



UNC  
INSTITUTE FOR  
THE ENVIRONMENT

---

## ***Visualization Environment for Rich Data Interpretation (VERDI 1.2): User's Manual***

U.S. EPA Contract No. EP-W-05-045, "Operation of the Center for  
Community Air Quality Modeling and Analysis"

---

Prepared for: William Benjey and Donna Schwede  
U.S. EPA, ORD/NERL/AMD/APMB  
E243-04  
USEPA Mailroom  
Research Triangle Park, NC 27711

Prepared by: Liz Adams and Qun He  
Institute for the Environment  
The University of North Carolina at Chapel Hill  
137 E. Franklin St., CB 6116  
Chapel Hill, NC 27599-6116

Date: December 4, 2009

## Contents

<b>1</b>	<b>Introduction.....</b>	<b>1</b>
1.1	Background.....	1
1.2	Where to Obtain VERDI.....	2
1.3	Where to Obtain VERDI Documentation.....	4
1.4	Help Desk Support for VERDI.....	6
1.5	Future VERDI Development.....	8
<b>2</b>	<b>Requirements for Using VERDI.....</b>	<b>8</b>
2.1	Java Runtime Environment.....	8
2.2	Memory and CPU Requirements.....	8
2.3	Requirements to run VERDI remotely.....	9
2.4	Graphics Requirements.....	9
<b>3</b>	<b>VERDI Installation Instructions.....</b>	<b>9</b>
3.1	VERDI Installation.....	9
3.2	Installation Instructions for Linux and Other non-Windows JRE™ 6 Supported System Configurations.....	9
3.3	Installation Instructions for Windows.....	11
3.4	Setting VERDI Preferences.....	14
<b>4</b>	<b>Starting VERDI and Getting Your Data into VERDI.....</b>	<b>15</b>
4.1	Starting VERDI.....	15
4.1.1	Windows.....	15
4.1.2	Linux and Other non-Windows JRE™ 6 Supported System Configurations.....	15
4.2	Main Window.....	15
4.3	Floating the Dataset and Formula Panes.....	17
<b>5</b>	<b>Navigating VERDI's Main Menu Options.....</b>	<b>17</b>
5.1	File Menu Options.....	18
5.1.1	Open Project.....	18
5.1.2	Save Project.....	18
5.2	Plots Menu Options.....	18
5.2.1	Undock all plots.....	18

5.2.2	Animate tile plots.....	19
5.3	Window Menu Options.....	20
5.3.1	Datasets and Formulas.....	20
5.3.2	List of plots.....	20
5.4	Help Menu Options.....	21
<b>6</b>	<b>Working with Gridded Datasets.....</b>	<b>21</b>
6.1	Gridded Input File Formats.....	21
6.2	Example Datasets.....	21
6.3	Adding and Removing a Gridded Dataset.....	22
6.4	Variables List.....	24
6.5	Time Steps, Layers Panels.....	24
6.6	Domain Panel.....	25
6.7	Saving Projects.....	25
6.8	Configuring Default Directories.....	26
<b>7</b>	<b>Working with Formulas.....</b>	<b>26</b>
7.1	Adding and Removing a Formula.....	26
7.2	Example Formulas.....	27
7.3	Selecting a Formula for Plotting.....	28
7.4	Saving Formulas.....	28
7.5	Time Step Range, Layer Range, and Edit Domain.....	28
<b>8</b>	<b>Working with Area Files.....</b>	<b>28</b>
8.1	Area File Formats.....	28
8.2	Example Area File.....	29
8.3	Adding and Removing an Area File.....	29
8.4	Areas List.....	29
8.5	Areal Interpolation.....	29
<b>9</b>	<b>Spatial and Temporal Data Subsetting.....</b>	<b>34</b>
9.1	Specify Time Step Range.....	34
9.2	Specify Layer Range.....	35
9.3	Specify Domain Range.....	35
9.4	Rules of Precedence for Subsetting Data.....	37
<b>10</b>	<b>Creating Plots.....</b>	<b>38</b>
10.1	Tile Plot.....	38
10.2	Fast Tile Plot.....	39

10.3	Areal Interpolation Plot.....	41
10.3.1	Option Pull-down Menu Item .....	41
10.3.2	Areal Values for Polygon Segment.....	44
10.3.3	View and Export Areal Plot Data in Spreadsheet Format .....	44
10.3.4	Export Areal Plot Data to Shapefiles .....	45
10.4	Vertical Cross Section Plot .....	46
10.5	Time Series Plot.....	48
10.6	Time Series Bar Plot .....	48
10.7	Scatter Plot.....	49
10.8	Vector Plot .....	51
10.9	Contour Plot.....	53
<b>11</b>	<b>Plot Menu Bar .....</b>	<b>53</b>
11.1	File Menu Options .....	54
11.2	Configure Menu Option.....	54
11.2.1	Configuring plots .....	54
11.2.2	Loading and saving configuration .....	57
11.2.3	Configure Maps .....	57
11.3	Controls Menu Options.....	60
11.3.1	Zooming in and out.....	60
11.3.2	Probing values at specific points.....	62
11.3.3	Probing a domain region of data.....	62
11.3.4	Set Data Ranges .....	63
11.3.5	Showing latitude and longitude .....	65
11.4	Plot Menu Options .....	65
11.4.1	Time series plots .....	66
11.4.2	Animate plots.....	66
11.4.3	Observational data overlays.....	67
11.4.4	Vector overlays .....	68
<b>12</b>	<b>Supported Grid and Coordinate Systems (Map Projections).....</b>	<b>69</b>
12.1	IO/API formatted data.....	70
12.2	CAMx Gridded Data.....	72
<b>13</b>	<b>I/O API Utilities, Data Conversion Programs, and Libraries .....</b>	<b>75</b>
<b>14</b>	<b>Contributing to VERDI Development .....</b>	<b>75</b>
<b>15</b>	<b>Known Bugs.....</b>	<b>76</b>

- 16 Mathematical Functions.....76**
- 17 Command Line Scripting.....79**
  - 17.1 Example Script for Linux Users .....79
  - 17.2 Example Script for Windows Users.....80
  - 17.3 Commands that can be used (listed in alphabetical order) .....81
- 18 Areal Interpolation Calculations .....84**
- 19 Licenses for JAVA Libraries used by VERDI .....85**
- 20 Acknowledgements .....86**
  - 20.1 Data Contributions: .....86
    - 20.1.1 Sample CAMx dataset .....86
    - 20.1.2 Sample Mercator Projection dataset .....86
  - 20.2 Data Reader Contributions.....86
    - 20.2.1 I/O Service Provider (IOSP) interface for CAMx: .....86
    - 20.2.2 Incorporating the IOSP into netCDF netcdf-java v4.1 library:.....86

## Figures

Figure 1-1 Top of Main VERDI Web Site Page.....	2
Figure 1-2 Downloading VERDI from the CMAS Web Site, Step 1.....	3
Figure 1-3 Downloading VERDI from the CMAS Web Site, Step 2.....	3
Figure 1-4 Downloading VERDI from the CMAS Web Site, Step 3.....	4
Figure 1-5 Getting Documentation on VERDI from the CMAS Web Site.....	5
Figure 1-6 VERDI Documentation on the CMAS Web Site.....	5
Figure 1-7 Top of Bugzilla Form for VERDI.....	6
Figure 1-8 Bottom of Bugzilla Form for VERDI.....	7
Figure 1-9 Add Attachment Form in Bugzilla.....	7
Figure 3-1 VERDI Setup Wizard.....	11
Figure 3-2 License Agreement.....	12
Figure 3-3 Selecting an Installation Directory.....	12
Figure 3-4 Setting the Start Menu Folder.....	13
Figure 3-5 File Extraction.....	13
Figure 3-6 Installation Complete.....	14
Figure 4-1 Starting VERDI in Windows.....	15
Figure 4-2 VERDI Main Window.....	17
Figure 5-1 VERDI Main Menu Options.....	18
Figure 5-2 Animate Plots Dialog and Tile Plots.....	20
Figure 6-3 Open Dataset File Browser.....	22
Figure 6-4 Datasets Pane Displaying Information about a Dataset.....	23
Figure 6-5 Right Click on Variable in Dataset Pane.....	24
Figure 6-6 Dataset Metadata Information.....	25
Figure 7-1 Adding Multiple Variables to Formula Editor.....	27
Figure 8-1 Areas Pane.....	30
Figure 8-2 Open Area File Browser.....	30
Figure 8-3 Open Area File: Select Name Field.....	31
Figure 8-4 Coordinate System.....	31
Figure 8-5 Projection Information.....	32
Figure 8-6 Additional Data Fields appear depending on projection selected.....	32
Figure 8-7 Area Name Fields available for Shapefile.....	33

Figure 9-1 Specify Time Step Range .....34

Figure 9-2 Edit Layer Range in Formula Pane .....35

Figure 9-3 Using the Slider to View the Domain Panel .....36

Figure 9-4 Edit Domain Dialog Box.....36

Figure 9-5 Error obtained when incompatible subset domains are created using the Dataset pane .....38

Figure 10-1 Tile Plot.....39

Figure 10-2 Fast Tile Plot .....40

Figure 10-3 Areal Plot: Area Average .....42

Figure 10-4 Areal Plot: Area Totals.....42

Figure 10-5 Areal Interpolation Plot: Show Grid (Gridded Data).....43

Figure 10-6 Areal Interpolation Plot: Show Selected Areas.....43

Figure 10-7 Areal Values for Polygon Segments .....44

Figure 10-8. Right Click on Area Plot .....45

Figure 10-9. Area Information Spreadsheet.....45

Figure 10-10. Export spreadsheet data to file (.txt or .csv).....45

Figure 10-11. Name and save spreadsheet file .....45

Figure 10-12 Export Shapefile.....46

Figure 10-13 Name and save shapefile .....46

Figure 10-14 Vertical Cross Section Plot .....47

Figure 10-15 Vertical Cross Section Dialog Box .....47

Figure 10-16 Time Series Plot .....48

Figure 10-17 Time Series Bar Plot .....49

Figure 10-18 Scatter Plot .....50

Figure 10-19 Scatter Plot Dialog Box.....50

Figure 10-20 Scatter Plot Export Data into a CSV file.....51

Figure 10-21. Vector Plot .....52

Figure 10-22. Vector Plot Dialog Box .....52

Figure 10-23. Contour Plot .....53

Figure 10-24. Contour Plot Menu Options .....53

Figure 11-1 Tile Plot Pull-down Menu Options .....54

Figure 11-2. Configure Plot, Titles Tab.....56

Figure 11-3. Configure Plot, Color Map Tab .....56

Figure 11-4. Configure Plot, Labels Tab .....56

Figure 11-5. Configure Plot, Other Tab.....56

Figure 11-6 Configure Maps for a Tile Plot .....58

Figure 11-7 Fast Tile Plot Map Options .....59

Figure 11-8 Manage Layers, Add Layer, Edit Style.....60

Figure 11-9 Right Click on Plot to Zoom Out .....61

Figure 11-10 Click on Plot to Probe: Data Value Shown in Lower Left of VERDI, Grid  
Values Shown in Lower Right .....62

Figure 11-11 Spreadsheet Showing Probed Values for Region of Interest .....63

Figure 11-12 Configure Min and Max Range Values .....64

Figure 11-13 Lat/Lon Values Shown in Lower Right of VERDI.....65

Figure 11-14 Plot Menu Options .....66

Figure 11-15 Animate Plot Dialog Box .....67

Figure 11-16 Tile Plot with Observational Data Overlay .....68

Figure 11-17 Wind Vector Overlay on an Ozone Tile Plot .....69

Figure 12-1 Lambert Conformal Conic Map Projection Example Plot.....70

Figure 12-2 Polar Stereographic Map Projection Example Plot.....71

Figure 12-3 Mercator Map Projection Example Plot.....72

Figure 12-4 Example CAMx diagnostic text file.....73

Figure 12-5 Models-3 IO/API Map Projection Parameters for Lambert.....73

Figure 12-6 Sample Projection File: camxproj.txt.....74

Figure 12-7 CAMx Example Plot.....74

Figure 16-1 Location of run.bat script in Windows.....80

Figure 16-2 Submit run.bat script from Run command.....81

# 1 Introduction

## 1.1 Background

This manual describes how to use the Visualization Environment for Rich Data Interpretation (VERDI). VERDI is a flexible and modular Java-based visualization software tool that allows users to visualize multivariate gridded environmental datasets created by environmental modeling systems such as the Community Multiscale Air Quality (CMAQ) modeling system and the Weather Research and Forecasting (WRF) modeling system. These systems produce files of gridded concentration and deposition fields that users need to visualize and compare with observational data both spatially and temporally. VERDI can facilitate these types of analyses.

Initial development of VERDI was done by the Argonne National Laboratory for the U.S. Environmental Protection Agency (EPA) and its user community. Argonne National Laboratory's work was supported by the EPA through U.S. Department of Energy contract DE-AC02-06CH11357. Further development has been performed by the University of North Carolina Institute for the Environment under U.S. EPA Contract No. EP-W-05-045 and EP-W-09-023, by Lockheed Corporation under U.S. EPA contract No. 68-W-04-005, and Argonne National Laboratory. VERDI is licensed under the Gnu Public License (GPL) version 3, and the source code is available through [verdi.sourceforge.net](http://verdi.sourceforge.net). Instructions for developers within the community are included in the Developer User Guide (see Section 1.2). VERDI is supported by the Community Modeling and Analysis System (CMAS) Center under U.S. EPA Contract No. EP-W-09-023. The CMAS Center is located within the Institute for the Environment at the University of North Carolina at Chapel Hill.

This guide describes VERDI version 1.2 released in December 2009.

The following are useful web links for obtaining VERDI downloads and support:

1. VERDI Visualization Tool web site:  
<http://www.verdi-tool.org>
2. CMAS download page for users of VERDI:  
<http://www.cmascenter.org/download/models.cfm>
3. CMAS SourceForge.net website for developers of VERDI:  
<http://sourceforge.net/projects/verdi/>
4. To submit bug reports, questions, and/or requests:  
[http://bugz.unc.edu/enter\\_bug.cgi?product=VERDI](http://bugz.unc.edu/enter_bug.cgi?product=VERDI)  
(If you do not already have a login and password, click the "Home" link for instructions on how to obtain them.)

## 1.2 Where to Obtain VERDI

You can download the latest version of VERDI from <http://www.verdi-tool.org/> (see Figure 1-1). When you click on the link to download VERDI, you will be sent to the CMAS Model Download Center. To download and install VERDI, follow the instructions below, skipping step 2. Alternatively, you may also begin at the CMAS web site <http://www.cmascenter.org>, and follow the instructions below:

1. Log in using an existing CMAS account, or create a new CMAS account.
2. Hover the cursor over the Download Center link on the left-hand side of the web site and choose MODELS from the menu that appears.
3. Select a model family to download, as shown in Figure 1-2. Use the pull-down list to select VERDI, and then click Submit.
4. Select the product you wish to download, as shown in Figure 1-3. Also specify the type of computer you are planning to run VERDI on (such as Linux PC, Windows, or Other) from the items in the scroll list. Note that the compilers question is not relevant for VERDI so it can be skipped. Finally, click Submit.
5. In the table that appears, follow the links to the Linux or Windows installation instructions, the release notes file, the user's manual, the test documentation, and either a zip archive for Windows or a gzipped tar archive for Linux (see Figure 1-4).

**Figure 1-1 Top of Main VERDI Web Site Page**

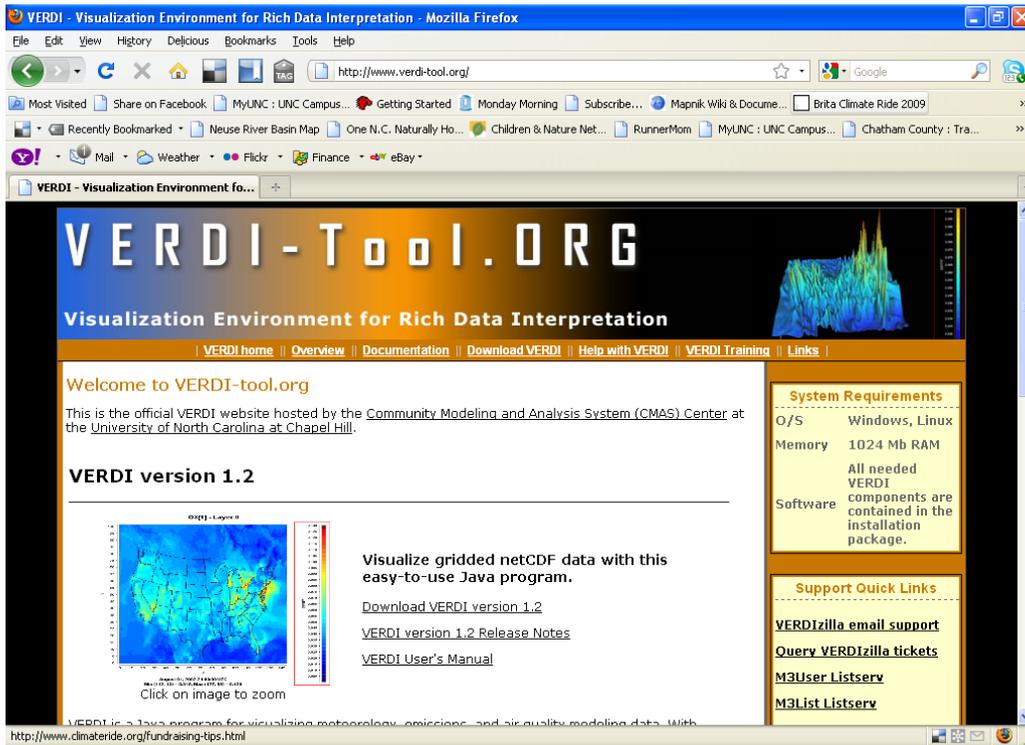


Figure 1-2 Downloading VERDI from the CMAS Web Site, Step 1

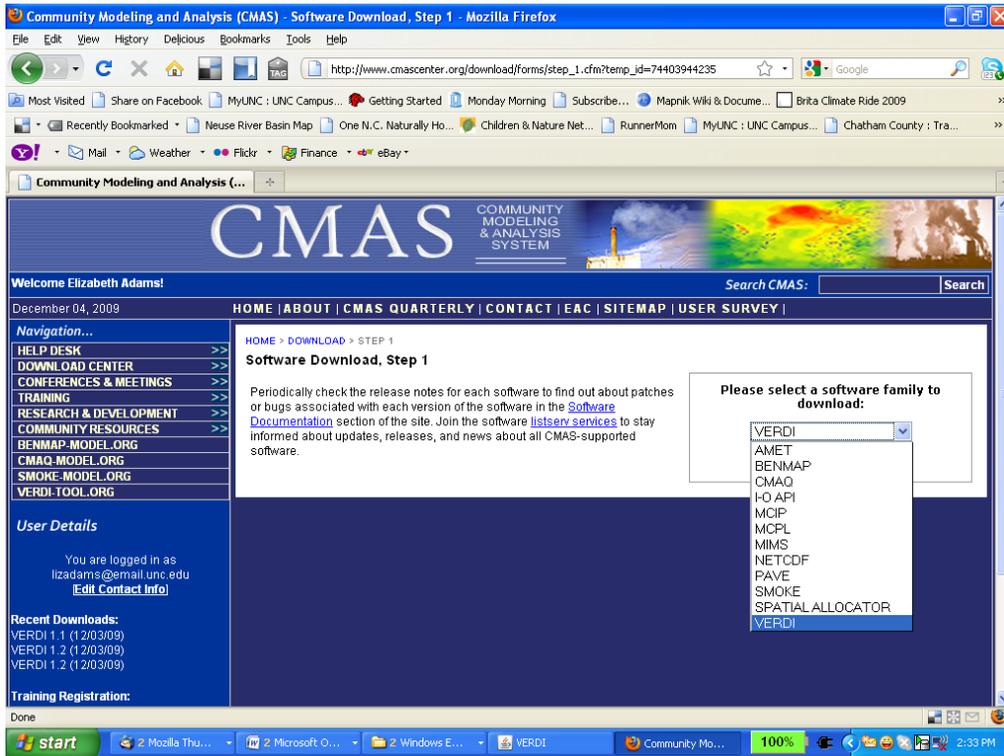


Figure 1-3 Downloading VERDI from the CMAS Web Site, Step 2

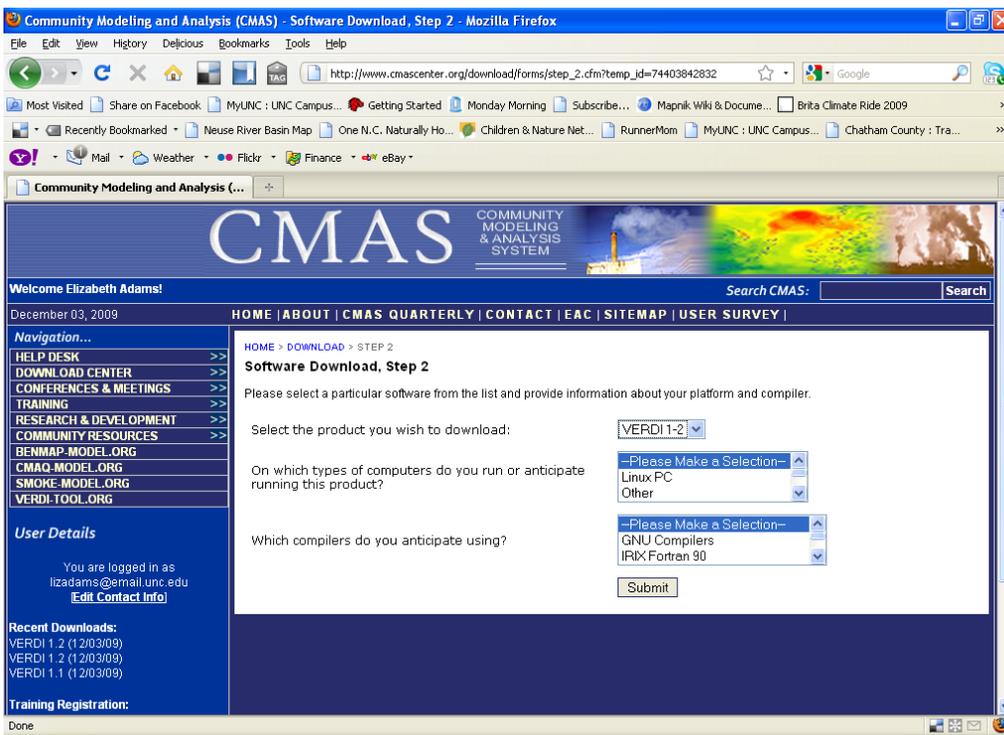
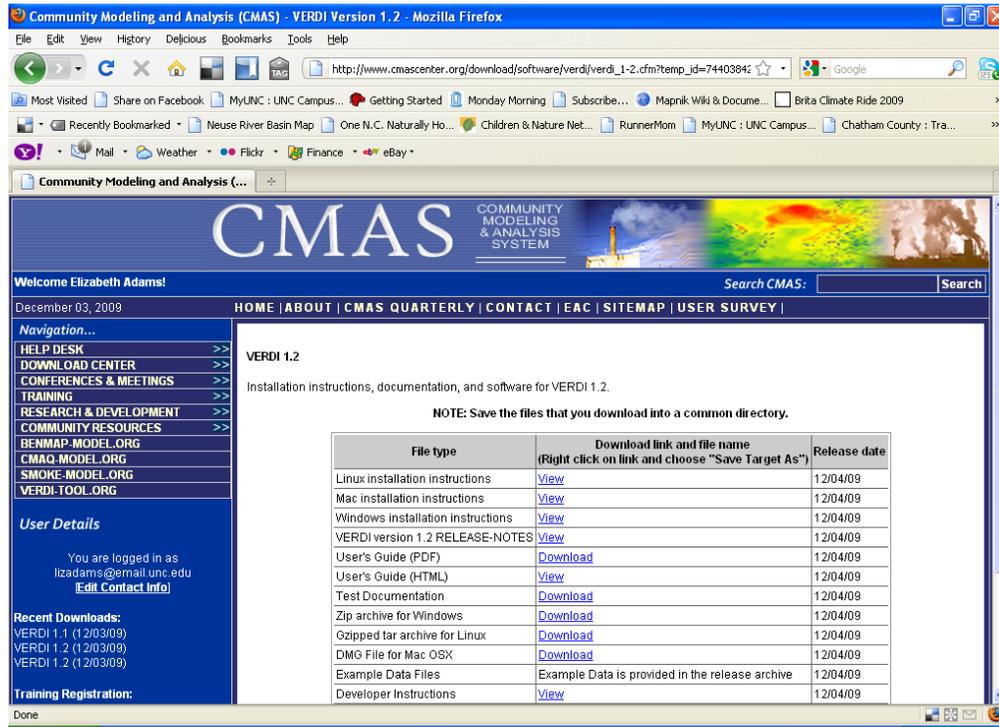


Figure 1-4 Downloading VERDI from the CMAS Web Site, Step 3



### 1.3 Where to Obtain VERDI Documentation

Documentation is available in several locations, described below. Each location provides links to the available documentation for VERDI, which can be viewed in your web browser or downloaded and saved to your computer.

- The VERDI download page on the CMAS website (accessed using step 4 in Section 1.2) contains links to all of the available documentation.
- On the left-hand side of the [www.cmascenter.org](http://www.cmascenter.org) web site, you can hover the cursor over the Help Desk link and choose DOCUMENTATION from the menu that appears. Select the product documentation that you want from the drop-down list (Figure 1-5) and click Submit. Select the model release from the drop-down list and click Search. The resulting documentation pane shows that the available documentation for the chosen release of VERDI.
- From the [www.verdi-tool.org](http://www.verdi-tool.org) web site, there are two ways to access the VERDI documentation on the [www.cmascenter.org](http://www.cmascenter.org) model documentation web page: (1) A link near the top of the [www.verdi-tool.org](http://www.verdi-tool.org) home page sends you to a new page where you choose the model version. When you click submit, you are taken to the [www.cmascenter.org](http://www.cmascenter.org) VERDI documentation web page. (2) Direct links from a box on the right-hand side of the [www.verdi-tool.org](http://www.verdi-tool.org) home page take you to the documentation available for VERDI. Figure 1-6 shows the list of documentation that is available for download.

Figure 1-5 Getting Documentation on VERDI from the CMAS Web Site

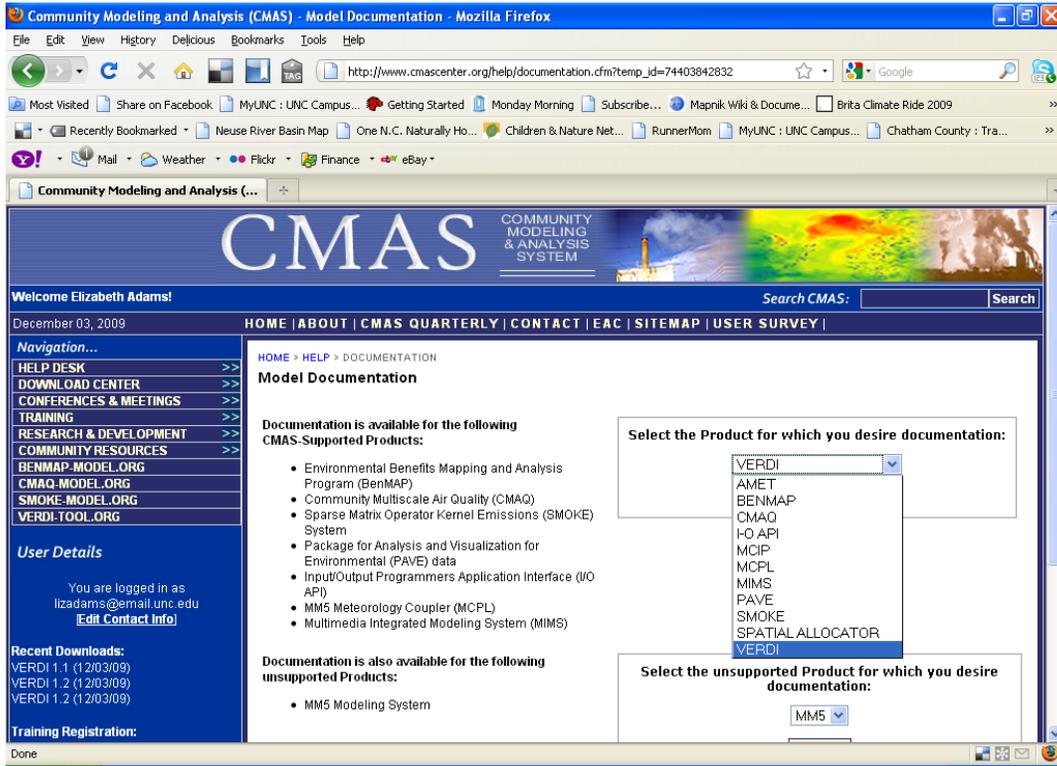
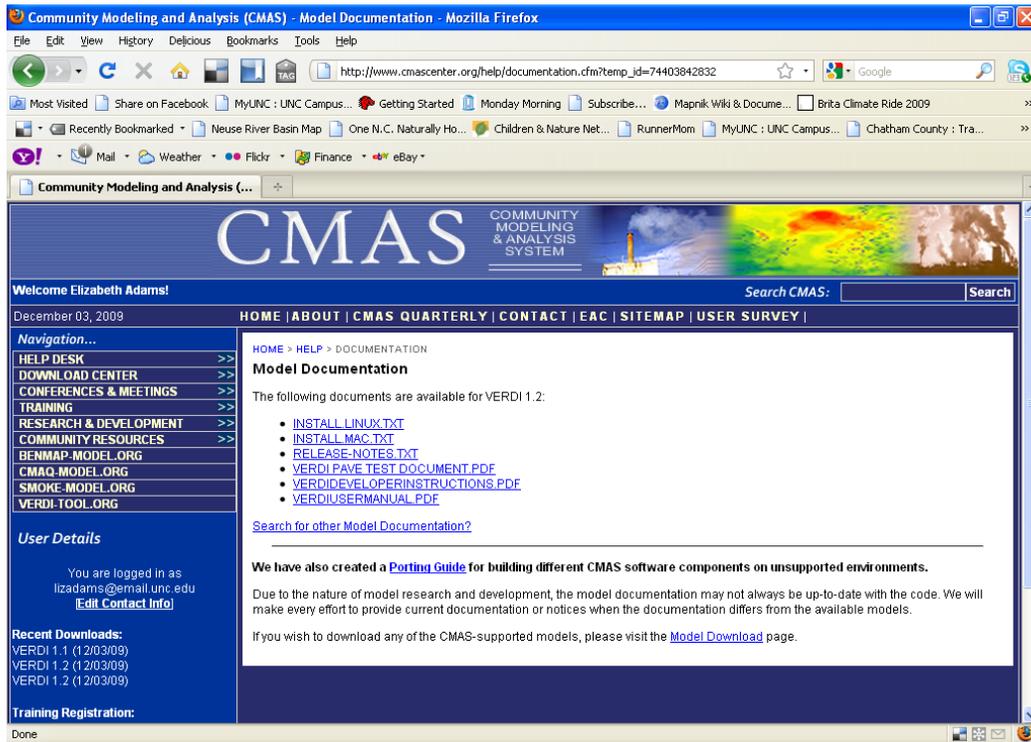


Figure 1-6 VERDI Documentation on the CMAS Web Site



## 1.4 Help Desk Support for VERDI

You are encouraged to submit questions, report errors, and make requests for enhancements using the web-based Bugzilla system, which is used by CMAS to track issues with supported products. The Bugzilla site for VERDI is [http://bugz.unc.edu/enter\\_bug.cgi?product=VERDI](http://bugz.unc.edu/enter_bug.cgi?product=VERDI). If you have not used Bugzilla before, first go to <http://bugz.unc.edu> and follow the instructions for requesting a new account. Once you are logged in, you can enter information about your question or bug using the form shown in Figure 1-7. When filing a report, you will need to choose from among the following components: Automating VERDI with Scripts, Configuring Plots, Documentation, Exporting Images and Animations, Formulas, GIS Layers, Importing Model Data, Importing Observation Data, Projects, Selections of Domains, Layers, and Time-steps, and Using the GUI.

Bugzilla also allows you to submit software code improvements for review and testing by others, to submit small test cases, to submit new GIS background maps, or to make other contributions to VERDI. To submit attachments to a bug report or request, click on the Add Attachment button in the bottom half of the form (Figure 1-8) to bring up a form requesting additional information about the attachment (Figure 1-9). Note that only minor code updates should be submitted through Bugzilla. If you are doing a substantial amount of software development on VERDI, please see Section 1.5 for additional information.

**Figure 1-7 Top of Bugzilla Form for VERDI**

The screenshot shows the top portion of the Bugzilla 'Enter Bug' form for VERDI. At the top, there is a navigation bar with links for Home, New, Search, Find, Reports, My Requests, Preferences, and Log out. Below this, a message advises users to read the bug writing guidelines and search for existing bugs. The form fields are organized into two columns. The left column contains Reporter (lizadams@email.unc.edu), Version (1.01), Severity (question), Priority (P3), Initial State (NEW), Assign To, QA Contact, Cc, Default CC, Estimated Hours (0.0), and Deadline (YYYY-MM-DD). The right column contains Product (VERDI), Component (Automating VERDI with Scripts), Platform (PC), and OS (Linux). The Component dropdown menu is currently open, displaying a list of options: Automating VERDI with Scripts, Configuring Plots, Documentation, Exporting Images and Animations, and Formulas.

Figure 1-8 Bottom of Bugzilla Form for VERDI

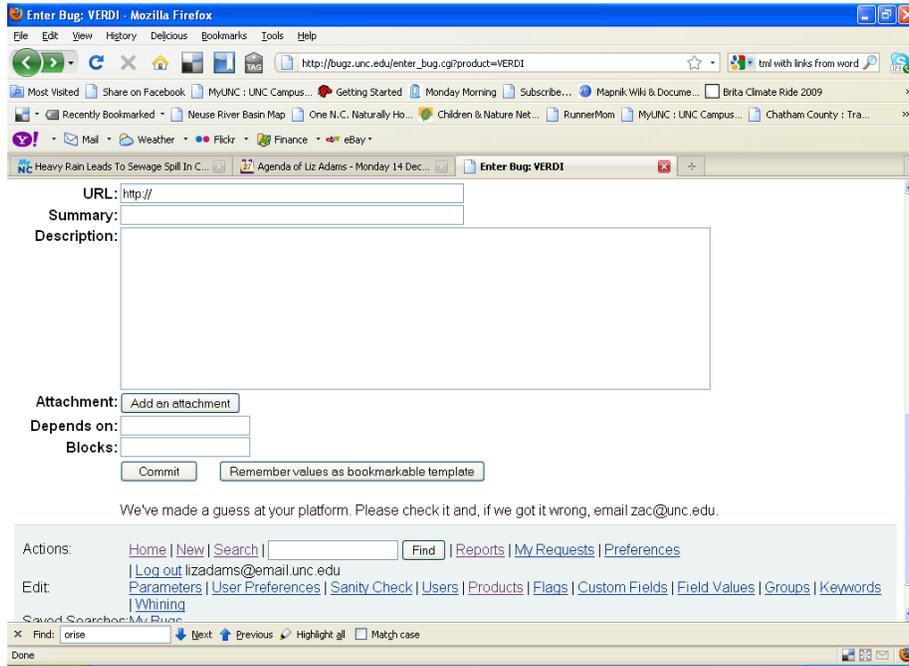
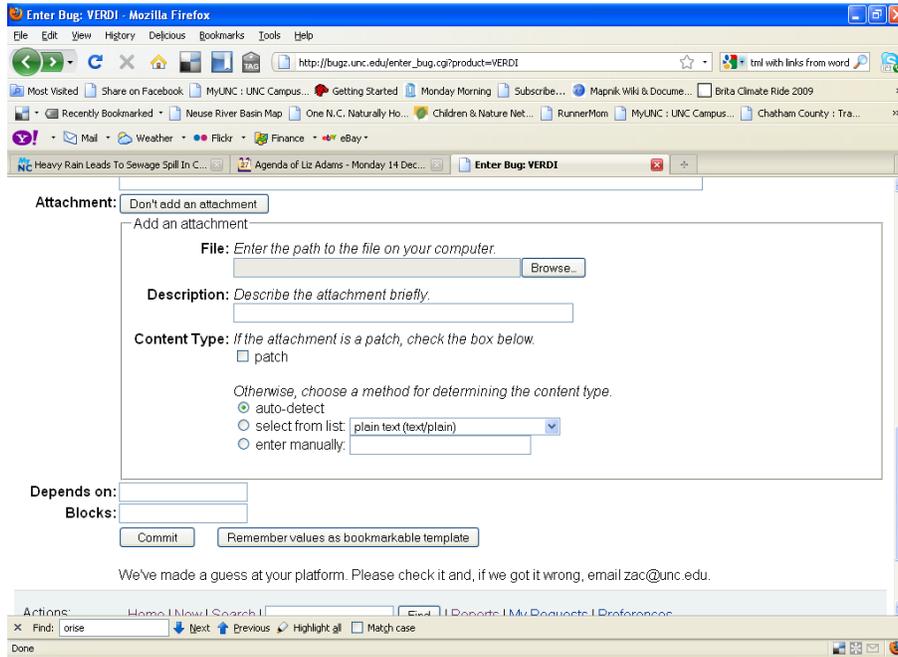


Figure 1-9 Add Attachment Form in Bugzilla



## 1.5 Future VERDI Development

As stated in Schwede et al. (2007),<sup>1</sup> “VERDI is intended to be a community based visualization tool with strong user involvement.” The VERDI source code is available to the public under a GPL license at <http://sourceforge.net/projects/verdi/>. This allows users who wish to make improvements to VERDI to download the software, and to develop enhancements and improvements that they feel may be useful to the modeling community. Examples could include user-developed readers for additional file formats and modules for additional plot types. Users may wish to contribute data analysis routines such as adding the ability to do bilinear interpolation (smoothing) or adding nonlinear scales (e.g., log) to the color map. The direction of future development will depend on the resources and the needs of the modeling community. If you are interested in contributing code to VERDI, please review the information in Chapter 14, “Contributing to VERDI Development.”

## 2 Requirements for Using VERDI

### 2.1 Java Runtime Environment

VERDI requires version 6 of the Java Standard Edition Runtime Environment (JRE). The JRE™ 6 is provided as part of the VERDI release for Linux 32 bit and Windows. Instructions for how to download JRE™ 6 and the additional third-party libraries for Linux 64 bit are provided in section 3.2.

### 2.2 Memory and CPU Requirements

VERDI’s memory and CPU requirements largely depend on the size of the datasets to be visualized. Small datasets can be visualized and manipulated using less than 1024 megabytes of RAM, while larger datasets may need considerably more. If you are using datasets that require either more or less than 1024 MB of memory, you can change the default maximum memory setting used by VERDI:

- On Windows, edit the Verdi.ini file to specify a different heap size from the default 1024M.
- On Linux or another Unix platform, you can edit verdi.sh and replace the 1024 in –Xmx1024M with a different value; for example, -Xmx2048M will allow VERDI to access up to 2048MB (or 2GB) of RAM.

Note that even slower CPUs can quickly view and animate smaller datasets, whereas larger datasets can take longer. Initially, VERDI’s performance was slow when it was used across the network via ssh. This problem was addressed in version 1.1 with the addition of the Fast Tile

---

<sup>1</sup> Schwede, D., N. Collier, J. Dolph, M.A. Bitz Widing, T. Howe, 2007: A New Tool for Analyzing CMAQ Modeling Results: Visualization Environment for Rich Data Interpretation (VERDI). Proceedings, CMAS 2007 Conference.

Plot. In VERDI version 1.2, the Fast Tile Plot supports the configuration options of the original tile plot including display lat/long, probing, zooming and adding grid lines. A future release will add observational and vector overlay capability to the Fast Tile Plot.

### **2.3 Requirements to run VERDI remotely**

VERDI may be used to run on a remote compute server and have the graphics display locally on your desktop machine (Unix workstation, Mac, or PC) using the Fast Tile Plot. Your computer needs to be configured to run X-Windows. Typically, you will connect to the remote compute server using secure shell (SSH). If you are using an X-Server and wish to generate 3-D plots using Open GL, you need to turn on Open GL support within the X-Server.

### **2.4 Graphics Requirements**

Three-dimensional contour plots require a graphics card with OpenGL or DirectX capability. By default VERDI uses OpenGL for 3D rendering. If you would like to use DirectX instead, add the line: `j3d=-Dj3d.rend=d3d` to the `verdi.ini` file.

## **3 VERDI Installation Instructions**

### **3.1 VERDI Installation**

Installing VERDI on a Windows operating system is accomplished by running a standard installer program. For Linux and other non-Windows platforms, VERDI is distributed as a gzipped tar file that contains the executable application and the third-party libraries needed for VERDI. Currently, the VERDI distributions provide a platform-specific Java Runtime Environment 6.0 (see Section 3.3) for Windows and Linux. Note that the version of Java provided with VERDI has been customized to support the three-dimensional plots used by VERDI's contour plot. For more information on supporting the contour plot on your non-windows platform of interest, see the Java3D discussion section 3.2 that follows. Instructions on how to download JRE™ 6 for other system configurations that run it are also provided.

### **3.2 Installation Instructions for Linux and Other non-Windows JRE™ 6 Supported System Configurations**

VERDI requires the following third party software: JRE™ 6, Java3D, and Java™ Advanced Imaging (JAI). The Linux 32 bit version of the required third party software is provided in the VERDI release tar file.

1. `tar -xvf verdi_1.2.tar` into a location where you would like to install VERDI.
2. Edit `verdi_1.2/verdi.sh`

- a. Change the path for the VERDI\_HOME variable to reflect the location where VERDI was installed (e.g., VERDI\_HOME=/usr/local/verdi)
  - b. If you download JRE™ 6 for a version other than Linux 32 bit that is provided with VERDI as described in the next section, change the path for the JAVA variable to reflect where the java executable has been installed.
3. If the settings for the variables in step 5 are correct, VERDI should now run if you execute the verdi.sh executable script (e.g., ./verdi.sh).

If your non-Windows computer requires a JRE™ 6 other than what was provided in the distribution:

1. Download Java 6 for your platform from <http://www.java.com/en/download/manual.jsp>.
2. Download Java3D
  - a. For all platforms, go to <http://java.sun.com/javase/technologies/desktop/java3d/>
  - b. Go to download section, click on Download Java 3D 1.5.1 Software, select your platform, agree to license agreement, and continue.
  - c. mv java3d-1\_5\_1-linux-i586.bin to the jre1.6.0\_06 directory
  - d. chmod a+x java3d-1\_5\_1-linux-i586.bin
  - e. ./java3d-1\_5\_1-linux-i586.bin (this will place the lib/ext and lib/i386 files where they are needed)
3. Download JAI 1.1.3
  - a. For all platforms, go to [https://jai.dev.java.net/binary-builds.html#Release\\_builds](https://jai.dev.java.net/binary-builds.html#Release_builds)
  - b. Installation instructions:  
[http://download.java.net/media/jai/builds/release/1\\_1\\_3/INSTALL.html](http://download.java.net/media/jai/builds/release/1_1_3/INSTALL.html)
  - c. Download the .tar.gz package and untar. This will create a directory "jai-1\_1\_3".
  - d. Copy the following files:
    - i. cp jai-1\_1\_3/lib/libmllib\_jai.so /path-to-/verdi\_1.2/jre1.6.0\_13/lib/i386
    - ii. cp jai-1\_1\_3/lib/\*.jar /path-to-/verdi\_1.2/jre1.6.0\_13/lib/ext

If JRE™ 6 is already installed on your Linux platform, and you would like to use the already installed version, you will need to add Java 3D support. Copy the following files from the included version of Java into the JRE\_HOME/bin directory used by your version of Java:

j3dcore-d3d.dll  
j3dcore-ogl.dll  
j3dcore-ogl-cg.dll  
j3dcore-ogl-chk.dll  
j3dutils.dll

For Linux platform with ia64 architecture, please find the installed jre or jdk library directory (ex. /usr/java/jre1.6.0\_13/lib/ext or /usr/java/jdk1.6.0\_11/jre/lib/ext) and copy all missing library files from \$VERDI\_HOME/jre1.6.0\_13/lib/ext directory. If ia64 jre/jdk package is not installed, please go to java.sun.com download the install file and install the package. Then copy the missing library files from \$VERDI\_HOME/jre1.6.0\_13/lib/ext directory.

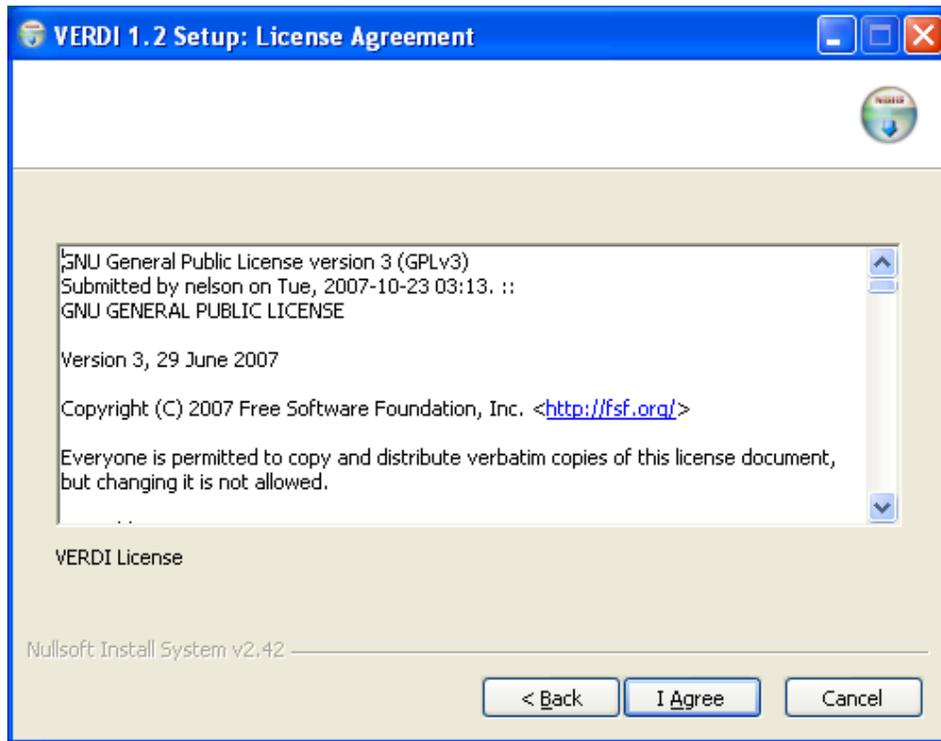
### 3.3 Installation Instructions for Windows

To install VERDI for Windows, run the installation program VERDI\_1.2\_Installer.exe on your Windows machine. Follow the instructions on the installation windows that appear, as shown in Figures 3-1 through 3-6. If you are unable to install VERDI on your computer, please check to see whether your user account is authorized to install software. You may need to request that a user with a computer administrator account install VERDI, or provide you with an account that has permission to install software. For more information about user account types, click Start and select Control Panel and then click on the User Account icon.

Figure 3-1 VERDI Setup Wizard



**Figure 3-2 License Agreement**



**Figure 3-3 Selecting an Installation Directory**

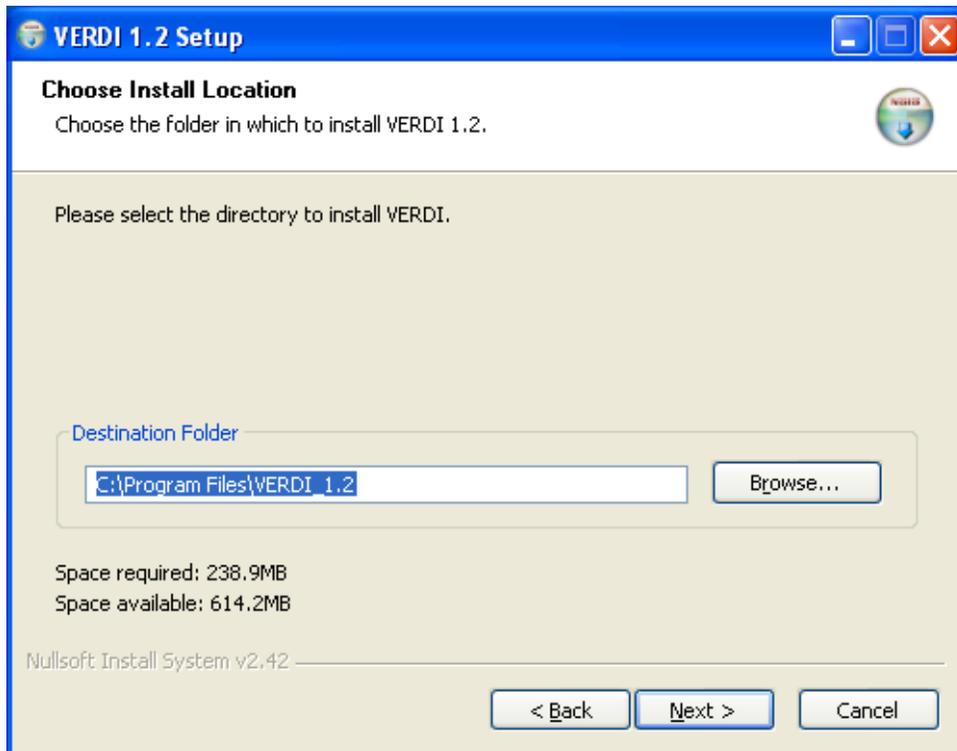


Figure 3-4 Setting the Start Menu Folder

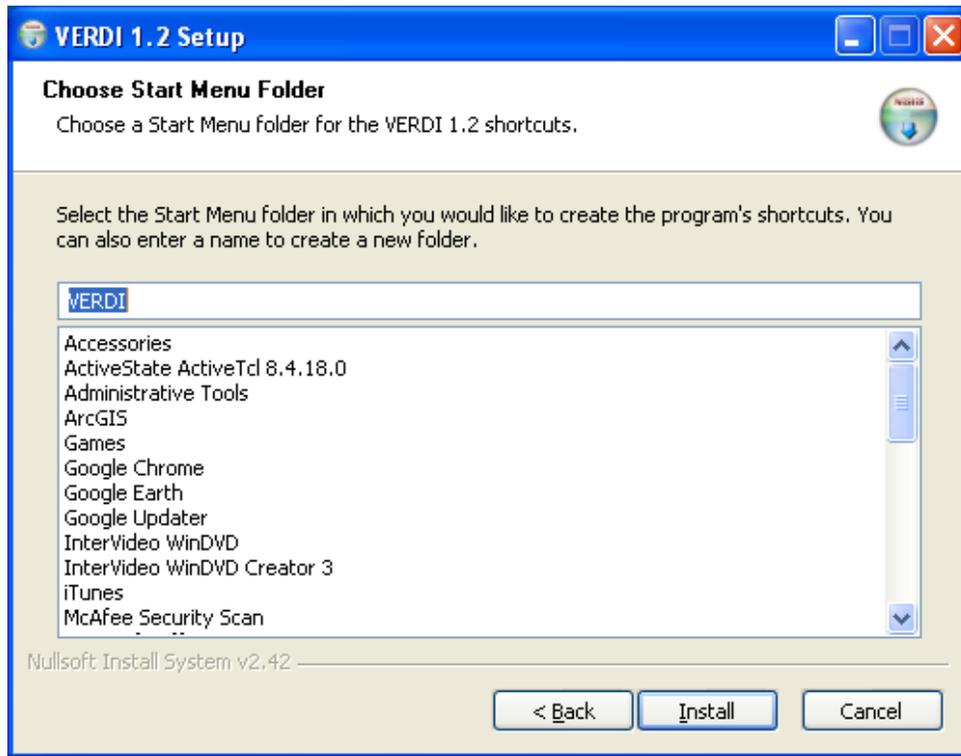
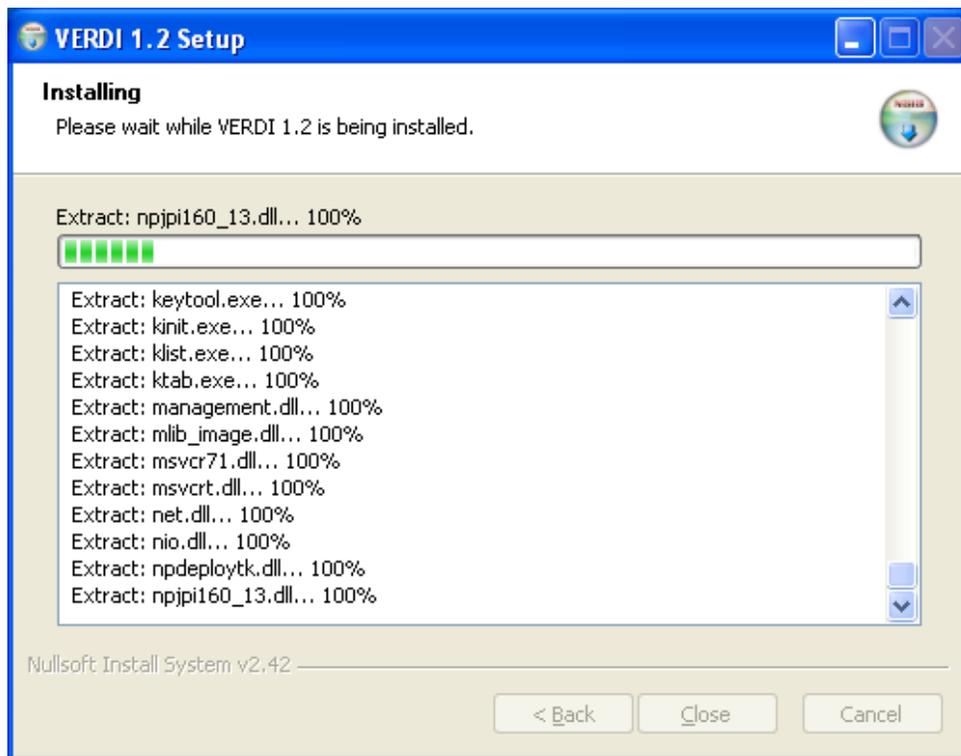
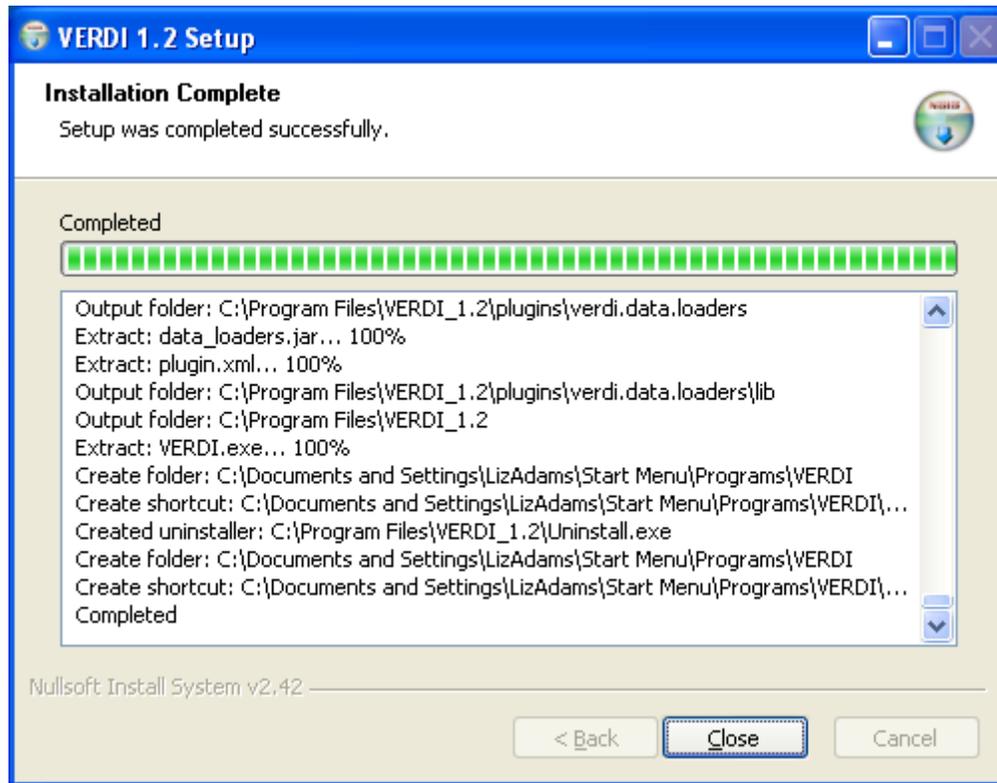


Figure 3-5 File Extraction



**Figure 3-6 Installation Complete**

### 3.4 Setting VERDI Preferences

The VERDI installation package contains a file called `config.properties`. On Windows, this file is copied into the `verdi` subdirectory of your `USERPROFILE` directory (e.g., “C:\Documents and Settings\yourusername\verdi”). Users are encouraged to edit this file to specify default directories for saving files, for placing the location of configuration files, and for saving project files. An example of the contents of a Windows version of this file is:

```
verdi.dataset.home=D:\\VERDI
verdi.project.home=C:\\Documents and Settings\\lizadams\\verdi
verdi.config.home=D:\\Program Files\\VERDI 1.2\\config
```

On Linux, this file is copied into the `$USER_HOME/verdi/` subdirectory. An example of the contents of a Linux version of this file is:

```
verdi.dataset.home=/home/user_a/verdi/data
verdi.project.home=/home/user_a/verdi/proj
verdi.config.home=/home/user_a/verdi/conf
```

**The items in the `config.properties` that is installed with VERDI are commented out, and will need to be uncommented by removing the starting ‘#’ sign if you do not want to specify the directories each time a file is loaded or saved. Example settings that are provided in the default file show how to specify the paths to these locations, depending on whether the**

installation is for a Windows or Linux platform. The `dataset.home` setting specifies the default location from which to load datasets. The `project.home` setting specifies the default location from which to load and save projects. The `config.home` setting specifies the default location from which to load and save plot configurations. Note that VERDI stores the most recently used directory for each of these functions and will go to that directory when you repeat the load or save.

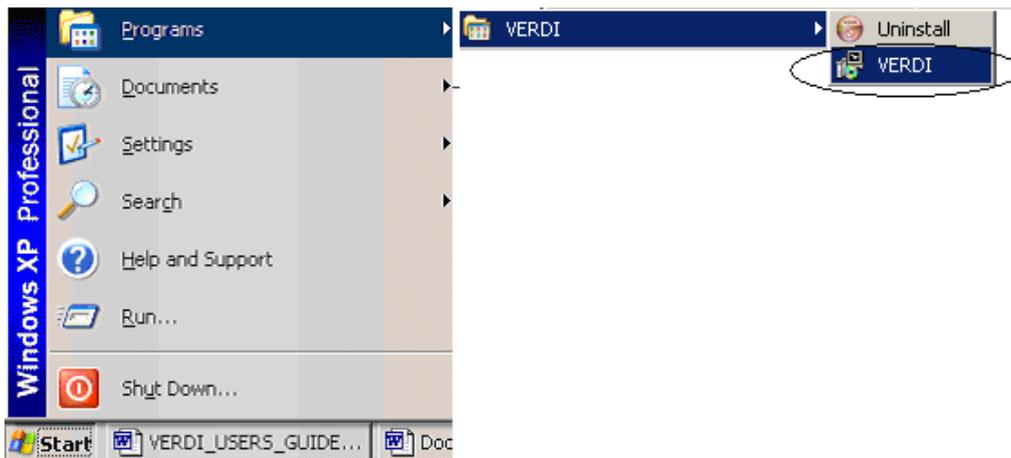
## 4 Starting VERDI and Getting Your Data into VERDI

### 4.1 Starting VERDI

#### 4.1.1 Windows

To start VERDI, open the **Start** menu, then highlight the **Programs** directory, followed by the **VERDI** directory, and then select the **VERDI icon**, as shown in Figure 4-1.

Figure 4-1 Starting VERDI in Windows



#### 4.1.2 Linux and Other non-Windows JRE™ 6 Supported System Configurations

To start VERDI from Linux and other non-Windows JRE™ 6 Supported System Configurations, find the directory where VERDI was installed, then run the `verdi.sh` script.

### 4.2 Main Window

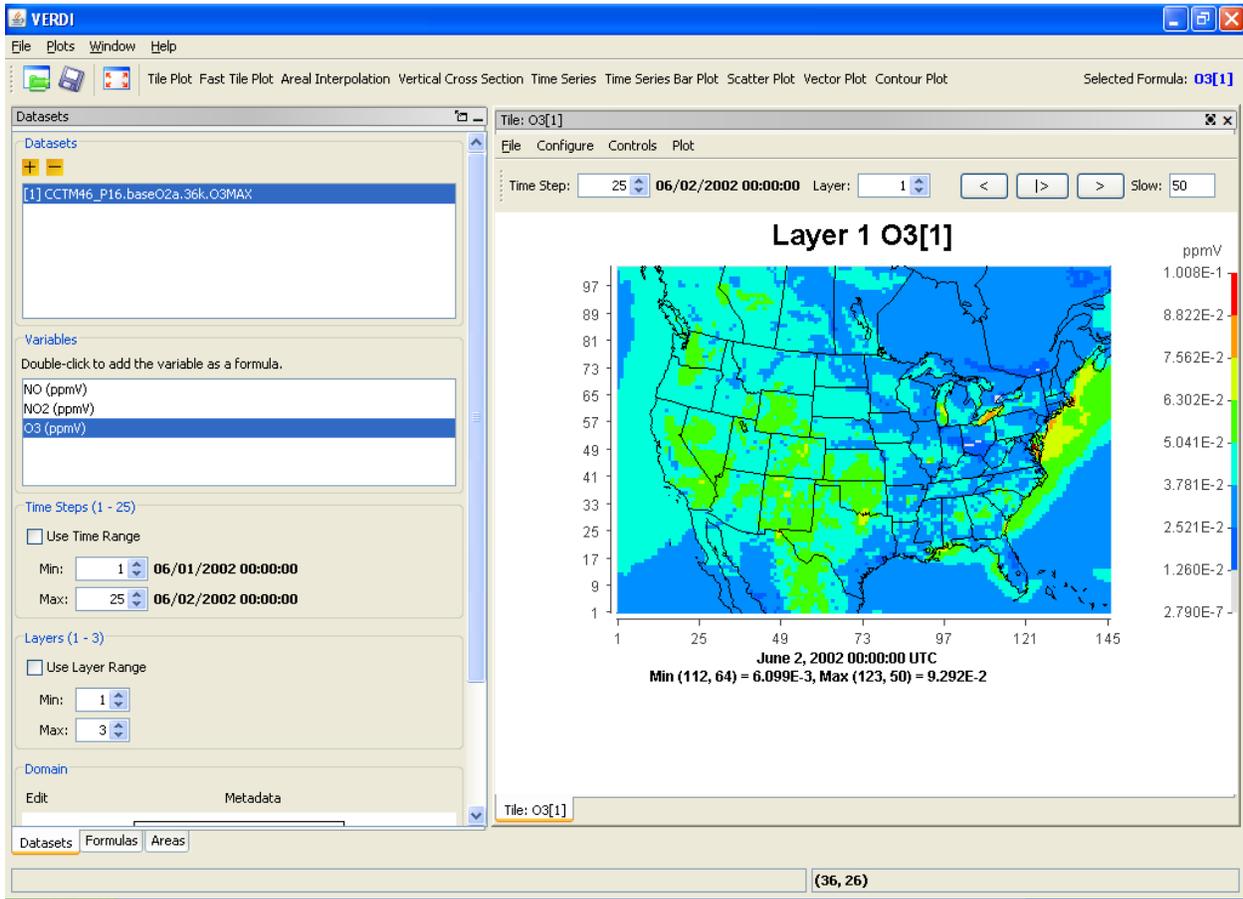
After starting VERDI, the main window will come up on the screen (Figure 4-2). At the top of the main window, there is a menu bar with the main window options (**File, Plots, Window, and Help**). Below the menu bar there are three icons that are shortcuts to some of the options

available in the Main Window Menu Bar; the first is an **Open Project** icon, the second is a **Save Project** icon, the third is an icon that allows you to **Undock All Plots**. These shortcuts and the options available in the Main Window Menu Bar are discussed further in Section 5, “Navigating VERDI’s Main Menu Options.” Next to these three shortcut icons are buttons that list all of the available plot types. To the right of all the plot buttons the Selected Formula is listed. The Selected Formula refers to the formula that has been selected in the **Formula** pane (discussed briefly below and in detail in Section 7), and that will be used to create plots. Below the icons and plot buttons, the window is divided into two main areas: a parameters area on the left side and a plots area on the right side. The parameters area contains three tabbed panes:

- The **Datasets** pane is used to load in the dataset files that you want to work with in this session (this is discussed in Section 6). Once the datasets are loaded, VERDI automatically displays the lists of variables that are in the datasets. To see the variables in a dataset, click on the dataset, and the variables will be displayed in the **Variables** panel underneath the list of datasets. In Version 1.05 and later, if you double click on the name of a variable listed on the variables panel, the variable will automatically be added as a formula on the **Formula** pane and will be the default formula for new plots that are created.
- The **Formula** pane is used to create a formula that refers to the variable and the dataset that you are interested in plotting (see Section 7 for more information). All plots in VERDI are generated from formulas. A formula can be as simple as a single variable from one dataset or it can be an equation that uses variable(s) from one or more datasets.
- The **Areas** pane is used to load area files that are used to create areal interpolation plots (see Section 8 for more information). Area files are defined as shapefiles that contain area features such as watersheds and counties, or any other shapefile that consists of a set of closed polygons.

Any plots that are created are shown in the plots area on the right-hand side of the main window. These plots can be placed into their own movable windows using Plots>Undock all Plots on VERDI’s main menu, as discussed in Section 5.2. The fast tile plot option was added in VERDI version 1.1. This will replace the current tile plot, once it is fully operational and configurable. The functions that are currently enabled for fast tile plots are described in Section 10, “Creating Plots.”

**Figure 4-2 VERDI Main Window**



### 4.3 Floating the Dataset and Formula Panes

The **Formula**, **Dataset**, and **Areas** panes can each be configured to float, to allow you to position them alongside one another. To allow a pane to float, click the icon at the top of the pane that looks like a rectangle with an arrow pointing out of the upper left corner. You can then click on the pane and move it to the left of the VERDI main window. This is useful when you are entering a formula in the **Formula** pane, if you have difficulty remembering the variables that are in a loaded dataset. To re-dock a window or pane, click on the icon in the upper right corner. The icon looks like a circle enclosed in a square. This will return the floating pane back to where it usually lives within the main window.

## 5 Navigating VERDI's Main Menu Options

Figure 5-1 shows a graphic of the main menu options that are available on the top menu bar in VERDI's main window (Figure 4-2 above). These options are discussed in detail below.

**Figure 5-1 VERDI Main Menu Options**

File	Plots	Window	Help
Open Project	Undock All Plots	Areas	Verdi Help Docs
Save Project	Animate Tile Plots	Datasets	About
Save Project As		Formulas	
Exit			

## 5.1 File Menu Options

### 5.1.1 Open Project

**Open Project** retrieves projects that were saved during a previous session (using the two **Save Project** options described next). Note that when you use a saved project, *it is very important* to load that project into VERDI *before* you load any additional datasets or create any additional variables/formulas. If you load a dataset that is not part of the previously saved project and then try to open a previously saved project, you will get a message that says “All currently loaded datasets will be unloaded”, and will be asked if you want to continue.

### 5.1.2 Save Project

The **Save Project** and **Save Project As** options save dataset lists and associated formulas as a “project” for later re-use.

Note that plots are not saved with a project; only datasets and formulas are saved. If you wish to save a plot configuration for later use, see Section 11.2.2, “Loading and saving configuration.”

## 5.2 Plots Menu Options

VERDI opens a single window for plots, to the right of the **Dataset**, **Formula**, and **Area** panes. As plots are created (each in its own sub-window), the most recent plot is displayed on top of previously created plots. Each plot has a tab beneath it listing the type of plot and the formula used to create it. If you want to view a previously created plot, select the tab associated with its sub-window underneath the current plot, and the desired plot will be brought to the front.

### 5.2.1 Undock all plots

As with the **Dataset**, **Formula**, and **Area** panes (Section 4.3), plot sub-windows can be undocked so that you can move them into separate windows. This allows side by side comparisons of plots. Note that undocking is performed only on previously created plots; any newly created plots are docked to the VERDI main window.

### 5.2.2 Animate tile plots

This option opens an **Animate Plots** dialog box (Figure 5-2) that allows you to select one or more plots, select a subset of the time range, and create an animated GIF file. There is also a separate way to create a Quicktime movie instead of a GIF, if desired.

Within the **Animate Plots** dialog box, you can **select plot(s)** to animate by clicking the check box beside each plot name.

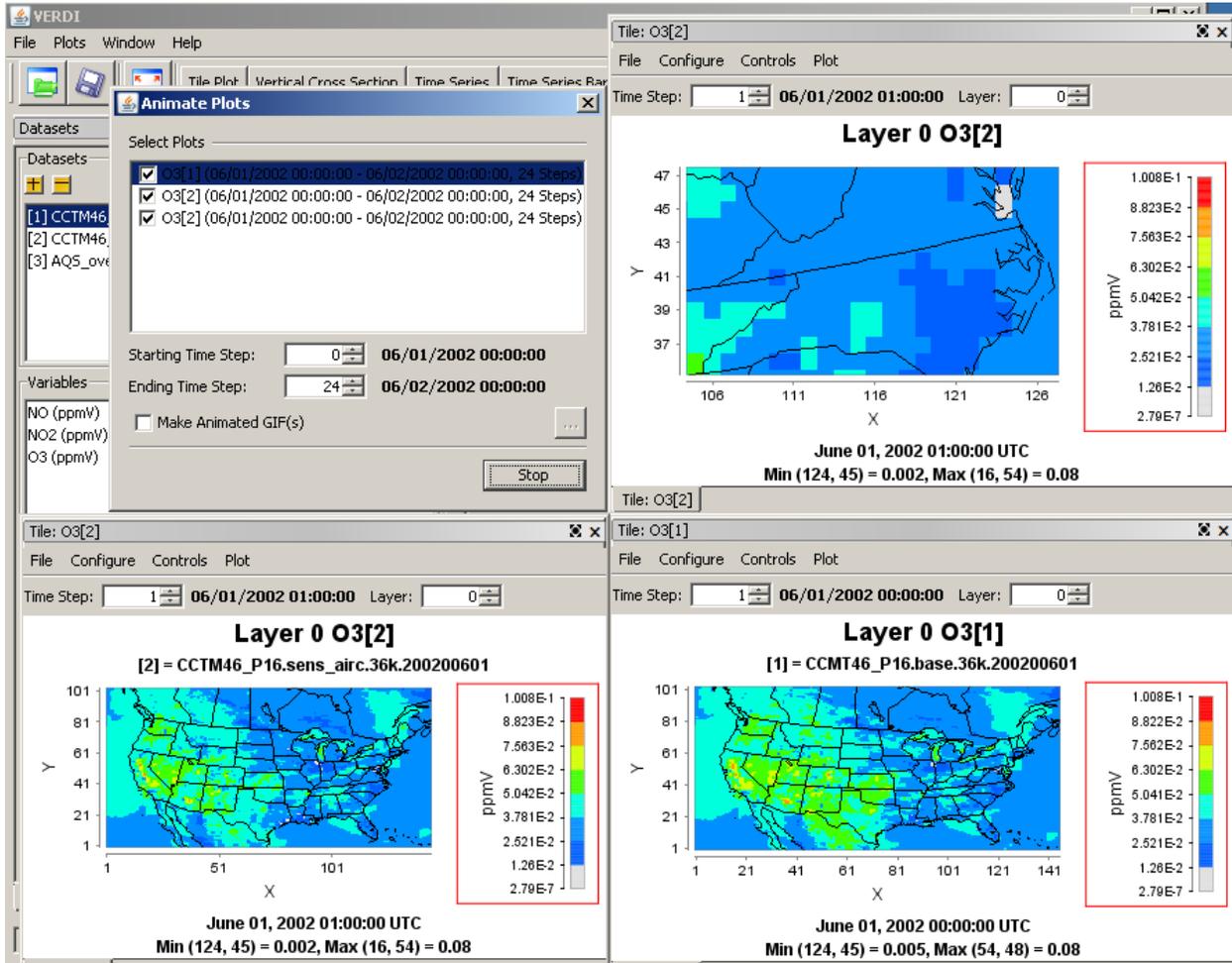
You can choose to animate a single plot, or animate multiple plots synchronously. To view multiple animated plots synchronously, undock the plots (see Section 5.2.1) and move them so that they are located side by side for visual comparison during the animation.

Once a plot has been selected, you can **select the time range** by specifying both the **starting time step** and **ending time step** of the animation.

To create an animated GIF, check the **Make Animated GIF(s)** option in the **Animate Plots** dialog box. In the **Save** dialog box that appears, select the directory in which to store the file and the name to use for the animated GIF, then click the save button. When saving as an animated GIF, when multiple plots are selected, each animated plot will be saved to a separate animated GIF file. For example, if three plots were selected, the animated plots would be saved as <filename>-1.gif, <filename>-2.gif, <filename>-3.gif. You can view the animated GIF by opening the file in a web browser.

Creating a Quicktime movie is also an option, but this is not done through the **Plots>Animate Tile Plots** main menu option. Instead, use the **Plot** menu option found at the top of each individual plot to make a Quicktime movie.

Figure 5-2 Animate Plots Dialog and Tile Plots



### 5.3 Window Menu Options

The **Window** menu provides an alternate way to select windows/panes to be brought to the front, and provides the same function as clicking on the tabs at the bottom of the windows/panes.

#### 5.3.1 Datasets and Formulas

Select from the **Window** pull-down menu to bring to the front either the **Areas** pane, **Datasets** pane or **Formulas** pane when those panes are docked.

#### 5.3.2 List of plots

The **Window** pull-down menu is automatically updated each time a plot is created in a VERDI session; each entry in the plot list indicates the type of plot and the formula used. Clicking on a given plot entry brings that plot to the front for viewing. Alternatively, you can bring a plot to the front by selecting the desired **plot tab** underneath the plots area of the main window. As in the menu entries, each **plot tab** is labeled with the plot type and the formula used.

## 5.4 Help Menu Options

The **Help** pull-down menu contains two items that you can use to learn more about VERDI. When you select **VERDI Help Docs**, the VERDI user's manual is displayed in a VERDI Help window. This window is not searchable, but it does allow you to navigate via hyperlinks in the Table of Contents, and to scroll down and read the user's manual. When you select **About** a pop-up window that contains the name of the product, the version number, and the date the software was built is displayed.

# 6 Working with Gridded Datasets

## 6.1 Gridded Input File Formats

VERDI currently supports visualizing files that can be read by the version 4.1 of the netCDF-java i/o library (<http://www.unidata.ucar.edu/software/netcdf-java/>). VERDI has been tested with files using CMAQ Input/Output Applications Programming Interface (I/O API) format, and the WRF netCDF format.

The CMAQ I/O API was designed as a high-level interface on top of the netCDF Java library. (see <http://www.baronams.com/products/ioapi> and <http://www.unidata.ucar.edu/software/netcdf/> for further info). The I/O API library provides a comprehensive 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 self-describing and include projection information within the gridded dataset. See section 12 for additional information on what projections and gridded data formats are supported by VERDI.

netCDF and I/O API files are portable across computing platforms. This means that these files can be read regardless of what computer type or operating system you are using. There are routines available to convert data to these formats or new code can be written and contributed to VERDI for use by the community. Discussion of the I/O API conversion programs and how to use them can be found in Section 13, "I/O API Utilities, Data Conversion Programs, and Libraries." If you write a routine for VERDI to read gridded data from other formats, please consider contributing your code to the user community using sourceforge.net, as described in Section 14.

## 6.2 Example Datasets

Several example datasets are provided under the \$VERDI\_HOME/data directory. These datasets may be used to re-create example plots that are provided in this user guide, including a tile plot with observational data overlay in Section 11.4.3, and the example datasets for the various dataset projections that VERDI supports including LCC, polar stereographic, UTM and Mercator. The data directory currently contains four subdirectories:

1. CAMx – contains sample CAMx dataset and camxproj.txt file
2. hucRegion – contains Hydrologic Unit (HUC) shapefiles for region 3 (southeast US)

3. Model – contains sample WRF and CMAQ I/O API datasets
4. Obs – contains observational dataset created by airs2m3 converter

### 6.3 Adding and Removing a Gridded Dataset

The first step after starting VERDI is to load a gridded dataset. To do this, press the yellow **plus** button at the top of the **Datasets** pane. A file browser (Figure 6-3) allows you to select a gridded dataset for use in VERDI.

After you specify a gridded dataset, VERDI will load header information and display the available variables, time steps, layers, and domain used by the file in the **Datasets** pane (Figure 6-4). (The actual model data are not loaded until later, when plots are created.) To view the variables for a particular dataset that has been loaded, click on the dataset name in the list to highlight it, and the variables will be listed in the panel below.

Datasets can be removed by highlighting the name of the dataset in the dataset list and pressing the yellow **minus** button. Note that although the dataset will be removed, the number that was assigned to that dataset will not be reused by VERDI during the current session.

**Figure 6-3 Open Dataset File Browser**

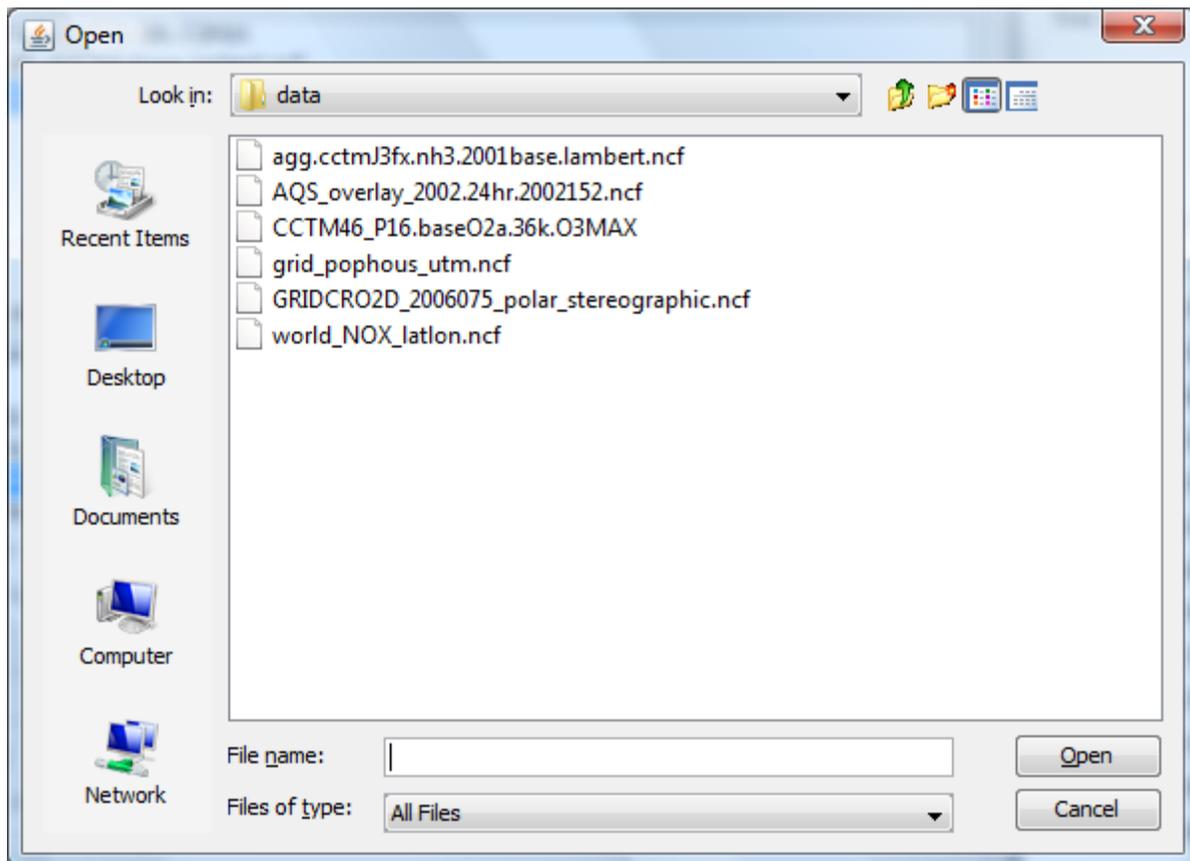
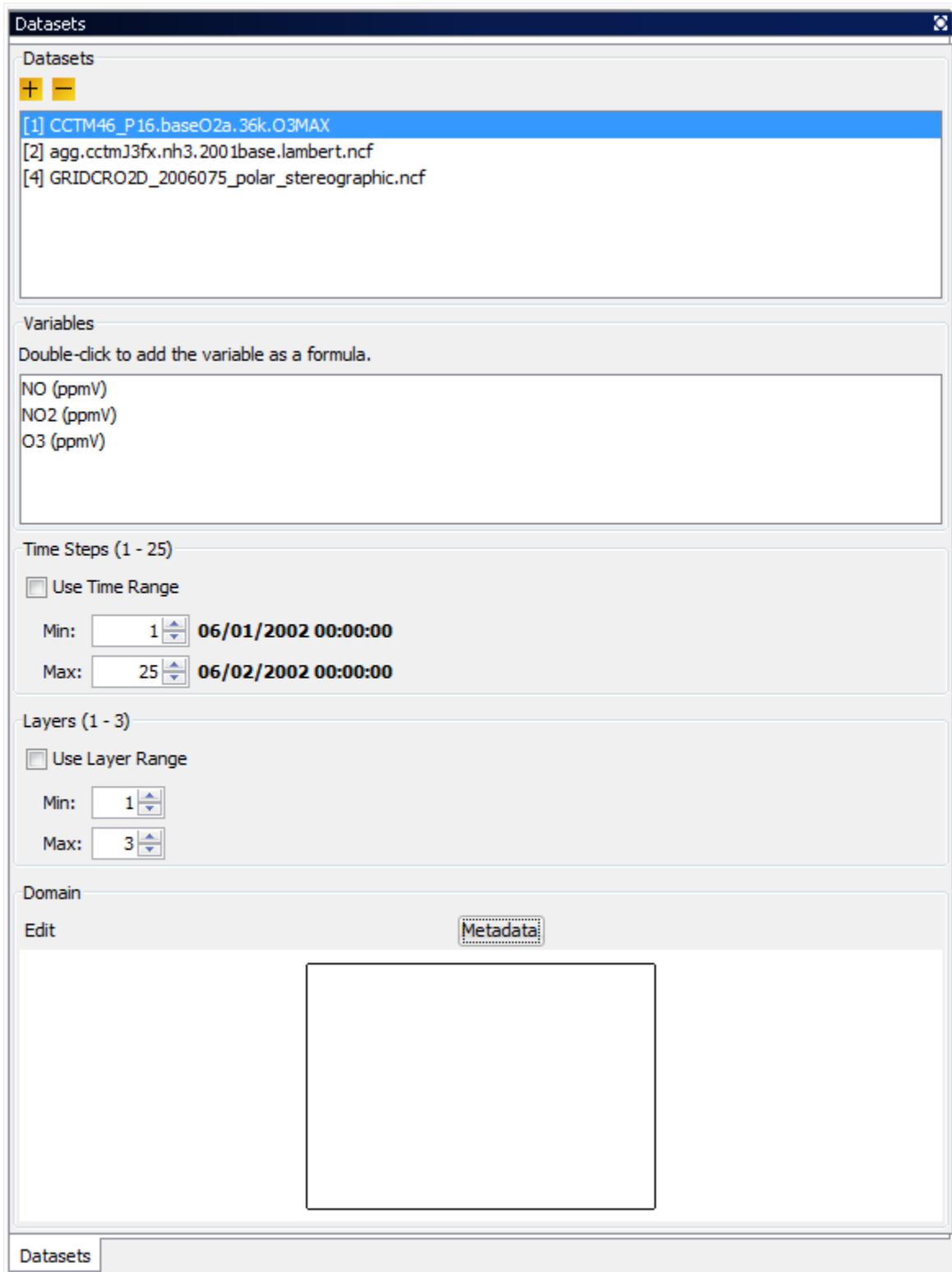


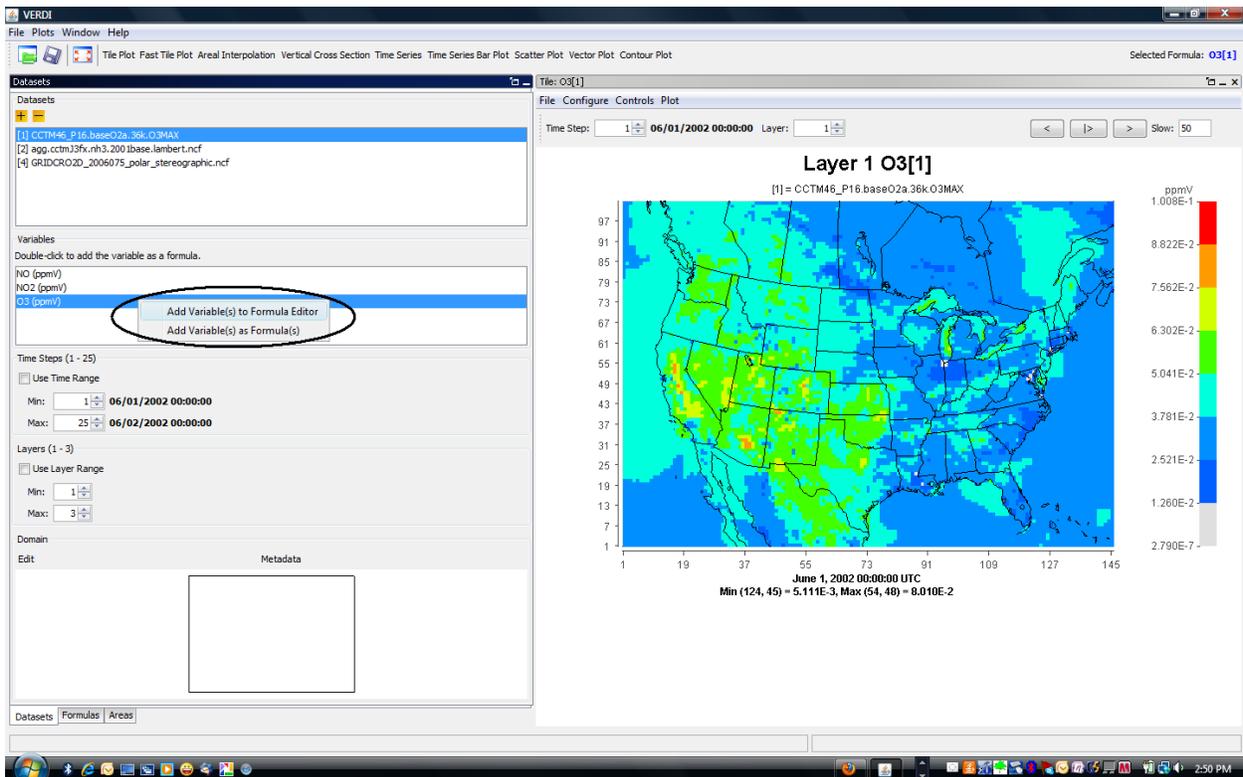
Figure 6-4 Datasets Pane Displaying Information about a Dataset



## 6.4 Variables List

The variables list shows all of the variables contained in a loaded dataset (see the example in Figure 6-4). To display a variables list, select the name of the dataset of interest in the **Datasets** pane. Each of the variables in the list can be used to create a formula in the **Formula** pane that can then be used to create plots. VERDI allows the user to automatically add a formula by double clicking on the name of a variable. Double clicking on a variable will automatically create a formula that contains the variable for the loaded dataset and it will become the default formula for making plots. In addition, you may right click on the name of the variable to show a popup menu as shown in Figure 6-5. From this menu you can either add the variable as a formula, or you can to add it into the formula editor so that it can be used to compose more complex formulas. Formulas are described in more detail in Section 7.

Figure 6-5 Right Click on Variable in Dataset Pane



## 6.5 Time Steps, Layers Panels

The range that is available for the dataset is listed in the Time Steps or Layers Panel in parenthesis next to the label for the panel, e.g. for a dataset that has 5 time steps, the range would be displayed as: Time Steps (1-5). The user can select to use a subset of the full time step range by clicking on the Use Time Range checkbox, and then using the Min and Max spinner controls to set a new minimum or maximum values, for example choosing time step 2 as the minimum time step, and time step 4 as the maximum. When a tile plot is created, it will only display time

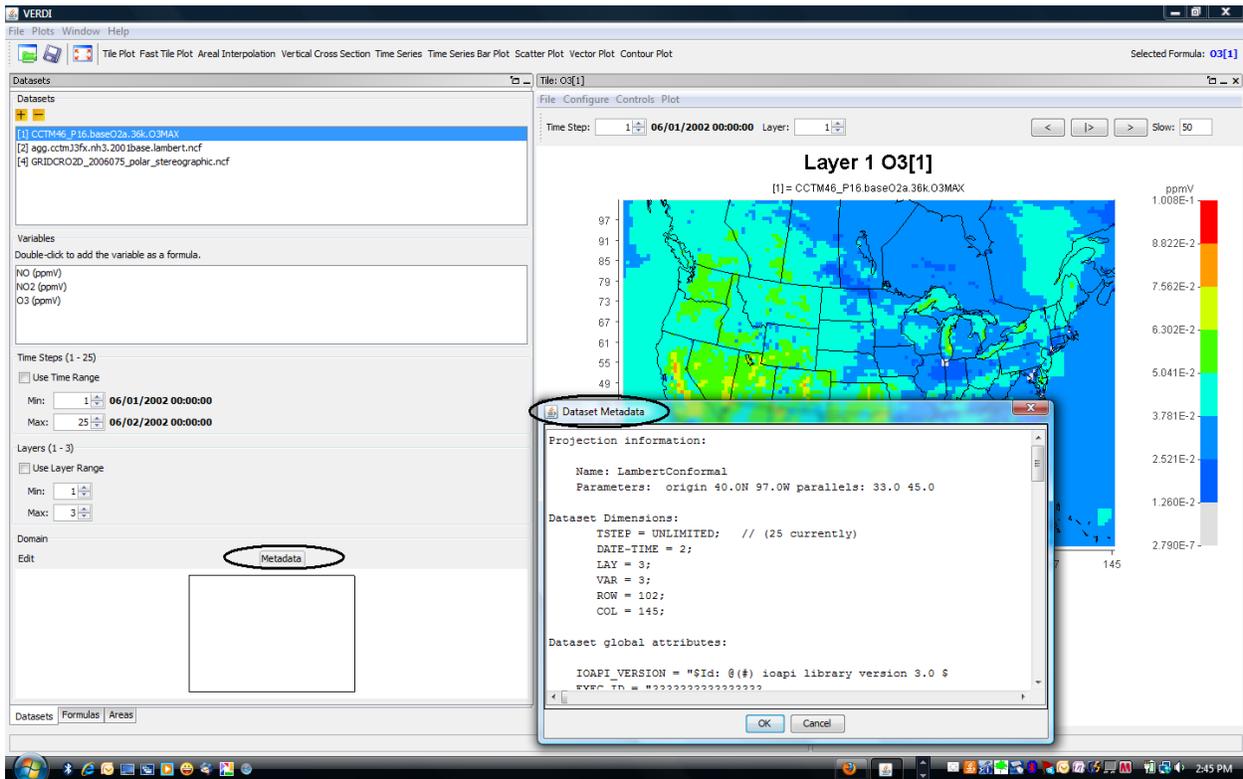
steps 2-4. Detailed instructions for using the **Time Steps**, **Layers**, and **Domain** panels that you see in Figure 6-4 are discussed in Section 9, “Spatial and Temporal Data Subsetting.”

## 6.6 Domain Panel

The domain panel contains an Edit button and a Metadata button. Detailed instructions for using the Edit button are provided in Section 9, “Spatial and Temporal Data Subsetting”.

To display the metadata information about a dataset, click on the **Metadata** button in the **Domain** panel. A window containing the metadata will appear (Figure 6-6). Each dataset includes metadata information that is part of the file header. The metadata provided include the map projection information, the number of time steps in the file, the number of columns and rows, and other information.

**Figure 6-6 Dataset Metadata Information**



## 6.7 Saving Projects

As noted in Section 5.1.2, lists of datasets and formulas can be saved as “projects” using the Save Project option in the **File** pull-down menu on the VERDI main window. Refer back to that section for discussion on saving new projects and loading existing projects. Note again that the plots created in VERDI are not saved with the project.

## 6.8 Configuring Default Directories

A `config.properties` file (discussed previously in Section 3.4) allows the user to customize where VERDI will look by default for the location of datasets and project files. This file is located in the `$USER_HOME/verdi/subdirectory`, and can be edited to specify where datasets and/or project files are saved and stored. An example is as follows:

```
# This file should be put in $USER_HOME/verdi/subdirectory
# Please use double backslash for windows platform or slash for UNIX
like platforms
# Please uncomment the following lines and modify them to suit your
local settings

verdi.dataset.home=D:\\data
verdi.project.home=D:\\programs
verdi.config.home=D:\\config
```

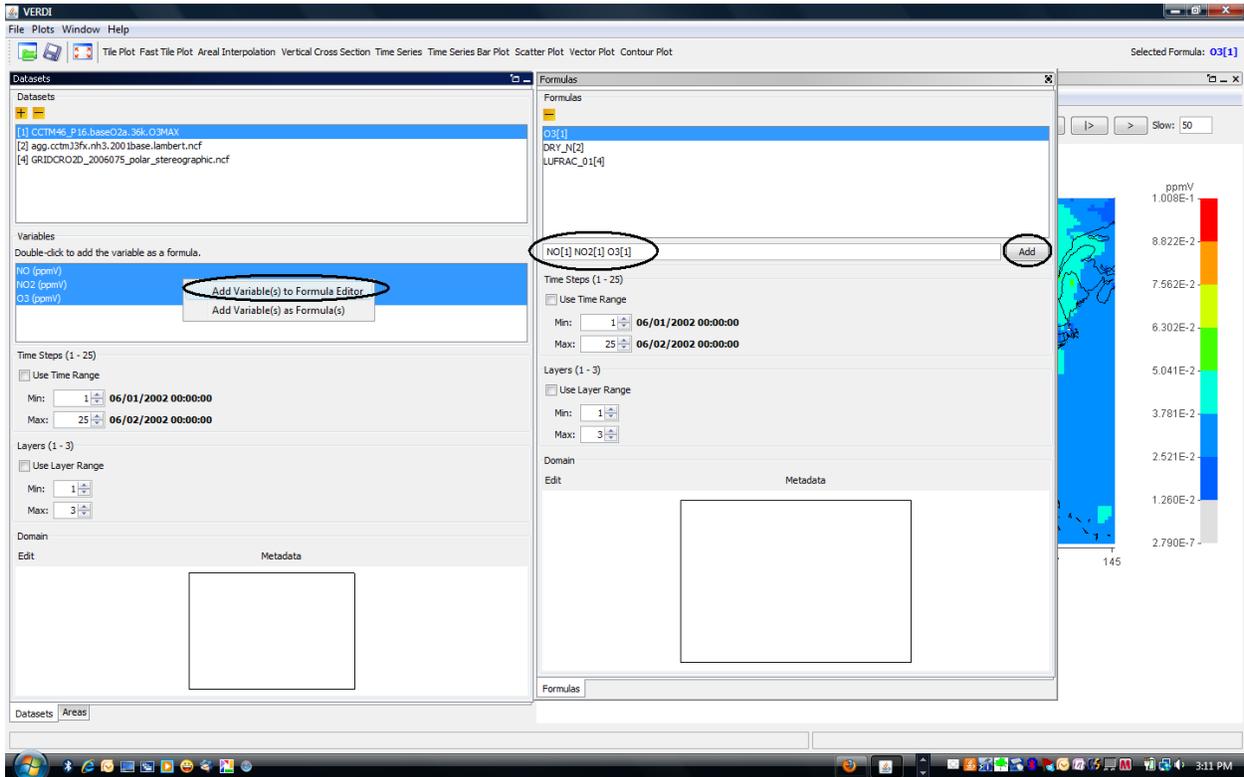
## 7 Working with Formulas

All plots in VERDI are generated from formulas. A formula is used to compare or manipulate variables in one or more gridded datasets. A formula can be as simple as a single variable from one gridded dataset or it can be an equation that uses variable(s) from one or more gridded datasets. Formulas are used to create visualizations that can assist with model performance evaluations, for example, or that help in comparing model results with observations.

### 7.1 Adding and Removing a Formula

After loading the desired gridded datasets, you can use the variables in them to create formulas. To use a variable to create a simple formula, double click on the name of the variable. This will add the formula `<Variable Name>[<Dataset Number>]` to the formula list in the **Formulas** pane—for example, `O3[1]`. To add a variable to the formula editor window, highlight the variable, **right click** on the variable name in the **Datasets** pane, and select **Add Variable(s) to Formula Editor**. To add all or a subset of variables from the **Dataset** pane to the formula editor window, click on the first variable to highlight it, hold the shift key down and click at the last variable that you want to include, then right click and select **Add Variables(s)**. The formulas that are highlighted using this method will be added to the formula editor (Figure 7-1).

**Figure 7-1 Adding Multiple Variables to Formula Editor**



After the variable names are added to the Formula Editor, click on the formula pane and use the cursor and the keyboard to type in the mathematical functions and operators where needed to create a valid formula (see Section 7.2 and Section 16). After the formula has been created in the Formula Editor, click the **Add** button, to place it in the list of formulas available in the **Formula** pane.

To remove a formula from the formulas list, highlight the name in the list and press the yellow **minus** button. Note that removing a formula from the formula list does not remove plots that were created prior to the deletion of the formula.

## 7.2 Example Formulas

To examine the values of ozone in dataset 1, the formula would be “O3[1]”.

To examine the difference in ozone between datasets 1 and 2, the formula would be “O3[1]-O3[2]”.

To calculate the percent difference in ozone between datasets 1 and 2, the formula would be “(O3[1]-O3[2])\*100/(O3[2]+.001)”.

To identify all cells where the ozone concentration exceeds a certain value, you can use the Boolean operators to “screen out” ranges of your data that are of particular interest. A Boolean

expression will evaluate to either True = 1 or False = 0. For example, to plot the cells in which the ozone values in dataset 1 exceed 0.080 ppm, you could use the formula “(O3[1]>0.080)\*O3[1]”. In the resulting plot, each cell where O3[1] exceeds 0.080 will show the value of O3[1] for that cell; for all other cells the value shown will be zero.

The notations that can be used in formulas to represent various mathematical functions, and the order of precedence of these functions, are listed in Section 16, “Mathematical Functions.”

### **7.3 Selecting a Formula for Plotting**

Before creating a plot, a formula must be selected. Check to see which formula is highlighted in the **Formula** pane, or look to the right of the plot buttons above the plots area of the main window to see the selected formula. By default, VERDI designates the most recently added formula as the selected formula. To change the selected formula to a different one in the list, click on a formula in the list on the **Formula** pane, and you will then see it displayed as the selected formula above the plots area.

### **7.4 Saving Formulas**

Both formulas and datasets can be saved using the **Save Project** item in the **File** pull-down menu on the VERDI main window. Saving new projects and loading existing projects was discussed in Section 5.1.

### **7.5 Time Step Range, Layer Range, and Edit Domain**

Instructions for using the **Time Steps**, **Layers**, and **Domain** panels that you see in Figure 7-1 are discussed in the next section, “Spatial and Temporal Data Subsetting.”

## **8 Working with Area Files**

### **8.1 Area File Formats**

Area files are defined as shapefiles that contain area features such as watersheds and counties, or any other shapefile that consists of a set of closed polygons.

The shapefile format (ESRI, 1998) consists of four files.

1. The \*.shp file contains the actual shape vertices.
2. The \*.shx file contains the index data pointing to the structures in the .shp file.
3. The \*.dbf file contains the attributes (e.g., gridded concentrations).
4. The \*.prj file contains the map projection information used for the gridded concentrations.

## 8.2 Example Area File

Shapefiles that contain polygon segments are used by VERDI to interpolate gridded data to geographic boundary regions to create Areal Interpolation Plots. Shapefiles containing state, county, or census block, for example, or any other shapefile containing polygon areas may be used in VERDI to calculate and map formulas to the user-selected geographic regions.

Example on-line data archives for these shapefiles include:

<http://datagateway.nrcs.usda.gov>

<http://www2.census.gov/cgi-bin/shapefiles/national-files>

## 8.3 Adding and Removing an Area File

To load a shapefile, press the yellow **plus** button at the top left corner of the **Areas** pane (Figure 8-1). A file browser (Figure 8-2) allows you to change directories and select a shapefile file for use in VERDI. Click on the shapefile name and click **Next**. The **Open Area** pop-up window will be displayed next which allows you to select the name of the field to read from the file and supply projection information if needed. Use the pull-down menu and click on the Name Field (Figure 8-3) to be used. If a \*.prj file is not provided with the dataset, the user is required to specify the projection information. A pull-down menu will prompt the user to select the coordinate system used in the file: either the Geographic (lat/lon) coordinate system or a Projected (x/y) coordinate system (Figure 8-4). If the user selects Projected, an additional pull-down menu asks what type of projection, and a text box allows users to provide the required information to do the projection (Figures 8-5 and 8-6). After the Name Field and the Coordinate System have been specified, select **Finish**. The resulting plot will be in the same projection as the gridded information used in the plot.

## 8.4 Areas List

The shapefile name(s) will be listed in the top panel of the **Areas** pane, and the name fields for the polygons provided in the shapefile(s) will be listed in the panel underneath (Figure 8-7). The actual model data are not loaded until later, when the Areal Interpolation plots are created. As additional shapefiles are added, the name fields associated with each shapefile are appended to the bottom of the Areas list. Use the scrollbar on the right side of the **Areas** pane to view the additional name fields that are available. To remove a shapefile, click on the name of the shapefile and press the yellow **minus** button at the top left corner of the **Areas** pane.

## 8.5 Areal Interpolation

When the user selects the Areal Interpolation Plot, the selected formula is interpolated over the polygon areas that are listed in the **Areas** pane. To select a subset of the polygon areas, and view the average and total values for selected formulas, see Section 10.3: Areal Interpolation Plot.

Figure 8-1 Areas Pane

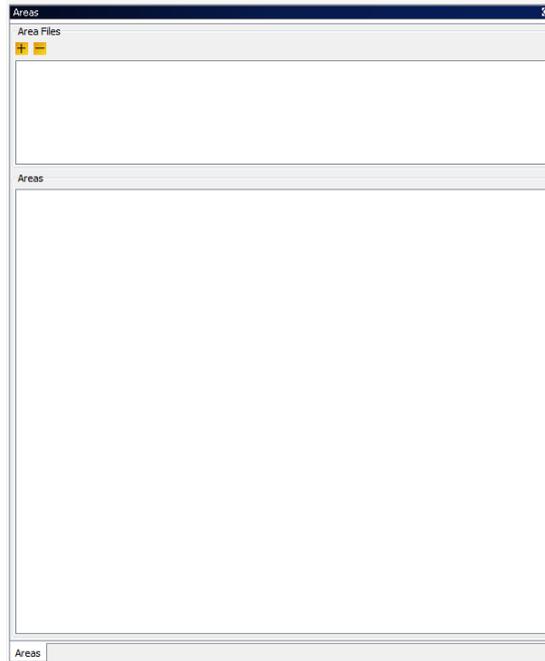
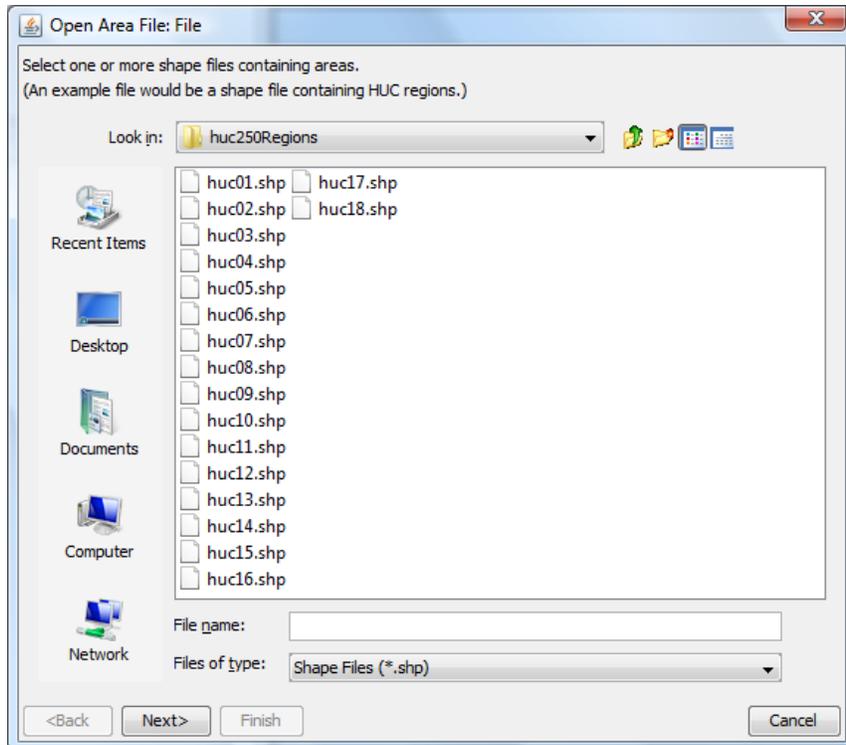
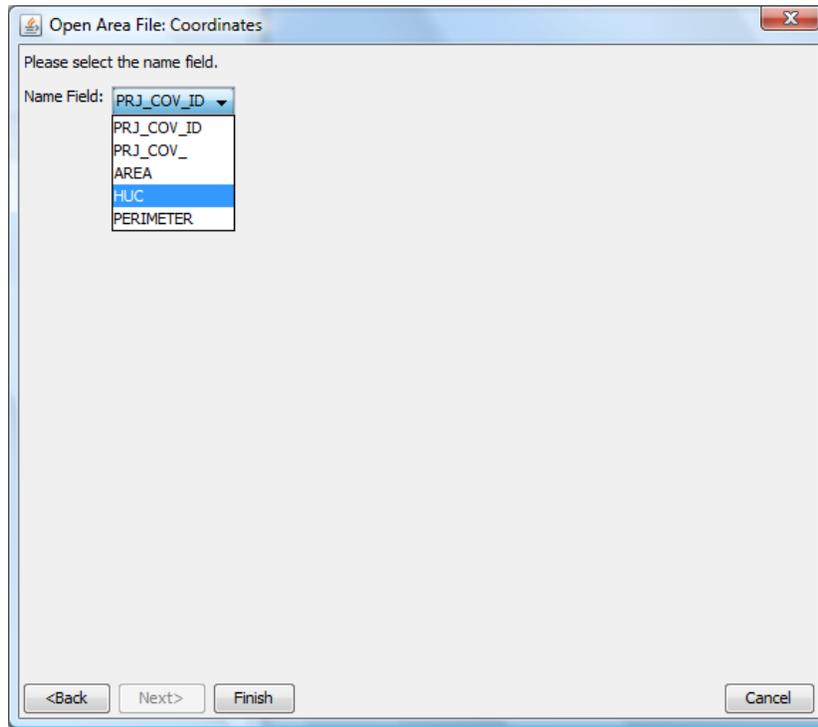


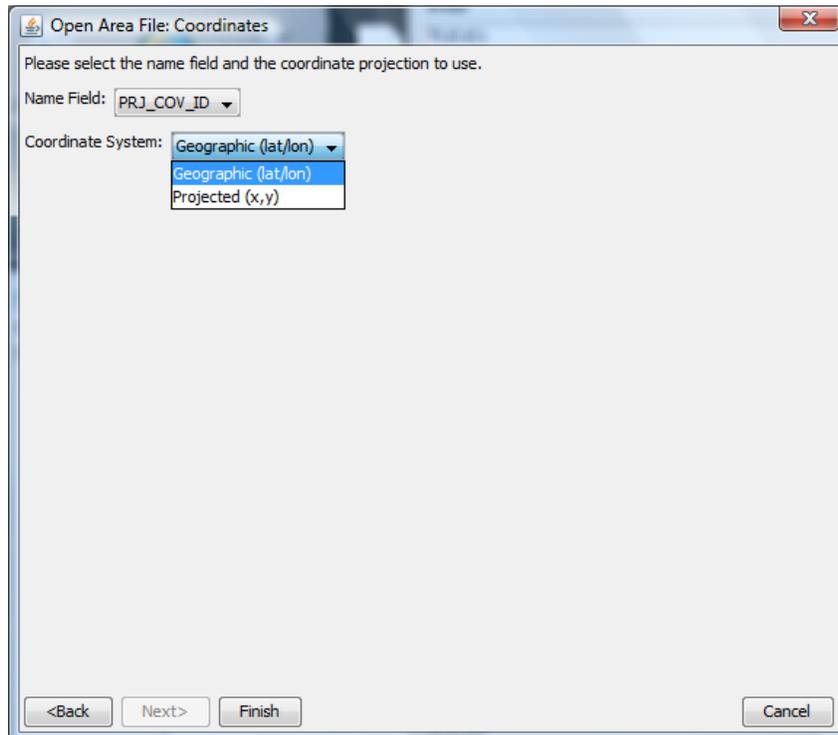
Figure 8-2 Open Area File Browser



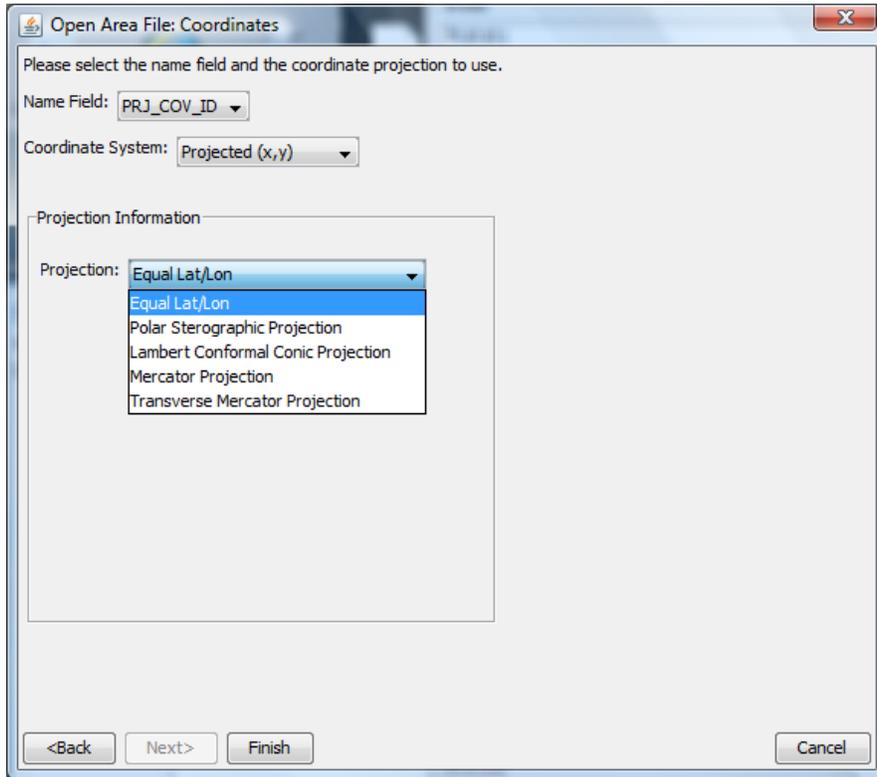
**Figure 8-3 Open Area File: Select Name Field**



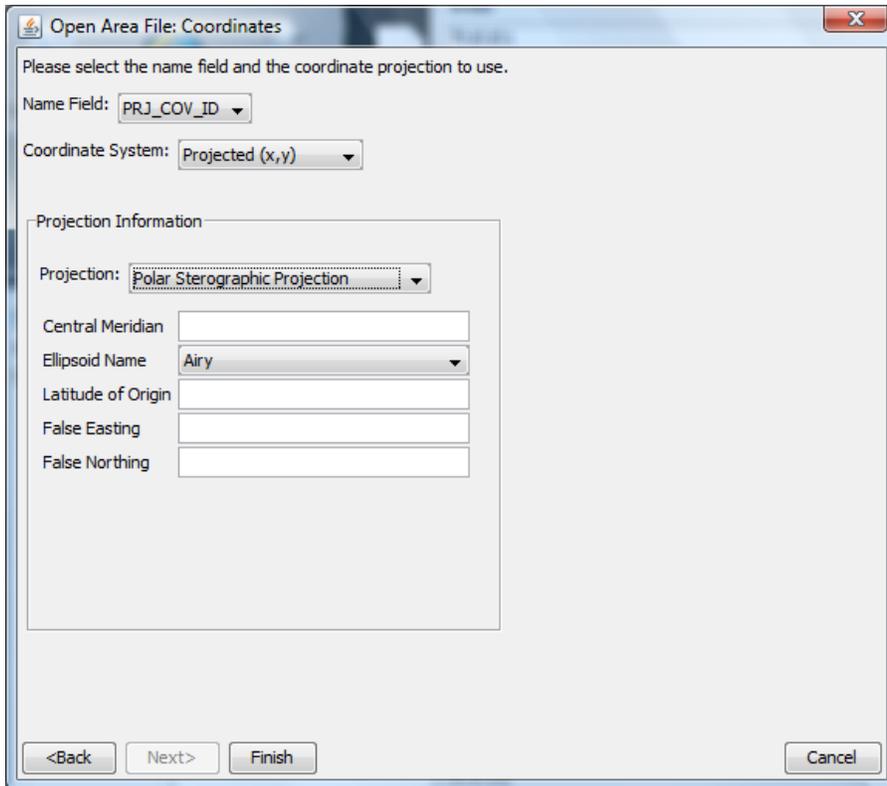
**Figure 8-4 Coordinate System**



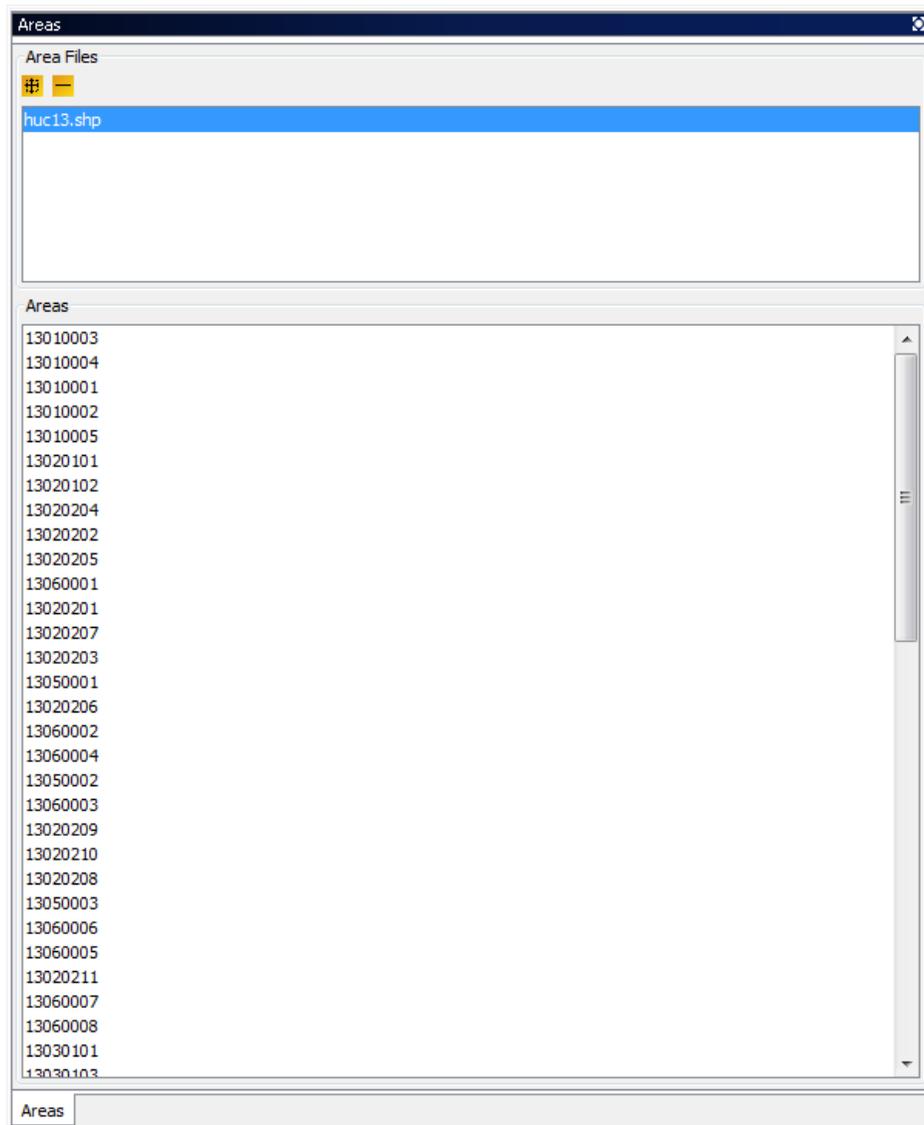
**Figure 8-5 Projection Information**



**Figure 8-6 Additional Data Fields appear depending on projection selected.**



**Figure 8-7 Area Name Fields available for Shapefile**



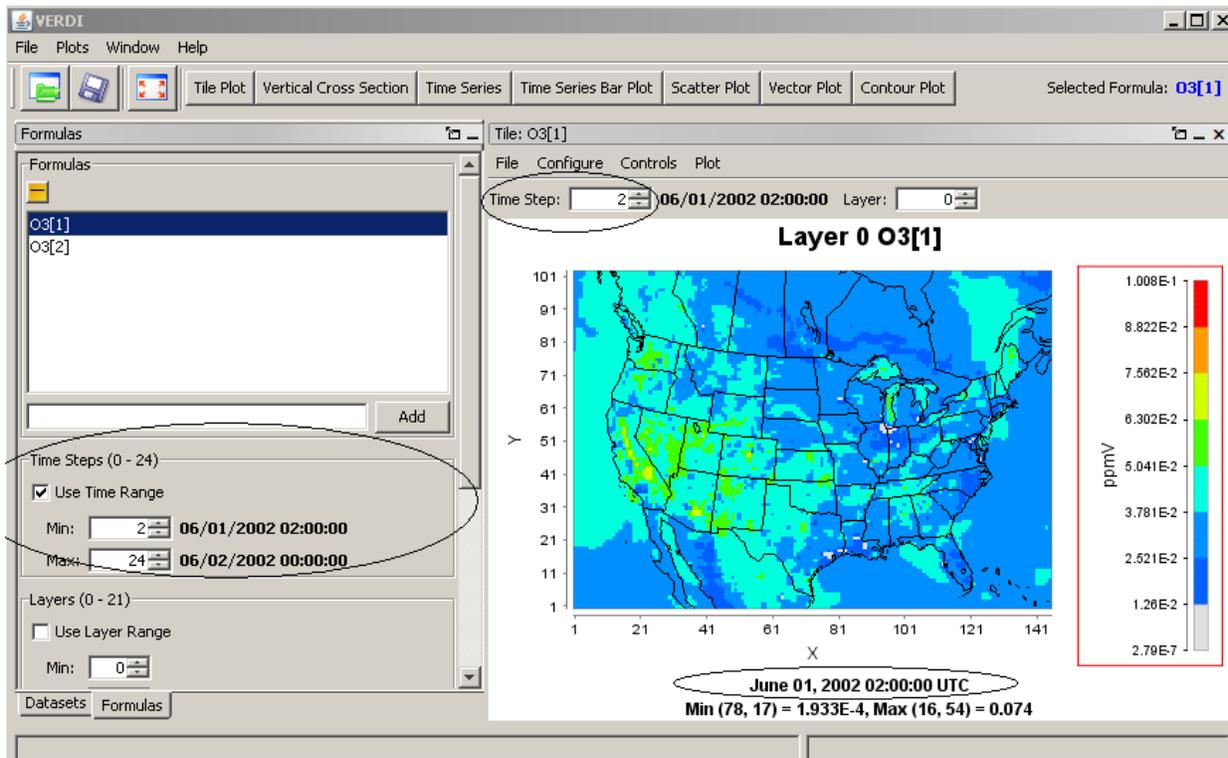
## 9 Spatial and Temporal Data Subsetting

Both the **Dataset** pane and the **Formula** pane include the three panels discussed in Sections 9.1 through 9.3: **Time Steps**, **Layers**, and **Domain**. Section 9.4 then discusses the precedence rules for subsetting data that determine which pane’s subsettings (**Dataset** or **Formula**) take priority.

### 9.1 Specify Time Step Range

Information about the range of time-step values included in a dataset is displayed in the **Time Steps** panel (Figure 9-1). The maximum time-step range that can be used for a dataset or formula is specified in the **Min** and **Max** spinner controls. You can use these controls to select a subset of the available time-step range for plotting. Check the **Use Time Range** box above the spinner controls to tell VERDI to use the time-step range values you have specified when it creates a plot. By default, a plot will initially display data for the minimum time step that was specified in the **Time Steps** panel. The range of the time steps available in the **Time Step** spinner control at the top of the plot will reflect the subset of time steps specified when the **Use Time Range** box is checked. The date and time of that time step are shown below the plot. Subsetting (also called cropping) a dataset’s or formula’s time-step range will affect any plots made; see Section 9.4 to learn the precedence rules.

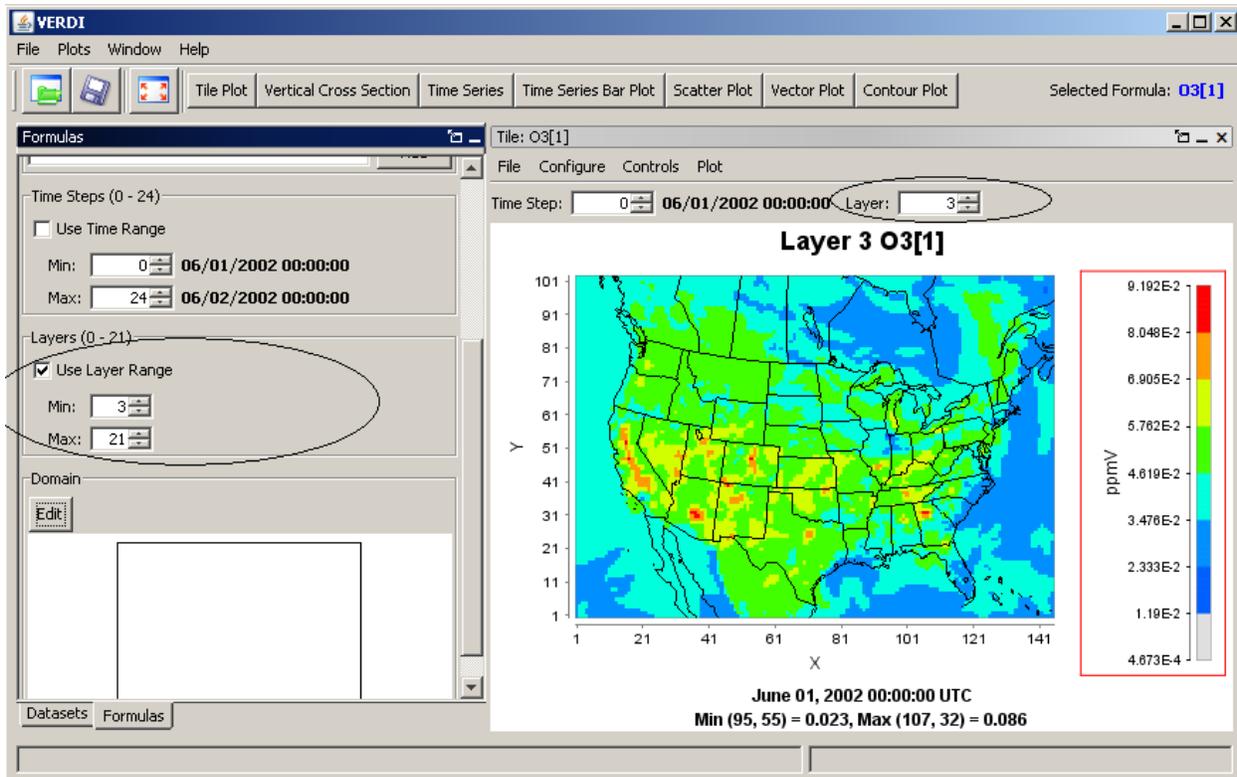
Figure 9-1 Specify Time Step Range



## 9.2 Specify Layer Range

Information on the range of vertical model layers included in a dataset is displayed in the **Layers** panel (Figure 9-2). You can use the **Min** and **Max** spinner controls to select a subset of the available layer data for plotting. Check the **Use Layer Range** box above the spinner controls to tell VERDI to use the layers you have specified. By default, a plot will initially display data for the minimum layer chosen in the **Layers** panel. The range of the layers available in the **Layer** spinner control at the top of the plot will reflect the subset of layers specified when the **Use Layer Range** box is checked. Subsetting a dataset's or formula's layer range will affect any plots made; see Section 9.4 to learn the precedence rules.

Figure 9-2 Edit Layer Range in Formula Pane

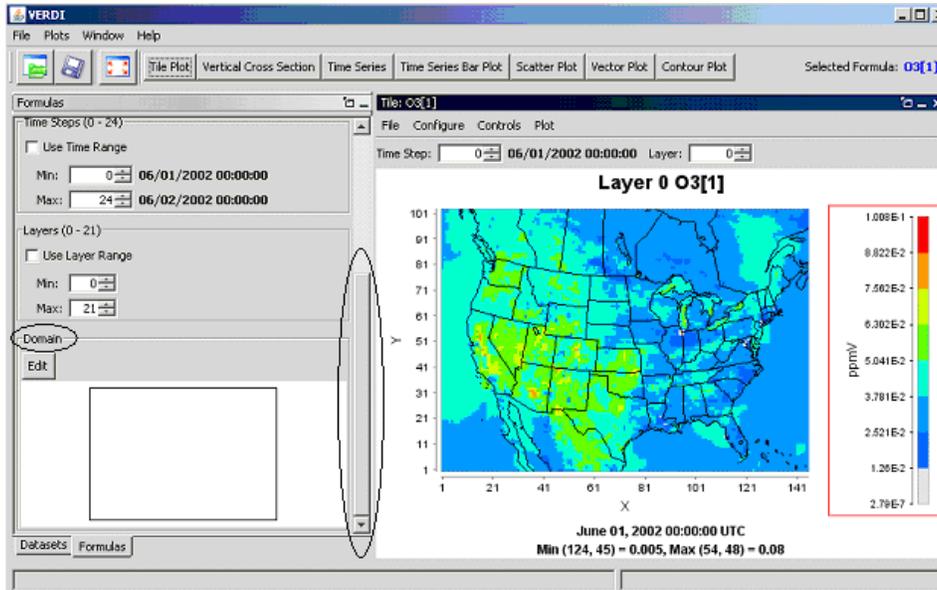


## 9.3 Specify Domain Range

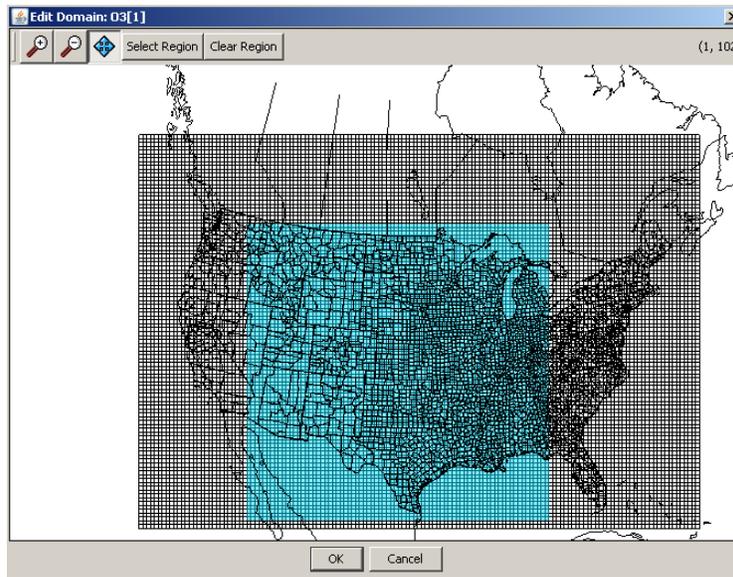
Datasets contain data for cells over a particular geographic area. In the VERDI program this area is referred to as a “domain.” By default, the entire domain contained in a dataset is used in creating plots. The **Edit Domain** dialog box can be used to select a subset of this domain for plotting. To access the **Edit Domain** dialog box, first use the vertical slide bar to bring the **Domain** panel (located below the **Layers** panel) into view if it is not already visible (Figure 9-3). To select a subset of the domain, press the **Edit** button. The **Edit Domain** dialog box will appear (Figure 9-4). If the area of interest is large, background data such as state and county outlines will be shown on the map. The **magnifying glass** buttons above the map can be used to zoom in

and out on the map. Use the **pan** icon to drag the map to a position where the desired area is displayed. To select the desired cells, first press the **Select Region** button. You can then drag a box around the area of interest; selected cells will appear in blue. To clear out the selected area, use the **Clear Region** button. When you are satisfied with the domain subset you have chosen, click the **OK** button. Subsetting a dataset's or formula's domain range will affect any plots made; see Section 8.4 to learn the precedence rules.

**Figure 9-3 Using the Slider to View the Domain Panel**



**Figure 9-4 Edit Domain Dialog Box**



## 9.4 Rules of Precedence for Subsetting Data

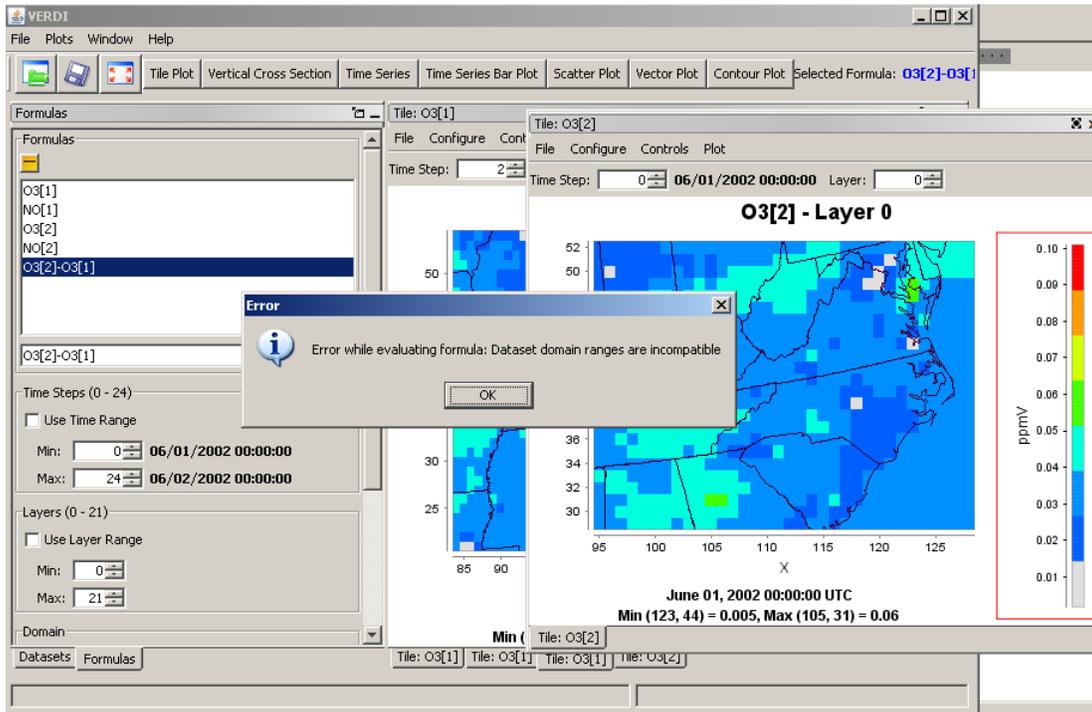
Because both the **Dataset** pane and the **Formula** pane have the **Time Steps**, **Layers**, and **Domain** panels described above, precedence rules were established that determine which pane's subsettings take priority. It is important to understand these rules.

- A subset of data specified in the **Dataset** pane takes precedence over any subset specified in the **Formula** pane. At the time of this release, the range displayed in the **Formula** pane does not get updated when a subset range is selected on the **Datasets** pane. For example, if a dataset has a full time-step range of 0-48, and a time-step range of 2-40 is selected on the **Dataset** pane, then the 0-48 time-step range that is listed for the formula in the **Formula** pane is not applicable. When a plot is then created, the time-step range subset you chose in the **Dataset** pane (2-40) is displayed.

If no subsetting has been done for a particular data type (time steps, layers, or domain) in the **Dataset** pane, then any subsetting done for that parameter in the **Formula** pane will take effect.

- When a subset of data is requested in the **Dataset** pane, this deactivates the ability to further subset the data for this range in the **Formula** pane. For example, if the data are subset in the **Dataset** pane to a layer range of 2-7, and then in the **Formula** pane you attempt to further subset the layer range to 4-5, the further subsetting will be ignored and the plot will be created with a layer range of 2-7.
- When the user wants to create formulas using variables from multiple datasets, it is important to use the **Formula** pane to subset the domain range (if such subsetting is needed). The user should not try to create matching subset domain ranges for multiple datasets using the **Dataset** pane for each dataset, because if all of the subsetted domains do not match exactly, an error will occur when the user attempts to create a formula using variables from those datasets (Figure 9-5). Using the **Formula** pane instead to do the domain subsetting allows the user to select a subset of the domain to be applied for all dataset variables used in the selected formula.

**Figure 9-5 Error obtained when incompatible subset domains are created using the Dataset pane**



## 10 Creating Plots

After creating a formula, you are ready to view some plots. The available plot types are shown on the buttons at the top of the VERDI main window: tile plot, fast tile plot, vertical cross section plot, time series plot, time series bar plot, scatter plot, vector plot, and contour plot. All of these are described in this section. Once you have selected (highlighted) a formula in the list of formulas you have created in the **Formula** pane, you generate each plot type by clicking on the appropriate button using your left mouse button. You can also see the selected formula to the right of all of the plot buttons. If additional information is required in order to generate the plot you have requested, a dialog box appears to prompt you for that information.

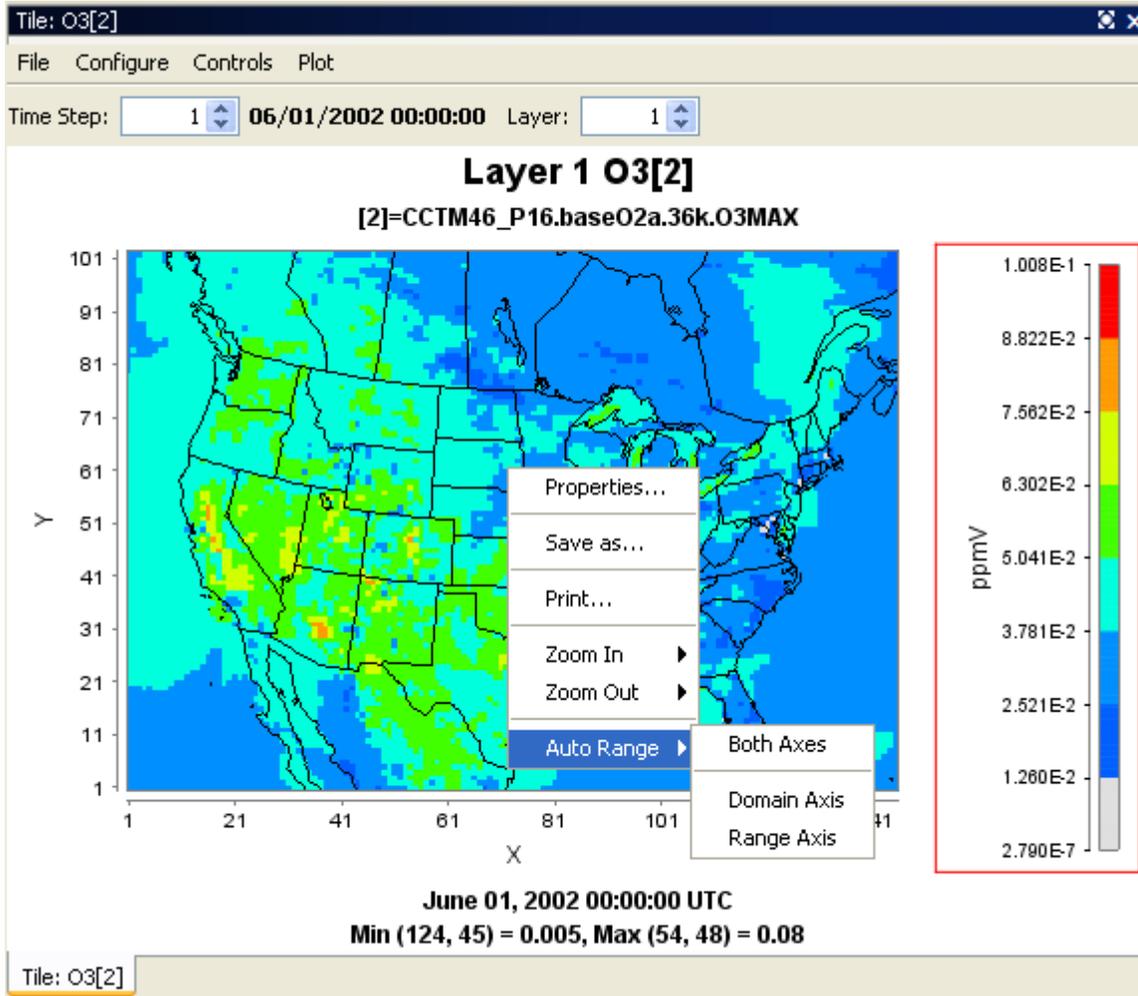
Each plot contains its own menu bar at the top that has options for configuring and exploring the plot; the four pull-down menus there are **File**, **Configure**, **Controls**, and **Plot**. The options for each of these menus are described in more detail in Section 11, “Configuring Plots Using the Plot Menu Bar.”

### 10.1 Tile Plot

**Tile plots** show a color contour of the selected formula. Figure 10-1 shows a tile plot created in VERDI. The legend on the right side of the window shows the range of values contained in the data displayed. The current time step can be changed using the **Time Step** spinner control above the plot. When you make a change, this is reflected in the information below the x-axis label on

the plot. You can use the **Layer** spinner control to change the current layer. To access the ability to zoom in and out, use the mouse to right click on the plot. Zooming and other plot configuration options are discussed further in Section 10.

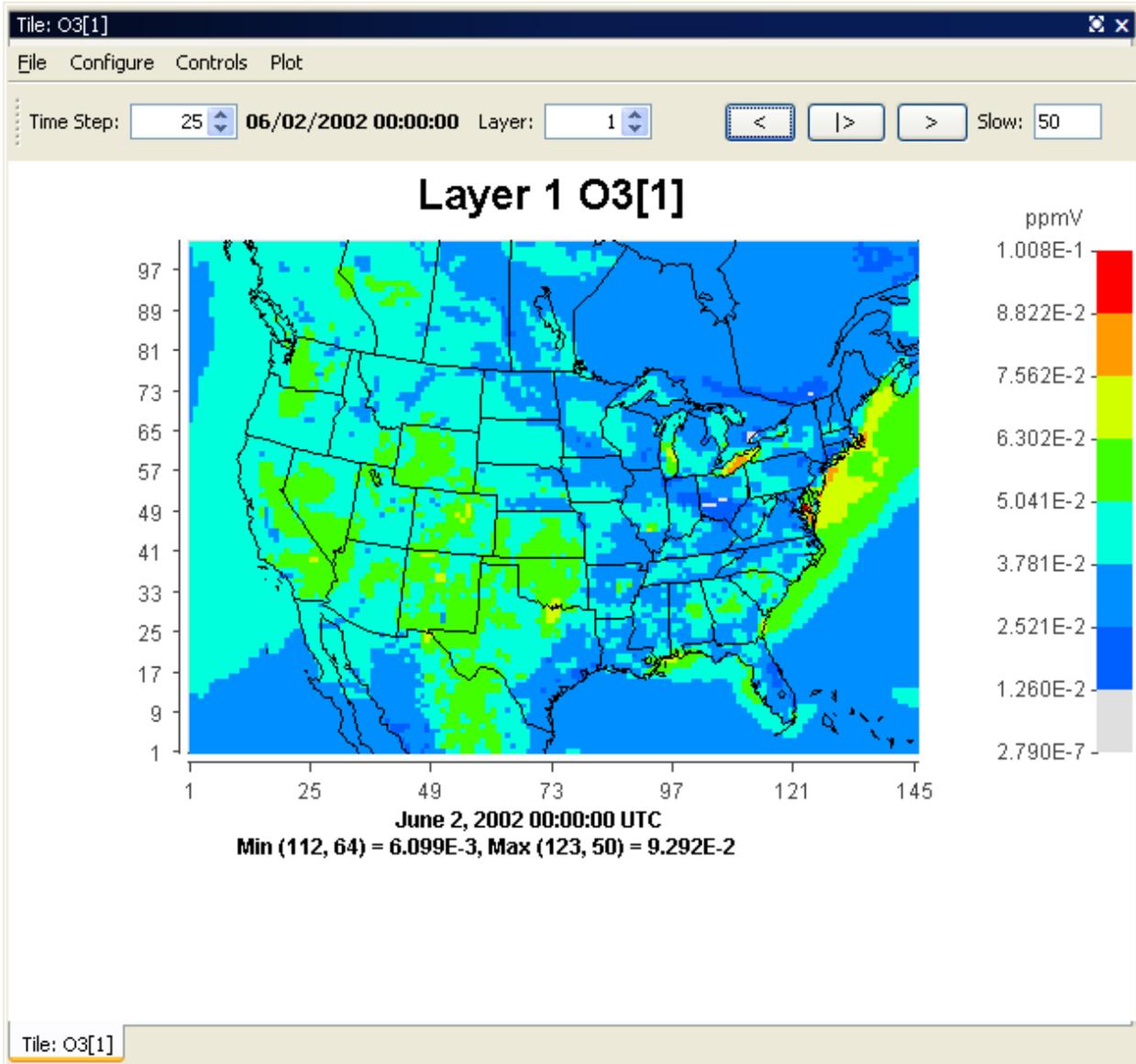
**Figure 10-1 Tile Plot**



## 10.2 Fast Tile Plot

The **fast tile plot** is currently under development, and once it is fully functional, it will replace the previous **tile plot** option. An example of the **Fast Tile Plot** window is shown in Figure 10-2. The motivation for developing the fast tile plot was the slow speed of the original tile plot when VERDI was used on a remote compute server. The fast tile plot draws and animates substantially faster than the original tile plot.

Figure 10-2 Fast Tile Plot



At the top left of the fast tile plot, the spin control can be used to incrementally change the time step by clicking on the up or down arrow. The Layer displayed for the plot can be controlled by clicking on the up or down arrow for the Layer spin control in the top center of the plot. The top right corner of the plot contains buttons that allow the user to use play, reverse, forward, and pause options to control the animation of the plot. A text box labeled Slow allows the user to add a delay between frames by entering a number (delay is in milliseconds); the default delay is 0 milliseconds. A larger plot with multiple map layers would not require the user to add a delay between frames, but a small zoomed-in plot with few map layers will require the user to add at least a 200-millisecond delay between frames, to comfortably view the data. Putting in a larger number will further slow the animation.

The fast tile plot uses a new format for all maps and GIS layers called the bin format to speed plotting the data. A shape2bin program that runs on linux platforms has been provided with the VERDI distribution to allow users to convert new map shapefiles to the bin format for the fast tile plot. The user can extract the shape2bin.zip and then follow the instructions in the README file to build and run the code. Once the bin files have been created, the user needs to copy them to the \$VERDI\_HOME/verdi\_1.2/plugins/bootstrap/data directory.

Follow the instructions to add a new map layer that are found in Section 11.2.3, Configure Maps, to add the new map to the Fast Tile Plot.

### **10.3 Areal Interpolation Plot**

The **areal interpolation** plot displays the interpolated value of the selected formula for each polygon in the selected area file. By comparing the colors of the polygons to those shown in the legend, users can see the relative values of the formula for each polygon area. The Areal Interpolation Plot includes several capabilities that are not available for other plot types, so these are described below, rather than in Section 11: Plot Menu Bar.

#### **10.3.1 Option Pull-down Menu Item**

The Areal Interpolation Plot Menu contains an **Options** pull-down menu to allow the user to change the map to display either the Area Averages (Figure 10-3), the Area Totals (Figure 10-4), or the value of the formula contained in the Gridded Dataset (uninterpolated) (Figure 10-5). The **Options** pull-down menu may also be used to display **All** area segments that are loaded in the area list, or to display only the area segments that are selected by highlighting the name field from the area list (Figure 10-6).

Figure 10-3 Areal Plot: Area Average

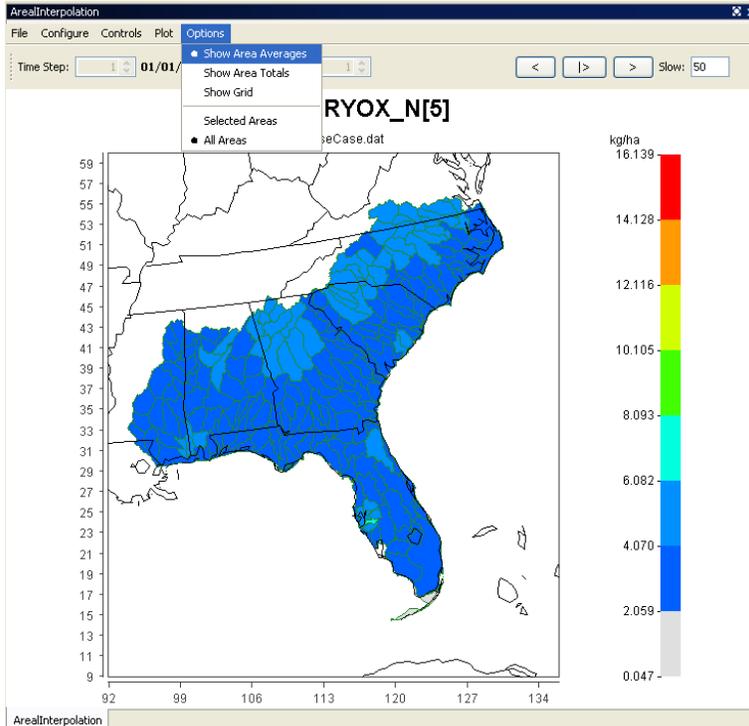
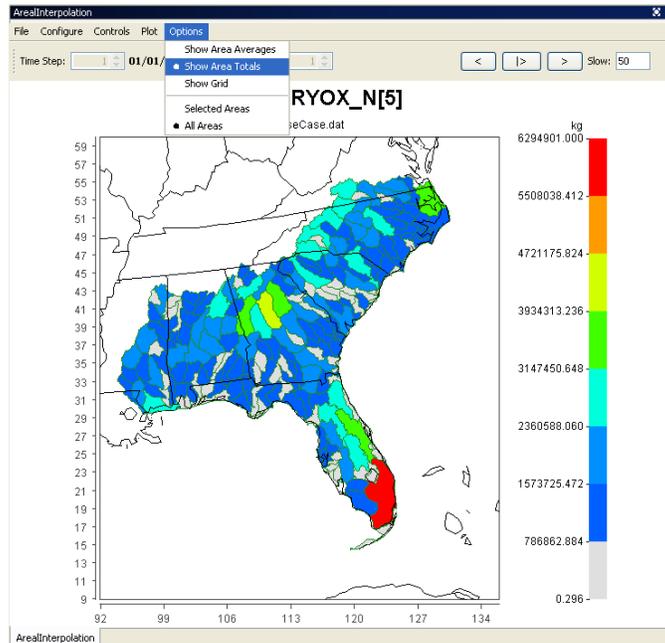
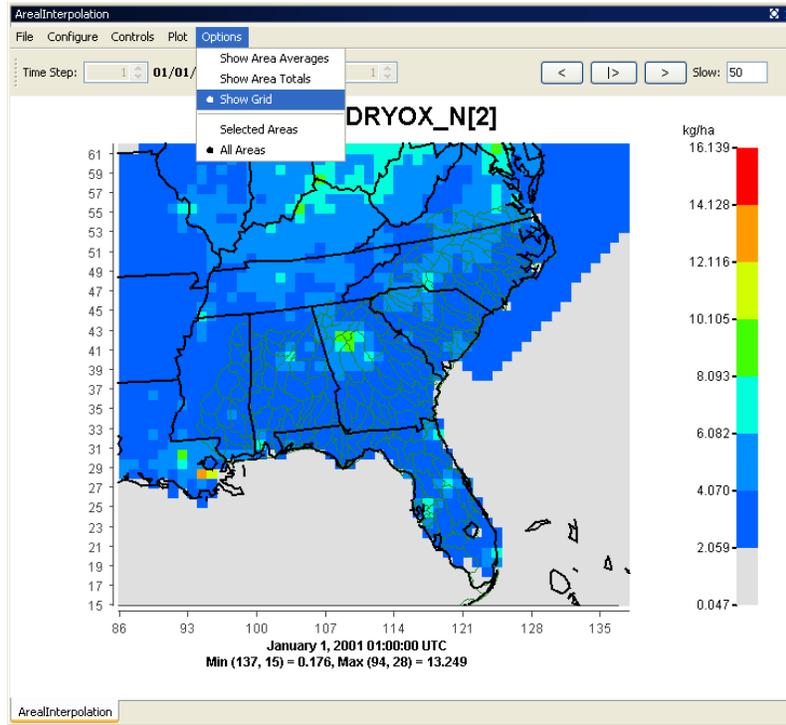


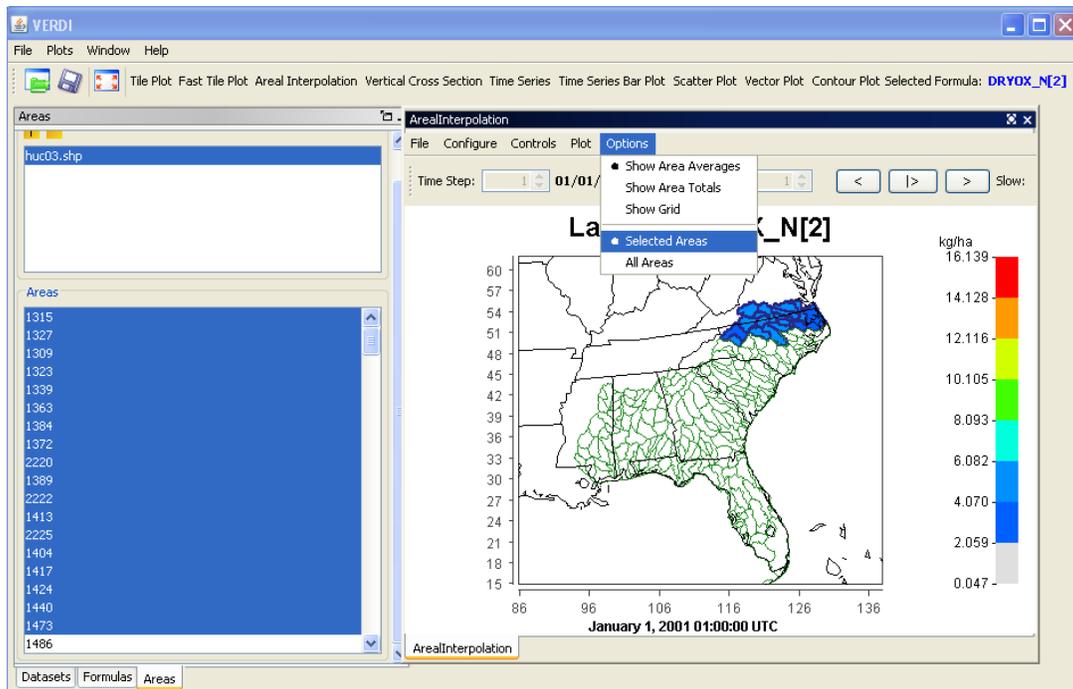
Figure 10-4 Areal Plot: Area Totals



**Figure 10-5 Areal Interpolation Plot:  
Show Grid (Gridded Data)**



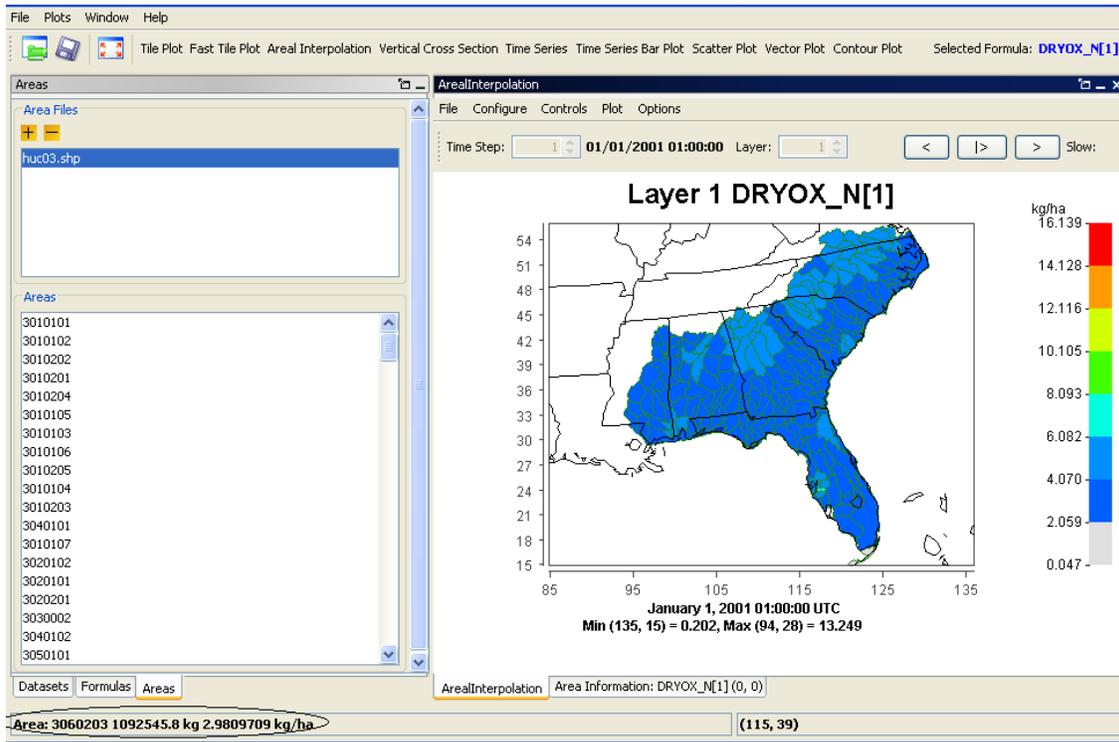
**Figure 10-6 Areal Interpolation Plot:  
Show Selected Areas**



### 10.3.2 Areal Values for Polygon Segment

To view the area, total value, and average value for a selected polygon segment, hover the mouse cursor over a polygon on the map. The values are shown at the bottom left of the information panel (Figure 10-7).

Figure 10-7 Areal Values for Polygon Segments



### 10.3.3 View and Export Areal Plot Data in Spreadsheet Format

To view the average and total interpolation values for selected formulas in a spreadsheet format, **right click** on the Areal Interpolation Plot and select **Area Information** (Figure 10-8). The Area Information Spreadsheet contains four columns: the identification number from the name field for the polygon, the total area, average interpolated value, and total interpolated value (Figure 10-9). At the top of the **Area Information** tab, the user may select **File>Export** to export the data to a spreadsheet file (Figure 10-10). The save pop-up window allows the user to specify with either a text (.txt) or comma-separated-values (\*.csv) format, also known as a comma-delimited file (Figure 10-11).

Figure 10-8. Right Click on Area Plot

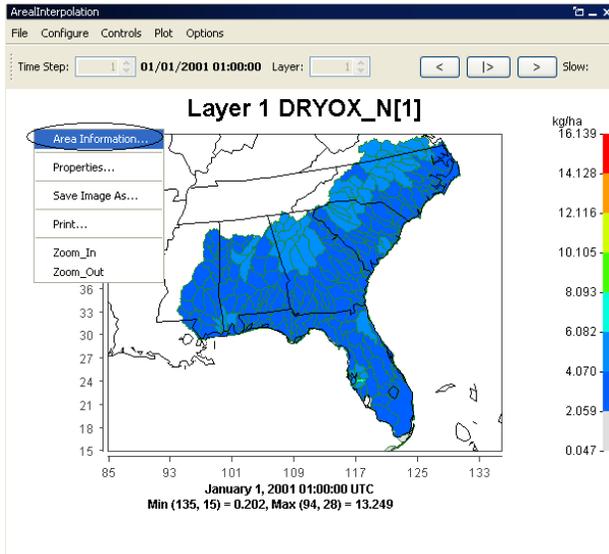


Figure 10-9. Area Information Spreadsheet

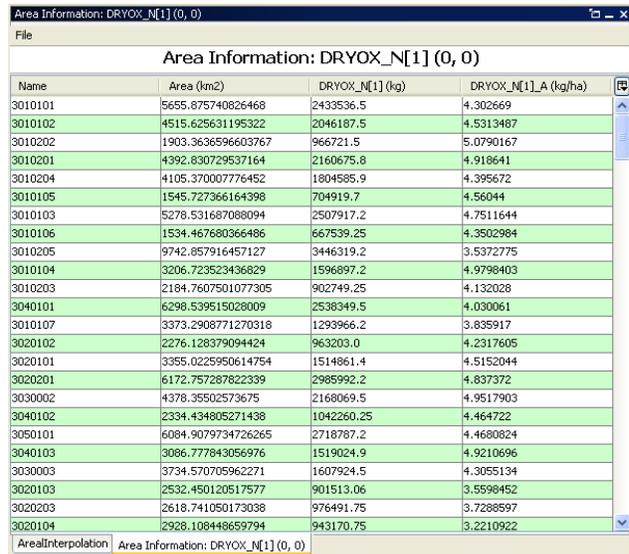


Figure 10-10. Export spreadsheet data to file (.txt or .csv)

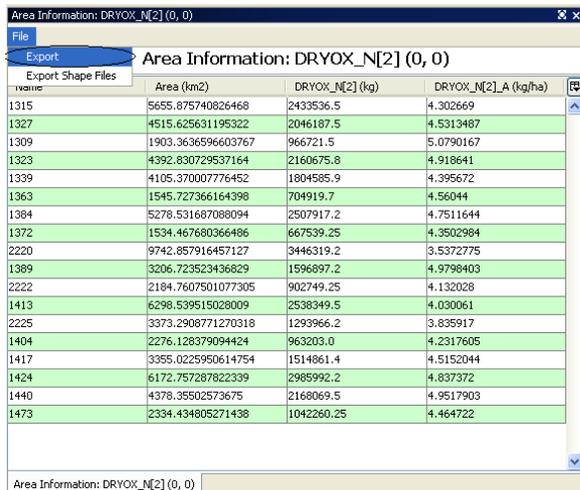
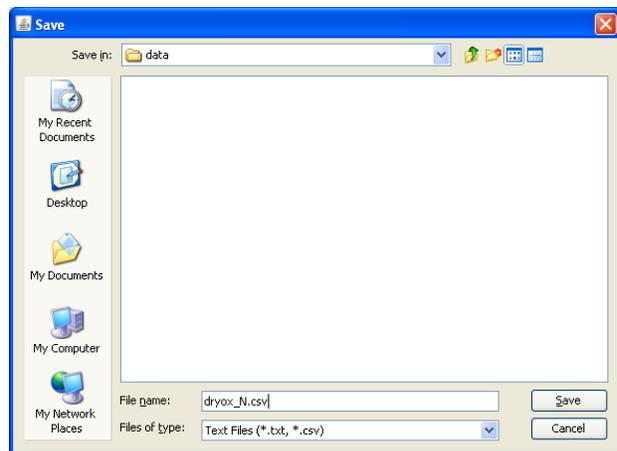


Figure 10-11. Name and save spreadsheet file



### 10.3.4 Export Areal Plot Data to Shapefiles

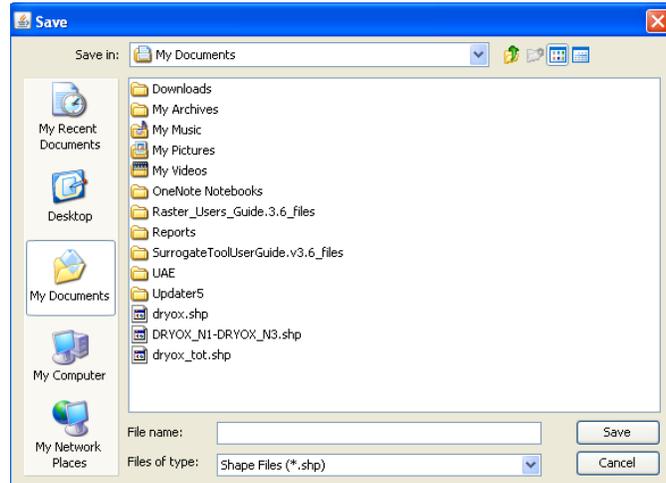
At the top of the **Area Information** tab, the user may select **File>Export Shape Files** to export the data to a shapefile. The data provided in the spreadsheet—name field, total area, average value, total value—are exported to the shapefile. A GIS program such as User-friendly Desktop Internet GIS (uDig; <http://udig.refractions.net/>), an open-source Java program, may be used to view the shapefiles generated by VERDI. The shapefiles are saved as five separate files that

must be kept together as part of the ESRI format (\*.shp, \*.dbf, \*.prj, \*.shx, and \*.fix). There are no units assigned to the data that are saved in the shapefile, so it is important for the user to keep a copy of the comma-delimited text file, or to keep some alternative text file that specifies the units for each data field.

Figure 10-12 Export Shapefile

	Area (km2)	DRYOX_N[2] (kg)	DRYOX_N[2]_A (kg/ha)
1315	5655.875740826468	2433536.5	4.302669
1327	4515.625631195322	2046187.5	4.5313487
1309	1903.3636596603767	966721.5	5.0790167
1323	4392.830729537164	2160675.8	4.918641
1339	4105.370007776452	1804585.9	4.395672
1363	1545.727366164398	704919.7	4.56044
1384	5278.531687088094	2507917.2	4.7511644
1372	1534.46780366486	667539.25	4.3502984
2220	9742.857916457127	3446319.2	3.5372775
1389	3206.723523436829	1596897.2	4.9798403
2222	2184.7607501077305	902749.25	4.132028
1413	6298.539515028009	2538349.5	4.030061
2225	3373.2908771270318	1293966.2	3.835917
1404	2276.128379094424	963203.0	4.2317605
1417	3355.0225950614754	1514861.4	4.5152044
1424	6172.757287822339	2985992.2	4.837372
1440	4378.35502573675	2168069.5	4.9517903
1473	2334.434805271438	1042260.25	4.464722

Figure 10-13 Name and save shapefile



## 10.4 Vertical Cross Section Plot

The **vertical cross section plot** allows you to show a slice of data (Figure 10-14). A popup dialog box (Figure 10-15) prompts you for information needed to create the plot. You will need to enter either the column to be used (for an *x*-axis cross section) or the row to be used (for a *y*-axis cross section) in the plot. The current time step on the plot can be changed using the **Time Step** spinner control above the plot. There is also a **Column** spinner control to change the column number (or row number). The cross-section column number (or row number) is included in the title of the plot.

Figure 10-14 Vertical Cross Section Plot

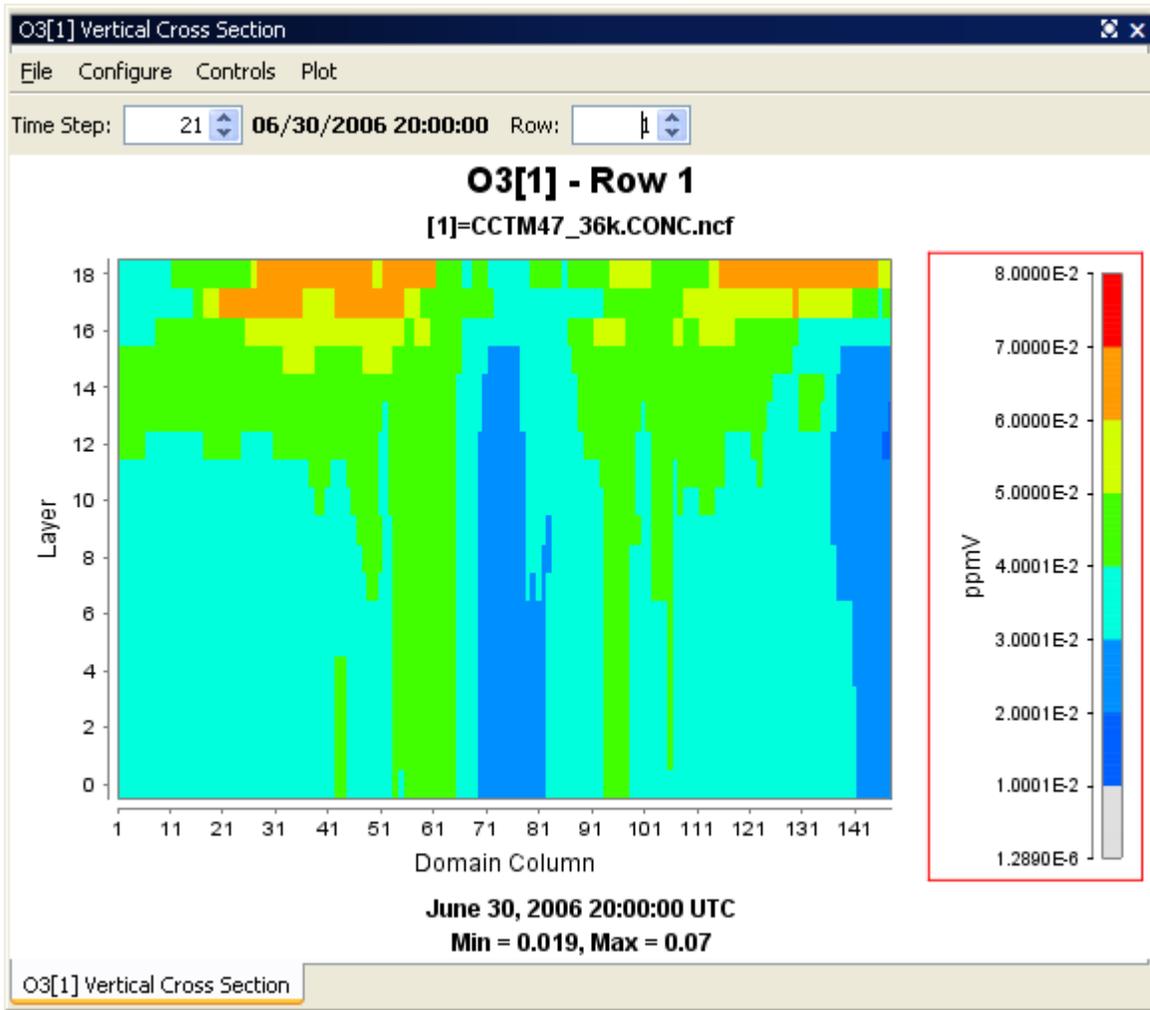
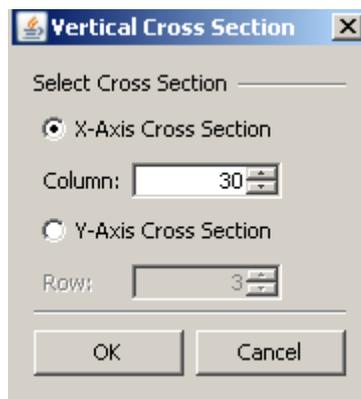


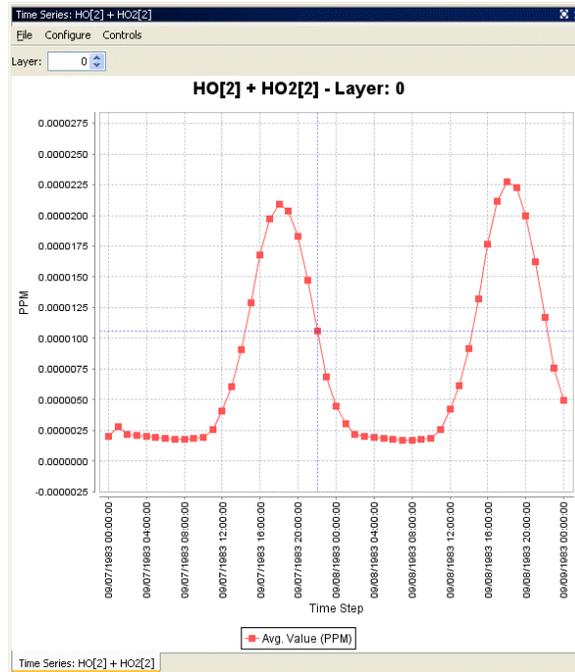
Figure 10-15 Vertical Cross Section Dialog Box



## 10.5 Time Series Plot

The **time series plot** shows a line graph with the average values over time (Figure 10-16). The plot is made for the formula's selected domain, layer range, and time-step range. Each time step's data are averaged linearly to produce that time step's data point. The current layer can be changed using the **Layer** spinner control above the plot. The layer value listed in the title is updated when you change the layer.

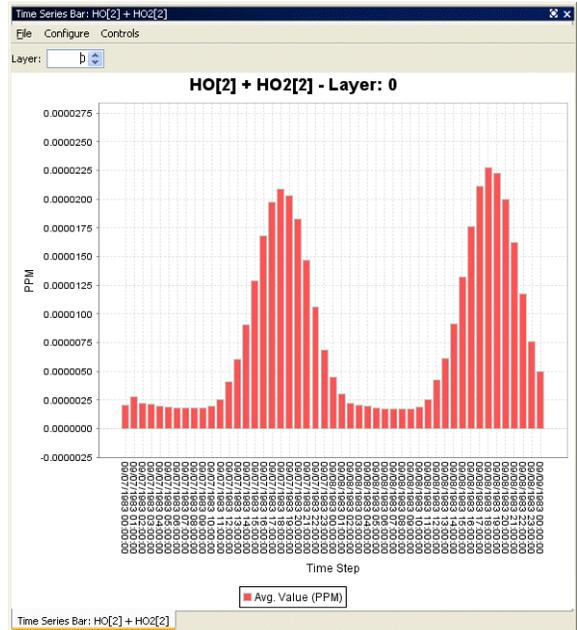
Figure 10-16 Time Series Plot



## 10.6 Time Series Bar Plot

The **time series bar plot** shows average values over time in a bar plot format (Figure 10-17) rather than a line format (Figure 10-16). Other than that, the description of this plot type is the same as for the time series line plot (see Section 10.5).

**Figure 10-17 Time Series Bar Plot**



## 10.7 Scatter Plot

The **scatter plot** shows the relationship between two formulas using an array of dots (Figure 10-18). The user specifies the formulas using the dialog box that comes up before the plot is displayed (Figure 10-19). The current time step and layer can be adjusted using the spinner controls above the plot. The data from a scatter plot may be exported by selecting the **File** menu option and then selecting Export data. A pop-up window (see Figure 10-18) allows users to specify whether they would like the data for the current layer, or for all layers, and for the current time step, or for all time steps. Specify the time and layer ranges, and then click the **OK** button. A Save pop-up dialog box will appear. Fill in a file name of your choice with a .csv extension, and specify the directory in which you would like to save this file. The CSV file will be comma delimited, and will contain the following columns of data: layer, time, x-axis formula, y-axis formula.

Figure 10-18 Scatter Plot

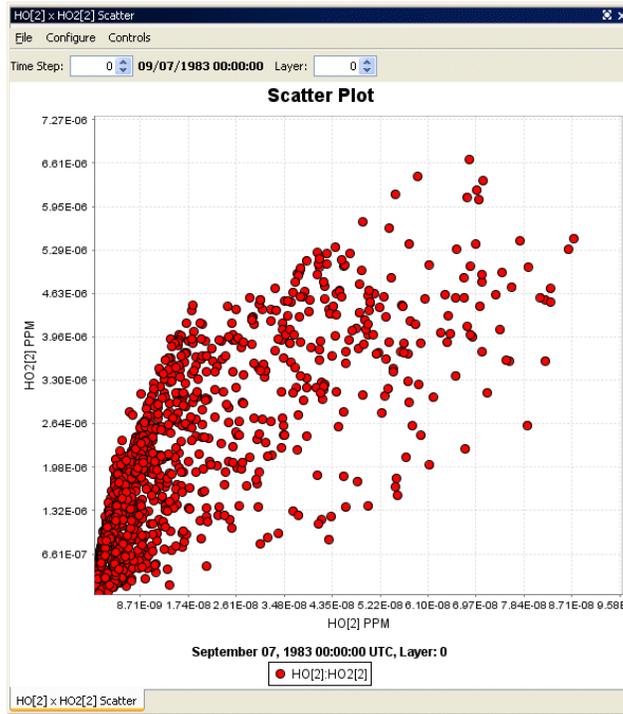
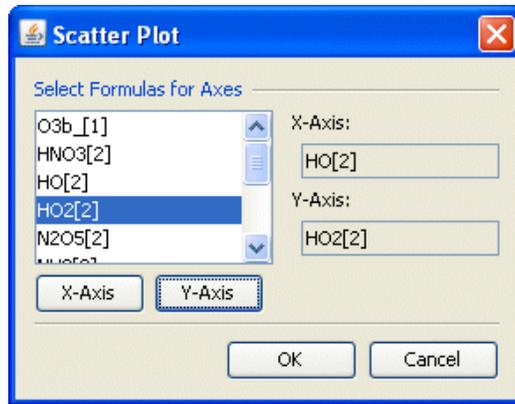
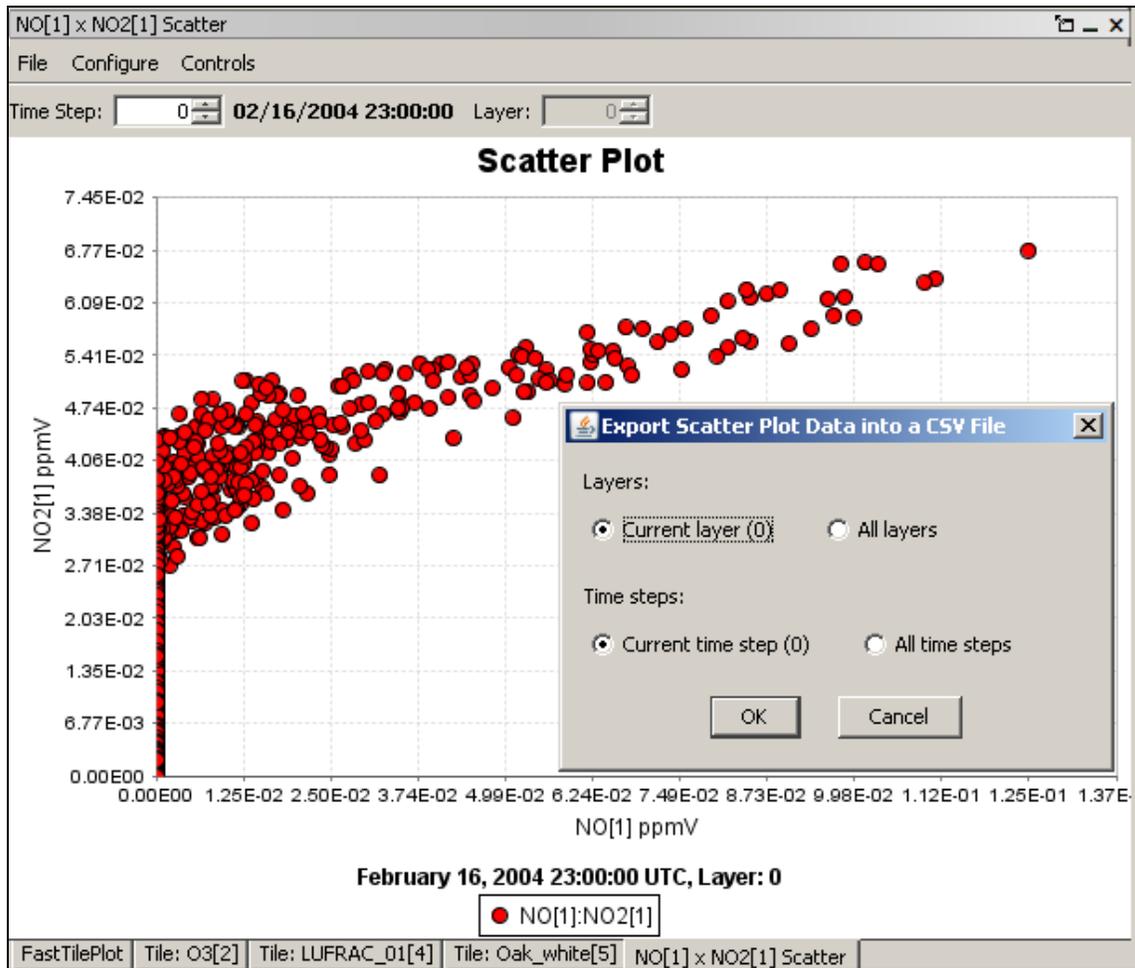


Figure 10-19 Scatter Plot Dialog Box



**Figure 10-20 Scatter Plot Export Data into a CSV file**



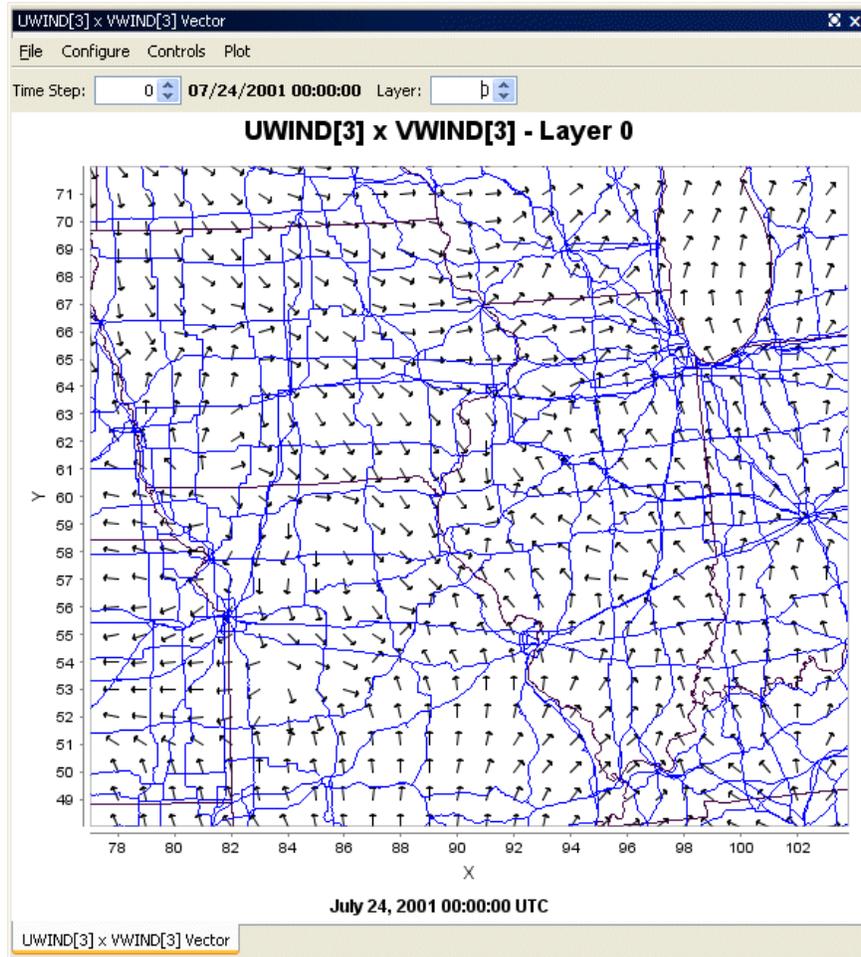
## 10.8 Vector Plot

The **vector plot** shows vector lines representing the vector formed by using two formulas (one for each component of the vector) across a geographic area; one formula represents the horizontal component, the other represents the vertical component (Figure 10-21). The formulas selected as components of the vector must match in domain, layer range, and time-step range. You specify the components of the vector using the dialog box that comes up before the plot is displayed (Figure 10-22). After the plot is displayed, the current time step and layer can be adjusted using the controls above the plot.

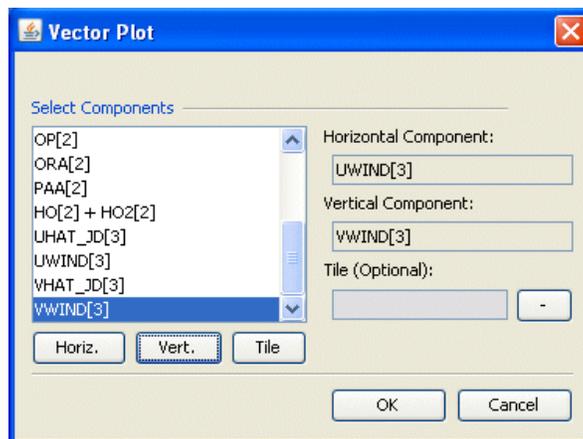
It is also possible to overlay vectors (e.g., wind vectors) on a tile plot. This can be done either from within the vector plot option (discussed here) or from within the tile plot option (discussed in Section 10.4.4). To do this overlay from within the vector plot option, you select a formula for the Tile option in the vector plot dialog box (Figure 10-22) in addition to specifying the formulas

to be used for the horizontal and vertical vector components. See Section 11.4.4 for more information on vector overlays.

**Figure 10-21. Vector Plot**



**Figure 10-22. Vector Plot Dialog Box**



## 10.9 Contour Plot

The **contour plot** shows a 3-D representation of values over a geographic area (Figure 10-23). The current time step and layer can be adjusted using controls above the plot. You can also animate the plot over time using an option in the **Plot** pull-down menu (Figure 10-24). In addition, the contour plots can be rotated in three dimensions to achieve different viewing angles by using the left mouse button to grab and rotate the plot.

Figure 10-23. Contour Plot

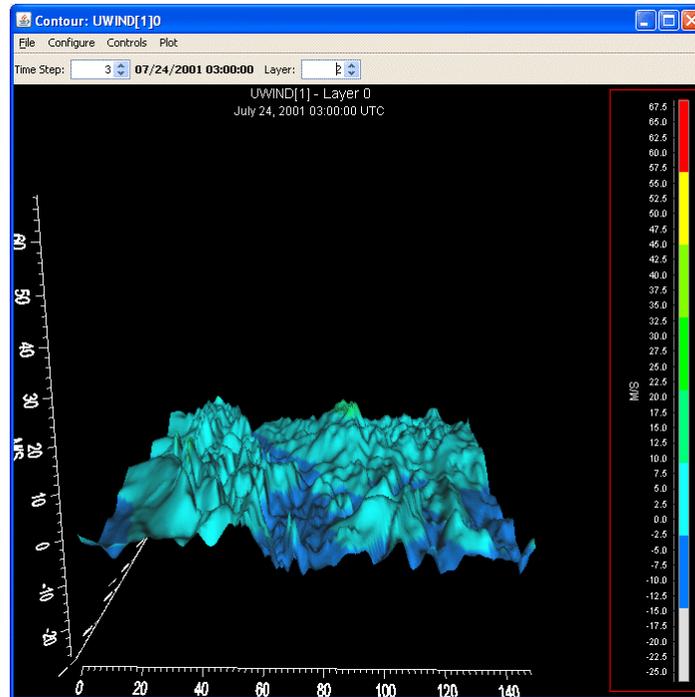


Figure 10-24. Contour Plot Menu Options

File	Configure	Controls	Plot
Print	Configure Plot	Show Lat/Lon	Animate Plot
Export As Image	Load Configuration	Show IsoContours	
	Save Configuration		

## 11 Plot Menu Bar

Each VERDI plot contains its own menu bar that has options specific to that type of plot. As an example, the menu options at the top of a tile plot include those shown in Figure 11-1. Most options are common to all plots, and function in the same way (unless the option is grayed out) for all plot types, so it is easier to describe the function of these configuration menu options

once, rather than repeat the same information for each type of plot. Only options that are applicable to a particular plot type are enabled; others are grayed out and cannot be selected.

Plot configuration options that are available in VERDI are discussed in this section.

**Figure 11-1 Tile Plot Pull-down Menu Options**

File	Configure	Controls	Plot
Print Export as Image	Configure Plot Load Configuration Save Configuration	Zoom Probe Set Data Ranges	Time Series of Probed Cell(s) Time Series Bar of Probed Cell(s) Time Series of Min. Cell(s) Time Series of Max. Cell(s)
	Maps	Show Lat/Lon	
			Animate Plot
			Add Overlay

### 11.1 File Menu Options

Options in the **File** menu include printing a plot and exporting a plot as an image. Plots can be saved as a PNG, JPG, BMP, or TIF files.

### 11.2 Configure Menu Option

The **Configure** pull-down menu contains the following options: Configure Plot, Load Configuration, Save Configuration, and Maps.

#### 11.2.1 Configuring plots

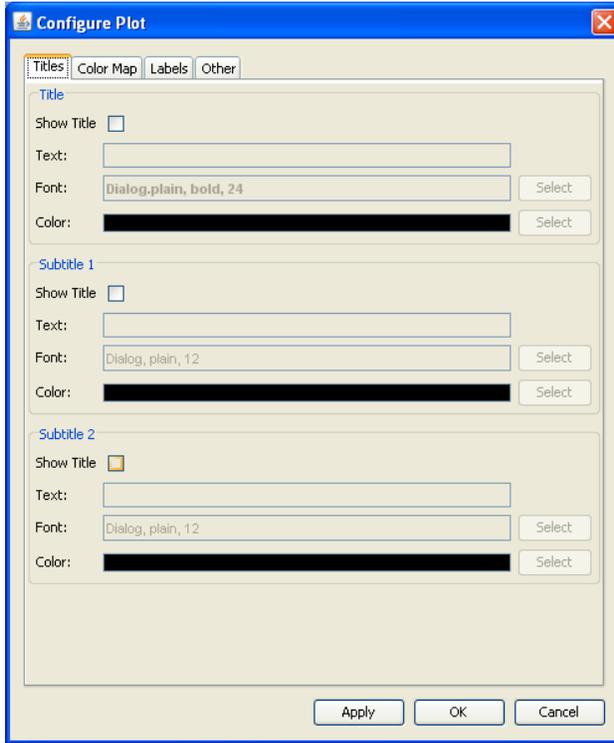
Figures 11-2 through 11-5 show the dialog boxes that appear when you select Configure Plot. The **Configure Plot** dialog box contains four tabbed options: **Titles**, **Color Map**, **Labels**, and **Other**.

- The **Titles** tab allows you to edit and set the font and color for the title and two subtitles of the plot.
- The **Color Map** tab allows you to select the number of tiles, the palette type to be used, the color interval, and the number format to be used. As you vary the number of tiles, the palette types that are available to choose from are automatically updated. The palette type options include sequential (similar to what users typically use for PAVE plots), qualitative, and diverging. To change the spacing between the color intervals, select Custom from the **Interval** pull-down menu and change the interval start values in the list at the bottom of the dialog box. This allows you to set the interval values for each color in the color map. To specify the number format used in the map legend, enter the format

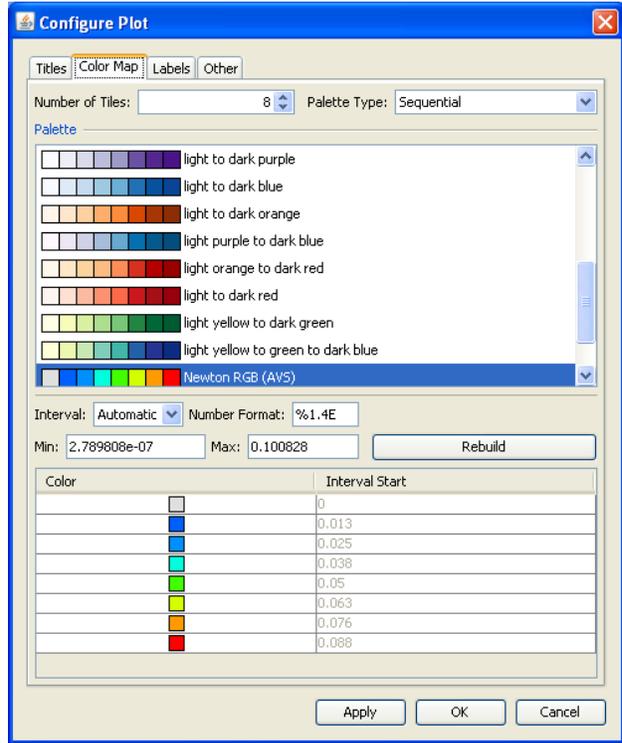
in the entry box using the C language's printf() routine's format syntax, e.g., for **%4.2E**, the numbers on the legend will look like: 9.49E-2. To reset the range for the values of the legend, change the values for the minimum and maximum and then left mouse click on **Rebuild**. Click **Apply** to review changes made to the map while the **Configure Plot** dialog is open. Use the **Apply** button as needed to make additional adjustments to the color map, legend range, or number of intervals in the range. Once the settings are finalized, click **OK** and the Configure Plot dialog will close.

- The **Labels** tab contains options to edit the labels of the Domain Axis (x- axis), the Range Axis (y-axis axis), and the Legend.
- The **Other** tab allows you to enable or disable showing the grid lines, and to edit the vector arrow color and the series color. The edit series color allows the user to specify the colors used to shade the bars for the time series bar plot, or the colors of the lines on the Time Series Plot.

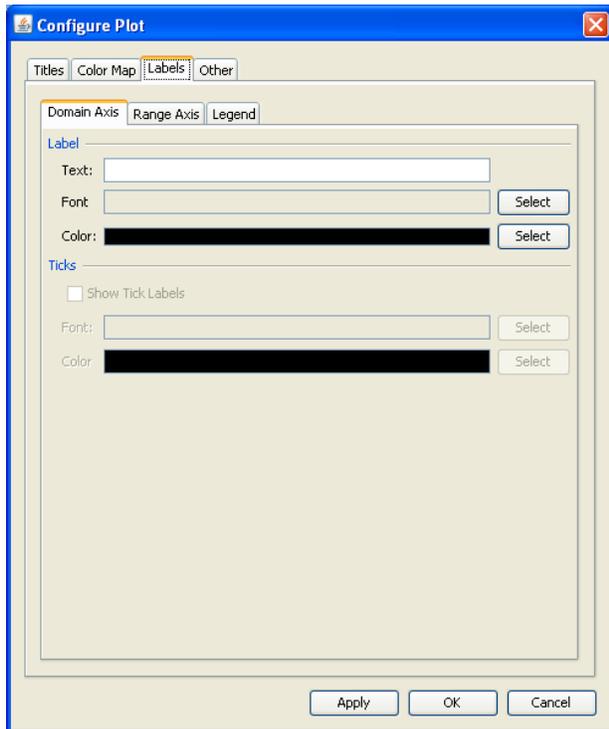
**Figure 11-2. Configure Plot, Titles Tab**



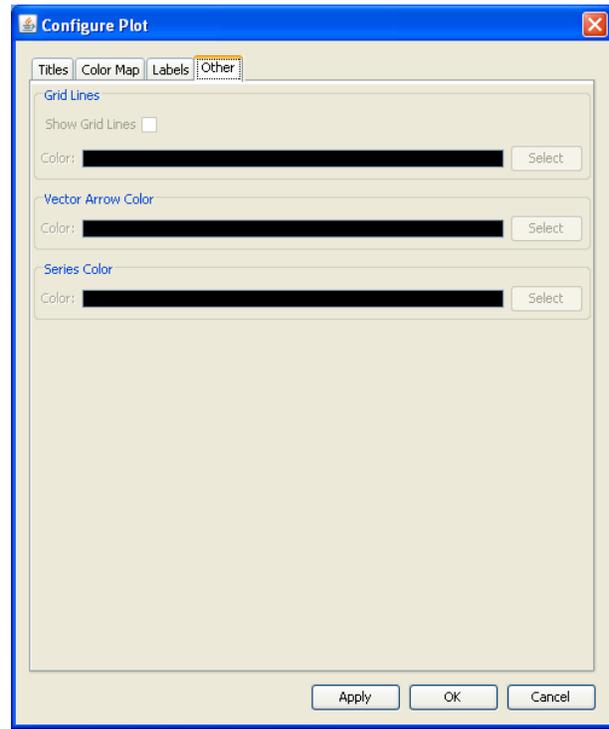
**Figure 11-3. Configure Plot, Color Map Tab**



**Figure 11-4. Configure Plot, Labels Tab**



**Figure 11-5. Configure Plot, Other Tab**



### 11.2.2 Loading and saving configuration

If you have made changes to the configuration of a plot, and would like to save the configuration for future use on other plots, use the **Save Configuration** option from the **Configure** pull-down menu on the plot. It is important to create a file name that indicates the formula name, the dataset, and the type of plot, along with the .cfg extension that indicates that it is a configuration file, so that you will know which plot the configuration file was saved for. An example file name is <FormulaName>\_<DatasetFilename>\_<PlotType>.cfg, or “O3\_CCTM\_base\_tile.cfg”. This is only an example name; you may use your own naming convention. What is important is not how you name the files, or how you organize the files into directories, but that you remember the datasets and variables to which a specific configuration may be applied.

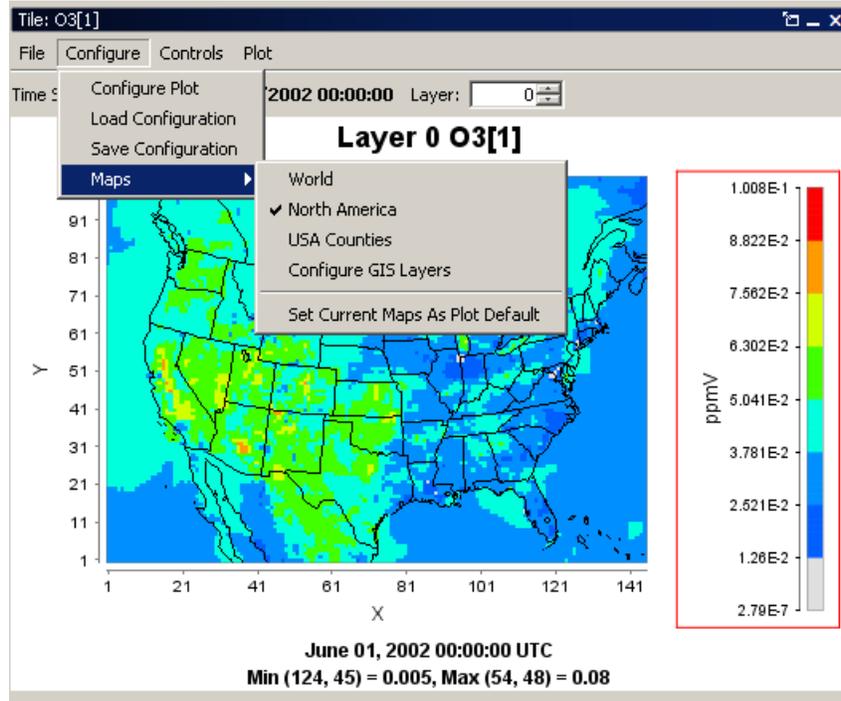
To load a previously saved plot configuration, first create a plot that is of the same type, and uses the same formula that is used within the configuration file. Then select the **Load Configuration** option from the **Configure** pull-down menu on the plot. An **Open File Dialog** window will allow you to look for the directory in which you saved the configuration file, and to open that file. The plot title, color map, and other plot configuration features will be applied to the plot. Note that it is possible to load a saved configuration file that does not apply to the selected plot, so before loading a saved plot configuration check carefully to be sure the plot type and formula of the configuration file match those of the plot.

### 11.2.3 Configure Maps

The **Maps** option in the **Configure** pull-down menu is used to add maps to a tile plot (Figure 11-6). The tile plot and fast tile plots are the only plots that have maps.

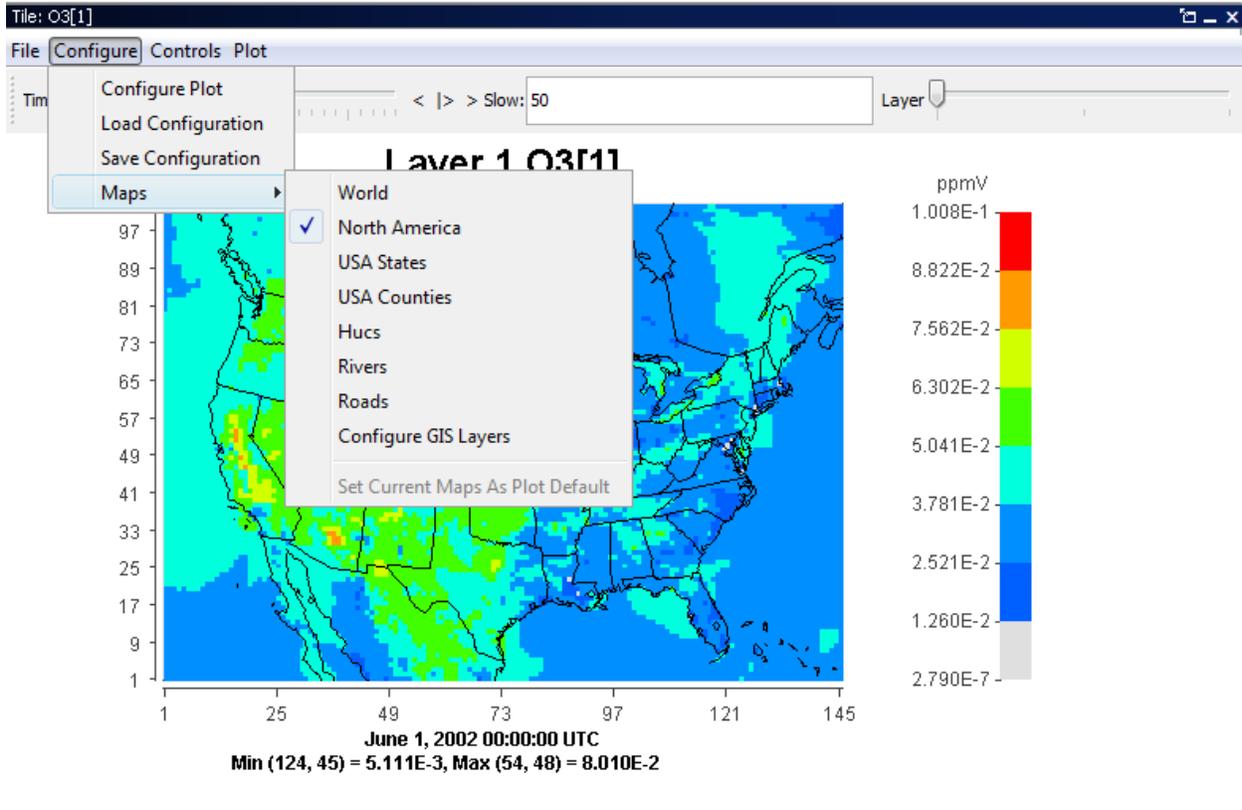
For the Tile Plot, a world map, a North America map, and the counties of the United States may be selected or de-selected by clicking with the left mouse button on one of these map names. This will cause a check button to appear or disappear next to the chosen map name, and the selected map to appear on the plot.

**Figure 11-6 Configure Maps for a Tile Plot**



The fast tile plot has different maps that can be selected (Figure 11-7), as was described in Section 10.2. Also, the fast tile plot automatically loads the counties map as you zoom in to a region of the United States.

**Figure 11-7 Fast Tile Plot Map Options**

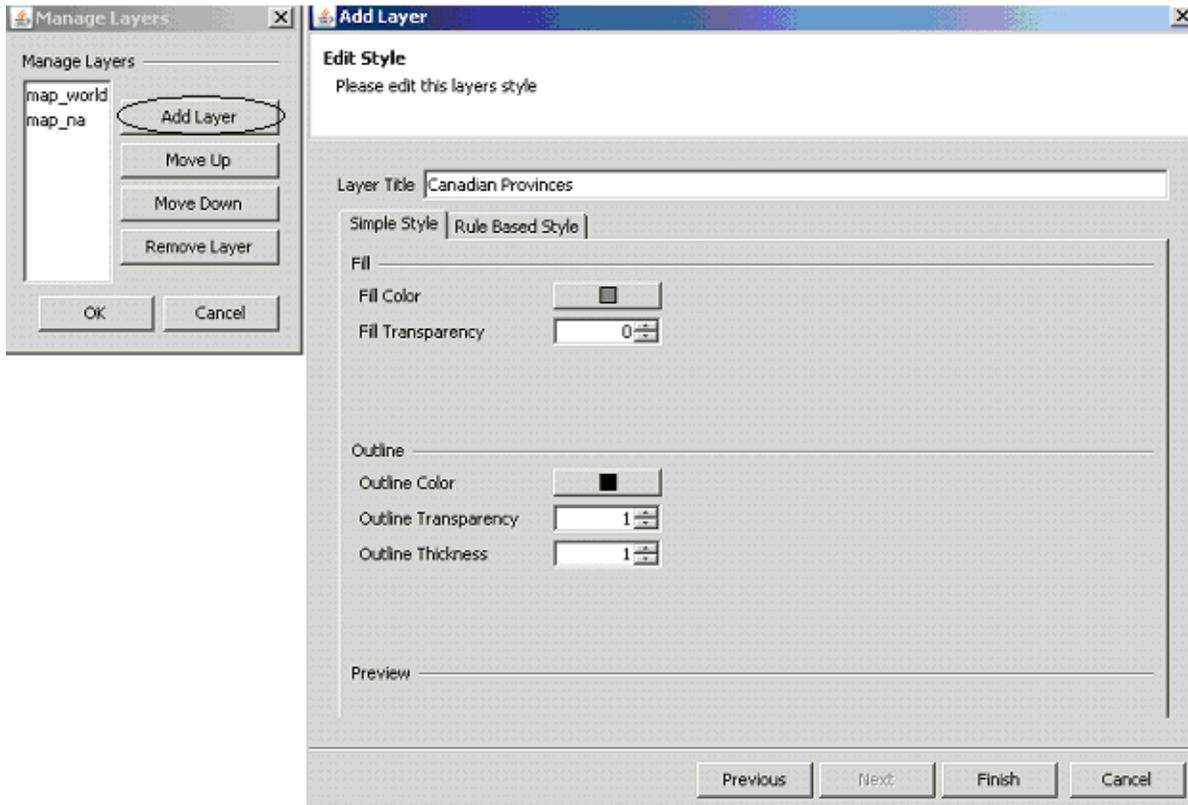


To show an additional map on the plot, select the **Configure GIS Layers** option. When you click on this item, a dialog box titled Manage Layers gives you the following options: Add Layer, Move Up, Move Down, and Remove Layer (Figure 11-8).

- When you select Add Layer, a window prompts you to select a shapefile (for a tile plot) or a shapefile that has been converted into a binary format (for a fast tile plot). VERDI looks for files with the .shp or .bin extension. After selecting a shapefile or a .bin file you are asked to edit the layer style for the shape (Figure 11-8). Edit the layer title to specify what type of shapefile you are adding. If the shapefile contains a fill, and you intend to have the map layer to be on top of other map layers, you will need to change the fill to be transparent, so that you can see the other map layers. Change the fill transparency value from 1 (1=opaque) to 0 (0=transparent) and select the Finish button. After the layer you have added is listed under Manage Layers, click OK to have the layer added to the plot. If you add multiple layers to the list in this dialog box, the one at the top of the list is the top layer shown on the plot.
- To rearrange the order in which the GIS layers are displayed on the plot, select a layer in the Manage Layers dialog box, and then select Move Up or Move Down to reposition the order of that layer within the list. If the layers that you are selecting are boundaries and were created to have a transparent fill, then rearranging the order of the layers will not change the look of the boundaries on the plot.

- To remove a GIS layer from the plot, select that layer in the list, then select Remove Layer, then click OK to remove it.

**Figure 11-8 Manage Layers, Add Layer, Edit Style**



### 11.3 Controls Menu Options

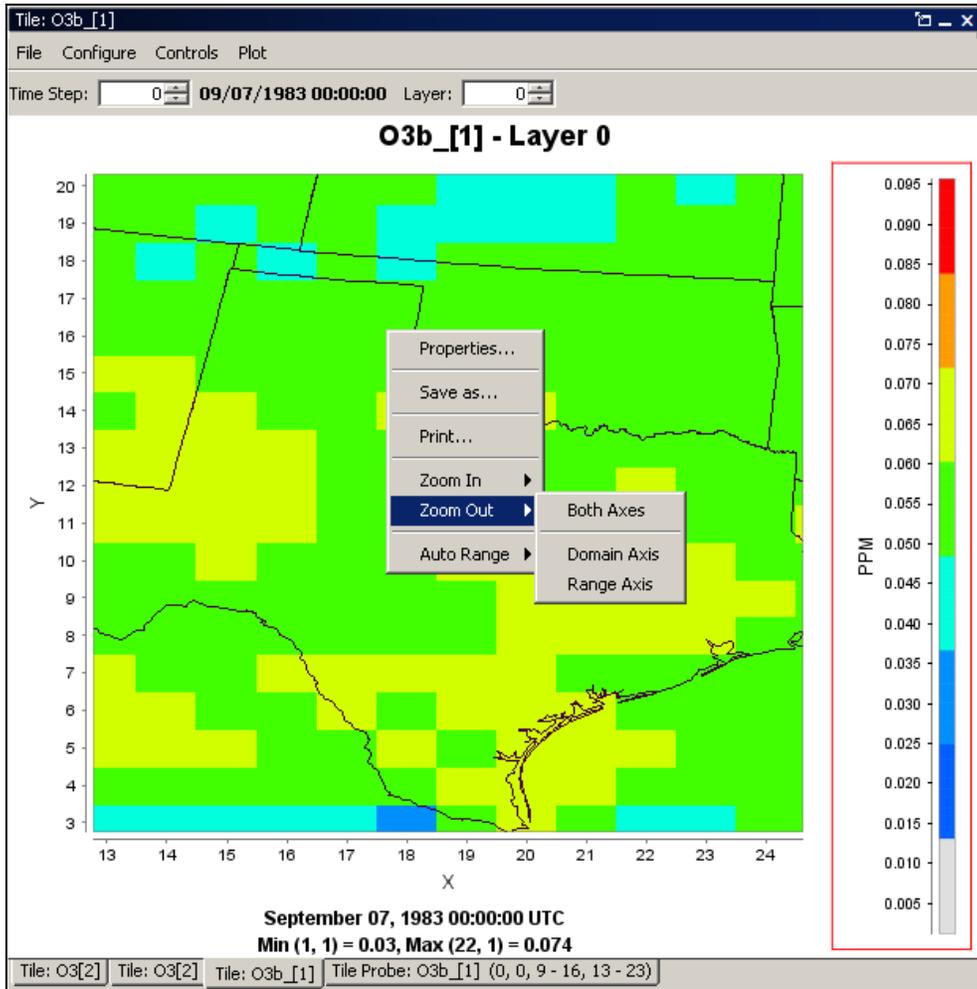
The **Controls** pull-down menu contains the following options: Zoom, Probe, Set Data Ranges, and Show Lat/Lon.

#### 11.3.1 Zooming in and out

To zoom in and enlarge a subdomain of the data, select the Zoom option. Then use the left mouse button to click on a region of the map and hold down the button to draw a rectangle around the region of interest.

To zoom out use your mouse to right click on the map. A pop-up window will appear, allowing you to select Zoom Out>Both Axes or Zoom In>Both Axes, or to select Auto Range>Both Axes to recover the plot of the full domain (Figure 11-9). The pop-up window also allows you to select the option to zoom only along the domain axis, or only along the range axis.

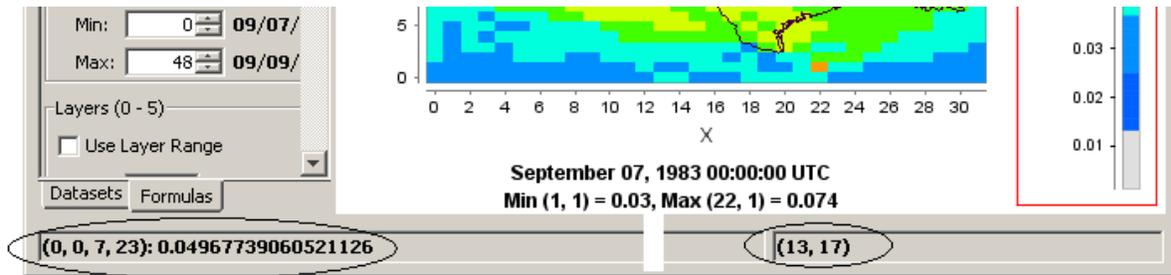
Figure 11-9 Right Click on Plot to Zoom Out



### 11.3.2 Probing values at specific points

To determine the data value at a specific point, select the Probe option under the **Controls** pull-down menu. To probe a single data point, use the mouse to hover the cursor over that grid point on the map; the coordinates of the grid points are shown in the lower right-hand side of the plot in the format (column, row). Once you click on the grid point of interest, the value of the datum at that grid point is displayed in the lower left-hand area of VERDI main window (Figure 11-10).

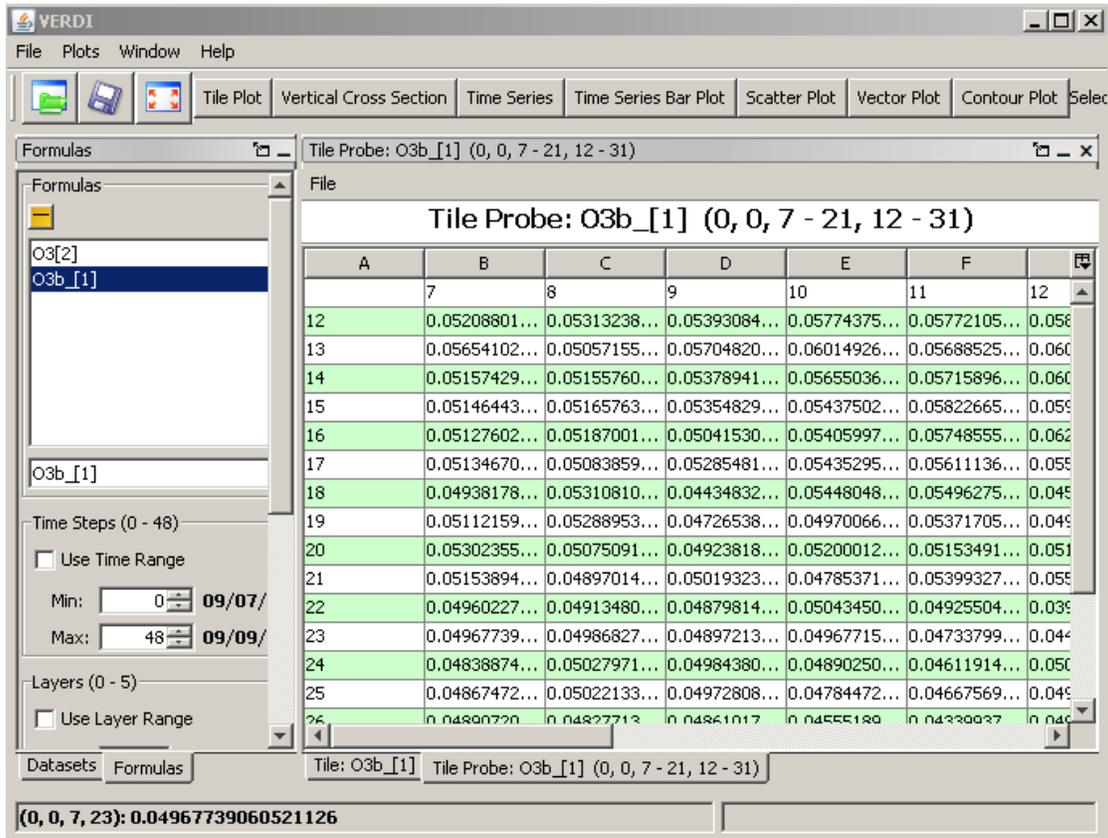
**Figure 11-10 Click on Plot to Probe: Data Value Shown in Lower Left of VERDI, Grid Values Shown in Lower Right**



### 11.3.3 Probing a domain region of data

To probe the values of a selected region of grid points, select the Probe option under the **Controls** pull-down menu, then draw a rectangle by clicking on a point on the map, holding down the left mouse button, and then releasing the mouse button once the rectangle encloses the region of interest. VERDI will create a spreadsheet displaying the grid values and will place it in the plot area of the VERDI main window as a tabbed window (Figure 11-12). The File>Export menu option at the top of the spreadsheet allows you to save probed data as a comma-delimited text file (\*.csv).

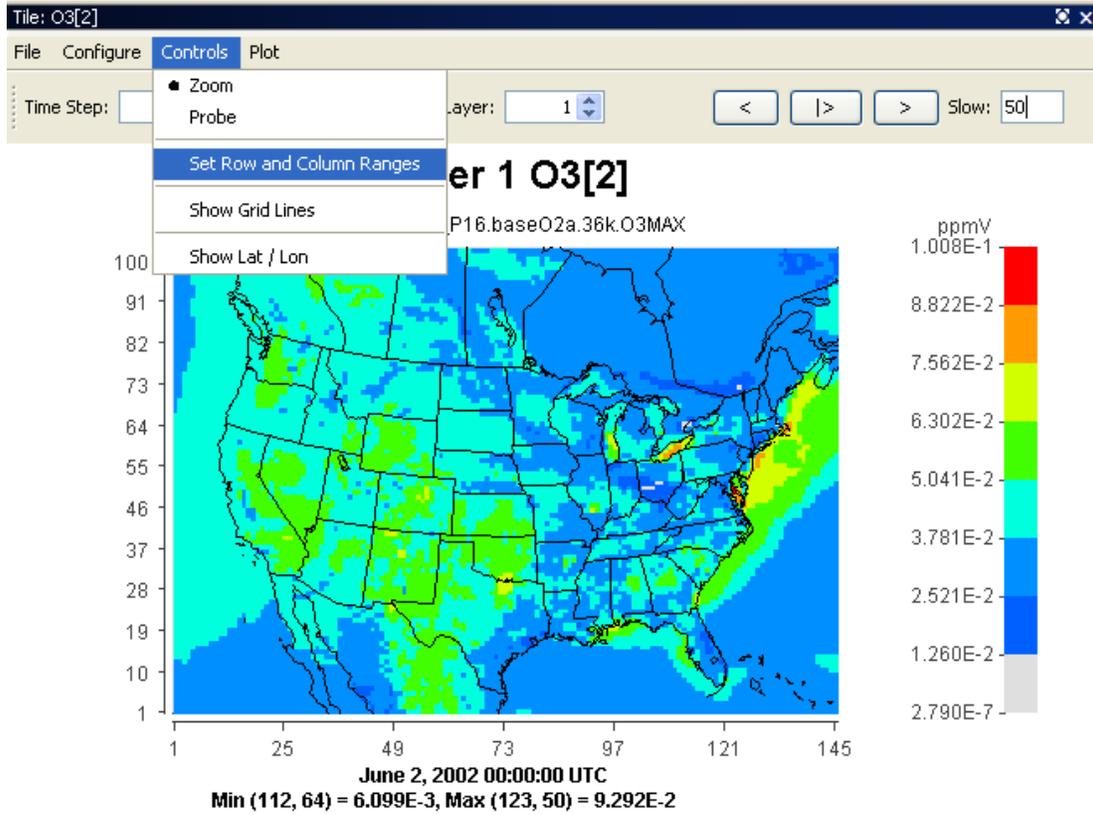
**Figure 11-11 Spreadsheet Showing Probed Values for Region of Interest**



### 11.3.4 Set Data Ranges

The **Controls>Set Data Ranges** menu item will bring up a pop-up window that allows the user to configure the minimum and maximum values used in the *x*-axis range and *y*-axis range (Figure 11-11).

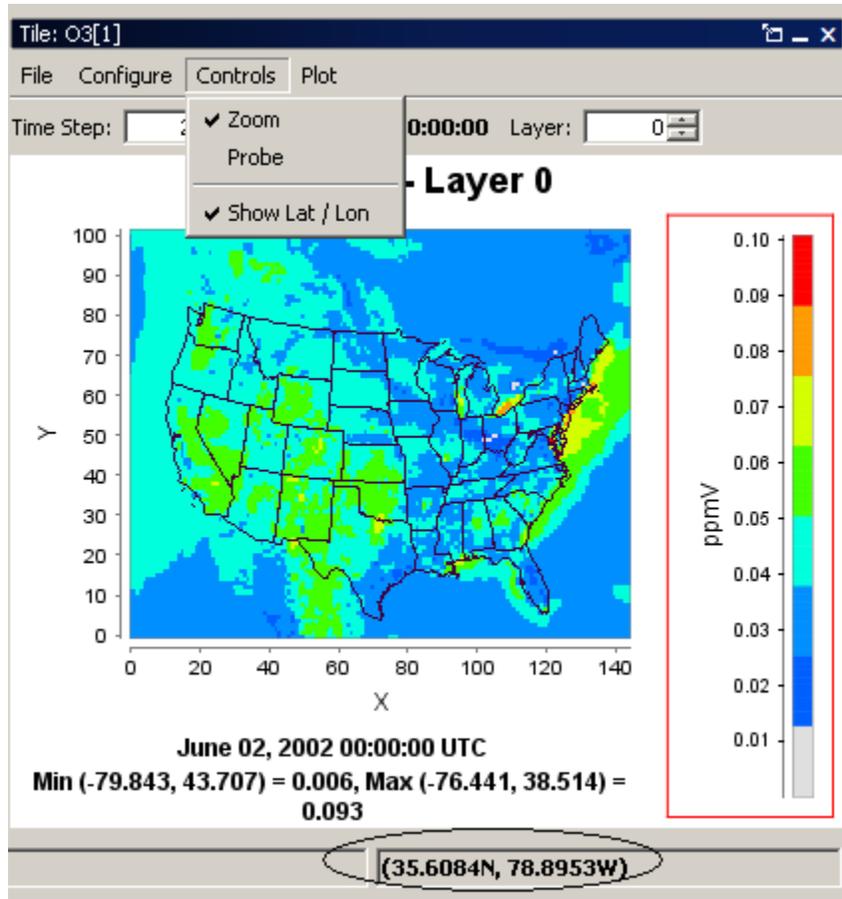
Figure 11-12 Configure Min and Max Range Values



### 11.3.5 Showing latitude and longitude

To view the latitude and longitude values for a point on the plot, select the Show Lat/Lon option on the **Controls** menu, then hover your cursor over the location for which you would like to know those values. The lat/lon coordinates will be displayed in the lower right-hand side of the window (Figure 11-13). The option to display the lat/lon coordinates may be selected, and works with either the Zoom or the Probe option.

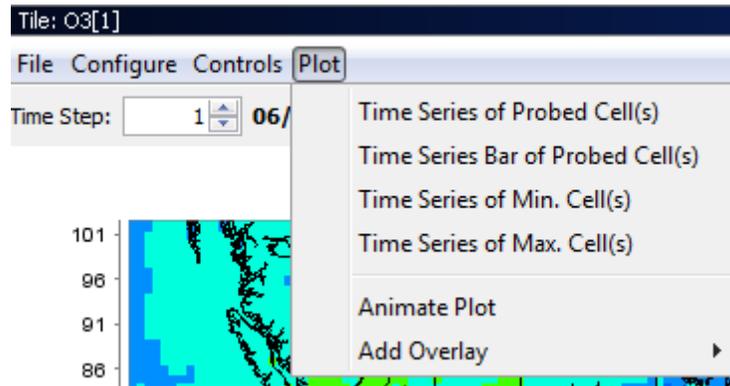
**Figure 11-13 Lat/Lon Values Shown in Lower Right of VERDI**



### 11.4 Plot Menu Options

The **Plot** pull-down menu contains the following options: Time Series of Probed Cell(s), Time Series Bar of Probed Cell(s), Time Series of Min. Cell(s), Time Series of Max Cell(s), Animate Plot, and Add Overlay.

**Figure 11-14 Plot Menu Options**



### 11.4.1 Time series plots

The Time Series of Probed Cell(s) and Time Series Bar of Probed Cell(s) allows the user to select a set of cells, and then produce a time series or time series bar plot of the chosen subset of probed cells. The Time Series of Min.[or Max.] Cell(s) option creates a time series plot using data for the currently selected formula at that formula’s domain, layer range, and time step range. The minimum [or maximum] value of that formula over the domain and layer range at that time step is calculated by VERDI and used for each of the time step’s data points.

### 11.4.2 Animate plots

You can create an animated plot by selecting the Animate Plot option. The Time Series and Time Series Bar Plots do not have an animate plot option. The plots that may be animated include: Tile, Fast Tile, Vertical Cross Section, Vector Plot, and Contour Plot. An Animate Plot dialog box (Figure 11-15) will appear, allowing you to save animations either as an animated GIF with a file extension of .gif or as a Quicktime movie with a file extension of .mov. This **Plot** menu option is plot-specific and so does not allow you to animate more than one plot at a time. To animate multiple plots, you will need to use the **Plots** pull-down menu at the top of the VERDI main window; see Section 5.2.2, “Animate Tile Plots.”

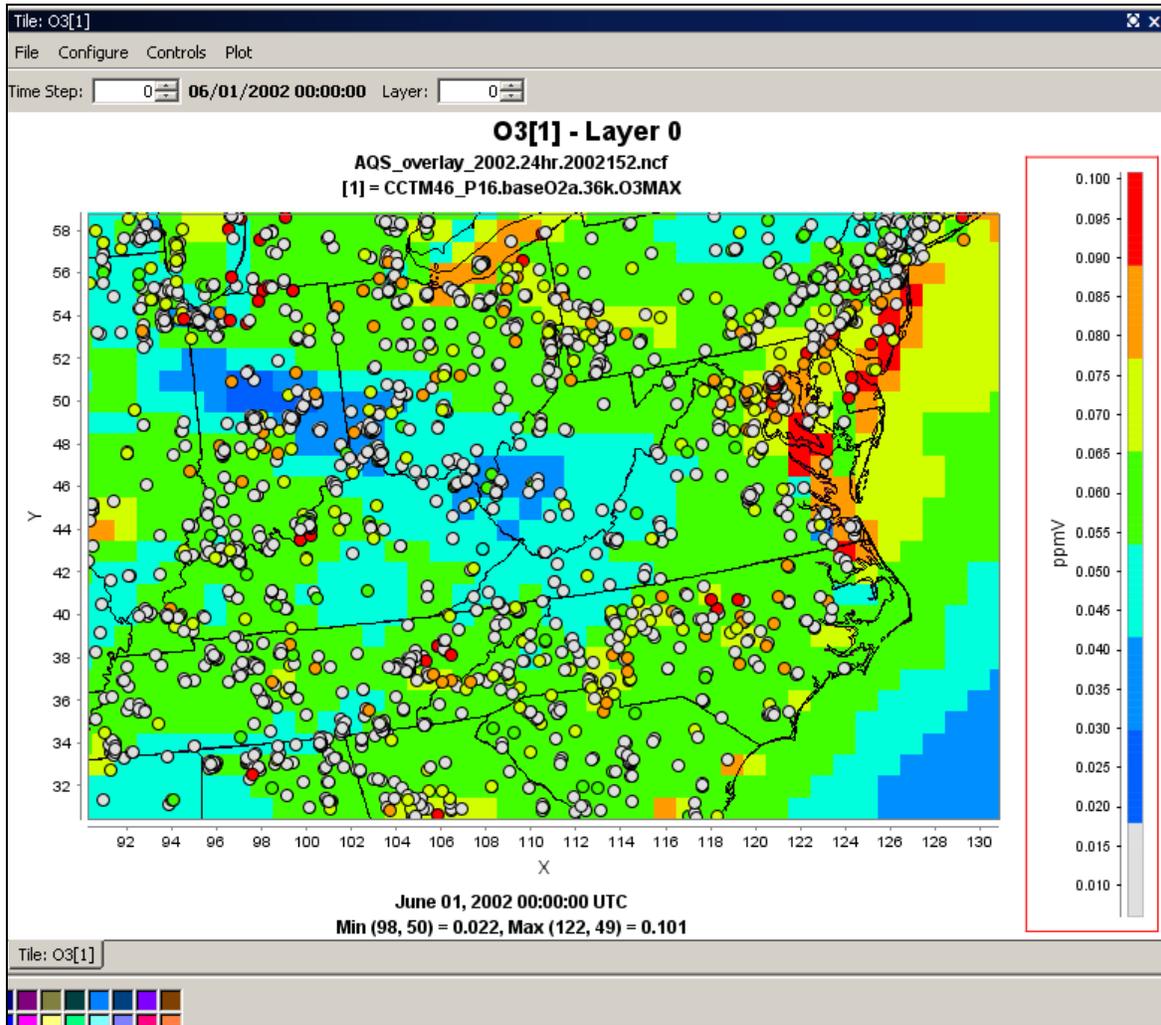
**Figure 11-15 Animate Plot Dialog Box**

### 11.4.3 Observational data overlays

It is useful to visually compare the results contained in model output datasets with the data points in observational datasets. You can do this by creating a tile plot of the model output and then overlaying observational data points on the tile plot. (This capability will be added to the Fast Tile Plot in a future release. This capability is not available in the version 1.2 release. The observational dataset needs to be in an I/O API observational data format (see Chapter 13 for more information about how to convert AIRS observational data into this format). To begin, load both a model output dataset and an observational dataset using the yellow **plus** button in the **Dataset** pane. Note that when the observational dataset is loaded in the **Dataset** pane, an (obs) label appears to the right of the dataset name. Next, create a formula in the **Formula** pane using a variable from the *model output* dataset. Use this formula to create a tile plot. (Note: If you create a formula referencing a variable from an *observational* dataset, this formula will appear in the formula list but it will be invalid and cannot be used to create a plot.)

To view observational data as an overlay on a tile plot, select Add Overlay>Observations from the tile plot's **Plot** pull-down menu. An **Observation** dialog box will appear containing the variables that are available in the observational dataset. Select the observational variable that you would like to overlay on the tile plot from the Observation Details list. You then have the option to control the appearance of the filled-in circles that represent the observational data. The stroke size controls the thickness of the line used to draw the circles, while the shape size controls their diameter. After you select the variable, and optionally set the stroke size and shape size, select **OK** to overlay the observational data on the tile plot (Figure 11-16). At this time, VERDI does not have an option to remove or to reset the size of the symbols on an observational data overlay on a tile plot. If the user wishes to resize or change the symbol of the observational data overlay, they will need to draw a new tile plot and re-add the observational data overlay using the new stroke size or shape size.

Figure 11-16 Tile Plot with Observational Data Overlay



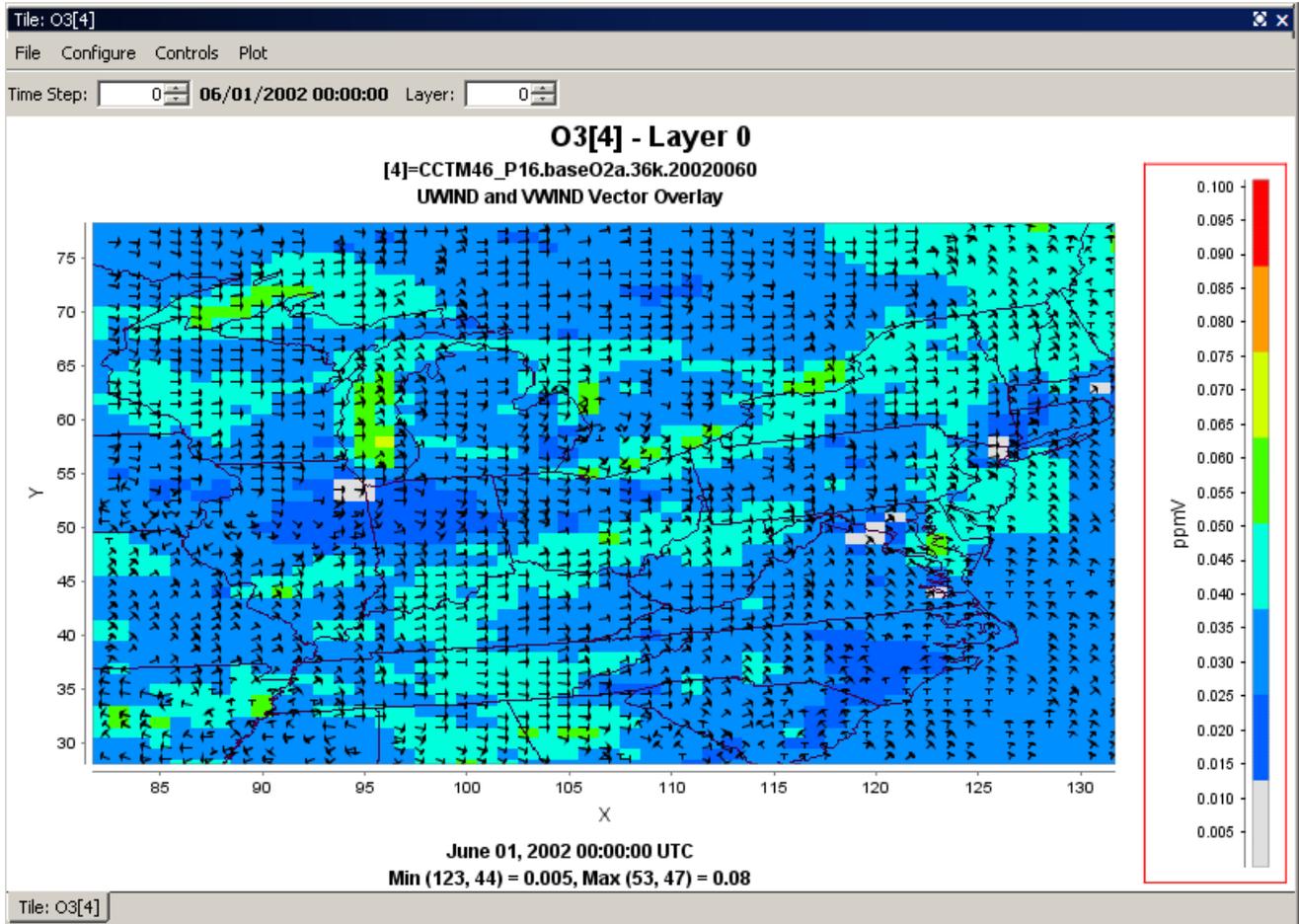
#### 11.4.4 Vector overlays

Users have two options to display vectors (e.g., wind vectors) on a plot. One option is to select the vector plot type, using the procedure described in Section 9.7 to display the vectors on a map. The other option is to display the vectors as an overlay on a tile plot, as described below.

To add a vector overlay to a tile plot, select the Add Overlay>Vectors option from the tile plot's **Plot** pull-down menu. (This capability is planned to be added to the Fast Tile Plot in a future release, these options are currently grayed out for the Fast Tile Plot) The Vector Overlay dialog box asks you to select the formula for the horizontal component and the formula for the vertical component. The formulas listed in the Vector Overlay dialog box are obtained from the formula list in the **Formula** pane. If the formula that you would like to use is not listed in the Vector Overlay dialog box, check to be sure that you have loaded the dataset that contains the variable of interest (for example, UWIND), and that you have created a formula UWIND[x], where x is the dataset number associated with the dataset that contains UWIND. Once you have selected the two components, click **OK**, and the vector overlays will be displayed on the plot.

An example of an ozone concentration tile plot with wind vector overlays is shown in Figure 11-17. By default the vectors are all given the same length. The thickness of the line width used to draw the vector indicates the magnitude of the vector; a larger-magnitude vector is drawn with a thick line, a smaller-magnitude vector with a thinner line. At this time there is no user control over how the vectors are displayed, and there is no option to remove the vectors from a tile plot.

Figure 11-17 Wind Vector Overlay on an Ozone Tile Plot



## 12 Supported Grid and Coordinate Systems (Map Projections)

VERDI makes calls to the netCDF Java library to obtain the grid and coordinate system information about the data directly from the model data input files when the input data files are self describing (CMAQ, SMOKE, WRF netCDF format files).

### 12.1 IO/API formatted data

For IO/API files, support for Lambert conformal conic (LCC) map projection, Universal Transverse Mercator (UTM) map projection, and polar stereographic map projection was added in VERDI 1.1., and Mercator projection in VERDI 1.2. Users are encouraged to provide small input datasets as attachments to Bugzilla, for testing and to facilitate future development efforts. Figures 12-1-12-3 shows example plots generated for a dataset with an LCC, polar stereographic, and Mercator map projections, respectively.

**Figure 12-1 Lambert Conformal Conic Map Projection Example Plot**

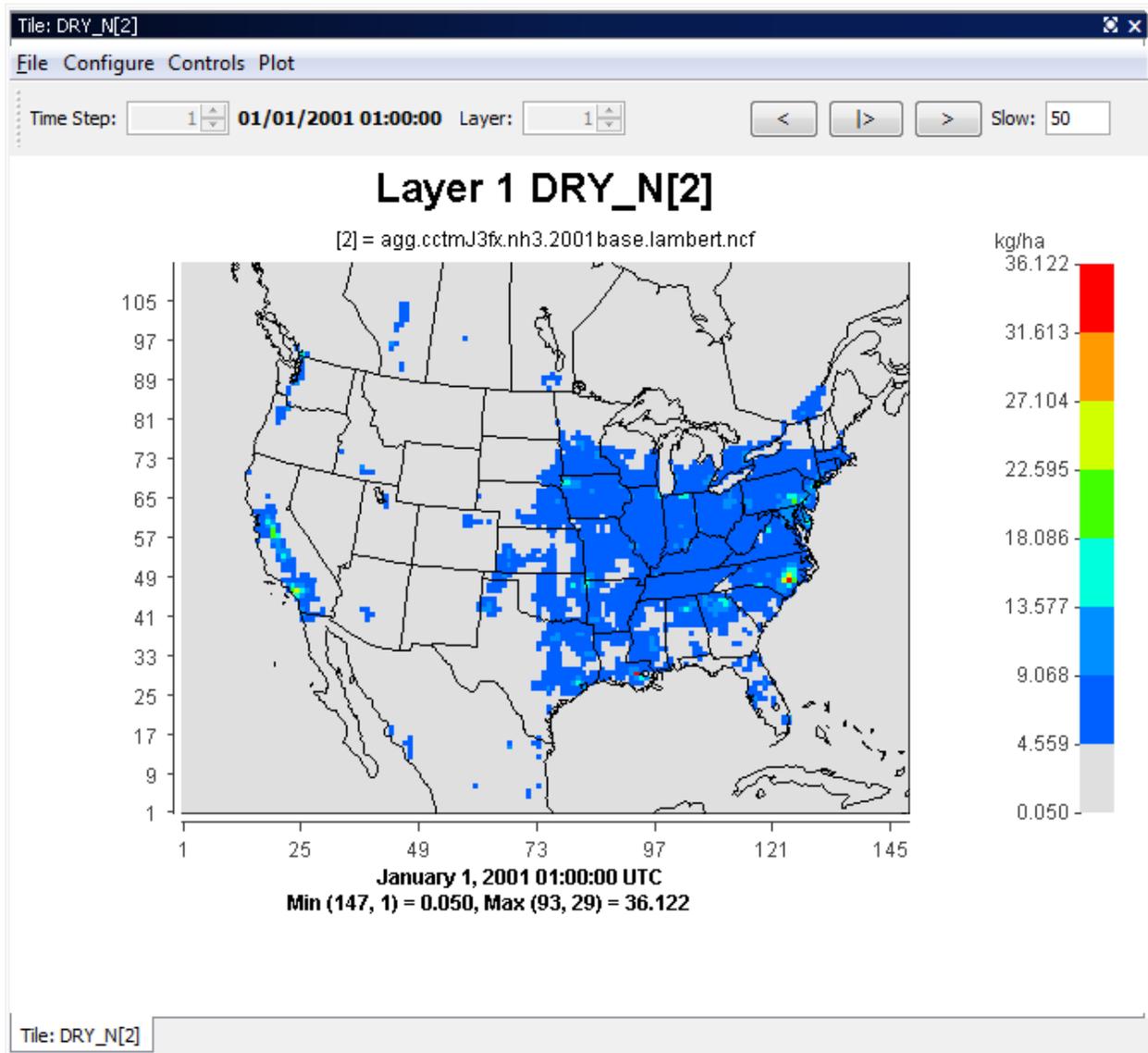
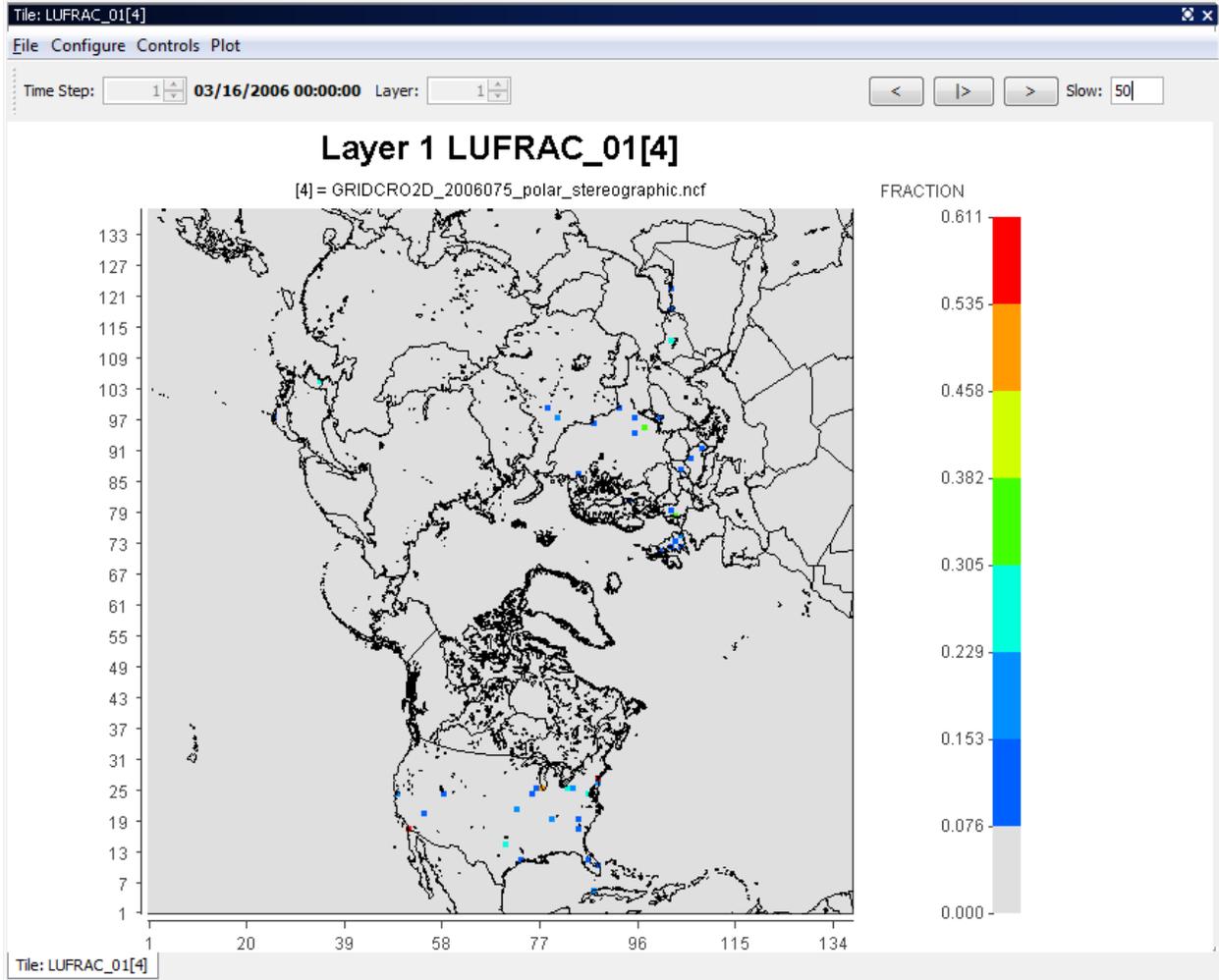
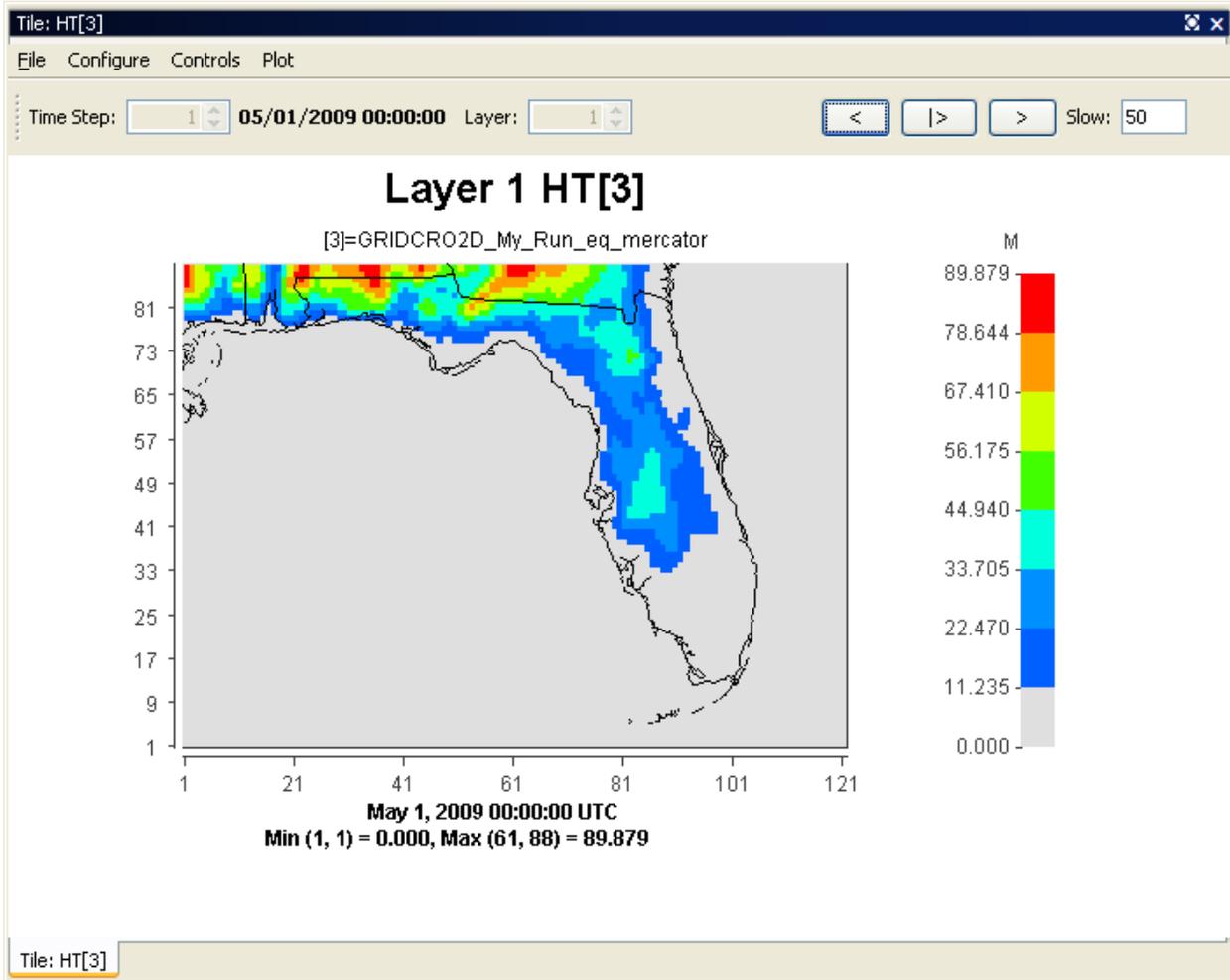


Figure 12-2 Polar Stereographic Map Projection Example Plot



**Figure 12-3 Mercator Map Projection Example Plot**



## 12.2 CAMx Gridded Data

The netCDF-java v4.1 library used in the VERDI 1.2 release includes support for CAMx binary files, using a pre-set default projection. CAMx or UAM binary files contain information about the x and y offsets from the center of the projection in meters, but do not contain information about the projection. The projection information is available in separate diagnostic files, which are part of the CAMx output along with the UAM binaries (Figure 12-4)

**Figure 12-4 Example CAMx diagnostic text file**

```

*****
                        CAMx
*****

CAMx

Simulation start time/date :           0.    060428    06118
Simulation end time/date   :           0.    060429    06119
Time zone                  :              0
Max timestep (min)        :           15.
Met Input interval (min)  :           60.
Emiss Input interval (min):           60.
Output interval (min)     :           60.
Grid Projection Type      : LAMBERT
LAMBERT: center lon/lat   :      -97.000    40.000
LAMBERT: true latitudes   :       45.000    33.000
Master Grid X/Y Origin    :     -2736.000 -2088.000
Master Grid cell size     :       36.000    36.000
Master Grid NCOL NROW NLAY :          148         112         19

```

The netCDF-java v4.1 library writes the default projection information to a text file in the directory where the CAMx binary file is located. This allows the user to review and edit the projection information to make it consistent with the projection specified in the CAMx diagnostic text files. The definitions of the projection parameters used in the camxproj.txt file are defined using Models-3 IO/API format, <http://www.baronams.com/products/ioapi/GRIDS.html>. The user must edit the camxproj.txt file to match the grid description information provided in the corresponding camx.diag file. Figure 12-5 shows the definition for the grid projection parameters for a Lambert projection.

**Figure 12-5 Models-3 IO/API Map Projection Parameters for Lambert****Lambert Conformal Conic:**

```

GDTYP = 2
P_ALP <= P_BET are the two latitudes which determine the projection cone;
P_GAM is the central meridian.
(X_CENT,Y_CENT) are the lat-lon coordinates for the center (0,0) of the
Cartesian coordinate system. Coordinate units are meters.

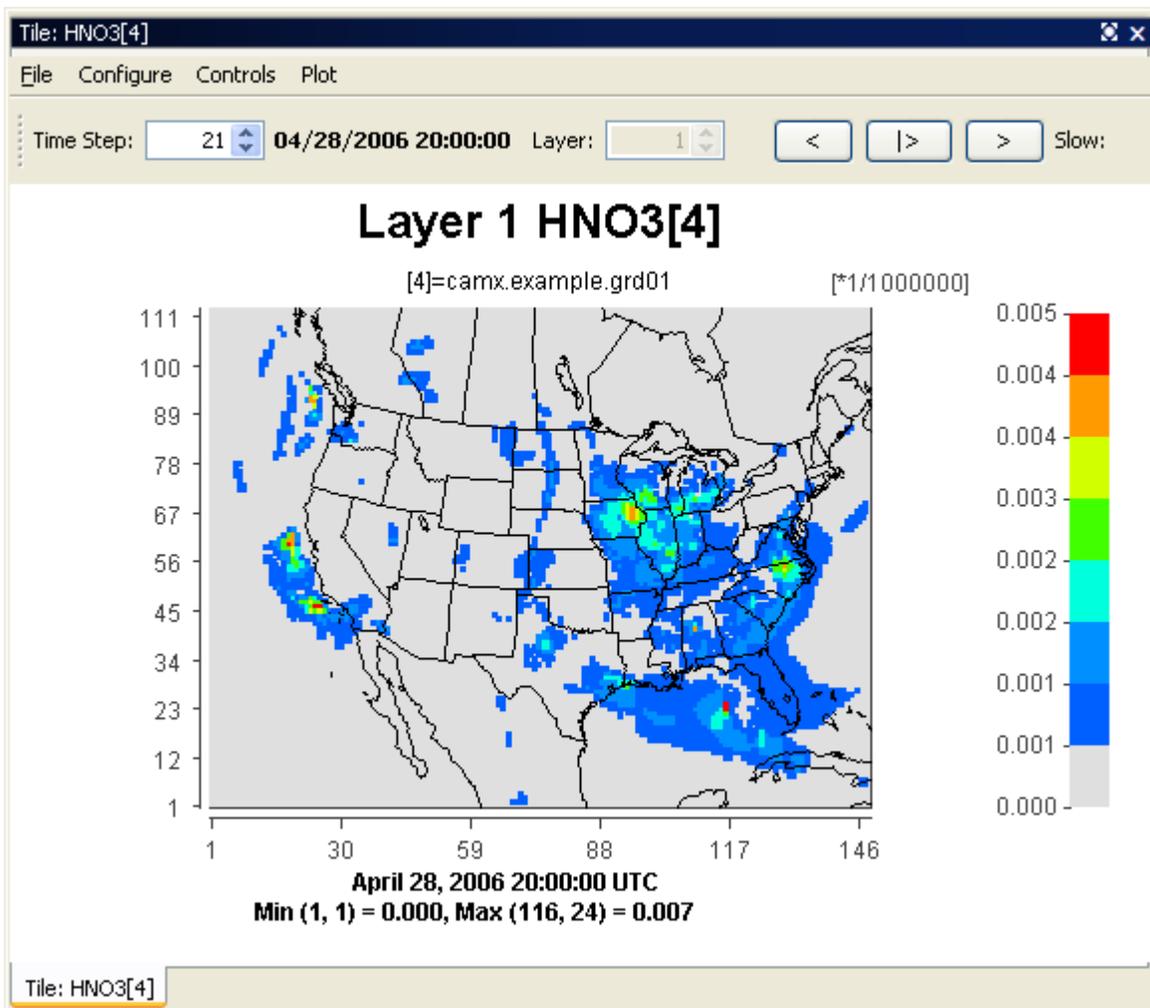
```

Figure 12-6 shows the values of the camxproj.txt once it was edited to match the values of the camx.diag file (Figure 12-4) using the definitions of the Models-3 grid parameters (Figure 12-5). Figure 12-7 shows the resulting Fast Tile Plot of the CAMx sample dataset.

Figure 12-6 Sample Projection File: camxproj.txt

```
# Projection parameters are based on IOAPI. For details, see  
www.baronams.com/products/ioapi/GRIDS.html  
GDTYP=2  
P_ALP=33.0  
P_BET=45.0  
P_GAM=-97.0  
XCENT=-97.0  
YCEN=40.0
```

Figure 12-7 CAMx Example Plot



## 13 I/O API Utilities, Data Conversion Programs, and Libraries

As discussed in Section 6.1, there are routines available to convert gridded input data to I/O API format or new code can be written and contributed to VERDI for use by the community. The I/O API routines that have been written to convert data into this format are discussed in this section. If you are unable to use the available routines to convert your data and have a gridded dataset that VERDI is unable to read, please submit a bugzilla bug or enhancement request with a description of the dataset, and attach a small example dataset for use in development and testing.

The I/O API Interface contains an extensive set of utility routines. There are example conversion programs to convert data from different data formats into the I/O API format. The I/O API Utilities are command line programs that are easy to script for automating analysis and postprocessing. An example of an I/O API Utility that may be useful to VERDI users is m3merge. This utility merges selected variables from a set of input files for a specified time period, and writes them to a single output file, with optional variable-renaming in the process. Another utility that you may find useful is m3extract. This program allows you to extract a few species from a large file and save them to a smaller file on your local computer so you can explore them using VERDI. The I/O API Related Programs and Examples can be found at the following web site: <http://www.baronams.com/products/ioapi/AA.html#tools>.

Airs2m3 is an example of a data conversion program that converts the standard AIRS AMP350 observational data format to the I/O API format. The airs2m3 program requires the following inputs:

- The input AIRS AMP350 print format file name.
- The time zone conversion file (provided with the obs2api program - tzt.dat).
- Additional hour shift variable. The AIRS data are hourly averaged, and a 00 time flag represents the hour 00-01. You may wish to represent that data segment by the ending hour. In that case, a 1 should be entered here.
- Starting year, month, day, hour (GMT) (e.g., 1997 07 10 12).
- Ending year, month, day, hour (GMT) (e.g., 1997 07 16 12).
- Name of output variable (8 characters max) (e.g., O3\_OBS).

## 14 Contributing to VERDI Development

If you have made an improvement to the development of VERDI source code, documentation, or anything else, please consider contributing it back to the community.

Instructions on how to set up the Eclipse Development Environment on Windows, and for running and building VERDI within Eclipse are available on [www.verdi-tool.org](http://www.verdi-tool.org). If you are doing a substantial amount of software development on VERDI, you should become a member of the VERDI project on SourceForge and submit your code updates through the VERDI

subversion repository. You may submit a request to become a member of the VERDI project by submitting a Bugzilla request (see below). The SourceForge web site for VERDI is <http://sourceforge.net/projects/verdi/>.

Once your code has been reviewed by the existing team of developers for VERDI, you will be added to the developer list. Contributions to the code will be tested by the VERDI developers and will be included along the documentation about these contributions in future VERDI releases. Note that anything you contribute must have the same license as the rest of VERDI, i.e. GPL.

Bugzilla can be used to report bugs, check the status of bugs, suggest enhancements, or submit code contributions using the following website:

[http://bugz.unc.edu/enter\\_bug.cgi?product=VERDI](http://bugz.unc.edu/enter_bug.cgi?product=VERDI). First, check to see if your issue is already listed as a bug or request for enhancement. If you do not see an entry that matches please submit a new request or bug using Bugzilla. Please use the Bugzilla comment section to inquire if the feature or bug is already under development. Use Bugzilla to notify the VERDI community about improvements to VERDI that you would like to contribute.

## 15 Known Bugs

As discussed in Section 1.4, you are encouraged to use Bugzilla for submitting bug reports, suggestions, and questions [http://bugz.unc.edu/enter\\_bug.cgi?product=VERDI](http://bugz.unc.edu/enter_bug.cgi?product=VERDI). To enter information on a bug, you must first register with Bugzilla <http://bugz.unc.edu>. If you have already registered, you can search for all bugs that reference VERDI by using the website <http://bugz.unc.edu/query.cgi> and entering the word VERDI then pressing search.

If you would like to be notified of any change of status for a bug that was submitted by another user, you can add your e-mail address to the CC list by clicking on the bug ID, then entering your e-mail address to the Add CC text box.

For a complete list of bugs and enhancement requests that have been submitted for VERDI, and the status of each one, visit the [www.verdi-tool.org](http://www.verdi-tool.org) web site and click on the Query VERDIzilla Tickets link under the Support Quick Links section. The html query that is used to create this list of bugs is extensive, and therefore is not listed here.

## 16 Mathematical Functions

All VERDI visualizations are the result of a formula evaluation. Formulas operate on the variables provided by the datasets. The simplest valid formula consists of a single variable; for example, "O3[1]." Using infix notation, more complicated formulas can be constructed using the following mathematical operators and functions. (Note that the documentation below derives from the equivalent documentation for the Package for Analysis and Visualization of

Environmental data [PAVE], which is available at [http://www.ie.unc.edu/cempd/EDSS/pave\\_doc/EntirePaveManual.html](http://www.ie.unc.edu/cempd/EDSS/pave_doc/EntirePaveManual.html).)

Listed in order of precedence, the functions and operators are:

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. \*\* (power)
3. /, \*
4. +, -
5. <, <=, >, >=
6. ==, !=
7. &&
8. ||

VERDI also supports the following constants:

1. E.....2.7182818284590452354
2. PI.....3.14159265358979323846
3. NROWS.....Number of rows in the formula's currently selected domain
4. NCOLS.....Number of columns in the formula's currently selected domain
5. NLEVELS.....Number of levels in the formula's currently selected domain

## Unary Functions

Unary functions are passed a single argument. Depending on the argument and the function type, the function will return a single value or a matrix of data by performing the function on each cell of the arguments array. For example:

- The function `sqrt(4)` will return 2.0.
- The function `sqrt(O3[1])` will return a matrix containing the square root of each value in the O3[1] variable's array.

The following functions will return a matrix when passed a dataset variable:

1. **abs**.....Returns the absolute value of the argument
2. **sqrt**.....Returns the square root of the argument.
3. **sqr**.....Returns the square of the argument.
4. **log**.....Returns the base 10 logarithm of the argument.
5. **exp**.....Returns Euler's number raised the power of the argument.
6. **ln**.....Returns the natural logarithm of the argument.
7. **sin**.....Returns the sine of the argument. The argument is in **radians**.

8. **cos** .....Returns the cosine of the argument. The argument is in **radians**.
9. **tan** .....Returns the tangent of the argument. The argument is in **radians**.
10. **sind** .....Returns the sine of the argument. The argument is in **degrees**.
11. **cosd** .....Returns the cosine of the argument. The argument is in **degrees**.
12. **tand** .....Returns the tangent of the argument. The argument is in **degrees**.

The following functions will return a single number in all cases when passed a dataset variable:

1. **mean** ..... Average cell value for all cells in currently selected domain.
2. **sum** ..... Sum of all cell values in currently selected domain.
3. **mint** ..... Time step index with minimum value in currently selected domain.
4. **maxt** ..... Time step index with maximum value in currently selected domain.
5. **minx** .....  $x$  index with minimum value in currently selected domain.
6. **maxx** .....  $x$  index with maximum value in currently selected domain.
7. **miny** .....  $y$  index with minimum value in currently selected domain.
8. **maxy** .....  $y$  index with maximum value in currently selected domain.
9. **minz** .....  $z$  index with minimum value in currently selected domain.
10. **maxz** .....  $z$  index with maximum value in currently selected domain.
11. **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  $(i,j,k,tmin..tmax)$ .
12. **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  $(i,j,k,tmin..tmax)$ .

## Binary Operators

Binary operators are not passed a value but operate on the operands to their left and right. Typically, they will return a matrix of data by performing the operation on each cell of the operand's arrays. If both of the operands are single numbers then these binary operators will return a single number. For example:

- $O3[1] * 2$  multiplies each item in the  $O3[1]$  array by 2 and returns the result.
- $O3[1] * O3[3]$  multiplies each item in the  $O3[1]$  array by the corresponding item in the  $O3[3]$  array and returns the result. (Note that this assumes that the arrays are of equivalent shape.)
- $3 * 2$  simply multiplies 3 by 2.

The binary operators:

1. **+** .....Returns the sum of the operands
2. **-** .....Returns the difference of the operands

3. \* .....Returns the product of the operands
4. / .....Returns the ratio of the operands
5. \*\* .....Returns the left operand raised to the power of the right operand.

Boolean binary operators return either 1 or 0 in each cell of the resulting matrix. If the operands are single numbers, then a single 1 or 0 is returned. The Boolean binary operators:

1. < .....Returns 1 if the left operand is less than the right operand, else 0.
2. <= .....Returns 1 if the left operand is less than or equal to the right operand, else 0.
3. > .....Returns 1 if the left operand is greater than the right operand, else 0.
4. >= .....Returns 1 if the left operand is greater than or equal to the right operand, else 0.
5. != .....Returns 1 if the left operand is not equal to the right operand, else 0.
6. == .....Returns 1 if the left operand is equal to the right operand, else 0.
7. && .....Returns 1 if both operands are non-zero, else 0.
8. || .....Returns 1 if either operand is non-zero, else 0.

### Time Step Index

A time step index can be specified after a variable name. For example, “O3[1]:0” is the value of the O3[1] variable at the first time step.

## 17 Command Line Scripting

The commands described in this section can be executed from the command line through either command line arguments or Windows batch files. In Linux, you can edit the verdi.sh script, adding the command options at the end of the last line of the script. If you are using Windows, edit the run.bat script, again adding the command options at the end of the last line, and submitting the script at the windows command line. An example syntax for all commands follows the format

```
<command> <command options> \
```

where the “\” at the end of the command is optional.

### 17.1 Example Script for Linux Users

The following script options will read in the file as the first dataset, select O3[1] as the formula from dataset 1, and create a tile plot of the O3[1].

```
./Verdi.sh -f "$VERDI_HOME/data/CCTM46_P16.O3MAX -s O3[1] -gtype tile"
```

Example script file (Note that quotes (as shown highlighted in red) may be needed around the entire list of parameters” :

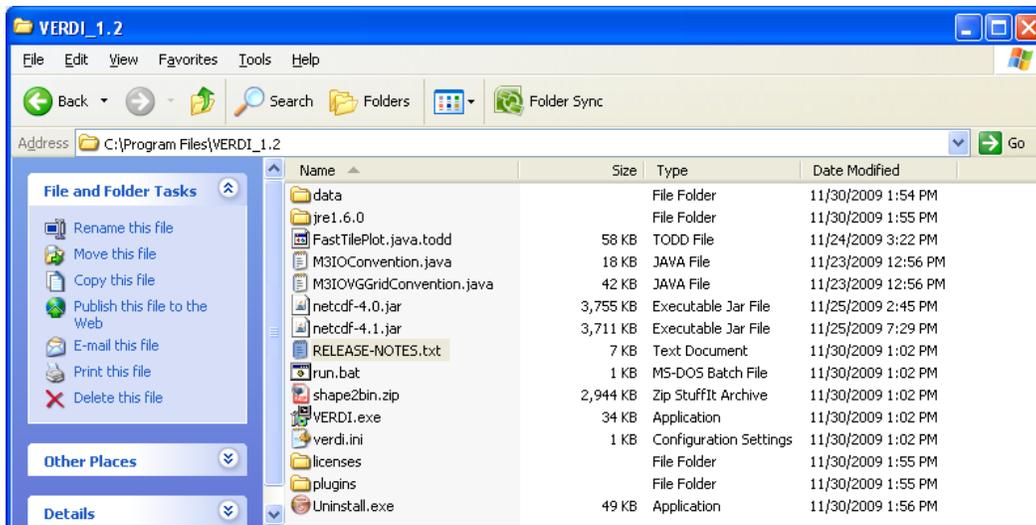
```
#! /bin/csh -f

#### 8hO3 Daily Max Plot
setenv DIR /home/training/verdi_1.2/data/OBS
../../verdi.sh \
"-f $DIR/ACONC_O3_8hr.dmax \
-f $DIR/AQS_overlay_2002_07.ncf \
-s O3[1] \
-s O38[2]*1000"
```

### 17.2 Example Script for Windows Users

Edit the run.bat script in the VERDI\_1.2 directory by right clicking on the file and selecting edit.

Figure 16-1 Location of run.bat script in Windows



The current run.bat in notepad contains a “%1” at the end, that allows it to accept input following the run.bat script using the Windows run command. Unfortunately, this command does not accept directory names that have a space them, such as the “Program Files”. If you would like to enter the script command line options after run.bat, please move the data directory to C:\VERDI\data or some other similar location.

Enter the following in the Run command:

```
"C:\Program Files\VERDI_1.2\run.bat" "-f C:\\Program Files\\VERDI_1.2\\data\\CCTM46_P16.baseO2a.36k.O3MAX -s O3[1] -gtype tile"
```

The other option is to place the script commands within the run.bat itself. Remove the “%1” statement at the end of the run.bat that is provided in the distribution, and add the script options that you would like to use. The following run.bat contains script options that will read in the file C:\Program Files\VERDI\_1.2\data\CCTM46\_P16.baseO2a.36k.O3MAX, select O3[1] as the formula, and create a fast tile plot.

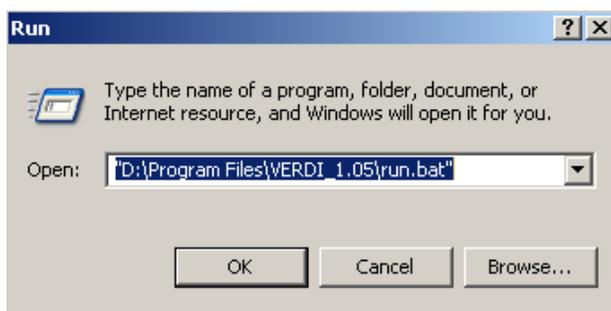
run.bat:

```
cd .\plugins\bootstrap
SET JAVA=..\jre1.6.0\bin\java
%JAVA% -Xmx512M -classpath ".\bootstrap.jar;./lib/saf.core.runtime.jar;./lib/commons-
logging.jar;./lib/jpf-boot.jar;./lib/jpf.jar;./lib\log4j-1.2.13.jar" saf.core.runtime.Boot -f
C:\Program Files\VERDI_1.2\data\CCTM46_P16.baseO2a.36k.O3MAX -s O3[1] -gtype fasttile
```

(The changes to the run.bat are highlighted in red.)

Run the run.bat script by clicking on Start then select Run then use Browse to find the run.bat, or type it in as follows:

**Figure 16-2 Submit run.bat script from Run command**



### **17.3 Commands that can be used (listed in alphabetical order)**

**[-alias <aliasname=definition>]** defines an alias. You can define an alias by creating a definition using variable names and derived variables that are calculated using the mathematical operators described in Section 15: Mathematical Functions. The alias definition does not include the dataset name. The alias is treated like any other formula once the alias definition and the dataset to which it should be applied are specified. If you need to redefine an alias definition, you must first use the **-unalias** command. The alias definitions are saved to a .verdi.alias file in your home directory. VERDI uses this type of optional file in your home directory to maintain a snapshot of the current aliases being used. The following warning will be reported if an alias is defined more than once: “WARNING: Alias <aliasname> already defined, new definition ignored.” You are also responsible for not making circular references. Use the **-printAlias** command to view what aliases are already defined.

- [-animatedGIF<filename>]** creates an animated GIF by doing an X Window Dump (XWD) of each of the time steps in the tile plot, then converting them to GIF images. If there are many time steps in the dataset, there will be a slight delay before you are again given control of the GUI.
- [-closeWindow "<windowid>"]** closes the window with the specified window ID.
- [-configFile <configFileName>]** specifies a configuration file for VERDI to use for configuring subsequent tile plots.
- [-copyright]** prints out copyright information for VERDI.
- [-drawDomainTicks ON|OFF]** turns the domain axis ticks on and off.
- [-drawGridLines ON|OFF]** turns the plot grid lines on and off.
- [-drawLegendTicks ON|OFF]** turns the ticks in the legend on and off.
- [-drawRangeTicks ON|OFF]** turns the range axis ticks on and off.
- [-f <filename>]** tells VERDI to load in this dataset and make it the currently selected dataset. All datasets will stay in memory.
- [-fulldomain]** sets the VERDI domain matching the currently selected dataset to be completely selected. The currently selected dataset is usually the most recently added dataset.
- [-g <tile|line|bar|contour>]** instructs VERDI to create a plot using the specified type and the currently selected formula's data.
- [-gtype <tile|line|bar|contour>]** instructs VERDI to create a plot using the specified type and the currently selected formula's data.
- [-help|fullhelp|usage]** display the information on all the command line arguments available. Each of these three versions performs the identical function.
- [-legendBins "<bin0,bin1,...,bin\_n>"]** causes VERDI to use the specified numbers as breaks between colors on subsequent plots. The value of this argument is a comma-separated list of numbers. For example, **-legendBins "1,10,100,1000"** will cause plots to be created with three colors that correspond to values of 1-10, 10-100, and 100-1000. To go back to the default method for determining breaks between bins, enter **-legendBins DEFAULT**.
- [-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.
- [-openProject <VERDIProjectName>]** opens a previously save VERDI project.
- [-printAlias]** prints existing alias definitions.
- [-quicktime (NEW)]** creates a Quicktime movie of the currently selected plot.

- [-quit|exit]** ends the VERDI session.
- [-raiseWindow <windowid>]** raises the window with the specified plot ID (i.e., brings it to the front).
- [-s "<formula>"]** loads the specified formula into VERDI's memory, and makes it the currently selected formula.
- [-save2ascii "<filename>"]** export data to a tab-delimited data file suitable for reading into a spreadsheet application such as Excel.
- [-saveImage "<image type>" <file name>]** saves the most recently created plot. This command works for all plot types. Supported formats include PNG, BMP, TIF, and JPG.
- [-scatter "<formula1>" "<formula2>"]** creates a scatter plot using the two formulas specified. Note that the formulas for the two components should already have been loaded into VERDI, and they are case sensitive.
- [-showWindow <windowId> <timestep>]** sets the time step of the window with the specified window ID to the specified time step. The time step must be within the allowable range for the dataset.
- [-subDomain <xmin> <ymin> <xmax> <ymax>]** sets the VERDI domain matching the currently selected dataset to the bounding box specified by its arguments. The currently selected dataset is the most recently added dataset. It is often handy to type **-subdomain** commands into VERDI's standard input if you are trying to select a very precise subdomain (such as that needed for a vertical cross-section plot).
- [-subTitle1 "<sub title 1 string>"]** allow you to control a plot's subtitles if desired. Subsequent plots will use the default subtitles, unless these arguments are used again.
- [-subTitle2"<sub title 2 string>"]** allow you to control a plot's subtitles if desired. Subsequent plots will use the default subtitles, unless these arguments are used again.
- [-subTitleFont <fontSize>]** allow you to control the font size of the subtitle of a plot.
- [-system "<system command>"]** sends the specified command to the operating system's command line.
- [-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.
- [-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.
- [-titleFont <fontSize>]** allows you to control the font size of the title of a plot.
- [-titleString "<title string>"]** sets the title for the next plot made to the specified title. Subsequent plots will use the default VERDI title, unless this argument is used again.

**[-ts <time step>]** sets the selected time step for each formula in VERDI's memory to the specified step number, where the first step number is denoted by 0. This will remain the selected time step until you change it. It affects only tile plots and vertical cross-section plots.

**[-unalias <aliasname>]** is used to undefine an alias.

**[-unitString "<unit string>"]** can be used to override the default unit label used for plots. The default value comes from the dataset(s) themselves.

**[-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 already have been loaded into VERDI, and they are case sensitive.

**[-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 already have been loaded into VERDI, and they are case sensitive.

**[-version]** prints out information about the VERDI version being used on the standard output stream.

**[-verticalCrossPlot X|Y <row/column> (NEW)]** creates a vertical cross-section plot. You indicate whether this will be an x or y cross-section plot and what row or column to use as the base.

**[-windowid]** prints the window ID of the currently selected plot.

## 18 Areal Interpolation Calculations

Before calculating the average value for a polygon segment, the area for each polygon is calculated using the projection of the grid system loaded. The system then calculates the area of overlay between each grid cell and the polygon segment.

The total contribution of a value (concentration, deposition, rainfall, etc) from each cell for a given polygon segment is calculated using the following equation:

$$TV_i = \sum (O_{rci} * V_{rc}) \quad \text{where}$$

$O_{rci}$  = Area of overlay of cell at row r and column c with segment i,

$V_{rc}$  = value of cell at row r and column c, and

r and c iterate across the rows and columns of the grid.

The Average Value is calculated by dividing the total value by the area of the polygon segment:

$$\text{Average}V_i = TV_i / A_i \quad \text{where}$$

$A_i$  = Area of the polygon segment i

## 19 Licenses for JAVA Libraries used by VERDI

VERDI has been developed using a number of open source Java libraries. Table 21-1 contains a list of the jar files or Java libraries that are used by VERDI, a link to where each library may be acquired, a link to the location where the license is referenced by the documentation for each library, as well as a link to each license. The distribution for VERDI contains a sub-directory called licenses. This directory contains a copy of the licenses for the open source Java libraries used by VERDI.

**Table 19-1. JAVA Libraries used by VERDI**

<b>List of Java Libraries</b>	<b>Where to acquire software</b>	<b>Reference to Software License for Software</b>	<b>Link to license or the specific license number</b>
JTS (java topology suite)	<a href="http://sourceforge.net/projects/jts-topo-suite/">http://sourceforge.net/projects/jts-topo-suite/</a>	<a href="http://www.vividsolutions.com/JTS/jts_frame.htm">http://www.vividsolutions.com/JTS/jts_frame.htm</a>	<a href="http://www.gnu.org/copyleft/lesser.html">http://www.gnu.org/copyleft/lesser.html</a>
log4j-1.2.13.jar	<a href="http://www.unidata.ucar.edu/software/netcdf-java/">http://www.unidata.ucar.edu/software/netcdf-java/</a>		<a href="http://logging.apache.org/log4j/1.2/license.html">http://logging.apache.org/log4j/1.2/license.html</a>
saf.core.runtime.jar	<a href="http://safr.sourceforge.net">http://safr.sourceforge.net</a>	<a href="http://www.eclipse.org/stp/cf/saf/SAFcore.html">http://www.eclipse.org/stp/cf/saf/SAFcore.html</a>	<a href="http://www.eclipse.org/legal/epl-v10.html">http://www.eclipse.org/legal/epl-v10.html</a>
vecmath.jar	<a href="http://java3d.j3d.org/">http://java3d.j3d.org/</a>	<a href="https://vecmath.dev.java.net/">https://vecmath.dev.java.net/</a>	<a href="http://www.gnu.org/licenses/old-licenses/gpl-2.0.html">http://www.gnu.org/licenses/old-licenses/gpl-2.0.html</a>
geoapi	<a href="http://geoapi.sourceforge.net/">http://geoapi.sourceforge.net/</a>	<a href="http://www.codeplex.com/GeoAPI/license">http://www.codeplex.com/GeoAPI/license</a>	LGPL V2.1, Feb. 1999
Gt2	<a href="http://sourceforge.net/projects/geotools/">http://sourceforge.net/projects/geotools/</a>	<a href="http://docs.codehaus.org/display/GEOTDOC/00+Source+License">http://docs.codehaus.org/display/GEOTDOC/00+Source+License</a>	<a href="http://www.gnu.org/licenses/lgpl-2.1.txt">http://www.gnu.org/licenses/lgpl-2.1.txt</a>
hsqldb	<a href="http://hsqldb.org/">http://hsqldb.org/</a>		<a href="http://hsqldb.org/web/hsqLlicense.html">http://hsqldb.org/web/hsqLlicense.html</a>
jfreechart	<a href="http://sourceforge.net/projects/jfreechart/">http://sourceforge.net/projects/jfreechart/</a>	<a href="http://developer.vrjuggler.org/browser/trunk/juggler/external/jfreechart/LICENSE.txt?rev=15441">http://developer.vrjuggler.org/browser/trunk/juggler/external/jfreechart/LICENSE.txt?rev=15441</a>	LGPL V2.1, Feb. 1999
jscience	<a href="http://jscience.org/">http://jscience.org/</a>	<a href="http://swik.net/Jean-Marie-Dautelle">http://swik.net/Jean-Marie-Dautelle</a>	<a href="http://swik.net/License:BSD/BS D+License+Text">http://swik.net/License:BSD/BS D+License+Text</a>
Piccolo 1.2	<a href="http://www.cs.umd.edu/hcil/jazz/">http://www.cs.umd.edu/hcil/jazz/</a>	<a href="http://www.cs.umd.edu/hcil/jazz/download/open-source.shtml">http://www.cs.umd.edu/hcil/jazz/download/open-source.shtml</a>	<a href="http://opensource.org/licenses/bsd-license.php">http://opensource.org/licenses/bsd-license.php</a>
Ucar_ma2	<a href="http://www.unidata.ucar.edu/software/netcdf-java/v2.2.16/javadoc/ucar_ma2/package-summary.html">http://www.unidata.ucar.edu/software/netcdf-java/v2.2.16/javadoc/ucar_ma2/package-summary.html</a>	<a href="http://www.unidata.ucar.edu/software/netcdf-java/">http://www.unidata.ucar.edu/software/netcdf-java/</a>	<a href="http://www.gnu.org/copyleft/lesser.html">http://www.gnu.org/copyleft/lesser.html</a>
Repast symphony GIS	<a href="http://repast.sourceforge.net/">http://repast.sourceforge.net/</a>		<a href="http://repast.sourceforge.net/repast-license.html">http://repast.sourceforge.net/repast-license.html</a>

## 20 Acknowledgements

Contributions to VERDI from the community are greatly appreciated.

### **20.1 Data Contributions:**

#### **20.1.1 Sample CAMx dataset**

Marco \_Rodriguez, CIRCA at Colorado State University <http://www.cira.colostate.edu/>

#### **20.1.2 Sample Mercator Projection dataset**

Tanya Otte, Atmospheric Modeling and Analysis Division, <http://www.epa.gov/amad/index.html>

### **20.2 Data Reader Contributions**

#### **20.2.1 I/O Service Provider (IOSP) interface for CAMx:**

Barron Henderson, ORISE Fellow, EPA, Ph.D. student UNC Chapel Hill

#### **20.2.2 Incorporating the IOSP into netCDF netcdf-java v4.1 library:**

John Carron, Unidata, <http://www.unidata.ucar.edu/software/netcdf/index.html>