The Visualization Framework

SimEnvVis

Documentation V1.00
August 2013

Thomas Nocke, Magnus Heitzler
Potsdam Institute for Climate Impact Research

SimEnvVis 1.00 - Documentation (August 2013)

User documentation

Installation and environment

Programmer documentation

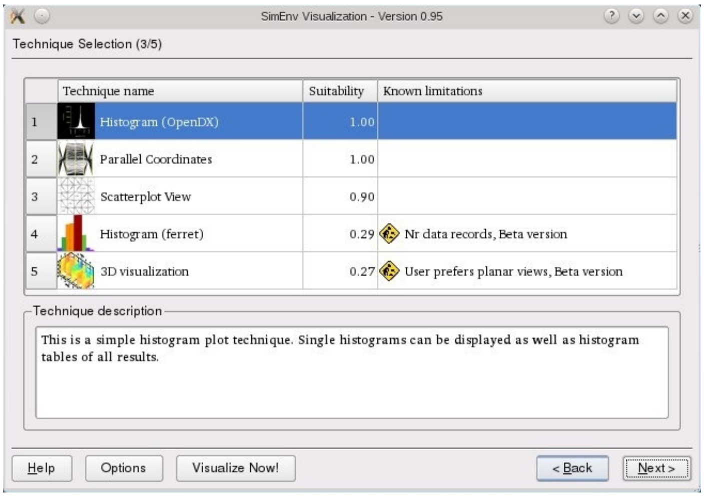
For questions and hints contact Thomas Nocke, nocke@pik-potsdam.de

SimEnvVis - General Information

Introduction

Researchers are familiar with certain visualization environments and techniques, and typically focus on a limited set of tasks (e.g. find trends, present/communicate the results). To reduce the obstacles to use new visualization systems, there must be a strong advantage doing so - in contrast to existing visualization tools such as Vis5D - especially providing sophisticated visualization techniques and easy-to-use interfaces to them. Therefore, the SimEnvVis framework has been developed, providing a variety of the presented techniques to climate modelers.

Besides these technical issues, the manifold of integrated visualization techniques - partly not known to climate modelers - requires a substantial support selecting and parameterizing these techniques. Our approach is to provide a visualization design wizard, which hides internal system details and enables modelers to concentrate on the analysis process itself. Figure 1 depicts part of the wizard, showing available visualization techniques ordered by an automatically calculated suitability.



Technique Selection (3/5)

	Technique name	Suitability	Known limitations
1	Histogram (OpenDX)	1.00	
2	Parallel Coordinates	1.00	
3	Scatterplot View	0.90	
4	Histogram (ferret)	0.29	Nr data records, Beta version
5	3D visualization	0.27	User prefers planar views, Beta version

Technique description

This is a simple histogram plot technique. Single histograms can be displayed as well as histogram tables of all results.

Fig 1: Selection of a visualization technique

Therefore, a two-step rule-based mechanism has been developed, which generates images based on metadata, user goals and profiles (see Nocke and Schumann 2002, Nocke and Schumann 2004, Lange, Nocke, and Schumann 2006, Nocke 2007). In a first step, the general suitability of a visualization technique is estimated based on important metadata only (e.g. no. of variables and data records). In a second step, these techniques are parameterized based on rules. This includes the selection of a visual mapping strategy (e.g. isolines or/and coloring), the selection of a mapping of certain variables to certain visual attributes (e.g. temperature to colors) and the fine-tuning of the parameters (e.g. selection of a color map).

To further increase the system acceptance to new users, parameter dialogs can be simplified on demand, focusing on main parameters only, using the others in default mode. Thus, learning phase can be simplified, still allowing full visualization control in later stages.

Concept

With SimEnvVis postprocessed experiment output can be visualized problem-oriented. Dependent on metadata suited visualization modules are selected and parametrised.
Visualization techniques are combined to modules for the reperesentation of similar data characteristica. For example, this isosurfaces and 3D sectional views are combined to a 3D visualization module. Modules are implemented in OpenDX, Ferret, OpenGL, Qt and/or C++. Subtechniques and their parametrizations are addressed by parametrization dialogues, enabling interactively suited visualizations dependent on the visulaization goals. This approach implies that the user can switch between visualization techniques easily.
Fig. 2 shows the general SimEnvVis approach: By means of an xml description of the visualization modules and of metadata in NetCDF a set of suited visualization modules are selected semi-automatically for a data set. Afterwards, suited parametrizations are determined based on these metadata and analysis goals. These parametrizations are used as a starting point for the visual analysis process to get a first problem-oriented visual representation of the data.

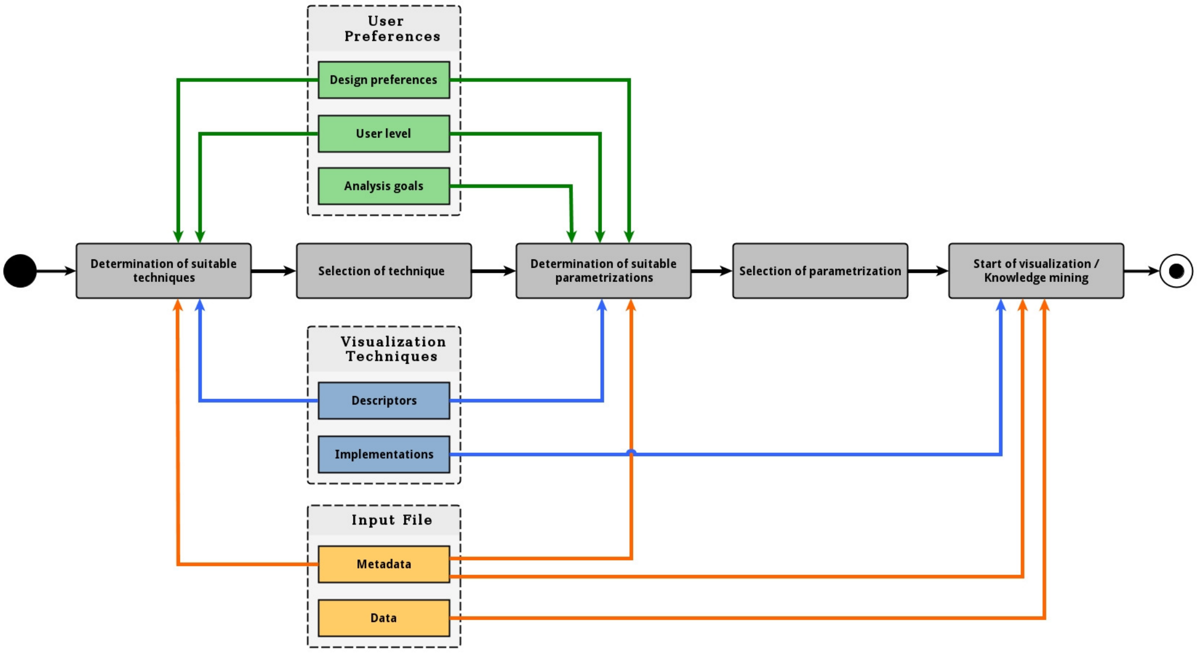


Fig 2: Basic structure of the visualisation design

Processing Pipeline

Based on this approach the following flowchart (see Fig. 3) for the description of the data and control flows was implemented for SimEnvVis:

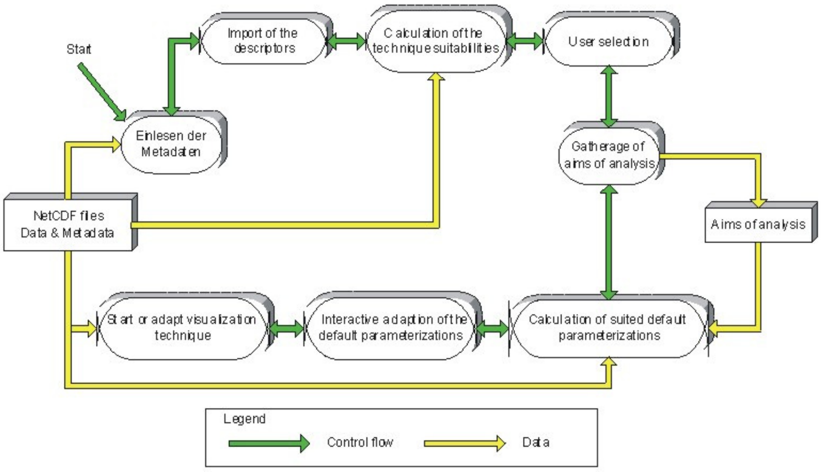

Start
Import of the descriptors
Calculation of the technique suitabilities
User selection
Einlesen der Metadaten
Gatherage of aims of analysis
NetCDF files Data & Metadata
Aims of analysis
Start or adapt visualization technique
Interactive adaption of the default parameterizations
Calculation of suited default parameterizations
Legend
Control flow
Data

Fig 3: SimEnvVis flowchart

Successively, the following processes are executed in the pipeline:

1. Import of the metadata
2. Import of the visualisation module descriptors
3. Calculation of the suitability of the modules
4. Interactive selection of a suitable module
5. Gathering of the aims of the analysis (not included in the current version)
6. Calculation of suited default and initial parametrizations
7. Interactive adaption of these parametrizations
8. Start and or adapt of complex visualization modules (e.g., OpenDX maps)

It can be returned to previous steps in this pipeline without closing already started visualizations.

Program Structure

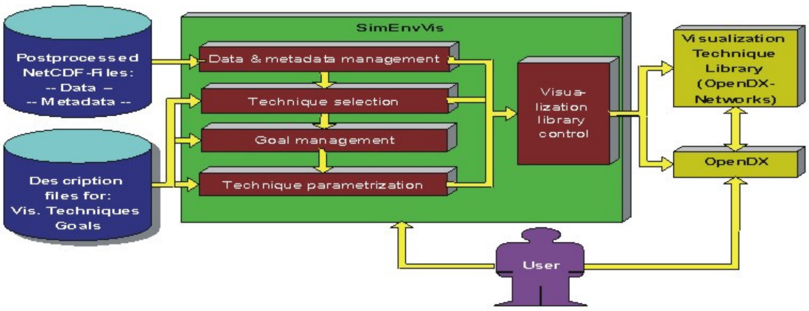

Postprocessed NetCDF-Files: -- Data -- -- Metadata --
Description files for: Vis. Techniques Goals
SimEnvVis
Data & metadata management
Technique selection
Goal management
Technique parametrization
Visua-lization library control
Visualization Technique Library (OpenDX-Networks)
OpenDX
User

Fig. 3: SimEnvVis program structure

API

The extension of software applications remains a mandatory property in order to adapt to new requirements or to benefit from recent/innovative software developments. For this purpose, SimEnvVis provides an object-oriented API to connect new visualization systems. This is mainly realized through the implementaion of an abstract C++-class ’’VisualizationTechniqueTemplate‘‘ and the description of the corresponding visualization systems and parameterizations through an XML-file.

The VisualizationTechniqueTemplate class provides the interface to connect new visualization systems to SimEnvVis. Implementing classes have to provide at least four methods which will be called by the SimEnvVis main application.

The method **setParameters** contains parameters which are needed to parameterize a template. This includes the name of the template to be used and the values corresponding to the chosen parameterization as well as the chosen datafile and the variables that should be visualized. The implementing class should therefore save these information and prepare the visualization accordingly.

Next, the method **adaptNCFile** is called and holds a MetaData-object (for detailed information see sec. xy). This should be used by the implementing class to adjust the internal structure of the NetCDF-file. Here, for example in the case of Avizo, CDO commands are performed to adjust the time axis and to transform longitudes from [0, 360] to [-180, 180] to ensure that the data can be visualized without difficulties.

Afterwards, when the template is parameterized and the NetCDF-file is adapted, SimEnvVis will call the method **execute**. This should launch the visualization software with the given parameterized template and NetCDF file, either by constructing a new window as a child of the SimEnvVis main window or by starting a child process through a system call.

Finally, the method **terminate** should be implemented which is called when the user shuts down the visualization. Here, functionality to clean up temporary files or saving the session state should be implemented.

Terms

For the SimEnvVis framework, the following terminology is used:

<table>
<tr><th>Term</th><th>Explanation</th></tr>
<tr><td>Independent variable</td><td>An independent variable holds values in dependency to which the values of the dependent variables vary. Typically in the context of the NetCDF-CF Convention, independent variables hold temporal (time) or spatial (longitude, latitude, level) information. In the context of the multi-run simulation environment SimEnv, independent variables can also hold other information, e.g. on run numbers or model input factors (parameters). When combining n independent variables they span up an n-dimensional space with a rectilinear grid structure. Equidistant values of an independent variable result in a regular grid structure, otherwise a block structure is generated. Combinations of both are possible. Independent variables must neither contain _FillValues nor missing_values (in NetCDF-CF terminology).
For the following minimalistic example, assume two independent variables longitude and latitude. Longitude holds the values 5.0°, 5.1°, 5.2° and latitude holds the values 49.4°, 49.5°, 49.6°. The two variables span up a two-dimensional (geographic) gridded space:
<table><tr><th></th><th>5.0°</th><th>5.1°</th><th>5.2°</th></tr><tr><td><b>49.4°</b></td><td></td><td></td><td></td></tr><tr><td><b>49.5°</b></td><td></td><td></td><td></td></tr><tr><td><b>49.6°</b></td><td></td><td></td><td></td></tr></table>
<b>Synonyms:</b> <i>Dimension</i> (especially in the context of NetCDF), <i>coordinate</i> (especially in the context of SimEnv).</td></tr>
</table>

Dependent variable

A dependent variable holds values which are generated in dependency to the values of independent variables. They are created as dependent model output and/or as post-processed results (e.g. precipitation). The number of values of a dependent variable equals the product of the values of the corresponding independent variables. For a dependent variable, a value has to be present for each grid cell. In other words, a dependent variable will hold a value for each combination of the independent variables it depends on. For the above example, a dependent variable precipitation may be modeled on the 2-dimensional grid spanned up from the independent variables longitude and latitude:

	5.0°	**5.1°**	**5.2°**
49.4°	20mm	21mm	19mm
49.5°	19mm	21mm	21mm
49.6°	17mm	22mm	21mm

Synonyms: *Variable*, *result* (especially in the context of SimEnv).

Subset

A subset is a set of dependent variables defined on the same grid (same dimensions, grid resolution, grid type and grid values).
Commonly, NetCDF files have more than one subset. For the above example, precipitation is a representative of the 2D atmospheric latitude-longitude subset while temperature can be a representative of a 3D latitude-longitude-level subset. (e.g. 3D atmospheric temperature and 2D surface temperature fall into seperate subsets).
See Programmers Reference for Metadata for an example.

Data record

For a given subset, a data record consists of two sets of values. First, a combination of the values of the independent variables. Second, the values of the dependent variables for this combination.
For the above example, in addition to the dependent variable precipitation, relative humidity could be another dependent variable. Thus, the data record for the values 5.0° and 49.4° consists also of the value 20mm for precipitation and the corresponding value of relative humidity at this location (not shown in the table).
Synonyms: *Feature* (especially in the context of Geographic Information Systems (GIS))

Attribute

Attributes are a term used for NetCDF-files which provide additional information for either a variable (independent or dependent) or for the whole file. The latter is also called a global attribute. For example, global attributes hold information about the title or history of the file. The units or value_range attribute can provide individual information for variables.

Visualization technique

A visualization technique is any visualization software which can be started and parameterized by SimEnvVis. For example, SimEnvVis parameterizes a pre-written script and uses this script as an input for the visualization software. For a given visualization software, there might be multiple scripts present. Each of these scripts will be handled by SimEnvVis as a separate visualization technique, even though they might use the very same underlying visualization software. It should however kept in mind that SimEnvVis also offers internal visualization techniques, which are tightly coupled with the main program of SimEnvVis. The parameterization of these techniques is being done internally without the use of a start-up script.
Synonyms: Sometimes a visualization technique is also called a *visualization module*.

Technique variation

For each visualization technique a set of variations is offered to the user. These define how the visualization technique will be parameterized. This is either done by replacing keywords in a script or it is handled internally.
Synonyms: Sometimes a variation is also called a *mode*.

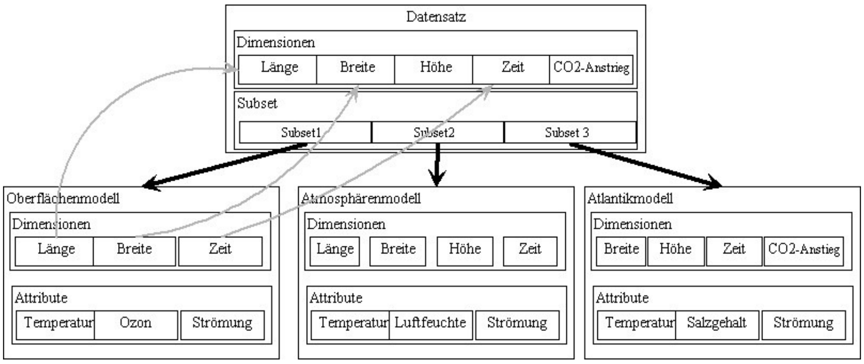
back to main page

SimEnvVis programmer documentation - Metadata

General information on Metadata

See Nocke, Schumann: Meta Data for Visual Data Mining, Proceedings Computer Graphics and Imaging, CGIM’02, Kauai, USA, August 2002.

Subset structure



For further explanation see subset definition.

back to main page

SimEnvVis - The Windows

General Information

SimEnvVis is a Wizard, that helps you to generate suited start images of your data.
On the pages, the user can use the following buttons:

- **Help Button:** launches helping pages for further explanation of the current page
- **"Visualize Now!"-Button:** directly generates a visualization based on default parameters in dependency of the current settings
- **Next Button:** to refine the visualization parameters to generate visualizations for the users context/goals (semi-automatically)
- **Back Button:** change former settings to generate another visualization
- **Histogram:** display simple histogram charts of all results in the selected data file
- **Options:** starts a dialog enabling users to specify the settings

In the following, the single pages of the wizard are described.

Window: Select Input File for Visualization

The NetCDF file to be processed and visualized must be selected.
This can be either a SimEnv-type NetCDF file or a Climate Forecast (CF) conform NetCDF file.

Window: Select Data for Visualization

This window enables users to select data subsets and model output results. Furthermore, it displays important metadata of the imported data set.
The metadata for each categegory (Dependent Variables, Independent Variables, Global Metadata, Spatial and Dataset Metadata) can be inspected by selecting the corresponging tab.
Then, in the main area below these tabs the detailed metadata for the selected category is displayed.
In the upper left combobox the subset of interest can be selected.
The data of interest can be further refined by selecting "Dependent Variables" of interest by checking the check boxes in the very right column "Select".
The buttons "Select All" and "Deselect All" can be used to ease the process of selecting dependent variables of interest when a lot of dependent variables are present.
Only the selected subset and selected variables will be used for visualization!
The metadata categories are the following:

Metadata for Dependent Variables

Parameter	Meaning
Standard Name	Name of the variable of interest. The image depicts the subset the variable is part of.
nD	Number of dimensions the grid of the subset consists of.
Long Name	A longer name with additional information on the variable.
Units	Units of the variable. If no units is given in the NetCDF file, SimEnvVis tries to estimate the units by its own.
Value Range	The expected value range of the variable. This is either taken from the attributes "data_range" or "valid_range" or - if not present - SimEnvVis calculates these values itself.
Select	Checkbox to define the variables of interest. Not truly part of the metadata.

Metadata for Independent Variables

Parameter	Meaning
Name	Name of the dimension (independent variable). The image depicts the subset the variable is part of.
Description	Short description on the dimension.
Units	Units of the variable. If no units is given in the NetCDF file, SimEnvVis tries to estimate the units by its own.
Value Range	The expected value range of the variable. This is either taken from the attributes "data_range" or "valid_range" or - if not present - SimEnvVis calculates these values itself.
Nr of Values	Number of cells along the dimension.

Global Metadata

This tab consists of a table with two columns.
It consists of the attribute names and contents of the global attributes present in the NetCDF file.

Spatial and Dataset Metadata

Dimensionality	**Meaning**
Dimensionality	Describes the dimensionality of the current subset. This includes all the number of all spatial coordinates plus a time time coordinate (if available).
Data representation	This metadata describes the influence of a data point to the data space (defined by the coordinates). If the nature of range is "global" (global influence) or "local" (local influence), the data values can be interpolated. If the nature of range is punctual, the influence of a mesurement or simulation is valid only at the distict location. Then no interpolation will be allowed.
Grid Type	The type of grid. Types are "regular", "block structured" and "irregular".
Cell type	Are the given data locations (defined by the coordinate tuples) located in the middle of the cells ("Data in cell interior") or at the vertices of the grid ("Data on grid vertices" - default)?
Maximum Nr of Data Values per Result	Number of Cells defined by the grid (product of all number of cells per dimension).

Window: Technique Selection

This window contains a table, displaying the available visualization modules (first column). For each technique a suitability value based on the data sets metadata of the imported NetCDF file has been calculated (second column). The techniques are ordered by their suitability value.
In the 3rd column the problems in the application of this module for the data set are displayed. Examples are:

- Nr of datasets: The number of data points in space/time not appropriate (too high or too low).
- Dimensionility: The technique can not display the spatial dimensionality of the data set. Example: A 1D data set (for instance time series) can not be diplayed with the 3D module.
- ...

Double clicking on a row launches the help for the respective module.
For help for the specific techniques see the visualization modules.

Window: Task Specification

At this wizard page the user can select the task to be fulfilled with the visualization. This helps in the later windows to fine tune the visualized image. Therefore, predefined (complex) tasks are provided in a combobox. For more details see the tasks page.

Window: Module Parametrization

In this window a modus of the current module can be selected (in a table). Again, the modules are ordered by suitability. Also, according to the task design space information on the task components is given and how they contribute to the overall suitability.
Visualization windows can be launched by applying the "Visualize" button.

Note:
Sometimes OpenDX-Visualization windows can crash. If both the "Close visualization" button and the application termination do not close the visualization, all the opendx-windows can be terminated usining the "killall dxui" in a shell.

Window: Active Vis

This window pops up when the first visualization is launched. It lists all active visualizations and allows to stop them by selecting the visualization in the list and afterwards applying the "Close Visualization" button.

---

back to main page

SimEnvVis programmer documentation - Analysis goals

SimEnvVis follows a novel approach for task definition as described in Schulz, H.-J. Nocke, T.; Heitzler, M.; Schumann, H.: A Design Space of Visualization Tasks, at IEEE InfoVis 2013. Each task is a point in a 5 dimensional design space, which are:

- **Goal**: What is the intent of the analysis?
  Examples: Exploratory/Confirmatory analysis, Presentation
- **Means**: How should the visualization task be carried out?
  Examples: extraction, abstraction, gathering
- **Characteristics**: What kind of facets of the data should be revealed?
  Examples: outlieres, clusters, distributions
- **Target**: On which part of the data should the visual analysis focus on?
  Examples: attributes(input), attributes(time), attributes(space)
- **Cardinality**: How many instances of the chosen target should be considered?
  Examples: all, multiple, one

Based on these categories, task components can be specified. These can be additionally placed in one of the following groups:

- neutral
- area specific or
- data specific.

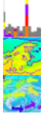
An **xml file format** for **storing elementary tasks** has been defined (see file data/goals.xml).
The task components of one category are combined by logic OR. Using logic AND all these combined tasks of the categories are combined to a complex task. The user can assign a name for a complex goal and can save them to or restore them from the file system.
Example: Complex goal "Analyse outliers" := global AND outlier AND (associate OR classify OR compare OR rank) AND (details_on_demand OR extract OR filter OR overview)
A further **file format to store** these **complex goals** has been defined (see file data/storedgoals.xml).
A dialog, to display, select and store complex goals based on elemantary goals has been implemented. This dialog includes hiding mechanisms for certain kind of elementary goals and adding, changing, deleting, renaming and storing buttons/functions for complex goals.
For more information see Nocke, Schumann: Goals of Analysis for Visualization and Visual Data Mining Tasks, Prague CODATA Workshop on Information Visualization, 2004.

---

back to main page

SimEnvVis - The Visualization Modules

Index

Preview image	Visualization module name	Visualization system / environment
	1D Visualization	OpenDX
	1D Visualization	Ferret
	Stacked bar charts	Ferret
	2D Scalar Visualization	OpenDX
	2D Flow Visualization	OpenDX
	3D Visualization	OpenDX
	Histogram Visualization	OpenDX
	Histogram Visualization	Ferret
	Parallel Coordinates	C++, OpenGL, QT
	Scatterplot Matrix	C++, OpenGL, QT
	Graphical Table	C++, QT
	Icon-based Visualization	C++, OpenGL, QT
	2D/3D Scalar Visualization	ParaView
	2D Scalar Visualization	Avizo
	3D Scalar Visualization	Avizo
	2D Vector Visualization	Avizo

General hints for all OpenDX modules

- Sometimes in a **OpenDX visualization window axes or axes labels do not fit** entirely into the window using the opendx default "Reset view" functionality. To get suited screenshots the user needs to use the "zoom" and "pan" functions in each module main parameter window.

- Sometimes the application of the sequencer (to animate) may lead to a **halt of OpenDX** (caused by the DXLink mechanism). In this case, close the visualization in the “Active Visualization” window and restart it using the visualization button.

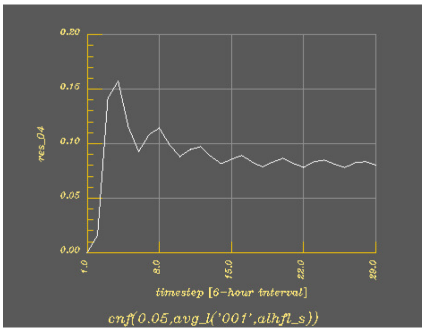
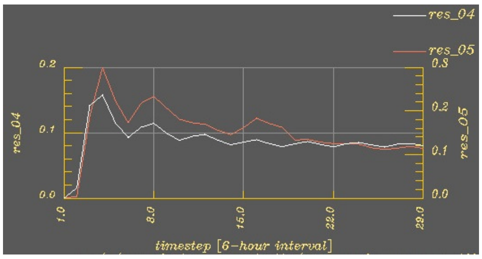
back to main page

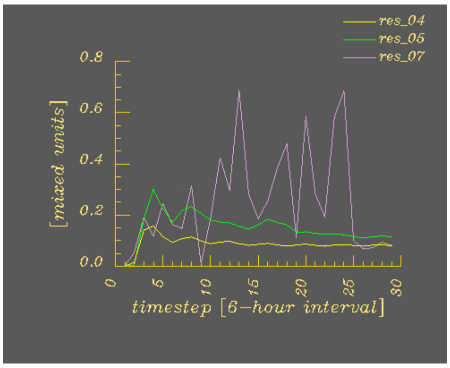
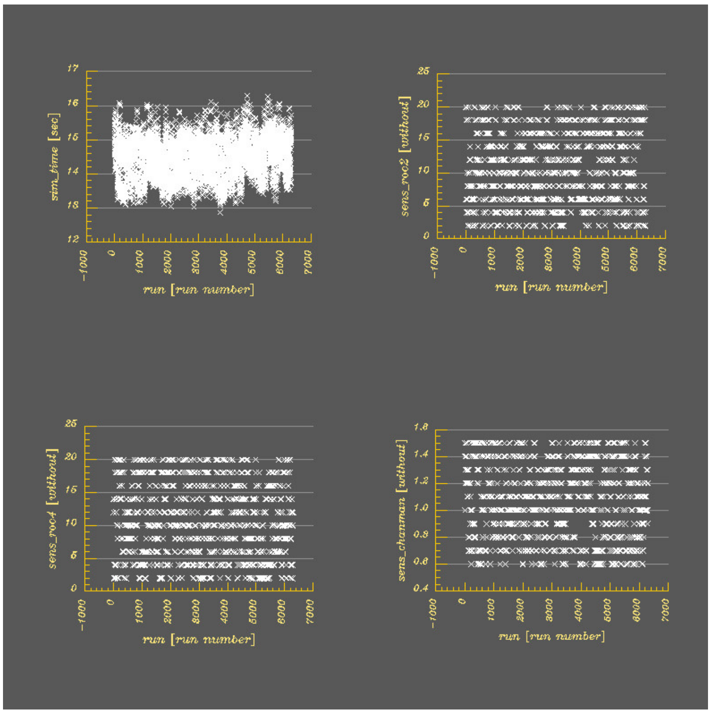
SimEnvVis - 1D Visualization Module

Technique description

This is a standard function plot visualization for one, two and more parameters. It allows to draw lines (for continuous data), dots and little crosses (for discrete data). Four modes allow to analyze single plots or to compare two or more plots.

Technique modes

Mode	Example Image	Description	Available Parameters
Single mode		The single mode includes the interactive visualization of one certain plot. In contrast to the table mode, the user can zoom into certain parts of the selected plot. Furthermore, in single mode two parameters can be compared in one window.	Result, Line style,Fill, Interactor, Reset Image
Double mode		The double mode displays two graphs in a plot with two separate axes. The user can zoom into certain parts of the selected plot. It is especially suitable for two graphs with different units or strongly varying data ranges (to compare similar trends on different scales), whereas mapping results of similar, but not equal data ranges may lead to misunderstanding.	Result (1&2), Line style, Interactor, Reset Image

Mode	Example Image	Description	Available Parameters
Multi-line mode		The multi line mode visualizes all the selected results in one plot. The user can zoom into certain parts of the selected plot. Units are lost.	Line Difference, Interactor, Reset Image
Table mode		The table mode allows to analyze a certain number of attribute plots concurrently. The number of columns and the size of the plots can be adapted.	Nr Columns, Line style, Fill

General technique parameters

Select dialog - Select a certain parameter dialog (in analogy to a tabbed dialog). Available dialogs are: General options, Save image options, Single mode options, Double mode options, Multi-line mode options, Table mode options.

Render mode – To switch between the four modes (Single mode, Double mode, Multi-line mode, Table mode)

Data normalized based on – To change data value normalization. Available options: Metadata min-max (determines lower and upper bound of plot based on min and max of result values from netcdf), Data min-max (determines lower and upper bound of plot based on calculated min-max values, if netcdf-values do not exist or are inconsistent with the data).

y-axis type - Specifies if the y-axis (the result axis) will be displayed normally (linear), or on a logarithmic scale. Be sure that you use the logarithmic value only if either all values are positive or all values are negative. Else the data will not be transformed correctly.

...

Mode specific parameters

Result (1 or 2) – Select one or two results (dependent variables) to be displayed (singe or double mode only).

Line style – How to map the data values (single, double and table mode only): X – display little crosses; dot – display small dots; line – draw a lines between the mapped data value locations. X and dot are suited for discrete or qualitative scale types and line is suited for continuous scale types .

Line difference – How to display the difference between the lines (multi-line mode only): none – display all in the same way, color – represent differences using color, glyphs - represent differences using small glyphs at the lines, color & glyphs - represent differences using color and small line glyphs.

Fill – Allows to fill the area under the graph (single and table mode only).

Interactor – Changes plot location (pan) and plot size (zoom). Click left mouse button and keep it pressed. Move plot where you want it using pan (to avoid overlapping with labels) and move up or down to zoom out resp. zoom in.

Reset Image – Sets the view back to default plot location and plot size (single, double and multi-line mode).

Nr Columns – changes the number of columns in the table mode display.

Technique restrictions

The display of min-max labeling (in contrast to OpenDX-Default) is implemented for single and double mode only.

<< previous | module overview | next>>

back to main page

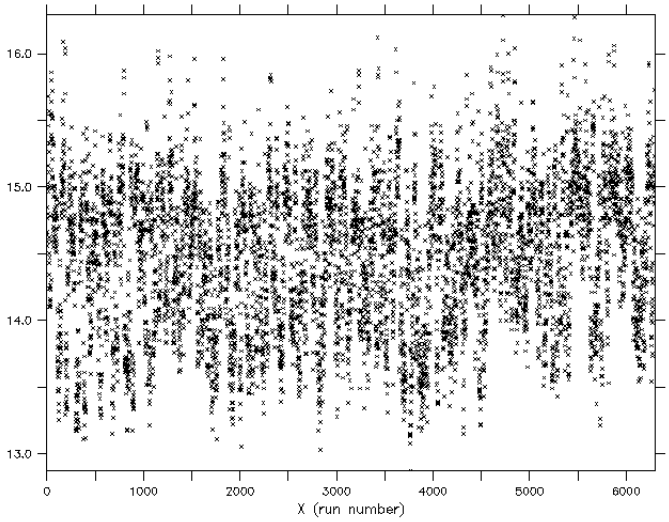
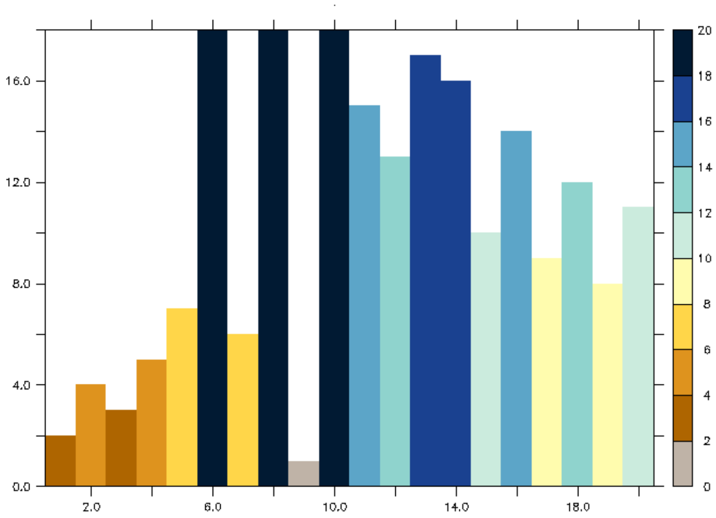
SimEnvVis - 1D Visualization (Ferret)

Technique description

This is a ferret visualization for one parameters other time. It displays the independent variable on the x-axis (e.g. time) and the dependent variable on the y-axis (e.g. temperature).

Furthermore, in Expert mode, the user can change the visualization by manual adaption of the script and save adapted scripts (with the currently selected parameters) for later use outside of SimEnvVis.

Technique modes

Mode	Example Image	Description
X glyphs		This is a ferret x glyph plot.
Bar charts		This is a temporal bar chart in ferret.

General technique parameters

Selected result – Select a certain result (variable) to display

Mapping mode – To switch between the available technique modes.

Colormap 1 and 2 – Switch between different ferret specific color maps (one for each mode).

Technique restrictions

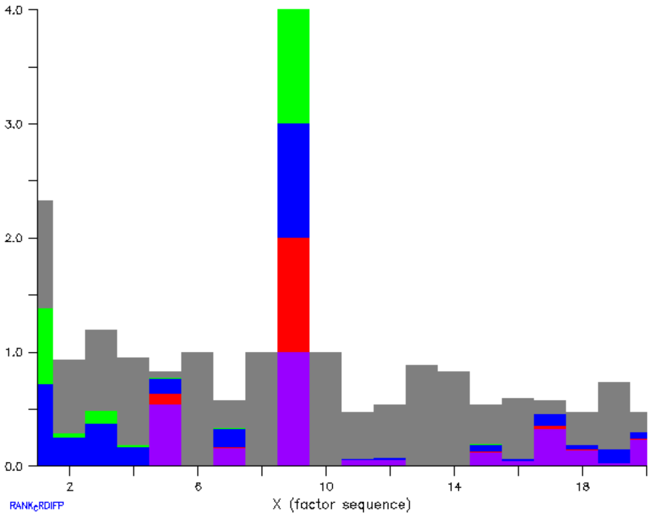
Currently, an extra mode has been added to show the color bar (mode 3). However in this mode labeling does not work correctly any more. On the other hand, the two bars near to variable minimum and maximum are drawn in half thickness of the other bars only (in mode 2). This avoids misunderstanding of ferret default bars that are drawn beyound the minimum and maximum values (see mode 3).

<< previous | module overview | next>>

back to main page

<< previous | module overview | next>>

SimEnvVis - Stacked bar charts (Ferret)



Technique description

This is a stacked bar chart visualization for up to 5 results in 1D. The user can select up to 5 results/variables, change their order and color.

Furthermore, in Expert mode, the user can change the visualization by manual adaption of the script and save adapted scripts (with the currently selected parameters) for later use outside of SimEnvVis.

General technique parameters

Result 1 - 5 – Select the single results (variables) and/or order them.

Colormap 1 - 5 – Map a certain color to a certain variable.

Technique restrictions

Currently, the two bars near to variable minimum and maximum are drawn in half thickness of the other bars only. This avoids misunderstanding of ferret default bars that are drawn beyound the minimum and maximum values.

Result/Variable labels underneath the bar chart are not yet drawn in correct color.

<< previous | module overview | next>>

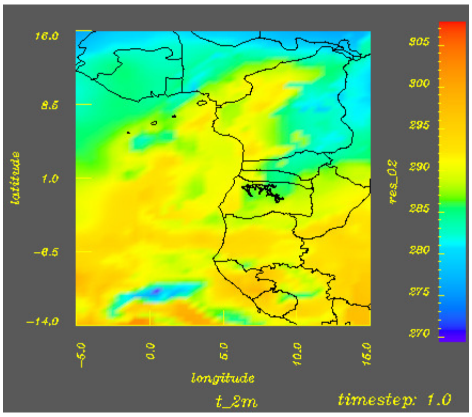
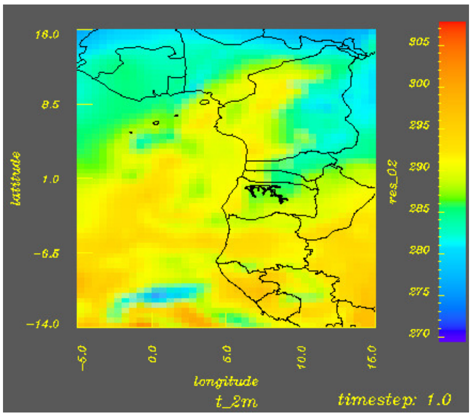
back to main page

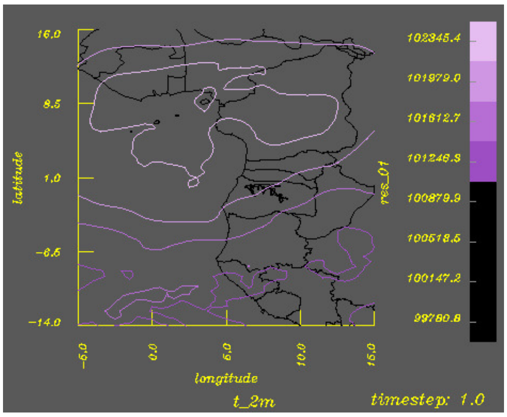
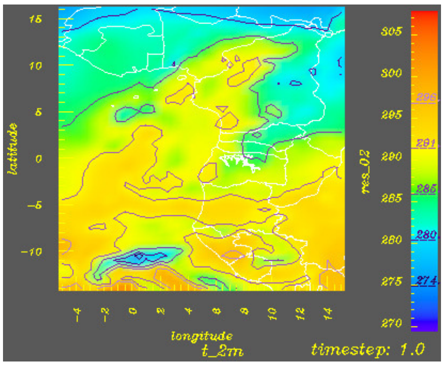
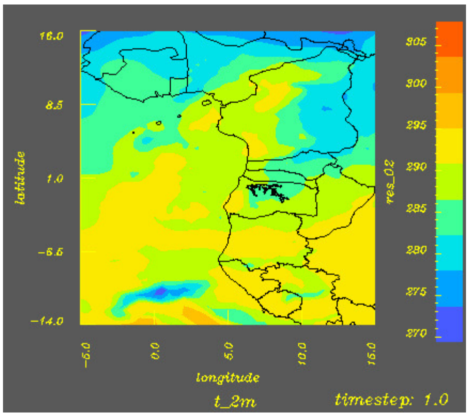
SimEnvVis - 2D Visualization Module

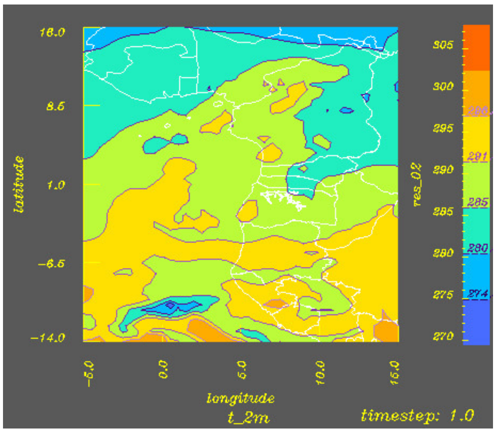
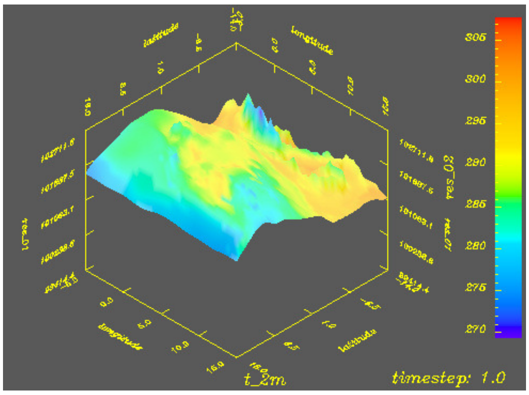
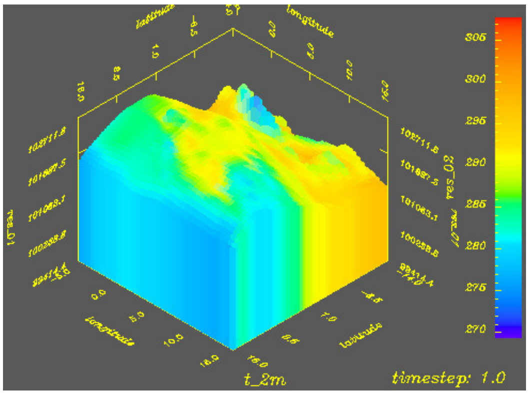
Technique description

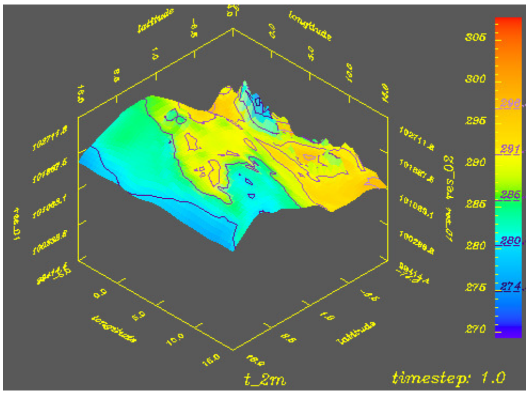
This is the planar 2D visualization module for regular and block-structured grids. It supports 2D and 3D data sets (with animation). Using coloring, isolines, band visualization and colored height maps a maximum of 3 parameters can be compared in one image. Furthermore, for 3D data, a selectable dimension can be interactively changed or automatically animated.

Technique modes

Mode	Example Image	Description	Supported (important) goals
Pure color	 16.0 8.5 1.0 -6.5 -14.0 latitude -5.0 0.0 5.0 10.0 15.0 longitude res_02 305 300 295 290 285 280 275 270 t_2m timestep: 1.0	The values of a certain parameter are mapped to colors on a 2D plane and are displayed using interpolation.	identify, localize, overview, scalars, outliers
Discrete pure color	 16.0 8.5 1.0 -6.5 -14.0 latitude -5.0 0.0 5.0 10.0 15.0 longitude res_02 305 300 295 290 285 280 275 270 t_2m timestep: 1.0	The values of a certain parameter are mapped to colors on a 2D plane without interpolation.	identify, localize, overview, pointwise, scalars,outliers, details

Mode	Example Image	Description	Supported (important) goals
Pure isolines	 16.0 8.5 1.0 −6.5 −14.0 latitude −5.0 0.0 5.0 10.0 15.0 longitude res_01 102345.4 101979.0 101612.7 101246.3 100879.9 100513.5 100147.2 99780.8 t_2m timestep: 1.0	Only isolines are displayed on 2D plane. Isolines can be grayscaled and captioned.	classify, scalars, shapes
Color and isoline	 15 10 5 0 −5 −10 latitude −4 −2 0 2 4 6 8 10 12 14 longitude res_02 305 300 296 295 291 290 285 285 280 280 275 274 270 t_2m timestep: 1.0	Additionally to the color mode, the same or another parameter can be mapped to an planar isoline visualization.	identify, localize, overview, scalars,outliers, shapes, associate
Color bands	 16.0 8.5 1.0 −6.5 −14.0 latitude −5.0 0.0 5.0 10.0 15.0 longitude res_02 305 300 295 290 285 280 275 270 t_2m timestep: 1.0	A discrete number of colored bands are displayed. Banding is the equivocal mapping of ranges of data values to a certain colors without color interpolation.	classify, scalars, shapes, rank

Mode	Example Image	Description	Supported (important) goals
Color bands and isolines	 latitude 16.0 8.6 1.0 −6.5 −14.0 −5.0 0.0 5.0 10.0 15.0 longitude t_2m timestep: 1.0 res_02 305 300 295 290 285 280 275 270	A discrete number of colored bands are displayed and their shapes are sharpened using isolines.	classify, scalars, shapes, rank
Colored height field	 latitude longitude t_2m timestep: 1.0 305 300 295 290 285 280 275 270	A certain parameter can be mapped to a height map, a second to the color of the map.	associate, shapes,outliers
Colored discrete height field	 latitude longitude t_2m timestep: 1.0 305 300 295 290 285 280 275 270	A certain parameter can be mapped to a height map, a second to the color. Both height and colors are non-interpolated.	localize, scalars, outliers, shapes,

Mode	Example Image	Description	Supported (important) goals
Colored height field with isolines		Colored height field mode with isolines - A certain parameter is mapped to the height, a second to the color, and a third to isolines.	associate, scalars, directions, shapes,outliers, classify

Technique parameters

Parameter name	Parameter description	Parameter values
Select dialog	Select a certain parameter dialog (in analogy to a tabbed dialog).	General options, Save image options, Height options, Color options, Isoline options, Animation options, Selection options
Data normalized based on	To change data value normalization. Determines min-max for height, color and isolines based on different min-max values	Metadata min-max (based on min and max of result values from netcdf); Global data min-max (based on calculated min-max values, if netcdf-values do not exist or are inconsistent with the data); Current data min-max (based on calculated min-max for currently selected data slice(3D data only))
Height	Umschalten zwischen Höhenfeldansicht und planer 2D Ansicht	
Isoline attribute	Which attribute will be represented by isolines?	
Color Attribute	Which attribute will be represented by color?	
Height Attribute	Which attribute will be represented by height?	

Sequenzer	Launch / hide sequencer (to control animations)	
Sequenzer Dim	In which dimension should be animated?	
Current Value	If sequences is hidden: direct control to "Sequencer dim" slices.	
Toggle dims	Vertauschen der x und y Achse des Datensatzes (die Weltkarte wird dabei nicht gedreht).	
Grid	Show/hide the grid	
Interpolation	An- und Abschalten der Interpolation	
View Mode	Umschalten zwischen Globus und planarer Ansicht	
Isoline Color	Auswahl der Isolinien-Farbekodierung	weiß, schwarz, oder Grauskalierung
Isoline Type		manual, automatic or no isolines
Isoline Values	If "Isoline Type" = manual: you can specify a list of isovalues (if the value is out of range dx will cause an error message (see restrictions))	
Isoline Numbers	If "Isoline Type" = automatic: computer will generate isoline values with equal distances.	
Values	Show/Hide isoline annotations	on/off
Fontsize	fontsize of isoline annotations	
Center Values	Display isoline annotation at the start or at the center of the their isoline?	
...	to be continued	

Technique restrictions

No isovalue annotation in globe mode and in height mode implemented.
Banding has problems with missing value regions (OpenDX sided), f.i. if only continents define values.
If the data are defined in the grid cell center, a first prototype has been implemented. It does not work for most modes and is buggy.

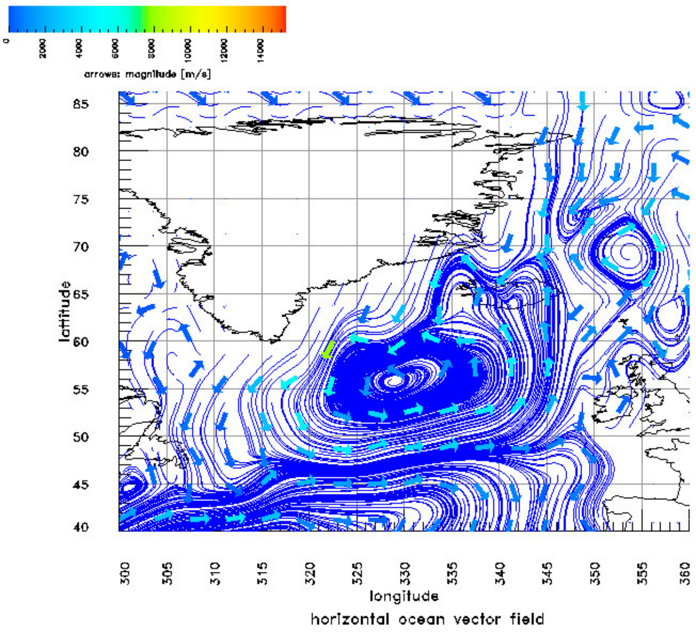
Grid display, min-max axes labeling are not totally correct for all modes and other parameter combinations (e.g. scaling).

<< previous | module overview | next>>

back to main page

<< previous | module overview | next>>

SimEnvVis - 2D Flow Visualization Module



Technique description

This is the planar 2D flow visualization module for regular and block-structured grids. Currently, it supports 2D data only (no animation).
It supports to map the local and global properties of the flow field using **streamlines**, local flow direction and magnitude using **arrows** and a scalar (flow fields derived) variable to the **background**. All can be mapped using a single color or a color map. To avoid overlapping and invisibility of low flow magnitude values arrows are of equal length.

Technique modes

Color mode – Currently, only a standard mode (see the figure) is available in the wizard. The user needs to change the values manually in the techniques to get other image parameterizations.

Technique parameters

Select dialog - Select a certain parameter dialog (in analogy to a tabbed dialog). Available dialogs are: General options, Save image options, Streamline options, Glyph options, Background options.

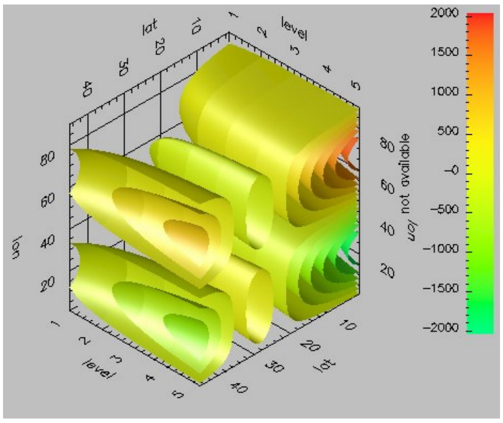
Technique restrictions

Currently, no time dependent flow fields are supported.

<< previous | module overview | next>>

back to main page

SimEnvVis - 3D Visualization Module



Technique description

The 3D visualization module includes a variety of modes for regular and blockstructured 3D and 4D data. These are direct volume rendering, glyph visualization, isosurface visualization and slice visualization.
Furthermore, a mode for the analysis of changes in a selected dimension is included. The first plane represents the starting values. The other slices represent the changes of the values at incresing dimension values.

Technique modi

Volume rendering - This technique casts rays into the 3D dataspace and collects information along this ray, resulting a certain color value. The current OpenDX default implementation does a simple alpha blending along the ray.
Glyph visualization - This technique constructs simple geometric objects (e.g. spheres or cube) and layouts them in 3D space. Color and size can be adapted owing to the data value at their 3D position. Because of occlusion and the rising amount of triangles for large data sets the technique is suited only for small 3D data sets.
Isosurface visualization - Automated or manual defined isosovalues form isosurfaces in 3D (see figure).
Data slice visualization - Along one dimension semi-transparent planes represent colored cuts in data space. The number of cuts can be interactively adapted (overview and detail-on-demand).
Changes slice visualizations - Along one dimension semi-transparent colored planes represent changes to a reference plane (e.g. timestep 0). The changes and reference plane can be displayed with different color maps. The number of cuts can be interactively adapted (overview and detail-on-demand).
Cutting planes visualizaion - 3 cutting planes ortogonal to all the 3 dimensions are displayed. The planes include color coding and isolines.

Technique parameters

Sequence dim - The ’sequence dim’ can be animated using the sequencer (4D data only). Furthermore certain slices of the ’sequence dim’ can be selected using the ’current value’ parameter.

Technique restrictions

No restrictions known.

<< previous | module overview | next>>

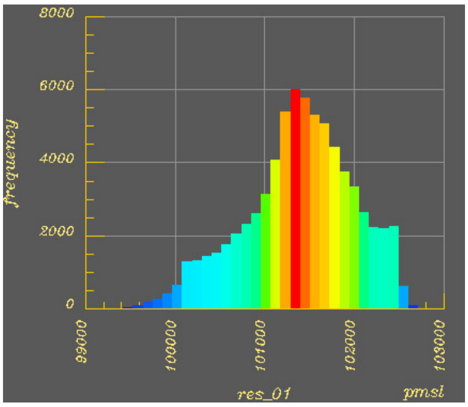
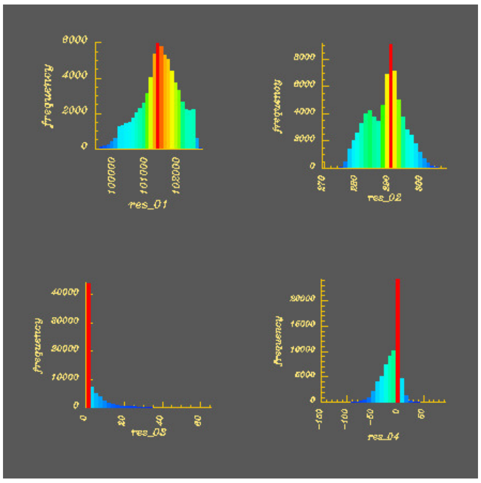
back to main page

SimEnvVis - Histogram Module (OpenDX)

Technique description

This is a simple histogram plot technique. Single histograms can be displayed as well as histogram tables of all results.

Technique modes

Mode	Example Image	Description
Single mode		The single mode includes the interactive visualization of one certain histogram. In contrast to the table mode, the user can zoom into certain parts of the selected histogram.
Table mode		The table mode allows to analyze a certain number of histograms concurrently. The number of columns and the size of the histograms can be adapted.

General technique parameters

Select dialog – Select a certain parameter dialog (in analogy to a tabbed dialog). Available dialogs are: General options, Save image options, Single mode options, Double mode options, Multi-line mode options, Table mode options.

Render mode – To switch between the four modes (Single mode, Double mode, Multi-line mode, Table mode)

y-axis type - Specifies if the y-axis (the amount axis) will be displayed normally (linear mode), or on a logarithmic scale. Be sure that you use the logarithmic value only if either all values are positive or all values are negative. Else the data will be transformed not correctly.

Technique restrictions

No restrictions known.

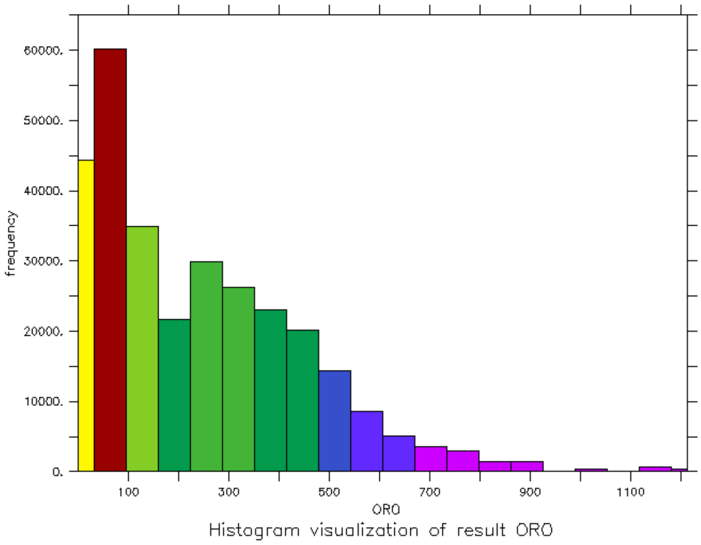
<< previous | module overview | next>>

---

back to main page

<< previous | module overview | next>>

SimEnvVis - Histogram Module (Ferret)



Technique description

This is a ferret histogram visualization based on bar charts. It uses an adapted ferret script which can be controlled from SimEnvVis. The user can change between different variables manually.

Furthermore, in Expert mode, the user can change the visualization by manual adaption of the script and save adapted scripts (with the currently selected parameters) for later use outside of SimEnvVis.

General technique parameters

Selected result – Select a certain result (variable) to display

Nr bins – To switch between 3 numbers of bars.

Colormap 1 – Switch between different ferret specific color maps.

Technique restrictions

Currently, the two bars near to variable minimum and maximum are drawn in half thickness of the other bars only. This avoids misunderstanding of ferret default bars that are drawn beyound the minimum and maximum values.

<< previous | module overview | next>>

back to main page

SimEnvVis - Parallel Coordinate Module

Technique description

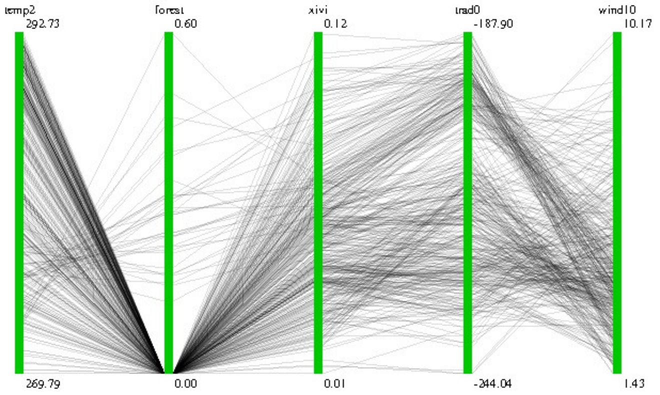
The parallel coordinates technique visualizes the different parameters of a data set by mapping the values to certain parallel lines. The data values of a single data record are mapped to a certain point between the minimum (bottom) and the maximum (top) value of such a parameter line, according to their value. Finally the points of a data record are connected by line segments. Thus, every data record is represented as a specific pattern of linked line segments.

In general, the technique allows to analyze parameter correlations. Furthermore, the color option allows to analyze, how the values of one attribute correspond with the values of all other attributes.

Technique modes

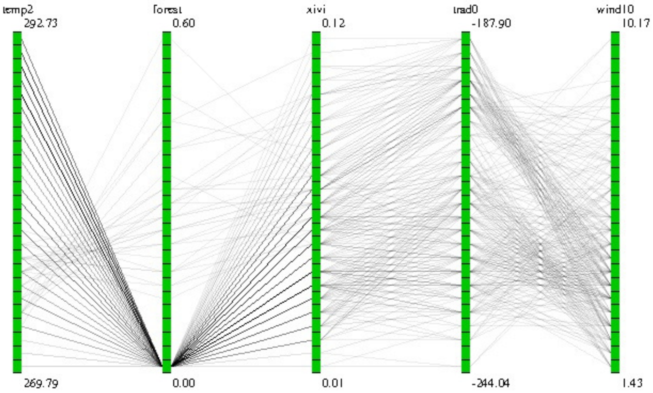
Max resolution mode

This mode is the typical parallel coordinate layout. All pixels are used. This mode allows the user to analyse the dataset in the most detailed representation possible with the current resolution. The disadvanages are high storage costs and slow response times for large datasets.



Regular axis interval mode

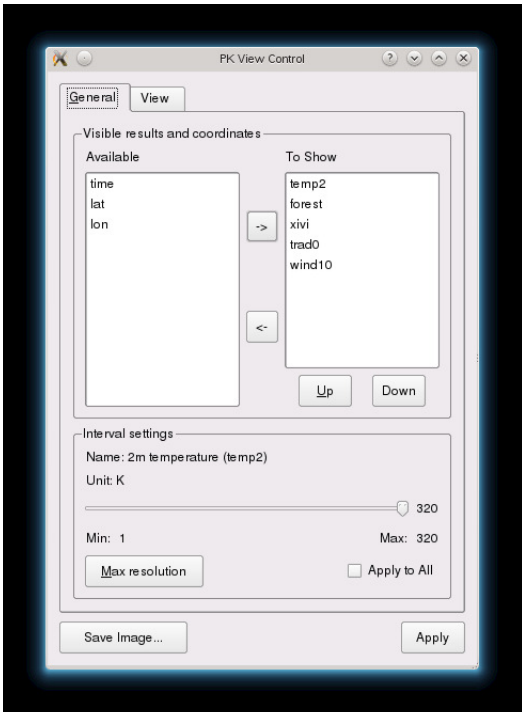
This mode uses a certain number of intervals on each axis. For larger data sets this mode leads to lower storage costs and response times. The disadvantage is however that detailed information will be lost. are lost.



Technique parameters

Parametrization of the technique can be performed in the View Control Dialog, which is divided into two tabs. The "General"-Tab allows the user to specify which variables should be shown and how many intervals should be present for each axis. The "View"-Tab allows the user to change the appearance of the lines. This includes the color of the selected lines, an overall alpha-value and if lines should be hidden if they don't match a specific criterion.

General Tab



Visible results and coordinates

Available	To Show
time	temp2
lat	forest
lon	xivi
	trad0
	wind10

Interval settings

Name: 2m temperature (temp2)

Unit: K

320

Min: 1 Max: 320

□ Apply to All

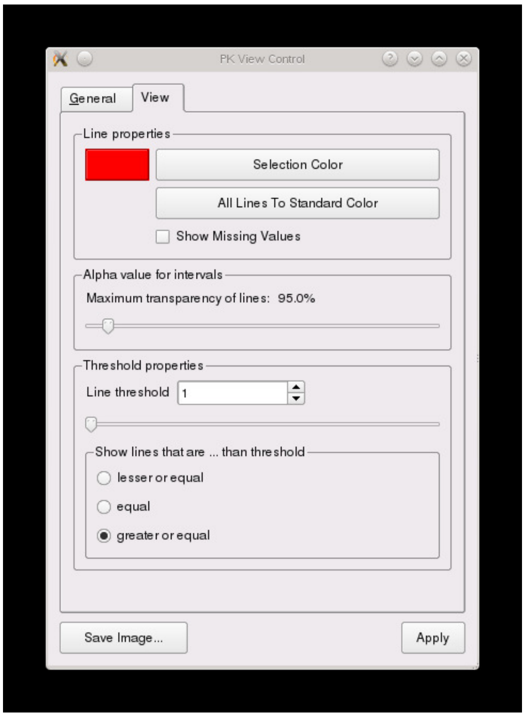
Visible results and coordinates
This region is divided into two Listboxes. The "Available"-Listbox shows the variables which are present in the current subset but not included in the current visualization. The "To Show"-Listbox lists the variables which are currently used in the current visualization. By selecting an item of one of these listboxes, the user can move the item to the other list using the arrow-icons. The Up- and Down-Icons

can be used to change the position of the axis of this variable.

Interval Settings

This region is used to set the number of intervals for one axis. The axis on which those parameters will affect can be selected on the visualization window using the mouse cursor. If the number of intervals should be applied to all axes, the "Apply to All"-Checkbox can be checked. The "Max resolution"-Button can be used for convenience to change the number of intervals for each variable to the maximum possible one resulting in each pixel of an axis to be one interval.

View Tab



□ Show Missing Values

Maximum transparency of lines: 95.0%

Line threshold 1

Show lines that are ... than threshold

○ lesser or equal

○ equal

● greater or equal

Line properties

When clicking and holding the left mouse button in the visualization window, the user can select a region. Every line which runs through one of the intervals which lies within the rectangle will be highlighted using a specific color. This color can be chosen by clicking on the "Selection Color"-Button. Clicking on the "All Lines to Standard Color" will set the color of every line to black. The "Show Missing Values"-Checkbox is used to show or hide lines that have one or more missing values. The missing values will then be located at specific regions beneath the axes. For dimension variables, missing value regions will not be displyed as dimensions are not supposed to have missing values.

Alpha value for intervals

With the slider, a transparency value can be applied to the lines. A line between two axes represents each data record whose values are located at the intervals of the axes defined by the start and end points of the line. A dark line indicates that many data records are represented by this lines and a light line indicatse only few data records are represented by this line. The slider can be used to define the maximum transparency of the lines. This means that lines with few data records get even lighter (even to near total transparency) whilst lines with many data records keep their dark color.
Hint: The logic behind this feature will be improved in future versions, because it may lead to artifacts in specific cases.

Threshold properties
In this region a threshold can be specified against which each line is checked by using the expression given by one of the radio buttons (lesser or equal, equal or greater or equal). If a line does not match the expression, it will not be displayed. This can be useful to remove lines which are not of interest.

Technique restrictions

Theoretically, this technique can be applied to an unlimited amount of lines as long as the processing power is sufficient. In practice, the efficiency of this technique is dependent on the internal characteristics of the dataset. In some cases, e.g. if the dataset is very homogenous and thus very regular representations are given, it can be hard to analyze the dataset.

<< previous | module overview | next>>

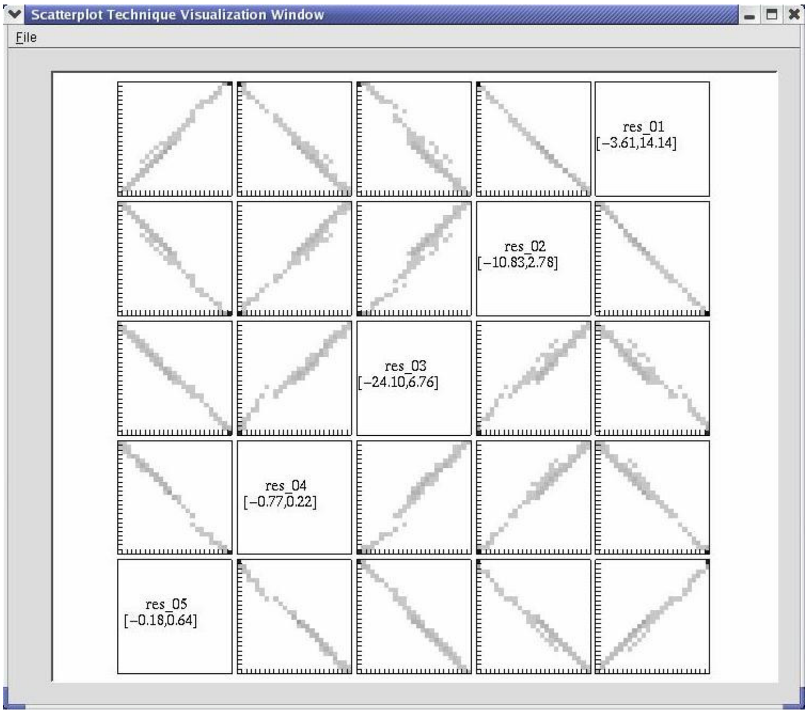
back to main page

<< previous | module overview | next>>

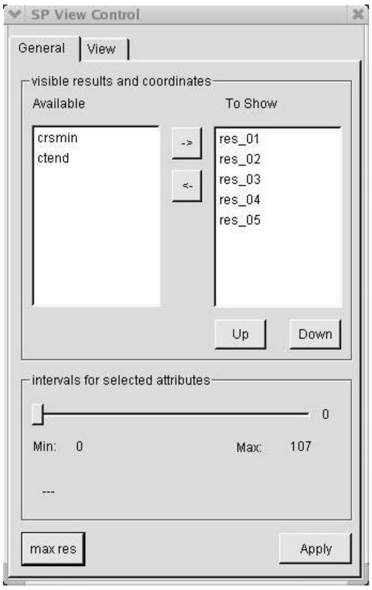
SimEnvVis - Scatterplot Matrix Module

Technique description

The scatterplot matrix consists of scatterplots of all pairs of results. Additionally, even the coordinates can be included.
In general, the technique allows to analyze parameter correlations. Furthermore, the color option allows to analyze, how the values of one attribute correspond with the values of all the other attributes.

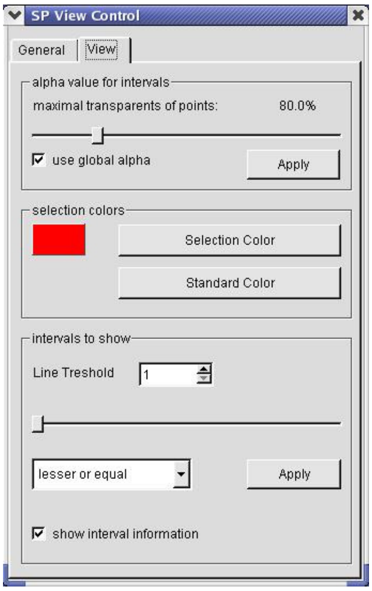


Control Dialog



Min: 0 Max: 107

This is the "General" tab of the Scatterplot Matrix Control Dialog. It’s slightly different from the one used for the Parallel Coordinates Technique. A big advantage is the possibility to choose more that one axis. This is done by selecting the corresponding axes in the upper right list box. Now it is possible to change the number of intervals for all of these axis at once. Another possibility to select axes is to click into the visualization window on a specific plot, both of the plotted axis now are selected. As well as in the Control Dialog of the Parallel Coordinates technique it’s possible to change the order and choice of axis to display. The "max res" button sets number of intervals for all axis to the possible maximum.



maximal transparents of points: 80.0%

☑ use global alpha

Line Treshold 1

lesser or equal

☑ show interval information

The "View" tab of the Control Dialog allows to change the appearance of the visualization. Starting from above there is the possibility to change trancparency, selection color, to switch back to standard color and to choose wich plots to display.

Technique modi

Max resolution mode

This mode is the typical parallel coordinate layout. All pixels are used. Disadvantage: in large data sets high storage and time costs.

Regular axis interval mode

This mode uses a certain number of intervals on each axis. Advantage: in larger data sets lower storage and time costs. Disadvantage: small value differences are lost.

Technique parameters

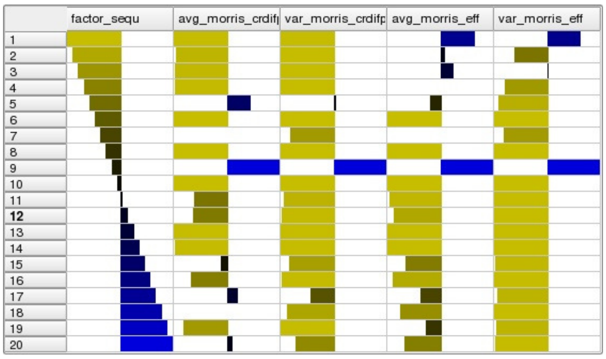
Technique restrictions

<< previous | module overview | next>>

---

back to main page

SimEnvVis – Graphical Table Module



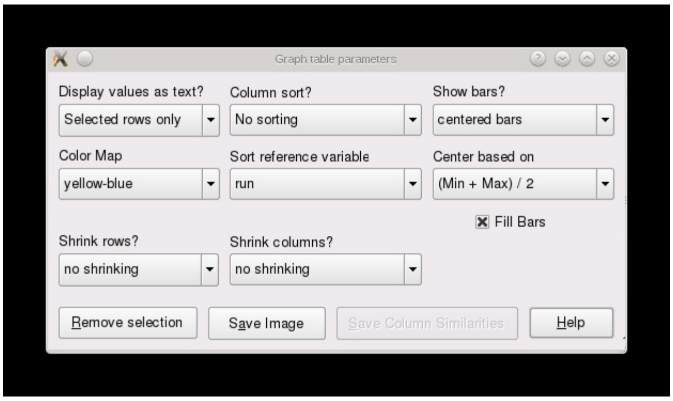
Technique description

This is the QT implementation of an interactive graphical table. It supports sorting, shrinking of lines, display of textual values, selection, changing of color mapping ...

Technique modes

Default mode – Currently, only a standard mode (see the figure) is available in the wizard. The user needs to change the values manually in the techniques to get other image parameterizations.

Technique parameters


Graph table parameters

Display values as text? Selected rows only

Column sort? No sorting

Show bars? centered bars

Color Map yellow-blue

Sort reference variable run

Center based on (Min + Max) / 2

☒ Fill Bars

Shrink rows? no shrinking

Shrink columns? no shrinking

Display values as text
This parameter allows the user to decide in which cells the corresponding numerical values should be displayed.
Options are:

No Text	Displays no values.
Selected rows only	Displays values in selected rows only.
All values	Displays values in all rows.

Show bars
This parameter allows the user to define how the bars are displayed.
Options are:

no bars	Bars are not displayed. Instead, a line is displayed in representing the value of the cell. The lowest value will be displayed at the very left of each cell.
centered bars	The bars are displayed with the center of each cell as the origin. The meaning of the origin can be changed by the "Center based on"-functionality.
left to right bars	The bars are displayed in left to right direction. The lowest value will be displayed at the very left of each cell.

Fill bars
This Checkbox allows the user to enable or disable the bar representation of the numerical value to displayed in each cell.
Options are:

checked	Bars are displayed in fill mode.
unchecked	Bars are not displayed in fill mode.

Center based on
This parameter allows the user to change the position of the origin of the bar in each cell in dependency to the input dataset.
Options are:

Data mean	The value at the origin is equal to the arithmetic mean of all values of the data.
Zero	The value at the origin is equal to zero.
(Min + Max)/2	The value at the origin is equal to the arithmetic mean of the maximum and the minimum of the data.

Color Map
This parameter allows the user to select a Color Map to define in which colors the bars will be displayed.
Currently, three Color Maps are supported:

yellow-blue
green-red
green-yellow

Column sort
This parameter allows the user to calculate the correlation coefficients between a reference variable as defined in the combobox named "Sort reference variable" and each other variable. The columns then are sorted in dependency to their corresponding correlation coefficient.
Options are:

No Sorting	No correlation coefficients are calculated and no sorting is performed.
By correlation (signed)	Correlation coefficients are being calculated and sorting is performed using the signed values.
By correlation (unsigned)	Correlation coefficients are being calculated and sorting is performed using the absolute values.

Sort reference variable
This combobox allows the user to choose a reference variable. The correlation coefficients will be calculated between this variable and each other variable.

Shrink rows
This parameter allows the user to shrink each row. This option is useful to get an overview about the data and to get detailed information about specific values within their overall context. The shrinking is performed in such a way so that each shrinked row is of same height. This functionality will try to display all values on one page. But the maximum height of each row is limited to one pixel. If there are more rows than pixels on the window, the rows will not fit on one page.
Options are:

no shrinking	The rows will not be shrinked. The height of each row is set to a standard value.
shrink unselected	All rows will be shrinked except the selected ones.
shrink all	All rows will be shrinked.

Hint: In case of very low numbers of rows, the shrinking may lead to expanding.

Shrink columns
This parameter allows the user to shrink each column. The shrinking is performed in such a way that the columns are of same width.
Options are:

no shrinking	The columns will not be shrinked. The width of each column is set to a standard value.
shrink all	All columns will be shrinked.

Hint: In case of very low numbers of columns, the shrinking may lead to expanding.

Buttons
There are several buttons on the bottom of the parameters dialog with different functionality. Buttons are:

Help	Displays the help page for this technique.
Save Column Similiarities	Saves the calculated similiarities as done by the functionality "Column sort" for each column to a text file.
Save Image	Saves the displayed graphical table to an image.
Remove Selection	Removes the current selection.

Technique restrictions

Currently, this technique can be used with a maximum of 10000 data records.

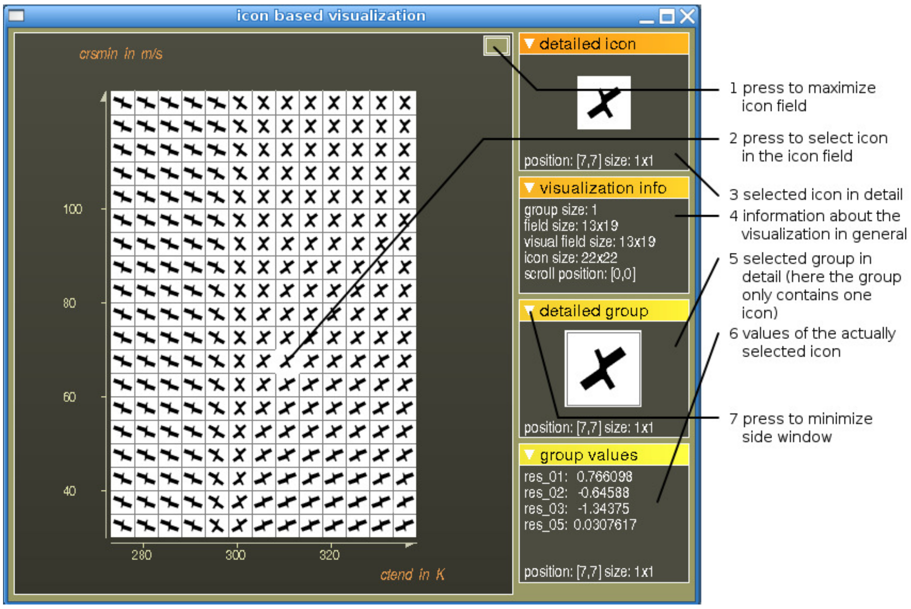
<< previous | module overview | next>>

back to main page

SimEnvVis - Icon Based Visualization Module

Technique description

The Icon Based Visualization represents data using small pictures (icons) to represent values. One icon can show different values by using attributes like color, size, angle and others. The following picture shows the main window.



position: [7,7] size: 1x1

group size: 1
field size: 13x19
visual field size: 13x19
icon size: 22x22
scroll position: [0,0]

position: [7,7] size: 1x1

res_01: 0.766098
res_02: -0.64588
res_03: -1.34375
res_05: 0.0307617

position: [7,7] size: 1x1

1 press to maximize icon field

2 press to select icon in the icon field

3 selected icon in detail

4 information about the visualization in general

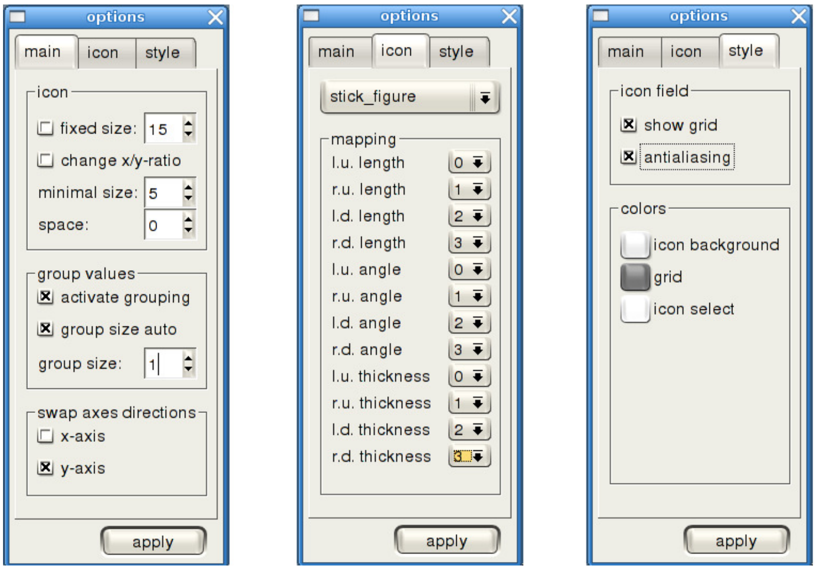
5 selected group in detail (here the group only contains one icon)

6 values of the actually selected icon

7 press to minimize side window

Control Dialog

The Control Dialog contains three tabs. The following picture shows all of them.



icon
- ☐ fixed size: 15
- ☐ change x/y-ratio
- minimal size: 5
- space: 0

group values
- ☒ activate grouping
- ☒ group size auto
- group size: 1

swap axes directions
- ☐ x-axis
- ☒ y-axis

stick_figure

mapping
- l.u. length 0
- r.u. length 1
- l.d. length 2
- r.d. length 3
- l.u. angle 0
- r.u. angle 1
- l.d. angle 2
- r.d. angle 3
- l.u. thickness 0
- r.u. thickness 1
- l.d. thickness 2
- r.d. thickness 3

icon field
- ☒ show grid
- ☒ antialiasing

colors
- icon background
- grid
- icon select

1. main tab

The main tab allows to control the appearance of the icons. Within the *icon group* one can choose a fixed size for the icons in pixel. If fixed size was deactivated the size of an icon depends on data size, minimal size and grouping options. With a fixed size the icon is always quadratical. The change x/y-ratio checkbox enables the possibility for different width and heigth of an icon. This only works when fixed size was deactivated. The minimal size of an icon (for both heigth and width when change x/y-ratio is enabled) prevents the icon's size to drop below the given number of pixels. The space parameter controls the space between the icons in the grid.
The group values parameters are used to paint one icon for more than one data value. If *grouping* is activated with the *group size auto* option disabled one can control the size of a group directly with the group size spinbox. In case
that *group size auto* is enabled the group size value is ignored and instead icon size or minimal icon size values are used to determine the group size.
The two checkboxes grouped together under *swap axes directions* allow to swap the direction of the axes :-).

2. icon tab

In the icon tab one can choose among different icons (at the moment between stick_figure and cross icon). The available icons are listed in a combo box. After having choosen an icon one has to specify a mapping. Every icon provides several named attributes and these are listed in the tab. Next to every attribute one can find a combo box with all available data attributes for mapping. It is possible to map the same data attribute several times to different visual attributes of the icon (e.g. to color and angle). It isn't possible to do so the other way round, a visual attribute can't have more than one data attribute.

3. style tab

The style tab allows to choose wether to paint a grid or not and if antialiasing should be used. Using antialiasing results in a better but slower visualization. In addition its possible to choose the color of grid, background and selection.

Technique modi

Technique parameters

Technique restrictions

<< previous | module overview | next>>

---

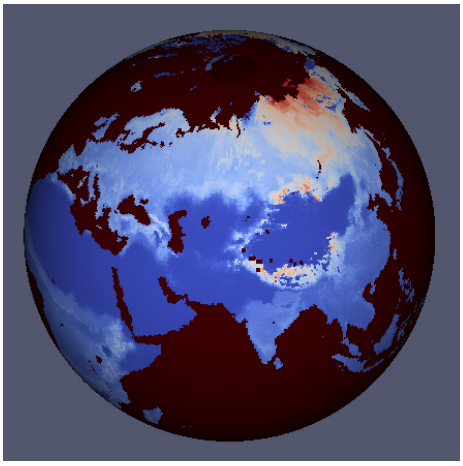
back to main page

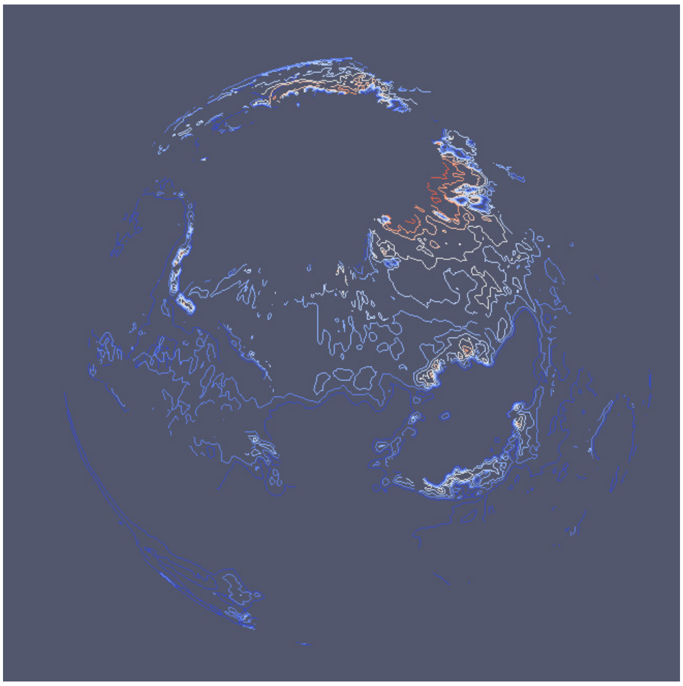
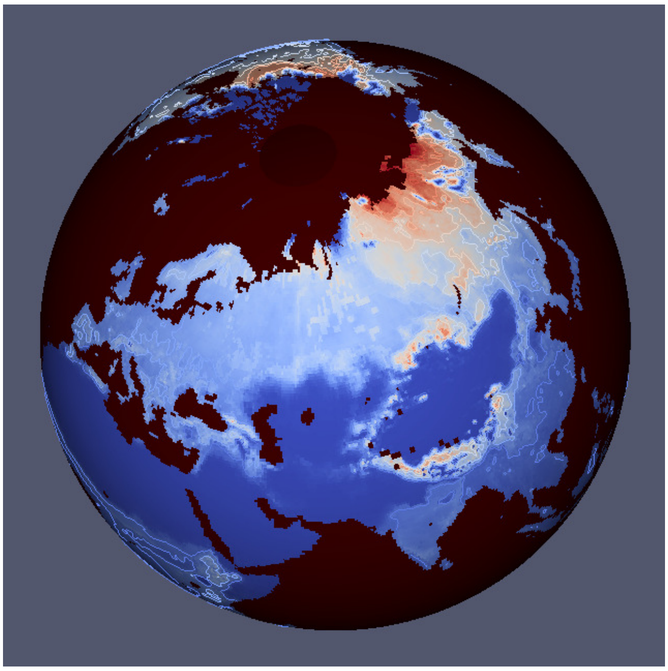
SimEnvVis - 2D/3D Visualization ParaView Module

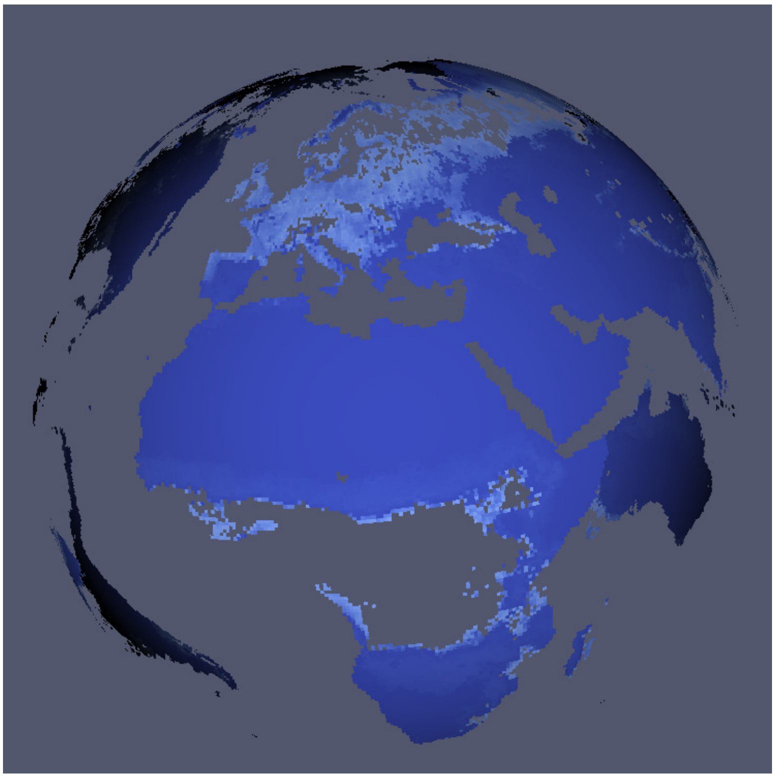
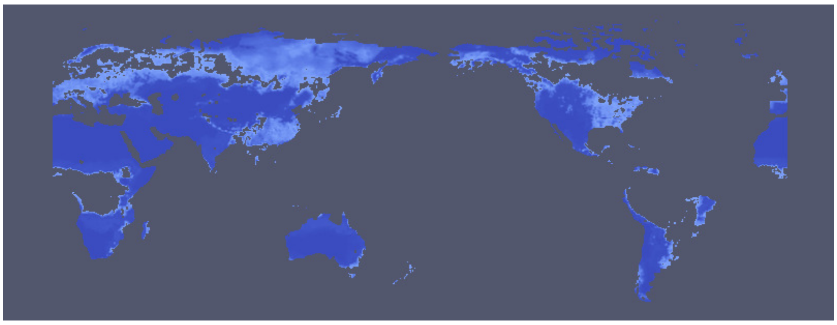
Technique description

This is the ParaView module for SimEnvVis which can be used for 2D and 3D visualizations. This module is suitable for 2D and 3D geospatial datasets with or without a temporal component. It will launch ParaView with a template suitable for the given dataset and ready for inspection out-of-the-box. After starting, the template can be dynamically adapted, if the user is already familiar with ParaView.

Technique modes

Mode	Example Image	Description
Color		The values of a certain variable are mapped to colors on a globe.

Mode	Example Image	Description
Isocontours		From the values of a certain variable, colored Isocontours are generated and mapped on a globe.
Color + Isocontours		A discrete number of colored bands are displayed. Banding is the equivocal mapping of ranges of data values to a certain colors without color interpolation.

Mode	Example Image	Description
Threshold		Only the values within a certain value range are mapped onto a globe. The range can be adapted within ParaView.
Threshold (Block)		Like "Threshold" but the the grid is constructed rather as a plane (2D) or block (3D) than a globe.

For a documentation on ParaView, please see its documentation. For information on the implementation details for ParaView see here.

Technique restrictions

The technique is only capable of displaying geospatial and temporal data.
Also, very large datasets maybe problematic for recent versions of ParaView (like > 4.0).

<< previous | module overview | next>>

back to main page

Implementation

To add more ParaView-based techniques to SimEnvVis, two appraoches can be followed. The already implemented techniques are based on a composition of both of them. First, ParaView state files (*.pvsm) are basically xml-files and can be treated as such. So, in Order to create a visualization template, the changing parst of the xml-file have to be located. Then, by using keywords, these parts can be exchanged with concrete information before ParaView ist launched. The used template for the state file is located in $SIMENVVISDIR/maps/ParaView_2D3D_Visualization.pvsm.
Second, ParaView can be started with a Python script. ParaView's Python API allows for modification in several ways, including loading files and constructing a new pipeline. In order to create new ParaView-based techniques, the script has to be extended. It is located in $SIMENVVISDIR/scripts/ParaViewStartUpScript.py. A very useful tool for writing Python scripts for ParaView can be found under Tools->Start Trace/Stop Trace. It displays the Python commands for all the taken actions when manipulating ParaView by using the GUI.
Both approaches have their pros and cons. For example, while python scripts are not able to create split views, state-files can not be adapted to create screenshots at start-up (which even works for multiple views!).

---

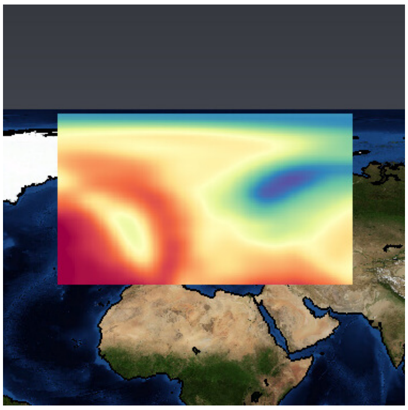
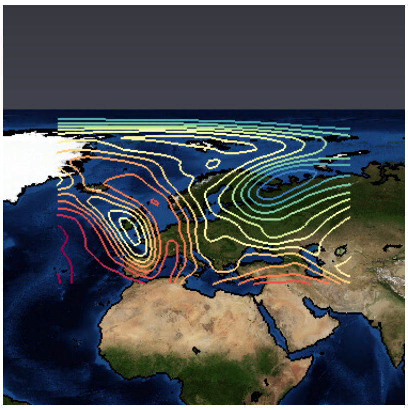
back to main page

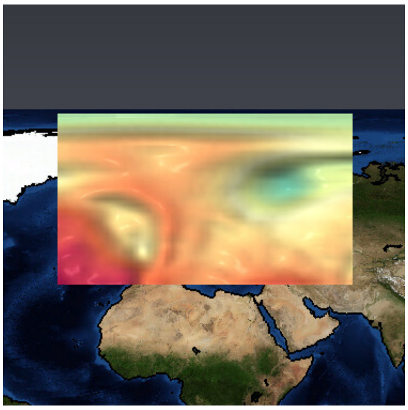
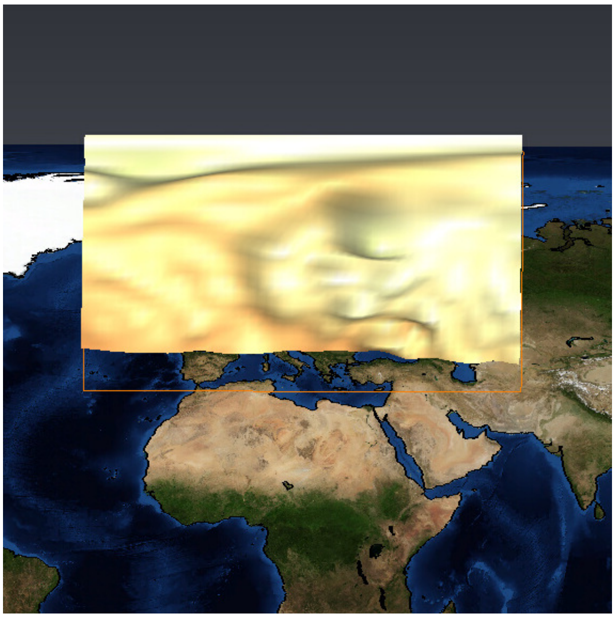
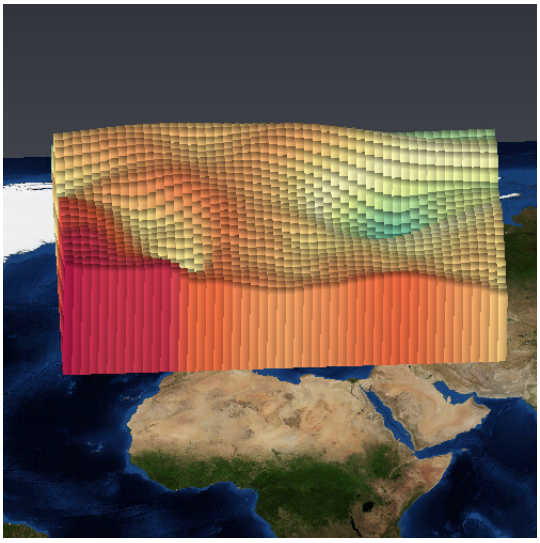
SimEnvVis - 2D Visualization Avizo Module

Technique description

This SimEnvVis' 2D visualization module based on the Avizo system. Currently, it supports five different techniques each being suitable for different contexts. After starting the technique, the Avizo template can be further refined manually if the user has experience with the Avizo visualization system. The technique might only work with Avizo Green Edition.

Technique modes

Mode	Example Image	Description
Color		The Color technique maps the values on a 2D plane.
Isocontours		Based on the variable input, isolines are constructed on a 2D plane.

Mode	Example Image	Description
Embossed Slice		The Embossed Slice module maps the values of a variable on color and uses a shading technique to mimic height/depth. This way, high/low values can be distinguished easier.
Height Map		The height map maps the values of a variable on both color and a height value.
Bar Plot		Using the bar plot, for each value a bar is constructed, leading to a discretized field of single bars, for example allowing to easily extract outliers.

For more detailed information on the work with Avizo, see the Avizo Green Tutorial by DKRZ.

Technique restrictions

The technique can only handle 2D geospatial data with or without a temporal component. There also might be problems with a high amount of data records.

<< previous | module overview | next>>

---

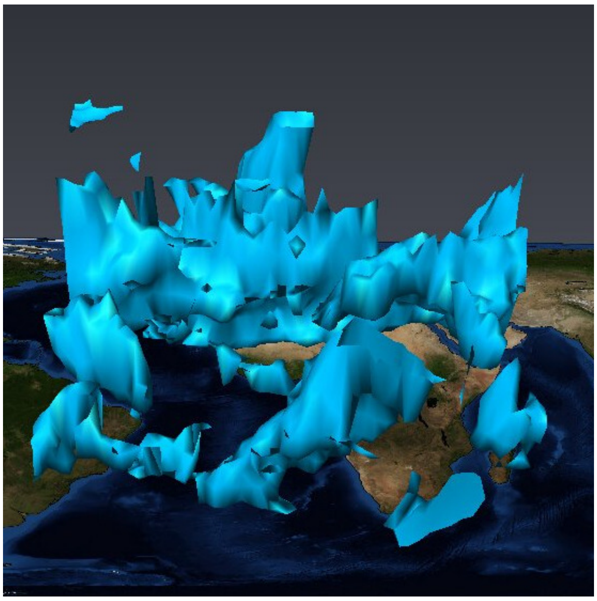
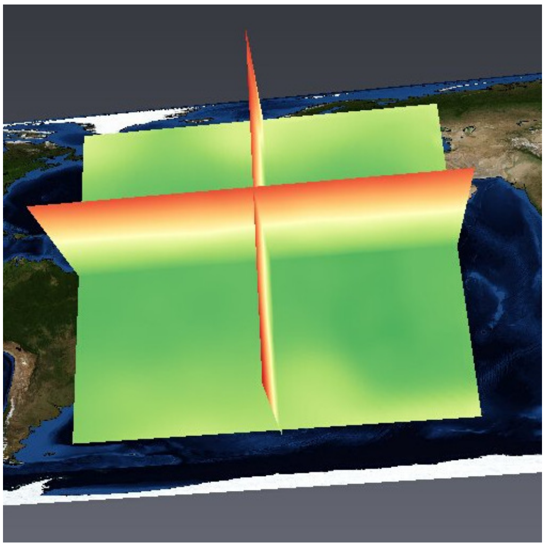
back to main page

SimEnvVis - 3D Visualization Avizo Module

Technique description

This SimEnvVis' 3D visualization module based on the Avizo system. Currently, it supports 2 different techniques each being suitable for different contexts. After starting the technique, the Avizo template can be further refined manually if the user has experience with the Avizo visualization system. The technique might only work with Avizo Green Edition.

Technique modes

Mode	Example Image	Description
Isosurfaces		This technique creates isosurfaces based on the variable's values. The isocontour values can be changed after Avizo has been started.
Cutting Planes		This technique creates cutting planes based on the variable's values. The planes' positions can be changed interactively after Avizo has been started.

For more detailed information on the work with Avizo, see the Avizo Green Tutorial by DKRZ.

Technique restrictions

The technique can only handle 3D geospatial data with or without a temporal component. If 2D data is given, the behaviour is undefined. There also might be problems with a high amount of data records.

<< previous | module overview | next>>

back to main page

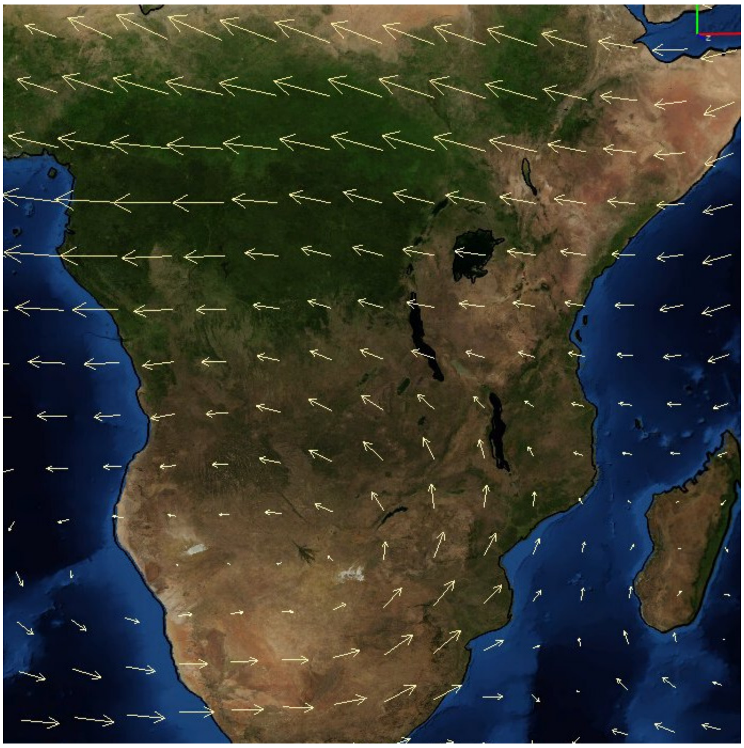
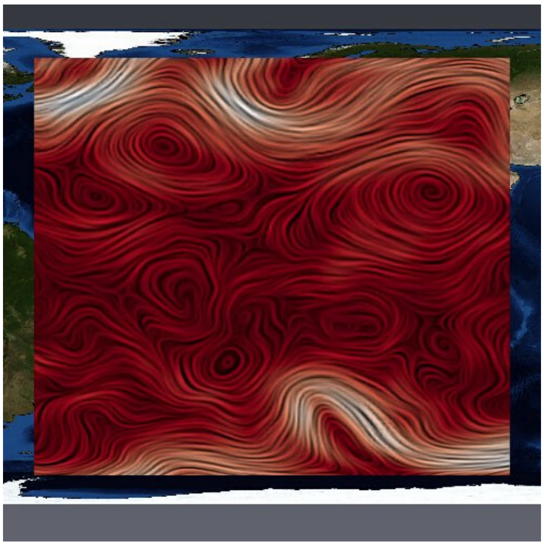
<< previous | module overview | next>>

SimEnvVis - 2D Vector Visualization Avizo Module

Technique description

This SimEnvVis' 2D visualization module based on the Avizo system. Currently, it supports two techniques each being suitable for different contexts. The techniques will use the first two variables and interprete them as components of a vector, so they have to be picked carefully. After starting the technique, the Avizo template can be further refined manually if the user has experience with the Avizo visualization system. The technique might only work with Avizo Green Edition.

Technique modes

Mode	Example Image	Description
Arrow Visualization		This technique uses the given vector components and constructs arrows based on their values.
Stream Visualization		Based on the vector components, this technique creates a stream visualization.

For more detailed information on the work with Avizo, see the Avizo Green Tutorial by DKRZ.

Technique restrictions

The technique can only handle 2D geospatial data with or without a temporal component. There also might be problems with a high amount of data records. Remember to pick the components carefully.

<< previous | module overview | next>>

back to main page

SimEnvVis - Installation and getting started

This page provides guidance for setting up and running SimEnvVis.
For questions, suggestions for improvement, etc., please contact:
Magnus.Heitzler#pik-potsdam.de or
Nocke#pik-potsdam.de
(substitude # with @)

Installation

SimEnvVis can be built with different visualization techniques available. This depends on the modules that are installed on the system. SimEnvVis searches for these dependencies in a couple of standard directories. The following section provides information about mandatory and optional dependencies and also how to specify environment variables to help SimEnvVis find these variables. The variables for which this environment variables can be set are listed below the corresponding dependencies (e.g. NetCDFHOME for NetCDF).

Setting up the environment

Mandatory dependencies

First, you will need the following dependencies which are mandatory for SimEnvVis. They will assure that SimEnvVis can be built and provide a minimal set of base techniques (Graphical Table, Parallel Coordinates, Scatterplot Matrix).

- NetCDF (Tested versions: 3.1 - 4.0.1)
  - NetCDFHOME -- If SimEnvVis' search for the NetCDF libraries fails, you might want to set this environment variable. It should point to the NetCDF libraries directory (e.g. /usr/lib64/).
- QT (Tested versions: 3.3.5 - 3.3.8)
- OpenGL
- NCO (NetCDF Operators) package.
- CDO (Climate Data Operators).
- Common linux commands like cp, rm, etc.

Optional dependencies

Second, the following dependencies assure that SimEnvVis can access further visualization techniques. Note however that SimEnvVis also runs without them but then will not be able to provide the corresponding techniques.

- OpenDX (Tested versions: 4.3.2 - 4.4.4)
  - DXROOT -- If SimEnvVis' search for the OpenDX libraries fails, you might want to set this environment variable. It should point to the OpenDX installation directory (e.g. /opt/dx/).
- Avizo (Tested version: 7)
  - AMIRA_ROOT -- This variable is needed for Avizo to find its installation directory where it can access additional needed for the visualization.
- Ferret (Tested version: 6.72)
  - FER_DIR -- If ferret based techniques should be available, this environment variable must be set. It must point to the ferret base directory (e.g. /usr/local/ferret/).

- ComVis
- ParaView (Tested versions: 3.98.0 - 4.0.1)

Additional mandatory environment variables

The following environment variables must be set when installing SimEnvVis.

- QTDIR -- QT installation directory (e.g. /usr/lib/qt3/).
- YACCBINARY -- This variable holds the executable of yacc (e.g. /usr/bin/yacc).

The PATH variable should contain the path for QT3 executables (e.g. $QTDIR/bin/) and the paths to the executables of the visualization techniques.

Executing the installation scripts

Once the environment is set up, SimEnvVis can be installed. This is done by the following scripts.

./init.sh
First, execute init.sh. This will generate the files simenvvis.cfg, simenvvis.kdevelop (for developers) and a new Makefile.

./make.sh
Then, execute make.sh. This performs the compilation and the linking of SimEnvVis and its dependencies.

./install.sh
Finally, execute install.sh which takes one argument as the installation directory. This will create the doxygen documentation and then copies all files necessary for execution to the installation directory.

Starting SimEnvVis

Once SimEnvVis has been built and installed successfully, the application can be started by using the simenvvis executable which resides in the bin directory of the installation directory. At start up simenvvis can be provided with arguments. These are explained in the following section.

Parameters

simenvvis [-f FileName [-direct AttributeName | -testInfo TestAttributes | -executeTest TestAttributes]] [-log UserName [-mname ModelName]]

Parameter	Description
-f FileName	A parameter defining a default NetCDF file.
-direct AttributeName	A certain attribute will be directly visualized without starting the SimEnvVis graphical user interface. Then, the visualization can be closed pushing the "Close Visualization" button in the main control panel. A filename must be given.
-log UserName	User name to be logged.
-mname	Model name instead of file name will be logged.
-testInfo TestAttributes	Returns number of subsets in the data (1), available visualization modules (2) and their modes (3) for a given data set (see 1), a selected subset (see 2) and a selected visualization module (see 3).
-executeTest TestAttributes	Executes SimEnvVis for a certain subset, a certain visualization module and a certain visualization mode (see testInfo) and generates an image file.

Configuration

SimEnvVis can be configured for individual needs by setting additional environment variables or by adapting the variables in the simenvvis.cfg file. This section provides some information about these topics.

Optional environment variables

The following variables are optional but may be used to reconfigure the behavior of SimEnvVis.

- SimEnvVisSaveImagePath -- Alternative path to write user saved images to (overwrites the value of ’SaveImagePath’ in simenvvis.cfg).

simenvvis.cfg

This file contains the external options of the SimEnvVis system. Most of them should not be changed as they are needed for external Techniques such as OpenDX to find mandatory files. However some of them might be useful and thus deserve further explanation:

Variable	Description
WorkingPath	This variable defines the directory which will be showed at start up of the SimEnvVis FileDialog. If you have a default folder to store your NetCDF-files in you might want to change this variable.
TempPath	During SimEnvVis’ execution, especially when starting certain techniques, temporary files will be created. These will be stored in the directory specified by this environment variable. Make sure this variable points to a directory which has enough space. This is crucial, especially when working with large datasets. SimEnvVis will remove all temporary files at shut down. However, if SimEnvVis crashes this might not be done, so if you run out of disk space, cleaning up SimEnvVis’ temporary directory might help.
SaveImagePath	This variable points to the directory SimEnvVis SaveImageDialog will show at start up. Change this variable to the location you prefer to save your images to.

back to main page

Update page - Changes from Version 0.99 to current Version 1.00

ParaView

- Added conversion from absolute to relative time axis prior to starting ParaView.
- Extended Script for more stable ParaView Start-Up. This also fixes a bug which led to a crash of ParaView when trying to edit the color scale.
- Added color bar on start-up.

General wizard functionality

- Adapted simenvvistequ.xml according to new schema.
- Added non-evaluating support for refinment information for attributes substask and for * task.
- Modified task descriptions using a new model of a design space consisting of Goals, Means, Characteristics, Target, and Cardinality.
- Added information on Boni and Mali on last wizard page.
- Added check if dimension has a corresponding variable.
- Selectable tasks are now dependant on selected visualization system.
- Added support for environmental variable $USER for working path.
- Added prototypical R support.
- When not present, SimEnvVis will now create userdata based on userdata.default on start-up.
- Added support for SubSets with dimensions of which some have attribute index_range and some which don't have attribute index_range.

Changes from Version 0.98 to current Version 0.99

General wizard functionality

- Added description for API in general_information.html.
- 1D Vis techniques suitabilities are dependent on spatial dimensionality anymore (only dependent on general dimensionality).
- Added suitability calculations for spatial dimensionality, abstract dimensionality and time suitability.

- Metadata dialog: Changed order of minimum and maximum values for dimensions.
- Fixed bug for QT/GL techniques resulting in data_range not to be read.
- Gatherage of missing metadata is now also performed when starting SimEnvVis in direct mode.
- When starting in direct mode, the variable to be displayed is now determined by its variable name instead of the standard_name.
- Improved UI of History Widget.
- Added alternative search for udunits2 when udunits is not found.
- Added support for $USER when specifying temporary directory.

Changes from Version 0.97 to Version 0.98

General wizard functionality

- Fixed bugs which prevented that the grid can be identified as block structured.
- Changed table headers to more CF conform ones and fixed some scaling issues on wizard page "Select Data for Visualization".
- When creating temporary directory, all needed parent directories are created.
- Added functionality to convert relative time format to human readable time format.
- Added search for udunits libraries at build time.

Parallel Coordinates

- Added layout tab to modify viewable items.
- Added support for human readable time format.

Scatterplot Matrix

- Added support for human readable time format.

Graphical Table

- Added support for human readable time format.

Avizo

- During check for Avizo at build time: the build process is not canceled if CDO is missing.

ParaView

- Added prototypical ParaView techniques to SimEnvVis that can be used to visualize 2D and 3D datasets.
- Added check for ParaView during build time.

Changes from Version 0.96 to Version 0.97

General wizard functionality

- Temporary files now are only accessible by the owner.
- Fixed a bug which lead to SimEnvVis crashing when boundary-dimension (e.g. nb2) was not the last dimension.

Ferret

- Additional error messages added if ferret is trying to acquire high amounts of memory.

Parallel Coordinates

- Changed save image functionality. The image is now created by grabbing the frame buffer.
  This fixes a bug on some installations which lead to black images.
  Still leads to black images when running in test mode.

Scatterplot Matrix

- Changed save image functionality. The image is now created by grabbing the frame buffer.
  This fixes a bug on some installations which lead to black images.
  Still leads to black images when running in test mode.

Avizo

- Removed dummy 2D techniques.
- Added 2D scalar visualization technique, 3D scalar visualization technique and 2D vector visualization technique.

OpenDX

- Added " -processors 1" to argument list when starting OpenDX which fixes a problem on start up for some systems.

Changes from Version 0.95 to Version 0.96

General wizard functionality

- Added ncrename, mkdir, rmdir to mandatory tool during build process.
- If temporary directory contains the symbol "%1", this symbol will be replaced by a random number. This directory will be created at start-up and deleted at shut-down.
- Removed warnings when userdata could not be found.

Ferret

- Histogram does not create frequency_h.dat and frequency_h.raw anymore.
- Histogram now has labels.
- Added helper scripts frequency_histogram2_mod.jnl and bar_chart3_mod.jnl to scripts-directory.
- Unsupported variable names (X, Y, Z, T) are renamed by adding the string "_SIMENVVIS".
- Imporved calculation of memory to be allocated by ferret.
- Added check for version-number when calculating memory to be allocated by ferret.
- Improved behavior when saving images.
- Moved extension of environmental variable FER_GO from start.sh to startFerretScript.sh. This also fixes a bug in test mode which lead to a white image to be saved.

Changes from Version 0.94 to Version 0.95

General wizard functionality

- Slightly improved general_information.html.
- Fixed bug: SimEnvVis does not crash anymore if file is invalid and histogram is started via shortcut on first page.
- User profiles are now saved in $HOME/.simenvvis/userdata. Default user profile is saved in $HOME/.simenvvis/userdata.default.

- Temporary files now contain the username.
- Fixed bug in start.sh: start.sh now also loads setSEVStartEnv.sh.
- Added check for ncks at build time.
- Removed dimensions from subsets view, if they are not directly used by any subset.
- start.sh now passes all arguments directly to SimEnvVis.
- Progress-Dialog now shows proper values during mdgatherage.
- long_name heuristic introduced in case of long_name missing.
- Added warning if a NetCDF-file is selected which is larger than 1GB.
- Added prototypical file browser which shows the subsets of a selected NetCDF-File.
- Added check for cdo when searching for Avizo.

Scatterplot Matrix

- Improved behavior when saving images.

Graphical Table

- Added checkbox to display long names instead of short names in column headers.

Avizo

- Added support for test mode.
- Mapping of the data range of longitudes [0° 360°] to [-180° 180°] is now performed automatically.
- Absolute time values are now converted to relative ones automatically, so that they can be displayed in Avizo.

Ferret

- Memory to be allocated by ferret is now calulcated in dependency to the dataset. If ferret is going to allocate more memory than available, an errormessage will be displayed.

OpenDX

- All kinds off possible missing metada for OpenDX techniques are now stored in a temporary nc file.
- OpenDX Histogram is stabilized: "include all" variables is not any more selectable in DX-GUI; showing all variables is possible using histogram button on fist page only; desicion of using table vs. single mode on first page now depends on the number of all dep. variables in data set.

- Inserted maximum memory requirement for dx.
- reduced cpu load of main executable during dx startup
- World map is hidden on default in 3D technique (because it is in test status)

Changes from Version 0.93 to Version 0.94

General wizard functionality

- Added check for available technique dependencies (OpenDX, Ferret, ComVis) at build time.
- Added extended search for OpenDX-dependencies of importnetcdf.c.
- Added search for file arch.mak in makefile of importnetcdf.c.
- Added file config.log for directory simenvvis holding information created during build process.
- Updated terminology in general_information.html.
- Only those techniques are displayed for which the needed libraries can be found during build time.
- Added separated search for files relevant to OpenDX.
- Added separated search for files relevant to NetCDF.
- Updated checks for OpenGL header file and libraries.
- Added prototypical visualization techniques for new visualization software: Avizo.
- If environmental variable SIMENVVISDIR is not set, SimEnvVis now tries to locate the standard directory.
- CF-conform boundary variables are not included in the SubSet-view anymore.
- Added table to display all global attributes. Removed textboxes to show global attributes.
- Added library libDX during build process. This is needed im some cases when error "undefined reference to _dxfImportBin_FP" is displayed.
- Added mechanism to remove temporary files after session has ended.
- If OpenGL is not available, Graphical Table can be used anyway.
- setSEVStartEnv.sh will now be created by running init.sh. The file will be created as a copy of setSEVStartEnv.sh.template.
- OpenDX-Techniques are now declared as "UnderConstruction".

Parallel Coordinates

- Fixed bug: Only black images were written when saving image.
- Improved behavior when saving images.

Changes from Version 0.92 to Version 0.93

General wizard functionality

- Added progress bar while retrieving missing metadata for variables.

Graphical Table

- When saving images the datatype is added to the end of the image-file if not given explicitly.

Scatterplot Matrix

- Fixed crashing issues especially for small datasets.
- Most left and upper intervals of a matrix can now be selected.
- Selections over more than one matrix can now be performed.

Parallel Coordinates

- Changed positioning behavior of axes to a more flexible one.
- Slightly changed appareance of OpenGL and Control windows.
- Global alpha is used in all cases instead of local alpha values.
- Changed default alpha value to 95.0%.
- Alpha values are now applied to brushes. For highlighting, the alpha value of the brushed lines is reduced by 10.0 percentage points. The latest brushed lines alpha values are reduced by 20.0 percentage points.
- Intervals are beeing brushed if their midpoints lie within the selection rectangle.
- All values of one axis can be brushed at once.
- Added regions for missing values below the axes of the dependent variables. Missing values of the lines are placed at these regions.
- Added a check box to show and hide missing value regions and the corresponding lines.
- Brushing can now be performed across axes and missing value regions.

- Thresholds for line counts are now working properly.
- Calculation of visible lines is now dependent on the height of the axis. Increases performance for short axes and accuracy for long axes.
- Fixed flickering when threshold settings changed.
- If no interval is selected, now no calculation is performed.
- Number of variables that are shown on start-up increased to 10.
- Changed formatting of axes and interval labels to more flexible values.

Changes from Version 0.91 to Version 0.92

General wizard functionality

- Fixed path to doxygen sub-documentation.
- Most QT/GL Techniques now should support datatypes "short" and "double".
- Fixed bug when SimEnvVis was crashing when reading long global attributes.
- Added calculation of minimum and maximum values for dependent variables if no "data_range" or "valid_range" attributes are given.
- Fixed calculation of minimum and maximum values for dimensions.
- Images created by OpenDX will now be saved in the correct location when test mode is running.

Graphical Table

- Added functionality to handle multi-dimensional data.
- Added functionality to support "add_offset" and "scale_factor".
- Default missing value is now 99999999 instead of 0.
- For dimensions no check for "missing_value" is performed, so every value will be displayed.
- Fixed wrong scaling behavior of control dialog.

Parallel Coordinates

- Now checking for missing value is also supported for data types other than float.
- Reduced number of apply-buttons to 1.
- Moved SaveImage-button from OpenGL-window to Control-window.

- Improved GUI slightly.
- Maximum resolution button now also applies to dimensions.
- Changed description of selected axis and changed selected axis at start-up to first one.

Ferret Histogram

- Added SaveImage-button.
- Fixed positions of histogram bars.
- Increased amount of memory used by ferret.

Scatterplot Matrix

- Improved GUI slightly.

---

back to main page

Programmer documentation

1. General design
2. Doxygen generated class, function and file documentation
3. Metadata
4. Goals of analysis
5. Implementation details
    1. OpenDX networks, macros and modules
    2. ParaView modules
    3. Avizo modules
6. ~~ToDoList~~
7. ~~Planned functions~~
8. View the visualization test run results.

For questions and hints contact Thomas Nocke, nocke@pik-potsdam.de

Programmer documentation - OpenDX networks, macros and modules

Networks

".net" file content the OpenDX networks. ".cfg" content the parameter dialogs for the network in the file of the same name before the dot.

- **1DTech.net / 1DTech.cfg** - implements the 1D visualization module
- **2DTech.net / 2DTech.cfg** - implements the 2D visualization module
- **2DFlowVis.net / 2DFlowVis.cfg** – implements the 2D flow visualization module
- **3DTech.net / 3DTech.cfg** - implements the 3D visualization module
- **histogram.net / histogram.cfg** - implements the Histogram module

Macros

- **AxesScale** (ScaleData.net, used in 1DTech.net, histogram.net) – Scales data logarithmic.
- **AddMissingMD** (AddMissingMDMakro.net, used in 3DTech.net) - Adds a variety of missing metadata to the input data.
- **AutoAxesReCalc** (AutoAxesReCalc.net, used in 2DTech.net) - This macro redefines the autoaxes functionality.
- **BarAnnotation** (BarAnnot.net, used in 1DTech.net) -
- **CheckCorners** (CheckCornersMakro.net, used in 1DTech.net) - Checks if there is a graph with all values equal.
- **ColorListMakro** (ColorListMakro.net, used in 1DTech.net) - Reads a List of Colors from a file.
- **compare** (compare.net, used in 3DTech.net) –
- **CreateCellCenteredData** (CreateCellCenteredDataMakro.net, used in 2DTech.net) - Maps Data to cell centers, works on regular grids only
- **CreateOptionDLGList** (CreateOptionDLGListMakro.net used in 2DTech.net) - Creates a list of all option dialog names and add the technique short name.
- **ExistsValidIsoline** (ExistsValidIsolineMacro.net used in 2DTech.net) – Checks if in a set of isolines real isolines exists or if they are all empty for the given values.
- **GenerateDescreteColorMap** (GenerateDescreteColorMap.net, used in 2DTech.net) - Generates a discrete ColorMap based on a continuous one.

- **GenerateDiscreteValuesAndLabels** (GenerateDiscreteValuesAndLabelsMakro.net, used in 1DTech.net) - Creates labels and values (by rounding) for discrete dimension tick locations.
- **IsolineCaption** (IsolineCaptionMacro.net, used in 2DTech.net) - Adds Values To isolines.
- **IsolineLegend** (IsolineLegendMakro.net, used in 2DTech.net) - Creates a Legend for the isovalues.
- **IsoSurfaceLegend** (IsoSurfaceLegend.net, used in 3DTech.net) - This macros constructs a non-interpolated legend for isosurfaces.
- **LineAndDotStyleMacro** (LineAndDotStyleMacro.net, used in 1DTech.net) - This macro defines the style of lines and dots in the display.
- **LineAndDotStyleMultiLineMacro** (LineAndDotStyleMultiLineMacro.net, used in 1DTech.net) - This macro defines the style of lines and dots in the display.
- LogarythmicDataTrafo (LogarythmicDataTrafoMacro.net used in ScaleData.net, MinMaxMacro.net) - Converts the data values of dependent variables to logarithmic scale.
- **MinMax** (MinMaxMacro.net, used in 1DTech.net, 2DTech.net, 3DTech.net) - Selection of MinMax values.
- **PlotLabelReCalc** (PlotLabelReCalc.net, used in 1DTech.net) - This macro redefines the autoaxes functionality.
- **Preprocessing** (PreprocessingMacro.net) in 1DTech.net, 2DTech.net, 3DTech.net, histogram.net) - Extracts metadata and cuts data range.
- **ReduceMetaData** (ReduceMetaData.net, used in 3DTech.net) - Reduces Attribute Metadata to the MD for the currently selected.
- **removeunuseddata** (rmveunusddata.net, used in 3DTech.net) - removes unused date with slab, include and transpose.
- **SingleLineStyleMacro** (SingleLineStyleMacro.net, used in 1DTech.net) - This macro adds color and style to a certain line.
- **Sphere** (sphere.net, used in 2DTech.net) - Transforms a field by warping onto a sphere.
- **StringListRemove** (StringListRemove.net, used in 2DTech.net, 3DTech.net) - Removes a string from a string list.
- **switch2colormaps** (switch2clrmps.net, used in 3DTech.net) - Switches between 2 color maps in dependency of the 2 input values.
- **WorldMap** (worldmap.net, used in 2DTech.net) - Create an object which is a map of the earth.

Modules

- **importnetcdf** - Imports the (SimEnv specific) data and metadata from netcdf files to the OpenDX data format.
- special: can handle subsets defined by a '-' sign in dimension name and index attributes and related dependent variables

---

back to main page

Implementation

Avizo state files are basically Amira scripts which makes adaption easier than in other programs (e.g. ParaView) where statefiles and scripts are two different things. So, in order to create a new Avizo technique, first, the user has to create the state that should serve as a template. The next step is to identify the places that will change with different input data (concrete technique, file name, minimum/maximum, etc.). At these places, a keyword has to be inserted (e.g. #var1_min for the minimum value of variable 1). Afterwards the prepareFile method in the Avizo class (avizovisualization.cpp) needs to be adapted in order to replace these keywords appropriatly.

---

back to main page