

# Magics support in CDO

---

Creating of contour, vector and graph plots  
January 2016

Kameswarrao Modali, Ralf Müller, Uwe Schulzweida  
*Max Planck Institute for Meteorology*

---

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Building CDO with Magics</b>	<b>4</b>
2.1	Magics . . . . .	4
2.2	Compilation . . . . .	4
2.3	Installation . . . . .	4
<b>3</b>	<b>Graphic reference manual</b>	<b>5</b>
3.0.1	MAGPLOT - Lat/Lon plot . . . . .	6
3.0.2	MAGVECTOR - Lat/Lon vector plot . . . . .	9
3.0.3	MAGGRAPH - Line graph plot . . . . .	11
	<b>Operator index</b>	<b>14</b>

# 1 Introduction

The Climate Data Operators (**CDO**) software is a collection of operators for standard processing of climate and forecast model data.

This document describes additional **CDO** operators to be used for generating plots.

Magics is the latest generation of the ECMWF's Meteorological plotting software MAGICS. Magics supports the plotting of contours, wind fields, observations, satellite images, symbols, text, axis and graphs (including box plots). Data fields to be plotted may be presented in various formats, for instance GRIB 1 and 2 code data, gaussian grid, regularly spaced grid and fitted data, BUFR and NetCDF format or retrieved from an ODB database. The produced meteorological plots can be saved in various formats, such as PostScript, EPS, PDF, GIF, PNG and SVG. [\[Magics\]](#)

In order to rapidly generate high quality pictures from the data obtained from the existing **CDO** operators, the **CDO** has been interfaced with the Magics library. As a first step, some **CDO** plotting operators are created to cater to the most essential/ frequently used plotting features viz., graph, contour, vector. These operators rely on the Magics and generate output files in the various formats supported by Magics. These operators can be used as terminal operators and chained with the existing operators.

Magics provides a vast number of parameters to control the attributes of various plotting features. Keeping in view, the usability of **CDO** users, currently only a few of these parameters are supported and accessible to the **CDO** users as command line arguments for the respective operators. The users are requested to refer to the Magics manual [\[Magics\]](#) for detailed description of the various parameters available for the various features. The description of the plotting operators and the various arguments that can be passed to these operators is provided in the subsequent sections.

## 2 Building CDO with Magics

This section describes how to build and install **CDO** with Magics support on a UNIX system.

### 2.1 Magics

Magics needs to be installed before building **CDO**. The Fortran and netCDF support can be disabled when building Magics from source.

### 2.2 Compilation

First go to the **CDO download** page (<https://code.zmaw.de/projects/cdo>) to get the latest distribution, if you do not have it yet. Compilation is done by performing the following steps:

1. Unpack the archive, if you haven't done that yet:

```
gunzip cdo-$VERSION.tar.gz    # uncompress the archive
tar xf cdo-$VERSION.tar       # unpack it
cd cdo-$VERSION
```

2. Configure **CDO** with Magics support:

```
./configure --with-magics=<Magics root directory>
```

For an overview of other configuration options use

```
./configure --help
```

3. Compile the program by running make:

```
make
```

The program should compile without problems and the binary (**cdo**) should be available in the **src** directory of the distribution.

### 2.3 Installation

After the compilation of the source code do a **make install**, possibly as root if the destination permissions require that.

```
make install
```

The binary is installed into the directory **<prefix>/bin**. **<prefix>** defaults to **/usr/local** but can be changed with the **--prefix** option of the configure script.

## 3 Graphic reference manual

This section gives a description of all **CDO** operators to generate plots. Related operators are grouped to modules. For easier description all single input files are named `infile` or `infile1`, `infile2`, etc., and an arbitrary number of input files are named `ifiles`. All output files are named `ofile` or `ofile1`, `ofile2`, etc.

Here is a short overview of all operators in this section:

<code>contour</code>	Contour plot
<code>shaded</code>	Shaded contour plot
<code>grfill</code>	Shaded gridfill plot
<code>vector</code>	Vector arrows plot
<code>graph</code>	Line graph plot

### 3.0.1 MAGPLOT - Lat/Lon plot

#### Synopsis

`<operator>,params ifile obase`

#### Description

The operators in this module generates 2D Lon/Lat plots. The data for the plot is read from `ifile`. Only data on rectilinear Lon/Lat grids are supported. The output file will be named `<obase>_<param>.<device>` where `param` is the parameter name and `device` is the device name. The default output file format is postscript, this can be changed with the device parameter. The type of the plot depends on the chosen operator.

Here is a list of all common plot parameters:

Keyname	Type	Description
device	STRING	Output device (ps, eps, pdf, png, gif, gif_animation, jpeg, svg, kml)
projection	STRING	Projection (cylindrical, polar_stereographic, robinson, mercator)
style	STRING	Contour line style (solid, dash, dot, chain_dash, chain_dot)
min	FLOAT	Minimum value
max	FLOAT	Maximum value
lon_max	FLOAT	Maximum longitude of the image
lon_min	FLOAT	Minimum longitude of the image
lat_max	FLOAT	Maximum latitude of the image
lat_min	FLOAT	Minimum latitude of the image
count	INTEGER	Number of Contour levels / Colour bands
interval	FLOAT	Interval in data units between two bands lines
list	INTEGER	List of levels to be plotted
RGB	STRING	TRUE or FALSE, to indicate, if the input colour is in RGB format
step_freq	INTEGER	Frequency of time steps to be considered for making the animation
		(device=gif_animation). Default value is "1" (all time steps).
		Will be ignored if input file has multiple variables.
file_split	STRING	TRUE or FALSE, to split the output file for each variable, if input has
		multiple variables. Default value is "FALSE". Valid only for "PS" format.

#### Operators

**contour** Contour plot

The operator **contour** generates the discrete contour lines of the input field values. The following additional parameters are valid for contour operator, module in addition to the common plot parameters:

Keyname	Type	Description
colour	STRING	Colour for drawing the contours
thickness	FLOAT	Thickness of the contour line
style	STRING	Line Style can be "SOLID", "DASH", "DOT", "CHAIN_DASH",
		"CHAIN_DOT"

**shaded** Shaded contour plot

The operator **shaded** generates the filled contours of the given input field values. The following additional parameters are valid for shaded contour and gridfill operator, in addition to the common plot parameters.

Keyname	Type	Description
colour_min	STRING	Colour for the Minimum colour band
colour_max	STRING	Colour for the Maximum colour band
colour_triad	STRING	Direction of colour sequencing for shading "CW" or "ACW", to denote "clockwise" and "anticlockwise" respectively.
		To be used in conjunction with "colour_min", "colour_max" options. Default is "ACW"
colour_table	STRING	File with user specified colours with the format as

Example file for 6 colours in RGB format:

```
6
RGB(0.0;0.0;1.0)
RGB(0.0;0.0;0.5)
RGB(0.0;0.5;0.5)
RGB(0.0;1.0;0.0)
RGB(0.5;0.5;0.0)
RGB(1.0;0.0;0.0)
```

### grfill

Shaded gridfill plot

The operator **grfill** is similar to satellite imaging and shades each cell (pixel) according to the value of the field at that cell.

## Parameter

*params*      STRING      Comma separated list of plot parameters

## Note

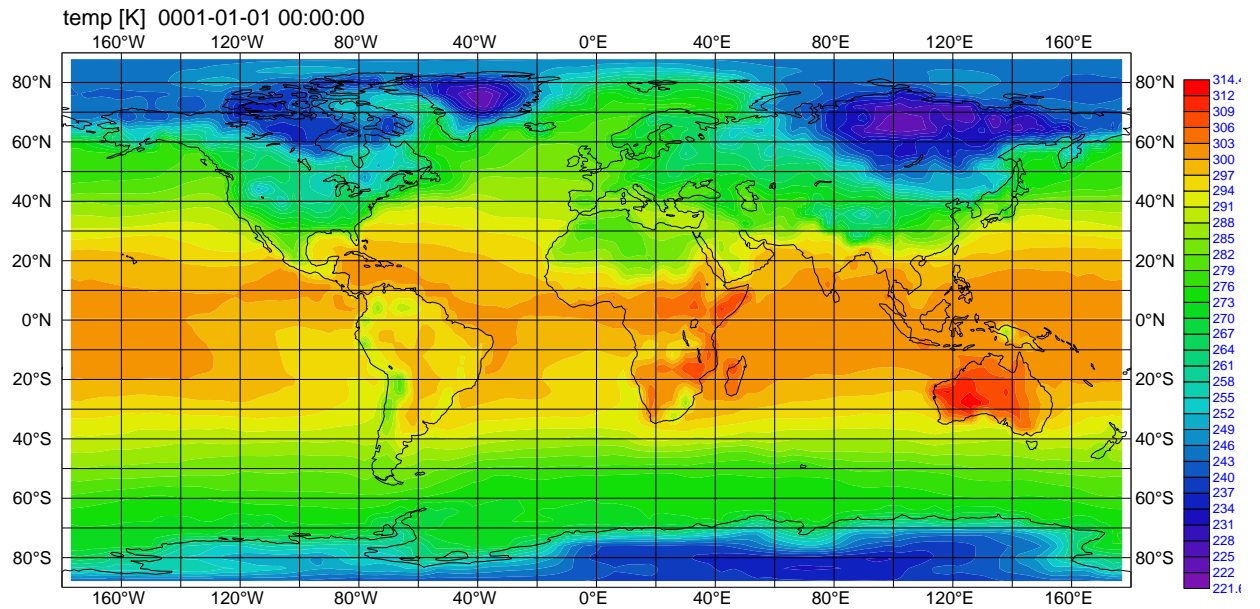
All colour parameter can be either standard name or in RGB format. The valid standard name strings for "colour" are:

"red", "green", "blue", "yellow", "cyan", "magenta", "black", "avocado", "beige", "brick", "brown", "burgundy", "charcoal", "chestnut", "coral", "cream", "evergreen", "gold", "grey", "khaki", "kellygreen", "lavender", "mustard", "navy", "ochre", "olive", "peach", "pink", "rose", "rust", "sky", "tan", "tangerine", "turquoise", "violet", "reddishpurple", "purplered", "purplishred", "orangishred", "redorange", "reddishorange", "orange", "yellowishorange", "orangeyellow", "orangishyellow", "greenishyellow", "yellowgreen", "yellowishgreen", "bluishgreen", "bluegreen", "greenishblue", "purplishblue", "bluepurple", "bluishpurple", "purple", "white"

## Example

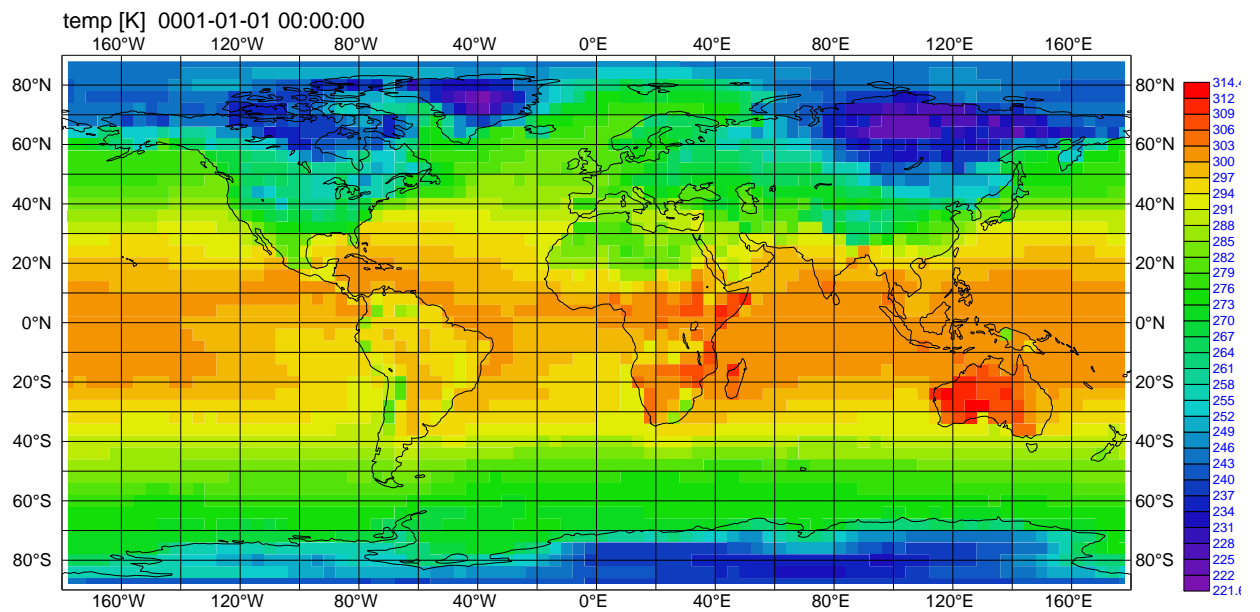
1) Shaded contour plot of a global temperature field with a resolution of 4 degree. The contour interval is 3 with a rainbow color table.

```
cdo shaded,interval=3,colour_min=violet,colour_max=red,colour_triad=cw temp plot
```



2) Shaded gridfill plot of a global temperature field with a resolution of 4 degree. The contour interval is 3 with a rainbow color table.

```
cdo grfill,interval=3,colour_min=violet,colour_max=red,colour_triad=cw temp plot
```





## 3.0.2 MAGVECTOR - Lat/Lon vector plot

### Synopsis

```
vector, params ifile obase
```

### Description

This operator generates 2D Lon/Lat vector plots. The data for the plot is read from `ifile`. The input is expected to contain two velocity components. Only data on rectilinear Lon/Lat grids are supported. The output file will be named `<obase>.<device>` where device is the device name. The default output file format is postscript, this can be changed with the device parameter.

Here is a list of all vector plot parameters:

Keyname	Type	Description
device	STRING	Output device (ps, eps, pdf, png, gif, gif_animation, jpeg, svg, kml)
projection	STRING	Projection (cylindrical, polar_stereographic, robinson, mercator)
thin_fac	FLOAT	Controls the actual number of wind arrows or flags plotted (default 2).
unit_vec	FLOAT	Wind speed in m/s represented by a unit vector (1.0cm)
step_freq	INTEGER	Frequency of time steps to be considered for making the animation
		(device=gif_animation). Default value is "1" (all time steps).
		Will be ignored if input file has multiple variables.

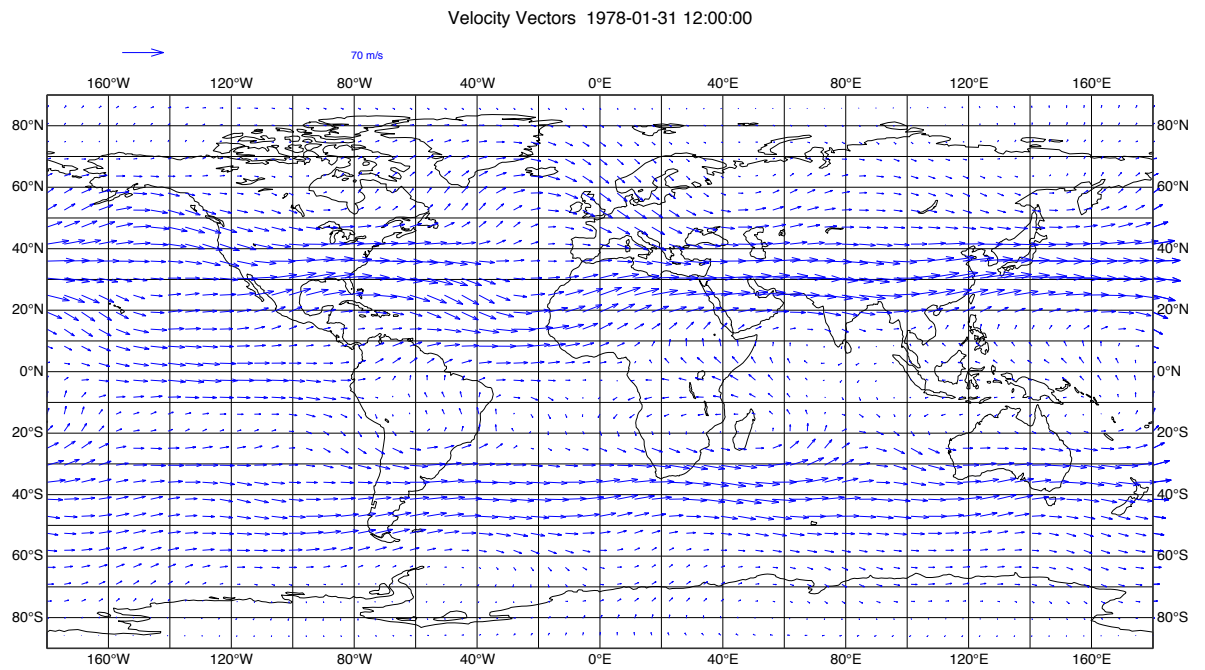
### Parameter

`params`      STRING      Comma separated list of plot parameters

### Example

Vector plot of global wind vectors with a resolution of 5 degree. The unit vector is set to 70 and all wind arrows are plotted.

```
cdo vector,thin_fac=1,unit_vec=70 uvdata plot
```



### 3.0.3 MAGGRAPH - Line graph plot

#### Synopsis

```
graph, params ifiles ofile
```

#### Description

This operator generates line graph plots. The data for the plot is read from **ifiles**. The result is written to **ofile**. The default output file format is postscript, this can be changed with the device parameter.

Here is a list of all graph plot parameters:

Keyname	Type	Description
device	STRING	Output device (ps, eps, pdf, png, gif, gif_animation, jpeg, svg, kml)
ymin	FLOAT	Minimum value of the y-axis data
ymax	FLOAT	Maximum value of the y-axis data
stat	STRING	"TRUE" or "FALSE", to switch on the mean computation. Default is "FALSE".
		Will be overridden to "FALSE", if input files have unequal number of time steps or different start/end times.
sigma	FLOAT	Standard deviation value for generating shaded back ground around the mean value.
		To be used in conjunction with 'stat="TRUE"'
obsv	STRING	To indicate if the input files have an observation data, by setting to "TRUE".
		Default value is "FALSE". The observation data should be the first file in the input file list. The observation data is always plotted in black colour.

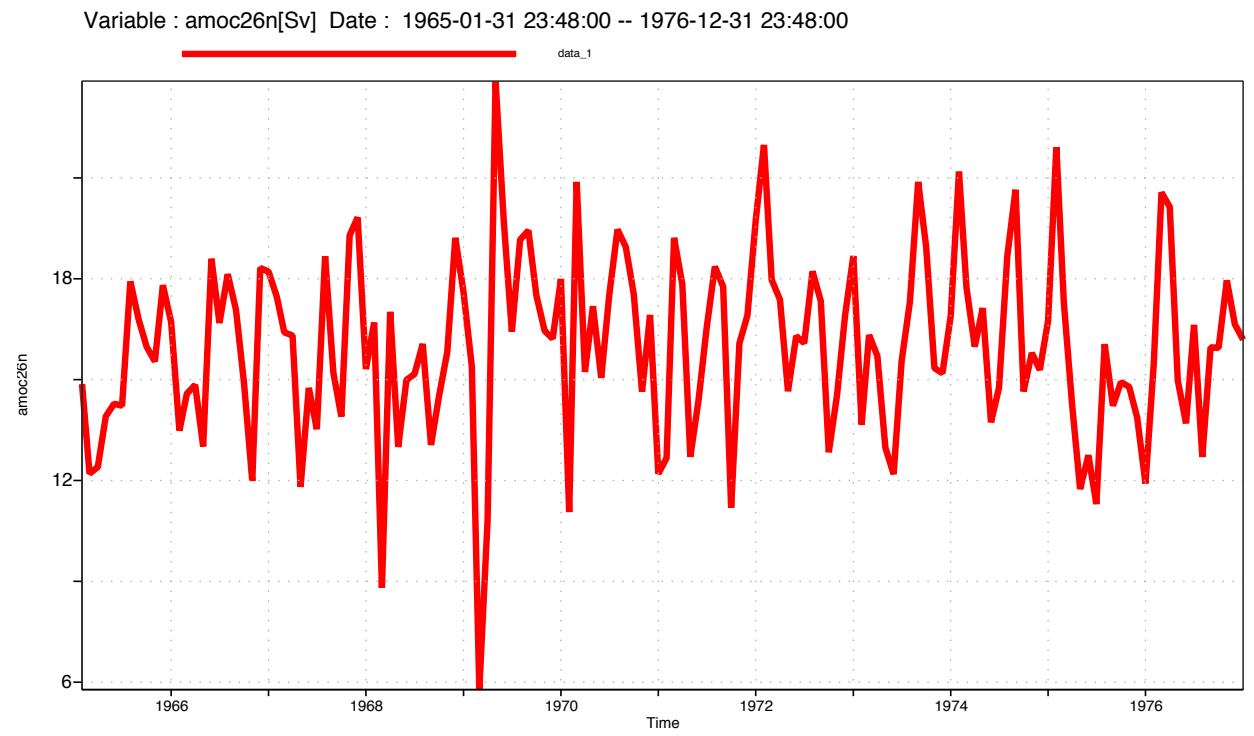
#### Parameter

*params*      STRING      Comma separated list of plot parameters

#### Example

Graph plot of an atlantic MOC time series from 1965 to 1976:

```
cdo graph amoc plot
```



# Bibliography

[CDI]

[Climate Data Interface](#), from the [Max Planck Institute for Meteorologie](#)

[CDO]

[Climate Data Operators](#), from the [Max Planck Institute for Meteorologie](#)

[Magics]

[Magics Software Package](#), from the [European Centre for Medium-Range Weather Forecasts \(ECMWF\)](#)

# Operator index

	<b>C</b>	
contour .....		<a href="#">6</a>
	<b>G</b>	
graph .....		<a href="#">11</a>
grfill .....		<a href="#">6</a>
	<b>S</b>	
shaded .....		<a href="#">6</a>
	<b>V</b>	
vector .....		<a href="#">9</a>