

# Best practices netCDF for viewing purposes

The netCDF format can be used in various ways. As data set it can be seen and used as the carrier of bulk information often with a time component. Within Deltare's netCDF's are used to store single data sets as well as time series of data set in scalars format as well as raster format. In a lot of cases netCDF files are handled as an archive of data, so the format does not really matter. But in case of viewing purposes there are some tips and tricks to be followed to be able to use netCDF's instantly. This wiki page describes the format that is preferably applied so that the data in the netCDF can be used in multiple ways.

## Geoserver and netCDF's

The main reason to use [GeoServer](#) as platform between data and visualization is to create WebServices from netCDF's, to style data and to enable the time variable so that the data in a netCDF can be animated in a Web interface. GeoServer can not handle netCDF files out of the box. You need to install an extension for GeoServer. These extensions can be found at the website of [boundlessgeo](#) (bear in mind that the extensions depend on the version of GeoServer that is used).

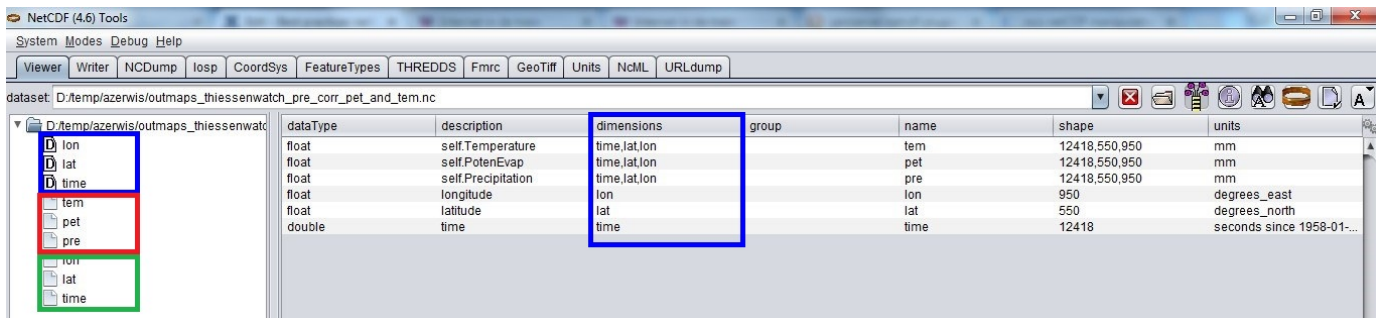
The GeoServer netCDF plugin [documentation](#) has some notes on supported netCDF's. It is quite clear that only netCDF's are supported with the following dimensions:

- lon
- lat
- (time)
- (z)

where dimensions between () are optional. The variables should have at least dimension lat, lon. If time and z are dimensions, the order is time, z, lat, lon.

The number of variables can vary from 1 to many as long as the dimensions are kept the same. This has implications in a lot of cases, especially where coordinates are not in lat/lon (WGS84) but in projections systems.

The picture below is a working example with time enable animation for several variables.



## Metadata

Metadata is always very important. The easiest way to get full metadata is in your script is using the template that [netCDF kickstarter](#) is providing you. In general, metadata definitions should be tested against standards. Available standards are the [ISO 19115](#), [19115-2](#) and [19119](#), which can be tested with [ncISO](#) software. In the geophysics sector, variables should also comply to the [CF](#) standards. Especially for time series, a user could benefit from a the use of an [unlimited time dimension](#).

## Matlab

It can be tricky getting data (e.g. model results) from a Matlab environment to be plotted on a Geoserver which communicates in Python. The main things to consider are:

- Always use the [coordinate system](#) WGS84 (EPSG:4326)
  - Coordinate conversion can be carried out with [convertCoordinates](#)
- The x and y axis both need to be monotonic increasing and in rows (i.e. no meshgrid)
- You define that the dimensions in the order longitude, latitude and time (no other order)
  - Therefore, the z axis or c values (colors) need to be same re-arranged so they match (function: [permute](#))

An example for Matlab can be found [here](#).