UV-CDAT

The Ultrascale Visualization Climate Data Analysis Tools

Charles Doutriaux[1], Aashish Chaudary[2], Samuel Fries[1], Sankhesh Jhaveri[2], Dan Lipsa[2], Denis Nadeau[1], Jeffrey Painter[1], Jim McEnerney[1], Dean Williams[1]

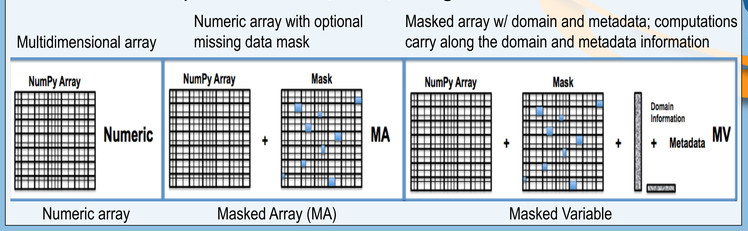
[1] AIMS Group, Lawrence Livermore National Laboratory
[2] Kitware, Inc

Introduction

UV-CDAT is the latest rendition of the Climate Data Analysis Tools, created by Dean Williams in 1996, it provides a user-friendly python API for retrieving, analyzing, and visualizing climate data in a wide variety of formats. It includes the giants of the Python ecosystem (SciPy, NumPy, Matplotlib) and bundles them together with the industry standard for interactive visualization, VTK, and leverages decades of climate software development to deliver the ultimate climate package along with a drag-and-drop GUI for direct manipulation and visualization of your data.

Data / IO

The Climate Data Management System (CDMS) is an object-oriented library for reading, writing, and analyzing climate data. It was first implemented as part of CDAT, and is now approaching its third major iteration. It allows users to seamlessly read data from multiple sources for intercomparison studies, and can read netCDF CF compliant data, HDF, GRIB, ASCII, binary, and other popular climate data file formats. It maintains an association between climate metadata and a NumPy array, providing highly performant operations that will still keep track of units, axes, and grids.

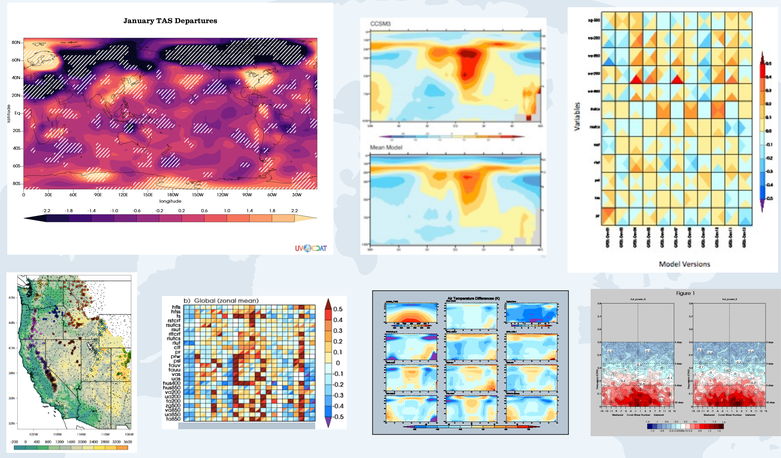


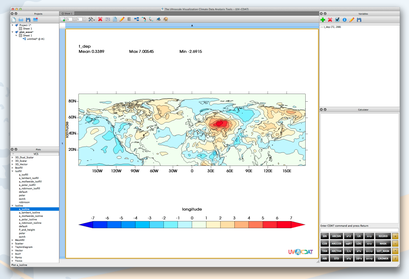
Analysis

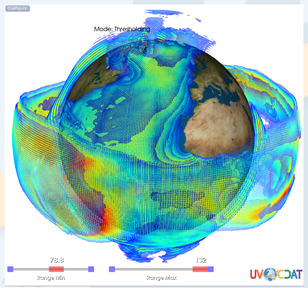
UV-CDAT is based on the numpy packages on top of which several packages to facilitate analysis of climate data have been implemented. The "General Utilities" package, `genutil`, contains a diverse array of useful functions; some common statistics functions, a grower function for expanding variables to match their axes, and some color matching functions. The "Climate Data Utilities" package, `cdutil`, contains utilities for time management, region extraction, and other tools specific to climate data. Every distribution of UV-CDAT includes SciPy which provides many useful tools that are well-documented. Also, UV-CDAT is capable of providing infrastructure for generating diagnostics for model runs; the DOE's Accelerated Climate Model for Energy (ACME) project and the Program for Climate Model Diagnosis and Intercomparison (PCMDI) both use UV-CDAT to generate a series of reference visualizations for visual analysis.

Visualization

The Visualization Control System (VCS) is custom built to meet the needs of climate scientists. It has a wide breadth of capabilities, which can be a useful tool for other scientific applications as well. VCS allows fine-grained modifications to be made to the appearance and content of every visualization, allowing users to create publication-quality graphics.

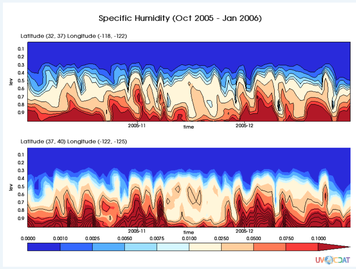






Capabilities and Highlights

- Friendly Python API
- Drag-and-Drop desktop application
- 1D, 2D and 3D plots (Boxfill, Isofill, Isoline, XvsY, Scatter, Meshfill, Hovmöller, Vector, and more)
- Interactive Visualization using VTK
- Supports Matplotlib
- 2D Patterns and Hatches
- Rendering in the background (Offline Rendering)
- View, select and modify attributes of data variables and their dimensions
- Customization using template attributes and graphics methods
- Export visualization in vector or bitmap format (PNG, PDF, EPS, SVG, JPEG)
- Grid transformations
- Custom color maps
- Regional zoom
- Portrait or landscape and partial or full-screen of a display
- Animate data interactively
- Support for different map projections
- Multiple interactive displays
- ESGF access
- Diagnostics
- Metrics



Driving Factors

Climate scientists have pressing questions to answer. UV-CDAT's purpose is to make that as easy as possible, and to provide all of the tools that a working scientist needs for both post-process and *in situ* analysis. It provides the whole toolchain, including ESGF data retrieval, SciPy and NumPy for analysis, MPI for HPC applications, and a host of others. Users shouldn't have to care about the fine details of programming, they should just have to express their solution in a structured way, and let the computers handle the rest. UV-CDAT aims to make that a reality. Most recently, this work has been driven by the need to supply DOE's Accelerated Climate Modeling for Energy (ACME) with a turnkey package that integrates workflow, provenance capture, analysis, visualization, automated testing, and evaluation capabilities.

Software Process

UV-CDAT is built using industry standard software process which includes sophisticated build system to build and package UV-CDAT dependencies and regression testing using CMake and CDash. UV-CDAT has a suite of automated tests, which are run by developers and by a homegrown continuous integration system. All tests are run under a variety of operating systems, versions, and build options for every change that developers make. Visualization output is compared to baseline images, and any significant deviation will trigger notifications to developers that the change is not up to the desired quality. This suite of tests is constantly expanding, with more than one hundred new tests added in the latest release. All development is done on GitHub, as UV-CDAT is an open-source project.

Future Directions

- New diagnostics
- Web Interface – CDATWeb
- Redesigned GUI and user experience
- Additional support for interactive and exploratory visualization
- New plot types
- Enhanced Parallel Processing Capabilities
- Improved documentation with user contributions
- Test coverage measurement
- More regression tests and performance tests\
- Python 3.x
- Windows support

This work is supported by the ESGF consortium: DOE, NASA, NOAA, IS-ENES, NCI. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DEAC52-07NA27344.