Analysis and Visualization of Space Science Model Output and Data with CISM-DX

M. Wiltberger, [1] R. S. Weigel, [2] M. Gehmeyr, [2] T. Guild [3]

Abstract. Among the many challenges facing space physics today is the need for an analysis and visualization package which can examine the results from numerical and empirical models as well as observational data. In this paper we introduce the CISM-DX package as a possible answer to this problem. We have created a tool which is capable of completing highly sophisticated visualizations of the results from a diverse set of models throughout the connected Sun-Earth system. We begin with an overview of the open source software packages, OpenDX and Octave, which form the basis of our package. Next we discuss the extensions to these packages we have added to enhance their utility to the space physics community. We also discuss the extensive data base of observations that is included with the distribution. Results from heliospheric simulations are used to highlight the ease at which a novice user can begin to utilize these tools. An energy budget calculation during a magnetospheric substorm simulation illustrates the more advanced analysis possible with the CISM-DX package. Next we highlight the extensive database of observational measurements included with the distribution, along with tools for displaying. This package is being released under an open source licensing agreement to facilitate and encourage community use and contribution.

1. Introduction

Improvement in space weather prediction and specification requires the transfer of models and analysis tools from the research community to the application community. Two important aspects of this transition are the ability to visualize the results from complex 3D physics based models and to compare the model output to observations for validation purposes. CISM-DX is a set of programs that provides access, through an open source software license, to a visualization and analysis tool for the results from numerical and empirical models as well as observational data.

One constant among space physics datasets and models is their diversity. Most models within the space physics community have unique grids, output file formats, and physical parameters. Modelers typically develops their own method for examining the results from his simulation often with little consideration for portability or ease of comparison with other models and data. Similarly, a diversity of data formats for measurements also exists. There are two main classes of approaches that can be used to address this problem. An analysis tool can be built from scratch and customized for specific models or data sources, or existing software packages can be adapted and built upon for use within the space physics community.

The first approach has been used to create the AF Geospace [*Hilmer*, 1999] and VisBard [*Roberts et al.*, 2004] packages. AF Geospace provides access to several empirical models, e.g. the CRERESELE radiation belt model, the Interplanetary Shock Propagation Model, the Parameterized Ionospheric Model, etc., as well as data from the DMSP satellites. VisBard provides a graphical interface for simultaneous viewing of scalar and vector data from multiple spacecraft. In each case the user installs a software package that provides access to the data and a user interface for customizing the visualization. This approach results in tools which are tailored to meet the need of original models and data, but may be difficult to extend to new domains especially by users outside the original development team.

Another approach is to extend the capabilities of existing software packages to meet the needs of the space physics community. One example is the Unidata project has developed the Integrated Data Viewer (IDV) as a tool analyzing and visualizing geoscience data [*Murray et al.*, 2003]. It is built upon the VisAD package developed for importing and visualizing data [*Hibbard et al.*, 1994; *Hibbard*, 2002]. The VisAD package has the strength of being written in Java and is therefore platform independent by default, but its ability to deal

[1]Hight Altitude Observatory, National Center for Atmospheric Research, Boulder, CO USA.

[2]Laboratory for Atmospheric and Space Physics, University of Colorado, Boulder, CO USA.

[3]Center for Space Physics, Boston University, Boston, MA USA.
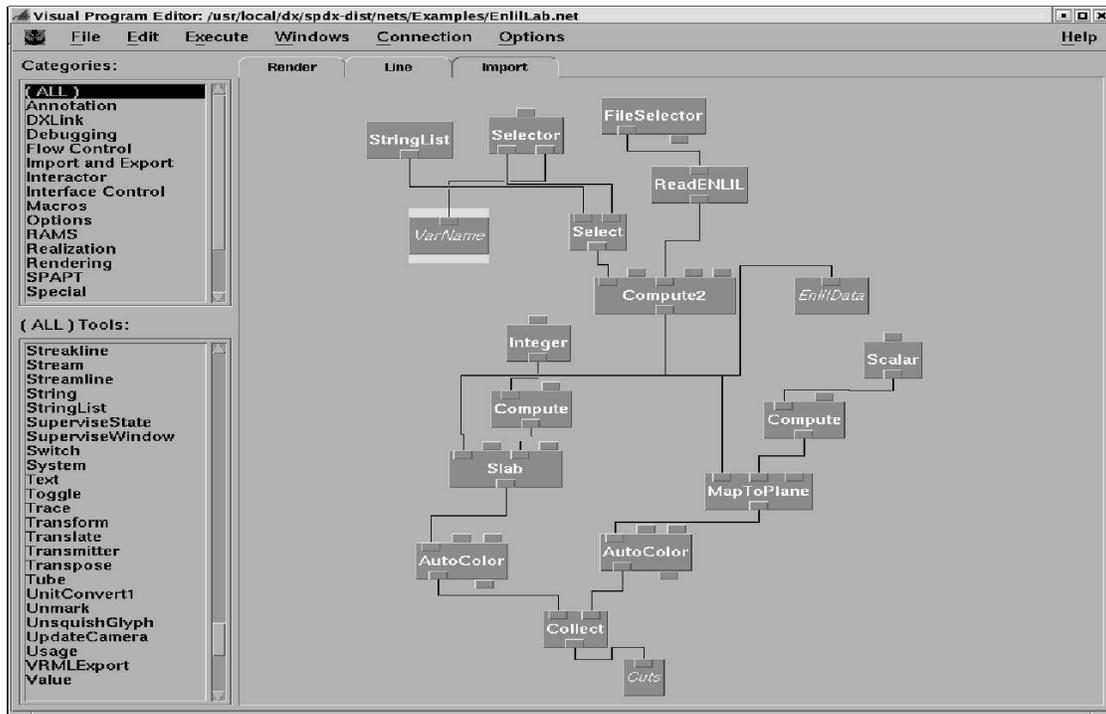
Figure 1. The VPE is where networks need for creating visualizations are written in OpenDX.

with irregularly gridded 3D data sets common in space physics is not clearly illustrated. IDV extends the capabilities of VisAD to process geoscience data such as satellite imagery and National Weather Service Radar data. It is designed to read netCDF files that conform to their best practice suggestions [*Unidata*, 2004]. This approach allows the development team to focus on creating a user interface which is flexible and can be customized and extended by users.

Our approach has been the development of CISM-DX, which extends the capabilities of two existing software packages, OpenDX and Octave. This strategy is similar to that used by IDV with OpenDX playing the role of VisAD. Open Data Explorer, or OpenDX for short, was originally developed by IBM as a commercial visualization package [*IBM*, 1998]. It forms the core of the CISM-DX visualization and analysis package. OpenDX has a graphical user interface (GUI) that allows the user to perform data visualization and analysis. Octave is a high level computer language for solving linear and nonlinear problems using vectors and matrices [*Eaton*, 2002]. In addition to these software tools, we also include a large database of observational measurements in a single data format to allow for easy analysis and direct comparison between measurements and model output.

Both OpenDX and Octave were chosen in part because they are Open Source software packages. There are a variety of different types of Open Source licenses, but they all share the common feature that the source code is included as part of the software distribution. These packages are backed by a development community that invites users to provide feedback and in some cases encourages them to add to the distribution. We are releasing CISM-DX under an open source licensing agreement to encourage community use and to allow for ongoing improvement and modification.

The primary goal of the CISM-DX project is to enable good science by providing a set of tools developed and used by scientists for accessing and analyzing numerical model output, measurement data, and empirical geophysical models. We provide the user with a basic tutorial on how to get started, access to tools for coordinate transformations and other commonly used packages within the space physics community, examples for visualizing numerical model results, and a mechanism for analyzing observational measurements.

This paper provides an overview of the uses and contents of the CISM-DX package. We begin, in Section 2, with an overview of OpenDX and Octave. In Section 3, we discuss the additions made to those packages to meet the needs of the space physics scientists and students.
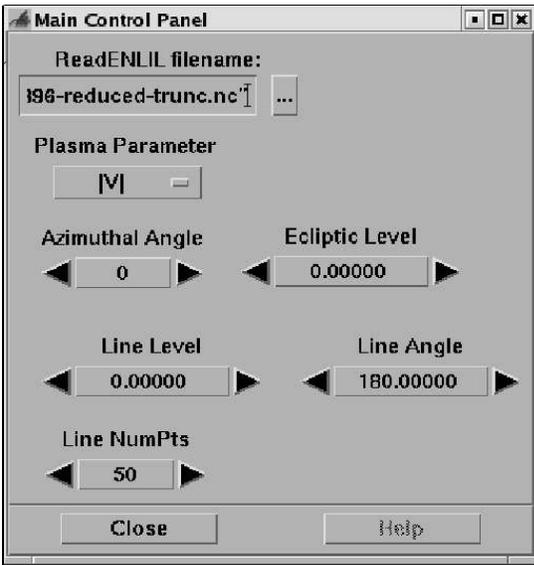
Figure 2. The VPE supports the creation of straight forward control panels that allow the novice user to manipulate the visualization without needing to learn how to write their own networks.

In addition, we provide an introduction to the database of observations that are included with the distribution. Section 4 presents selected examples from the package, beginning with a discussion of a heliospheric model network as an illustration of the basic visualization class of networks. We also discuss a more complicated analysis by examining the energy distribution in the magnetosphere during a substorm. This section concludes with a network showing how the observations within the database can be visualized with CISM-DX. In the final section we summarize how the user may gain access to this package and contribute to its open source development process.

2. Overview of Software Infrastructure

As previously mentioned CISM-DX is built from two key pieces of software. OpenDX provides the visualization infrastructures as well as the process for creating user interfaces to the model and data. Octave allows for extensive data manipulation and reduction operations. In this section, we provide an overview of each software package concentrating on its intrinsic capabilities.

2.1. Overview of OpenDX

The OpenDX package allows for many levels of users to benefit from the same package. A novice can use the menus and control panels provided by the GUI to adjust

an existing OpenDX program. More advanced users can create their own program to produce customized visualizations or detailed analysis of the simulation results. Since the GUI is backed up by a fully developed application program interface (API) an expert programmer can create customized modules that extend the capability of OpenDX by adding new functions or importing highly structured data sets.

The core of the OpenDX graphical user interface is the Visual Programming Environment (VPE). An example program is shown in Figure 1. The VPE is where programs, often referred to as networks, are created that are executed to produce visualizations. The left hand side of the VPE contains the list of modules that perform various portions of the visualization process. The Import module is one way of bringing data into OpenDX. This module can read simple arrays from a variety of formats, e.g. Common Data Format (CDF), network Common Data Form (netCDF), Hierarchical Data Format (HDF), or can be used in combination with the data prompter mode of OpenDX to import user specific data formats. The AutoColor module allows an object to be colored based upon data values. The Isosurface module creates constant value surfaces which become contour lines for a 2-dimensional object. The Streamline module traces the the instantaneous motion of a particle through a vector field. The AutoGlyph module can be used to create a series of arrows through the data, whose size and direction are controlled by the underlying vector field. The Image module completes the visualization process by taking the various graphical components and rendering them into an image window where the user can directly interact with the image.

The utility of the OpenDX is greatly expanded by the inclusion of two additional modules that do not perform standard visualization tasks. The Compute module is a general purpose tool which performs mathematical operations, e.g. addition, vector operations, trigonometric functions, etc., on the data. In addition, the Compute module can manipulate the underlying grid, making grid transformations as simple as entering mathematical expressions. The Map module is used to interpolate data to any other object ranging from streamlines to isosurfaces. The inclusion of these modules moves OpenDX beyond a simple visualization tool into the arena of a sophisticated data analysis tool.

The strength and flexibility of these modules is possible because of the underlying data structure known as the DX Field Object. This object is composed of three components: positions, connections, and data. The po-

Table 1. OpenDX modules in CISM-DX package.

Name	Type	Description
ReadLFM	Import	Read LFM Global Magnetospheric Data
ReadRaeder	Import	Read Raeder Global Magnetospheric Data
ReadBATSRUS	Import	Read BATSRUS Global Magnetospheric Data
ReadENLIL	Import	Read ENLIL Heliospheric Data
ReadWSA	Import	Read WSA Heliospheric Data
ReadTING	Import	Read TING Thermosphere Ionosphere Data
ReadMAS	Import	Read MAS Solar Corona Data
T89,96,01,03	Import	Create DX Field with Tsyganenko Model
IGRF	Import	Creates DX Field with International Geomagnetic Reference Field
ReadTrajectory	Import	Reads spacecraft Trajectory in SSC format
OrbitSync	Transformation	Determines current location of spacecraft
YrDayHr	Transformation	Convert Seconds in UT
SPTransform	Transformation	Converts vectors between coordinate systems
OpenClose	Transformation	Determine if Field Lines are Open or Closed
MapFtPts	Transformation	Map Foot points along dipole field lines to ionosphere
FileLoop	Interactor	Create list of files to analyze
OctaveOp	Special	Call and pass data to/from an Octave program

sitions component describes the locations in space of the points which compose the computational grid. The connections component describes how the positions interconnect with each other. This allows the interpolation functions of the Map module to work on grids as simple as regular Cartesian mesh to those as complex a finite element meshes. The data component of the field contains the actual information and can be a scalar, vector, or matrix at each location in space. The Field Object is limited by its inability to directly support Spherical and Cylindrical coordinate systems. In fact, the modules that operate on vector fields assume a Cartesian description of the field.

It is important to note that while the DX Field Object is very flexible in adapting to the grid structures used within numerical models in the space physics environment, it is often necessary to move beyond the features available in the simple Import module and create customized Import modules such as the ReadENLIL module shown in Figure 1. OpenDX comes with a complete set of documentation for the API functions which make it possible to create and manipulate the three fundamental components of the DX Field Object for anyone with an understanding of the C programming language.

The VPE also provides access to a series of interface modules that allow the creation of networks with intuitive control panels, as seen in Figure 2. These control panels allow the novice user to easily control the visualization. Without needing to understand how to create networks the novice can adjust the data being imported, control the parameter being plotted, and the location of cuts through the data space. We believe that this is an important feature because it allows us to create a series of basic networks which permit the user to begin using the CISM-DX package with a minimal amount of initial work.

The actual execution of networks to produce visualization is conducted on a client-server basis. Within the VPE menu the user can request the execution of the network and then manipulate the resulting visualizations in the image window created by the Image module. This client-server approach adds additional capabilities beyond the VPE approach. Individual networks can be called in a batch mode, which allows for processing of large amounts of data without additional user input. The client can also be a Java applet, which allows a the creation of web based visualizations.

2.2. Overview of Octave

The second primary open source software package that CISM-DX relies on is Octave. Octave is a high–

Table 2. Octave Function Categories in CISM-DX package.

Type	Description
Validation	Code used to compute skill scores and other metrics
Forecasting	Select models from the CISM Forecast Models
Plotting	Plot routines for time series data
Analysis	Averaging, re-sampling and filtering
Tutorials	Example usage of key Octave functions in CISM-DX
Download	Download data from original source
Parse	Transform data files from original provider into HDF
Import	Extracts time series from HDF file data base

level mathematical language is used to expand the numerical and computational capabilities of the CISM-DX package. The Octave–interpreted programs distributed with CISM-DX have been written to be cross-compatible with the commercial mathematical analysis product Matlab [*Works*, 2002]. We introduce scripts that rely on Octave for the following reasons: (1) it is a familiar program to many space scientists who use Matlab because its syntax is similar; (2) it is available on many operating systems; and (3) we have developed an OpenDX module, OctaveOp, that allows the Octave interpreter to be called from OpenDX. This relieves us of the burden of having to re-write many pre-existing Octave data analysis and preparation programs for OpenDX.

There are several other programming languages that are similar in scope and capability as Octave. IDL is a language that shares a user population similar to Matlab [*RSI*, 2003]. In common with Matlab, recent additions to the language have allowed IDL–interpreted scripts, compiled with a valid IDL license, to be deployed and used stand–alone on computers without a license. We have chosen to use Octave's stand–alone ability instead of these other products because Octave's open–source license gives increased certainty that our code developments will be usable long into the future. Since we are using OpenDX as the GUI, the advantage of other products over Octave is diminished.

We anticipate that not all users will have need for the high–performance 3–D visualization and GUI capabilities available in OpenDX. One subset of our user and contributor base includes those interested in doing extensive data manipulation and reduction, or detailed numerical analysis of time series data and subsets of physics model output. Although many computations

are possible with OpenDX from the Compute module, the code base of Octave is large and well–developed and its scripts are often well–suited for certain types of analysis. For example, Octave has high–dimensional matrix manipulation capabilities, bindings to the BLAS and LAPACK libraries, and a suite of statistics programs. At present CISM-DX has been configured so that users that do not have OpenDX available on their system may still use all of the Octave programs distributed with the CISM-DX package. However, this may change as software is developed and included in the CISM-DX code base which integrates the two programs more closely.

3. CISM-DX Extensions

The CISM-DX package extends the capabilities of the core software packages to meet the needs of the space physics researchers. In Section 3.1 we discuss the new modules that have been added to OpenDX. Most these modules import data from various numerical models, while others add specific space physics tools such as the capability to transform between coordinate system and empirical models such as the International Geomagnetic Reference Field. The extensions added to Octave, including data manipulation functions and empirical model codes, are discussed in Section 3.2. We conclude this section with a discussion of the extensive observational database that is part of the CISM-DX package.

3.1. OpenDX Extensions

The CISM-DX package comes with a large number of modules and networks designed to provide tools for the novice and advanced user. It is not practical to give detailed discussions of each of these networks in this

Table 3. Data directly viewable with Octave and OpenDX code in the CISM-DX package.

Source	Type	Approx. Time res.	Time span
WDC C1[a]	Ground Magnetometers (30)	1–min	Avg of 8 yrs/magnetometer
FMI[b]	Ground Magnetometers (20)	1–min	Avg of 5 yrs/magnetometer
WDC C2[c]	Dst index	1–hr	1957–2002
NGDC[d]	Ap, F10.7 Indices	1–day	1963–2002, 1947–2002
NSSDC[e]	ULYSSES, GOES-8, GOES-10	1–min	1990–2003, 1995–2003, 1999–2004
NSSDC[e]	ACE, WIND, GEOTAIL	1–min	1998–2002, 1993–2000, 1992–2002
NSSDC	OMNIWeb and OMNIWeb2	1–hr	1963–2002
ISGI[f]	quiet day list, aa index, am index	1–day,3–hr,3 –hr	1932–2004
SEC[g]	ACE real-time data	1–minute	2000-2004
[h]	SAMPEX MeV electron Flux	1–day	1993–2002

[a]World Data Center for Geomagnetism, Copenhagen

[b]Finnish Meterological Institute

[c]World Data Center for Geomagnetism, Kyoto

[d]National Space Science Data Center

[e]National Geophysical Data Center

[f]International Service of Geomagnetic Indices

[g]Space Environment Center

[h]Dr. S. Kanekal, Personal Communication

paper. Table 1 provides a summary of the additional modules included in the CISM-DX package which are explained in more detail in this subsection.

One of the more challenging aspects of using OpenDX is importing data from external simulations into the OpenDX field object, so the majority of new modules for CISM-DX deal with importing data from space physics simulations. The ReadENLIL module imports data from the the netCDF files created by the ENLIL [*Odstrcil et al.*, 2002] heliosphere mode. This module handles the conversion of data from spherical coordinates into Cartesian coordinates. The RealLFM module imports data from the HDF output files created by the Lyon-Fedder-Mobbary global magnetospheric model [*Lyon et al.*, 2004]. The ReadBATSRUS module imports data from the BATSRUS global magnetospheric simulation [*Gombosi et al.*, 1998]. The BATSRUS code outputs data for the MHD state vector into a single data file at each time step. The simulation uses a block adaptive mesh, so the major task of this module is to construct the connections portion of the grid information, which must be done manually using information from the dump file. Data from the UNH global magnetospheric simulation [*Raeder et al.*, 1997; *Raeder*, 2003] can be imported using the ReadRaeder

module. In this case the developers have created their own customized machine independent data format. We have written a module that utilizes subroutines initially developed by them to read the data and convert it into the OpenDX field. *Mikic et al.* [1999] developed a MHD model for the global solar corona. The simulations are completed in spherical coordinates on a highly staggered mesh. In addition, each variable is output into a separate HDF file. The ReadMAS module imports data from each file and places them onto a single regular mesh. The module also creates a copy of the vector quantities in Cartesian coordinates to allow them to work with other OpenDX modules. Since the spherical mesh is periodic in the azimuthal direction, we also manually construct the connections between grid points which allows the tracing of field lines to pass through this artificial boundary. The Thermosphere Ionosphere Nested Grid developed by *Wang et al.* [1999] is imported into OpenDX with the ReadTING module. In this case the reader imports data from a HDF file containing multiple time steps each identified by its UT. The state vector is quite extensive and includes the density and temperature of electrons, neutrals and several ion species. In addition, the velocity for the neutral fluid is reported. The simulation uses a regular spher-
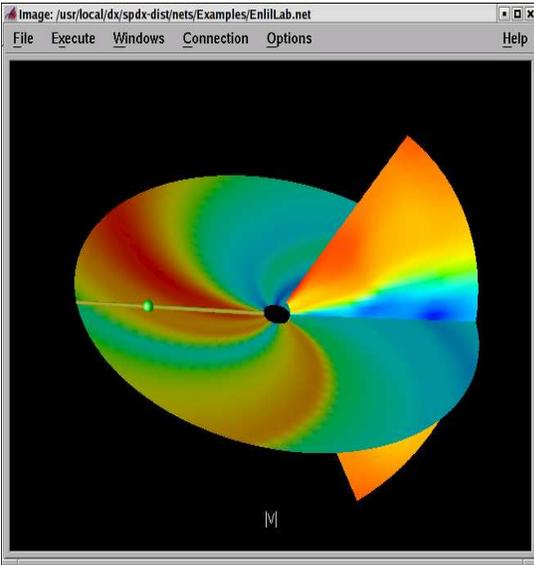
Figure 3. A visualization of the heliosphere produced by CISM-DX from the ENLIL solar wind model. The magnitude of velocity is shown on an ecliptic plane as well as a cut directly opposite the Earth indicated by the green sphere. The stream structure is clearly seen in the ecliptic plane, while the other cut shows the increase in flow speed with heliographic latitude.

ical grid which is periodic in the azimuthal direction. The T89, T96, T01, and T03 modules place results from the empirical magnetic field models onto a grid with user inputs for the various parameters required by each version of the model [*Tsyganenko*, 1995, 2002; *Tsyganenko et al.*, 2003]. Each of these import modules is accompanied by an introductory network similar to the one presented for the ReadENLIL module in section 4.1.

In addition to the import modules we have created a series of utility modules that accomplish various tasks. The ReadTrajectory and OrbitSync modules are designed to extract spacecraft information from simulations. The ReadTrajectory module imports spacecraft trajectories previously downloaded from the Satellite Situation Center (SSC) website and creates a DX Field with a UT vector in the data component at the location of the spacecraft in the position component. The Orbit-Sync module takes a UT time input along with the trajectory and creates a user customizable cloud of points around the current location of the spacecraft in the simulation domain. For simulations that require it, the Yr-DayHr module will convert a given simulation time into a UT vector which works with the other modules. The SPTransform module uses the cxform library to convert

vectors from one coordinate system to the other. The OctaveOp module allows data to be passed from CISM-DX into Octave scripts for further analysis and then returned to CISM-DX. The OpenClosed and MapFootPts are designed to work with groups of streamlines. The OpenClosed module classifies field lines as being closed if both endpoints line within a sphere of user defined radius. If only one endpoint lies within the sphere it is considered to be an open field line. The MapFootPts module is designed to map foot points from magnetospheric simulations down to ionospheric altitudes by assuming a constant dipole magnetic field. The use of these modules is demonstrated in a series of tutorial networks that come with the CISM-DX package.

In addition to the tutorial networks that provide instruction on how to use the various utility modules and beginning user interfaces for each of the modules for which we have import modules, the CISM-DX distribution includes several networks designed to assist the advanced user. The process of looping over multiple time-steps and creating images which can be made into digital movies is explained by several networks. A macro module which allows the user to define a camera path and look direction is also included along with several networks which demonstrate its use.

3.2. Octave Extensions

The CISM-DX package contains over 50 Octave scripts that can be used for common research and analysis tasks. There are eight categories of scripts that are listed in Table 2.

The Validation category includes functions that are used to compute metrics on the CISM Forecast model. The Forecasting category includes two empirical models - a model that predicts the daily–averaged Ap [*McPherron*, 1999] and a model that predicts the daily–averaged low–altitude MeV electron in the range L=1.1–10.0 [*Vassiliadis et al.*, 2002]. The code base of the Validation and Forecasting will be expanded as new results become available.

The Plotting functions in Table 2 refer to scripts that can plot each of the time series in the data base. In addition, plotting functions that were used for visual inspection and checking of individual time series are included. Other routines create plots to represent the output of the CISM Forecast Models and results from their validation.

The Analysis function does several types of averaging on data with flagged values and can re-sample or reformat time series onto a common time grid. The

Tutorials section includes examples of using the Octave codes to perform basic tasks such as extracting a time series from the data set in a given time range and running the Forecast models.

As discussed in the following section, the user of CISM-DX has access to a large set of time series data stored in a common file format. The Download and Parse categories in Table 2 refer to all of the code needed to re-create this data set. The Download functions include Octave scripts written to make calls to ftp or Internet sites and download each of the many individual data files (usually packaged to contain one day of data per file). The Parse functions extract and concatenate all of the individual files into one temporary Octave binary file per instrument.

The Download and Parse functions are distributed in the CISM-DX package for several reasons. First, they allow a user to see exactly what processing steps were taken to transform the original data into the final form. If a user wanted to make a change in the way the data were processed, he would simply need to modify the relevant section of the code. To facilitate validation efforts, the code used to compute metrics and skill scores is included. If a researcher wants to do a new analysis based on an expanded or different data set, the previous validation results can be exactly reproduced and built upon for the new analysis.

3.3. CISM-DX time series data set

A key component of the CISM-DX package is a uniform time series data set intended for basic research and model validation. Table 3 lists data sets from many sources that is easily accessible from programs included in the distribution. In order to provide transparency, reproducibility, error checking, and versioning, the CISM-DX software package also contains information on the exact steps a user would have to take, from downloading instructions to the code execution, in order to re–create the data set from the data in its original format.

Most of the data are from world data centers. One exception is a set of ACE real–time data obtained from the Space Environment Center FTP server, which is important for validation purposes. Although only a month of data is archived at this FTP site, we have collected original the raw, real–time data used to drive a real–time Dst model [*Temerin and Li*, 2002]. Without this archive, performance results from new real–time models could not be easily and directly compared with that of previous or currently–running models. To encourage proper usage of such an unusual data set, the meta-data includes information about where this data were
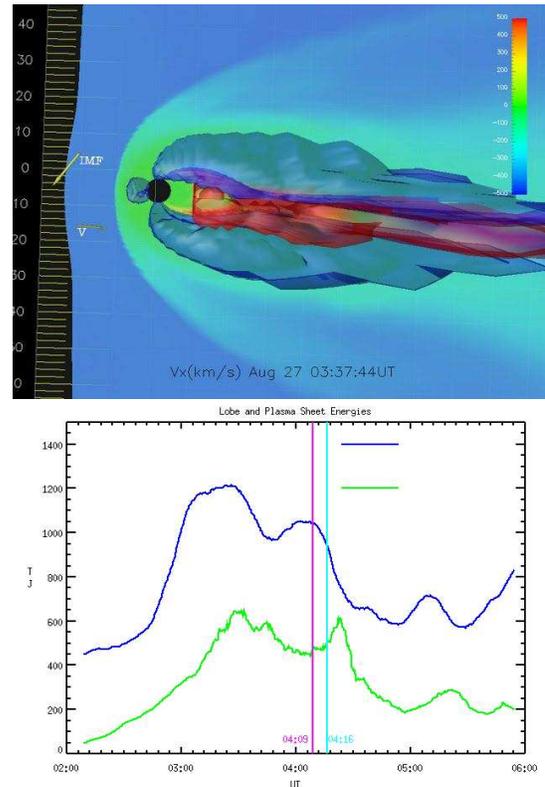
Figure 4. The top panel in this figure shows a snapshot of the LFM during the August 27, 2001 substorm simulation. The cut plane shows the plasma density in the XZ plane. The blue isosurface bounds a region which fits the criteria used to define the lobes, while the red isosurface bounds the region matching the conditions for the plasma sheet. In the bottom panel the time history of energy distribution in these regions during the substorm is displayed. The line at 0409 UT indicates the beginning of transfer of lobe magnetic energy into thermal and kinetic energy in the plasma sheet which precedes the CL substorm onset time of 0416 UT.

obtained and the caveat that it is not intended for scientific analysis.

The file format currently used is the Hierarchical Data Format. The user does not need to directly interact with these files, rather they can use the Octave codes discussed in Section 3.2 to call a given time series based on its name and time range of interest. Code with examples of loading and viewing all of the data are also included for IDL and OpenDX. The data set is an assembly of data drawn from many sources. The data have been reduced and put into HDF files so that if a researcher or student wants to do an analysis based on data from many sources they do not need to spend time learning and converting all of the different data file formats into that required by their plotting software.

The data set is versioned and static. If a data set is cleaned or filtered in any way, the result is given a new version number and the old data set is kept available for easy reference and comparison. The data are all on uniformly–spaced time grids with the same flag and fill values. Fill values were added so that all time series start on the first time interval of the year with the first available data point and end on the last time interval of the year with the last available data point. The data files contain extensive meta-data such as information about where and how the data were obtained, how to acknowledge the data source, and the Internet address or local file directory of any relevant README files. Only a few (approximately 100 MB in total size) of the most common data sets are installed with the CISM-DX package. Octave programs are included that automatically download parts of other uniform time series data sets (approximately 2 GB in total size) when the required data file is not found on the users local disk.

4. Visualization and Analysis with CISM-DX

The CISM-DX package comes with numerous networks which provide examples of how to access all of the model data included in the distribution, utilize the tools for coordinate transformation, run the empirical geophysical models, and conduct detailed analysis of the model output. Obviously we cannot cover all of these networks in great detail, so we have chosen to highlight three examples. Section 4.1 summarizes the class of networks which provide basic access to the model results by showing the results from a heliospheric model. We exemplify the more complicated analysis possible with this package by an energy budget analysis from a simu-

lation of a magnetospheric substorm in Section 4.2. In Section 4.3 we present the results from a network used to visualize the observations stored in the CISM-DX database.

4.1. Visualization of the Heliosphere

ENLIL is a 3D ideal MHD model [*Odstrcil et al.*, 2002] that is designed to model the state of the heliosphere. It typically covers a domain which extends from the solar equator to within 30 degrees of both poles. The inner boundary is placed near 20 R_S, which is past the critical point for transition from subsonic to supersonic flow in the solar wind. The outer boundary is typically placed near 1 AU, but can be extended further into the heliosphere. As an ideal MHD simulation the fundamental parameters it produces are density and temperature, velocity and magnetic field vectors. The model solves the ideal MHD equations in spherical coordinates and, hence, stores magnetic field and velocity as spherical variables. The model works on a regular grid which is periodic in the azimuthal direction.

Since the ENLIL model produces netCDF files which contain both the grid and a complete description of the state vector at a given time we created a customized import module, in the C programming language and using the OpenDX API, for placing this data into an OpenDX Field. The ReadENLIL module takes a netCDF file name as input and creates this OpenDX Field as output. The module first reconstructs the grid by converting the metric information stored in the file into Cartesian coordinates. It then reads in the MHD state vector converting the vector quantities from Spherical coordinates into Cartesian variables using information from the grid to determine the correct description of the unit vector at each location. At this time additional auxiliary variables like the field polarity and radius are also computed to facilitate later analysis.

The network shown in Figure 1 is taken from a example included in the distribution that was used as part of the 2004 CISM Space Weather Summer school. After executing the network the user sees the visualization shown in Figure 3 which is also available as an animation containing a fly-through of the heliosphere. The visualization shows the variation of speed along the ecliptic plane, as well as a cut through the heliosphere opposite the current location of the Earth, shown by the green sphere. In the ecliptic view the structure of multiple high speed streams is clearly evident while the cut through the heliosphere illustrates the increase in speed with heliographic latitude.

By means of the control panel shown in Figure 2 the
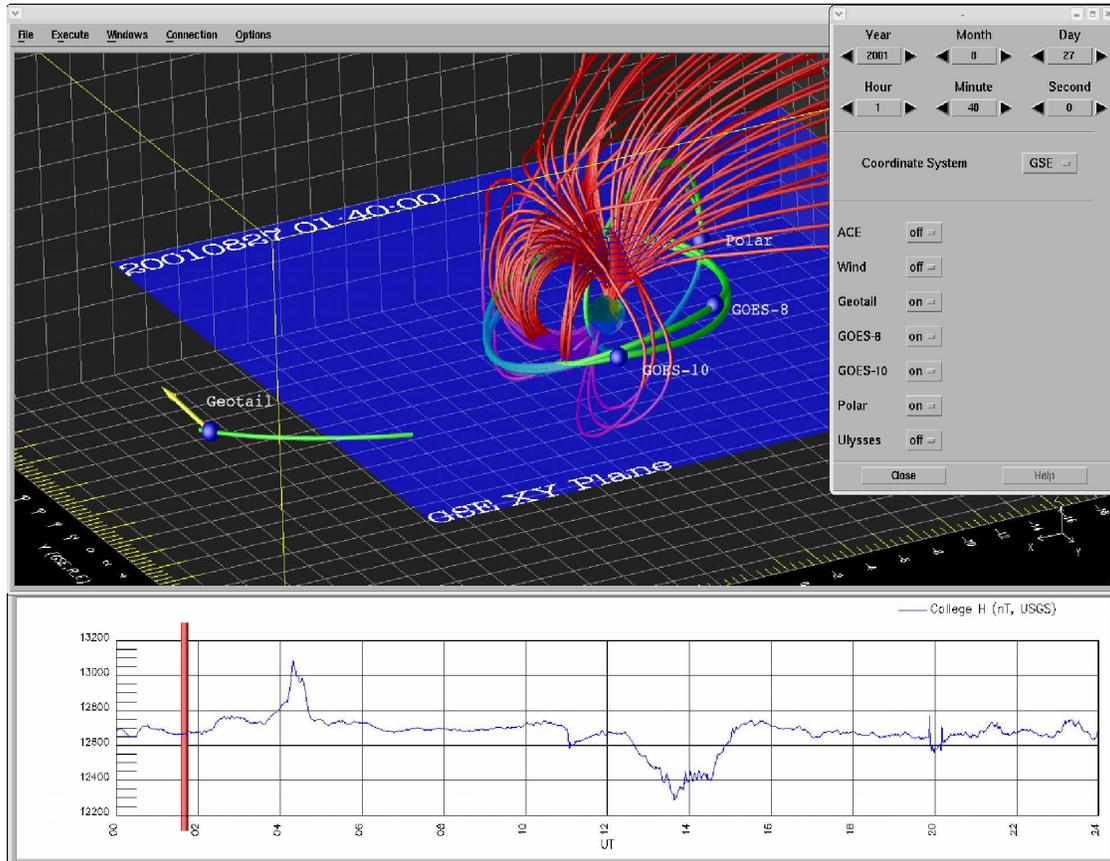
Figure 5. The top panel in this figure shows a snapshot of data and empirical model results. The red field lines are generated from the Tsyganenko magnetic field model using the Streamline module. Green tubes show the orbit segments of four near Earth satellites with their respective positions at 01:40 indicated with blue spheres. The magnetic field strength and direction measured by the GEOTAIL spacecraft is shown by the yellow arrow. An Octave function extracted the H component magnetic field measurement made by the College station from the uniform time series data set and it is plotted in the lower panel of this figure.

user can easily manipulate the visualization. The Read-ENLIL dialog box allows the user to change the source file for visualization. The user can either enter a new file name by typing directly into the box or click on the three ellipses and open a file browser to load the next file. The Plasma Parameter interface allows the user to change the variable plot on the cut planes by means of a drop down list interface. The available options include the MHD state vector as well as derived quantities like speed, density times radius squared, and log of the magnetic field magnitude. By changing the Azimuthal Angle and Ecliptic Level dialogs the user can change the location of the cuts through the computational domain. The dialog boxes allow for direct input or incremental changes which have been customized for the computational domain of the ENLIL simulation. The Line dialog boxes allow the user to change the location of the yellow line shown in Figure 3, which is then passed to a Map module and the Plasma Parameter is interpolated along this line and is plotted in another window not shown in this paper.

Upper level undergraduates and beginning graduate students were able to use this network to explore the heliosphere after an hour of instruction on OpenDX. This network is typical of the basic visualization networks constructed for each of the numerical models data sets included in the distribution. The dialog boxes are provided during the construction of the network in the VPE and no additional programming by the developer is required to monitor them during utilization.

4.2. Energy Distribution in the Magnetosphere

The Lyon-Fedder-Mobarry (LFM) global magnetospheric model uses the ideal MHD equations to simulate the interaction of the solar wind with the Earth's magnetic field [*Lyon et al.*, 2004]. It typically covers a computational domain which extends from approximately 30 R_E upstream to 300 R_E downstream with a radius of 100 R_E. As an ideal MHD simulation its output includes the standard MHD state vector. In addition it also supplies information about the currents and electric fields present within the simulation domain. This is particularly useful given that the LFM uses a regular non-orthogonal mesh to conduct its calculations and deriving these parameters can be computationally expensive. The simulation also includes a simple model for the Earth's ionosphere which is also included in the output files. The LFM uses HDF files for its output and typically includes multiple time steps in each file. We have created an import module, ReadLFM, which transfers this data into a set of OpenDX Field Objects

for the magnetospheric and ionospheric domains.

Wiltberger et al. [2004] used the LFM to simulate the magnetosphere during an isolated substorm which occurred on August 27, 2001. The simulation showed excellent agreement with geosynchronous and mid-tail spacecraft observations. The results indicate that the simulation accurately captured the energy loading, small scale activations, and global onset of the substorm. In particular, the simulation showed a series of complex discrete flow channels which disrupted the thin current sheet during the expansion phase of the substorm.

Figure 4 presents the results of an energy budget analysis of the LFM during this substorm interval. The top panel of Figure 4 is available as animation showing the temporal evolution of these regions during the growth, expansion, and recovery phases of the substorm. The first step in this process was to determine a set of criteria that could be used to define regions within the magnetosphere. The magnetospheric lobes are considered to be regions within the magnetosphere which have a thermal pressure less than 0.01 [nPa] and a density less then 0.2 [#/cc]. In order to define the plasma sheet the region tail-ward of 10 R_E was considered to be part of the plasma sheet if it had a plasma β less then 0.5 and a temperature greater than 1 [keV]. Once the criteria for defining the regions had been determined a complicated network was constructed, using existing CISM-DX components, to determine locations within the simulation which meet those conditions. The first step in this process uses the Compute module to calculate the needed plasma parameters within the simulation domain and then passes them into an Include module. The Include module marks each point according to its compliance with the applied condition, e.g. the density is less then 0.2 [#/cc]. The Compute module also supports logical operations enabling the application of multiple conditions by using a series of AND operators. After this process is complete we have a new OpenDX Field that has a new data member which states if this position is located within a given region. By feeding it to a isosurface module we can produce the visualization seen in the top panel of Figure 4.

Once the regions are defined, we move onto the next step in the process of analyzing the energy distribution within the magnetosphere during this substorm. A series of Compute modules are used to determine the thermal, kinetic, and magnetic energy densities at each location within the region. The connections component of the OpenDX field is used to determine the volume of each cell within the region. The entire amount of each type of energy present within the regions is then deter-

mined by integrating the energy density times volume over the entire domain. The bottom panel of Figure 4 shows the temporal behavior of the magnetic energy with the lobes and thermal and kinetic energy in the plasma sheet. The line at 0409 UT indicates when energy is beginning to be transfered from the magnetic field in the lobes into heating and moving the plasma within the plasma sheet. It is quite interesting to note that this activity precedes the auroral onset of the substorm as determined by the simulated AL index by seven minutes.

This energy distribution study is just one example of the advanced analysis of simulation data possible with the CISM-DX suite of tools. It exploits the key feature of OpenDX, all renderable objects are Fields. It is possible to use the combination of the Streamline and Map modules to calculate field line integrals. An isosurface of density in the ionosphere can be painted with the plasma temperature. The user willing to undertake the effort to create their own networks will be able to produce excellent graphics as well as mine their simulations for interesting results.

4.3. Satellite Data

In this section we give an example visualization of some of the time series data available during the isolated substorm event on August 27, 2001 described in the previous section. Figure 5 shows data that is easily viewable using Octave scripts and OpenDX networks distributed in CISM-DX. The measurement data were drawn from the time series data set described in Section 3.3 by calling an Octave function that from OpenDX. The Octave function takes inputs of the time series label and initial and final universal times and returns data on a 1–minute time grid.

Green tubes show the orbit segments of near–Earth satellites on this day, and a blue dot marks their position along this path at UT=01:40. The GEOTAIL spacecraft also has a yellow vector that shows the direction magnetic field vector it measured at that time. The lower panel shows the H component of the magnetic field as measured by the ground magnetometer instrument in Boulder, CO.

The magnetic field lines are generated with a field line tracing module available in OpenDX and the magnetic field was computed using a data–derived magnetic model [*Tsyganenko and Mukai*, 2003]. The coordinate transforms were performed using the OpenDX module SPTransform, which calls the cxform library.

5. Conclusions

In this paper we have presented a overview of the CISM-DX package for visualizing and analyzing space science model output and observational data. This package provides support for several levels of users. A beginning user can manipulate the OpenDX networks included in the distribution to rapidly examine results from numerous numerical and empirical models. They are also able to access an extensive data base of observations. The more advanced user can follow the examples provided to create their own visualizations and conduct sophisticated analysis of the models. An expert user can join the development community and extend the capabilities of the distribution. We have striven to provide users at all levels with a high level of integration between software, data, and analysis tools.

The entire package is being released to the community under an Open Source licensing agreement. While the core packages included in CISM-DX are available in MS-Windows, MacOS, and Unix version we are currently only providing distributions for Linux platforms. Interested users can download the package from http://lasp.colorado.edu/cism/CISM_DX. The current distribution draws upon the contributions of over 20 researchers. The success of this project depends upon continued community contributions. We look forward to meeting the needs of the space physics community through collaborative work to enhance the capabilities of CISM-DX.

References

Eaton, J. W., *GNU Octave Manual*, 1st ed., Network Theory Ltd., 2002.

Gombosi, T. I., D. L. DeZeeuw, C. P. T. Groth, K. G. Powell, and P. Song, The length of the magnetotail for northward IMF: Results from 3d MHD simulations, in *Phys. Space Plasmas*, edited by T. Chang and J. R. Jasperse, vol. 15, p. 121, AGU, 1998.

Hibbard, W., Building 3-d user interface components using a visualization library, *Computer Graphics*, *36*, 4–7, 2002.

Hibbard, W., B. Paul, D. Santek, C. Dyer, A. Battaiola, and M.-F. Voidrot-Martinez, Interactive visualization of earth and space science computations, *Computer*, *27*, 65–72, 1994.

Hilmer, R., *AF Geospace User's Manual Version 1.4 and Version 1.4P*, USAF, Hanscom, MA, 1999.

IBM, *IBM Visualization Data Explorer User's Guide*, 3rd ed., IBM, Yorktown Heights, NY USA, 1998.

Lyon, J. G., J. G. Fedder, and C. M. Mobarry, The Lyon-Fedder-Mobarry (LFM) global MHD magnetospheric simulation code, *J. Atm. Space Phys.*, p. in press, 2004.

McPherron, R. L., Predicting the Ap index from past behavior and solar wind velocity, *Phys. Chem. Earth (C)*, *24*, 45–56, 1999.

Mikic, Z., J. A. Linker, D. D. Schnack, R.Lionello, and A. Tarditi, Mangetohydrodynamic modeling of the global solar corona, *Phys. Plasmas*, p. 2217, 1999.

Murray, D., J. McWhirter, S. Wier, and S. Emmerson, The integrated data viewer: a web-enabled application for scientific analysis and visualization, *Preprints, 19th Intl Conf. on IIPS for Meteorology, Oceanography and Hydrology*, 2003.

Odstrcil, D. J., J. A. Linker, R. Lionello, Z. Mikic, P. Riley, and V. J. Pizzo, 3d MHD simulations of CMEs by coupled coronal and heliospheric models, in *Proc. 10th European Solar Physics Meeting*, edited by A. Wilson, pp. 95–98, ESA, 2002.

Raeder, J., Global Magnetohydrodynamics – A Tutorial, in *Space Plasma Simulation*, edited by J. Büchner, C. T. Dum, and M. Scholer, Springer Verlag, Berlin Heidelberg New York, 2003.

Raeder, J., J. Berchem, M. Ashour-Abdalla, L. A. Frank, W. R. Paterson, K. L. Ackerson, S. Kokubun, T. Yamamoto, and J. A. Slavin, Boundary layer formation in the magnetotail: Geotail observations and comparisons with a global MHD simulation, *Geophys. Res. Lett*, *24*, 951–954, 1997.

Roberts, D. A., M. Luckyanova, V. Rezapkin, and R. Boller, Insights into planar magnetic solar wind structures using new visualization methods, *AGU Fall Meeting Abstracts*, pp. C432+, 2004.

RSI, *Introduction to IDL*, RSI, Inc., 2003.

Temerin, M., and X. Li, A new model for the prediction of Dst on the basis of the solar wi nd, *Journal of Geophysical Research (Space Physics)*, *107*, 31–1, 2002.

Tsyganenko, N. A., Modeling the earth's magnetospheric magnetic field confined within a realistic magnetopause, *J. Geophys. Res.*, *100*, 5599, 1995.

Tsyganenko, N. A., A model of the near magnetosphere with a dawn-dusk asymmetry 1. mathematical structure, *J. Geophys. Res.*, *107*, doi:10.1029/2001JA000,219, 2002.

Tsyganenko, N. A., and T. Mukai, Tail plasma sheet models derived from Geotail particle data, *J. Geophys. Res.*, *108*, 23–1, 2003.

Tsyganenko, N. A., H. J. Singer, and J. C. Kasper, Storm-time distorion of the inner magnetosphere: How severe can it get, *J. Geophys. Res.*, *108*, doi:10.1029/2002JA009,808, 2003.

Unidata, *Integrated Data Veiwer: User's Guide*, 1 ed., University Cooperation for Atmospheric Research, Boudler, CO, 2004.

Vassiliadis, D., A. J. Klimas, S. Kanekal, D. Baker, and R. S. Weigel, Long-term solar-cycle, and seasonal response of magnetospheric energetic electrons to the solar wind speed, *J. Geophys. Res*, *107*, 10.1029/2001JA000,506, 2002.

Wang, W., T. L. Killeen, A. G. Burns, and R. G. Roble, A high-resolution, three dimensional, time dependent, nested grid model of the coupled thermosphere-ionosphere, *J. Atm. Solar Terr. Phys.*, *61*, 385–397, 1999.

Wiltberger, M., S. R. Elkington, T. Guild, D. N. Baker, and J. G. Lyon, Comparison of mhd simulations of isolated and storm time substorms, in *Proc. Chapman Conf. on Inner Magnetosphere Modeling*, p. submitted, AGU, 2004.

Works, T. M., *Using Matlab*, The Math Works, Inc., 2002.

M. Wiltberger, National Center for Atmospheric Research, High Altitude Observatory, 3450 Mitchel Lane, Boulder, CO 80301 (wiltbemj@ucar.edu)

R. S. Weigel and M. Gehmeyr, University of Colorado Laboratory for Atmospheric and Space Physics, 1234 Innovation Drive, Boulder, CO 80303-7814. (Robert.Weigel@lasp.colorado.edu; gehmeyr@lasp.colorado.edu)

T. G. Guild, Boston University, Center for Space Physics, 725 Commonwealth Avenue, Boston, MA 02215 (tguild@bu.edu)