Determination of model setups:

Same atmospheric forcing: UERRA + corrections
Same river input: E-HYPE
Recommendation for Spinup: from 1961 to 01.07.2004, then restart from beginning

Horizontal resolution: preferable: max. 2 nm, mandatory: at least not coarser than 10 km
Vertical resolution: preferable: max. 2 m
Temporal coverage: as long as possible (Jan 1961 – Dec 2018; at least Jan 1978 – Dec 2017 (40 years))

Input data (model forcing):

Can be found at the IOW thredds server:
https://thredds-iow.io-warnemuende.de/thredds/catalogs/projects/bmip/catalog_bmip.html
Here you find three datasets for download:

1. River forcing (based on E-HYPE data), which is decribed in the following document in detail: "Germo Väli, H.E. Markus Meier, Manja Placke, Christian Dieterich: River runoff forcing for ocean modeling within the Baltic Sea Model Intercomparison Project. Meereswiss. Ber., Warnemünde, 113(2019), doi:10.12754/msr-2019-0113"
2. Atmospheric forcing: UERRA data from SMHI. **These require the following changes before used as model forcing:** Precipitation contains negative values which should be set to zero. Also, precipitation should be multiplied by a factor of 0.8 to correct for a wet bias.
3. Cloudiness: Since UERRA cloudiness is corrupted, we suggest to use coastDat-2 cloudiness for those models which require it as a forcing. Please cite the dataset like this: "Geyer, B., 2014. High-resolution atmospheric reconstruction for Europe 1948-2012: coastDat2. Earth System Science Data 6, 147."

All but the river forcing data require a password. The username is "bmip" and the password is available upon request from Markus Meier, IOW.
So, to access the data, you can e.g. use the following command:

```
ncdump -h "https://bmip:PASSWORD@thredds-iow.io-warnemuende.de/thredds/dodsC/BMIP/IOW-THREDDS-BMIP_uerra_forcing_2019-10-21-11-lwdrad.nc"
```

It is important that the software you use is linked to a new NetCDF version (from 4.6 on) since otherwise the https connection will fail.

Data output in NetCDF files:

The following table lists the output variables you should write out. Red means that here something is important and not obvious. Yellow means that names which do not follow the cdf conventions had to be used.
For some data we state "separate algorithm attached". We provide these algorithms as well as the table with the coordinates of stations, transects and upwelling areas under the following link:
https://owncloud.io-warnemuende.de/index.php/s/LVZbDvSvcTnECpb
The password is the same.

Planned data output		Variable name	Description	Dimension		Unit	Time step	http://cfconventions.org/cf-conventions/cf-conventions.html, http://cfconventions.org/Data/cf-standard-names/60/build/cf-standard-name-table.html
Time axis	Time	time	Time axis in hours, days or months	1D		days	Dependent on variable	double time(time) ; time:long_name = "time" ; time:units = "days since 1900-01-01 00:00:00" ; time:calendar = "gregorian"
Model grid	Latitude	lat	Latitude range of model domain	1D		degree	-	double lat(lat) ; lat:long_name = "latitude" ; lat:units = "degrees_north" ; lat:standard_name = "latitude" ; lat:unit_long = "Degrees North" ; lat:axis = "Y" ;
	Longitude	lon	Longitude range of model domain	1D		degree	-	double lon(lon) ; lon:long_name = "longitude" ; lon:units = "degrees_east" ; lon:standard_name = "longitude" ; lon:unit_long = "Degrees East" ; lon:axis = "X" ;
	Depth	depth	Depth	1D		m	-	double depth(depth) ; depth:long_name = "depth" ; depth:standard_name = "depth" ; depth:units = "meters" ; depth:positive = "down" ; depth:unit_long = "meters" ; depth:axis = "Z"
	Area	areao	Area of the ocean grid cell	2D	lat, lon	m²	-	float areao(lat, lon) ; areao:standard_name = "sea_area" ; areao:long_name = "Sea area" ; areao:units = "m2" ; areao:unit_long = "square meters" ; areao:valid_range = -3.f, 40.f ; areao:_FillValue = -999.f ; areao:missing_value = -999.f ;
Hydrographic variables at tide gauge stations **TABLE ATTACHED** lat, lon are singleton dimensions	Sea level anomaly	sla	Sea level anomaly	1D	time, lat, lon	m	hourly averages	float sla(time, lat, lon) ; sla:standard_name = "sea_surface_height_above_sea_level" ; sla:long_name = "Sea level anomalies" ; sla:units = "m" ; sla:unit_long = "meters" ; sla:valid_range = -10.f, 10.f ; sla:_FillValue = -999.f ; sla:missing_value = -999.f ;

Hydrographic variables at representative stations
 	
TABLE ATTACHED
 	
lat, lon are singleton dimensions
	Potential Temperature	thetao	Temperature of sea water	2D	time, depth, lat, lon	degree Celsius	At least daily averages	`float thetao(time, depth, lat, lon) ;`
								`thetao:standard_name = "sea_water_potential_temperature" ;`
								`thetao:long_name = "Potential temperature" ;`
								`thetao:units = "degrees_C" ;`
								`thetao:unit_long = "degree Celsius" ;`
								`thetao:valid_range = -3.f, 40.f ;`
								`thetao:_FillValue = -999.f ;`
								`thetao:missing_value = -999.f ;`

	Salinity	so	Salinity of sea water	2D	time, depth, lat, lon	g kg^{-1}	At least daily averages	`float so(time, depth, lat, lon) ;`
								`so:standard_name = "sea_water_salinity" ;`
								`so:long_name = "Salinity" ;`
								`so:units = "0.001" ;`
								`so:unit_long = "practical salinity unit" ;`
								`so:valid_range = 0.f, 40.f ;`
								`so:_FillValue = -999.f ;`
								`so:missing_value = -999.f ;`

	Zonal current velocity	uo	Ocean current in x-direction	2D	time, depth, lat, lon	m s^{-1}	At least daily averages	`float uo(time, depth, lat, lon) ;`
								`uo:standard_name = "eastward_sea_water_velocity" ;`
								`uo:long_name = "Eastward current" ;`
								`uo:units = "m s-1" ;`
								`uo:unit_long = "meters per second" ;`
								`uo:_FillValue = -999.f ;`
								`uo:missing_value = -999.f ;`

	Meridional current velocity	vo	Ocean current in y-direction	2D	time, depth, lat, lon	m s^{-1}	At least daily averages	`float vo(time, depth, lat, lon) ;`
								`vo:standard_name = "northward_sea_water_velocity" ;`
								`vo:long_name = "Northward current" ;`
								`vo:units = "m s-1" ;`
								`vo:unit_long = "meters per second" ;`
								`vo:_FillValue = -999.f ;`
								`vo:missing_value = -999.f ;`

	Age of water	age	Age of water since surface contact	2D	time, depth, lat, lon	s	At least daily averages	`float age(time, depth, lat, lon) ;`
								`age:standard_name = "sea_water_age_since_surface_contact" ;`
								`age:long_name = "Sea water age since surface contact" ;`
								`age:units = "s" ;`
								`age:unit_long = "seconds" ;`
								`age:_FillValue = -999.f ;`
								`age:missing_value = -999.f ;`

Horizontal 2D fields	Sea surface temperature	surfaceT	Sea surface temperature	3D	time, lat, lon	degree Celsius	Daily averages	`float surfaceT(time, lat, lon) ;`
								`surfaceT:standard_name = "sea_surface_temperature" ;`
								`surfaceT:long_name = "Sea surface temperature" ;`
								`surfaceT:units = "degrees_C" ;`
								`surfaceT:unit_long = "degree Celsius" ;`
								`surfaceT:valid_range = -3.f, 40.f ;`
								`surfaceT:_FillValue = -999.f ;`
								`surfaceT:missing_value = -999.f ;`

	Sea floor potential temperature	bottomT	Sea floor temperature	3D	time, lat, lon	degree Celsius	Daily averages	`float bottomT(time, lat, lon) ;`
								`bottomT:standard_name = "sea_water_potential_temperature_at_sea_floor" ;`
								`bottomT:long_name = "Sea water potential temperature at sea floor" ;`
								`bottomT:units = "degrees_C" ;`
								`bottomT:unit_long = "degree Celsius" ;`
								`bottomT:valid_range = -3.f, 40.f ;`
								`bottomT:_FillValue = -999.f ;`
								`bottomT:missing_value = -999.f ;`

	Sea surface salinity	surfaceS	Sea surface salinity	3D	time, lat, lon	g kg^{-1}	Daily averages	`float surfaceS(time, lat, lon) ;`
								`surfaceS:standard_name = "sea_surface_salinity" ;`
								`surfaceS:long_name = "Sea surface salinity" ;`
								`surfaceS:units = "0.001" ;`
								`surfaceS:unit_long = "practical salinity unit" ;`
								`surfaceS:valid_range = 0.f, 40.f ;`
								`surfaceS:_FillValue = -999.f ;`
								`surfaceS:missing_value = -999.f ;`

	Sea floor salinity	bottomS	Sea floor salinity	3D	time, lat, lon	g kg^{-1}	Daily averages	`float bottomS(time, lat, lon) ;`
								`bottomS:standard_name = "sea_water_salinity_at_sea_floor" ;`
								`bottomS:long_name = "Sea water salinity at sea floor" ;`
								`bottomS:units = "0.001" ;`
								`bottomS:unit_long = "practical salinity unit" ;`
								`bottomS:valid_range = 0.f, 40.f ;`
								`bottomS:_FillValue = -999.f ;`
								`bottomS:missing_value = -999.f ;`

	Bottom age of water	bottomAge	the length of time elapsed since the sea water in a grid cell was last in the surface level	3D	time, lat, lon	s	Daily averages	`float bottomAge(time, lat, lon) ;`
								`bottomAge:standard_name = "sea_water_age_since_surface_contact_at_sea_floor" ;`
								`bottomAge:long_name = "Sea water age since surface contact at sea floor" ;`
								`bottomAge:units = "s" ;`
								`bottomAge:unit_long = "seconds" ;`
								`bottomAge:_FillValue = -999.f ;`
								`bottomAge:missing_value = -999.f ;`

	Thermocline depth	td	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	m	Daily averages, or calculated from daily S and T averages	`float td(time, lat, lon) ;`
								`ts:standard_name = "thermocline_depth" ;`
								`ts:long_name = "Thermocline depth" ;`
								`ts:units = "m" ;`
								`ts:unit_long = "meters" ;`
								`ts:_FillValue = -999.f ;`
								`ts:missing_value = -999.f ;`

	Thermocline strength	tg	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	K/m	Daily averages, or calculated from daily S and T averages	float tg(time, lat, lon) ; tg:standard_name = "thermocline_gradient" ; tg:long_name = "Thermocline gradient" ; tg:units = "K/m" ; tg:unit_long = "Kelvin per meter" ; tg:_FillValue = -999.f ; tg:missing_value = -999.f ;
	Halocline depth	sd	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	m	Daily averages, or calculated from daily S and T averages	float sd(time, lat, lon) ; sd:standard_name = "halocline_depth" ; sd:long_name = "Halocline depth" ; sd:units = "m" ; sd:unit_long = "meters" ; sd:_FillValue = -999.f ; sd:missing_value = -999.f ;
	Halocline strength	sg	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	g/kg/m	Daily averages, or calculated from daily S and T averages	float sg(time, lat, lon) ; sg:standard_name = "halocline_gradient" ; sg:long_name = "Halocline gradient" ; sg:units = "g/kg/m" ; sg:unit_long = "grams per kilogram and meter" ; sg:_FillValue = -999.f ; sg:missing_value = -999.f ;
	Pycnocline depth	rd	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	M	Daily averages, or calculated from daily S and T averages	float rd(time, lat, lon) ; rd:standard_name = "pycnocline_depth" ; rd:long_name = "Pycnocline depth " ; rd:units = "m" ; rd:unit_long = "meters" ; rd:_FillValue = -999.f ; rd:missing_value = -999.f ;
	Pycnocline strength	rg	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	kg/m4	Daily averages, or calculated from daily S and T averages	float rg(time, lat, lon) ; rg:standard_name = "pycnocline_gradient " ; rg:long_name = "Pycnocline gradient " ; rg:units = "kg/m4" ; rg:unit_long = "kilograms per cubic meter and meter" ; rg:_FillValue = -999.f ; rg:missing_value = -999.f ;
	Thermocline depth based on thermal density gradient	rtd	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	m	Daily averages, or calculated from daily S and T averages	float rtd(time, lat, lon) ; rtd:standard_name = "thermocline_depth_based_on_density" ; rtd:long_name = "Thermocline depth based on density" ; rtd:units = "m" ; rtd:unit_long = "meters" ; rtd:_FillValue = -999.f ; rtd:missing_value = -999.f ;
	Thermocline depth based on thermal density gradient	rtg	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	kg/m4	Daily averages, or calculated from daily S and T averages	float rtg(time, lat, lon) ; rtg:standard_name = "thermocline_gradient_based_on_density" ; rtg:long_name = "Thermocline gradient based on density" ; rtg:units = "kg/m4" ; rtg:unit_long = "kilograms per cubic meter and meter" ; rtg:_FillValue = -999.f ; rtg:missing_value = -999.f ;

	Halocline depth based on haline density gradient	rsd	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	m	Daily averages, or calculated from daily S and T averages	float rsd(time, lat, lon) ; rsd:standard_name = "halocline_depth_based_on_density" ; rsd:long_name = "Halocline depth based on density" ; rsd:units = "m" ; rsd:unit_long = "meters" ; rsd:_FillValue = -999.f ; rsd:missing_value = -999.f ;
	Halocline depth based on haline density gradient	rsg	SEPARATE ALGORITHM ATTACHED	3D	time, lat, lon	kg/m4	Daily averages, or calculated from daily S and T averages	float rsg(time, lat, lon) ; rsg:standard_name = "halocline_gradient_based_on_density" ; rsg:long_name = "Halocline gradient based on density" ; rsg:units = "kg/m4" ; rsg:unit_long = "kilograms per cubic meter and meter" ; rsg:_FillValue = -999.f ; rsg:missing_value = -999.f ;
	Sea-ice thickness	sithick	Mean of sea-ice thickness over all area except the open ocean fraction of the grid cell	3D	time, lat, lon	m	Daily averages	float sithick(time, lat, lon) ; sithick:standard_name = "sea_ice_thickness" ; sithick:long_name = "Sea ice thickness" ; sithick:units = "m" ; sithick:unit_long = "meters" ; sithick:_FillValue = -999.f ; sithick:missing_value = -999.f ;
	Sea-ice mass	simass	Mass of sea ice in a horizontal grid cell (not divided by cell area)	3D	time, lat, lon	kg	Daily averages	float simass(time, lat, lon) ; simass:standard_name = "sea_ice_mass" ; simass:long_name = "Sea ice mass" ; simass:units = "kg" ; simass:unit_long = "kilograms" ; simass:_FillValue = -999.f ; simass:missing_value = -999.f ;
	Sea level anomalies	sla	Water level of the ocean	3D	time, lat, lon	m	Daily averages	float sla(time, lat, lon) ; sla:standard_name = "sea_surface_height_above_sea_level" ; sla:long_name = "Sea level anomalies" ; sla:units = "m" ; sla:unit_long = "meters" ; sla:valid_range = -10.f, 10.f ; sla:_FillValue = -999.f ; sla:missing_value = -999.f ;
	Vertically averaged salinity	Svave	Vertical integral of salinity divided by water depth	3D	time, lat, lon	g kg^{-1}	Daily averages	float Svave(time, lat, lon) ; Svave:standard_name = "vertically_averaged_salinity" ; Svave:long_name = "Vertically averaged salinity" ; Svave:units = "0.001" ; Svave:unit_long = "practical salinity unit" ; Svave:valid_range = 0.f, 40.f ; Svave:_FillValue = -999.f ; Svave:missing_value = -999.f ;

Horizontally integrated fields	Sea ice extent	siextent	Sum over the total areas of grid cells filled with >=10% ice on daily average To follow CF Conventions, the threshold must be specified by supplying a scalar coordinate variable with the standard name of sea_ice_area_fraction.	1D	time	m^2	Daily averages	float siextent(time) ; siextent:standard_name = "sea_ice_extent " ; siextent:long_name = "Sea ice extent" ; siextent:units = "m2" ; siextent:unit_long = "square meters" ; siextent:_FillValue = -999.f ; siextent:missing_value = -999.f ;
	Overturning circulation	oc	SEPARATE ALGORITHM ATTACHED	3D	Time, depth, along-thalweg grid cell	m^3/s	Daily averages	float oc(time) ; oc:standard_name = "overturning_circulation " ; oc:long_name = "Overturning ciculation" ; oc:units = "m3/s" ; oc:unit_long = "cubic meters per second" ; oc:_FillValue = -999.f ; oc:missing_value = -999.f ;
Transects **TABLE ATTACHED** One of the lat, lon is a singleton dimension	Potential Temperature	thetao	Temperature of sea water	3D	time, depth, lat, lon	degree Celsius	Daily averages	float thetao(time, depth, lat, lon) ; thetao:standard_name = "sea_water_potential_temperature" ; thetao:long_name = "Potential temperature" ; thetao:units = "degrees_C" ; thetao:unit_long = "degree Celsius" ; thetao:valid_range = -3.f, 40.f ; thetao:_FillValue = -999.f ; thetao:missing_value = -999.f ;
	Salinity	so	Salinity of sea water	3D	time, depth, lat, lon	g kg^{-1}	Daily averages	float so(time, depth, lat, lon) ; so:standard_name = "sea_water_salinity" ; so:long_name = "Salinity" ; so:units = "0.001" ; so:unit_long = "practical salinity unit" ; so:valid_range = 0.f, 40.f ; so:_FillValue = -999.f ; so:missing_value = -999.f ;

	Zonal volume transport	uuo	Eastward ocean current (in x-direction) multiplied by vertical cell thickness	3D	time, depth, lat, lon	m2 s^{-1}	Daily averages	float uuo(time, depth, lat, lon) ; uuo:standard_name = "eastward_sea_water_velocity_times_cell_thickness" ; uuo:long_name = "Eastward sea water velocity times cell thickness" ; uuo:units = "m2 s-1" ; uuo:unit_long = "square meters per second" ; uuo:_FillValue = -999999.f ; uuo:missing_value = -999999.f ;
	Meridional volume transport	vvo	Northward ocean current (in x-direction) multiplied by vertical cell thickness	3D	time, depth, lat, lon	m2 s^{-1}	Daily averages	float vvo(time, depth, lat, lon) ; vvo:standard_name = "eastward_sea_water_velocity_times_cell_thickness" ; vvo:long_name = "Eastward sea water velocity times cell thickness" ; vvo:units = "m2 s-1" ; vvo:unit_long = "square meters per second" ; vvo:_FillValue = -999999.f ; vvo:missing_value = -999999.f ;
	Grid cell thickness	h	Vertical extent of the grid cell	3D	time, depth, lat, lon	m	Daily averages	float h(time, depth, lat, lon) ; h:standard_name="magnitude_of_derivative_of_position_wrt_model_level_number" ; h:long_name = "Vertical thickness of grid cell" ; h:units = "m" ; h:unit_long = "meters" ; h:_FillValue = -999.f ; h:missing_value = -999.f ;
	Age of water	age	Age of water since surface contact	3D	time, depth, lat, lon	s	Daily averages	float age(time, depth, lat, lon) ; age:standard_name = "sea_water_age_since_surface_contact" ; age:long_name = "Sea water age since surface contact" ; age:units = "s" ; age:unit_long = "seconds" ; age:_FillValue = -999.f ; age:missing_value = -999.f ;
(longitudinal transects only)	Zonal ocean heat transport	xtransT	Heat transport density multiplied by vertical cell thickness	3D	time, depth, lat, lon	W/m	Daily averages	float xtransT(time, depth, lat, lon) ; xtransT:standard_name = "eastward_ocean_heat_transport_divided_by_distance" ; xtransT:long_name = "Eastward ocean heat transport divided by distance" ; xtransT:units = "W m-1" ; xtransT:unit_long = "Watts per meter" ; xtransT:_FillValue = -1.e20f ; xtransT:missing_value = -1.e20f ;
(longitudinal transects only)	Zonal ocean salt transport	xtransS	Salt transport density multiplied by vertical cell thickness	3D	time, depth, lat, lon	kg m-1 s^{-1}	Daily averages	float xtransS(time, depth, lat, lon) ; xtransS:standard_name = "eastward_ocean_salt_transport_divided_by_distance" ; xtransS:long_name = "Eastward ocean salt transport divided by distance" ; xtransS:units = "kg m-1 s-1" ; xtransS:unit_long = "kilograms per meter and second" ; xtransS:_FillValue = -1.e20f ; xtransS:missing_value = -1.e20f ;

(latitudinal transects only)	Meridional ocean heat transport	ytransT	Heat transport density multiplied by vertical cell thickness	3D	time, depth, lat, lon	W/m	Daily averages	float ytransT(time, depth, lat, lon) ; ytransT:standard_name = "northward_ocean_heat_transport_divided_by_distance" ; ytransT:long_name = "Northward ocean heat transport divided by distance" ; ytransT:units = "W m-1" ; ytransT:unit_long = "Watts per meter" ; ytransT:_FillValue = -1.e20f ; ytransT:missing_value = -1.e20f ;
(latitudinal transects only)	Meridional ocean salt transport	ytransS	Salt transport density multiplied by vertical cell thickness	3D	time, depth, lat, lon	kg m-1 s^{-1}	Daily averages	float ytransS(time, depth, lat, lon) ; ytransS:standard_name = "northward_ocean_salt_transport_divided_by_distance" ; ytransS:long_name = "Northward ocean salt transport divided by distance" ; ytransS:units = "kg m-1 s-1" ; ytransS:unit_long = "kilograms per meter and second" ; ytransS:_FillValue = -1.e20f ; ytransS:missing_value = -1.e20f ;
Upwelling areas **TABLE ATTACHED** From 1990 on only	Potential Temperature	thetao	Temperature of sea water	3D	time, depth, lat, lon	degree Celsius	3-hourly averages	float thetao(time, depth, lat, lon) ; thetao:standard_name = "sea_water_potential_temperature" ; thetao:long_name = "Potential temperature" ; thetao:units = "degrees_C" ; thetao:unit_long = "degree Celsius" ; thetao:valid_range = -3.f, 40.f ; thetao:_FillValue = -999.f ; thetao:missing_value = -999.f ;
	Salinity	so	Salinity of sea water	3D	time, depth, lat, lon	g kg^{-1}	3-hourly averages	float so(time, depth, lat, lon) ; so:standard_name = "sea_water_salinity" ; so:long_name = "Salinity" ; so:units = "0.001" ; so:unit_long = "practical salinity unit" ; so:valid_range = 0.f, 40.f ; so:_FillValue = -999.f ; so:missing_value = -999.f ;
	Zonal volume transport	uuo	Eastward ocean current (in x-direction) multiplied by vertical cell thickness	3D	time, depth, lat, lon	m2 s^{-1}	3-hourly averages	float uuo(time, depth, lat, lon) ; uuo:standard_name = "eastward_sea_water_velocity_times_cell_thickness" ; uuo:long_name = "Eastward sea water velocity times cell thickness" ; uuo:units = "m2 s-1" ; uuo:unit_long = "square meters per second" ; uuo:_FillValue = -999999.f ; uuo:missing_value = -999999.f ;

	Meridional volume transport	vvo	Northward ocean current (in x-direction) multiplied by vertical cell thickness	3D	time, depth, lat, lon	m2 s^{-1}	3-hourly averages	float vvo(time, depth, lat, lon) ; vvo:standard_name = "eastward_sea_water_velocity_times_cell_thickness" ; vvo:long_name = "Eastward sea water velocity times cell thickness" ; vvo:units = "m2 s-1" ; vvo:unit_long = "square meters per second" ; vvo:_FillValue = -999999.f ; vvo:missing_value = -999999.f ;
	Grid cell thickness	h	Vertical extent of the grid cell	3D	time, depth, lat, lon	m	3-hourly averages	float h(time, depth, lat, lon) ; h:standard_name="magnitude_of_derivative_of_position_wrt_model_level_number" ; h:long_name = "Vertical thickness of grid cell" ; h:units = "m" ; h:unit_long = "meters" ; h:_FillValue = -999.f ; h:missing_value = -999.f ;
	Age of water	age	Age of water since surface contact	3D	time, depth, lat, lon	s	Daily averages	float age(time, depth, lat, lon) ; age:standard_name = "sea_water_age_since_surface_contact" ; age:long_name = "Sea water age since surface contact" ; age:units = "s" ; age:unit_long = "seconds" ; age:_FillValue = -999.f ; age:missing_value = -999.f ;