-Star	QV	Real IMAX x JMAX x KSIG	kPa kg kg^{-1}	Cross	IDRY = 0
Cloud Water Mixing Ratio coupled with P-Star	QC	Real IMAX x JMAX x KSIG	kPa kg kg^{-1}	Cross	IDRY = 0, IMOIST = 2
Rain Water Mixing Ratio coupled with P-Star	QR	Real IMAX x JMAX x KSIG	kPa kg kg^{-1}	Cross	IDRY = 0, IMOIST = 2
Ice Water Mixing Ratio coupled with P-Star	QI	Real IMAX x JMAX x KSIG	kPa kg kg^{-1}	Cross	IDRY = 0, IMOIST = 2, IICE = 1
Snow Water Mixing Ratio coupled with P-Star	QS	Real IMAX x JMAX x KSIG	kPa kg kg^{-1}	Cross	IDRY = 0, IMOIST = 2, IICE = 1
Graupel Mixing Ratio coupled with P-Star	QG	Real IMAX x JMAX x KSIG	kPa kg kg^{-1}	Cross	IDRY = 0, IMOIST = 2, IICEG = 1
Number Concentration of Graupel coupled with P-Star	QNC	Real IMAX x JMAX x KSIG	kPa number m^{-3} J kg^{-1}	Cross	IDRY = 0, IMOIST = 2, IICEG = 1
Turbulent Kinetic Energy	TKE	Real IMAX x JMAX x KSIG	J kg^{-1}	Cross	INAV = 1
Vertical Velocity coupled with P-Star	W	Real IMAX x JMAX x KSIGP1	kPa m s^{-1}	Cross	INHYD = 1
Press. Perturbation coupled with P-Star	PP	Real IMAX x JMAX x KSIG	kPa Pa	Cross	INHYD = 1
P-Star	PSTAR	Real IMAX x JMAX	kPa	Cross	
Ground Temperature	TGRND	Real IMAX x JMAX	K	Cross	ITGFLG = 1
Accumulated Convective Precipitation	PCON	Real IMAX x JMAX	cm	Cross	IDRY = 0
Accumulated Non-Convective Precipitation	PNON	Real IMAX x JMAX	cm	Cross	IDRY = 0
Terrain Elevation	TERR	Real IMAX x JMAX	m	Cross	
Map-Scale Factor	MAPC	Real IMAX x JMAX	dimensionless	Cross	
Map-Scale Factor	MAPD	Real IMAX x JMAX	dimensionless	Dot	

Coriolis Parameter	CORI	Real	IMAX x JMAX	s^{-1}	Dot	
Substrate Temperature	SUBT	Real	IMAX x JMAX	K	Cross	ITGFLG ? 3
Latitude	XLAT	Real	IMAX x JMAX	degree	Cross	
Longitude	XLON	Real	IMAX x JMAX	degree	Cross	
Land-use	LU	Real	IMAX x JMAX	category	Cross	
Snow Cover	SNOW	Real	IMAX x JMAX	dimensionless	Cross	

Sample Program to Read MM5 Output

```
      PROGRAM READMM5

C***  This is a FORTRAN 77 program that reads raw output from MM5.

      IMPLICIT NONE

      INTEGER IMAX
        PARAMETER (IMAX = 77)
      INTEGER IUNIT
        PARAMETER (IUNIT = 12)
      INTEGER JMAX
        PARAMETER (JMAX = 83)
      INTEGER KSIG
        PARAMETER (KSIG = 30)
      INTEGER KSIGP1
        PARAMETER (KSIGP1 = KSIG + 1)
      INTEGER NTIMES
        PARAMETER (NTIMES = 11)

      REAL          CORI    (IMAX, JMAX)
      INTEGER       IDRY
      INTEGER       IICE
      INTEGER       IICEG
      INTEGER       IMOIST
      INTEGER       INHYD
      INTEGER       INAV
      INTEGER       ITGFLG
      REAL          LU      (IMAX, JMAX)
      REAL          MAPC    (IMAX, JMAX)
      REAL          MAPD    (IMAX, JMAX)
      INTEGER       MIF     (1000, 20)
      CHARACTER*80 MIFC     (1000, 20)
      REAL          MRF     (1000, 20)
      CHARACTER*80 MRFC     (1000, 20)
      INTEGER       NN
      REAL          PCON    (IMAX, JMAX)
      REAL          PNON    (IMAX, JMAX)
      REAL          PP      (IMAX, JMAX, KSIG)
```

```
      REAL          PSTAR   (IMAX, JMAX)
      REAL          QC      (IMAX, JMAX, KSIG)
      REAL          QG      (IMAX, JMAX, KSIG)
      REAL          QI      (IMAX, JMAX, KSIG)
      REAL          QR      (IMAX, JMAX, KSIG)
      REAL          QS      (IMAX, JMAX, KSIG)
      REAL          QV      (IMAX, JMAX, KSIG)
      REAL          QNC     (IMAX, JMAX, KSIG)
      REAL          SNOW    (IMAX, JMAX)
      REAL          SUBT    (IMAX, JMAX)
      REAL          T       (IMAX, JMAX, KSIG)
      REAL          TERR    (IMAX, JMAX)
      REAL          TGRND   (IMAX, JMAX)
      REAL          TKE     (IMAX, JMAX, KSIG)
      REAL          U       (IMAX, JMAX, KSIG)
      REAL          V       (IMAX, JMAX, KSIG)
      REAL          W       (IMAX, JMAX, KSIGP1)
      REAL          XLAT    (IMAX, JMAX)
      REAL          XLON    (IMAX, JMAX)

C***  Loop over all times.  Note that fields will be overwritten
C     unless this program is modified to process output.

      DO 1000 NN = 1, NTIMES

C***  Read header of model input.  Verify that IMAX, JMAX, and KSIG
C     are correct.

      READ (IUNIT) MIF, MRF, MIFC, MRFC    ! Header
      IF ( MIF(104,1) .NE. IMAX ) THEN
        PRINT*, 'IMAX IN PARAMETER DOES NOT MATCH IMAX IN HEADER'
        STOP
      ENDIF
      IF ( MIF(105,1) .NE. JMAX ) THEN
        PRINT*, 'JMAX IN PARAMETER DOES NOT MATCH JMAX IN HEADER'
        STOP
      ENDIF
      IF ( NINT(MIF(101,5)) .NE. KSIG ) THEN
        PRINT*, 'KSIG IN PARAMETER DOES NOT MATCH KSIG IN HEADER'
        STOP
      ENDIF

C***  Read physics options from header.

      IDRY   = MIF(3,6)       ! Are moisture processes simulated?
      IMOIST = MIF(4,6)       ! Type of explicit moisture physics
      INHYD  = MIF(5,6)       ! Is this a non-hydrostatic run?
      ITGFLG = MIF(6,6)       ! Ground temperature flag
      IICE   = MIF(7,6)       ! Explicit ice and snow mixing ratios?
```

```
      INAV   = MIF(8,6)       ! Explicit TKE?
      IICEG  = MIF(9,6)       ! Explicit graupel?

C***  Read coupled 3-D forecast variables.

      READ (IUNIT) U          ! U-Component Wind
      READ (IUNIT) V          ! V-Component Wind
      READ (IUNIT) T          ! Temperature

C***  Read coupled 3-D moisture variables.

      IF ( IDRY .NE. 0 ) THEN
        READ (IUNIT) QV       ! Specific Humidity
        IF ( IMOIST .EQ. 2 ) THEN
          READ (IUNIT) QC     ! Cloud Water Mixing Ratio
          READ (IUNIT) QR     ! Rain Water Mixing Ratio
          IF ( IICE .EQ. 1 ) THEN
            READ (IUNIT) QI   ! Ice Water Mixing Ratio
            READ (IUNIT) QS   ! Snow Water Mixing Ratio
          ENDIF
          IF ( IICEG .EQ. 1 ) THEN
            READ (IUNIT) QG   ! Graupel
            READ (IUNIT) QNC  ! Number Concentration
          ENDIF
        ENDIF
      ENDIF

C***  Read coupled PBL variables.

      IF ( INAV .EQ. 1 ) THEN
        READ (IUNIT) TKE      ! Turbulent Kinetic Energy
      ENDIF

C***  Read non-hydrostatic coupled variables.

      IF ( INHYD .EQ. 1 ) THEN
        READ (IUNIT) W        ! Vertical Velocity
        READ (IUNIT) PP       ! Pressure Perturbation
      ENDIF

C***  Read 2-D forecast variables.

      READ (IUNIT) PSTAR
      IF ( ITGFLG .EQ. 1 ) THEN
        READ (IUNIT) TGRND    ! Ground Temperature
      ENDIF
      IF ( IDRY .EQ. 0 ) THEN
        READ (IUNIT) PCON     ! Accumulated Convective Precipitation
        READ (IUNIT) PNON     ! Accumulated Non-Convective Precipitation
```

```
      ENDIF

C***  Read 2-D non-forecast variables.

      READ (IUNIT) TERR      ! Terrain Elevation
      READ (IUNIT) MAPC      ! Map-Scale Factor (cross points)
      READ (IUNIT) MAPD      ! Map-Scale Factor (dot points)
      READ (IUNIT) CORI      ! Coriolis Parameter
      IF ( ITGFLG .NE. 3 ) THEN
        READ (IUNIT) SUBT    ! Substrate Temperature
      ENDIF
      READ (IUNIT) XLAT      ! Latitude
      READ (IUNIT) XLON      ! Longitude
      READ (IUNIT) LU        ! Land-use
      READ (IUNIT) SNOW      ! Snow Cover

 1000 CONTINUE

      END
```

10. Plume Dynamics Model (PDM)

a. Input Files

i.	**STACK_MEPSE**	See 11.b.ii	OUTPRO output file.
ii.	**GRID_CRO_2D**	See 8.b.iii	MCIP output file.
iii.	**GRID_DOT_2D**	See 8.b.v	MCIP output file.
iv.	**MET_CRO_2D**	See 8.b.viii	MCIP output file.
v.	**MET_CRO_3D**	See 8.b.ix	MCIP output file.
vi.	**MET_DOT_3D**	See 8.b.x	MCIP output file.

b. Output Files

i. PDM_PING_1

Plume Dynamic Model calculations provided to grid domain

VARIABLE NAME	DESCRIPTION	UNITS
PLUME_ID	Plume ID	
XC_PLUME	X of plume center	m
YC_PLUME	Y of plume center	m
ZC_PLUME	Z of plume center	m
WIDTH	width of plume center	m
DEPTH	depth of plume center	m
PBL	Planetary boundary layer	m
ANGLE	ARC ANGLE	ARC degrees
WIDTH/DT	plume growth rate	m/s
DEPTH/DT	plume depth growth rate	m/s
DPBL/DT	PBL growth rate	m/s
DDIR	direction shear	ARC degrees
DSPD	speed shear	m/s
ID_LL	grid #- lower left cell	
ID_RU	grid #- upper right cell	
PLUME_FLAG	plume flag #	
INITIAL_WIND	initial wind speed	m/s

ii. NFOUT1

The PDM processor generates a set of diagnostic print files containing various parameters that may be useful for interpretation. Sample results of each diagnostic output file are provided below. The files NFOUT1, NFOUT2, and NFOUT3 in the open statements are linked to OUT1,

OUT2, and OUT3 in the run script as environmental variables, and the print to each file is directed through unit numbers 25, 8, and 11, respectively. The results printed to NFOUT1 are output from subroutine PR to display the fractional plume amounts in each model layer with a sample print shown below.

```
 NOTE: STATS AT END OF PR
IUNSTAB=  1 ISTABLE=  24 IPARTB=   0 IABOVB =   0
LAYER=   1 FREQ=   1 TOTAL=   0.48690
LAYER=   2 FREQ=   6 TOTAL=   3.02227
LAYER=   3 FREQ=  20 TOTAL=  19.49082
LAYER=   4 FREQ=   2 TOTAL=   2.00000
LAYER=   5 FREQ=   0 TOTAL=   0.00000
LAYER=   6 FREQ=   0 TOTAL=   0.00000
```

iii. NFOUT2

The results printed to NFOUT2 are written from subroutine OUTPX and contain selected plume parameters for each MEPSE for each time step during the simulation. A list of the parameters is given in the WRITE statement. A sample of the output in this file is provided below.

```
1995192     0
1  30970100  583762.  329226.  560.8  428.7   59.0  0.00 676.15  0.48  0.07
1  35660200  557596.  238835.  835.4  764.2   57.9  0.00 619.08  0.85  0.06
1  90220300  697848.  446445.   69.5   71.1  315.8  0.00 791.05  0.08  0.35
1 187770400  120092.  237892. 1059.5  927.2   63.4  0.00 446.79  1.03  0.07
1 187800500  121100.  240081. 1426.5 1340.7   38.1  0.00 446.79  1.49  0.04
1 189040600  228246.  394701.  609.5  439.9   64.8  0.00 988.66  0.49  0.07
1 189100700  225238.  425024.  634.1  404.6   64.9  0.001034.18  0.45  0.07
```

iv. NFOUT3

The results printed to NFOUT3 are also written from subroutine OUTPX and contain additional plume parameter data for each MEPSE plume section for each time step during the simulation. The list of parameter names is given in the WRITE statement. A small sample of this output file is provided below.

```
1995192     0
1   0.0  590.3  531.3   0.0   0.0   99.7 17 10 17 10 17 10  2  2  0
1   0.0  864.3  806.4   0.0   0.0  177.7 16  7 16  7 16  7  2  2  0
1   0.0  315.8    0.0   0.0   0.0   16.5 20 13 20 13 20 13  1  1  0
1   0.0 1091.2 1027.8   0.0   0.0  215.6  4  7  4  7  4  7  3  3  0
```

```
1  0.0 1445.6 1407.5   0.0   0.0  311.8 4 7 4 7 4 7 3 3 0
1  0.0  641.9  577.1   0.0   0.0  102.3 7 11 7 11 7 11 2 2 0
1  0.0  666.5  601.7   0.0   0.0   94.1 7 12 7 12 7 12 2 2 0
```

11. Models-3 Emission Projection & Processing System (MEPPS)

a. Input Files

i.	**GRID_DOT_2D**	See 8.b.v	MCIP output file.
ii.	**MET_CRO_2D**	See 8.b.viii	MCIP output file.
iii.	**MET_CRO_3D**	See 8.b.ix	MCIP output file.
iv.	**MET_DOT_3D**	See 8.b.x	MCIP output file.

The MEPPS I/O API files that are made available to ECIP (and CTM) are: 2D Speciated Emission Files, MAJOR Stack Files, and MEPSE Stack Files. These files are described below.

2D Speciated Emission Files

OUTPRO generates the following 2D speciated emission files:

SPEMIS2D Speciated 2D emissions in moles/second
SPEMIS2D_G Speciated 2D emissions in grams/second

Each file contains the ground level emission values for each chemical species generated in the grid by time step. The time step increment is one hour. Each chemical species is contained in a variable whose name is the MEPPS species ID. A species is only written to the files if there is a nonzero value during the case period.

The CB-IV chemical species are shown in the following table:

Variables	Variable Definition	Units
AERO	CB-IV Species ID	Moles or Grams per Second
ALD2	CB-IV Species ID	Moles or Grams per Second
CO	CB-IV Species ID	Moles or Grams per Second
ETH	CB-IV Species ID	Moles or Grams per Second
FORM	CB-IV Species ID	Moles or Grams per Second
ISOP	CB-IV Species ID	Moles or Grams per Second
NH3	CB-IV Species ID	Moles or Grams per Second
NO	CB-IV Species ID	Moles or Grams per Second
NO2	CB-IV Species ID	Moles or Grams per Second
NR	CB-IV Species ID	Moles or Grams per Second
OLE	CB-IV Species ID	Moles or Grams per Second
PAR	CB-IV Species ID	Moles or Grams per Second
PM10	CB-IV Species ID	Moles or Grams per Second
PM2_5	CB-IV Species ID	Moles or Grams per Second
SO2	CB-IV Species ID	Moles or Grams per Second
SULF	CB-IV Species ID	Moles or Grams per Second
TERPB	CB-IV Species ID	Moles or Grams per Second
TOL	CB-IV Species ID	Moles or Grams per Second
XYL	CB-IV Species ID	Moles or Grams per Second

The RADM chemical species are shown in the following table:

Variables	Variable Definition	Units
AERO	RADM Species ID	Moles or Grams per Second
ALD	RADM Species ID	Moles or Grams per Second
CH4	RADM Species ID	Moles or Grams per Second
CO	RADM Species ID	Moles or Grams per Second
CSL	RADM Species ID	Moles or Grams per Second
ETH	RADM Species ID	Moles or Grams per Second
HC3	RADM Species ID	Moles or Grams per Second
HC5	RADM Species ID	Moles or Grams per Second
HC8	RADM Species ID	Moles or Grams per Second
HCHO	RADM Species ID	Moles or Grams per Second
ISO	RADM Species ID	Moles or Grams per Second
KET	RADM Species ID	Moles or Grams per Second
NH3	RADM Species ID	Moles or Grams per Second
NO	RADM Species ID	Moles or Grams per Second
NO2	RADM Species ID	Moles or Grams per Second
OL2	RADM Species ID	Moles or Grams per Second
OLI	RADM Species ID	Moles or Grams per Second
OLT	RADM Species ID	Moles or Grams per Second
ORA2	RADM Species ID	Moles or Grams per Second
PM10	RADM Species ID	Moles or Grams per Second
PM2_5	RADM Species ID	Moles or Grams per Second
SO2	RADM Species ID	Moles or Grams per Second
SULF	RADM Species ID	Moles or Grams per Second
TERPB	RADM Species ID	Moles or Grams per Second
TOL	RADM Species ID	Moles or Grams per Second
XYL	RADM Species ID	Moles or Grams per Second

MAJOR Stack Files

OUTPRO generates the following files for major stacks:

STAK_MAJORMajor stack parameters
CRO_SP_MAJORMajor stack emissions in moles/second
CRO_SP_MAJGMajor stack emissions in grams/second

The variables in these files contain on entry for each major stack. The dimension used for the number of stacks is the row number. The number of columns and layers are set to one. The stack parameter file does not have any time steps. The emission files have a time step increment of one hour.

The variables on the stack parameter file are in the following table:

Variables	Variable Definition	Units
ISTACK	Stack number	-
LATITUDE	Stack latitude	degrees
LONGITUDE	Stack longitude	degrees
STKDM	Inside stack diameter	meters
STKHT	Stack height above ground surface	meters
STKTK	Stack exit temperature	degrees K
STKVE	Stack exit velocity	meters/sec
STKFLW	Stack exit flow rate	meters**3/sec
ROW	Grid row number for stack	-
COL	Grid column number (stack)	-
XLOCA	Projection x coordinate	projection units
YLOCA	Projection y coordinate	projection units
STKCNT	Number of stacks in stack	always one

The emissions files contain the emission values for each stack by time step. The time step increment is one hour. Each chemical species is contained in a variable whose name is the MEPPS species ID. A species is only written to the files if there is a nonzero value during the case period.

A list of the possible species for each mechanism is shown in the description of 2D speciated emission files.

MEPSE Stack Files

OUTPRO generates the following files for MEPSE stacks:

STAK_MEPSEMEPSE stack parameters
CRO_SP_MEPSEMEPSE stack emissions in moles/second
CRO_SP_MEPGMEPSE stack emissions in grams/second

The variables in these files contain on entry for each MEPSE stack group. A stack group is a set of stacks within a plant that have similar stack parameters. The dimension used for the number of groups is the row number. The number of columns and layers are set to one. The stack parameter file does not have any time steps. The emission files have a time step increment of one hour.

The variables on the MEPSE stack parameter file below are similar to those listed for the Major Stacks.

All stack parameters are the averages values for the stacks in the group.

The emissions files contain the total emissions for each stack group by time step. The time step increment is one hour. Each chemical species is contained in a variable whose name is the MEPPS species ID. A species is only written to the files if there is a non-zero value during the case period.

A list of the possible species for each mechanism is shown in the description of 2D speciated emission files, above.

Variables	Variable Definition	Units
ISTACK	Stack Group number	-
LATITUDE	Stack Group latitude	degrees
LONGITUDE	Stack Group longitude	degrees
STKDM	Inside stack diameter	meters
STKHT	Stack height above ground surface	meters
STKTK	Stack exit temperature	degrees K
STKVE	Stack exit velocity	meters/sec
STKFLW	Stack exit flow rate	meters**3/sec
ROW	Grid row number	-
COL	Grid column number	-
XLOCA	Projection x coordinate	projection units
YLOCA	Projection y coordinate	projection units
STKCNT	Number of stacks in Group	always one

APPENDIX K

Emission System Software, Data Files, and Environment Variables

1.0 GENERAL DESCRIPTION

A major part of the Models-3 Emission Processing and Projection System (MEPPS) is the Emission Processor (EMPRO). To maintain its autonomy, system files for EMPRO reside in a separate set of directories. The remainder of the MEPPS system files contain or support the numerous EMPRO pre- and post-processors developed for Models-3 and reside in their own set of directories. The Models-3 Emissions Projection processor (MEPRO) is another emissions application that passes files to EMPRO and whose system files are kept separate in a directory specified in the MEPRO_HOME environment variable.

MEPPS is a combination of FORTRAN, SAS, SAS/AF, and ARC/INFO programs. The following file suffixes have been used throughout the system:

.aml	ARC/INFO program (ARC Macro Language)
.bat	Windows NT execution scripts
.exe	FORTRAN executable
.ext	INCLUDE file used by FORTRAN source program
.f	FORTRAN source program
.hlp	SAS/AF help screen layouts
.icon	ARC/INFO icon
.menu	ARC/INFO menu
.pgm	SCL program for SAS/AF screens
.pmenu	SAS program to define PMENUs (selection buttons on top of screen)
.sas	SAS program
.sasm	File containing macros used by SAS programs
.sasl	INCLUDE file containing link modules for SAS programs
.scr	SAS/AF screen layouts
.sct01	SAS program library for Sun, contains SAS/AF screen programs
.sc2	SAS program library for NT, contains SAS/AF screen programs
.ssd01	SAS data set for Sun
.sd2	SAS data set for NT
.txt	ARC/INFO text file

During MEPPS execution, the environment variable MEPPS_HOME contains the name of the base directory for MEPPS software and system data. EMPRO_HOME contains the name of the base directory for EMPRO software and system data.

The MEPRO application is written in Superbase, a relational database language requiring Microsoft Windows 3.1 or higher (including Windows NT). The environment variable M3MMEPRO contains the name of the base directory for MEPRO system files. The

following file suffixes are used by the Superbase language:

.sbp	Programs written in Superbase
.sbv	Screens developed in Superbase
.sbf	Superbase format database files
.sbd	Superbase format database file descriptions
.dbf	dbase format database files
.[number]	Index files to Superbase-format database file

The following additional file suffixes are required by the MEPRO application for ASCII format data that may need to be imported to the MEPRO database:

.afs	Point-source base-year emission inventory files
.ams	Area-source base-year emission inventory files
.mob	On-road mobile-source base-year emission inventory files
.mef	On-road mobile-source base-year emission factor files
.vmt	Vehicle Miles Traveled files
.scc	Growth factor files for CO, NOx, and VOC
.sgf	Growth factor files for SO_2
.pgf	Growth factor files for PM10
.pcf	Point-source control factor files
.acf	Area-source control factor files
.mcf	Mobile-source control factor files

Sections 2 and 3 discuss the contents of the MEPPS_HOME and the EMPRO_HOME directories. Section 4 discusses the structure of user data directories. Section 5 lists environment variables by program that can be entered when creating studies via Study Planner.

2.0 MEPPS SYSTEM DIRECTORIES

The MEPPS system directories contain all software for the emission system, except the emission processor (EMPRO). The following subdirectories exist under the MEPPS_HOME directory:

bin	Executable directory. Contains all software files needed for system execution. This includes scripts, SAS/AF program libraries, and SAS programs.
bin/sun4	Executable directory for SUN. Contains FORTRAN program executables compiled for Sun.

bin/win	Executable directory for NT. Contains FORTRAN program executables for NT.
lib	System data directory. Contains system lookup tables.
inv	Inventory directory. Contains landuse, emission inventories, etc.
src	Source file directory.

The bin, lib, and inv directories are required for system execution. The files in src are only used to create the files in the bin directory.

The names of these directories are stored in the following environment variables during MEPPS execution:

MEPPS_HOME	MEPPS base directory
GIP_BIN	MEPPS executable directory ($MEPPS_HOME/bin)
GIP_EXE	MEPPS system-specific executable directory ($MEPPS_HOME/bin/sun4 or $MEPPS_HOME/bin/win)
GIP_SYS	MEPPS system tables ($MEPPS_HOME/lib)
GIP_INV	MEPPS emission inventories ($MEPPS_HOME/inv)

2.1 Executable Directory

The executable directory (bin) contains the software files necessary to execute the system. The name of this directory is stored in environment variable GIP_BIN when the MEPPS is executing. The MEPPS executable directory contains the following types of files:

UNIX scripts (no suffix)
SAS program libraries for Sun (.sct01)
SAS program libraries for NT (.sc2)
SAS programs (.sas)
SAS macros (.sasm)

The executable directory also contains two subdirectories:

sun4	Contains the program executables (.exe) and compile scripts (no suffix) for the Sun
win	Contains the program executables (.exe) and compile scripts (.bat) for the NT

The UNIX scripts are:

mepps	MEPPS execution script (stand-alone mode)
mepps_bat	Main script for executing MEPPS batch runs
mepps_models3	MEPPS execution script (executed through Models-3 Tool Manager)
conv_ar	Create emission files from user-specified NET format area source inventory
conv_ar85	Create emission files from 1985 area source inventory
conv_ar88	Create emission files from 1988 area source inventory
conv_ar90	Create emission files from 1990 area source inventory
conv_ar90n	Create emission files from 1990 area source NET inventory
conv_ar90u	Create emission files from updated 1990 area source inventory
conv_ar95n	Create emission files from 1995 area source NET inventory
conv_arm	Script to extract emission data from a user-specified MIDPRO format area source inventory
conv_mv	Create emission files from user-specified NET format mobile source inventory
conv_mv85	Create emission files from 1985 mobile source inventory
conv_mv88	Create emission files from 1988 mobile source inventory
conv_mv90	Create emission files from 1990 mobile source inventory
conv_mv90n	Create emission files from 1990 mobile source NET inventory
conv_mv90u	Create emission files from 1990 mobile source inventory
conv_mv95n	Create emission files from 1995 mobile source NET inventory
conv_mvm	Create emission files from user-specified MIDPRO format mobile source inventory
conv_pt	Create emission files from user-specified NET format point source inventory
conv_pt85	Create emission files from 1985 point source inventory
conv_pt88	Create emission files from 1988 point source inventory
conv_pt90	Create emission files from 1990 point source inventory
conv_pt90n	Create emission files from 1990 point source NET inventory
conv_pt90u	Create emission files from updated 1990 point source inventory
conv_pt95n	Create emission files from 1990 point source NET inventory
conv_ptm	Create emission files from user-specified MIDPRO format point source inventory
gmip	Execute gmip program to extract met data for a case episode
goppt	Execute program to copy ASCII stack files to I/O API files
gopxem	Execute program to copy ASCII 2D temporalized emission files to I/O API files
gopscc	Execute program to copy ASCII SCC level temporalized emission

	files to I/O API files
gopxsp	Execute program to copy ASCII 2D speciated emission files to I/O API files
gridcy	Generate list of counties in grid
makgrid1	Define directories for a new MEPPS grid
makgrid2	Define base directories for a new MEPPS grid, used when copying data from an existing grid.
makproj	Define directories for a new MEPPS project, using standard tables to initialize lookup table directory
makproj1	Define directories for a new MEPPS project, linking to lookup directory from another project
makscen1	Define directories for a new MEPPS case

The NT scripts are:

mepps.bat	MEPPS execution script (stand-alone mode)
mepps_bat.bat	Main script for executing MEPPS batch runs
mepps_batx.bat	Second level main script for executing MEPPS batch runs
mepps_models3.bat	MEPPS execution script (executed through Models-3 Tool Manager)
chgdir.bat	Change working drive and path
conv_ar.bat	Create emission files from user-specified NET format area source inventory
conv_ar85.bat	Create emission files from 1985 area source inventory
conv_ar88.bat	Create emission files from 1988 area source inventory
conv_ar90.bat	Create emission files from 1990 area source inventory
conv_ar90n.bat	Create emission files from 1990 area source NET inventory
conv_ar90u.bat	Create emission files from updated 1990 area source inventory
conv_ar95n.bat	Create emission files from 1995 area source NET inventory
conv_arm.bat	Create emission files from user-specified IDA format area source inventory
conv_mv.bat	Create emission files from user-specified NET format mobile source inventory
conv_mv85.bat	Create emission files from 1985 mobile source inventory
conv_mv88.bat	Create emission files from 1988 mobile source inventory
conv_mv90.bat	Create emission files from 1990 mobile source inventory
conv_mv90n.bat	Create emission files from 1990 mobile source NET inventory
conv_mv90u.bat	Create emission files from 1990 mobile source inventory
conv_mv95n.bat	Create emission files from 1995 mobile source NET inventory
conv_mvm.bat	Create emission files from user-specified IDA format mobile source inventory files

conv_pt.bat	Create emission files from user-specified NET format point source inventory
conv_pt85.bat	Create emission files from 1985 point source inventory
conv_pt88.bat	Create emission files from 1988 point source inventory
conv_pt90.bat	Create emission files from 1990 point source inventory
conv_pt90n.bat	Create emission files from 1990 point source NET inventory
conv_pt90u.bat	Create emission files from updated 1990 point source inventory
conv_pt95n.bat	Create emission files from 1990 point source NET inventory
conv_ptm.bat	Create emission files from user-specified IDA format point source inventory
gmip.bat	Execute gmip program to extract met data for a case episode
goppt.bat	Execute program to copy ASCII stack files to I/O API files
gopxem.bat	Execute program to copy ASCII 2D temporalized emission files to I/O API files
gopscc.bat	Execute program to copy ASCII SCC level temporalized emission files to I/O API files
gopxsp.bat	Execute program to copy ASCII 2D speciated emission files to I/O API files
gridcy.bat	Generate list of counties in grid
makgrid1.bat	Define directories for a new MEPPS grid
makgrid2.bat	Define base directories for a new MEPPS grid, used when copying data from an existing grid.
makproj.bat	Define directories for a new MEPPS project, using standard tables to initialize lookup table directory
makproj1.bat	Define directories for a new MEPPS project, linking to lookup directory from another project
makscen1.bat	Define directories for a new MEPPS case
setapi.bat	Add I/O API library directory to system path
setsas.bat	Set SAS base paths on NT
xsas.bat	Execute a SAS program in batch mode on NT

The SAS Program Libraries are:

gip.sct01	SAS program library for Sun, contains MEPPS screen programs
gip.sc2	SAS program library for NT, contains MEPPS screen programs
gip.trans	Transport version of SAS program library

The SAS Programs are:

autoexec.sas	MEPPS autoexec program for UNIX, starts MEPPS screen interface (stand_alone)

auto_nt.sas	MEPPS autoexec program for NT, starts MEPPS screen interface (stand_alone)
MEPPS_CIMPORT.sas	Import MEPPS program library from transport file (ad hoc program)
MEPPS_CPORT.sas	Port MEPPS program library to a transport file (ad hoc program)
OFFSHORE_FIX.sas	Modify EPA offshore point source files (ad hoc program)
b_arcalc.sas	Batch sub-program - compute area emissions for an episode day
b_arspec.sas	Batch sub-program - speciate area emissions for an episode day
b_bicalc.sas	Batch sub-program - compute biogenic emissions for an episode day
b_bispec.sas	Batch sub-program - speciate biogenic emissions for an episode day
b_mvcnet.sas	Batch sub-program - compute link-specific MV emissions for an episode day
b_mvcoff.sas	Batch sub-program - compute off_net MV emissions for an episode day
b_mvcofp.sas	Batch sub-program - compute land survey MV emissions for an episode day
b_mvspec.sas	Batch sub-program - speciate MV emissions for an episode day
b_ptcalc.sas	Batch sub-program - compute point emissions for an episode day
b_ptspec.sas	Batch sub-program - speciate point emissions for an episode day
bat_arcalc.sas	Batch main program - calculate area emissions
bat_argrid.sas	Batch main program - get area spatial surrogates for gridding
bat_arload.sas	Batch main program - load area source foundation files
bat_arspec.sas	Batch main program - speciate area emissions
bat_bicalc.sas	Batch main program - calculate biogenic emissions
bat_bigrid.sas	Batch main program - get biogenic spatial surrogates for gridding
bat_bispec.sas	Batch main program - speciate biogenic emissions
bat_emis2d.sas	Batch main program - create 2D emission I/O API file
bat_extract.sas	Batch main program - extract emission data from standard U.S. inventory
bat_exuser.sas	Batch main program - extract emission data from user-specified inventory
bat_metload.sas	Batch main program - load meteorological data
bat_mvcalc.sas	Batch main program - calculate motor vehicle emissions
bat_mvgrid.sas	Batch main program - get motor vehicle spatial surrogates for gridding
bat_mvload.sas	Batch main program - load motor vehicle foundation files
bat_mvspec.sas	Batch main program - speciate motor vehicle emissions
bat_ptcalc.sas	Batch main program - calculate point source emissions
bat_ptgrid.sas	Batch main program - assign point sources to grid cells

bat_ptload.sas	Batch main program - load point source foundation files
bat_ptspec.sas	Batch main program - speciate point source emissions
bat_spec2d.sas	Batch main program - create 2D speciated emission I/O API file
bat_split.sas	Batch main program - define stack splits
bat_stack.sas	Batch main program - create I/O API stack files
batset.sas	Set emission system global macro variables from system environmental variables
case_get.sas	Get name of master case for episode
days_dir.sas	Get name of episode directories for current day of case
days_get.sas	Extract base episode data for case
edscenx.sas	Update run descriptions for a multi-day case
environ.sas	Define global macro variables that will contain emission system environmental variables
gethdr.sas	Get header information from an IDA format inventory
gopdays.sas	Define episode data for a case
goppt.sasm	Create ASCII point source emission files for case
gopxem.sasm	Create ASCII 2D summary of temporalized emissions
gopxsp.sasm	Create ASCII 2D summary of speciated emissions from MEPPS models
load_epatier.sas	Load standard EPA tiers (ad hoc program)
load_stcy.sas	Load state and county lookup tables (ad hoc program)
makgridm.sas	Add Models-3 grid information to grid description
makscenm.sas	Add Models-3 case information to case description
makscenx.sas	Initialize episode directories for a multi-day case
ptedgrp.sas	Manually select MEPSE stack groups
ptedstak.sas	Manually select major/minor stacks
ptfreq.sas	Generate point source frequency report for stacks or stack groups
ptgroup.sas	Assign stacks to stack groups for MEPSE selection
ptsplitf.sas	Generate base stack file for use in point source split
ptvars.sas	Generate variable lists of stack parameters and pollutants
sccfilg.sas	Create ASCII SCC level summary of gridded emissions
sccfilt.sas	Create ASCII SCC level summary of temporalized emissions
sccfilu.sas	Create ASCII SCC level summary of ungridded emissions
selpol.sas	Edit pollutant table
set_cnum.sas	Set country number
setcase.sas	Batch sub-program - set case level environment variables + SAS references
setctry.sas	Set country macro variables from country table
setdom.sas	Batch sub-program - set source (raw data) level environment variables + SAS references
setgrid.sas	Batch sub-program - set grid level environment variables + SAS

	references
setproj.sas	Batch sub-program - set study level environment variables + SAS references
settaf.sas	Batch sub-program - set temporal allocation factor file references for emission calculations
start_m3.sas	MEPPS autoexec program, starts MEPPS screen interface (from Models-3 tool manager)
start_nt.sas	MEPPS autoexec program for NT, starts MEPPS screen interface (from Models-3 tool manager)
tierload.sas	Load user defined tiers for SCC reports

The SAS Macro Files are:

batutil.sasm	Batch run utility macros
cgemet.sasm	Macro to generate meteorological data files in the episode directories for a case
lst_cell.sasm	Macro to generate grid cell emission summary report
lst_cy.sasm	Macro to generate county level emission summary report
lst_dom.sasm	Macro to generate domain level emission summary report
lst_st.sasm	Macro to generate state level emission summary report
ptsplit.sasm	Macro to create ASCII version of stack and stack group base files
ptsplitc.sasm	Macro to classify stacks and stack groups by emissions or stack parameters
rnk_cell.sasm	Macro to generate report ranking grid cells by emission values
rnk_cy.sasm	Macro to generate report ranking counties by emission values
rnk_st.sasm	Macro to generate report ranking states by emission values
rnk_stak.sasm	Macro to generate report ranking stacks by emission values or stack parameters
sccextr.sasm	Macros to extract data for SCC reports
sccrpt.sasm	Macros to generate SCC reports
xavemis.sasm	Macros to extract average temporalized emissions for case
xavspee.sasm	Macros to extract average speciated emissions for case
xgremis.sasm	Macros to extract gridded emissions from point + area sources
xhremis.sasm	Macros to extract temporalized emissions for a case hour
xhrspee.sasm	Macros to extract speciated emissions for a case hour

Copies of the FORTRAN program executables exist in both the sun4 and win subdirectories. The executables are:

beis2.exe	Biogenics model - BEIS-2
cvar85.exe	Create area source emission files from 1985 inventory

cvar90.exe	Create area source emission files from interim 1990 format inventory
cvar90u.exe	Create area source emission files from updated interim 1990 format inventory
cvar95.exe	Create area source emission files from NET format inventory
cvmv85.exe	Create mobile source emission files from 1985 inventory
cvmv90.exe	Create mobile source emission files from interim 1990 format inventory
cvpt85.exe	Create point source emission files from 1985 inventory
cvpt90.exe	Create point source emission files from interim 1990 format inventory
cvpt90u.exe	Create point source emission files from updated interim 1990 format inventory
cvpt95.exe	Create point source emission files from NET format inventory
gridcy.exe	Generate list of counties in grid
gmip.exe	Create emission system meteorology files
goppt.exe	Create I/O API stack files
gopxem.exe	Create 2D I/O API file from temporalized emissions
gopxsp.exe	Create 2D I/O API file from speciated emissions
landmg.exe	Merge BEIS-2 state landuse files into a national landuse file
soar85.exe	Create sulfate tables from 1985 area source inventory
sopt85.exe	Create sulfate tables from 1985 point source inventory

The UNIX scripts that compile FORTRAN programs in subdirectory sun4 are:

compile_all	Script to compile entire system
compile	INPRO program compile script
compile_gmip	Script to compile gmip
compile_gop	OUTPRO program compile script
compile_mv5a	MOBILE5A compile script
complib	INPRO utility routine compile script
complib_gop	OUTPRO utility routine compile script

The NT scripts that compile FORTRAN programs in subdirectory win are:

compile.bat	INPRO program compile script
compile_gmip.bat	Script to compile gmip
compile_gop.bat	OUTPRO program compile script
compile_mv5a.bat	MOBILE5A compile script
complib.bat	INPRO utility routine compile script
complib_gop.bat	OUTPRO utility routine compile script

The utility routine object files for the FORTRAN programs in subdirectory sun4 are:

goplib.o	OUTPRO utility routines
lib1.o	INPRO utility routines - MEPPS specific
lib2.o	INPRO utility routines - generic
libsun4.o	MEPPS utility routines - Sun specific

The utility routine object files for the FORTRAN programs in subdirectory win are:

goplib.obj	OUTPRO utility routines
lib1.obj	INPRO utility routines - MEPPS specific
lib2.obj	INPRO utility routines - generic
libeom.obj	MEPPS utility routines - NT specific

2.2 System Data Directory

The system data directory (lib) contains the lookup tables needed for MEPPS. The name of this directory is stored in environment variable GIP_SYS when the MEPPS is executing. The tables in the directory are shown below. Unless otherwise noted the files are ASCII data sets.

country_table	Country lookup table
state_table	State lookup table
county_table	County lookup table
state_plane_table	State plane table
epatier1.ssd01	EPA tiers - set 1 (SAS data set for Sun)
epatier2.ssd01	EPA tiers - set 2 (SAS data set for Sun)
epatier3.ssd01	EPA tiers - set 3 (SAS data set for Sun)
epatier1.sd2	EPA tiers - set 1 (SAS data set for NT)
epatier2.sd2	EPA tiers - set 2 (SAS data set for NT)
epatier3.s2	EPA tiers - set 3 (SAS data set for NT)
mvscc.eps	MovEM SCC to areatype/factype/vehicle type lookup table
oic_asct	Area Source ASCT to OIC lookup table
oic_desc	OIC descriptions
polutant_table	Pollutant selection table
scc10ams.data	10-digit SCCs
sicscc_oic_table	Point Source SIC-SCC to OIC lookup table
stkdflt.txt	Stack parameter defaults (by SCC)
TIER.ar	Standard EPA tiers for area sources
TIER.pt	Standard EPA tiers for point sources

2.3 Inventory Directory

The inventory directory (inv) contains the landuse files and emission inventories. The name of this directory is stored in environment variable GIP_INV when the MEPPS is executing. The directory has the following subdirectories:

y1985	1985 emission inventories
y1988	1988 emission inventories
y1990	1990 emission inventories
y1995	1995 emission inventories

The files in the landuse directory are:

landuse.dat	BEIS-2 county level landuse for U.S., Canada, and Mexico
us_landuse.dat	BEIS-2 county level landuse for U.S.
us_landuse.summary	Report of BEIS-2 county level landuse for U.S.

The 1985 emission inventory files are:

area.dat	U.S. area source emission inventory
point.dat	U.S. point source emission inventory
taff.ar	Area source temporal allocation factors
taff.pt	Point source temporal allocation factors
vmt.data	U.S. vehicle miles travelled data

The 1988 emission inventory files are:

us88ar.v1.data.d110893	U.S. area source emission inventory
us88pt.v1.data.d110893	U.S. point source emission inventory
us88vmt.v1.data.d060993	U.S. vehicle miles travelled data

The 1990 emission inventory files are:

aaaread.doc	General description of interim regional emission inventories
artaff90.ssd01	Area source temporal allocation factor file (Sun version)
artaff90.sd2	Area source temporal allocation factor file (NT version)
inter.ar90.data	U.S. interim area source emission inventory
inter.ar90.data.d060895	U.S. updated interim area source inventory
inter.pt90.data	U.S. interim point source emission inventory
inter.pt90.data.d060895	U.S. updated interim point source emission inventory

inter.vmt90.data	U.S. annual vehicles miles travelled
jeiog.readme	Temporal allocation file field descriptions
net90a02.txt	Net Emissions Trend (NET) area source emissions inventory
net90p02_srtd.txt	NET point source emissions inventory
partic.ar	Area source particulate inventory
partic.pt	Point source particulate inventory
pstaff90.ssd01	Older version of 1990 point source temporal allocation factor file
pstaffxx.ssd01	Point source temporal allocation factor file (Sun version)
pstaffxx.sd2	Point source temporal allocation factor file (NT version)
so4_table.ar	U.S. area source SO4 table
so4_table.pt	U.S. point source SO4 table
sulfate.ar	U.S. area source sulfate inventory
sulfate.pt	U.S. point source sulfate inventory

The 1995 emission inventory files are:

net90a05.txt	NET area source emissions inventory
net90p05.txt	NET point source emissions inventory
vmt95.txt	U.S. annual vehicles miles travelled

2.4 Source File Directory

The source directory (src) basically contains files used to create the system executables. This directory is not needed to execute the software. The MEPPS source directory contains the following types of files:

UNIX scripts to compile the FORTRAN programs (no suffix)
Source code for FORTRAN programs (.f)
INCLUDE files for FORTRAN programs (.exe)
Object modules created from utility routine compiles (.o)
PMENU programs for SAS interface screens (.pmenu)
SCL programs for SAS interface screens (.pgm)
Layouts for SAS interface screens (.scr)

The SAS interface screens are completely defined in the SAS program library gip.sct01 in the executable directory. The SCL program files are included in the program library. The screen layout files are here for informational reasons only, the full screen definition is contained in the SAS program library.

The FORTRAN program source files are:

beis2.f	Biogenics model - BEIS-2
cvar85.f	Create area source emission files from 1985 inventory
cvar90.f	Create area source emission files from interim 1990 format inventory
cvar90u.f	Create area source emission files from updated interim 1990 format inventory
cvar95.f	Create area source emission files from NET format inventory
cvarmid.f	Create area source emission files from IDA format inventory
cvmv85.f	Create mobile source emission files from 1985 inventory
cvmv90.f	Create mobile source emission files from interim 1990 format inventory
cvmv95.f	Create mobile source emission files from 1995 format inventory
cvpt85.f	Create point source emission files from 1985 inventory (NET or MIPRO format on screen)
cvpt90.f	Create point source emission files from interim 1990 format inventory
cvpt90u.f	Create point source emission files from updated interim 1990 format inventory
cvpt95.f	Create point source emission files from NET format inventory
cvptmid.f	Create point source emission files from IDA format inventory
gmip.f	Create emission system meteorology files
goplib.f	OUTPRO utility routines
goppt.f	Create I/O API stack files
gopscc.f	Create scc-based I/O API file from temporalized emissions
gopxem.f	Create 2D I/O API file from temporalized emissions
gopxsp.f	Create 2D I/O API file from speciated emissions
gridcy.f	Generate county list from grid description
landmg.f	Merge BEIS-2 state landuse files into a national landuse file
lib1.f	INPRO utility routines - MEPPS specific
lib2.f	INPRO utility routines - generic
libsun4.f	INPRO utility routines - Sun specific
libwin.f	INPRO utility routines - NT specific
mobile5a.f	MOBILE-5A
soar85.f	Create sulfate tables from 1985 area source inventory
sopt85.f	Create sulfate tables from 1985 point source inventory
utmcon.f	Original Field Survey Program for UTM to Geographic Coordinate Conversions

The INCLUDE files used by the FORTRAN programs are:

area.ext	Data from area source emission inventory record
arso4.ext	Data from area source SO4 inventory record
biemf.ext	BEIS-2 Emission factor tables
biparam.ext	BEIS-2 General parameters
bland.ext	BEIS-2 Landuse data
bmetw.ext	BEIS-2 Meteorological Data Work File Variables
crparam.ext	BEIS-2 General parameters
factor.ext	Conversion factors
fipsel.ext	County selection table
flderr.ext	Data field error count block
gemap.ext	MEPPS grid and case information
gmip.ext	GMIP File information
goppt.ext	Point source work file common block
gopscc.ext	Lsst record read common block for program gopscc.f
headg.ext	Page heading information for reports
lumap.ext	BEIS-2Landuse mapping table
oicds.ext	OIC table
padj.ext	Pollutant adjustment table
param.ext	General system parameters
point.ext	Emission inventory data for a point source
pointu.ext	1990 Particulate inventory data for a point source
polut.ext	Pollutant table
polutd.ext	SAROADs and Pollutant IDs for pollutant table
ptso4.ext	SO4 data for a point source
stcy.ext	State-county tables
stkdf.ext	Stack default table (by SCC)

The PMENU programs used by the SAS/AF Screens are:

main.pmenu	PMENU definitions for main screen for UNIX
main_nt.pmenu	PMENU definitions for main screen for NT

The SCL programs for SAS/AF screens are:

batset.pgm	Set MEPPS environmental variables for batch processing
bldname.pgm	Build full file name for a file from name levels
bldprog.pgm	Build full file name for an executable from name levels
cgemet.pgm	Load meteoroligical data into emission system
cgemis.pgm	Create emissions files from standard U.S. inventories

cgemisx.pgm	Create emission files from a user-specified inventory
checkdir.pgm	Check if directory exists
checkdom.pgm	Check if domain directory exists
checkenv.pgm	Checks if base INPRO environment variables set
checklow.pgm	Checks if subdirectory exists under a directory
chekdays.pgm	Checks if two dates are same day of week
currdir.pgm	Get name of current directory
deldirek.pgm	Delete directory (all files and subdirectories)
deldom.pgm	Delete a domain
delfile.pgm	Delete a file
delgrid.pgm	Delete a grid
delproj.pgm	Delete a project
delscen.pgm	Delete a case
dummy.pgm	Dummy program
edscenx.pgm	Update case descriptions for a multi_day case
envclear.pgm	Clears levels (project,domain,grid,case) of emission system environment variables
gemapenv.pgm	Set all emission system environment variables
gemapset.pgm	Setup emission system environment
getdir.pgm	Get directory name
gethdr.pgm	Get header information from a IDA format emission inventory
gethdrx.pgm	Get IDA format file header information from user
getscen.pgm	Build the full path name for a MEPPS CASE
gopadj.pgm	Specify pollutant adjustment file for input processor
gopchkpd.pgm	Check if start + end times need data outside case episodes
gopdays.pgm	Edit case episode file
goppd.pgm	Get case episode information
goppt.pgm	Create I/O API stack files
gopxem.pgm	Create 2D I/O API file from temporalized emissions
gopxsp.pgm	Create 2D I/O API file from speciated emissions
m3getvar.pgm	Get value of environment variable sent from Models-3
main.pgm	Main (controlling) screen program for UNIX
main_nt.pgm	Main (controlling) screen program for NT
makdirek.pgm	Create a directory
makdom.pgm	Create an emission system domain
makgr_m3.pgm	Create an emission system grid for a Models-3 grid
makgr_nt.pgm	Create a grid, copying data from an existing grid (NT version)
makgrid.pgm	Create an emission system grid
makgridc.pgm	Create a grid, copying data from an existing grid (UNIX version)
makpr_m3.pgm	Create an emission system study for a Models-3 study
makproj.pgm	Create an emission system project

maksc_m3.pgm	Create an emission system case for a Models-3 case
makscen.pgm	Create an emission system case
makscenx.pgm	Create an emission system multi-day case
mechget.pgm	Select a chemical mechanism
notavail.pgm	Not available screen
ptgroup.pgm	Point source split - Assign stack groups
ptsplit.pgm	Point source split - Main screen
ptsplitc.pgm	Point source split - Classify by stack/group properties
ptsplitr.pgm	Point source split - Ranking report
report.pgm	Reports - Main screen
reportr.pgm	Reports - Rank by emissions
reports.pgm	Reports - Summary report
repsel.pgm	Reports - Enter selection criteria
repselh.pgm	Reports - Sample logical expressions (Help Screen)
runbatch.pgm	Submit batch run
sasfile.pgm	Build a SAS file name(libref.name) given a library reference and the full file name
sccfilg.pgm	Generate SCC level gridded emission files
sccfilt.pgm	Generate SCC level temporalized emission files
sccfilu.pgm	Generate SCC level ungridded emission files
sccrpt.pgm	SCC Reports - Main Screen
sccrptr.pgm	SCC Reports - Rank by Emissions
sccrpts.pgm	SCC Reports - Summary Report
sccsel.pgm	SCC Reports - Enter Selection Criteria
sccselh.pgm	SCC Reports - Sample Logical Expressions
scctier.pgm	SCC Reports - Assign User Tier File
scctierc.pgm	SCC Reports - Create User Tier File
selpol.pgm	Edit table input processor uses to select pollutants for emission files
sepchar.pgm	Get directory name level separation character for operating system
setbin.pgm	Initialize MEPPS execution environment
setenv.pgm	Set a MEPPS environment variable (global SAS macro variable)
setenvx.pgm	Set a system environment variable
sethome.pgm	Set emission system home directory
setref.pgm	Set emission system file or library reference (SAS+environment)
start.pgm	MEPPS start program (stand-alone mode)
start_m3.pgm	MEPPS start program (when executed from Models-3)
uiedgrid.pgm	Edit Grid and Projection Information
uigdig.pgm	Grid Definition Input Generator
uipopask.pgm	Pop-up prompt screen

uipopmsg.pgm	Pop-up message screen
getcase.pgml	SCL commands included in programs to get project, grid, and case information from screen fields

SAS/AF Screen Layouts are shown in the following files:

cgemet.scr	Load meteoroligical data into emission system
cgemis.scr	Create emissions files from standard U.S. inventories
cgemisx.scr	Create emission files from a user-specified inventory
deldirek.scr	Delete directory (all files and subdirectories)
deldom.scr	Delete a domain
delgrid.scr	Delete a grid
delproj.scr	Delete a project
delscen.scr	Delete a case
edscenx.scr	Delete a case
gemapset.scr	Setup emission system environment
gopadj.scr	Specify pollutant adjustment file for input processor
gopdays.scr	Edit case episode file
goppt.scr	Create I/O API stack files
gopxem.scr	Create 2D I/O API file from temporalized emissions
gopxsp.scr	Create 2D I/O API file from speciated emissions
main.scr	Main (controlling) screen for UNIX
main_nt.scr	Main (controlling) screen for NT
makdom.scr	Create an emission system domain
makgr_m3.scr	Create an emission system grid for a Models-3 grid
makgrid.scr	Create an emission system grid
makgridc.scr	Create a grid, copying data from an existing grid (UNIX version)
makpr_m3.scr	Create an emission system study for a Models-3 study
makproj.scr	Create an emission system project
maksc_m3.scr	Create an emission system case for a Models-3 case
makscen.scr	Create an emission system case
makscenx.scr	Create an emission system multi-day case
notavail.scr	Not available screen
ptsplit.scr	Point source split - Main screen
ptsplitc.scr	Point source split - Classify by stack/group properties
ptsplitr.scr	Point source split - Ranking report
report.scr	Reports - Main screen
reportr.scr	Reports - Rank by emissions
repsel.scr	Reports - Enter selection criteria
repselh.scr	Reports - Sample logical expressions (Help Screen)
sccfilg.scr	Generate SCC level gridded emission files

sccfilt.scr	Generate SCC level temporalized emission files
sccfilu.scr	Generate SCC level ungridded emission files
sccrpt.scr	SCC Reports - Main Screen
sccrptr.scr	SCC Reports - Rank by Emissions
sccrpts.scr	SCC Reports - Summary Report
sccsel.scr	SCC Reports - Enter Selection Criteria
sccselh.scr	SCC Reports - Sample Logical Expressions
scctier.scr	SCC Reports - Assign User Tier File
scctierc.scr	SCC Reports - Create User Tier File
selpol.scr	Edit table input processor uses to select pollutants for emission files
sethome.scr	Set emission system home directory
uiedgrid.scr	Edit Grid and Projection Information
uigdig.pgm	Grid Definition Input Generator
uipopask.scr	Pop-up prompt screen
uipopmsg.scr	Pop-up message screen

SAS/AF Help Screen Layouts are shown in the following files:

begdate.hlp	Case Start
begday.hlp	Output File Case Start
direk.hlp	Directory Field
domain.hlp	Source Name
filename.hlp	Full Filename
grid.hlp	Grid Name
gridin.hlp	Grid Name on Create Emission File Screen
mech.hlp	Split Factor ID
mechtype.hlp	Chemical Mechanism
metdirek.hlp	Met Data Directory
metload.hlp	Met Data Load Fields
ogtype.hlp	Organic Gas Type
outdirek.hlp	Output Directory
outperd.hlp	Output File Period
project.hlp	Study ID
repsel.hlp	Report Select
scchel.hlp	SCC Select
scen.hlp	Case Name
scenmast.hlp	Master Case
scenpref.hlp	Case Prefix
sgrid1.hlp	Source grid for ungridded emission data
sgrid2.hlp	Source grid for ungridded GIS data

sgrid3.hlp	Source grid for gridded emission data
sgrid4.hlp	Source grid for ungridded GIS data
srcgrid.hlp	Home directory and study ID of source grids
srclook.hlp	Home directory and study Id of source study
tierfile.hlp	Tier File Description
tierin.hlp	ASCII Tier Input File
timezone.hlp	Timezone

3.0 EMPRO SYSTEM DIRECTORIES

The EMPRO system directories contain all software for the emission processor (EMPRO). The following subdirectories exist under the EMPRO_HOME directory:

bin	Executable directory. Contains all software files needed for system execution.
lib	System data directory.
src	Source file directory.

The bin and lib directories are required for system execution. The files in src are only used to create the files in the bin directory. The lib directory has two subdirectories, lookup and sysdata. The lookup subdirectory contains the standard lookup tables supplied when a user creates a study using standard tables. Subdirectory sysdata contains system lookup tables.

The names of these directories are stored in the following environmental variables during EMPRO execution:

EMPRO_HOME	EMPRO base directory
EMS_BIN	EMPRO executable directory ($EMPRO_HOME/bin)
EMS_SYS	EMPRO system tables ($EMPRO_HOME/lib/sysdata)

3.1 Executable Directory

The executable directory (bin) contains the software files necessary to execute the system. The name of this directory is stored in environment variable EMS_BIN when the EMPRO is executing. The EMPRO executable directory contains the following types of files:

UNIX scripts (no suffix)
NT scripts (.bat)
SAS program libraries for Sun (.sct01)
SAS program libraries for NT (.sc2)
SAS programs (.sas)

SAS macros (.sasm)
ARC/INFO programs (.aml)
ARC/INFO menu files (.menu)
ARC/INFO icon files (.icon)
ARC/INFO text files (.txt)

The UNIX scripts are:

beis2	Execute BEIS-2 from within EMPRO
esdls.csh	Define directories containing TIGER/Line road coverages for states
mobile	Execute MOBILE5
xarcinfo	Execute ARC aml from within EMPRO

The NT scripts are:

beis2.bat	Execute BEIS-2 from within EMPRO
esdls.bat	Define directories containing TIGER/Line road coverages for states
mobile.bat	Execute MOBILE5
xarcinfo.bat	Execute ARC aml from within EMPRO
xrunbatch.bat	Execute batch run from within EMPRO
xrunbld.bat	Execute run build batch program from within EMPRO

The SAS Program Libraries are:

gemap.sct01	SAS program library for Sun, contains EMPRO screen programs
gemap.sc2	SAS program library for NT, contains EMPRO screen programs
gemap.trans	Transport version of SAS program library

The SAS Programs are:

EMPRO_CIMPORT.sas	Import EMPRO program library from transport file (ad hoc program)
EMPRO_CPORT.sas	Port EMPRO program library to a transport file(ad hoc program)
arctled.sas	Area Edit control factors
arctlld.sas	Area Load control factors
ardayld.sas	Area Day specific file load
ardayqa.sas	Area Day specific file QC reports - Main program
areasrc.sas	Area Source Processor (grids area source emissions)

areassap.sas	Area Spatial surrogates - Check assignments file
areassdp.sas	Area Spatial surrogates - Check lookup tables
areassr.sas	Area Spatial surrogate processor (US)
areatap.sas	Area Temporal allocation processor (EMPRO temporal profiles)
areatapf.sas	Area Temporal allocation processor (FREDS temporal profiles)
areatpg.sas	Area Temporal profile generator
argroed.sas	Area Edit growth factors
argrold.sas	Area Load growth factors
arload.sas	Area Load foundation files
arqa.sas	Area Foundation file QC reports - Main program
arqaday.sas	Area Day-specific file QC - emissions file
arqaemis.sas	Area Foundation file QC reports - emissions file
arqaprof.sas	Area Day-specific file QC - temporal profiles
arqatprl.sas	Area Foundation file QC reports - temporal profiles
arssrcan.sas	Area Compute Canadian spatial surrogates
ascd.sas	Load area source category lookup table (ad hoc)
b2ecp.sas	BEIS-2 Compute Emissions
b2edit_emf.sas	BEIS-2 Edit Emission Factors Table
b2edit_frost.sas	BEIS-2 Edit Frost Data
b2edit_lumap.sas	BEIS-2 Edit table mapping vegetation IDs to landuse codes
b2ssc.sas	BEIS-2 Generate spatial surrogate files from national coverages (also used by Area Source)
b2ssr.sas	BEIS-2 Spatial Surrogates Processor
caseday.sas	Get episode day information from case master directory
casedir.sas	Assign master episode for a case
censum.sas	Load census summary (censum) table (ad hoc)
create_lookup_dat.sas	Create SAS transport file containing SAS files in lookup table directory (ad hoc)
cy_ncell.sas	Find the maximum number of grid cells used by a county
facclas.sas	Load motor vehicle factype (roadtype) table (ad hoc)
getday.sas	Get episode information from run description file
gisview.sas	Start ARC/INFO GIS Viewer
gridview.sas	Start ARC/INFO Grid Viewer
initops.sas	Load hourly and weekly profile tables (ad hoc)
mv5a_dig.sas	MovEM MOBILE 5A data input generator
mvascii.sas	MovEM Process On-network ASCII files for GIS
mvdir.sas	MovEM Foundation file load
mvecedaf.sas	MovEM Edit tab;e that assigns emission factors by area and road type
mvecedca.sas	MovEM Assign county aliases for emission factors
mvecedhr.sas	MovEM Edit table that assigns emission factors by time of day

mvecemfx.sas	MovEM extract emission factors for ecas
mvecnet.sas	MovEM Network link-specific emission calculation
mvecoff.sas	MovEM Off-network county level emission calculation
mvecofp.sas	MovEM Land Survey 1/4 section emission calculation
mvecrpt.sas	MovEM Emission calculation error and summary reports
mvexnatl.sas	MovEM Extract data from national coverages
mvextrct.sas	MovEM Define extract control file for ARC/INFO
mvgrnet.sas	MovEM Grid Network link-specific vehicle miles travelled
mvgroff.sas	MovEM Grid off-network county level vehicle miles travelled
mvgrofp.sas	MovEM Grid Land Survey 1/4 section vehicle miles travelled
mvnet.sas	MovEM Process network definition file
mvparam.sas	MovEM Setup general motor vehicle parameters
mvqaadj.sas	MovEM QC reports - seasonal/daily adjustment factors
mvqafmix.sas	MovEM QC reports - off-network county vehicle mix data
mvqafspd.sas	MovEM QC reports - off-network county transportation speed data
mvqafvmt.sas	MovEM QC reports - off-network county vmt data
mvqaonet.sas	MovEM QC reports - link definition file
mvqaonim.sas	MovEM QC reports - network inspection/maintenance program factors
mvqaospd.sas	MovEM QC reports - link-specific speed profiles
mvqaovmt.sas	MovEM QC reports - link-specific vmt data
mvqaspdu.sas	MovEM QC reports - user-specified speed profiles
mvqavmix.sas	MovEM QC reports - link-specific vehicle mix data
mvqavmtp.sas	MovEM QC reports - land survey motor vehicle data
mvsapld.sas	MovEM Load transportation speed data
mvsplit.sas	MovEM Split motor vehicle emissions for density QC plots ?
mvsscy.sas	MovEM Compute spatial surrogate rations from county areas within grid cells
mvssfha.sas	MovEM Compute gridding ratios from FHA-based spatial surrogate data
mvsslink.sas	MovEM Compute gridding ratios from link lengths within grid cells
mvsstig.sas	MovEM Compute gridding ratios from TIGER/Line spatial surrogate data
mvtapld.sas	MovEM Load temporal adjustment data
oic_asct.sas	Load ASCT to OIC lookup table (oic_asct) (ad hoc)
oic_ces.sas	Load ASCT to OIC lookup table (oic_asct) from ARB data (ad hoc)
oic_desc.sas	Load OIC description table (oic_desc) (ad hoc)
oic_scc.sas	Load SCC to OIC lookup table (oic_scc) (ad hoc)

polutant.sas	Load pollutant table (ad hoc)
ptcemld.sas	Point Load continuous emission monitoring (CEM) data
ptctled.sas	Point Edit control factors
ptctlld.sas	Point Load control factors
ptdayld.sas	Point Load day-specific data
ptdefstk.sas	Point Create default stack parameter file
ptgridld.sas	Point Load point source grid definition data
ptgroed.sas	Point Edit growth factors
ptgrold.sas	Point Load growth factors
ptload.sas	Point Foundation file load
ptqa.sas	Point Foundation file QC report - Main program
ptqadev.sas	Point Foundation file QC report - Device file
ptqaemis.sas	Point Foundation file QC report - Emission file
ptqafac.sas	Point Foundation file QC report - Facility file
ptqaproc.sas	Point Foundation file QC report - Process file
ptqastk.sas	Point Foundation file QC report - Stack file
ptsrc.sas	Point Source Processor (grids emissions)
pttapf.sas	Point Temporal allocation processor (FREDS temporal profiles)
pttoarc.sas	Point Grid processor (assigns points to grid cells using ARC)
pttpg.sas	Point Temporal profile generator
qarept.sas	QC report program used for point and areas source foundation files
scc_tabl.sas	Load source classification code (SCC) table (ad hoc)
sic_tabl.sas	Load standard industrial classification code (SIC) table (ad hoc)
spcaprof.sas	Speciation RADM/Other - Compute split factors
spcargtg.sas	Speciation RADM/Other - Compute ROG to TOG factors
spcbdp.sas	Speciation CB-IV - Load lumping data
spcbsfp.sas	Speciation CB-IV - Compute split factors
spchadj.sas	Speciation Edit adjustment factors
spchdp.sas	Speciation CB-IV - Load VOC chemical data
spchmpro.sas	Speciation Processor (Speciates emissions)
spchplot.sas	Speciation QC plots
spcspp.sas	Speciation CB-IV - Compute ROG to TOG factors
spdadjd.sas	Table Load Load default transportation speed values (MV)
specfl.sas	Speciation Initialize data sets
spldmap.sas	Speciation RADM/Other - Load profile mapping table
spldprc.sas	Speciation CB-IV - Load profile mapping table
spldprof.sas	Speciation RADM/Other - Load split factors (non-HC, biogenic, ...)
spldrgtg.sas	Speciation RADM/Other - Load ROG to TOG factors
sprd_arb.sas	Convert ARB supplied split factor profiles by SCC to split factor

	profiles by hydrocaraabon profile number (ad hoc)
spunprof.sas	Speciation Unload speciation profiles data set (ad hoc)
srg_cens.sas	Load table mapping census codes to spatial surrogate codes
srg_lu.sas	Load table mapping landuse codes to spatial surrogate codes
srg_tigr.sas	Load table mapping TIGER/LINE codes to spatial surrogate codes
ssad.sas	Load area source spatial surrogate code description table
ssd.sas	Load area source spatial surrogate assignment table
tigerfac.sas	Load table that maps TIGER/Line census feature codes to area and road types
uigdig.sas	Grid Definition Input Generator
uigrdasc.sas	Write ASCII grid description file
uigrdini.sas	Initialize grid description file
uirunasc.sas	Write ASCII run description file
uirunini.sas	Initialize run description file
unld_ar.sas	GIS-VIEW Create ASCII emissions file from area source foundation file
unld_arcell.sas	GIS-VIEW Create ASCII grid cell summary file from gridded area source emissions
unld_argrid.sas	GIS-VIEW Create ASCII county-cell summary file from gridded area source emissions
unld_pl.sas	GIS-VIEW Create ASCII plant summary from point source foundation files
unld_plcem.sas	GIS-VIEW Create ASCII plant summary from CEM emissions
unld_plgrid.sas	GIS-VIEW Create ASCII plant summary from gridded point source emissions
unld_pt.sas	GIS-VIEW Create ASCII stack summary from point source foundation files
unld_ptcem.sas	GIS-VIEW Create ASCII STACK summary from CEM emissions
unld_ptgrid.sas	GIS-VIEW Create ASCII stack summary from gridded point source emissions
unld_pttemp.sas	GIS-VIEW Create ASCII stack summary from temporalized point source emissions
unld_temp.sas	GIS-VIEW Create ASCII file containing temporalized emission data from an emission model. Summarized by hour, hour, stid, cyid, icell, and.jcell
utmvdefv.sas	Load default vehicle mix table (DEFVMIX) (ad hoc)

The SAS program include files are:

arcntl.sasl	Link modules included to fetch control factors for area source

	projections
argrow.sasl	Link modules included to fetch growth factors for area source projections
chek_af.sasl	Link modules included to check area and road types
chek_sc.sasl	Link modules included to check states and counties
mvecadj.sasl	Link modules included to fetch daily adjustment factors in the motor vehicle emission calculation
mvecdif.sasl	Link modules included to fetch diurnal emission factors in the motor vehicle emission calculation
mvecemf.sasl	Link modules included to fetch emission factors in the motor vehicle emission calculation
mvecim.sasl	Link modules included to fetch inspection/maintenance program factors in the motor vehicle emission calculation
mvecspd.sasl	Link modules included to fetch transportation speed profiles in the motor vehicle emission calculation
mvecspdl.sasl	Link modules included to fetch link-specific transportation speed profiles in the motor vehicle emission calculation
mvectmp.sasl	Link modules included to fetch temporal adjustment factors in the motor vehicle emission calculation
mvecvmx.sasl	Link modules included to fetch vehicle mix profiles in the motor vehicle emission calculation
mvecvmxl.sasl	Link modules included to fetch link-specific vehicle mix profiles in the motor vehicle emission calculation
ptcntl.sasl	Link modules included to fetch control factors for point source projections
ptgrow.sasl	Link modules included to to fetch growth factors for point source projections
pttapf.sasl	Link modules included in the point source temporal allocation programs to summarize emissions for stacks to the scc polid level

The SAS macro Files are:

mv5a_dig.sasm	Macros used to compute motor vehicle emission factors
mvecutil.sasm	Macros used by motor vehicle emission calculation processors
mvqarpt.sasm	Macros used for motor vehicle input file QC checking and file updating
qaplotb.sasm	Foundation file map plot macros
spinit.sasm	Speciation table initialization macros
utility.sasm	Utility macro library
utilplot.sasm	Utility graphics macro library

The ARC/INFO programs are:

classedit.aml	GIS-VIEW - Modify emission display class ranges
classenv.aml	GIS-VIEW - Control emission display class environment
cmproj.aml	Coverage projection - main program
cmproj2.aml	Coverage projection - project a coverage
cmproj2a.aml	Coverage projection - verify a coverage's projection parameters
cmproj3.aml	Coverage projection - write a projection parameter file for a coverage
cmprojnr.aml	Coverage projection - initialize projection parameters
cmprojo.aml	Coverage projection - open projection parameter menus
cmprojor.aml	Coverage projection - initialize projection parameters
cmprojq.aml	Coverage projection - initialize projection menus
cntyrd_grid.aml	MOBILE - create output file based on county grid coverage for county based mobile source estimation
create_centracts.aml	Create a census tracts coverage for the grid area in the appropriate projection
dict.aml	Utility program for storing and searching global "dictionaries"
emis_load.aml	GIS-VIEW - user defined request for ASCII emissions files from SAS and conversion to INFO database files for ARC/INFO (drives gemap_sasread.aml)
emprolib.aml	Program run during installation to create connections to TIGER and hydrography data
err_report.aml	Generate a description of unpredicted errors (write to display and error log file)
fhard_grid.aml	Create a Federal Highway Administration (FHA) roads coverage for the grid area in the appropriate projection
fileman.aml	Utility for checking permissions and deleting coverages, files, and INFO files
gdig.aml	Grid Definition Input Generator - main program
gdig2.aml	Grid Definition Input Generator - generate and project grid coverage
gdig_create.aml	Grid Definition Input Generator - identify required coverages
gemap_draw.aml	GIS-VIEW - generate map display
gemap_emmis.aml	GIS-VIEW - display emission values (gridded and raw) for any pollutant
gemap_env.aml	Set up global variables
gemap_exit.aml	GIS-VIEW - delete temporary files and exit GIS-VIEW
gemap_fmenu.aml	GIS-VIEW - utility program to create a variety of menus
gemap_query.aml	GIS-VIEW - logical or graphical selection of coverage features - main program

gemap_sasread.aml	GIS-VIEW - request ASCII emissions files from SAS and create INFO database files for ARC/INFO
gemap_setup.aml	GIS-VIEW - verify required files, coverages, and environment variables before entering GIS-VIEW
gemap_substudy.aml	GIS-VIEW - create workspaces and coverages for an area that covers a portion of the current grid
gemap_symsets.aml	GIS-VIEW - select from families of symbols for map display
gemap_tools.aml	GIS-VIEW - a collection of map query and display tools
gemap_view.aml	GIS-VIEW - start ArcPlot and the GIS-VIEW main menu
grd_desc_in_read.aml	Read grid description information
grid_view.aml	Display grid resulting from grid definition without generating coverages
list_items.aml	GIS-VIEW - List selected coverage attributes
lulcdraw.aml	Land Use / Land Cover Processor (not implemented) - display (not implemented)
lulcfull.aml	Land Use / Land Cover Processor (not implemented) - display full view
lulcmany.aml	Land Use / Land Cover Processor (not implemented) - select many features
lulcmcov.aml	Land Use / Land Cover Processor (not implemented) -
lulcp.aml	Land Use / Land Cover Processor (not implemented) - set up edit environment
lulcpan.aml	Land Use / Land Cover Processor (not implemented) -
lulcsel.aml	Land Use / Land Cover Processor (not implemented) - select an edit coverage
majmepse.aml	GIS-VIEW - classify Minor/Major/MEPSE point sources based on emissions or stack parameters
mape_compare.aml	Compare the geographic extents of coverages and map display
message.aml	GIS-VIEW - display a variety of message banners and forms
mvascii.aml	MOBILE - create road and / or area surrogate coverages - main program
mvascii1.aml	MOBILE - grenerate road coverage from ASCII file
mveuna.aml	MOBILE - erase offnetwork roads from a coverage - main program
mveuna1.aml	MOBILE - erase offnetwork roads from a coverage
mvexnatl.aml	MOBILE - extract surrogates from national coverages
mvextig.aml	MOBILE - extract surrogates for off-net vmt from national TIGER/Line coverages
mvextrc1.aml	MOBILE - generate surrogate files - processor
mvextrct.aml	MOBILE - generate network files - main program
mvnagd.aml	MOBILE - overlay network line coverages with grid - main

	program
mvnagd1.aml	MOBILE - overlay network line coverages with grid - processor
mvnativ.aml	MOBILE - open an interactive ARC/INFO session
mvpagd.aml	MOBILE - overlay network area (polygon) coverages with grid - main program
mvpagd1.aml	MOBILE - overlay network area (polygon) coverages with grid - processor
mvpaif.aml	MOBILE - import coverages from ARC/INFO export format files - main program
prj_compare.aml	Compare the map projections of two coverages
psg.aml	Point Source Generation - create point source coverage - main program
psg_create.aml	Point Source Generation - processor
psgp.aml	Point Source Grid Processor - overlay point source and grid coverages - main program
psgp_create.aml	Point Source Grid Processor - overlay point source and grid coverages - processor
psgp_prj.aml	Point Source Grid Processor - overlay point source and grid coverages - project coverage
psgp_rundsc.aml	Point Source Grid Processor - overlay point source and grid coverages - verify files
pz_main.aml	GIS-VIEW - pan and zoom tools to control map extent
reset_menus.aml	Redraw menus after pin has been pulled
runmode.aml	Set ARC/INFO environment for interactive or batch mode
scalebar.aml	GIS-VIEW Draw a scale bar on a map
screenmap.aml	GIS-VIEW - create a hardcopy map based on the extent and data visible in the current display
screensize.aml	GIS-VIEW - utility to determine screen dimensions (in pixels)
set_menu_size.aml	GIS-VIEW - utility to set default sizes for menu windows based on screen size
sscp.aml	Spatial Surrogate Coverage Processor - generate ASCII files with gridded data for land use, census, and network coverages - main program
sscp_create.aml	Spatial Surrogate Coverage Processor - processor
startems.aml	main program for all ARC/INFO calls
stdname.aml	renames coverages and INFO files using standard names (to make previous versions compatible with version 2.1)

The ARC/INFO icon files are:

hourgls32.icon	hour glass icon used in "wait" menus

The ARC/INFO menus are:

arcprompt.menu	GIS-VIEW - allow direct entry of ArcPlot commands
classedit.menu	GIS-VIEW - Modify emission display class ranges
cmproj.menu	Coverage projection - main menu
cmprojn.menu	Coverage projection - output projection parameters
cmprojo.menu	Coverage projection - input projection parameters
covsel.menu	Land Use / Land Cover Processor (not implemented) - background coverage selection
draw_again.menu	GIS-VIEW - select pollutant for emissions display
draw_again_temp.menu	GIS-VIEW - select pollutant and hour for temporal emissions display
emis_colors.menu	GIS-VIEW - select pollutants and classification for color shaded emissions display
emis_dots.menu	GIS-VIEW - select pollutants and parameters for dot density emissions display
emis_load.menu	GIS-VIEW - user defined request for ASCII emissions files from SAS and conversion to INFO database files for ARC/INFO
featsel.menu	Land Use / Land Cover Processor (not implemented) - feature selection
gdig.menu	Grid Definition Input Generator - select coverages for processing
gemap_grid.menu	GIS-VIEW - display grid parameters
gemap_logexp.menu	GIS-VIEW - logical expression query builder for coverage feature selection
gemap_main.menu	GIS-VIEW - main menu
gemap_numsel.menu	GIS-VIEW - display number of selected features for each coverage
gemap_query.menu	GIS-VIEW - graphical selection of coverage features
gemap_sym.menu	GIS-VIEW - select a display symbol for each coverage
gemap_symsets.menu	GIS-VIEW - select from families of symbols for use in symbol selection
gemap_theme.menu	GIS-VIEW - determine which coverages draw on the display (and show their symbols)
gemap_varenv.menu	GIS-VIEW - display and modify environment variable settings
list_items.menu	GIS-VIEW - select database items to list for selected features
lulcmain.menu	Land Use / Land Cover Processor (not implemented) - main menu
majmepse.menu	GIS-VIEW - enter emission and stack parameters for Minor/Major/MEPSE point source classification
majmepse_leg.menu	GIS-VIEW - display Minor/Major/MEPSE point source classification results

message.menu	Generic message display menu
message_inp.menu	Generic menu to request user text input
message_ok.menu	Generic menu to display a message and wait for user input
message_tol.menu	Request numeric input to override default spatial overlay setting
message_yns.menu	Generic menu to request a user decision (yes / no)
mveuna.menu	MOBILE - erase offnetwork roads from a coverage
mvextrct.menu	MOBILE - generate network files
mvnagd.menu	MOBILE - overlay network line coverages with grid
mvpagd.menu	MOBILE - overlay network area (polygon) coverages with grid
mvpaif.menu	MOBILE - import coverages from ARC/INFO export format files
mvroads.menu	MOBILE - create road and / or area surrogate coverages - select data source
pz_main.menu	GIS-VIEW - pan and zoom tools to control map extent
sscp.menu	Spatial Surrogate Coverage Processor - generate ASCII files with gridded data for land use, census, and network coverages
substudy.menu	GIS-VIEW - select, delete, and create substudies of the current grid
substudy_create.menu	GIS-VIEW - select a method for creating a substudy of the current grid

The ARC/INFO text (ASCII) files are:

cem.txt	GIS-VIEW - help on "CEM" menu items
charts.txt	GIS-VIEW - help on "Charts" menu items
emissions.txt	GIS-VIEW - help on "Emissions" menu items
file.txt	GIS-VIEW - help on "File" menu items
general.txt	GIS-VIEW - general overview help file
majmepse.txt	GIS-VIEW - help on "MEPSE/Major" menu items
query.txt	GIS-VIEW - help on "Query" menu items
samples.txt	GIS-VIEW - help on performing some tasks in GIS-VIEW
substudy.txt	GIS-VIEW - help on "Substudy" menu items
tools.txt	GIS-VIEW - help on "Tools" menu items
view.txt	GIS-VIEW - help on "View" menu items

3.2 System Data Directory

The system data directory (lib) contains two subdirectories, lookup and sysdata. The lookup subdirectory contains base files used to initialize the lookup table directory under a new project. The sysdata directory contains system level data used during EMPRO execution. The name of the sysdata directory is contained in environment variable EMS_SYS during EMPRO

execution.

The files in the lookup subdirectory are shown below. Unless otherwise noted the files are ASCII data sets. The speciation files contain base data from RADM speciation factors that can be modified for use in a generic speciation mechanism.

beis2_frost	BEIS-2 Frost Data
beis2_lumap	BEIS-2 Vegid to landuse code
beis2_summer	BEIS-2 Summer Emission Factors
beis2_winter	BEIS-2 Winter Emission Factors
polutant_table	Pollutant selection table
lookup.dat	SAS transport file containing standard lookup tables
anrsd.in	Speciation - Non-ROG SAROAD codes
cat_radm2.in	Speciation - Category to species table (RADM 15 species)
cat_radm2b.in	Speciation - Category to species table (RADM 32 species)
class.in	Speciation - SAROAD code to speciation category
prof_radm2.in	Speciation - Non-computed split factors (RADM 15 species)
prof_radm2b.in	Speciation - Non-computed split factors (RADM 32 species)
spprcr.in	Speciation - SCC to hydrocarbon speciation profile mapping data
ssppr.in	Speciation - Profile to SAROAD table

The files in the sysdata subdirectory are shown below. The ssd01 files are SAS data sets for the Sun. The sd2 files are SAS data sets for the NT. The other files are ASCII data sets that were used to create lookup tables.

acblu.in	CB4 Speciation - lumping data
acbmd.in	CB4 Speciation - model species data
achd.in	CB4 Speciation - VOC chemical data table
anrsd.in	RADM/CB4 Speciation - Non-ROG SAROAD codes
asppr.in	CB4 Speciation - speciation profiles
cat_radm2.in	RADM Speciation - category to species table
class.in	RADM Speciation - SAROAD code to speciation category
hourprof.in	Hourly profiles for original EMPRO point and area temporalization
lulcdes.in	BIOEM - landuse/land cover descriptions
lulcpc.in	BIOEM - landuse/land cover plant community cross reference
lulcspec.in	BIOEM - landuse/land cover plant species cross reference
oic_asct.in	OIC to area source classification code (ASCT) table
oic_ces.in	OIC to CES code table recieved from ARB
oic_desc.in	OIC descriptions
oic_scc.in	OIC to SCC table

prof_cb4.in	CB4 Speciation - non-computed split factors
prof_radm2.in	RADM Speciation - non-computed split factors
scc_tabl.in	Source Classification Code (SCC) descriptions
sic_tabl.in	Standard Industrial Classification (SIC) descriptions
spprcr.in	RADM/CB4 Speciation - SCC to hydrocarbon speciation profile mapping data
srg_cens.in	AREA - table mapping census codes to spatial surrogate codes
srg_lu.in	AREA - table mapping landuse codes to spatial surrogate codes
srg_tigr.in	AREA - table mapping TIGER/line census feature classification codes to spatial surrogate codes
ssad.in	AREA - table mapping SCCs to spatial surrogate codes
ssd.in	AREA - spatial surrogate code descriptions
ssppr.in	RADM/CB4 Speciation - Profile to SAROAD table
weeklyop.in	Weekly operation codes for original EMPRO point and area temporalization
areaclas.mv	MOBILE - area type descriptions
defvmix.mv	MOBILE - default vehicle mix profiles
emisclss.mv	MOBILE - emission classes
facclas.mv	MOBILE - facility type (road type) descriptions
fracvmtd.mv	MOBILE - default vmt temporal profiles
procesmv.mv	MOBILE - process descriptions
spdadjd.mv	MOBILE - default transportation speeds
techtymv.mv	MOBILE - techtype descriptions
tigerfac.mv	MOBILE - table mapping TIGER/Line data to area and road types
vmixordr.mv	MOBILE - vehicle type descriptions
map_cy.ssd01	County map file for North America
map_st.ssd01	State map file for North America
profile.ssd01	RADM/CB4 Speciation - split factors
rogtotog.ssd01	RADM/CB4 Speciation - ROG-to-TOG adjustment factors
scblu.ssd01	CB4 Speciation - lumping data
scbmd.ssd01	CB4 Speciation - model species data
schdt.ssd01	CB4 Speciation - VOC chemical data table
spprcr.ssd01	RADM/CB4 Speciation - SCC to hydrocarbon speciation profile mapping data
ssppr.ssd01	CB4 Speciation - speciation profile
map_cy.sd2	County map file for North America
map_st.sd2	State map file for North America
profile.sd2	RADM/CB4 Speciation - split factors
rogtotog.sd2	RADM/CB4 Speciation - ROG-to-TOG adjustment factors
scblu.sd2	CB4 Speciation - lumping data

scbmd.sd2	CB4 Speciation - model species data
schdt.sd2	CB4 Speciation - VOC chemical data table
spprcr.sd2	RADM/CB4 Speciation - SCC to hydrocarbon speciation profile mapping data
ssppr.sd2	CB4 Speciation - speciation profile

3.3 Source File Directory

The source directory (src) basically contains files used to create the system executables. This directory is not needed to execute the software. The EMPRO source directory contains the following types of files:

Make files used to compile the FORTRAN programs
Source code for FORTRAN programs (.f)
PMENU programs for SAS interface screens (.pmenu)
SCL programs for SAS interface screens (.pgm)
Layouts for SAS interface screens (.scr)

The SAS interface screens are completely defined in the SAS program library gip.sct01 in the executable directory. The SCL program files are included in the program library. The layout files (.scr) are here for informational reasons only, the full screen definition is contained in the SAS program library.

The PMENU programs used by the SAS/AF Screens are:

main.pmenu	Main screen menus (stand-alone execution)
main_gip.pmenu	Main screen menus (execution through MEPPS)

The SCL programs for SAS/AF screens are:

arctlld.pgm	Area Load control factors
ardayld.pgm	Area Day-specific file load
ardayqa.pgm	Area Day-specific file QC report
areasrc.pgm	Area Source processor (grids area source emissions)
areassap.pgm	Area Spatial surrogates - Check assignments file
areassdp.pgm	Area Spatial surrogates - Check lookup tables
areassr.pgm	Area Spatial surrogate processor (US)
areatapf.pgm	Area Temporal allocation processor
areatpg.pgm	Area Temporal profile generator
argrold.pgm	Area Load growth factors
arload.pgm	Area Foundation file load

arqa.pgm	Area Foundation file QC reports
arssrcan.pgm	Area Spatial surrogate processor (CANADA)
b2ecp.pgm	BEIS-2 Calculate emissions
b2edit.pgm	BEIS-2 Edit tables
b2ssc.pgm	BEIS-2 Get coverages
b2ssr.pgm	BEIS-2 Compute gridding ratios
bldname.pgm	Build full file name from name levels
bldprog.pgm	Build full file name for an executable
checkdir.pgm	Check if directory exists
dayload.pgm	Original SCL program for day-specific file load (included)
dummy.pgm	Dummy screen program
filechek.pgm	Include file containing SCL instructions to execute a model step. Used in original EMPRO
filechex.pgm	Include file containing SCL instructions to execute a model step. Improved to allow background execution of step.
fileload.pgm	Include file containing SCL instructions for an ASCII file load. Used in unchanged sections of Motor Vehicle.
getdir.pgm	Get directory name
getfile.pgm	Select file name from a directory listing
getrun.pgm	Get case time information
getspec.pgm	Get case speciation information
hbatch.pgm	HELP BATCH run description
hbldasgn.pgm	HELP Run build assign program file
hbldexec.pgm	HELP Run build execute program file
hbuild.pgm	HELP Run build description
hpush.pgm	HELP Pushbutton description
hrun.pgm	HELP Interactive run description
main.pgm	Main screen (stand-alone execution)
main_gip.pgm	Main screen (execution through input processor)
mv5a_dig.pgm	MovEM MOBILE 5A data input generator
mvascii.pgm	MovEM Process On-network ASCII files for GIS
mvdir.pgm	MovEM Foundation file load
mveced.pgm	MovEM Assign emission factors
mvecnet.pgm	MovEM Network link-specific emission calculation
mvecoff.pgm	MovEM Off-network county level emission calculation
mvecofp.pgm	MovEM Land Survey 1/4 section emission calculation
mvexnatl.pgm	MovEM Extract Data from National Coverages
mvgrnet.pgm	MovEM Grid Network link-specific vehicle miles travelled
mvgroff.pgm	MovEM Grid off-network county level vehicle miles travelled
mvgrofp.pgm	MovEM Grid Land Survey 1/4 section vehicle miles travelled
mvnet.pgm	MovEM Process network definition file

mvplotb.pgm	MovEM Foundation file QC plots
mvsapld.pgm	MovEM Speed file load
mvtapld.pgm	MovEM Temporal adjustment file load
ptcemld.pgm	Point CEM file load
ptdayld.pgm	Point Day day-specific file load
ptgrold.pgm	Point Load growth factors
ptload.pgm	Point Foundation file load
ptqa.pgm	Point Foundation file QC reports
ptsrc.pgm	Point Source processor (grids emissions)
pttapf.pgm	Point Temporal allocation processor
pttoarc.pgm	Point Grid processor (assigns points to grid cells using ARC)
pttpg.pgm	Point Temporal profile generator
qaplotb.pgm	Foundation file QC plots for Point and Area
runbatch.pgm	Submit batch run
runbld.pgm	Run Build - Add step to program file
runblda.pgm	Run Build - Assign program file
runbldin.pgm	Run Build - Initialize SAS data set
runbldx.pgm	Run Build - Execute program file (Submit in background)
sasfile.pgm	Check if file a SAS data set
setenv.pgm	Set an environment variable
spcaprof.pgm	Speciation RADM - Compute split factors
spcaprox.pgm	Speciation Other - Compute split factors
spcbdp.pgm	Speciation CB-IV - Load lumping data
spcbsfp.pgm	Speciation CB-IV - Compute split factors
spchadj.pgm	Speciation Edit Adjustment Factors
spchdp.pgm	Speciation CB-IV - Load VOC chemical data
spchmpro.pgm	Speciation Processor (Speciates emissions)
spchplot.pgm	Speciation QC plots
spcspp.pgm	Speciation CB-IV - Compute ROG to TOG factors
specfl.pgm	Speciation Initialize data sets
spldmap.pgm	Speciation RADM - Load profile mapping table
spldmapx.pgm	Speciation Other - Load profile mapping table
spldprc.pgm	Speciation CB-IV - Load profile mapping table
spldprof.pgm	Speciation RADM/Other - Load split factors (non-HC, biogenic, ...)
spldrgtg.pgm	Speciation RADM/Other - Load ROG to TOG factors
uichkall.pgm	Check if environment variables set to run EMPRO - Routine 1
uichkenv.pgm	Check if environment variables set to run EMPRO - Routine 3
uichkgrd.pgm	Check if environment variables set to run EMPRO - Routine 2
uiconfrm.pgm	Ask user to confirm an action
uiedgrid.pgm	Edit grid description

uiedrun.pgm	Edit scenario description
uigdig.pgm	Grid Definition Input Generator
uilookup.pgm	Utility Edit a lookup table
uimastgr.pgm	Check if grid master directory exists
uiopen.pgm	Open a grid or case for processing
uipopmsg.pgm	Popup message screen
xchek.pgm	Include file containing SCL instructions for loading ASCII files
xdayld.pgm	Include file containing SCL instructions for ASCII day-specific files
xdayqa.pgm	Include file containing SCL instructions for day-specific file QC reports
xload.pgm	Include file containing SCL instructions for loading ASCII foundation files
xqa.pgm	Include file containing SCL instructions for foundation file QC reports
xqaday.pgm	Include file containing SCL instructions for day-specific file QC reports

SAS/AF screen layouts are shown in the following files:

ardayld.scr	Area Day-specific file load
ardayqa.scr	Area Day-specific file QC report
areasrc.scr	Area Source processor (grids area source emissions)
areassap.scr	Area Spatial surrogates - Check assignments file
areassdp.scr	Area Spatial surrogates - Check lookup tables
areassr.scr	Area Spatial surrogate processor (US)
areatapf.scr	Area Temporal allocation processor
areatpg.scr	Area Temporal profile generator
arload.scr	Area Foundation file load
arqa.scr	Area Foundation file QC reports
arssrcan.scr	Area Spatial surrogate processor (CANADA)
b2ecp.scr	BEIS-2 Calculate emissions
b2edit.scr	BEIS-2 Edit tables
b2ssc.scr	BEIS-2 Get coverages
b2ssr.scr	BEIS-2 Compute gridding ratios
dummy.scr	Dummy screen program
hbatch.scr	HELP BATCH run description
hbldasgn.scr	HELP Run build assign program file
hbldexec.scr	HELP Run build execute program file
hbuild.scr	HELP Run build description
hpush.scr	HELP Pushbutton description

hrun.scr	HELP Interactive run description
main.scr	Main screen (stand-alone execution)
main_gip.scr	Main screen (execution through input processor)
mv5a_dig.scr	MovEM MOBILE 5A data input generator
mvascii.scr	MovEM Process On-network ASCII Files
mvdir.scr	MovEM Foundation file load
mveced.scr	MovEM Assign emission factors
mvecnet.scr	MovEM Network link-specific emission calculation
mvecoff.scr	MovEM Off-network county level emission calculation
mvecofp.scr	MovEM Land Survey 1/4 section emission calculation
mvexnatl.scr	MovEM Extract Data from National Coverages
mvgrnet.scr	MovEM Grid Network link-specific vehicle miles travelled
mvgroff.scr	MovEM Grid off-network county level vehicle miles travelled
mvgrofp.scr	MovEM Grid Land Survey 1/4 section vehicle miles travelled
mvnet.scr	MovEM Process network definition file
mvplotb.scr	MovEM Foundation file QC plots
mvsapld.scr	MovEM Speed file load
mvtapld.scr	MovEM Temporal adjustment file load
ptcemld.scr	Point Load CEM Data
ptdayld.scr	Point Load Day-Specific Data
ptload.scr	Point Foundation file load
ptqa.scr	Point Foundation file QC reports
ptsrc.scr	Point Source processor (grids emissions)
pttapf.scr	Point Temporal allocation processor
pttoarc.scr	Point Grid processor (assigns points to grid cells using ARC)
pttpg.scr	Point Temporal profile generator
qaplotb.scr	Foundation file QC plots for Point and Area
runbatch.scr	Submit batch run
runbld.scr	Run Build - Add step to program file
runblda.scr	Run Build - Assign program file
runbldin.scr	Run Build - Initialize SAS data set
runbldx.scr	Run Build - Execute program file (Submit in background)
sasfile.scr	Check if file a SAS data set
sefp.scr	BIOEM Bottom-up emission flux processor
setenv.scr	Set an environment variable
spcaprof.scr	Speciation RADM - Compute split factors
spcaprox.scr	Speciation Other - Compute split factors
spcbdp.scr	Speciation CB-IV - Load lumping data
spcbsfp.scr	Speciation CB-IV - Compute split factors
spchadj.scr	Speciation Edit adjustment factors
spchdp.scr	Speciation CB-IV - Load VOC chemical data

spchmpro.scr	Speciation Processor (Speciates emissions)
spchplot.scr	Speciation QC plots
spcspp.scr	Speciation CB-IV - Compute ROG to TOG factors
specfl.scr	Speciation Initialize data sets
spldmap.scr	Speciation RADM - Load profile mapping table
spldmapx.scr	Speciation Other - Load profile mapping table
spldprc.scr	Speciation CB-IV - Load profile mapping table
spldprof.scr	Speciation RADM/Other - Load split factors (non-HC, biogenic, ...)
spldrgtg.scr	Speciation RADM/Other - Load ROG to TOG factors
uichkall.scr	Check if environment variables set to run EMPRO - Routine 1
uichkenv.scr	Check if environment variables set to run EMPRO - Routine 3
uichkgrd.scr	Check if environment variables set to run EMPRO - Routine 2
uiconfrm.scr	Ask user to convirm an action
uiedgrid.scr	Edit grid description
uiedrun.scr	Edit scenario description
uigdig.scr	Grid Definition Input Generator
uilookup.scr	Utility Edit a lookup table
uimastgr.scr	Check if grid master directory exists
uiopen.scr	Open a grid or scenario for processing
uipopmsg.scr	Popup message screen

SAS/AF Help Screen Layouts are shown in the following files:

ardayfl.hlp	Area Day-Specific File
ascdirek.hlp	Directory Name
begdate.hlp	Case Start
cat_spid.hlp	Category-Species Table (Speciation - Compute Split Factors)
direk.hlp	Directory Name
domain.hlp	Source Name
domdir.hlp	Source (raw data) directory name
emfacyr.hlp	Emission Factor Year (MV Emission Calculation)
ep_day.hlp	Episode Day
ex_facts.hlp	Non-computed split factors file (Speciation - Compute Split Factors)
grid.hlp	Grid Name
mech.hlp	Split Factor ID
mechtype.hlp	Chemical Mechanism
mv5dom.hlp	Source Name for MOBILE-5A base data
mv5opmod.hlp	Opmode fields on MOBILE-5A screen
mv5speed.hlp	Speed fields on MOBILE-5A screen

mv5temp.hlp	Temperature fields on MOBILE-5A screen
non_rog.hlp	Non-ROG SAROAD Codes File (Speciation - Compute Split Factors)
ogtype.hlp	Organic Gas Type
prof_srd.hlp	Profile-SAROAD Table (Speciation - Compute Split Factors)
ptdayfl.hlp	Point Day-Specific File
rundira.hlp	Run Build Assign - Directory Field
rundirx.hlp	Run Build Execute - Directory Field
sasfile.hlp	Full name of SAS data set
sasproga.hlp	Name of SAS program file assigned to run build
sasprogx.hlp	Name of SAS program file to be executed
scen.hlp	Case Name
srd_cat.hlp	SAROAD_Category Table (Speciation - Compute Split Factors)
timezone.hlp	Timezone

4.0 MEPPS USER DATA DIRECTORIES

The MEPPS data directories are derived from the following environmental variables:

EMS_HOME	Home Directory (Models-3 Workspace Path)
EMS_PROJECT	Study ID
EMS_GRID	Grid Name
EMS_SCENARIO	Case Name
EMS_DOMAIN	Source Name

In general, a study can contain the following types of data:

Lookup tables
Raw data
Grid data
Case data

The following sections describe where the types of data are stored and how MEPPS handles multi-day cases.

4.1 Lookup Tables

Lookup tables are defined at the study level. The full path name of the directory that contains the lookup tables is:

$EMS_HOME/project/$EMS_PROJECT/lookup

Two environmental variables contain the directory name:

EMS_CAT	SAS library reference ems_cat.
EMS_DATA	SAS file reference ems_data.

4.2 Raw Data

Environment variable EMS_LOC defines the full path name of a raw data directory. It is derived in the following manner:

$EMS_HOME/project/$EMS_PROJECT/raw_data/$EMS_DOMAIN/common

The raw data directories contain ASCII files which are loaded into EMPRO. Generally these contain emission data extracted from standard emission inventories.

A study may have multiple raw data directories that contain data for different geographic areas and conditions.

4.3 Grid Data

Environment variable EMS_GRID defines the grid name. MEPPS defines the master directory for a grid in the following manner:

$EMS_HOME/project/$EMS_PROJECT/gridspec/$EMS_GRID

This directory has the following subdirectories:

Subdirectory	Type of Data	Variable	Environment SAS Reference
common/sa	Ungridded Emission	EMS	library EMS
		EMS_CVRT	file EMS_CVRT
common/gis	Ungridded GIS	EMS	file EMSG
sas	Gridded Emission	EMS_GRD	library EMS_GRD
		EMSF_GRD	file EMSF_GRD
gis	Gridded GIS	EMSG	file EMSG_GRD
scenario	Case data		

The ungridded data directories contain data for the grid's geographic area, that has not been processed for a grid. MEPPS loads the data from the raw data directories into the EMS directory. The EMSG directory contains coverages that have not been overlayed with a grid.

The gridded data directories contained data that has been processed for the grid. The EMS_GRD directory contains the grid description and gridded emission data. The EMSG_GRD directory contains coverages that have been overlayed with the grid.

4.4 Case Data

All case data for a grid is contained in subdirectory scenario under the grid master directory. Subdirectories for specific cases are defined using environment variable EMS_SCENARIO in the following manner:

$EMS_HOME/project/$EMS_PROJECT/gridspec/$EMS_GRID/
scenario/$EMS_SCENARIO/sas

The full path name of the case directory is contained in the following environment variables:

EMS_RUN	SAS library reference EMS_RUN
EMS_FILE	SAS file reference EMS_FILE

The case directory contains data calculated by the emission processor for a 24 hour episode. The episode covers the period midnight to midnight in the case time zone. All case processing in EMPRO is for a single 24 hour episode.

4.5 Multi-Day Cases

MEPPS can create and process multi-day cases. Basically the system creates an episode directory for each day in the case. The names of each episode will be the case prefix followed by the day number. The episode directories will be created under directory:

$EMS_HOME/project/$EMS_PROJECT/gridspec/$EMS_GRID/scenario

For example, if the following parameters are entered on the create case screen:

Case Prefix: day
Start Date: 08/01/9 Number of Days: 3

MEPPS will create the following episodes:

Name	Date	Subdirectory
day1	08/01/95	day1/sas
day2	08/02/95	day2/sas
day3	08/03/95	day3/sas

The master case is the one defined for the first day. It will have a run description file and the case episode file. The remaining cases will only contain a run description file.

The name of the master case will be stored in the environment variable EMS_SCENARIO and form the basis for EMS_RUN and EMS_FILE.

The case episode file defines the episode directories that each emission model will use for the episode days in the period. The case episode file generated for the preceding example is shown below. The case period is Tuesday to Thursday. All days in the period are summer weekdays.

Day Number	Date	Area Episode	Point Episode	Mobile Episode	Biogenic Episode
1	08/01/95	day1	day1	day1	day1
2	08/02/95	day1	day1	day2	day2
3	08/03/95	day1	day1	day3	day3

In this example, MEPPS will only compute point and area source emissions for episode day1. It will compute mobile and biogenic emissions for episodes day1, day2, and day3. The MEPPS Output Processor functions will use area and point emissions from episode day1 for all episode days in the case period.

5.0 Environment Variables Entered via Study Planner

Environment variables can be used to pass information to studies. They may be used in execution scripts that drive programs, and they may be used by the programs themselves. Following is a list of environment variables used by emission software when run from Study Planner:

Global environment variables

MEPPS_HOME	Base directory for MEPPS software + system data
EMPRO_HOME	Base directory for EMPRO software + system data
EMS_HOME	Home directory for emission files
EMS_PROJECT	Project ID (Models-3 Study ID)
EMS_GRID	Grid Name
EMS_SCENARIO	Case Name

Environment variable to extract emissions data from a national inventory
(Used by Models-3 programs: mepps_extract_ar, mepps_extract_mv, and mepps_extract_pt)

EMS_DOMAIN	Source directory name

Environment variables to extract emissions data from a user-specified inventory
(Used by Models-3 programs: mepps_exuser_ar, mepps_exuser_mv, and mepps_exuser_pt)

FORMAT	Format of inventory (NET or MIDPRO)
INVENTORY	Full name of inventory file
COUNTRY	Name of country on inventory (US, CANADA, MEXICO)
EMS_DOMAIN	Source directory name

Environment variables to load meteorological data into case directories
(Used by Models-3 program: mepps_load_met)

MET_CRO_2D	Full (path/file) name of Cross 2D Met file
MET_CRO_3D	Full (path/file) name of Cross 3D Met file
MET_DOT_3D	Full (path/file) name of Dot 3D Met file
GRID_DOT_2D	Full (path/file) name of Grid Dot 2D file
MET_DATE	Met data-day to be loaded into first day of case (yyyy/mm/dd). If MET_DATE is not specified it defaults to the case start date.

Environment variables to load foundation data for a model
(Used by programs: mepps_load_ar, mepps_load_mv, mepps_load_pt)

EMS_DOMAIN	Source directory name (optional)
EMS_LOC	Raw directory name (optional)

Either EMS_LOC or EMS_DOMAIN must be input. If EMS_LOC is input, it will be used directly as the raw directory name. Otherwise the directory name is built from EMS_HOME, EMS_PROJECT and EMS_DOMAIN.

Environment variables that invoke generation of spatial surrogate files
(Used by Models-3 programs: mepps_grid_ar, mepps_grid_mv, mepps_grid_pt)

File will be generated if environment variable is yes or y (case-insensitive). If the environment variable is missing or contains another value, the file is not generated.

Area	
COUNTY	Gridded County Areas
CENSUS	Gridded Census Tract Areas

Biogenics

COUNTY	Gridded County Areas

Mobile

COUNTY	Gridded County Areas
FHAROAD	Gridded FHA Roads
TIGER	Gridded TIGER/Line Roads for Off-Net

Environment variables to calculate emissions for a model
(Used by Models-3 programs: mepps_calc_ar, mepps_calc_bi, mepps_calc_mv, mepps_calc_pt)

TAFLIB	Name of directory containing TAF File	(ar,pt)
TAFNAME	Name of TAF File	(ar,pt)
BIOMASS	Biomass option	(bi)
EF_TYPE	Emission Factor Type	(mv)
EF_YEAR	Emission Factor Year	(mv)

TAFLIB and TAFNAME are used to define the name of a TAF SAS data set in the following form taflib.tafname. The default is to a standard file for model.

BIOMASS specifies one of the following options:

S	summer biomass
W	winter biomass
other	determine biomass from frost data (default)

EF_TYPE specifies one of the following two ways of assigning emission factors:

HR	by hour
AF	by areatype/factype (road type)

Environment variable to adjust emissions speciation for a model
(Used by Models-3 programs: mepps_spec_ar, mepps_spec_bi, mepps_spec_mv, mepps_spec_pt)

POLADJ Pollutant adjustment option

If POLADJ is YES or Y, the emissions input to speciation are adjusted. If POLADJ is any other value or is missing, no adjustments are made.

Environment variable to define stack splits
(Used by Models-3 program: mepps_def_splits)

UNITS	Units type (MKS or ENGLISH)
MAJOR_COND	SAS logical expression used to select major stacks
MEPSE_COND	SAS logical expression used to select MEPSE groups

Environment variables to generate Major and MEPSE stack I/O API Files
(Uses Models-s program: mepps_create_stk

MOLES	Generate Moles Emission Files (YES,NO)
GRAMS	Generate Grams Emission Files (YES,NO)
STAK_MAJOR	Full Name of Major Stack File
CRO_SP_MAJOR	Full Name of Major Stack Emission File (moles)
CRO_SP_MAJG	Full Name of Major Stack Emission File (grams)
STAK_MEPSE	Full Name of MEPSE Stack File
CRO_SP_MEPSE	Full Name of MEPSE Stack Emission File (moles)
CRO_SP_MEPG	Full Name of MEPSE Stack Emission File (grams)
LOGFILE	Full Name of I/O API Log File

Environment variables to generate I/O API files containing 2D speciated emission files
(Used by Models-3 program: mepps_create_2d)

MOLES	Generate Moles Emission File (YES,NO)
GRAMS	Generate Grams Emission File (YES,NO)
SPEMIS2D	Full Name of 2D Speciated Emission File (moles)
SPEMIS2D_G	Full Name of 2D Speciated Emission File (grams)
LOGFILE	Full Name of IO/AP Log File
AREA	Include Area Source Emissions (YES or NO)
POINT	Include Point Source Emissions (YES or NO)
MOBILE	Include Mobile Source Emissions (YES or NO)
BIO	Include Biogenic Emissions (YES or NO)

Environment variables to run MEPRO from Study Planner

MEPRO_ST_CNTY	Full name for file of FIPS state/county codes for study area
MEPRO_STUDYDIR	Full name for existing or new MEPRO study directory
MEPRO_OUTDIR	Sub-directory to study directory for output for EMPRO
MEPRO_POINT	1 to perform point-source projections
MEPRO_AREA	1 to perform area-source projections
MEPRO_MOBILE	1 to perform mobile-source projections

MEPRO_POLLUTANT	CO, NOX, PM10, SO2, VOC, or blank for (CO, NOX, VOC)
MEPRO_PROJYEAR	Projection year (1991 through 2010 are valid)

APPENDIX L

Inventory Data Analyzer Software, Data Files, and Environment Variables

1.0 GENERAL DESCRIPTION

The Inventory Data Analyzer (IDA) is a combination of SAS, SAS/AF, and ARC/INFO programs. It is run from the Models-3 Tools window. It can not be run as part of a study plan. The following file suffixes have been used throughout the system:

.aml	ARC/INFO program (ARC Macro Language)
.bat	Windows NT execution scripts
.hlp	SAS/AF help screen layouts
.icon	ARC/INFO icon
.menu	ARC/INFO menu
.pgm	SCL program for SAS/AF screens
.pmenu	SAS program to define PMENUs (selection buttons on top of screen)
.sas	SAS program
.sasm	File containing macros used by SAS programs
.sasl	INCLUDE file containing link modules for SAS programs
.scr	SAS/AF screen layouts
.sct01	SAS program library for Sun, contains SAS/AF screen programs
.sc2	SAS program library for NT, contains SAS/AF screen programs
.ssd01	SAS data set for Sun
.sd2	SAS data set for NT
.txt	ARC/INFO text file

During IDA execution, the environment variable MIDPRO_HOME contains the name of the base directory for IDA software and system data. The following subdirectories exist under the MEPPS_HOME directory:

bin	Executable directory. Contains all software files needed for system execution. This inciudes scripts, SAS/AF program libraries, and SAS programs.
lib	System data directory. Contains system lookup tables
src	Source file director

The bin, lib, and inv directories are required for system execution. The files in the src are only used to create the files in the bin directory.

The names of these directories are stored in the following environment variables during MEPPS execution:

MID_BIN	MEPPS executable directory ($MEPPS_HOME/bin)
MID_SYS	MEPPS system tables($MEPPS_HOME/lib

1.1 Executable Directory

The executable directory contains the software files necessary to execute the system. The name of this directory is stored in environment variable MID_BIN when IDA is executing. The IDA executable directory contains the following types of files:

UNIX scripts (no suffix)
NT scripts (.bat)
SAS program libraries for Sun (.sct01)
SAS program libraries for NT (.sc2)
SAS programs (.sas)
SAS program include files (.sasl)
SAS macros (.sasm)
ARC/INFO programs (.aml)
ARC/INFO menu files (.menu)
ARC/INFO icon files (.icon)
ARC/INFO test files (.txt)

The UNIX scripts are:

midpro	IDA execution script
xARC/INFO	Execute ARC/INFO aml from within IDA

The NT scripts are:

midpro.bat	IDA execution script
xARC/INFO .bat	Execute ARC/INFO aml from within IDA
chgdir.bat	Change current drive and path

The SAS program libraries are:

midpro.sct01	SAS program library for Sun, contains IDA screen programs
midpro.sc2	SAS program library for NT, contains IDA screen programs
midpro.trans	Transport version of SAS program library

The SAS programs are:

auto_nt.sas	IDA autoexec program for NT, starts IDA screen interface
autoexec.sas	IDA autoexec program for UNIX, starts IDA screen interface

MIDPRO_CIMPORT.sas	Import EMPRO program library from transport file (ad hoc program)
MIDPRO_CPORT.sas	Port EMPRO program library to a transport file (ad hoc program)
aradd.sas	Import area emissions from another working directory
areachek.sas	Check table mapping states/counties to geographic areas
areaload.sas	Load table mapping states/counties to geographic areas
arload.sas	Import area emissions from a NET or IDA format inventory
arqaemck.sas	QC area emissions - Generate QC Report
arqaemco.sas	QC area emissions - Correct Emissions
arqaemed.sas	QC area emissions - Edit
arqaso4.sas	QC area emissions - Add SO_4 Emissions
arwrit.sas	Export area emissions to a NET or IDA format inventory
defareaa.sas	Define Geographic Area - Set to Entire Country
defareas.sas	Define Geographic Area - Select from state and county tables
defareaw.sas	Define Geographic Area - Copy from Another Working Directory
defdesc.sas	Create working directory description file
editset.sas	Set edit level for editor statement
export_data.sas	Create transport version of IDA system data library (ad hoc program)
getdesc.sas	Get information from working directory description file
gethdr.sas	Get header information from an IDA format inventory
import_data.sas	Import IDA system data library from a transport file (ad hoc program)
makdir_nt.sas	Create directory on NT, filling in intermediate directories that do not exist.
mvadd.sas	Import motor vehicle VMT data from another working directory
mvload.sas	Import motor vehicle VMT from an ASCII file
mvqavmck.sas	QC motor vehicle VMT - Generate QC Report
mvqavmed.sas	QC motor vehicle VMT - Edit
mvwrit.sas	Export motor vehicle VMT data
ptadd.sas	Import point emissions from another working directory
ptedgrp.sas	Define Stack Splits - Manually Select MEPSE Stack Groups
ptedstak.sas	Define Stack Splits - Manually Select Major Stacks
ptfreq.sas	Define Stack Splits - Generate Frequency Reports
ptgroup.sas	Define Stack Splits - Assign Stacks to Groups
ptldcem.sas	Import CEM data
ptload.sas	Import point emissions to a NET or IDA format inventory
ptqacem.sas	QC CEM Data - Generate Intermediate File
ptqaceme.sas	QC CEM Data - Edit Intermediate File
ptqacemu.sas	QC CEM Data - Update Point Emissions
ptqaemck.sas	QC point emissions - Generate QC Report

ptqaemco.sas	QC point emissions - Correct Emissions
ptqaemed.sas	QC point emissions - Edit
ptqaemfl.sas	QC point emissions - Add Missing Emissions
ptqaplck.sas	QC point plants - Generate QC Report
ptqapled.sas	QC point plants - Edit
ptqaplfl.sas	QC point plants - Add Missing Plants
ptqaprck.sas	QC point processes - Generate QC Report
ptqapred.sas	QC point processes - Edit
ptqaprfl.sas	QC point processes - Add Missing Processes
ptqaso4.sas	QC point emissions - Add SO_4 Emissions
ptqastck.sas	QC point stacks - Generate QC Report
ptqastco.sas	QC point stacks - Correct Stack Data
ptqasted.sas	QC point stacks - Edit
ptqastfl.sas	QC point stacks - Add Missing Stacks
ptsplitf.sas	Define Stack Splits - Create Stack Definition File
ptvars.sas	Generate macro variables that specify variable lists for stack parameters and pollutants
ptwrit.sas	Export point emissions to a NET or IDA format inventory
selpack.sas	Set states to full selections if all counties in state selected
setctry.sas	Get country number for a working directory
stk_dflt.sas	Generate SCC stack defaults
stk_parm.sas	Generate report showing completeness of stack parameters
stkcount.sas	QC Point Stacks - Analyze Completeness of Stack Parameters by SCC
stkemis.sas	QC Point Stacks - Statistically Analyze Stack Emissions by SCC
stkparm.sas	QC Point Stacks - Statistically Analyze Stack Paramters by SCC
tierload.sas	Load SCC tiers form an ASCII file to a SAS data set
unld_ar.sas	GIS View - Create ASCII county level emission file from area emissions
unld_cem.sas	GIS View - Create ASCII stack level emissions file from CEM data
unld_mv.sas	GIS View - Create ASCII motor vehicle VMT file
unld_pcem.sas	GIS View - Create ASCII plant level emissions file from CEM data
unld_plant.sas	GIS View - Create ASCII plant level emission file from point emissions
unld_pt.sas	GIS View - Create ASCII county level emission file from point emissions
unld_stak.sas	GIS View - Create ASCII stack file. Contain location, stack parameters, and emissions
unld_wdir.sas	GIS View - Create ASCII versions of the working directory description and geographic area file

The SAS program include files are:

areamap.sasl	Link modules included to convert countis to geographic areas
chek_sc.sasl	Link modules included to check states and counties
ptcheksc.sasl	Link modules included to check states, counties, and locations of point source data
selcy.sasl	Link modules included to select counties for the working directory geographic area

The SAS macro files are:

batutil.sasm	Utility macros
ptsplit.sasm	Macro to create ASCII version of stack and stack group base files
ptsplitc.sasm	Macros to classify stacks and stack groups by emissions or stack parameters
rnk_stak.sasm	Macros to generate report ranking stacks by emission values or stack parameters
sccextr.sasm	Macros to extract data for SCC reports
sccrpt.sasm	Macros to generate SCC reports
utility.sasm	Utility macros

The ARC/INFO programs are:

classedit.aml	GIS-VIEW - Modify emission display class ranges
classenv.aml	GIS-VIEW - Control emission display class environment
descread.aml	Read case description information
dict.aml	Utility program for storing and searching global "dictionaries"
draw.aml	GIS-VIEW - generate map display
emis_load.aml	GIS-VIEW - user defined request for ASCII emissions files from SAS and conversion to INFO database files for ARC/INFO (drives gvsasread.aml)
emisdraw.aml	GIS-VIEW - display emission values for any pollutant
err_report.aml	Generate a description of unpredicted errors (write to display and error log file)
gisview.aml	GIS-VIEW - start ArcPlot and the GIS-VIEW main menu
gvexit.aml	GIS-VIEW - delete temporary files and exit GIS-VIEW
gvfmenu.aml	GIS-VIEW - utility program to create a variety of menus
gvmapproj.aml	GIS-VIEW - set map projection for display
gvquery.aml	GIS-VIEW - logical or graphical selection of coverage features - main program

gvsasread.aml	GIS-VIEW - request ASCII emissions files from SAS and create INFO database files for ARC/INFO
gvsymsets.aml	GIS-VIEW - select from families of symbols for map display
gvtools.aml	GIS-VIEW - a collection of map query and display tools
list_items.aml	GIS-VIEW - List selected coverage attributes
majmepse.aml	GIS-VIEW - display stacks or plants classified as Minor/Major/MEPSE point sources
menuopen.aml	GIS-VIEW - utility program to create a variety of menus
message.aml	GIS-VIEW - display a variety of message banners and forms
pz_main.aml	GIS-VIEW - pan and zoom tools to control map extent
runmode.aml	Set ARC/INFO environment for interactive or batch mode
scalebar.aml	GIS-VIEW - draw a scale bar on a map
screenmap.aml	GIS-VIEW - create a hardcopy map based on the extent and data visible in the current display
screensize.aml	GIS-VIEW - utility to determine screen dimensions (in pixels)
set_menu_size.aml	GIS-VIEW - utility to set default sizes for menu windows based on screen size
startems.aml	main program for all ARC/INFO calls

The ARC/INFO menus are:

arcprompt.menu	GIS-VIEW - allow direct entry of ArcPlot commands
classedit.menu	GIS-VIEW - Modify emission display class ranges
draw_again.menu	GIS-VIEW - select pollutant for emissions display
emis_colors.menu	GIS-VIEW - select pollutants and classification for color shaded emissions display
emis_dots.menu	GIS-VIEW - select pollutants and parameters for dot density emissions display
emis_load.menu	GIS-VIEW - user defined request for ASCII emissions files from SAS and conversion to INFO database files for ARC/INFO
gvenv.menu	GIS-VIEW - display and modify environment variable settings
gvlogexp.menu	GIS-VIEW - logical expression query builder for coverage feature selection
gvquery.menu	GIS-VIEW - graphical selection of coverage features
gvsym.menu	GIS-VIEW - select a display symbol for each coverage
gvsymsets.menu	GIS-VIEW - select from families of symbols for use in symbol selection
list_items.menu	GIS-VIEW - select database items to list for selected features
main.menu	GIS-VIEW - main menu
majmepse_leg.menu	GIS-VIEW - Minor/Major/MEPSE stacks legend
majmepse_pleg.menu	GIS-VIEW - Minor/Major/MEPSE plants legend

message.menu	Generic message display menu
message_inp.menu	Generic menu to request user text input
message_ok.menu	Generic menu to display a message and wait for user input
message_yns.menu	Generic menu to request a user decision (yes / no)
numsel.menu	GIS-VIEW - display number of selected features for each coverage
pz_main.menu	GIS-VIEW - pan and zoom tools to control map extent
theme.menu	GIS-VIEW - determine which coverages draw on the display (and show their symbols)

The ARC/INFO text (ASCII) files are:

cem.txt	GIS-VIEW - help on "CEM" menu items
charts.txt	GIS-VIEW - help on "Charts" menu items
emissions.txt	GIS-VIEW - help on "Emissions" menu items
file.txt	GIS-VIEW - help on "File" menu items
general.txt	GIS-VIEW - general overview help file
majmepse.txt	GIS-VIEW - help on "MEPSE/Major" menu items
query.txt	GIS-VIEW - help on "Query" menu items
tools.txt	GIS-VIEW - help on "Tools" menu items
view.txt	GIS-VIEW - help on "View" menu items

1.2 System Data Directory

The system data directory (lib) contains system level data used during IDA execution. The name of the directory is contained in environment variable MID_SYS during execution.

The files in the directory are shown below. All files are SAS data sets.

lookup.dat	SAS transport file containing all tables
country.sd2	Country table (NT version)
county.sd2	County table (NT version)
state.sd2	State table (NT version)
country.ssd01	Country table (Sun version)
county.ssd01	County table (Sun version)
state.ssd01	State table (Sun version)

1.3 Source File Directory

The source directory (src) contains files used to create the system executables. This directory is not needed to execute the software. The IDA source directory contains the following types of files:

PMENU programs for SAS interface screens (.pmenu)
SCL programs for SAS interface screens (.pgm)
Layouts for SAS help screens (.hlp)

The PMENU program used by the SAS/AF screens is:

main.pmenu	PMENU definitions for main screen

The SCL programs used by the SAS/AF screens are:

arload.pgm	Import Area Source Emission Data
arqaemis.pgm	QC Area Source Emission Data
arqaso4.pgm	QC Area - Add SO4 Emissions
arwrit.pgm	Export Area Source Emission Data
bldname.pgm	Utility program to build a full file name from name levels
bldprog.pgm	Utility program to build the file name for an executable
checkdir.pgm	Utility program to check if a directory exists
defarea.pgm	Define Geographic Area
defareaw.pgm	Define Geographic Area - Copy from Another Working Directory
deldirek.pgm	Delete a directory and sub-directories
delfile.pgm	Utility program to delete a file
delwork.pgm	Delete Working Directory
getdesc.pgm	Utility program to check IDA system environment variables
gethdr.pgm	Utility program to get header information from an IDA format emission inventory
gethdrx.pgm	Get header information for an IDA format emission inventory from user
gridview.pgm	Start IDA GIS Viewer
main.pgm	IDA Main Screen
makdirek.pgm	Create a new directory
makwork.pgm	Create a Working Directory
mvload.pgm	Import Motor Vehicle VMT Data
mvqavmt.pgm	QC Motor Vehicle VMT Data
mvwrit.pgm	Export Motor Vehicle VMT Data
outf_set.pgm	SCL link modules to get the name of an output file

ptgroup.pgm	Define Stack Splits - Assign Stack Groups
ptldcem.pgm	Import CEM Data
ptload.pgm	Import Point Source Data
ptqacem.pgm	QC CEM Data
ptqaemis.pgm	QC Point Source Emission Data
ptqaplnt.pgm	QC Point Source Plant Data
ptqaproc.pgm	QC Point Source Process Data
ptqaso4.pgm	QC Point Source Emission Data - Add SO4 Emission
ptqastak.pgm	QC Point Source Stack Data
ptqastck.pgm	QC Point Source Stack Data - Analyze Stack Data
ptqastco.pgm	QC Point Source Stack Data- Correct Stack Data
ptsplit.pgm	Define Stack Splits
ptsplitc.pgm	Define Stack Splits - Classify by Stack Properties
ptsplitr.pgm	Define Stack Splits - Generate Ranking Report
ptwrit.pgm	Export Point Source Data
sasfile.pgm	Utility to split a SAS file name into SAS components
sccarea.pgm	Generate SCC Reports - Assign a User Area File
sccareac.pgm	Generate SCC Reports - Create a User Area File
sccrpt.pgm	Generate SCC Reports
sccrpta.pgm	Generate SCC Reports - Generate Statistical Analysis
sccrptr.pgm	Generate SCC Reports - Generate Ranking Report
sccrpts.pgm	Generate SCC Reports - Generate Summary Report
sccsel.pgm	Generate SCC Reports - Enter Selection Criteria
scctier.pgm	Generate SCC Reports - Assign a User Tier File
scctierc.pgm	Generate SCC Reports - Create a User Tier File
setbin.pgm	Utility program to setup execution environment
setenv.pgm	Utility program to set an envrionment variable
setref.pgm	Utility program to set a SAS library/file reference and the corresponding environment variable
so4_set.pgm	SCL link modules to set the file name for the sulfate table
src_set.pgm	SCL link modulse to set an IDA working directory
start.pgm	Utility program to check IDA system environment variables
stk_dflt.pgm	Generate SCC Stack Defaults
uipopask.pgm	Popup prompt screen
uipopmsg.pgm	Pop-up message screen
wdir_set.pgm	SCL link modules to set an IDA working directory
workget.pgm	Utility program to get name of current working directory
workset.pgm	Assign a Working Directory

SAS/AF help screen layouts are shown in the following files:

addwork.hlp	Add data from Another Working Directory
areafile.hlp	Description of area file that maps states and counties to subregions for reporting
areain.hlp	Description of ASCII input file loaded when creating an area file
cemdate.hlp	Description of Selection Period for CEM File
cemfile.hlp	Description of CEM File
country.hlp	Description of working directory Country field
datayear.hlp	Description of working directory Data year field
desc.hlp	Description of working directory description field
direk.hlp	Description of a directory screen field
errlim.hlp	Description of error limit field for QC reports
filename.hlp	Description of a filename screen field
loadmid.hlp	Load data from an IDA format inventory
loadnet.hlp	Load data from a NET format inventory
mvfile.hlp	Description of a motor vehicle VMT file
outdirek.hlp	Description of an output directory name
sccsel.hlp	Description of logical expressions for selections in SCC reports
so4tabar.hlp	Description of area sulfate table
so4tabpt.hlp	Description of point sulfate table
srcmv.hlp	Description of Source Name/Directory field on Import MV VMT screen
srcname.hlp	Description of Source Name/Directory field
srcwdir.hlp	Description of Source Directory field
stkdflt.hlp	Description of stack default file
tierfile.hlp	Description of a user-defined tier file
tierin.hlp	Description of ASCII input file used to create a user-defined tier file
workdir.hlp	Description of a working directory

APPENDIX M

Chemical Mechanism and Process Analysis Details

Appendix M

Chemical Mechanism and Process Analysis Details

Table of Contents

Table M-1. Gas-phase Chemical Species for RADM2 Mechanism (plus CIS_1)

CHEMICAL SPECIES IN RADM2:

Stable Inorganic Compounds

Nitrogen	*Variable Name*	*Description*
NO_2	NO2	Nitrogen dioxide
NO	NO	Nitric oxide
HONO	HONO	Nitrous acid
NO_3	NO3	Nitrogen trioxide
N_2O_5	N2O5	Nitrogen pentoxide
HNO_4	HNO4	Pernitric acid
HNO_3	HNO3	Nitric acid
Oxidants		
O_3	O3	Ozone
H_2O_2	H2O2	Hydrogen peroxide
Sulfur		
SO_2	SO2	Sulfur dioxide
SULF	SULF	Sulfuric acid
Carbon Oxides		
CO	CO	Carbon monoxide
CO_2	CO2	Carbon dioxide (product only)
H_2	H2	Hydrogen (product only)

Inorganic Short-lived Intermediates

Atomic Species		
O^3P	O3P	Ground state oxygen atom
O^1D	O1D	Excited state oxygen atom
Odd Hydrogen		
HO	HO	Hydroxy radical
HO_2	HO2	Hydroperoxy radical

Abundant Stable Species

O_2	O2	Oxygen
N_2	N2	Nitrogen
H_2O	H2O	Water

Stable Organic Compounds

Alkanes		
CH_4	CH4	Methane
ETH	ETH	Ethane
HC3	HC3	Alkanes with HO rate constant (298, 1 atm) between 2.7 x $10^{-13}$and 3.4 x 10^{-12}
HC5	HC5	Alkanes with HO rate constant (298, 1 atm) between 3.4 x 10^{-12} and 6.8 x 10^{-12}
HC8	HC8	Alkanes with HO rate constant (298, 1 atm) greater than 6.8 x 10^{-12}
Alkenes		
OL2	OL2	Ethene
OLT	OLT	Terminal alkenes
OLI	OLI	Internal alkenes
ISO	ISO	Isoprene
TOL	TOL	Toluene and less reactive aromatics
CSL	CSL	Cresol and other hydroxy substituted aromatics
XYL	XYL	Xylene and more reactive aromatics
Carbonyls		
HCHO	HCHO	Formaldehyde
ALD	ALD	Acetaldehyde and higher aldehydes
KET	KET	Ketones
GLY	GLY	Glyoxal
MGLY	MGLY	Methylglyoxal
DCB	DCB	Unsaturated Dicarbonyl
Organic Nitrogen		
PAN	PAN	Peroxyacetyl nitrate and higher PANs
TPAN	TPAN	H(CO)CH=CHCO3NO2
ONIT	ONIT	Organic nitrate
Organic Peroxides		
OP1	OP1	Methyl hydrogen peroxide
OP2	OP2	Higher organic peroxides
PAA	PAA	Peroxyacetic acid
Organic Acids		
ORA1	ORA1	Formic acid
ORA2	ORA2	Acetic acid and higher acids

Organic Short-lived Intermediates

Peroxy Radicals from Alkanes		
MO_2	MO2	Methyl peroxy radical
ETHP	ETHP	Peroxy radical formed from alkane, ETH
HC3P	HC3P	Peroxy radical formed from alkane, HC3
HC5P	HC5P	Peroxy radical formed from alkane, HC5
HC8P	HC8P	Peroxy radical formed from alkane, HC8
Peroxy Radicals from Alkenes		
OL2P	OL2P	Peroxy radical formed from alkene, OL2
OLTP	OLTP	Peroxy radical formed from alkene, OLT
OLIP	OLIP	Peroxy radical formed from alkene, OLIP
Peroxy Radicals from Aromatics		
TOLP	TOLP	Peroxy radical formed from aromatic, TOL
XYLP	XYLP	Peroxy radical formed from aromatic, XYL
Peroxy Radicals with Carbonyl Groups		
ACO_3	ACO3	Acetylperoxy Radical
KETP	KETP	Peroxy radical formed from ketone, KET
TCO_3	TCO3	$H(CO)CH=CHCO_3$
Peroxy Radicals Involving Nitrogen		
OLN	OLN	NO3-alkene adduct
XNO_2	XNO2	Accounts for additional organic nitrate
XO_2	XO2	Accounts for additional NO to NO2 conversions affected by the lumped species

Additional species in RADM2_CIS1:

Peroxy Radicals from isoprene reactions:

ISO_RO2
ISON_RO2
IP_RO2

Lumped products from isoprene reactions:

ISOPROD

Species added to RADM2 for aerosols:

Counter species to track reaction rates:

SULAER
HC8AER
OLIAER
TOLAER
XYLAER
CSLAER
TERPAER

Monoterpenes:

TERP

Table M-2. Gas-phase Chemical Species for the CB-IV Mechanism (plus AQ)]

Chemical Species in the CBM-IV:

Stable Inorganic Compounds

Nitrogen

NO	NO	Nitric oxide
NO_2	NO2	Nitrogen dioxide
NO_3	NO3	Nitrogen trioxide
N_2O_5	N2O5	Dinitrogen pentoxide
HONO	HONO	Nitrous acid
HNO_3	HNO3	Nitric acid
PNA	PNA	Peroxynitric acid (HO_2NO_2)

Inorganic Short-lived Intermediate

Atomic Species

O1D	O1D	Oxygen atom (singlet)
O	O	Oxygen atom (triplet)

Odd Hydrogen

OH	OH	Hydroxyl radical
HO_2	HO2	Hydroperoxy radical

Abundant Stable Species

H_2O	H2O	Water

Oxidants

O_3	O3	Ozone
H_2O_2	H2O2	Hydrogen peroxide

Carbon Oxides

CO	CO	Carbon monoxide

Stable Organic Compounds

FORM	FORM	Formaldehyde ($CH_2=O$)
ALD_2	ALD2	High molecular weight aldehydes (RCHO, R>H)
C_2O_3	C2O3	Peroxyacyl radical ($CH_3C(O)OO\cdot$)
PAN	PAN	Peroxyacyl nitrate ($CH_3C(O)OONO_2$)
PAR	PAR	Paraffin carbon bond (C-C)
ROR	ROR	Secondary organic oxy radical
OLE	OLE	Olefinic carbon bond (C=C)
ETH	ETH	Ethene ($CH_2=CH_2$)
TOL	TOL	Toluene ($C_6H_5-CH_3$)

CRES	CRES	Cresol and higher molecular weight phenols
Stable Organic Compounds (continued)		
TO_2	TO2	Toluene-hydroxyl radical adduct
CRO	CRO	Methylphenoxy radical
OPEN	OPEN	High molecular weight aromatic oxidation ring fragment
XYL	XYL	Xylene (C_6H_4-$(CH_3)_2$)
MGLY	MGLY	Methylglyoxal ($CH_3C(O)C(O)H$)
ISOP	ISOP	Isoprene
XO_2	XO2	NO-to-NO_2 operation
XO_2N	XO2N	NO-to-nitrate operation

Species added to CB4 for aerosols:

Counter species to track reaction rates:

SULAER
TOLAER
XYLAER
CSLAER
TERPAER

Monoterpenes:

TERP

Species added to CB4 for aqueous chemistry:

Formic Acid	FACD
Acetic and higer Acids	AACD
Peroxy Acetic Acid	PACD
Upper limit of methylhydroperoxide	UMHP

Table M-3. Aerosol Species and Molecular weights

Description	Species Name	Molecular weight
Accumulation mode sulfate aerosol	ASO4J	96.0
Aitken mode sulfate aerosol	ASO4I	96.0
Accumulation mode ammonium aerosol	ANH4J	18.0
Aitken mode ammonium aerosol	ANH4I	18.0
Accumulation mode nitrate aerosol	ANO3J	62.0
Aitken mode nitrate aerosol	ANO3I	62.0
Accumulation mode anthropogenic secondary organic aerosol	AORGA	120.0
Aitken mode anthropogenic secondary organic aerosol	AORGAI	120.0
Accumulation mode primary organic aerosol	AORGPAJ	120.0
Aitken mode primary organic aerosol	AORGPAI	120.0
Accumulation mode biogenic secondary organic aerosol	AORGB	120.0
Aitken mode biogenic secondary organic aerosol	AORGBI	120.0
Accumulation mode elemental carbon	AECJ	12.0
Aitken mode elemental carbon	AECI	12.0
Accumulation mode unspecified anthropogenic aerosol	A25J	200.0
Aitken mode unspecified anthropogenic aerosol	A25I	200.0
Coarse mode unspecified anthropogenic aerosol	ACORS	100.0
Coarse mode marine aerosol	ASEAS	58.0
Coarse mode soil-derived aerosol	ASOIL	100.0
Aitken mode number concentration	NUMATKN	1.0
Accumulation mode number concentration	NUMACC	1.0
Coarse mode number concentration	NUMCOR	1.0
Accumulation mode water concentration	AH2O	18.0
Aitken mode water concentration	AH2OI	18.0

Table M-4. Names for RADM2 and CB-IV Emission Species

Description	RADM2 Emission Name	CB4 Emission Name
Nitrogen oxide	NO	NO
Nitrogen dioxide	NO2	NO2
Carbon monoxide	CO	CO
Sulfur dioxide	SO2	SO2
Sulfuric acid	SULF	SULF
Ammonia	NH3	NH3
Particulates < 2.5um	PM2_5	PM2_5
Particulates < 10um	PM10	PM10
Ethene	OL2	ETH
Formaldehyde	HCHO	FORM
Acetaldehyde	ALD	ALD2
Isoprene	ISO	ISOP
Toluene	TOL	TOL
Xylene	XYL	XYL
Biogenic terpenes	TERPB	TERPB
Ethane	ETH	
Slow reacting alkanes	HC3	
Medium reacting alkanes	HC5	
Fast reacting alkanes	HC8	
Terminal alkenes	OLT	
Internal alkenes	OLI	
Cresol	CSL	
Ketones	KET	
Acetic acid	ORA2	
Paraffins		PAR
Olefins		OLE

Note: Missing species name indicates species not defined for that mechanism.

Table M-5. Deposition Velocity Surrogate Names

Description	CMAQ Surrogate Name
Gas-phase species:	
Nitrogen oxide	VD_NO
Nitrogen dioxide	VD_NO2
Ozone	VD_O3
Nitric acid	VD_HNO3
Hydrogen peroxide	VD_H2O2
Sulfur dioxide	VD_SO2
Sulfuric acid	VD_SULF
Formaldehyde	VD_HCHO
Acetaldehyde	VD_ALD
Peroxyacetic acid	VD_PAA
Organic peroxides	VD_OP
Organic acids	VD_ORA
Aerosol species:	
Accumulation mode mass	VMASSJ
Aitken mode mass	VMASSI
Coarse mode mass	VMASSC
Accumulation mode number	VNUMACC
Aitken mode number	VNUMATKN
Coarse mode number	VNUMCOR
Non-reactive species:	
Ammonia	VD_NH3

Table M-6. Surrogate Names for Gas/Aerosol/Aqueous Chemistry/Non-reactive/Tracer Links

Description	Surrogate Name
Gas to Aerosol links (G2AE):	
Nitric acid	HNO3
Sulfuric acid	SULF
Terpenes	TERPSP
Alkane reaction rate counter species	ALKRXN
Olefin reaction rate counter species	OLIRXN
Toluene reaction rate counter species	TOLRXN
Xylene reaction rate counter species	XYLRXN
Cresol reaction rate counter species	CSLRXN
Terpene reaction rate counter species	TERPRXN
Gas to Aqueous Chemistry links (G2AQ):	
Nitric acid	HNO3
Sulfuric acid	H2SO4
Sulfur dioxide	SO2
Nitrogen pentoxide	N2O5
Hydrogen peroxide	H2O2
Ozone	O3
Formic acid	FOA
Peroxy acetic acid	PAA
Methyl hydrogen peroxide	MHP
Aerosol to Aqueous Chemistry links (A2AQ):	
Aitken mode sulfate aerosol	SO4_AITKEN
Accumulation mode sulfate aerosol	SO4_ACCUM
Aitken mode ammonium aerosol	NH4_AITKEN
Accumulation mode ammonium aerosol	NH4_ACCUM
Aitken mode nitrate aerosol	NO3_AITKEN
Accumulation mode nitrate aerosol	NO3_ACCUM
Coarse mode nitrate aerosol	NO3_COARSE
Aitken mode organic aerosol	ORG_AITKEN
Accumulation mode organic aerosol	ORG_ACCUM
Aitken mode primary aerosol	PRI_AITKEN
Accumulation mode primary aerosol	PRI_ACCUM
Coarse mode primary aerosol	PRI_COARSE
Calcium carbonate	CACO3
Magnesium carbonate	MGCO3

Table M-6. Surrogate Names for Gas/Aerosol/Aqueous Chemistry/Non-reactive/Tracer Links (continued)

Description	Surrogate Name
Sodium chloride	NACL
Iron	A3FE
Manganese	B2MN
Potassium chloride	KCL
Aitken mode number concentration	NUM_AITKEN
Accumulation mode number concentration	NUM_ACCUM
Coarse mode number concentration	NUM_COARSE
Non-reactive to aerosol links (N2AE):	
Ammonia	NH3
Non-reactive to Aqueous chemistry links (N2AQ):	
Ammonia	NH3
Carbon dioxide	CO2

Table M-7. Surrogate Names for Cloud Scavenging (AQ)

Chemical Compound	Surrogate Name
Gas-phase species:	
Ozone	O3
Hydroxy radical	HO2
Hydrogen peroxide	H2O2
Nitric oxide	NO
Nitrogen dioxide	NO2
Nitrate radical	NO3
Nitrogen pentoxide	N2O5
Nitrous acid	HNO2
Nitric acid	HNO3
Pernitric acid	HNO4
Sulfur dioxide	SO2
Sulfuric acid	H2SO4
Methane	METHANE
Ethane	ETHANE
Propane	PROPANE
Butane	BUTANE
Pentane	PENTANE
Hexane	HEXANE
Octane	OCTANE
Nonage	NONAGE
Decade	DECADE
Ethene	ETHENE
Propene	PROPENE
Isoprene	ISOPRENE
Acetylene	ACETYLENE
Benzene	BENZENE
Toluene	TOLUENE
O-xylene	O-XYLENE
Methanol	METHANOL
Ethanol	ETHANOL
2-cresol	2-CRESOL
4-cresol	4-CRESOL
Methyl hydrogen peroxide	METHYLHYDROPEROX
Formaldehyde	FORMALDEHYDE
Acetaldehyde	ACETALDEHYDE
Generic aldehyde	GENERIC_ALDEHYDE

Table M-7. Surrogate Names for Cloud Scavenging (AQ) (continued)

Glyoxal	GLYOXAL
Acetone	ACETONE
Formic acid	FORMIC_ACID
Acetic acid	ACETIC_ACID
Methyl glyoxal	METHYL_GLYOXAL
Carbon monoxide	CO
Peroxyacetyl nitrate and higher PANs	PAN
Peroxyacetyl nitrate compound produced from methacrolein	MPAN
Hydroxy radical	OH
Methyl-peroxy radical	METHYLPEROXY_RAD
Peroxy acetic acid	PEROXYACETIC_ACI
Aerosol species:	
Accumulation mode sulfate aerosol	SO4_ACCUM
Aitken mode sulfate aerosol	SO4_AITKEN
Accumulation mode ammonium aerosol	NH4_ACCUM
Aitken mode ammonium aerosol	NH4_AITKEN
Accumulation mode nitrate aerosol	NO3_ACCUM
Aitken mode nitrate aerosol	NO3_AITKEN
Accumulation mode anthropogenic secondary organic aerosol	ORGA_ACCUM
Aitken mode anthropogenic secondary organic aerosol	ORGA_AITKEN
Accumulation mode primary organic aerosol	ORGPA_ACCUM
Aitken mode primary organic aerosol	ORGPA_AITKEN
Accumulation mode biogenic secondary organic aerosol	ORGB_ACCUM
Aitken mode biogenic secondary organic aerosol	ORGB_AITKEN
Accumulation mode elemental carbon	EC_ACCUM
Aitken mode elemental carbon	EC_AITKEN
Accumulation mode unspecified anthropogenic aerosol	A25_ACCUM
Aitken mode unspecified anthropogenic aerosol	A25_AITKEN
Coarse mode unspecified anthropogenic aerosol	CORS
Coarse mode marine aerosol	SEAS
Coarse mode soil-derived aerosol	SOIL

Table M-7. Surrogate Names for Cloud Scavenging (AQ) (continued)

Aitken mode number concentration	NUM_AITKEN
Accumulation mode number concentration	NUM_ACCUM
Coarse mode number concentration	NUM_COARSE
Accumulation mode water concentration	H2O_ACCUM
Aitken mode water concentration	H2O_AITKEN
Non-reactive species:	
Ammonia	NH3
Carbon dioxide	CO2

Gas-phase Chemical Reactions Input Format

Introduction. The Models-3/CMAQ system employs a generalized chemical mechanism processor (MP) for gas-phase chemical mechanisms. The generalized processor greatly simplifies the task of altering chemical mechanisms and provides the capability of easily using different mechanisms in the CMAQ system. This appendix describes the format that must be used for entering a gas-phase chemical mechanism in the CMAQ system. As described in Section 7.5.2, the mechanism can be entered directly in the Models-3 framework or can be imported into the system via an ASCII text file that contains the reaction data. Since gas-phase chemical mechanisms are usually lengthy, the latter approach is recommended. The ASCII file can be created with any editor or word processor that generates a standard ASCII file.

The gas-phase mechanism data are formatted according to a simple set of rules and a free-form format that are similar to the approaches used by Jeffries et al. (1990) and Gery and Crouse (1990). For reference, example listings of two chemical mechanism input files are included at the end of the Appendix. The mechanism input data consist of four major parts: an optional mechanism name, a Reactions keyword that can be used to specify the units of the rate coefficients, a series of reaction and rate constant lines, and an end of reactions list keyword. Each of these will be discussed separately below. The discussion begins first, however, with a description of some general rules for entering the mechanism data.

General rules. The free form format generally ignores white spaces and allows line wrap around (i.e., entries can be continued on a subsequent line after a hard return). The free form format also allows embedded comments and makes use of special symbols to indicate the type of input data. Special rules for naming species, entering labels, and specifying numerical values such as stoichiometric coefficients and rate constant parameters are also used. Each are discussed below.

> *Comments.* All lines that have an exclamation point in column 1 are treated as comment lines and are ignored by the MP. Any text

enclosed in braces or parentheses is also treated as comment and ignored by the MP.

Species names. The MP recognizes two types of species names - predefined and user defined. The predefined names include M (any molecule in the atmosphere), O2 (oxygen), N2 (nitrogen), and H2O (water vapor). These particular species can be referenced anywhere in the mechanism using the short names but should not be used to represent any other species in the mechanism. The following special rules have also been established for user-defined species names.

a) The input names must not contain any blanks and can be up to 81 characters long, but only the first 16 characters are significant.

b) The name must begin with an alphabetic character, but may contain any alphanumeric character (i.e., "A-Z", "a-z"., and "0-9") or the characters ":" and "_" after the first position.

c) The name is case sensitive. Thus, NO2 and no2 would represent two different species.

d) A name can have embedded comments, but cannot span two lines.

Label names. Special rules also apply to label names that are used in the chemical reaction definitions.

a) A label is defined as any string starting with the character "<" and ending with ">".

b) A label can be up to 81 characters long, but only the first 16 non-blank characters are used.

c) A label may contain any character except a single comment or a label delimiter. Embedded blanks in the label name are discarded.

d) A label can span lines and can contain embedded comments.

Numbers. Numerical inputs can be either integer (e.g., 5), floating point (e.g., 5.0), or exponential (e.g., 5.0E+00). With the exponential format, the "E" may be either upper or lower case and a positive exponent will be assumed if the sign of the exponent is missing.

Mechanism Name. The mechanism name is an optional input. If it is included, it must be the first non-comment entry in the mechanism. The name entered in the input file is *not* the name that will be used in the Models-3 framework however. The name assigned here will be output in a comment line in the mechanism include files that are generated by the MP, but it is not used by any CMAQ processor or model. Rules for the mechanism name are the same as those for species names, except that only the first eight characters are used. Although no delimiter is required at the end of the name, a "hard return" after the entry is suggested for clarity in reading the input file.

Reaction keyword. The list of reactions must be preceded by the symbolic keyword REACTIONS, but only the first four characters (i.e., REAC) are actually required. If no mechanism name is supplied, it would be the first non-comment entry. The reaction key word is followed by a units keyword, enclosed in brackets, to indicate the units of the input rate constants that are to follow. The units options are "PP" and "CM", which stand for ppm-min units and molecule-cc-sec units, respectively. (Note that longer entries could be used, but only the first two characters are significant. Also, the first two characters can either be upper or lower case provided they are both of the same case and these entries are not-case sensitive). Finally, the units option must be followed by an equal sign. A delimiter is not required after the equal sign, but a "hard return" after the entry is suggested for clarity in reading the input file. Examples of valid inputs include the following:

REAC[PP]=
REACTIONS[CM]=
REACTIONS[ppm]=
REAC[cms]=

Reaction Definitions. The reaction keyword is followed by a cluster of "reaction lines". The reaction lines consist of the following: 1) an optional label, 2) up to three reactants 3) an equal sign to separate reactants from products, 4) up to 12 products with optional numerical coefficients, 5) a reaction rate constant in one of the prescribed formats that will be described below, and 6) an ending semicolon. One reaction line is entered for each chemical reaction in the mechanism. Since line wrapping is allowed, a single reaction definition can span multiple lines in the input file. The generic format for the reaction portion of the reaction line follows, where the quantities inside brackets are optional, $reac_n$ represents a reactant species name, $prod_n$ a product species name, and p_n a stoichiometric coefficient.

[<label>] $[reac_1[+reac_2[+reac_3]]] = [\pm[p_1*]prod_1\ [\pm[p_2*]prod_2\ [... \pm\ [p_{12}*]prod_{12}]]]$

Each of the components of the reaction are described below:

- *Labels.* Reaction labels are optional, but highly recommended. They are used to reference specific reactions in some reaction rate forms, and may be used by other CMAQ processors such as the Process Analysis Control Program. As indicated above, they are enclosed by "<>", and any embedded blanks included in the label are omitted from the label.

- *Reactants.* The reactants consist of up to three predefined or user defined species names separated by plus signs. Stoichiometric coefficients are not allowed, and are always assumed to equal 1.0. Note that if a pre-defined species name is used as a reactant, the concentration of that species will be used in determining the rate constant. Since the number of reactants may range from zero to three, reactions from zero order to third order can be included.

- *Products.* The products also consist of a series of predefined or user-defined species names separated by plus or minus signs and optional numerical coefficients. As noted above, up to 12 products may be included and the stoichiometric coefficients may be either real, integer, or E-format. They must be separated from the species names by an asterisk.

The final part of each reaction line contains the rate coefficient parameters for the reaction. Rate constant parameters are delimited from the reaction information by either a # sign or a delimiter of the form "%n #", where n is a number (currently 1, 2, or 3). There are currently 10 rate coefficient forms that can be used, and these are defined in Table M-8. Special notes about the forms defined in Table M-8 follow.

For photolysis rate reactions, the expressions <J_Label> refers to a label used to identify photolysis rates produced by the CMAQ photolysis rate pre-processor. This processor has been set up to produce photolysis rates for various species in the chemical mechanism. See Table M-9 for a listing of the photolysis rates that can be generated and the valid label names that are assigned to each. If one is not available for a particular species of interest, it is still possible to develop a surrogate rate by selecting an appropriate value of A to scale the available rate to approximate the desired rate.

The expression <k_Label> in photolytic reaction type 2 and thermal reaction types 5 and 6 refers to the labels of other reactions in the mechanism. The term k_n in the corresponding equation refers to the rate constant for the reaction with that label. Thus, for photolytic type 2 and thermal type 6, the effect of using one of these forms is to scale the rate for the reaction to that of another. Thermal reaction form 5 is a reverse equilibrium reaction form, where A@C is the equilibrium constant and the expression <k_Label> refers to the rate constant for the forward reaction. Note that in thermal reaction type 5 and 6, the symbols E and K must be uppercase.

Finally, P in the equation for thermal reaction type 7 is pressure. Also, the default values for the parameters F and N in thermal reaction rate type 10 (i.e., Falloff expression 2) are 0.6 and 1.0, respectively. Values need not be specified for these parameters if the defaults are being used.

END keyword. The last entry in the file is the keyword to indicate the end of the mechanism. The first three characters of that keyword must be "END" or "end". Because these characters are used to indicate the end of a mechanism, the first reactant in any reaction cannot begin with these same characters unless it is preceded by a reaction label.

REFERENCES

Gery, M.W. and Crouse, R.R. (1990). User's Guide for Executing OZIPR. EPA/600/8-90, U.S. Environmental Protection Agency, Research Triangle Park, NC 27711.

Jeffries, H.E. (1990). User's Guide to Photochemical Kinetics Simulation System PC-PKSS Software Version 3, Chapel Hill, NC 27514.

Table M-8: Rate Constant Forms

Photolytic Reactions:

Type	Input form	Equation	Example
1	A / <J_label>	$k = A\ J$	# 1.0 / <NO2_CBIV88>;
2	A / <k_label>	$k = A\ k_n$	# 2.0 / < P1>;

Thermal reactions:

Type	Input Form	Equation	Example
1	A	$k = A$	# 9.3E-12;
2	A^B	$k = A\ (T/300)^B$	# 1.2E-12 ^ -1.3;
3	A@E	$k = A\ e^{E/T}$	# 1.70E-11 @ 1400.0;
4	A^B@E	$k = A\ (T/300)^B\ e^{E/T}$	# 6.165E-13 ^ 2 @ 444.0;
5	A@C*E<k_label>	$k = k_n / (A\ e^{C/T})$	# 2.1E-27 @ -10900.0 * E< 10>;
6	A*K<k_label>	$k = A\ k_n$	# 2.0E-02 * K < 100>;
7	A	$k = A\ (\ 1.0 + 0.6\ P)$	%1 # 1.50E-13;
8	A0@E0&A2&E2@A3@E3	Falloff Expression 1	%2 # 7.2E-15 @ -785 & 4.1E-16 @ -1440 & 1.9E-33 @ -980.0;
9	A1@E1&A2@E2	Special Expression 1	%3 # 2.2E-13 @ -620.0 & 1.9E-33 @ -980.0;
10	A0^B0@E0&A1^B1@E1&F^N Falloff	Expression 2	# 1.8E-31 ^ -3.2 & 4.7E-12 ^ -1.4 & 0.6 ^ 1.0;

Falloff Expression 1:

$$k_0 = A_0\, e^{E_0/T} \qquad k_2 = A_2\, e^{E_2/T} \qquad k_3 = A_3\, e^{E_3/T}$$

$$k = k_0 + k_3[M]/(1 + k_3[M]/k_2)$$

Special Expression 1

$$k_1 = A_1\, e^{E_1/T} \qquad k_2 = A_2\, e^{E_2/T}$$

$$k = k_1 + k_2[M]$$

Falloff Expression 2:

$$k_0 = A_0\, e^{E_0/T} \qquad k_1 = A_1\, e^{E_1/T}$$

$$k = \{k_0[M]/(1 + k_0[M]/k_1)\}\ F^{\{1+[\log_{10}(k_0[M]/k_1)]^2\}^{-1}}$$

Table M-9. Photolysis Reaction Mapping to Chemical Mechanisms

Photolysis Reaction	Description	RADM2 Mechanism Label	CB-IV Mechanism Label	SAPRC Mechanism Label	Recent NASA Published Data Label
O2 + hv -> O + O	Molecular Oxygen photolysis	O2_RADM88			O2_NASA94
O3 + hv -> O2 + O(1D)	Ozone Photolysis to O1D	O3O1D_RADM88	O3O1D_CBIV88	O3O1D_SAPRC91	O3O1D_NASA94
O3 + hv > O2 + O(3P)	Ozone Photolysis to O3P	O3O3P_RADM88		O3O3P_SAPRC91	O3O3P_NASA94
NO2 + hv > NO + O	Nitrogen Dioxide Photolysis	NO2_RADM88	NO2_CBIV88	NO2_SAPRC91	NO2_NASA94
NO3 + hv > NO + O2	Nitrate Photolysis to NO	NO3NO_RADM88		NO3NO_SAPRC91	NO3NO_NASA94
NO3 + hv > NO2 + O(3P)	Nitrate Photolysis to NO2	NO3NO2_RADM88		NO3NO2_SAPRC91	NO3NO2_NASA94
HONO + hv > OH + NO	Nitrous Acid Photolysis	HONO_RADM88		HONO_SAPRC91	HONO_NASA94
HNO3 + hv > OH + NO2	Nitric Acid Photolysis	HNO3_RADM88		HNO3_SAPRC91	HNO3_NASA94
HNO4 + hv > HO2 + NO2	Pernitric Acid Photolysis	HNO4_RADM88			HNO4_NASA94
H2O2 + hv > OH + OH	Hydrogen Peroxide Photolysis	H2O2_RADM88		H2O2_SAPRC91	H2O2_NASA94
HCHO + hv > H + HCO	Formaldehyde Photolysis to Radicals	HCHOrad_RADM88	HCHOrad_CBIV88	HCHOrad_SAPRC91	HCHOrad_NASA94
HCHO + hv > H2 + CO	Formaldehyde Photolysis to Molecular Hydrogen	HCHOmol_RADM88	HCHOmol_CBIV88	HCHOmol_SAPRC91	HCHOmol_NASA94

CH3CHO + hv (+2O2) > CH3OO + HO2 + CO	Acetaldehyde Photolysis	ALD_RADM88	ALD_CBIV88	ALD_SAPRC91	
CH3COCH3 + hv > CH3 + CH3CO	Acetone Photolysis	ACETONE_RADM88		ACETONE ???	
CH3COC2H5 + hv > ACO3 + ETH	Methyl Ethyl Ketone Photolysis	KETONE_RADM88		KETONE_SAPRC91	
HCOCHO + hv > HCHO + CO	Glyoxal Photolysis to Formaldehyde	GLYform_RADM88			
CH3COCHO + hv > ACO3 + HO2 + CO	Methyl Glyoxal Photolysis	MGLY_RADM88		MGLY_SAPRC91	
HCOCH=CHCHO + hv > 0.98HO2 + TCO3 + 0.02ACO3	Unsaturated Dicarbonyl Photolysis	UDC_RADM88			
CH3OOH + hv > products	Methyl Hydrogen Peroxide Photolysis	MHP_RADM88		MHP_SAPRC91	MHP_NASA94
CH3ONO2 + hv > 0.2ALD + 0.8KET + HO2 + NO2	Organic Nitrate Photolysis	ORGNIT_RADM88			
HCOCHO + hv > 2CO + H2	Glyoxal Photolysis to Molecular Hydrogen	GLYmol_RADM88			
RCHO + hv > CCHO + RO2 R + RO2 + CO + HO2	Propionaldehyde Photolysis			PROPA_SAPRC91	
AFG2 + hv > HO2 + CO + CCO O2 + RCO3	Unknown Aromatic Ring Fragment Photolysis			UARF_SAPRC91	
C3H4O + hv -> products	Acrolein Photolysis	ACROLEIN	ACROLEIN		

Example M-1: CB4 Mechanism Listing

```
! Created by Jerry Gipson,  May, 1997
! The reaction labels correspond to those used in "A Photochemical Kinetics
! Mechanism for Urban and Regional Scale Computer Modeling" M. Gery et al.
! (JGR, 9/89) and to the numbers assigned to reactions in the UAM-V model
! for new reactions.  Reactions of methanol and ethanol are optional
! in UAM, and thus are commented out in this version. They need to be
! "un-commented" if they are to be included. See Volume 6 for a description
! of the changes that have been made to the original version.

! Modified by Jerry Gipson 3/98 to update isoprene chemistry to Carter's
! one product form, to change CO rate constant to the pressure dependent
! form, and to to correct miscellaneous typos.

! Modified by Jerry Gipson 3/98 to track all stable nitrogen products as
!      AACD  - Acetic and higher acids
!      PACD  - Peroxy acetic acid
!      UMHP  - Upper limit estimate of methylhydroperoxide


  CB4.2

REACTIONS[CM] =

<  R1>  NO2          =  NO +  O                    # 1.0 /<NO2_CBIV88>;
<  R2>  O  + O2      =  O3                         # 6.0E-34 ^ -2.3 & 2.8E-12;
<  R3>  O3 + NO      =  NO2                        # 1.8E-12 @ 1370;
<  R4>  O  + NO2     =  NO                         # 9.3E-12;
<  R5>  O  + NO2     =  NO3                        # 9.0E-32 ^ -2.0 & 2.2E-11;
<  R6>  O  + NO      =  NO2                        # 9.0E-32 ^ -1.5 & 3.0E-11;
<  R7>  O3 + NO2     =  NO3                        # 1.2E-13 @ 2450;
<  R8>  O3           =  O                          # 0.053 / <NO2_CBIV88>;
<  R9>  O3           =  O1D                        # 1.0 / <O3O1D_CBIV88>;
<R10a>  O1D + N2     =  O                          # 1.8E-11 @ -107;
<R10b>  O1D + O2     =  O                          # 3.2E-11 @ -67;
< R11>  O1D + H2O    =  2.0*OH                     # 2.2E-10;
< R12>  O3  + OH     =  HO2                        # 1.6E-12 @ 940;
< R13>  O3  + HO2    =  OH                         # 1.4E-14 @ 580;
< R14>  NO3          = 0.89*NO2 + 0.89*O +  0.11*NO # 33.9 / <NO2_CBIV88>;
< R15>  NO3 + NO     =  2.0*NO2                    # 1.3E-11 @ -250;
< R16>  NO3 + NO2    =  NO + NO2                   # 2.5E-14 @ 1230;

< R17>  NO3 + NO2    =      N2O5                   # 2.2E-30 ^ -4.3 &
                                                     1.5E-12 ^ -0.5;

< R18>  N2O5 + H2O   =   2.0*HNO3                  # 1.3E-21;
< R19>  N2O5         =   NO3  +  NO2               # 3.5E+14 @ 10897;
< R20>  NO   + NO  + O2  =   2.0*NO2               # 3.3E-39 @ -530;
< R21>  NO   + NO2 + H2O =   2.000*HONO            # 4.4E-40;

< R22>  OH   + NO    =   HONO                      # 6.7E-31 ^ -3.3 &
                                                     3.0E-11 ^  1.0;

< R23>  HONO         =   OH    +      NO           # 0.1975 / <NO2_CBIV88>;
< R24>  HONO + OH    =   NO2                       # 6.6E-12;
< R25>  HONO + HONO  =   NO    +      NO2          # 1.0E-20;
< R26>  OH   + NO2   =   HNO3                      # 2.6E-30 ^  3.2 &
                                                     2.4E-11 ^ -1.3;
```

```
< R27>  OH    + HNO3  =   NO3                          %2 # 7.2E-15 @ -785  &
                                                         4.1E-16 @ -1440 &
                                                         1.9E-33 @ -725;

< R28>  HO2   + NO    =   OH      +     NO2              # 3.7E-12 @ -240;

< R29>  HO2   + NO2   =   PNA                            # 2.3E-31 ^ -4.6 &
                                                         4.2E-12 ^ 0.2;

< R30>  PNA           =   HO2       +   NO2              # 4.8E+13 @ 10121;
< R31>  PNA + OH      =   NO2                            # 1.3E-12 @ -380;
< R32>  HO2 + HO2     =   H2O2                           # 5.9E-14 @ -1150;
< R33>  HO2 + HO2 + H2O  =   H2O2                        # 2.2E-38 @ -5800;
< R34>  H2O2          =  2*OH                            # 0.255 / <HCHOmol_CBIV88>;
< R35>  H2O2 + OH     =  HO2                             # 3.1E-12 @ 187;
< R36>  CO    + OH    =  HO2                         %1  # 1.5E-13;
< R37>  FORM + OH     =  HO2    +  CO                    # 1.0E-11;
< R38>  FORM          =  2*HO2 +  CO                     # 1.0 / <HCHOrad_CBIV88>;
< R39>  FORM          =  CO                              # 1.0 / <HCHOmol_CBIV88>;
< R40>  FORM + O      =  OH     +  HO2 + CO              # 3.0E-11 @ 1550;
< R41>  FORM + NO3    =  HNO3   +  HO2 + CO              # 6.3E-16;
< R42>  ALD2 + O      =  C2O3   +  OH                    # 1.2E-11 @ 986;
< R43>  ALD2 + OH     =  C2O3                            # 7.0E-12 @ -250;
< R44>  ALD2 + NO3    =  C2O3   +  HNO3                  # 2.5E-15;
< R45>  ALD2          =  XO2 + 2*HO2 + CO + FORM         # 1.0 / <ALD_CBIV88>;
< R46>  C2O3 + NO     =  NO2 + XO2    + FORM + HO2       # 3.49E-11 @ 180;
< R47>  C2O3 + NO2    =  PAN                             # 2.63E-12 @ -380;
< R48>  PAN           =  C2O3   + NO2                    # 2.00E+16 @ 13500;
< R49>  C2O3 + C2O3   =  2*XO2 + 2*FORM + 2*HO2          # 2.5E-12;
< R50>  C2O3 + HO2    =  0.79*FORM + 0.79*XO2 +
                         0.79*HO2  + 0.79*OH +
                         0.21*PACD                       # 6.5E-12;

< R51>  {CH4 +} OH    =  XO2  +  FORM  +  HO2            # 1.1E+02 @ 1710;

< R52>  PAR   + OH    =  0.87*XO2 + 0.13*XO2N +
                         0.11*HO2 + 0.11*ALD2 +
                         0.76*ROR - 0.11*PAR             # 8.1E-13;

< R53>  ROR           =  1.1*ALD2 + 0.96*XO2 +
                         0.94*HO2 - 2.10*PAR +
                         0.04*XO2N + 0.02*ROR            # 1.0E+15 @ 8000;

< R54>  ROR           =   HO2                            # 1.6E+03;
< R55>  ROR  + NO2    =   NTR                            # 1.5E-11;

< R56>  OLE + O       =  0.63*ALD2 + 0.38*HO2 +
                         0.28*XO2  + 0.3*CO  +
                         0.2*FORM  + 0.02*XO2N +
                         0.22*PAR +  0.2*OH              # 1.2E-11 @ 324;

< R57>  OLE + OH      =  FORM + ALD2 + XO2 +
                         HO2 - PAR                       # 5.2E-12 @ -504;

< R58>  OLE + O3      =  0.5*ALD2 + 0.74*FORM +
                         0.33*CO + 0.44*HO2 +
                         0.22*XO2 + 0.1*OH +
                         0.20*FACD + 0.20*AACD -
                         PAR                             # 1.4E-14 @ 2105;

< R59>  OLE + NO3     = 0.91*XO2 + 0.09*XO2N +
                        FORM + ALD2 - PAR + NO2          # 7.7E-15;
```

```
< R60>  ETH + O      = FORM + 0.7*XO2 + CO +
                       1.7*HO2  + 0.3*OH          # 1.0E-11 @ 792;

< R61>  ETH + OH     =  XO2 + 1.56*FORM + HO2 +
                        0.22*ALD2                 # 2.0E-12 @ -411;

< R62>  ETH + O3     =  FORM + 0.42*CO + 0.12*HO2 +
                        0.40*FACD                 # 1.3E-14 @ 2633;

< R63>  TOL + OH     = 0.08*XO2 + 0.36*CRES +
                       0.44*HO2 + 0.56*TO2  +
                       TOLAER                     # 2.1E-12 @ -322;

< R64>  TO2 + NO     = 0.9*NO2 + 0.9*HO2 +
                       0.9*OPEN +0.1*NTR          # 8.1E-12;

< R65>  TO2          = CRES  +  HO2                # 4.2;

< R66>  CRES + OH    = 0.4*CRO + 0.6*XO2 +
                       0.6*HO2 + 0.3*OPEN +
                       CSLAER                     # 4.1E-11;

< R67>  CRES + NO3   = CRO   +  HNO3 + CSLAER      # 2.2E-11;

< R68>  CRO  + NO2   = NTR                         # 1.4E-11;

< R69>  XYL  + OH    = 0.7*HO2 + 0.5*XO2 +
                       0.2*CRES + 0.8*MGLY +
                       1.1*PAR + 0.3*TO2 +
                       XYLAER                     # 1.7E-11 @ -116;

< R70>  OPEN + OH    = XO2 + 2*CO + 2*HO2 +
                       C2O3 + FORM                # 3.0E-11;

< R71>  OPEN         = C2O3  +  HO2  + CO          # 9.04 /<HCHOrad_CBIV88>;

< R72>  OPEN + O3    = 0.03*ALD2 + 0.62*C2O3 +
                       0.7*FORM + 0.03*XO2 +
                       0.69*CO + 0.08*OH +
                       0 76*HO2 + 0.2*MGLY        # 5.4E-17 @ 500;

< R73>  MGLY + OH    = XO2  +  C2O3                # 1.7E-11;

< R74>  MGLY         = C2O3  + HO2  + CO           # 9.64 / <HCHOrad_CBIV88>;

< R75>  ISOP + O     = 0 75*ISPD + 0.50*FORM +
                       0.25*XO2  + 0.25*HO2 +
                       0.25*C2O3 + 0.25*PAR       # 3.6E-11;

< R76>  ISOP + OH    = 0.912*ISPD + 0.629*FORM +
                       0.991*XO2  + 0.912*HO2  +
                       0.088*XO2N                 # 2.54E-11 @ -407.6;

< R77>  ISOP + O3    = 0.65*ISPD + 0.60*FORM +
                       0.20*XO2  + 0.066*HO2 +
                       0.266*OH  + 0.20*C2O3 +
                       0.15*ALD2 + 0.35*PAR  +
                       0.066*CO                   # 7.86E-15 @ 1912;

< R78>  ISOP + NO3   = 0.20*ISPD + 0.80*NTR  +
                       1.0*XO2   + 0.80*HO2  +
                       0.20*NO2  + 0.80*ALD2 +
```

```
                      2.4*PAR                          # 3.03E-12 @ 448;

< R79>  XO2  + NO     =   NO2                          # 8.1E-12;
< R80>  XO2  + XO2    =                                # 1.7E-14 @ -1300;
< R81>  XO2N + NO     = NTR                            # 8.12E-12;
< R82>  SO2  + OH     = SULF + HO2 + SULAER            # 4.39E-13 @ -160;
< R83>  SO2           = SULF + SULAER                  # 1.36E-06;
!< R84>  MEOH + OH    =                                # 1.08E-12;
!< R85>  ETOH + OH    =                                # 1.61E-12 @ -176;
< R86>  XO2  + HO2    =  UMHP                          # 7.67E-14 @ -1300.0;
< R87>  XO2N + HO2    =                                # 7.67E-14 @ -1300.0;
< R88>  XO2N + XO2N   =                                # 1.73E-14 @ -1300.0;
< R89>  XO2N + XO2    =                                # 3.45E-14 @ -1300.0;

< R90>  ISPD + OH     = 1.565*PAR  + 0.167*FORM +
                        0.713*XO2  + 0.503*HO2  +
                        0.334*CO   + 0.168*MGLY +
                        0.273*ALD2 + 0.498*C2O3        # 3.36E-11;

< R91>  ISPD + O3     = 0.114*C2O3 + 0.150*FORM +
                        0.850*MGLY + 0.154*HO2  +
                        0.268*OH   + 0.064*XO2  +
                        0.020*ALD2 + 0.360*PAR  +
                        0.225*CO                       # 7.11E-18;

< R92>  ISPD + NO3    = 0.357*ALD2 + 0.282*FORM +
                        1.282*PAR  + 0.925*HO2  +
                        0.643*CO   + 0.850*NTR  +
                        0.075*C2O3 + 0.075*XO2  +
                        0.075*HNO3                     # 1.00E-15;

< R93>  ISPD          = 0.333*CO   + 0.067*ALD2 +
                        0.900*FORM + 0.832*PAR  +
                        1.033*HO2  + 0.700*XO2  +
                        0.967*C2O3                     # 0.0036 / <ACROLEIN>;

< R94>  ISOP + NO2    = 0.20*ISPD + 0.80*NTR  +
                        1.00*XO2  + 0.80*HO2  +
                        0.20*NO   + 0.80*ALD2 +
                        2.4*PAR                        # 1.49E-19;


< AE1> TERP + OH      = TERPAER + OH                   # 1.07E-11  @ -549.0;
< AE1> TERP + NO3     = TERPAER + NO3                  # 3.23E-11  @ 975.0;
  AE1> TERP + O3      = TERPAER + O3                   # 7.29E-15  @ 1136.0;


END MECH
```

Example M-2: RADM2 Mechanism Listing

```
! RADM2 Mechanism originally based on OZIPR version and modified
! to be consistent with the RADM2 mechanism embedded in
! version h26.f of HR-RADM.  Reaction labels below correspond
! to the reaction numbers in the HR-RADM model. (Note: Reaction 40
! in the original RADM mechanism is represented by reactions 40 and
! 40a below.) Jeff & Jerry 23 Aug 96

! Modified by Jerry Gipson 3/12/97 to include gaseous species that
! are necessary to link gas-phase cemistry to aerosol formation.
! New species added:
!    SULAER:  Gives amount of sulfuric acid produced
!    TOLAER:  Gives total amount of TOL reacted
!    XYLAER:  Gives total amount of XYL reacted
!    CSLAER:  Gives total amount of CSL reacted
!    HC8AER:  Gives total amount of HC8 reacted
!    OLIAER:  Gives total amount of OLI reacted
!    TERP:    Monoterpenes
!    TERPAER: Gives total amount of monoterpenes reacted
! Three new reactions were added: AE1 AE2 and AE3

RADM2

REACTIONS [cm] =
< P1> NO2          =      O3P    +    NO              # 1.0  /<NO2_RADM88>;
< P2> O3           =      O1D                         # 1.0  /<O3O1D_RADM88>;
< P3> O3           =      O3P                         # 1.0  /<O3O3P_RADM88>;
< P4> HONO         =      HO     +    NO              # 1.0  /<HONO_RADM88>;
< P5> HNO3         =      HO     +    NO2             # 1.0  /<HNO3_RADM88>;
< P6> HNO4         =      HO2    +    NO2             # 1.0  /<HNO4_RADM88>;
< P7> NO3          =      NO                          # 1.0  /<NO3NO_RADM88>;
< P8> NO3          =      NO2    +    O3P             # 1.0  /<NO3NO2_RADM88>;
< P9> H2O2         = 2.0* HO                          # 1.0  /<H2O2_RADM88>;
<P10> HCHO         =      CO                          # 1.0  /<HCHOmol_RADM88>;
<P11> HCHO         =      HO2    +    HO2    +   CO   # 1.0  /<HCHOrad_RADM88>;
<P12> ALD          =      MO2    +    HO2    +   CO   # 1.0  /<ALD_RADM88>;
<P13> OP1          =      HCHO   +    HO2    +   HO   # 1.0  /<MHP_RADM88>;
<P14> OP2          =      ALD    +    HO2    +   HO   # 1.0  /<HOP_RADM88>;
<P15> PAA          =      MO2    +    HO              # 1.0  /<PAA_RADM88>;
<P16> KET          =      ACO3   +    ETHP            # 1.0  /<KETONE_RADM88>;
<P17> GLY          = 0.13*HCHO + 1.870*CO             # 1.0  /<GLYform_RADM88>;
<P18> GLY          = 0.45*HCHO + 1.550*CO + 0.800*HO2 # 1.0  /<GLYmol_RADM88>;
<P19> MGLY         =      ACO3   +    HO2    +   CO   # 1.0  /<MGLY_RADM88>;
<P20> DCB          = 0.98*HO2 +  0.020*ACO3   +  TCO3 # 1.0  /<UDC_RADM88>;
<P21> ONIT         = 0.20*ALD +  0.800*KET + HO2 + NO2 # 1.0  /<ORGNIT_RADM88>;
<  1> O3P + O2 + M =      O3                          # 6.0E-34^-2.3;
<  2> O3P + NO2    =      NO                          # 6.5E-12  @ -120.0;
<  3> O1D + N2     =      O3P                         # 1.8E-11  @ -110.0;
<  4> O1D + O2     =      O3P                         # 3.2E-11  @ -70.0;
<  5> O1D + H2O    = 2.0* HO                          # 2.20E-10;
<  6> O3  +  NO    =      NO2                         # 2.00E-12  @1400.0;
<  7> O3  +  HO    =      HO2                         # 1.60E-12  @ 940.0;
<  8> O3  +  HO2   =      HO                          # 1.10E-14  @ 500.0;
<  9> HO2 +  NO    =      NO2    +    HO              # 3.70E-12  @ -240.0;
< 10> HO2 +  NO2   =     HNO4                  # 1.8E-31^-3.2 &4.7E-12^1.4;
< 11> HNO4         =      HO2    +    NO2             # 2.1E-27 @  10900.0 *E< 10>;
< 12> HO2 + HO2    =      H2O2        %3 # 2.20E-13@-620.0  & 1.90E-33@-980.0;
< 13> HO2 + HO2 + H2O =   H2O2        %3 # 3.08E-34@-2820.0 & 2.66E-54@-3180.0;
< 14> H2O2 + HO       =   HO2                      # 3.300E-12  @ 200.0;
< 15> NO  +  HO    =      HONO           # 7.0E-31^-2.6 &1.5E-11^-0.5;
```

```
< 16> NO  +  NO  + O2 = 2.0*NO2                              # 3.300E-39 @ -530.0;
< 17> O3  +  NO2   =      NO3                                # 1.4000E-13 @ 2500.0;
< 18> NO3 +  NO    =   2.0*NO2                               # 1.7000E-11 @ -150.0;
< 19> NO3 +  NO2   =       NO     +     NO2                 # 2.5000E-14  @ 1230.0;
< 20> NO3 +  HO2   =       HNO3                             # 2.5000E-12;
< 21> NO3 +  NO2   =       N2O5                  # 2.2E-30^-4.3 &1.5E-12^-0.5;
< 22> N2O5         =       NO2    +     NO3     # 1.10E-27 @ -11200.0 *E<21>;
< 23> N2O5 + H2O   =    2.0*HNO3                 # 2.00E-21;
< 24> HO  +  NO2   =       HNO3                 # 2.6E-30^-3.2 &2.4E-11^-1.3;
< 25> HO  +  HNO3  =       NO3     %2 #7.2E-15@-785&4.1E-16@-1440&1.9E-33@-725;
< 26> HO  +  HNO4  =       NO2                              # 1.3000E-12 @ -380.0;
< 27> HO  +  HO2   =                                        # 4.6000E-11  @ -230.0;
< 28> HO  +  SO2   = SULF  +  HO2 + SULAER      # 3.0E-31^-3.3 &1.5E-12^0.0;
< 29> CO  +  HO    =       HO2                 %1 #1.5000E-13;
< 30> HO           =       MO2    (X 300 SQUARED)           # 28.3^2   @ 1280.0;
< 31> ETH +  HO    =       ETHP   (X 300 SQUARED)           # 1.233E-12^2 @ 444.0;
< 32> HC3 +  HO    =  0.83*HC3P + 0.17*HO2 +  0.009*HCHO  + 0.075*ALD
                     + 0.025*KET                           # 1.59E-11  @ 540.0;
< 33> HC5 +  HO    =     HC5P  +  0.250*XO2                 # 1.73E-11  @ 380.0;
< 34> HC8 +  HO    =     HC8P  +  0.750*XO2 + HC8AER        # 3.64E-11  @ 380.0;
< 35> OL2 +  HO    =     OL2P                               # 2.15E-12  @ -411.0;
< 36> OLT +  HO    =     OLTP                               # 5.32E-12  @ -504.0;
< 37> OLI +  HO    =     OLIP  + OLIAER                     # 1.07E-11  @ -549.0;
< 38> TOL +  HO    = 0.75*TOLP + 0.250*CSL + 0.250*HO2
                     + TOLAER                              # 2.10E-12  @ -322.0;
< 39> XYL +  HO    = 0.83*XYLP + 0.170*CSL + 0.170*HO2
                     + XYLAER                              # 1.89E-11  @ -116.0;
< 40> CSL +  HO    = 0.10*HO2  + 0.900*XO2 + 0.900*TCO3
                     + CSLAER                              # 4.00E-11;
<40a> CSL +  HO    =     CSL                                # 0.9*K<40>;
< 41> HCHO + HO    =     HO2     +    CO                    # 9.0000E-12;
< 42> ALD +  HO    =     ACO3                               # 6.8700E-12 @ -256.0;
< 43> KET +  HO    =     KETP                               # 1.2000E-11 @ 745.0;
< 44> GLY +  HO    =     HO2  +  2.000*CO                   # 1.1500E-11;
< 45> MGLY + HO    =     ACO3    +    CO                    # 1.7000E-11;
< 46> DCB +  HO    =     TCO3                               # 2.8E-11;
< 47> OP1 +  HO    = 0.5*MO2 + 0.500*HCHO +0.500*HO       # 1.0000E-11;
< 48> OP2 +  HO    = 0.5*HC3P + 0.500*ALD + 0.500*HO      # 1.0000E-11;
< 49> PAA +  HO    =     ACO3                               # 1.0000E-11;
< 50> PAN +  HO    =     HCHO + NO3 + XO2 (x 300 sq)        # 6.1650E-13^2  @ 444.0;
< 51> ONIT + HO    =     HC3P    +    NO2                   # 1.5500E-11  @ 540.0;
< 52> ISO +  HO    =     OLTP                               # 2.5500E-11  @ -409.0;
< 53> ACO3 + NO2   =     PAN                                # 2.8000E-12  @ -181.0;
< 54> PAN          =     ACO3    +    NO2                   # 1.9500E+16  @ 13543.0;
< 55> TCO3 + NO2   =     TPAN                               # 4.7000E-12;
< 56> TPAN         =     TCO3    +    NO2                   # 1.9500E+16  @ 13543.0;
< 57> MO2 +  NO    =     HCHO    +    HO2   +    NO2       # 4.2000E-12  @ -180.0;
< 58> HC3P + NO    = 0.75*ALD  +  0.25*KET + 0.09*HCHO  + 0.036*ONIT
                     + 0.964*NO2 +  0.964*HO2              # 4.2000E-12  @ -180.0;
< 60> HC5P + NO    = 0.38*ALD + 0.69*KET + 0.08*ONIT
                     +  0.92*NO2 + 0.92*HO2                # 4.2000E-12  @ -180.0;
< 62> HC8P + NO    = 0.35*ALD + 1.06*KET + 0.04*HCHO
                     + 0.24*ONIT + 0.76*NO2 +   0.76*HO2   # 4.2000E-12  @ -180.0;
< 64> OL2P + NO    = 1.6*HCHO + HO2 + NO2 + 0.20*ALD      # 4.2000E-12  @ -180.0;
< 65> OLTP + NO    =     ALD  + HCHO + HO2 + NO2          # 4.2000E-12  @ -180.0;
< 66> OLIP + NO    =     HO2 + 1.45*ALD +  0.28*HCHO
                     +  0.1*KET   +      NO2               # 4.2000E-12  @ -180.0;
< 67> ACO3 + NO    =     MO2    +     NO2                 # 4.2000E-12  @ -180.0;
< 68> TCO3 + NO    =     NO2 + 0.920*HO2 +  0.890*GLY  + 0.110*MGLY
                     + 0.050*ACO3 +  0.950*CO  +  2.000*XO2
                                                           # 4.2000E-12  @ -180.0;
< 69> TOLP + NO    =     NO2 +  HO2 +  0.17*MGLY
```

```
                  +   0.16*GLY +   0.70*DCB                    # 4.2000E-12  @ -180.0;
< 70> XYLP + NO    =  NO2 + HO2 + .45*MGLY + .806*DCB           # 4.2000E-12  @ -180.0;
< 71> ETHP + NO    =  ALD  +   HO2 +    NO2                     # 4.2000E-12  @ -180.0;
< 72> KETP + NO    =  MGLY +       NO2  +       HO2             # 4.2000E-12  @ -180.0;
< 73> OLN  + NO    =  HCHO   +    ALD +    2.0*NO2             # 4.2000E-12  @ -180.0;
< 74> HCHO + NO3   =  HO2    +     HNO3   +     CO              # 6.000E-13   @ 2058.0;
< 75> ALD  + NO3   =  ACO3   +     HNO3                         # 1.400E-12   @ 1900.0;
< 76> GLY  + NO3   =  HNO3    +    HO2 +  2.000*CO              # 6.000E-13   @ 2058.0;
< 77> MGLY + NO3   =  HNO3    +    ACO3    +    CO              # 1.400E-12   @ 1900.0;
< 78> DCB  + NO3   =  HNO3    +    TCO3                         # 1.400E-12   @ 1900.0;
< 79> CSL  + NO3   =  HNO3   +   XNO2  + 0.500*CSL
                  + 0.500*CSLAER                               # 2.200E-11;
< 80> OL2  + NO3   =  OLN                                      # 2.000E-12   @ 2923.0;
< 81> OLT  + NO3   =  OLN                                      # 1.000E-11   @ 1895.0;
< 82> OLI  + NO3   =  OLN + OLIAER                             # 3.230E-11   @ 975.0;
< 83> ISO  + NO3   =  OLN                                      # 5.810E-13;
< 84> OL2  + O3    =  HCHO + 0.400*ORA1 + 0.420*CO
                  + 0.120*HO2                                  # 1.200E-14   @ 2633.0;
< 85> OLT  + O3    = 0.53*HCHO + 0.500*ALD +  0.33*CO  + 0.20*ORA1
                  + 0.20*ORA2 + 0.23*HO2 +  0.22*MO2 +  0.10*HO
                                                               # 1.3200E-14  @ 2105.0;
< 86> OLI  + O3    = 0.18*HCHO + 0.72*ALD +  0.10*KET  + 0.23*CO  + 0.06*ORA1
                  + 0.29*ORA2 + 0.26*HO2 +  0.14*HO + 0.31*MO2
                  + OLIAER                                     # 7.2900E-15  @ 1136.0;
< 87> ISO  + O3    = 0.53*HCHO + 0.500*ALD +  0.33*CO  +  0.20*ORA1
                  + 0.20*ORA2 + 0.23*HO2 +  0.22*MO2 +  0.10*HO
                                                               # 1.230E-14   @ 2013.0;
< 88> HO2  + MO2   =    OP1                                   # 7.700E-14   @ -1300.0;
< 89> HO2  + ETHP  =    OP2                                   # 7.700E-14   @ -1300.0;
< 90> HO2  + HC3P  =    OP2                                   # 7.700E-14   @ -1300.0;
< 91> HO2  + HC5P  =    OP2                                   # 7.700E-14   @ -1300.0;
< 92> HO2  + HC8P  =    OP2                                   # 7.700E-14   @  1300.0;
< 93> HO2  + OL2P  =    OP2                                   # 7.700E-14   @ -1300.0;
< 94> HO2  + OLTP  =    OP2                                   # 7.700E-14   @ -1300.0;
< 95> HO2  + OLIP  =    OP2                                   # 7.700E-14   @ -1300.0;
< 96> HO2  + KETP  =    OP2                                   # 7.700E-14   @ -1300.0;
< 97> HO2  + ACO3  =    PAA                                   # 7.700E-14   @ -1300.0;
< 98> HO2  + TOLP  =    OP2                                   # 7.700E-14   @ -1300.0;
< 99> HO2  + XYLP  =    OP2                                   # 7.700E-14   @ -1300.0;
<100> HO2  + TCO3  =    OP2                                   # 7.700E-14   @  1300.0;
<101> HO2  + OLN   =    ONIT                                  # 7.700E-14   @ -1300.0;
<102> MO2  + MO2   = 1.5*HCHO   +    HO2                     # 1.90E-13    @ -220.0;
<103> MO2  + ETHP  = 0.75*HCHO   +    HO2 +  0.75*ALD        # 1.40E-13    @ -220.0;
<104> MO2  + HC3P  = 0.84*HCHO + 0.770*ALD +  0.260*KET
                  + 1.000*HO2                                  # 4.20E-14 @ -220.0;
<105> MO2  + HC5P  = 0.77*HCHO + 0.41*ALD +  0 75*KET
                  + 1.000*HO2                                  # 3.40E-14  @ -220 0;
<106> MO2  + HC8P  = 0.80*HCHO + 0.46*ALD +  1.39*KET
                  + 1.000*HO2                                  # 2.90E-14  @ -220.0;
<107> MO2  + OL2P  = 1.55*HCHO + 0.350*ALD +  HO2              # 1.40E-13    @ -220.0;
<108> MO2  + OLTP  = 1.25*HCHO + 0.750*ALD +  HO2              # 1.40E-13    @ -220.0;
<109> MO2  + OLIP  = 0.89*HCHO + 0.725*ALD +  HO2
                  + 0.55*KET                                   # 1.70E-14  @ -220.0;
<110> MO2  + KETP  = 0.75*HCHO + 0.750*MGLY + HO2              # 1.70E-14  @ -220.0;
<111> MO2  + ACO3  =  HCHO + 0.5*HO2 +  0.5*MO2
                  + 0.50*ORA2                                  # 9.60E-13  @ -220.0;
<112> MO2  + TOLP  =  HCHO + 0.17*MGLY + 0.16*GLY
                  +  0.70*DCB  +  2.0*HO2                      # 1.70E-14  @ -220.0;
<113> MO2  + XYLP  =  HCHO + 0.45*MGLY + 0.806*DCB
                  + 2.000*HO2                                  # 1.70E-14  @ -220.0;
<114> MO2  + TCO3  = .50*HCHO + 0.445*GLY +  0.055*MGLY
                  + 0.50*ORA2 + 0.025*ACO3 + 0.460*HO2
```

```
                  +  0.475*CO   +  XO2                    # 9.60E-13  @ -220.0;
<138> MO2 +  OLN  =  1.75*HCHO + 0.50*HO2 + ALD + NO2     # 1.70E-14  @ -220.0;
<115> ETHP + ACO3 =  ALD + 0.5*HO2 +  0.5*MO2
                  + 0.5* ORA2                             # 3.40E-13  @ -220.0;
<116> HC3P + ACO3 =  .77*ALD + 0.26* KET +  0.5*HO2
                  + 0.50*MO2 +  0.5*ORA2                  # 1.00E-13  @ -220.0;
<117> HC5P + ACO3 = 0.41*ALD + 0.75*KET +  0.5*HO2
                  + 0.50*MO2 +  0.5*ORA2                  # 8.40E-14  @ -220.0;
<118> HC8P + ACO3 = 0.46*ALD + 1.39*KET +  0.5*HO2
                  + 0.50*MO2 +  0.5*ORA2                  # 7.20E-14  @ -220.0;
<119> OL2P + ACO3 = 0.80*HCHO + 0.6*ALD +  0.5*HO2
                  + 0.5*MO2 +  0.5*ORA2                   # 3.40E-13  @ -220.0;
<120> OLTP + ACO3 =  ALD + 0.5*HCHO + 0.5*HO2
                  + 0.5*MO2 +  0.5*ORA2                   # 3.40E-13  @ -220.0;
<121> OLIP + ACO3 = 0.725*ALD + 0.55*KET +  0.14*HCHO
                  + 0.5*HO2 +  0.50*MO2  + 0.5*ORA2       # 4.20E-14   @ -220.0;
<122> KETP + ACO3 =  MGLY + 0.5*HO2 +  0.5*MO2
                  + 0.5*ORA2                              # 4.20E-14   @ -220.0;
<123> ACO3 + ACO3 =  2.0*MO2                              # 1.19E-12  @ -220.0;
<124> ACO3 + TOLP =  MO2 + 0.170*MGLY + 0.16*GLY
                  +  0.70*DCB  +    HO2                   # 4.20E-14   @ -220.0;
<125> ACO3 + XYLP =  MO2 + 0.45*MGLY + 0.806*DCB
                  +  HO2                                  # 4.20E-14   @ -220.0;
<126> ACO3 + TCO3 =  MO2 + 0.92*HO2 +  0.89*GLY
                  +  0.11*MGLY + 0.05*ACO3 + 0.95*CO
                  + 2.0*XO2                               # 1.19E-12  @ -220.0;
<139> ACO3 + OLN  =  HCHO +  ALD +  0.5*ORA2
                  +  NO2 +  0.5* MO2                      # 4.20E-14   @ -220.0;
<140> OLN +  OLN  = 2.0*HCHO + 2.0*ALD +  2.0*NO2       # 3.60E-16  @ -220.0;

<127> XO2 +  HO2  =    OP2                               # 7.70E-14  @ -1300.0;
<128> XO2 +  MO2  =    HCHO   +     HO2                  # 1.70E-14  @ -220.0;
<129> XO2 +  ACO3 =    MO2                               # 4.20E-14   @ -220.0;
<130> XO2 +  XO2  =                                      # 3.60E-16  @ -220.0;
<131> XO2 +  NO   =    NO2                               # 4.2000E-12  @ -180.0;
<132> XNO2 + NO2  =    ONIT                              # 4.2000E-12  @ -180.0;
<133> XNO2 + HO2  =    OP2                               # 7.70E-14  @ -1300.0;
<134> XNO2 + MO2  =    HCHO   +     HO2                  # 1.70E-14  @ -220.0;
<135> XNO2 + ACO3 =    MO2                               # 4.20E-14   @ -220.0;
<136> XNO2 + XNO2 =                                      # 3.60E-16  @ -220.0;
<AE1> TERP + HO    = TERPAER + HO                        # 1.0*K<37>;
<AE1> TERP + NO3   = TERPAER + NO3                       # 1.0*K<82>;
<AE1> TERP + O3    = TERPAER + O3                        # 1.0*K<86>;
endmech
```

Process Analysis (PROCAN)

Introduction. Process Analysis is a diagnostic tool that captures model generated data not routinely output by models and provided quantitative information on the contributions of individual physical and chemical processes to model predictions. This quantitative information can then be used to form a picture of how the model obtains its predictions. In the CMAQ modeling system, two types of Process Analysis data can be captured during a CCTM simulation: integrated process rates (IPRs) and integrated reaction rates (IRRs). IPRs give the contributions of individual physical processes and the net effect of chemical reactions to the overall model concentrations. IRRs give the contributions of individual chemical reactions to the net effects of chemical reaction on species concentrations. Because the amount of IPR and IRR

data that can be obtained may be large and the analysis of such data can be fairly complex, the user is advised to read Chapter 17 in the Models-*3 Volume 6b: Science of the Community Multi-scale Air Quality (CMAQ) Modeling System* before attempting to use this tool. The discussion here focuses primarily on the procedures that must be followed to capture Process Analysis data rather than on what data should be captured and how it might be analyzed. That is discussed to some degree in the Volume 6b Science document and in much greater detail in the references that are cited in that document.

As noted in Section 7.5.2 of the User's Guide, all of the predefined mechanisms have been set up to omit process analysis. If process analysis is to be invoked using one of the predefined mechanisms, it will be necessary to modify the mechanism in the Models-3 Chemical Mechanism Manager. That chemical mechanism will then have to be assigned a new name to reflect the addition of process analysis. See the section on modifying a chemical mechanism for the procedures to follow. The process analysis data themselves are entered by selecting the *Process Analysis* tab in the Chemical Mechanism Manager. As described in Section 7.5.2, the process analysis inputs consist of a series of commands that can be entered directly in the window or can be imported into the framework via an ASCII text file. Since the process analysis commands can be somewhat lengthy, the latter approach is recommended. The ASCII file can be created with any editor or word processor that generates a standard ASCII file.

The commands in a command file are processed transparently in the Models-3 framework by the Process Analysis Control Program (PACP). This program reads and interprets the commands and then generates three include files (.ext) necessary to instrument the CCTM to produce the Process Analysis outputs that were requested in the commands. These include files are automatically made available to the Model Builder for inclusion in the configuration file used to make an executable with your requested process analysis. Since this is handled automatically within the framework, users need only concern themselves with developing the appropriate input PACP commands. For reference, see Example M-3 of an input command file and a corresponding output report PA_REPORT (Example M-4) generated by the PACP are included at the end of the Appendix. The reader may wish to refer to these examples for illustrations of the commands that are described below.

The process analysis commands themselves are formatted according to a simple set of rules and a free-form format. Nevertheless, each command has a special syntax that must be followed and each command makes use of special keywords and/or operators that have specific meaning to the PACP. The commands are of three major types: global commands, IPR commands, and IRR commands. The discussion begins first, however, with a description of some general rules for entering the commands.

General rules. The free form format is similar to that used by the general mechanism processor. In general, white spaces are ignores and line wrap around is allowed (i.e., commands can be

continued on a subsequent line after a hard return). The free form format also allows embedded comments and makes use of special symbols to indicate the type of input data. Special rules for naming species, entering labels, and specifying numerical values such as stoichiometric coefficients and rate constant parameters are also used. Each are discussed below:

- *Comments.* All lines that have an exclamation point in column 1 are treated as comment lines and are ignored by the PACP. Any text enclosed in braces or parentheses is also treated as comment and ignored by the PACP.

- *Species names.* The PACP recognizes two types of species names - model species and user defined Process Analysis species. Model species refer to species names in the Species Tables (see the first part of this Appendix). These names must be spelled exactly as they appear in that table. The following special rules have been established for user-defined species names.

  a) The Process Analysis species names must not contain any blanks and can be up to 16 characters long.

  b) The name must begin with an alphabetic character, but may contain any alphanumeric character (i.e., "A-Z", "a-z"., and "0-9") or the characters ":" and "_" after the first position.

  c) The name is case sensitive. Thus, NO2 and no2 would represent two different species.

  d) A name can have embedded comments, but cannot span two lines.

- *Label names.* For some of the IRR commands, reaction labels appearing in the chemical mechanism reaction list input file can be referenced. These labels would normally be spelled exactly as they appear in the chemical mechanism reaction list input file, except embedded comments and their delimiters should be omitted. However, any embedded blanks in those label names should be omitted. and the label name should contain no more that 16 non-blank characters.

- *Numbers.* Numerical inputs in the command file can be either integer (e.g., 5), floating point (e.g., 5.0), or exponential (e.g., 5.0E+00). With the exponential format, the "E" may be either upper or lower case and a positive exponent will be assumed if the sign of the exponent is missing.

- *Command Line Terminator.* Input command lines are terminated by a semicolon.

Process Analysis Commands. This section describes the individual process analysis commands that are used to construct a PACP command file. In the description of these commands, the following conventions will be used. Bold type is used for PACP keywords and normal type for user supplied inputs. Alternative inputs are separated by vertical bars (|), and optional inputs are enclosed in braces ({}). Again, examples of these commands can be found in Exhibits 1 and 2 at the end of this Appendix. The commands are as follows:

- *Process Analysis Global Commands.* The three process analysis global commands are shown in Table M-10. Each is discussed further below:

  OUTPUT_DOMAIN. As described above, Process Analysis outputs consist of IPRs and/or IRRs. These outputs are generated for the same time periods as the standard model predicted concentrations. Like the concentration outputs, the Process Analysis outputs will be generated for every model grid by default unless this command is used to limit the Process Analysis output domain. Only one output domain can be specified and it must be a block of contiguous cells. Although this command can appear anywhere in the input file, it is recommended that it be inserted near the beginning of the file. If the command occurs more than one time in the input file, the last entry will be used.

  DEFINE FAMILY. This command is used to define a new species which can be referenced in other Process Analysis commands. The new species is limited to a linear combination of model species names (i.e., a family definition cannot reference another family). Families must be defined before they are referenced in other commands.

  ENDPA. This command signifies the end of the PACP commands and thus should be the last command in the file.

- *Integrated Process Rate Command.* There is only one command for IPRs, and it controls the specific IPRs that are output. Its syntax and description is shown in Table M-11. Note that one command can cause many IPR outputs to be generated. For example, if one species or family is specified in a command but no process codes are specified, then one IPR will be generated for each science process for that species (i.e., 12 IPRs will be output). Similarly if the keyword ALL is used

for the species name and no process code is specified, 12 IPRs will be generated for every model species. This would generate an output file that would be approximately 12 times as large as the corresponding concentration file! Also the impact on the CCTM memory requirements would be substantial since adding a single IPR output has roughly the same effect as adding a model species. Thus, some caution should be exercised when formulating the commands to request IPR outputs.

The IPR outputs are written to an IO/API output file in exactly the same format and with the same number of time steps as the concentration output file. Since the IO/API currently has a limit of 120 output variables on a file, multiple files will be output if this limit is exceeded.

Integrated Reaction Rate Commands. There are two types of IRR commands -- global and output -- and these are shown in Table M-12 and Table M-13, respectively. Each is discussed further below:

IRR_TYPE. This command controls the form of the IRR output that is generated. If it is omitted, then it will be assumed that a partial IRR analysis has been requested (i.e., IRR output commands are expected in the file.) If it is included, it should be inserted at the very beginning of the PACP command file. Note IRR type equal to full will cause one IRR to be generated for each chemical reaction in the gas-phase mechanism. If the IO/API limit of 120 variables per output file is exceeded, multiple output files will be generated.

DEFINE CYCLE. As noted in Table M-12, this command computes the net of the production and loss of a species and assigns that value to a cycle name that can be referenced in the IRR output command. The command must appear before the cycle name is referenced. This command is useful when the same cycle is needed for multiple IRR outputs.

DEFINE RXNSUM. This command is similar to the cycle command, except that it allows the user to construct any linear combination of IRRs for individual chemical reactions. Again, this command is most useful when the same combination of IRRs is needed for multiple IRR outputs. The RXNSUM definition must precede any references to it in the command file.

IRR_OUTPUT. This is the command that causes a specific IRR output to be generated. As noted in Table M-13, the command can include a reference to a previously defined cycle or reaction sum name, a chemical reaction label, or one of the special operators that are described in Table M-14. One output variable is

generated for each IRR_OUTPUT command. Since the operators can be fairly complicated, the user is encouraged to review the Process Analysis output report whenever the IRR_OUTPUT command is used to ensure that the requested outputs are actually what the user desires. The reader is also referred to the Models-*3 Volume 6b: Science of the Community Multi-scale Air Quality (CMAQ) Modeling System* document for a more detailed discussion of the operators and their use.

DESCRIPTION. This command allows the user to provide a long description of the output variable. This description will be assigned to the long variable name on the IO/API output file.

Table M-10
Process Analysis Global Commands

OUTPUT_DOMAIN = {LOCOL[n_1] + HICOL[n_2] + LOROW[n_3] + HIROW[n_4] + LOLEV[n_5] + HILEV[n_6]};

The OUTPUT_DOMAIN command provides the capability to limit the IPR and IRR output data to a portion of the modeling domain. The n_i in brackets are numbers that define the bounds of the output domain relative to the number of columns, rows, and vertical levels in the modeling domain. Thus, for example, the value for n_1 must be greater than or equal to one and less than or equal to the number of columns in the domain. If any one domain specifier is omitted, the corresponding end of the modeling domain is used as a default. If the command is omitted entirely, output is generated for the entire domain.

DEFINE FAMILY familyname = {c_1*}species$_1$ {+ {c_2*}species$_2$ + ...};

The DEFINE FAMILY command is used to define a group of species as members of a family. The user specified "familyname" must be unique, and can be referenced in subsequent commands. The c_i are numerical coefficients that default to one if not specified; "species$_i$" represents the model species names.

ENDPA;

The ENDPA command signifies the end of the command input in the PACP command file.

Table M-11
Integrated Process Rate Output Command

IPR_OUTPUT species|familyname|**ALL** = {$pcode_1$ + $pcode_2$ + ...};

The IPR_OUTPUT command defines specific IPR outputs to be generated during a CMAQ CTM simulation. A model species name, family name, or the keyword ALL must follow the IRR_OUTPUT keyword. The keyword ALL refers to all model species. IPRs are generated for the selected species or family, and they are controlled by the specified values of $pcode_i$, where $pcode_i$ corresponds to one of the process codes listed below. If no process codes are specified, IPRs will be generated for every science process (i.e., the first 12 codes shown below). The output variables that are generated are named either species_$pcode_i$ or familyname_$pcode_i$.

Process codes and definitions:

XADV	Advection in the E-W direction
YADV	Advection in the N-S direction
ZADV	Vertical advection
ADJC	Mass adjustment
HDIF	Horizontal diffusion
VDIF	Vertical diffusion
EMIS	Emissions
DDEP	Dry deposition
CHEM	Chemistry
AERO	Aerosols
CLDS	Cloud processes and aqueous chemistry
PING	Plume-in-grid
XYADV	Sum of XADV and YADV
XYZADV	Sum of XADV, YADV, ZADV
TOTADV	Sum of XADV, YADV, ZADV, and ADJC
TOTDIF	Sum of HDIF and VDIF
TOTTRAN	Sum of XADV, YADV, ZADV, ADJC, HDIF and VDIF

Table M-12
Integrated Reaction Rate Global Commands

IRR_TYPE = FULL|PARTIAL|NONE;

The IRR_TYPE command defines the type of IRR analysis. With the type set to FULL, IRRs for each individual reaction will be calculated and written to the IRR output file, and all other IRR commands will be ignored. IRR_TYPE set to PARTIAL indicates that the IRR commands following this command are to be processed to produce user defined IRR outputs. Type set to NONE causes all IRR commands to be ignored and no IRR output to be generated. If the command is omitted, type PARTIAL is assumed.

DEFINE CYCLE cyclename = species$_1$;

The DEFINE CYCLE command is used to compute the net of all chemical production and loss of a species. Thus, this quantity is computed by summing the IRRs for all reactions in which a species is consumed, and then subtracting that sum from the sum of the IRRs for all reactions in which the species is produced. The "cyclename" is a user defined name that must be unique, and can be referenced in subsequent IRR_OUTPUT commands.

DEFINE RXNSUM sumname = {c_1*}<rxlabl$_1$> { ± {c_2*} <rxlabl$_2$> ± ...};

The RXSUM command is used to compute a linear combination of the IRRs for individual reactions that can then be referenced in a subsequent IRR_OUTPUT command; "sumname" is user defined and must be unique. The linear combination of IRRs is defined according to the expressions following the equal signs that specify which reactions's IRRs to sum. The "rxlabl$_i$" is the reaction label that is used in the generalized mechanism. The "c_i" are optional numerical coefficients that default to one if not specified.

Table M-13
Integrated Reaction Rate Output Commands

IRR_OUTPUT irrname = {c_1*}op_1|cyclname{$qual_1$}|sumname{$qual_1$}|<$rxlabl_1$>
{ ± {c_2*}op_2|cyclname{$qual_2$}|sumname{$qual_2$}|<$rxlabl_2$> + ...};

The IRR_OUTPUT command defines a specific IRR output to be generated during a CMAQ simulation. It is constructed by specifying a linear combination of IRR operators, IRR global definitions, or IRRs for specified reactions. Each individual term in the combination must include either one of the five IRR operators described in Table M-14 (i.e., op_i), a cycle name, a reaction sum name, or a reaction label enclosed in "greater than" and "less than" signs. The optional qualifiers ($qual_i$) for cyclename or reaction sum name can be either POSONLY or NEGONLY. With these qualifiers, the defined quantity is included as a term only when it is positive or negative, respectively. If the name is not qualified, the quantity is included regardless of sign. The numerical coefficients for each term (c_i) are assumed to be one unless they are explicitly included. The irrname that is supplied by the user will be assigned as the variable name in the IO/API IRR output file.

DESCRIPTION = 'description';

The description command is provided to allow the user to specify a long description of the output variable that will be included on the IO/API IRR output name. If a description is not specified for an IRR_OUTPUT variable, the irrname (or short name) will be used in the output file. If the description command is used, it should be located immediately following the IRR_OUTPUT command to which it applies.

Table M-14
Integrated Reaction Rate Output Operators

PROD[species$_1$] {**FROM**[species$_2$] {**AND**|**OR** [species$_3$] }}

The production operator (PROD) is used to compute the total production of a species by summing the IRRs of all reactions in which species$_1$ appears as a product. The optional qualifiers FROM and/OR restrict the sum to include only those reactions in which species$_2$ and/or species$_3$ are reactants. The "species$_i$" can be any gas-phase mechanism species or a family of gas-phase species; "species$_2$" or species$_3$" may also be the keyword HV to restrict the selection to photolytic reactions.

NETP[species$_1$] {**FROM**[species$_2$] {**AND**|**OR** [species$_3$] }}

The net production operator (NETP) is very similar to the production operator since it is used compute the production of a species. Whereas the PROD operator includes every reaction in which species$_1$ occurs as a product, the NETP operator includes only those reactions in which the net production of species$_1$ is greater than zero. Thus, if species$_1$ or any member of the family species appears as both a reactant and a product with equal stoichiometry in a reaction, the PROD operator will include it, but the NETP operator will not. This operator is useful for getting the net production of a family, for example.

LOSS[species$_1$] {**AND**|**OR** [species$_2$] }

The loss operator (LOSS) is used to compute the total loss of a species by summing the IRRs of all reactions in which species$_1$ appears as a reactant. The optional qualifier AND restricts the sum to include only those reactions in which both species$_1$ and species$_2$ are reactants. Similarly, the OR qualifier includes all reactions in which either "species$_1$" or "species$_2$" appears as a reactant, where "species$_i$" or "species$_2$" can be any gas-phase species in the mechanism, a family name that includes only gas-phase mechanism species, or the keyword HV to restrict the selection of reactions to those that are photolytic.

NETL[species$_1$] {**AND**|**OR** [species$_2$] }}

The net loss operator (NETL) is very similar to the loss operator since it is used to compute the loss of a species. However, it includes only those reactions in which there is a net loss of "species$_1$" and/or "species$_2$" . Thus, if species $_1$ or any member of the family species appears as both a reactant and a product with equal stoichiometry in reaction, the NETL operator will not include it in summing the loss of that species, whereas the LOSS operator will include the IRR for that reaction.

NET[species$_1$]

The net operator (NET) is similar to the CYCLE definition since it gives the net of the production and the loss of a species for all reactions in which "species$_1$" appears either as reactant or a product; "species$_1$" may be any gas-phase, mechanism species or any family consisting wholly of gas-phase mechanism species.

Example M-3 Process Analysis Control Program Input Command File (PA_INPUT)

```
FAMILY OX = O3 + NO2 + 2*NO3 + O3P +O1D + PAN + HNO4
          +3*N2O5 + TPAN + OLN;

IRR_OUTPUT OXprod = NETP[OX];

DESCRIPTION = 'OX Production';


IRR_OUTPUT Oxloss = NETL[OX];

DESCRIPTION = 'OX Loss';

!=============================================================
! IRR_OUTPUT EXAMPLE 3 --  new OH
!=============================================================

DEFINE FAMILY VOCA = OL2 + OLI + OLT + ISO;

DEFINE CYCLE HONOcyc = H IRRTYPE = PARTIAL;

OUTPUT_DOMAIN = LOLEV[1] + HILEV[5];

!=============================================================
! IRR_OUTPUT EXAMPLES 1 and 2 -- OX Production & Loss
!=============================================================
DEFINE ONO;
DEFINE RXNSUM H2O2_OHcyc =  <9> -  <12> -  <13>;
DEFINE RXNSUM HNO3_OHcyc =  <P5> -  <24>;
DEFINE RXNSUM OP1_OHcyc  = <P13> -  <88>;
DEFINE RXNSUM OP2_OHcyc  = <P14> -  <89> -  <90> -  <91> -  <92> -
                  <93>  -  <94> -  <95> -  <96> -  <97> -
                  <98>  -  <99> - <100> - <101>;
DEFINE RXNSUM PAA_OHcyc  = <15>  -  <97>;

IRR_OUTPUT newOH = PROD [HO] FROM [O3] AND [VOCA] + 2*H2O2_OHcyc[POSONLY]
+
              HNO3_OHcyc[POSONLY] +  HONOcyc[NEGONLY]  +
              OP1_OHcyc[POSONLY]  + OP2_OHcyc[POSONLY] +
```

```
            PAA_OHcyc[POSONLY];

DESCRIPTION = 'new OH';

!=============================================================================
! IRR_OUTPUT EXAMPLE 4 -- NO2 from HO2
!=============================================================================


IRR_OUTPUT NO2fromHO2 = PROD[NO2] FROM [HO2];

DESCRIPTION = 'NO2 FROM HO2';


!=============================================================================
! IRR_OUTPUT EXAMPLE 5 -- Net ONIT Production
!=============================================================================

IRR_OUTPUT netONIT = NET[ONIT];

DESCRIPTION = 'Net production of ONIT';

!=============================================================================
! IRR_OUTPUT EXAMPLE 6 -- OH reacting with VOC
!=============================================================================

DEFINE FAMILY VOC = {CH4 +} CO + ETH +HC3 + HC5 + HC8 + OL2 + OLT +
            OLI + ISO + TOL + CSL + XYL + HCHO + ALD + KET +
            GLY + MGLY + DCB;


IRR_OUTPUT lossOH_HC = LOSS[VOC] AND [HO];

DESCRIPTION = 'Loss of HC plus OH';


!=============================================================================
! IPR_OUTPUT EXAMPLES
!=============================================================================

IPR_OUTPUT NO;
```

```
IPR_OUTPUT VOC;

IPR_OUTPUT HCHO = CHEM + EMIS;

ENDPA;
```

Example M-4 Process Analysis Report (PA_REPORT)

```
********** PROCESS ANALYSIS CONTROL PROGRAM OUTPUT REPORT **********
           Created  12:57:53 EDT June 15, 1998

ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c                  OUTPUT DOMAIN SECTION
c  Process Monitoring and IRR outputs will be generated for the
c  following cells relative to the full modeling domain.
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
     Starting column:  1
     Ending column:   20
     Starting row:     1
     Ending row:      20
     Starting layer:   1
     Ending layer:     5

ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c              INTEGRATED PROCESS RATE SECTION
c  The following section lists the IPR outputs generated from the
c  command file inputs.
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced families are defined as follows:
c
c VOC=CO+ETH+HC3+HC5+HC8+OL2+OLT+OLI+ISO+TOL+CSL+XYL+HCHO+ALD+KET

c      +GLY+MGLY+DCB
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
    IPR_OUTPUT(  1) = XADV of NO
    IPR_OUTPUT(  2) = YADV of NO
    IPR_OUTPUT(  3) = ZADV of NO
```

```
      IPR_OUTPUT(  4) = ADJC of NO
      IPR_OUTPUT(  5) = HDIF of NO
      IPR_OUTPUT(  6) = VDIF of NO
      IPR_OUTPUT(  7) = EMIS of NO
      IPR_OUTPUT(  8) = DDEP of NO
      IPR_OUTPUT(  9) = CLDS of NO
      IPR_OUTPUT( 10) = CHEM of NO
      IPR_OUTPUT( 11) = AERO of NO
      IPR_OUTPUT( 12) = PING of NO
      IPR_OUTPUT( 13) = XADV of VOC
      IPR_OUTPUT( 14) = YADV of VOC
      IPR_OUTPUT( 15) = ZADV of VOC
      IPR_OUTPUT( 16) = ADJC of VOC
      IPR_OUTPUT( 17) = HDIF of VOC
      IPR_OUTPUT( 18) = VDIF of VOC
      IPR_OUTPUT( 19) = EMIS of VOC
      IPR_OUTPUT( 20) = DDEP of VOC
      IPR_OUTPUT( 21) = CLDS of VOC
      IPR_OUTPUT( 22) = CHEM of VOC
      IPR_OUTPUT( 23) = AERO of VOC
      IPR_OUTPUT( 24) = PING of VOC
      IPR_OUTPUT( 25) = CHEM of HCHO
      IPR_OUTPUT( 26) = EMIS of HCHO


ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c                INTEGRATED REACTION RATE SECTION
c  The following section gives a symbolic representation of how the
c  PARTIAL IRR outputs are calculated given the commands that were
c  included in the command file.  The report includes pseudocode showing
c  how individual integrated reactions rates are combined to produce the
c  requested output.  In the pseudocode that follows
c
c       IRR< x > = the integrated reaction rate for
c                  the reaction with label x as
c                  defined in the chemical mechanism
c
c       IRROUT( n ) = the nth integrated reaction rate
c                     output that was requested
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
```

```
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Summary of Input IRR Commands for OUTPUT 1: OXprod
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The IRR_OUTPUT command was specified as follows:
c
c     IRROUT(  1) =          NETP [OX]
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced families are defined as follows:
c
c    OX=O3+NO2+2*NO3+O3P+O1D+PAN+HNO4+3*N2O5+TPAN+OLN
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced operators are defined as follows:
c
c..Net production of OX
c     NPRD(  1) =
c    &  +        IRR<  P5> ! HNO3+hv=HO+NO2
c    &  +        IRR< P21> ! ONIT+hv=.2*ALD+.8*KET+HO2+NO2
c    &  +        IRR<   9> ! HO2+NO=NO2+HO
c    &  + 2.0000 * IRR<  16> ! NO+NO+O2=2*NO2
c    &  + 2.0000 * IRR<  25> ! HO+HNO3=NO3
c    &  +        IRR<  50> ! PAN+HO=HCHO+NO3+XO2
c    &  +        IRR<  51> ! ONIT+HO=HC3P+NO2
c    &  +        IRR<  57> ! MO2+NO=HCHO+HO2+NO2
c    &  + 0.9640 * IRR<  58> ! HC3P+NO=.75*ALD+.25*KET+.09*HCHO
c                     !    +.036*ONIT+.964*NO2+.964*HO2
c    &  + 0.9200 * IRR<  60> ! HC5P+NO=.38*ALD+.69*KET+.08*ONIT+.92*NO2
c                     !    +.92*HO2
c    &  + 0.7600 * IRR<  62> ! HC8P+NO=.35*ALD+1.06*KET+.04*HCHO
c                     !    +.24*ONIT+.76*NO2+.76*HO2
c    &  +        IRR<  64> ! OL2P+NO=1.6*HCHO+HO2+NO2+.2*ALD
c    &  +        IRR<  65> ! OLTP+NO=ALD+HCHO+HO2+NO2
c    &  +        IRR<  66> ! OLIP+NO=HO2+1.45*ALD+.28*HCHO+.1*KET+NO2
c    &  +        IRR<  67> ! ACO3+NO=MO2+NO2
c    &  +        IRR<  68> ! TCO3+NO=NO2+.92*HO2+.89*GLY+.11*MGLY
c                     !    +.05*ACO3+.95*CO+2*XO2
c    &  +        IRR<  69> ! TOLP+NO=NO2+HO2+.17*MGLY+.16*GLY+.7*DCB
```

```
c  &  +        IRR<  70> ! XYLP+NO=NO2+HO2+.45*MGLY+.806*DCB
c  &  +        IRR<  71> ! ETHP+NO=ALD+HO2+NO2
c  &  +        IRR<  72> ! KETP+NO=MGLY+NO2+HO2
c  &  +        IRR<  73> ! OLN+NO=HCHO+ALD+2*NO2
c  &  +        IRR< 131> ! XO2+NO=NO2
c  &  + 0.9120 * IRR< IS2> ! ISO_RO+NO=.088*ONIT+.912*NO2+.912*HO2
c                  !    +.912*ISOPRO+.629*HCHO
c  &  + 1.2000 * IRR< IS9> ! ISON_R+NO=NO2+.8*ALD+.8*ONIT+.8*HO2
c                  !    +.2*ISOPRO+.2*NO2
c  &  +        IRR<IS14> ! IP_RO2+NO=NO2+HO2+.59*CO+.55*ALD+.25*HCH
c                  !    +.08*GLY+.34*MGLY+.63*KET
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Pseudocode for OUTPUT 1: OXprod
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
    IRROUT(  1) =         NPRD(  1)

ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Summary of Input IRR Commands for OUTPUT 2: Oxloss
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c The IRR_OUTPUT command was specified as follows:
c
c    IRROUT(  2) =         NETL [OX]
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c The referenced families are defined as follows:
c
c   OX=O3+NO2+2*NO3+O3P+O1D+PAN+HNO4+3*N2O5+TPAN+OLN
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c The referenced operators are defined as follows:
c
c..Net loss of OX
c   NLOS(  1) =
c  &  + 2.0000 * IRR<  P7> ! NO3+hv=NO
c  &  + 2.0000 * IRR<   2> ! O3P+NO2=NO
c  &  +        IRR<   5> ! O1D+H2O=2*HO
c  &  +        IRR<   7> ! O3+HO=HO2
```

```
c  & +        IRR<  8> ! O3+HO2=HO
c  & + 2.0000 * IRR< 19> ! NO3+NO2=NO+NO2
c  & + 2.0000 * IRR< 20> ! NO3+HO2=HNO3
c  & + 3.0000 * IRR< 23> ! N2O5+H2O=2*HNO3
c  & +        IRR< 24> ! HO+NO2=HNO3
c  & + 2.0000 * IRR< 74> ! HCHO+NO3=HO2+HNO3+CO
c  & + 2.0000 * IRR< 75> ! ALD+NO3=ACO3+HNO3
c  & + 2.0000 * IRR< 76> ! GLY+NO3=HNO3+HO2+2*CO
c  & + 2.0000 * IRR< 77> ! MGLY+NO3=HNO3+ACO3+CO
c  & + 2.0000 * IRR< 78> ! DCB+NO3=HNO3+TCO3
c  & + 2.0000 * IRR< 79> ! CSL+NO3=HNO3+XNO2+.5*CSL+.5*CSLAER
c  & +        IRR< 80> ! OL2+NO3=OLN
c  & +        IRR< 81> ! OLT+NO3=OLN
c  & +        IRR< 82> ! OLI+NO3=OLN+OLIAER
c  & +        IRR< 84> ! OL2+O3=HCHO+.4*ORA1+.42*CO+.12*HO2
c  & +        IRR< 85> ! OLT+O3=.53*HCHO+.5*ALD+.33*CO+.2*ORA1
c                  !    +.2*ORA2+.23*HO2+.22*MO2+.1*HO
c  & +        IRR< 86> ! OLI+O3=.18*HCHO+.72*ALD+.1*KET+.23*CO
c                  !    +.06*ORA1+.29*ORA2+.26*HO2+.14*HO
c                  !    +.31*MO2+OLIAER
c  & +        IRR< 101> ! HO2+OLN=ONIT
c  & +        IRR< 132> ! XNO2+NO2=ONIT
c  & +        IRR< IS6> ! ISO+O3=.6*HCHO+.65*ISOPRO+.39*ORA1+.27*H
c                  !    +.07*HO2+.07*CO+.2*XO2+.2*ACO3
c                  !    +.15*ALD
c  & +        IRR< IS7> ! ISO+O3P=.75*ISOPRO+.25*ACO3+.25*HCHO
c                  !    +.25*MO2
c  & + 2.0000 * IRR< IS8> ! ISO+NO3=ISON_R
c  & +        IRR<IS18> ! ISOPRO+O3=.268*HO+.1*HO2+.114*ACO3
c                  !    +.054*MO2+.07*XO2+.155*CO+.146*HCHO
c                  !    +.02*ALD+.01*GLY+.85*MGLY+.09*KET
c                  !    +.462*ORA1
c  & + 2.0000 * IRR<IS20> ! ISOPRO+NO3=.075*ACO3+.075*HNO3+.643*CO
c                  !    +.282*HCHO+.925*ONIT+.282*ALD
c                  !    +.925*HO2+.925*XO2
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Pseudocode for OUTPUT 2: Oxloss
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
    IRROUT( 2) =         NLOS( 1)
```

```
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Summary of Input IRR Commands for OUTPUT 3: newOH
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The IRR_OUTPUT command was specified as follows:
c
c    IRROUT(  3) =          PROD [HO] FROM [O3] AND [VOCA]
c    &           + 2.0000 * H2O2_OHcyc[POSONLY]
c    &           +          HNO3_OHcyc[POSONLY]
c    &           +          HONOcyc[NEGONLY]
c    &           +          OP1_OHcyc[POSONLY]
c    &           +          OP2_OHcyc[POSONLY]
c    &           +          PAA_OHcyc[POSONLY]
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced families are defined as follows:
c
c   VOCA=OL2+OLI+OLT+ISO
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced cycles are defined as follows:
c
c..CYCLE 1 HONOcyc
c    CYSM(  1) =
c    & -         IRR<  P4> ! HONO+hv=HO+NO
c    & +         IRR<  15> ! NO+HO=HONO
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced reaction sums are defined as follows:
c
c..RXSUM 1 H2O2_OHcyc
c    RXSM(  1) =
c    & +         IRR<   9> ! HO2+NO=NO2+HO
c    & -         IRR<  12> ! HO2+HO2=H2O2
c    & -         IRR<  13> ! HO2+HO2+H2O=H2O2
c
c..RXSUM 2 HNO3_OHcyc
c    RXSM(  2) =
```

```
c   & +        IRR< P5> ! HNO3+hv=HO+NO2
c   & -        IRR< 24> ! HO+NO2=HNO3
c
c..RXSUM 3 OP1_OHcyc
c    RXSM( 3) =
c   & +        IRR< P13> ! OP1+hv=HCHO+HO2+HO
c   & -        IRR< 88> ! HO2+MO2=OP1
c
c..RXSUM 4 OP2_OHcyc
c    RXSM( 4) =
c   & +        IRR< P14> ! OP2+hv=ALD+HO2+HO
c   & -        IRR< 89> ! HO2+ETHP=OP2
c   & -        IRR< 90> ! HO2+HC3P=OP2
c   & -        IRR< 91> ! HO2+HC5P=OP2
c   & -        IRR< 92> ! HO2+HC8P=OP2
c   & -        IRR< 93> ! HO2+OL2P=OP2
c   & -        IRR< 94> ! HO2+OLTP=OP2
c   & -        IRR< 95> ! HO2+OLIP=OP2
c   & -        IRR< 96> ! HO2+KETP=OP2
c   & -        IRR< 97> ! HO2+ACO3=PAA
c   & -        IRR< 98> ! HO2+TOLP=OP2
c   & -        IRR< 99> ! HO2+XYLP=OP2
c   & -        IRR< 100> ! HO2+TCO3=OP2
c   & -        IRR< 101> ! HO2+OLN=ONIT
c
c..RXSUM 5 PAA_OHcyc
c    RXSM( 5) =
c   & +        IRR< 15> ! NO+HO=HONO
c   & -        IRR< 97> ! HO2+ACO3=PAA
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced operators are defined as follows:
c
c..Production of HO from O3 and VOCA
c    PROD( 1) =
c   & + 0.1000 * IRR< 85> ! OLT+O3=.53*HCHO+.5*ALD+.33*CO+.2*ORA1
c                !    +.2*ORA2+.23*HO2+.22*MO2+.1*HO
c   & + 0.1400 * IRR< 86> ! OLI+O3=.18*HCHO+.72*ALD+.1*KET+.23*CO
c                !    +.06*ORA1+.29*ORA2+.26*HO2+.14*HO
c                !    +.31*MO2+OLIAER
```

```
c   &  +  0.2700 * IRR< IS6> ! ISO+O3=.6*HCHO+.65*ISOPRO+.39*ORA1+.27*H
c                    !     +.07*HO2+.07*CO+.2*XO2+.2*ACO3
c                    !     +.15*ALD
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Pseudocode for OUTPUT 3: newOH
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      IRROUT(  3) = 0.0

      IF( RXSM(  1) .GT. 0.0 ) THEN
        IRROUT(  3) = IRROUT(  3) +  2.0000 * RXSM(  1)
      ENDIF

      IF( RXSM(  2) .GT. 0.0 ) THEN
        IRROUT(  3) = IRROUT(  3) + RXSM(  2)
      ENDIF

      IF( CYSM(  1) .LT. 0.0 ) THEN
        IRROUT(  3) = IRROUT(  3) + ABS(CYSM(  1))
      ENDIF

      IF( RXSM(  3) .GT. 0.0 ) THEN
        IRROUT(  3) = IRROUT(  3) + RXSM(  3)
      ENDIF

      IF( RXSM(  4) .GT. 0.0 ) THEN
        IRROUT(  3) = IRROUT(  3) + RXSM(  4)
      ENDIF

      IF( RXSM(  5) .GT. 0.0 ) THEN
        IRROUT(  3) = IRROUT(  3) + RXSM(  5)
      ENDIF

      IRROUT(  3) = IRROUT(  3)
     &          +         PROD(  1)


ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Summary of Input IRR Commands for OUTPUT 4: NO2fromHO2
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
```

```
c  The IRR_OUTPUT command was specified as follows:
c
c     IRROUT(  4) =          PROD [NO2] FROM [HO2]
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced operators are defined as follows:
c
c..Production of NO2 from HO2
c     PROD(  1) =
c    &  +         IRR<   9> ! HO2+NO=NO2+HO
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Pseudocode for OUTPUT 4: NO2fromHO2
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
     IRROUT(  4) =          PROD(  1)

ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Summary of Input IRR Commands for OUTPUT 5: netONIT
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The IRR_OUTPUT command was specified as follows:
c
c     IRROUT(  5) =          NET  [ONIT]
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c  The referenced operators are defined as follows:
c
c..Net reaction of ONIT
c     NETR(  1) =
c    &  -         IRR< P21> ! ONIT+hv=.2*ALD+.8*KET+HO2+NO2
c    &  -         IRR<  51> ! ONIT+HO=HC3P+NO2
c    &  + 0.0360 * IRR<  58> ! HC3P+NO=.75*ALD+.25*KET+.09*HCHO
c                      !     +.036*ONIT+.964*NO2+.964*HO2
c    &  + 0.0800 * IRR<  60> ! HC5P+NO=.38*ALD+.69*KET+.08*ONIT+.92*NO2
c                      !     +.92*HO2
c    &  + 0.2400 * IRR<  62> ! HC8P+NO=.35*ALD+1.06*KET+.04*HCHO
c                      !     +.24*ONIT+.76*NO2+.76*HO2
c    &  +         IRR< 101> ! HO2+OLN=ONIT
c    &  +         IRR< 132> ! XNO2+NO2=ONIT
```

```
c  & + 0.0880 * IRR< IS2> ! ISO_RO+NO=.088*ONIT+.912*NO2+.912*HO2
c                  !    +.912*ISOPRO+.629*HCHO
c  & + 0.8000 * IRR< IS9> ! ISON_R+NO=NO2+.8*ALD+.8*ONIT+.8*HO2
c                  !    +.2*ISOPRO+.2*NO2
c  & +        IRR<IS10> ! ISON_R+HO2=ONIT
c  & +        IRR<IS11> ! ISON_R+ACO3=.5*HO2+.5*MO2+.5*ORA2+ALD
c                  !    +ONIT
c  & +        IRR<IS12> ! ISON_R+MO2=.5*HCHO+.5*HO2+ALD+ONIT
c  & + 0.9250 * IRR<IS20> ! ISOPRO+NO3=.075*ACO3+.075*HNO3+.643*CO
c                  !    +.282*HCHO+.925*ONIT+.282*ALD
c                  !    +.925*HO2+.925*XO2
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Pseudocode for OUTPUT 5: netONIT
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
    IRROUT( 5) =         NETR( 1)

ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Summary of Input IRR Commands for OUTPUT 6: lossOH_HC
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c The IRR_OUTPUT command was specified as follows:
c
c    IRROUT( 6) =         LOSS [VOC] AND [HO]
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c The referenced families are defined as follows:
c
c VOC=CO+ETH+HC3+HC5+HC8+OL2+OLT+OLI+ISO+TOL+CSL+XYL+HCHO+ALD+KET

c      +GLY+MGLY+DCB
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c
c The referenced operators are defined as follows:
c
c..Loss of VOC and HO
c   LOSS( 1) =
c  & +        IRR< 29> ! CO+HO=HO2
c  & +        IRR< 31> ! ETH+HO=ETHP
```

```
c   & +        IRR< 32> ! HC3+HO=.83*HC3P+.17*HO2+.009*HCHO
c                   !    +.075*ALD+.025*KET
c   & +        IRR< 33> ! HC5+HO=HC5P+.25*XO2
c   & +        IRR< 34> ! HC8+HO=HC8P+.75*XO2+HC8AER
c   & +        IRR< 35> ! OL2+HO=OL2P
c   & +        IRR< 36> ! OLT+HO=OLTP
c   & +        IRR< 37> ! OLI+HO=OLIP+OLIAER
c   & +        IRR< 38> ! TOL+HO=.75*TOLP+.25*CSL+.25*HO2+TOLAER
c   & +        IRR< 39> ! XYL+HO=.83*XYLP+.17*CSL+.17*HO2+XYLAER
c   & +        IRR< 40> ! CSL+HO=.1*HO2+.9*XO2+.9*TCO3+CSLAER
c   & +        IRR<40a> ! CSL+HO=CSL
c   & +        IRR< 41> ! HCHO+HO=HO2+CO
c   & +        IRR< 42> ! ALD+HO=ACO3
c   & +        IRR< 43> ! KET+HO=KETP
c   & +        IRR< 44> ! GLY+HO=HO2+2*CO
c   & +        IRR< 45> ! MGLY+HO=ACO3+CO
c   & +        IRR< 46> ! DCB+HO=TCO3
c   & +        IRR<IS1> ! ISO+HO=ISO_RO+.079*XO2
c
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c  Pseudocode for OUTPUT 6: lossOH_HC
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
    IRROUT( 6) =         LOSS( 1)
```

********* END OF PROCESS ANALYSIS CONTROL PROGRAM OUTPUT REPORT ********

APPENDIX N

Copyrights and Trademarks

Operating System - Solaris

---

which license also covers Sun's licensees who implement OPEN LOOK GUI's and otherwise comply with Sun's written license agreements.

X Window System is a trademark of X Consortium, Inc.

Operating System - Digital Unix

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

The following are trademarks of Digital Equipment Corporation:

ALL-IN-1, Alpha AXP, AlphaGeneration, AlphaServer, AlphaStation, AXP, Bookreader, CDA, DDIS, DEC, DEC Ada, DEC Fortran, DEC FUSE, DECnet, DECstation, DECssytem, DECterm, DECUS, DECwindows, DTIF, MASBUS, MicroVAX, OpenVMS, POLYCENTER, Q-bus, StorageWorks, TruCluster, TURBOchannel, ULTRIX, ULTRIX Mail Connection, ULTRIX Worksystem Software, UNIBUS, VAX, VAXstation, VMS, XUI, and the DIGITAL logo.

COMPAQ is a registered trademark of Compaq Computer Corporation. Java and Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. Netscape Navigator is a trademark of Netscape Communications Corporation. NFS is a registered trademark of Sun Microsystems, Inc. ONC is a trademark of Sun Microsystems, Inc. Open Software Foundation, OSF, OSF/1, OSF/Motif, and Motif are trademarks of the Open Software Foundation, Inc. Adobe, PostScript, and Display PostScript are registered trademarks of Adobe Systems, Inc. Proteon is a trademark of Proteon Associates, Inc. UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Ltd. X/Open is a trademark of X/Open Company Ltd.

SoftWindows

Database ObjectStore

ObjectStore Release 4.0.2.0.0 for SPARC Solaris 2.4/SunPro (SunOS 5.4)

This tape or disk and the program recorded on it are the property of Object Design Inc. Use of the programs (including examination, reproduction, modification, translation and disclosure) and transfer of the programs and the right to use the programs are all prohibited except by the express written authorization of Object Design, Inc.

"RESTRICTED RIGHTS LEGEND -- Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at 252.227-7013 (48 CFR, Ch. 2). Contractor/manufacturer is Object Design, Inc., 25 Burlington Mall Road, Burlington, MA 01803."

GUI Galaxy

License, Trademark, and Copyright Information

License Information

You have access to the Galaxy Application Environment (TM) software and documentation pursuant to the terms of a Software License Agreement granted by Visix Software Inc. As a user of this software and documentation, you are bound by the terms of the Software License Agreement. All rights, title, and interest to this software remain with Visix Software Inc.

Use, reproduction, and disclosure of the software and documentation by the U.S. Government is subject to restrictions set forth in (i) FAR 52.227-14, (ii) FAR 52.227-19, (iii) DFAR 252.227-7013, or (iv) the Software License Agreement, as applicable, for limited rights technical data and restricted rights computer software. For purposes of the FAR, the software shall be deemed "unpublished" and licensed with disclosure prohibitions. Contractor/subcontractor: Visix Software Inc., 11440 Commerce Park Drive, Reston, VA 20191.

Trademark Acknowledgments

Galaxy Application Environment is a registered trademark and Vibe is a trademark of Visix Software Inc. Alpha, DEC, DECnet, and Digital are registered trademarks of Digital Equipment Corporation. The Graphics Interchange Format is a copyright property and GIF is a service mark of CompuServe Incorporated. Helvetica is a registered trademark of Linotype Corporation. Macintosh is a registered trademark of Apple Computer, Inc. Microsoft is a registered trademark and Microsoft Windows, Windows NT, and ActiveX are trademarks of Microsoft Corporation. IRIX and Silicon Graphics are trademarks of Silicon Graphics, Inc. Motif and Open Software Foundation are trademarks of the Open Software Foundation, Inc. OPEN LOOK is a registered trademark of Novell, Inc. OS/2, AIX, and IBM are registered trademarks of International Business Machines Corporation. PostScript and Display Postscript are registered trademarks of

Adobe Systems Incorporated. The PBM, PGM, PPM, and XBM image formats are owned and used by permission of Massachusetts Institute of Technology. X Window System is a trademark of X Consortium Inc. Sun, Java, Solaris, SunOS, and OpenWindows are trademarks or registered trademarks of Sun Microsystems, Inc. Unicode is a trademark of Unicode, Inc. UNIX and XWindow System are registered trademarks licensed exclusively by X/Open Company Ltd. Sybase is a registered trademark of Sybase, Inc. Oracle is a registered trademark of Oracle Corporation.

Any other products or services mentioned or depicted in this document are identified by the trademarks or service marks of their respective companies or organizations.

The information contained herein is subject to change without notice. Therefore, Visix Software Inc. assumes no responsibility for the accuracy of the information presented in this document beyond its current release date.

Orbix

Orbix is a Registered Trademark of IONA Technologies Ltd.

While the information in this publication is believed to be accurate, IONA Technologies Ltd makes no warranty of any kind to this material including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. IONA Technologies Ltd shall not be liable for errors contained herein, or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Netscape

BY CLICKING THE ACCEPTANCE BUTTON OR INSTALLING THE SOFTWARE, YOU ARE CONSENTING TO BE BOUND BY AND ARE BECOMING A PARTY TO THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, CLICK THE BUTTON THAT INDICATES YOU DO NOT ACCEPT THE TERMS AND DO NOT INSTALL THE SOFTWARE. (IF APPLICABLE, YOU MAY RETURN THE PRODUCT TO THE PLACE OF PURCHASE FOR A FULL REFUND.) IF THE COPY OF THE SOFTWARE YOU RECEIVED WAS ACCOMPANIED BY A PRINTED OR OTHER FORM OF "HARD-COPY" END USER LICENSE AGREEMENT WHOSE TERMS VARY FROM THIS AGREEMENT, THEN THE HARD-COPY END USER LICENSE AGREEMENT GOVERNS YOUR USE OF THE SOFTWARE.

NETSCAPE CLIENT SOFTWARE END USER LICENSE AGREEMENT
REDISTRIBUTION OR RENTAL NOT PERMITTED.

This Agreement sets forth the terms and conditions of your use of the accompanying Netscape client software product(s) (the "Software"). Any third party software that is provided with the Software with such third party's license agreement (in either electronic or printed form) is included for use at your option. If you choose to use such software, then such use shall be governed by such third party's license and not by this Agreement.

This Agreement has 3 parts. Part I applies if you have a free of charge license to the Software.

Part II applies if you have purchased a license to the Software. Part III applies to all license grants. If you initially acquired a copy of the Software free of charge and you wish to purchase a license, contact Netscape Communications Corporation ("Netscape") on the Internet at http://home.netscape.com. As used in this Agreement, for residents of Europe, the Middle East and Africa, "Netscape" shall refer to Netscape Communications Ireland Limited; for residents of Japan, "Netscape" shall refer to Netscape Communications (Japan), Ltd.; for residents of all other countries, "Netscape" shall refer to Netscape Communications Corporation. For purposes of this Agreement, "Licensor" shall be defined as follows: If you have acquired a third party product or service and such product or service included the Software, then such third party shall be the Licensor. Otherwise, Netscape shall be the Licensor.

PART I - TERMS APPLICABLE WHEN LICENSE FEES NOT (YET) PAID (LIMITED TO EVALUATION, EDUCATIONAL AND NON-PROFIT USE).

LICENSE GRANT. Licensor grants you a non-exclusive license to use the Software free of charge if (a) you are a student, faculty member or staff member of an educational institution (K-12, junior college, college or university, or the international equivalent, or a library), a staff member of a religious organization or an employee of an organization which meets Licensor's criteria for a charitable non-profit organization; or (b) your use of the Software is for the purpose of evaluating whether to purchase an ongoing license to the Software. The evaluation period for use by or on behalf of a commercial entity is limited to ninety (90) days; evaluation use by others is not subject to this ninety (90) day limit. Government agencies (other than public libraries) are not considered educational, religious or charitable non-profit organizations for purposes of this Agreement. If you are using the Software free of charge, you are not entitled to hard-copy documentation, support or telephone assistance. If you fit within the description above, you may use the Software in the manner described in Part III below under "Scope of Grant."

DISCLAIMER OF WARRANTY. Free of charge Software is provided on an "AS IS" basis, without warranty of any kind, including without limitation the warranties that the Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Software is borne by you. Should the Software prove defective in any respect, you and not Licensor or its suppliers assume the entire cost of any service and repair. In addition, the security mechanisms implemented by the Software have inherent limitations, and you must determine that the Software sufficiently meets your requirements. This disclaimer of warranty constitutes an essential part of this Agreement. No use of the Software without payment of license fees to Licensor is authorized hereunder except under this Disclaimer.

PART II - TERMS APPLICABLE WHEN LICENSE FEES PAID

LICENSE GRANT. Subject to payment of applicable license fees, Licensor grants to you a non-exclusive license to use the Software and accompanying documentation ("Documentation") in the manner described in Part III below under "Scope of Grant."

LIMITED WARRANTY. Licensor warrants that for a period of ninety (90) days from the date of acquisition, the Software, if operated as directed, will substantially achieve the functionality described in the Documentation. Licensor does not warrant, however, that your use of the Software will be uninterrupted or that the operation of the Software will be error-free or secure. In addition, the security mechanisms implemented by the Software have inherent limitations, and you must determine that the Software sufficiently meets your requirements. Licensor also warrants that the media containing the Software, if provided by Licensor, is free from defects in material and workmanship and will so remain for ninety (90) days from the date you acquired the Software. Licensor's sole liability for any breach of this warranty shall be, in Licensor's sole discretion: (i) to replace your defective media or Software; or (ii) to advise you how to achieve substantially the same functionality with the Software as described in the Documentation through a procedure different from that set forth in the Documentation; or (iii) if the above remedies are impracticable, to refund the license fee you paid for the Software. Repaired, corrected, or replaced Software and Documentation shall be covered by this limited warranty for the period remaining under the warranty that covered the original Software, or if longer, for thirty (30) days after the date (a) of delivery to you of the repaired or replaced Software, or (b) Licensor advised you how to operate the Software so as to achieve substantially the same functionality described in the Documentation.

Only if you inform Licensor of your problem with the Software during the applicable warranty period and provide evidence of the date you purchased a license to the Software will Licensor be obligated to honor this warranty. Licensor will use reasonable commercial efforts to repair, replace, advise or, for individual consumers, refund pursuant to the foregoing warranty within thirty (30) days of being so notified.

If any modifications are made to the Software by you during the warranty period; if the media is subjected to accident, abuse, or improper use; or if you violate the terms of this Agreement, then this warranty shall immediately terminate. Moreover, this warranty shall not apply if the Software is used on or in conjunction with hardware or software other than the unmodified version of hardware and software with which the Software was designed to be used as described in the Documentation.

THIS IS A LIMITED WARRANTY, AND IT IS THE ONLY WARRANTY MADE BY LICENSOR OR ITS SUPPLIERS. LICENSOR MAKES NO OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTIES' RIGHTS. YOU MAY HAVE OTHER STATUTORY RIGHTS. HOWEVER, TO THE FULL EXTENT PERMITTED BY LAW, THE DURATION OF STATUTORILY REQUIRED WARRANTIES, IF ANY, SHALL BE LIMITED TO THE ABOVE LIMITED WARRANTY PERIOD. MOREOVER, IN NO EVENT WILL WARRANTIES PROVIDED BY LAW, IF ANY, APPLY UNLESS THEY ARE REQUIRED TO APPLY BY STATUTE NOTWITHSTANDING THEIR EXCLUSION BY CONTRACT. NO DEALER, AGENT, OR EMPLOYEE OF LICENSOR IS AUTHORIZED TO

MAKE ANY MODIFICATIONS, EXTENSIONS, OR ADDITIONS TO THIS LIMITED WARRANTY.

PART III - TERMS APPLICABLE TO ALL LICENSE GRANTS

SCOPE OF LICENSE GRANT.

You may:

* use the Software on any single computer;

* use the Software on a second computer so long as only one (1) copy is used at a time;

* use the Software on a network, provided that a licensed copy of the Software has been acquired for each person permitted to access the Software through the network;

* make a single copy the Software for archival purposes, provided the copy contains all of the original Software's proprietary notices; or

* if you have purchased a license for multiple copies of the Software, make the total number of copies of Software (but not the Documentation) stated on the packing slip(s), invoice(s), or Certificate(s) of Authenticity, provided any copy must contain all of the original Software's proprietary notices. The number of copies on the packing slip(s), invoice(s), or Certificate(s) of Authenticity is the total number of copies that may be made for all platforms. Additional copies of Documentation may be purchased from Licensor.

You may not:

* permit other individuals to use the Software except under the terms listed above;

* permit concurrent use of the Software;

* modify, translate, reverse engineer, decompile, disassemble (except and solely to the extent an applicable statute expressly and specifically prohibits such restrictions), or create derivative works based on the Software;

* copy the Software other than as specified above;

* rent, lease, grant a security interest in, or otherwise transfer rights to the Software; or

* remove any proprietary notices or labels on the Software.

ENCRYPTION. If the Software contains cryptographic features, then you may wish to obtain a

signed digital certificate from a certificate authority or a certificate server in order to utilize certain of the cryptographic features. You may be charged additional fees for certification services. You are responsible for maintaining the security of the environment in which the Software is used and the integrity of the private key file used with the Software. In addition, the use of digital certificates is subject to the terms specified by the certificate provider, and there are inherent limitations in the capabilities of digital certificates. If you are sending or receiving digital certificates, you are responsible for familiarizing yourself with and evaluating such terms and limitations. If the Software is a Netscape product with Fortezza, you will also need to obtain PC Card Readers and Fortezza Crypto Cards to enable the Fortezza features.

TITLE. Title, ownership rights, and intellectual property rights in the Software shall remain in Netscape and/or its suppliers. The Software is protected by copyright and other intellectual property laws and by international treaties. Title and related rights in the content accessed through the Software is the property of the applicable content owner and is protected by applicable law. The license granted under this Agreement gives you no rights to such content.

TERMINATION. This Agreement and the license granted hereunder will terminate automatically if you fail to comply with the limitations described herein. Upon termination, you must destroy all copies of the Software and Documentation.

EXPORT CONTROLS. None of the Software or underlying information or technology may be downloaded or otherwise exported or reexported (i) into (or to a national or resident of) Cuba, Iraq, Libya, Sudan, North Korea, Iran, Syria or any other country to which the U.S. has embargoed goods; or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nationals or the U.S. Commerce Department's Table of Denial Orders. By downloading or using the Software, you are agreeing to the foregoing and you are representing and warranting that you are not located in, under the control of, or a national or resident of any such country or on any such list. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Software.

If the Software is identified as a not-for-export product (for example, on the box, media or in the installation process), then, unless you have an exemption from the United States Department of State, the following applies: EXCEPT FOR EXPORT TO CANADA FOR USE IN CANADA BY CANADIAN CITIZENS, THE SOFTWARE AND ANY UNDERLYING TECHNOLOGY MAY NOT BE EXPORTED OUTSIDE THE UNITED STATES OR TO ANY FOREIGN ENTITY OR "FOREIGN PERSON" AS DEFINED BY U.S. GOVERNMENT REGULATIONS, INCLUDING WITHOUT LIMITATION, ANYONE WHO IS NOT A CITIZEN, NATIONAL OR LAWFUL PERMANENT RESIDENT OF THE UNITED STATES. BY DOWNLOADING OR USING THE SOFTWARE, YOU ARE AGREEING TO THE FOREGOING AND YOU ARE WARRANTING THAT YOU ARE NOT A "FOREIGN PERSON" OR UNDER THE CONTROL OF A "FOREIGN PERSON."

LIMITATION OF LIABILITY. UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL

THEORY, TORT, CONTRACT, OR OTHERWISE, SHALL LICENSOR OR ITS SUPPLIERS OR RESELLERS BE LIABLE TO YOU OR ANY OTHER PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES. IN NO EVENT WILL LICENSOR BE LIABLE FOR ANY DAMAGES IN EXCESS OF THE AMOUNT LICENSOR RECEIVED FROM YOU FOR A LICENSE TO THE SOFTWARE, EVEN IF LICENSOR SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY THIRD PARTY. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM LICENSOR'S NEGLIGENCE. TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

HIGH RISK ACTIVITIES. The Software is not fault-tolerant and is not designed, manufactured or intended for use or resale as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the Software could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). Accordingly, Licensor and its suppliers specifically disclaim any express or implied warranty of fitness for High Risk Activities.

MISCELLANEOUS. This Agreement represents the complete agreement concerning the license granted hereunder and may be amended only by a writing executed by both parties. THE ACCEPTANCE OF ANY PURCHASE ORDER PLACED BY YOU IS EXPRESSLY MADE CONDITIONAL ON YOUR ASSENT TO THE TERMS SET FORTH HEREIN, AND NOT THOSE IN YOUR PURCHASE ORDER. If any provision of this Agreement is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This Agreement shall be governed by California law, excluding conflict of law provisions (except to the extent applicable law, if any, provides otherwise). The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded.

U.S. GOVERNMENT END USERS. The Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire the Software with only those rights set forth herein.

ARC/INFO

RESTRICTED/LIMITED RIGHTS LEGEND
U.S. Government Restricted/Limited Rights: Any software, documentation, and/or data delivered hereunder is subject to the terms of the License Agreement. In no event shall the Government acquire greater than RESTRICTED/LIMITED RIGHTS. At a minimum, use, duplication or disclosure by the Government is subject to restrictions as set forth in FAR 12.211/12.212 [Commercial Technical Data/Computer Software]; DFARS 252.227-7015 (NOV 1995) [Technical Data]; and/or DFARS 227.7202 [Computer Software], as applicable. Contractor/Manufacturer is Environmental Systems Research Institute, Inc., 380 New York Street, Redlands, CA 92373-8100 USA.

ARC Version 7.0.4 (Sun Jan 21 22:27:17 PST 1996)

This software is provided with RESTRICTED AND LIMITED RIGHTS. Use, duplication, or disclosure by the Government is subject to the restrictions as set forth in FAR 52.227-14 (JUN 1987) Alternate III (g)(3) (JUN 1987), FAR 52.227-19 (JUN 1987), or DFARS 552.227-7013 (c)(1)(ii) (OCT 1988), as applicable. Contractor/Manufacturer is Environmental Systems Research Institute, Inc. (ESRI) 380 New York St. Redlands, CA 92373.

SAS

RESTRICTED RIGHTS LEGEND
Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in

subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina, USA 27513.

Sun Fortran Compiler

The OPEN LOOK (R) and Sun (TM) Graphical User Interfaces were developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUI's and otherwise comply with Sun's written license agreements.

X Window System is a trademark and product of the Massachusetts Institute of Technology.

Some of the material in this manual is based on the Bell Laboratories document entitled "A Portable Fortan 77 Compiler," by S.I. Feldman and P.J. Weinberger, dated 1 August 1987. Material on the I/O Library is derived from the paper entitled "Introduction to the f77 I/O Library", by David L. Wasley, University of California, Berkeley, California 94720. Further work was done at Sun Microsystems.

Digital Fortran Compiler for Windows

First Printing, June 1994
Revised, April 1995
Revised, April 1997

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication [i.e., Digital Fortran Compiler for Windows] will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

The following are third-party trademarks:

CRAY is a registered trademark of Cray Research, Inc.
IBM is a registered trademark of International Business Machines, Inc.
IEEE is a registered trademark of the Institute of Electrical and Electronics Engineers, Inc.
Intel and Pentium are registered trademarks of Intel Corporation.
Microsoft, MS, MS-DOS, Windows, and Windows 95 are registered trademarks, and NT and WindowsNT are trademarks of Microsoft Corporation.
OSF and OSF/1 are registered trademarks of the Open Software Foundation, Inc.
UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Ltd.

All other trademarks and registered trademarks are the property of their respective holders.

Digital Fortran Compiler For Alpha

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

The following are trademarks of Digital Equipment Corporation: Alpha AXP, AXP, BASEstar, Bookreader, DEC, DEC Ada, DEC C++, DEC C, DEC COBOL, DEC Fortran, DEC FUSE, DEC GKS, DEC MAILbus, DEC MAILworks, DECmessageQ, DECmigrate, DECnet, DEC Open3D, DEC Pascal, DEC PHIGS, DECserver, DEC SNA, DEC SNA Domain Gateway, DECtalk, DECwindows, DIGITAL, HUBwatch, MAILbus, MailWorks, MUXserver, NAS, ObjectBroker, OpenVMS, PATHWORKS, POLYCENTER< PrintServer, Reliable Transaction Router, RRD42, StorageWorks, VMS, and the DIGITAL logo.

The following are third-party trademarks:
PostScript is a registered trademark of Adobe Systems, Incorporated.
Modicon is registered trademark of Modicon, Inc.
Motif and OSF/1 are registered trademarks of the Open Software Foundation, Inc.
TUXEDO is a registered trademark of UNIX System Laboratories Inc.
Unitree is a trademark of General Atomics, licensed to Digital Equipment Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Ltd.

All other trademarks are registered trademarks are the property of their respective holders.

CRAY Fortran Compiler

CRAY®, CRAY-1®, Cray Ada®, CRAY Y-MP®, HSX®, SSD®, UNICOS® and X-MP EA® are federally registered trademarks and Autotasking™, CF77™, CFT™, CFT2™, CFT77™, COL™, CRAY X-MP™, CRAY Y-MP2E™, CRAY-2™, CSIM™, Delivering the power...™, IOS™, OLNET™, RQS™, SEGLDR™, and SUPERLINK™ are trademarks of Cray Research Inc.

UNIX is a trademark of AT&T. VMEbus is a trademark of Motorola, Inc.

The UNICOS operating system is derived from the AT&T UNIX Operating System V operating system. UNICOS is also based in part on the Fourth Berkeley Software Distribution under license from The Regents of the University of California.

Operating System - CRAY UNICOS

CRAY, CRAY-1, SSD, and UNICOS are registered trademarks and CFT, CFT77, CFT2, COS, CRAY-2, CRAY X-MP, CRAY X-MP EA, CRAY Y-MP, CSIM, HSX, IOS, SEGLDR, and SUPERLINK are trademarks of Cray Research, Inc.

3B20 is a trademark of AT&T. Apollo and DOMAIN are registered trademarks of Apollo Computer Inc. CDC is a registered trademark of Control Data Corporation. DEC, VAX, and VMS are trademarks of Digital Equipment Corporation. IBM is a registered trademark of International Business Machines Corporation. IRIS is a trademark of Silicon Graphics, Inc. Pyramid is a trademark of Pyramid Technology corporation. Sun Microsystems is a trademark

of Sun Microsystems, Inc. UNIX is a trademark of AT&T.

The TCP-IP documentation is copyrighted by the Wollongong Group and may not be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, except as provided in the license agreement governing the documentation or by written permission of The Wollongong Group, Inc., 1129 San Antonio Road, Palo Alto, California, 943403. The Wollongong software and documentation is based in part on the Fourth Berkeley Software Distribution under license from The Regents of the University of California. ©The Wollongong Group, Inc., 1985.

The UNICOS operating system is derived from the AT&T UNIX Operating System V operating system. UNICOS is also based in part on the Fourth Berkeley Software Distribution under license from The Regents of the University of California.

Operating System - SGI IRIX

Restricted Rights Legend

Use, duplication or disclosure of the technical data contained in this document by the Government is subject to restrictions as set forth in subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 52.227-7013, and/or in similar or successor clauses in the FAR, or the DOD or NASA FAR Supplement. Unpublished rights reserved under the Copyright Laws of the United States. Contractor/manufacturer is Silicon Graphics, inc., 2011 N. Shoreline Blvd., Mountain View, CA 94039-7311.

Silicon Graphics and IRIS are registered trademarks and IRIX, Personal IRIS, and Efast are trademarks of Silicon Graphics, Inc. Ethernet is a registered trademark of the Xerox Corporation. Motif is a trademark of Open Software foundation. NFS is a trademark of Sun Microsystems, Inc. UNIX is a registered trademark of UNIX System Laboratories

---

Operating System - Windows NT

IBM Data Explorer

(See **http://www/almaden.ibm.con/dx/DXDLLicense.html**)

DX trial version license agreement

PLEASE READ THIS AGREEMENT CAREFULLY BEFORE DOWNLOADING, INSTALLING OR USING THE SOFTWARE WHICH ACCOMPANIES THIS AGREEMENT. BY DOWNLOADING THE SOFTWARE FROM THE INTERNET, INSTALLING IT ON ANY COMPUTER OR USING THIS SOFTWARE, YOU INDICATE YOUR ACCEPTANCE OF THESE TERMS AND CONDITIONS.

1. GRANT OF LICENSE

Subject to all of the terms and conditions of this license agreement ("Agreement"), International Business Machines Corporation ("IBM") grants you, for no charge, a limited, non-exclusive,

non-transferable right to execute, display and perform this copy of IBM Visualization Data Explorer Version 3.1.x Trial Version ("Software") and accompanying soft-copy documentation ("Documentation") in the country in which you acquired or downloaded the Software and Documentation from the Internet. You may (1) download the Software and Documentation to only one machine, (2) use the Software solely for the purposes of testing and evaluating its performance, (3) copy the Software and Documentation for backup purposes only and (4) modify the Software only as permitted in the Documentation, to evaluate its functionality. You obtain no rights other than those granted to you under this license. The Software is a trial version of IBM Visualization Data Explorer (TM) for SMP Version 3.1.x (product #5765-587), IBM Visualization Data Explorer Version 3.1.x (product #5765-586), and IBM Visualization Data Explorer Personal Edition Version 3.1.x (product #5799-GBX), three Data Explorer products with similar functions (collectively "Data Explorer"). IBM grants you this license to allow you to determine whether Data Explorer may be useful to you. You may not use the Software or Documentation for any other purpose, and you may not distribute any part of it, either alone or with any other software products which you may distribute.

2. COPYRIGHT

The Software and Documentation are protected under the following copyright: Copyright (c) 1991-1997 by International Business Machines Corporation.
YOU MUST REPRODUCE THE COPYRIGHT NOTICE AND ANY OTHER LEGEND OF OWNERSHIP ON EACH COPY OR PARTIAL COPY OF THE SOFTWARE OR DOCUMENTATION YOU CREATE.

You may be held legally responsible for any copyright infringement that is caused or encouraged by your failure to abide by the terms of this Agreement or any applicable laws. The Software and Documentation are licensed and not sold to you. IBM retains title to the Software and Documentation and to any copies you make of them.

3. TRIAL VERSION

Downloading and installing the Software will give you immediate access to certain limited functions of the Software, which will be available to you for five minutes at a time. The Software contains a time-disabling device that will stop the Software from running at the end of five minutes. NO DATA WILL BE SAVED AT ANY TIME DURING OR AFTER THE END OF THE SESSION. Therefore you should take precautions to avoid losing any data.

You may also sample all of the functions of the Software for a thirty day period. Download and install the Software on a workstation where you would like to evaluate its functionality. At the end of the installation process, a unique CPU identifier corresponding to that workstation will appear on your screen. Send an e-mail request to IBM at "ibmdx@watson.ibm.com", with the CPU identifier, and IBM will e-mail you an enabling key that will function only on that workstation. YOUR ENABLING KEY WILL STOP FUNCTIONING THIRTY DAYS FROM

THE DATE ON WHICH IT WAS ISSUED, at which time only limited functions of the Software will be available to you, for five minutes at a time. At the end of the thirty day period, you will no longer be able to modify or add to the data which you have input during the thirty day period.

No hard-copy documentation or telephone assistance is available in connection with this free trial.

4. USE RESTRICTIONS

You agree NOT to do any of the following: (1) use, copy or modify the Software and Documentation, except as provided in this license; (2) reverse assemble, reverse compile or otherwise translate the Software; (3) transfer this license to anyone; (4) sublicense, rent, lease or assign this license to the Software and Documentation or (5) transmit or otherwise distribute the Software and Documentation to anyone.

5. GOVERNING LAW

If you acquired the Software and Documentation in the U.S. or downloaded them from the Internet to a computer in the U.S., this Agreement is governed by the laws of the State of New York. If you acquired the Software and Documentation in Canada or downloaded them from the Internet to a computer in Canada, this Agreement is governed by the laws of the Province of Ontario. Otherwise, this Agreement is governed by the laws of the country in which you acquired the Software and Documentation or, if the Software and Documentation were downloaded from the Internet, the country in which the receiving computer was located.

6. EXPORT CONTROLS

None of the Software, Documentation or underlying information or technology may be downloaded or otherwise exported or reexported (i) into (or to a national or resident of) Cuba, Iraq, Libya, the Federal Republic of Yugoslavia (Serbia and Montenegro), North Korea, Iran, Syria or any other country to which the U.S. has embargoed goods; or (ii) to anyone on the Denied Parties List distributed by the U.S. Export Regulations Office. By downloading or using the Software and Documentation, you agree to the foregoing, and you represent and warrant that you are not located in, under the control of, or a national or resident of any of these countries or on such list.

7. TERM AND TERMINATION

The term of this license will begin with your initial trial session. If you access only the limited functions of the Software for five minutes at a time, the term of this license will expire after thirty days. If you obtain an enabling key from IBM for full access to the Software, this license will expire thirty days from the date on which the enabling key was issued. This license will

terminate upon your failure to comply with any of the terms of this Agreement, whichever comes first. When this license ends, for whatever reason, you must destroy all copies of the Software and Documentation.

8. NO WARRANTY

THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS-IS," AND IBM EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, WITH REGARD TO THE SOFTWARE AND DOCUMENTATION, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTY OF TITLE OR NON-INFRINGEMENT OR IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

9. LIMITATION ON LIABILITY

IN NO EVENT WILL IBM BE LIABLE TO YOU OR TO ANY THIRD PARTY FOR ANY DAMAGES WHATSOEVER, INCLUDING, BUT NOT LIMITED TO, SPECIAL, DIRECT, INDIRECT AND CONSEQUENTIAL DAMAGES OR ANY DAMAGES RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION BASED ON CONTRACT, TORT OR ANY OTHER THEORY, ARISING OUT OF OR IN CONNECTION WITH THE USE OF THE SOFTWARE AND DOCUMENTATION, EVEN IF IBM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE MAY NOT APPLY TO YOU.

10. SEVERABILITY, ENTIRE AGREEMENT

If any part of this Agreement is found to be void or unenforceable, it will not affect the rest of the Agreement, which shall remain valid and enforceable according to its terms.

This Agreement is the only understanding and agreement between you and IBM with respect to the Software and Documentation, and it supersedes all prior related communications, understandings or agreements between you and IBM.

11 FULLY FUNCTIONAL SOFTWARE

If you are interested in purchasing a license to the Data Explorer product, contact IBM for information about the license fee and to order:

For information, call 1-800-388-9820 or e-mail "ibmdx@watson.ibm.com"

To order, call 1-800-IBM-CALL

Microsoft Internet Explorer (See **http://www.microsoft.com/ie**)

NOTICE SPECIFIC TO SOFTWARE AVAILABLE ON THIS WEBSITE

Any software that is made available to download from this server ("Software") is the copyrighted work of Microsoft and/or its suppliers. Use of the Software is governed by the terms of the end user license agreement, if any, which accompanies or is included with the Software ("License Agreement"). An end user will be unable to install any Software that is accompanied by or includes a License Agreement, unless he or she first agrees to the License Agreement terms.

The Software is made available for downloading solely for use by end users according to the License Agreement. Any reproduction or redistribution of the Software not in accordance with the License Agreement is expressly prohibited by law, and may result in severe civil and criminal penalties. Violators will be prosecuted to the maximum extent possible.

WITHOUT LIMITING THE FOREGOING, COPYING OR REPRODUCTION OF THE SOFTWARE TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION IS EXPRESSLY PROHIBITED.

THE SOFTWARE IS WARRANTED, IF AT ALL, ONLY ACCORDING TO THE TERMS OF THE LICENSE AGREEMENT. EXCEPT AS WARRANTED IN THE LICENSE AGREEMENT, MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES AND CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT.

RESTRICTED RIGHTS LEGEND. Any Software which is downloaded from this Server for or on behalf of the United States of America, its agencies and/or instrumentalities ("U.S. Government"), is provided with Restricted Rights. Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software - Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Microsoft Corporation, One Microsoft Way, Redmond, WA 98052-6399.

NOTICE SPECIFIC TO DOCUMENTS AVAILABLE ON THIS WEBSITE

Permission to use Documents (such as white papers, press releases, data sheets and FAQs) from this server ("Server") is granted, provided that (1) the below copyright notice appears in all copies and that both the copyright notice and this permission notice appear, (2) use of such Documents from this Server is for informational and non-commercial or personal use only and will not be copied or posted on any network computer or broadcast in any media, and (3) no modifications of any Documents are made. Use for any other purpose is expressly prohibited by law, and may result in severe civil and criminal penalties. Violators will be prosecuted to the maximum extent possible.

Documents specified above do not include the design or layout of the microsoft.com web site or any other Microsoft owned, operated, licensed or controlled site. Elements of Microsoft web sites are protected by trade dress and other laws and may not be copied or imitated in whole or in part. No logo, graphic, sound or image from any Microsoft website may be copied or retransmitted unless expressly permitted by Microsoft.

MICROSOFT AND/OR ITS RESPECTIVE SUPPLIERS MAKE NO REPRESENTATIONS ABOUT THE SUITABILITY OF THE INFORMATION CONTAINED IN THE DOCUMENTS AND RELATED GRAPHICS PUBLISHED ON THIS SERVER FOR ANY PURPOSE. ALL SUCH DOCUMENTS AND RELATED GRAPHICS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. MICROSOFT AND/OR ITS RESPECTIVE SUPPLIERS HEREBY DISCLAIM ALL WARRANTIES AND CONDITIONS WITH REGARD TO THIS INFORMATION, INCLUDING ALL IMPLIED WARRANTIES AND CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL MICROSOFT AND/OR ITS RESPECTIVE SUPPLIERS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF INFORMATION AVAILABLE FROM THIS SERVER.

THE DOCUMENTS AND RELATED GRAPHICS PUBLISHED ON THIS SERVER COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN. MICROSOFT AND/OR ITS RESPECTIVE SUPPLIERS MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED HEREIN AT ANY TIME.

NOTICES REGARDING SOFTWARE, DOCUMENTS AND SERVICES AVAILABLE ON THIS WEBSITE.

IN NO EVENT SHALL MICROSOFT AND/OR ITS RESPECTIVE SUPPLIERS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF SOFTWARE, DOCUMENTS, PROVISION OF OR FAILURE TO PROVIDE SERVICES, OR INFORMATION AVAILABLE FROM THIS SERVER.

LINKS TO THIRD PARTY SITES
THE LINKS IN THIS AREA WILL LET YOU LEAVE MICROSOFT'S SITE. THE LINKED SITES ARE NOT UNDER THE CONTROL OF MICROSOFT AND MICROSOFT IS NOT RESPONSIBLE FOR THE CONTENTS OF ANY LINKED SITE OR ANY LINK CONTAINED IN A LINKED SITE, OR ANY CHANGES OR UPDATES TO SUCH SITES. MICROSOFT IS PROVIDING THESE LINKS TO YOU ONLY AS A CONVENIENCE, AND THE INCLUSION OF ANY LINK DOES NOT IMPLY ENDORSEMENT BY MICROSOFT OF THE SITE.

TRADEMARKS. Microsoft, Windows, Windows NT, MSN, The Microsoft Network and/or other Microsoft products referenced herein are either trademarks or registered trademarks of Microsoft. Other product and company names mentioned herein may be the trademarks of their respective owners.

The names of companies, products, people, characters and/or data mentioned herein are fictitious and are in no way intended to represent any real individual, company, product or event, unless otherwise noted.

Contact www@microsoft.com or webmaster@msn.com with questions or problems, as appropriate.