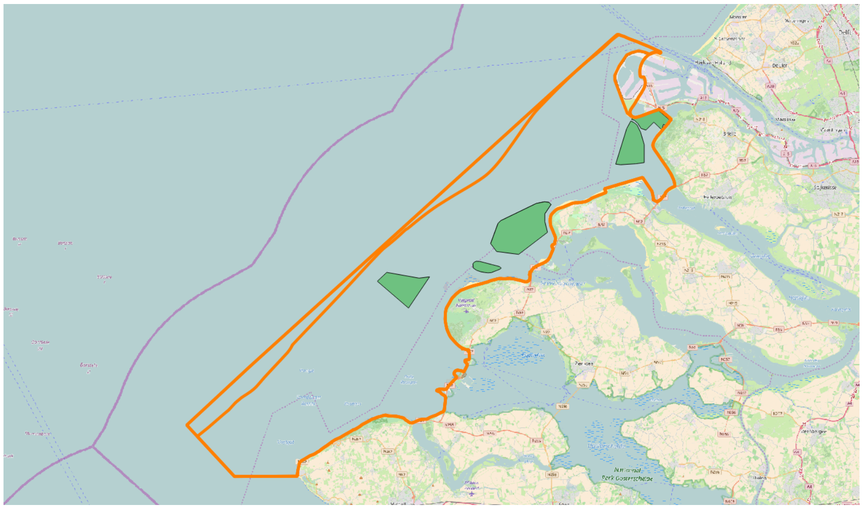
PMR-NCV

The project PMR-NCV is a large scale monitoring project in the Voordelta area



(see the orange shape in the picture. Several 100 thousands of observations has been done on various parameters. Following picture shows the variety of the domains observed.

Fish, birds, benthic species and also modelling has been carried out for the area. Observations on biota are available as a WMS and WFS via http://pmr-geoserver.deltares.nl/geoserver/web/, the model results are stored in rasters and made available via http://pmr-geoserver.deltares.nl/thredds/catalog.html.

The Modelresults consist of interpolated and anonimised VMS ship intensities and a variety of Delft3D output.

A small course in spring of 2018 has been given to make Deltares staff familiair with working with netCDF's via downloaded files, services and also to disseminate via geoserver. Below some code snippets can be found to get starting with the data. These codes snippets are in Python.

Code snippet downloaded files

```
import netCDF4
f = r"D:\pmr-ncv\abiotiekdata\_raster_2016_S_TBS_.nc"
ds = netCDF4.Dataset(f)
ds.dimensions.keys()
ds.variables.keys()
ar_lon = ds.variables['lon'][:]
ar_lat = ds.variables['lat'][:]
visint = ds.variables['Wdir'][:,:]
ds.close()
```

This first start gives insight in the structure of the netCDF. The comming code snippets support the extraction of data for a certain lon/lat.

Code snippet get index from lat lon arrays

```
def find_nearest(array, value):
    """Finds nearest value in array.
    taken from stack^ question 2566412"""
    idx = (np.abs(array - value)).argmin()
    return array[idx], int(idx)
```

Code snippet define coords

```
lon = 3.615
lat = 51.727
coords = [lon,lat]
```

If we combine this and the find_nearest function is called befor visint it is not necessary to load all the data.

Code snippet get index

```
idx = find_nearest(ar_lon, coords[0])
idy = find_nearest(ar_lat, coords[1])
visint = ds.variables[vars[2]][idx[1],idy[1]]
```

This should yield the value 1.0006409.

If the above is put in a loop over de number of years that are available in the THREDDS Data catalog the complete code looks like the following working example.

code snippet online data retrieval

```
import numpy as np
import netCDF4

def getvalue(f,coords):
    ds = netCDF4.Dataset(f,'r')
    ds.dimensions.keys()
    vars = ds.variables.keys()
    ar_lon = ds.variables['lon'][:]
    ar_lat = ds.variables['lat'][:]
    idx = find_nearest(ar_lon, coords[0])
    idy = find_nearest(ar_lat, coords[1])
    visint = ds.variables[vars[2]][idx[1],idy[1]]
    ds.close()
    return visint

def find_nearest(array, value):
    """Finds nearest value in array.
    taken from stack^ question 2566412"""
    idx = (np.abs(array - value)).argmin()
    return array[idx], int(idx)

lon = 3.615
lat = 51.727
coords = [lon,lat]

fpath = 'http://pmr-geoserver.deltares.nl/thredds/dodsC/PMR-NCV/vmsdata/_raster_year_S_TBS_.nc'
for yr in range(2003,2017,1):
    f =  fpath.replace('year',str(yr))
    print str(yr),getvalue(f,coords)
```