Interface Control Document

Ocean Heat Flux

Reference D-80
Issue 1.0
Date of Issue 2 Dec 2014

Ifremer, Brest
Technopôle Brest-Iroise, Plouzané, 29280, France
Tel: +33 (0)2 98 22 43 11   Fax: +33 (0)2 98 22 45 33
www.ifremer.fr

Document Signature Table				
Author	J-F Piollé	IFREMER	JF Piollé	
Approval	Pierre Philippe Mathieu	ESA		

Document Change Record			
Issue	**Date**	**Type**	**Change Description**
1.0	2/12/2014	Initial draft	

Table of Contents

1 INTRODUCTION

1.1 SCOPE OF THIS DOCUMENT

This is the Interface Control Document Document [ICD] (deliverable D-80) for the OceanHeatFlux project. The purpose of this ICD is to describe the interfaces of the system, including :

- format of OceanHeatFlux products
- access to OceanHeatFlux products
- interfaces of the processors to be integrated into OceanHeatFlux system

1.2 PURPOSE OF THIS DOCUMENT

The purpose of the ICD is to describe the external interfaces of the system, including :

1. actors and remote services interacting with OceanHeatFlux
2. protocols to access source EO datasets
3. format of OceanHeatFlux output products

1.3 STRUCTURE OF THIS DOCUMENT

The document is structured as follows:

1. Section 1 (this section) the introduction gives an overview of the document aims and structure.
2. Section 2 gives an overview of OceanHeatFlux system and its main external interfaces
3. Section 3 gives the format of OceanHeatFlux data exposed to users

1.4 TERMINOLOGY

For a better reading, the following table defines a list of terms commonly used in the rest of the document.

term	**Definition**	**synonyms**

Table 1 Terminology used in this document

1.5 REFERENCE AND APPLICABLE DOCUMENTS

[AD-1] Statement of Work for DUE OceanHeatFlux project (SoW)

2

1.6 ACRONYMS

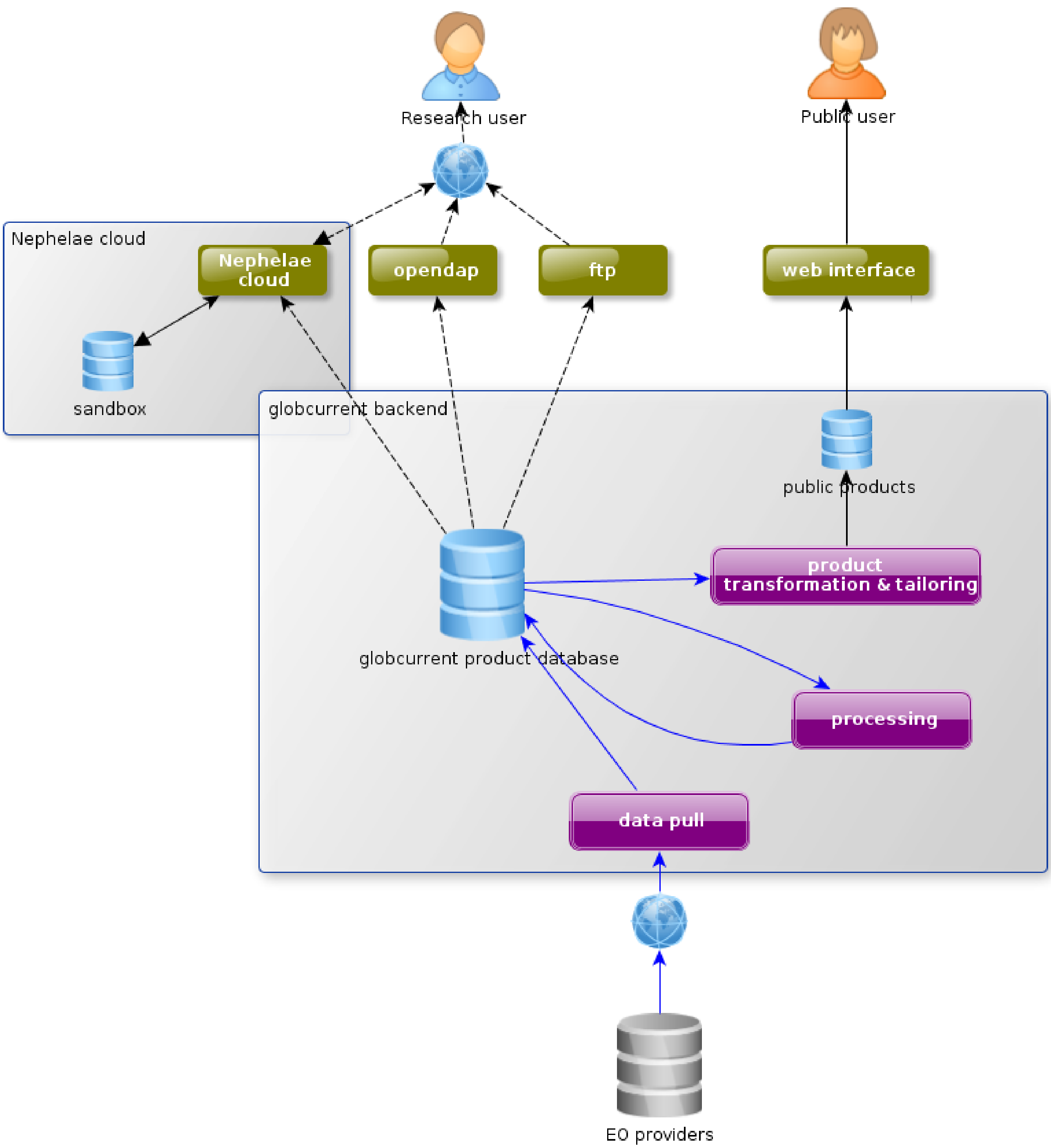
AATSR	Advanced Along Track Scanning Radiometer (ESA instrument)
AMSRE	Advanced Microwave Scanning Radiometer – E (of NASA's EoS Aqua)
API	Application Programming Interface
ATSR-1	Along Track Scanning Radiometer onboard ERS-1 (ESA instrument)
ATSR-2	Along Track Scanning Radiometer onboard ERS-2 (ESA instrument)
AMSR-E	Advanced Microwave Scanning Radiometer for EOS (NASA instrument)
ASAR	Advanced Synthetic Aperture Radar
ASCAT	Advanced SCATterometer (of MetOp)
AVHRR	Advanced Very High Resolution Radiometer (NOAA instruments)
CDR	Critical Design Review
CEOS	Committee on Earth Observation Satellites
CERSAT	Centre de Recherche et d'Exploitation Satellitaire (Ifremer Satellite Data Center)
DARD	Data Access and Requirements Document
DVP	Development and Validation Plan
ECMWF	European Centre for Medium-Range Weather Forecasts
ENVISAT	Environment Satellite
EO	Earth observation
EOS	Earth Observing System
ERS	European Remote Sensing satellite (ESA instrument)
ESA	European Space Agency
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
FOAM	Forecast Ocean Assimilation Model
FR	Final Report
FP	Final Presentation
FTP	File transfer protocol
GHRSST	Group for High Resolution Sea Surface Temperature
GMES	Global Monitoring for Environment and Security
Hs	Significant Wave Height (also SWH)
ICD	Interface Control Document
IFREMER	Institut Français de Recherche pour l'Exploitation de la Mer
ITT	Invitation To Tender
Jason-1	Altimetry mission (NASA/France instrument)
Jason-2	Altimetry mission (NASA/France instrument)
KO	Kick Off
MERIS	Medium Resolution Imaging Spectrometer (ESA instrument)
MODIS	Moderate Resolution Imaging Spectrometer (NASA instrument)

NASA	National Aeronautics and Space Administration (US)
NetCDF	Network Common Data Form
NetCDF CF	NetCDF Climate and Forecast Metadata Convention
NOAA	National Oceanographic and Atmospheric Administration (US)
NOC	National Oceanography Centre (UK)
NPOESS	National Polar-orbiting Operational Environmental Satellite System
NRT	Near Real Time
NWP	Numerical Weather Prediction
OC	Ocean colour
OPeNDAP	Open-source Project for a Network Data Access Protocol
OSTIA	Operational Sea Surface Temperature and Sea Ice Analysis (UK Meteorological Office)
PML	Plymouth Marine Laboratory
PR	Progress Report
RB	Requirements Baseline
RD	Reference Document
RRS	Remote Sensing Reflectance
RUG	Reference User Group
SaaS	Software as a Service
SAR	Synthetic Aperture RADAR
SeaWIFS	Sea-viewing Wide Field-of-view Sensor
SEVIRI	Spinning Enhanced Visible and Infrared Imager (of Meteosat Second Generation)
SRR	System Requirements Review
SoW	HR-DD Statement of Work
SRR	System Requirements Review
SSH	Sea Surface Height
SSM/I	Special Sensor Microwave Imager (of DMSP)
SSS	Sea Surface Salinity
SST	Sea Surface Temperature
TBC	To Be Confirmed
TBD	To Be Determined
TDP	Technical Data Package
TDS	Test Data Set
TN	Technical Note (short report 10-50 pages)
TO	Technical Officer (of the Agency)
TR	Technical Report (long report > 50 pages)
TS	Technical Specification
TOPEX	TOPEX-Poseidon altimetry mission (NASA/France)
UM	User Manual
URL	Universal Resource Locator
VIIRS	The NOAA Visible Infrared Imager Radiometer Suite
WP	Work package
WPD	Work package description
1D	One dimensional
3D	Three dimensional

Table 3 List of abbreviations and acronyms

2 EXTERNAL INTERFACES

The following data flow and system breakdown presents the external interfaces to the ***OceanHeatFlux*** system :



These external interfaces include :

1. **EO data providers** : they provide the input data to the system and are interfaced with OceanHeatFlux through the **data pull** subsystem. There is unfortunately no standard protocol for data delivery by providers and OceanHeatFlux has therefore to be ready to interface with several possible access means : FTP, OpenDAP and local network access are initially provided and can be extended through connection plugins.
2. interfaces for a **direct access** to the OceanHeatFlux product archive through standard protocols such as FTP and OpenDAP
3. a **web user interface** communicating to provide user-friendly display and analysis functions of the OceanHeatFlux products.
4. A **direct access** to a sandbox environment on Ifremer *Nephelae* platform, allowing users or partners to remotely work with, (re)process or analyse the OceanHeatFlux data

2.1 EO DATA PROVIDERS

In OceanHeatFlux, the ingestion of the data is performed in a data driven way. The system must detect the availability of new data files (meaning data files not yet seen and ingested by OceanHeatFlux), which is the function of the **data pull subsystem**. This subsystem interfaces with the source data provider's archives which can be remote or local, available through various network protocols. It is not possible to cover all possible protocols but the OceanHeatFlux data pull subsystem is by design fully extensible so that new protocols can be added later. Initially, the following protocols are provided :

1. FTP
2. OpenDAP

2.1.1 FTP

FTP is a standard protocol that is still used by most of the data providers. It is natively supported by numerous clients, libraries and softwares. It is natively supported by the ***OceanHeatFlux*** data pull subsystem too.

FTP servers require login and password, that have to requested to the original provider for each dataset ingested in ***OceanHeatFlux*** and configured in the data pull subsystem.

Data files can not be read directly through FTP and must be first downloaded to a local mirror archive or rolling archive, that are natively proposed by the data pull system.

The file organization and naming on a FTP server is completely dependent on the provider and may be different for each dataset. It has to be configured for each dataset in the data pull subsystem (refer also to [DARD]).

2.1.2 OPENDAP

The OPeNDAP Data Access Protocol (DAP) is a protocol for requesting and transporting data across the web. DAP 2.0 uses HTTP to frame the requests and responses.

Few datasets are uniquely available through OpenDAP though it is starting to be the case for several operational projects (such as MyOcean). It is therefore implemented within OceanHeatFlux for access to some existing datasets.

OpenDAP access works with HTTP URLs linking to repositories and files, like in a FTP site. There is no need to download a complete file (meaning that in OceanHeatFlux the data pull subsystem only transmits an OpenDAP file link to the ingestion subsystem that subsets the source data through OpenDAP protocol), limiting the amount of data to download (contrary to FTP).

The standard OpenDAP server does not provide user authentication and access is therefore public.

The file organization and naming on a OpenDAP server is completely dependent on the provider and may be different for each dataset. It has to be configured for each dataset in the data pull subsystem (refer also to [DARD]).

For a complete description of a DAP interface, see Data Access Protocol (DAP) version 2, a complete technical description of the data access protocol is available at :
http://www.opendap.org/pdf/ESE-RFC-004v1.1.pdf

2.2 INTERFACES TO OCEANHEATFLUX PRODUCT ARCHIVE FOR DIRECT ACCESS

The OceanHeatFlux archive is visible in read-only access through FTP and OpenDAP. Local access though Ifremer cloud is also provided.

2.2.1 FILENAMING

2.2.1.1 Gridded products

Gridded products include simple regridding of satellite swath data (for instance over one day, or with respect to ascending/descending passes) – referred as L3 products in satellite community, multi-sensor merging (several satellite sources, or satellite/model/insitu blending) – referred as L4 products, model outputs, climatologies, periodic (monthly etc...) means,... For such product, there will be **one single file per time step**. A file may include several fields for the same time step.

Such files will be named as follow :

YYYYMMDDHHMMSS-OHF-<**Level**>-<**Parameter**>-<**Source**>-<**freeform**>-v<**XX.Y**>-fv<**ZZ.W**>.nc

where :

- YYYYMMDDHHMMSS is the date and time of the product time step
- **Level** may be *L3* for simple data binning or gridding, *L4* when implying sensor merging, modelling, analysis
- **Parameter** is an abbreviated designation for the main geophysical phenomenon described by the product (SST for sea surface temperature, WND for wind, CUR for currents, ...)
- **Source** is the observation source (name of the satellite or sensor, name of model,...)
- **freeform** is a free field to further explicit the product content (could be for instance the name of the original provider institution when reformatting existing data)
- **XX.Y** is the generating software/algorithm version
- **ZZ.W** is the file version in case a product file is processed in several steps or updated (01.0 by default)

2.2.1.2 Swath products

Swath products contain satellite data in their original satellite projection. For such product, there will be **one single file per orbit**.

File naming convention is similar to gridded product (as described above), except :

- YYYYMMDDHHMMSS refers to the start date and time of the orbit file.
- the **Level** field should be L2.

Note that products from geostationary satellite are classified as gridded products.

2.2.1.3 In situ products

Granularity and file naming of in situ data shall follow the convention adopted for the in situ data within GlobWave project, which itself was based on CF convention and existing efforts (at NDBC for instance). This convention is described below.

There shall be one file per platform, instrument and month. For platforms (such as buoys) having different onboard instruments, the respective data will therefore be put in two different files (in different repositories as show later in the data organization section). Different instruments may indeed have different sampling times and often different availability time ranges too.

A product containing the data from one instrument by one platform (buoy, ship,...) during one specific month will be named as follow :

<**Platform**>_<**date start**>_<**date stop**>_<**freeform**>.nc

- **Platform** is the identifier of the acquisition platform : if it as a WMO identifier, it shall be labelled WMO<identifier> otherwise it will use the provider identifier.
- **Date start** is the date and time of the first measurement in file (expressed as YYYYMMDDTHHMM, ex: 20120503T2300) for the current month
- **Date stop** is the date and time of the last measurement in file (expressed as YYYYMMDDTHHMM, ex: 20120503T2300)for the current month
- **freeform** is a free field further expliciting the content or origin of the data. For instance for a moored buoy, it may be the location of the buoy expressed as 'Lat_xx.yy[N|S]_Lon_zz.ww[W|E]' (exemple : Lat_27.35N_Lon_84.28W)

Note that if there is a significant interruption time (to be defined, more than one day for instance) or a significant change in the acquisition process (ex: maintenance change on the instrument, or replacement), then the data product for a specific platform/instrument may be splitted into two or more chronologically consecutive files within the current reference month.

2.2.2 ORGANIZATION

The data organization will have the following levels :

- product origin : o*ceanheatflux* for datasets specifically produced by the project, *others* for any other input or validation data
- domain of data : *satellite, insitu, model, references* (climatologies, bathymetry, etc...)
- main quantity : *currents, mdt, sea_surface_temperature, wind, wave, ...*
- product level (for satellite data) : l2, l3, l4, ...
- product name
- date, expressed as *YYYY/DDD*, *YYYY/MM* or *YYYY*

As an example :

```
data
├── oceanheatflux
└── others
    ├── insitu
    ├── model
    ├── references
    │   └── climatologies
    │       ├── sea_surface_temperature
    │       └── ...
    └── satellite
        └── wind
            ├── l2
            ├── l3
            └── l4
```

2.2.3 ACCESS PROTOCOLS

2.2.3.1 FTP access

FTP access to OceanHeatFlux products is accessible to any identified user. Login will link directly to the root of the data organization described above. Only datasets authorized for the user profile will be visible and accessible.

Login and password are requested and obtained through the help desk of each instance (for Ifremer : cersat@ifremer.fr).

2.2.3.2 OpenDAP

OceanHeatFlux products are accessible through OpenDAP at the following URL :
It is not yet possible to provide login protected access to OpenDAP so OpenDAP access will be restricted to public OceanHeatFlux products. This will be revised once a new OpenDAP version is released with improved management of user access restrictions.

2.2.3.3 Cloud access

Ifremer makes available to OceanHeatFlux users a “sandbox” on its *Nephelae* cloud to remotely process and analyse the OceanHeatFlux data archive, without having to download the data.

Each user needs to be registered and own an Ifremer account, to be requested to Ifremer/CERSAT help desk (cersat@ifremer.fr). Access will be granted on approval by Ifremer, depending on the intended usage and available resource on the cloud.

The user can then connect through ssh to a virtual machine allocated to him only, with the following properties :

1. Ubuntu Linux system
1. access permission to the OceanHeatFlux data archive
2. python distribution with most usual scientific packages : Numpy, Matplotlib, netCDF4,... (more packages on request)

Interactive access (for visualization for instance) can be obtained by installing a remote desktop client such as *NX Client* on the user side instead of interacting with the virtual machine through a ssh terminal.

Any other Linux distribution or third party tools (such as Matlab) to match more closely the user environment and habits must be arranged with Ifremer team.

Tools to run batch processing over large portions of the data archive on several servers in a distributed way will also be made available to the users on demand.

For more information, refer on the dedicated service page set up for OceanFlux project : http://www.oceanflux-ghg.org/Products/Tools

3 FORMAT

OceanHeatFlux internal (collected and/or transformed) and public products are all stored in netCDF4 format although the source data used may be in a different format. The collected data are stored in their native projection and pattern, and therefore their format may vary depending on the pattern of the source data.

In any case, the format is CF convention compliant. It provides :

1. unified data model time and coordinates variables and dimensions
2. specific attributes for geophysical variables
3. specific list global attributes for metadata (metadata from source file are also copied here)
4. a mask variable allowing to mask measurements out of the site area (as the extracted miniProd will bound the site limits including pixels out of the defined polygonal area for the site)

3.1.1 GEOLOCATION INFORMATION

The geolocation information includes the variables and dimensions necessary to structure the data arrays and locate in time and space each measurement.

This is strongly dependent on the pattern of the data : satellite swath, regular grid, non regular grid, time series (buoy), trajectory (ship, drifting buoy). The corresponding specifications for each of these pattern are listed below, in CDL language :

3.1.1.1 swath

This format is used for swath satellite data (radiometer, scatterometer,...) in native projection (L1, L2 data).

```
dimensions:
      row = 1615 ;
      cell = 19 ;
      alias = 4 ;

variables:
      int time(row, cell) ;
            time:long_name = "time" ;
            time:units = "seconds since 1990-01-01 00:00:00" ;
            time:calendar = "standard" ;
      float lat(row, cell) ;
            lat:long_name = "latitude" ;
```

```
            lat:units = "degrees_north" ;
            lat:standard_name = "latitude" ;
      float lon(row, cell) ;
            lon:long_name = "longitude" ;
            lon:units = "degrees_east" ;
            lon:standard_name = "longitude" ;
```

The dimension of a variable should be (row,cell). Example :

```
      float model_wind_speed(row, cell) ;
            model_wind_speed:_FillValue = -32768s ;
            model_wind_speed:long_name = "model wind speed" ;
            model_wind_speed:units = "m s-1" ;
            model_wind_speed:valid_min = 0. ;
            model_wind_speed:valid_max = 60. ;
            model_wind_speed:coordinates = "lat lon" ;
```

If different depths are provided, the dimensions of a variable should be (depth, row, cell). Example :

```
      float geostrophic_velocity_northward(depth, row, cell) ;
            geostrophic_velocity:_FillValue = -32768s ;
            geostrophic_velocity:long_name = "model wind speed" ;
            geostrophic_velocity:units = "m s-1" ;
            geostrophic_velocity:valid_min = 0. ;
            geostrophic_velocity:valid_max = 10. ;
            geostrophic_velocity:coordinates = "lat lon" ;
```

3.1.1.2 regular grid

This format is used for model outputs or satellite L4 and L3 products on grids in cylindric projections.

```
dimensions:
      lon = 7200 ;
      lat = 521 ;
      time = 1 ;

variables:
      float lon(lon) ;
            lon:standard_name = "longitude" ;
            lon:long_name = "longitude" ;
```

```
            lon:units = "degrees_east" ;
            lon:axis = "X" ;
      float lat(lat) ;
            lat:standard_name = "latitude" ;
            lat:long_name = "latitude" ;
            lat:units = "degrees_north" ;
            lat:axis = "Y" ;
      int time(time) ;
            time:standard_name = "time" ;
            time:long_name = "reference time of SST field" ;
            time:units = "seconds since 1981-01-01 00:00:00" ;
            time:axis = "T" ;
            time:calendar = "standard" ;
```

The dimensions of a variable will then be (time,lat,lon) :

```
      short analysed_sst(time, lat, lon) ;
            analysed_sst:standard_name = "sea_surface_temperature" ;
            analysed_sst:long_name = "analysed sea surface temperature" ;
            analysed_sst:units = "kelvin" ;
            analysed_sst:valid_min = -300.f ;
            analysed_sst:valid_max = 4500.f ;
            analysed_sst:scale_factor = 0.01f ;
            analysed_sst:add_offset = 273.15f ;
            analysed_sst:_FillValue = -32768s ;
            analysed_sst:type = "foundation" ;
```

3.1.1.3 Irregular grid

This format is used for model outputs or satellite L4 and L3 products on grid with curvilinear coordinates.

```
dimensions:
      time = 1 ;
      nv = 2 ;
      xc = 790 ;
      yc = 830 ;

variables:
      int Polar_Stereographic_Grid ;
            Polar_Stereographic_Grid:grid_mapping_name = "polar_stereographic" ;
            Polar_Stereographic_Grid:straight_vertical_longitude_from_pole = 0.f ;
            Polar_Stereographic_Grid:latitude_of_projection_origin = -90.f ;
            Polar_Stereographic_Grid:standard_parallel = -70.f ;
```

```
            Polar_Stereographic_Grid:false_easting = 0.f ;
            Polar_Stereographic_Grid:false_northing = 0.f ;
            Polar_Stereographic_Grid:semi_major_axis = 6378273.f ;
            Polar_Stereographic_Grid:semi_minor_axis = 6356890.f ;
            Polar_Stereographic_Grid:proj4_string = "+proj=stere +a=6378273
+b=6356889.44891 +lat_0=-90 +lat_ts=-70 +lon_0=0" ;
      double time(time) ;
            time:axis = "T" ;
            time:long_name = "reference time of product" ;
            time:standard_name = "time" ;
            time:units = "seconds since 1978-01-01 00:00:00" ;
            time:calendar = "standard" ;
            time:bounds = "time_bnds" ;
      double time_bnds(time, nv) ;
            time_bnds:units = "seconds since 1978-01-01 00:00:00" ;
      double xc(xc) ;
            xc:axis = "X" ;
            xc:units = "km" ;
            xc:long_name = "x coordinate of projection (eastings)" ;
            xc:standard_name = "projection_x_coordinate" ;
            xc:grid_spacing = "10.0000 km" ;
      double yc(yc) ;
            yc:axis = "Y" ;
            yc:units = "km" ;
            yc:long_name = "y coordinate of projection (northings)" ;
            yc:standard_name = "projection_y_coordinate" ;
            yc:grid_spacing = "10.0000 km" ;
      float lat(yc, xc) ;
            lat:long_name = "latitude coordinate" ;
            lat:standard_name = "latitude" ;
            lat:units = "degrees_north" ;
      float lon(yc, xc) ;
            lon:long_name = "longitude coordinate" ;
            lon:standard_name = "longitude" ;
            lon:units = "degrees_east" ;
```

The dimensions of a variable will then be (time,yc,xc) :

```
      short ice_conc(time, yc, xc) ;
            ice_conc:long_name = "concentration of sea ice" ;
            ice_conc:standard_name = "sea_ice_area_fraction" ;
            ice_conc:units = "%" ;
            ice_conc:_FillValue = -32767s ;
            ice_conc:valid_min = 0s ;
            ice_conc:valid_max = 10000s ;
            ice_conc:grid_mapping = "Polar_Stereographic_Grid" ;
            ice_conc:coordinates = "lat lon" ;
            ice_conc:scale_factor = 0.01f ;
```

```
```

3.1.1.4 Time series

Used for any fixed station such a moored buoy.

```
dimensions:
      time = UNLIMITED ; // (1412 currently)
      station = 1 ;

variables:
      int time(time) ;
            time:units = "seconds since 1970-01-01 00:00:00" ;
            time:long_name = "time of measurement" ;
      float lat(station) ;
            lat:units = "degrees_north" ;
            lat:long_name = "latitude" ;
            lat:standard_name = "latitude" ;
      float lon(station) ;
            lon:units = "degrees_east" ;
            lon:long_name = "longitude" ;
            lon:standard_name = "longitude" ;
      float depth(station) ;
            depth:units = "m" ;
            depth:long_name = "depth" ;
```

The dimensions of a variable will then be (time). Example :

```
      float dominant_wave_period(time) ;
            dominant_wave_period:_FillValue = 9.96921e+36f ;
            dominant_wave_period:long_name = "dominant wave period (spectral peak
period, Tp)" ;
            dominant_wave_period:units = "s" ;
            dominant_wave_period:comment = "inverse of the frequency with the
highest energy density in the reported spectrum." ;
            dominant_wave_period:sampling_type = "Mean" ;
            dominant_wave_period:averaging_length_in_minutes = 26.6666666666667 ;
```

3.1.1.5 trajectory, along-track

Used for any moving platform such as a float, drifting buoy or ship, as well as for along-track data (altimeter).

```
dimensions:
```

```
      time = 1412 ;

variables:
      int time(time) ;
            time:units = "seconds since 1970-01-01 00:00:00" ;
            time:long_name = "time of measurement" ;
      float lat(time) ;
            lat:units = "degrees_north" ;
            lat:long_name = "latitude" ;
            lat:standard_name = "latitude" ;
      float lon(time) ;
            lon:units = "degrees_east" ;
            lon:long_name = "longitude" ;
            lon:standard_name = "longitude" ;
      float depth(time) ;
            depth:units = "m" ;
            depth:long_name = "depth" ;
```

The dimensions of a variable will then be (time). Example :

```
      float dominant_wave_period(time) ;
            dominant_wave_period:_FillValue = 9.96921e+36f ;
            dominant_wave_period:long_name = "dominant wave period (spectral peak
period, Tp)" ;
            dominant_wave_period:units = "s" ;
            dominant_wave_period:comment = "inverse of the frequency with the
highest energy density in the reported spectrum." ;
            dominant_wave_period:sampling_type = "Mean" ;
            dominant_wave_period:averaging_length_in_minutes = 26.6666666666667 ;
```

3.1.2 GEOPHYSICAL INFORMATION

The geophysical variables in NetCDF files shall follow the dimensions recommended for the sampling pattern (previous section) and follow CF 1.6 convention whenever possible.

3.1.2.1 Attributes

In particular the following variable attributes shall be provided :

- **long_name**
- **units :** units of the measurd phenomenon. Units must be expressed following udunits package convention (http://www.unidata.ucar.edu/software/udunits/)
- **standard_name** : if available for the described phenomenon. Refer to the list of

existing standard names on CF convention page (http://cf-pcmdi.llnl.gov/documents/cf-standard-names/). If no standard name is existing yet, it shall not be invented!

- **_FillValue** : missing value used for the variable. It should be, by convention, the minimum value of the storage type (ex:-32768 for a signed short)
- **scale_factor**
- **add_offset** : offset to add to the stored data to get the actual physical value (in case for instance they are stored on integers). Actual value = stored value * scale_factor + offset
- **valid_min** and **valid_max** : minimum and maximum valid value for the measured phenomenon. They must be expressed in the same type as the variable.

```
float analysed_sst(time, lat, lon) ;
      analysed_sst:standard_name = "sea_surface_temperature" ;
      analysed_sst:long_name = "analysed sea surface temperature" ;
      analysed_sst:units = "kelvin" ;
      analysed_sst:valid_min = -3.f ;
      analysed_sst:valid_max = 45.f ;
      analysed_sst:scale_factor = 0.01f ;
      analysed_sst:add_offset = 273.15f ;
      analysed_sst:_FillValue = -32768s ;
      analysed_sst:type = "foundation" ;
```

Note : as the data are stored in NetCDF4 format, floating values should be stored on floats or doubles. Therefore there is no need anymore of scale_factor and add_offset attributes as no scaling is applied to the stored values (internal features of NetCDF4 such as compression and digit precision being used instead).

3.1.2.2 Variable naming

Avoid ambiguous naming such as 'u' and 'v' (for vector components). Explicit and unambiguous names are recommended.

Vectors should be stored as northward and eastward components (not module and direction). For better integration in visualization tools, vector components must be named as :

- northward_<*geophysical quantity*>
- eastward_<*geophysical quantity*>

Suggested variable names in OceanHeatFlux :

Variable name	Standard name	Description
northward_wind_speed eastward_wind_wind_speed wind_speed	TBD	
northward_wind_stress eastward_wind_wind_stress wind_stress		
....		

3.1.3 METADATA

The metadata attributes to be filled in each NetCDF file are required, like the variables, to comply with CF 1.6 convention (http://cf-pcmdi.llnl.gov/documents/cf-conventions/1.6/cf-conventions.html) and Unidata Common Data Model (CDM, http://www.unidata.ucar.edu/software/netcdf-java/CDM/) and data discovery attributes (http://www.unidata.ucar.edu/software/netcdf-java/formats/DataDiscoveryAttConvention.html).

They are completed by a set of global attributes, based on the metadata specifications from previous projects.

3.1.3.1 General attributes

The mandatory attributes applying to any product file include :

Attribute name	Description / value
Conventions	"CF-1.6"
netcdf_version_id	"4.1.1 of Dec 22 2011 16:33:39 $"
date_created	Product creation date (YYYY-MM-DDTHH:MM:SS)
date_modified	Product last modification date (YYYY-MM-DDTHH:MM:SS)
id naming_authority	The "id" and "naming_authority" attributes are intended to provide a globally unique identification for each dataset. The "id" value should attempt to uniquely identify the dataset. The naming authority allows a further refinement of the "id". The combination of the two should be globally unique for all time. We recommend using reverse-DNS naming for the naming authority. For example, naming_authority="fr.ifremer.cersat" and

	id="WW3/hindcast_global_050".
institution	Institution which the data originally come from. If a dataset is a simple reformatting without any modification the source institution is to be used. Ex: Institut Francais de Recherche pour l'Exploitation de la Mer
institution_abbreviation	Ex: Ifremer
title	The "title" attribute gives a brief description of the dataset. Its use is highly recommended and its value will be used by THREDDS as the name of the dataset. It therefore should be human readable and reasonable to display in a list of such names. The "title" attribute is recommended by the "NetCDF Users Guide" and the CF convention.
summary	The "summary" attribute gives a longer description of the dataset. Its use is highly recommended. In many discovery systems, the title and the summary will be displayed in the results list from a search. It should therefore capture the essence of the dataset it describes. For instance, we recommend this field include information on the type of data contained in the dataset, how the data was created (e.g., instrument X; or model X, run Y), the creator of the dataset, the project for which the data was created, the geospatial coverage of the data, and the temporal coverage of the data. This should just be a summary of this information, more detail should be provided in the recommended creator attributes, the recommended geospatial attributes, and the recommended temporal attributes.
cdm_feature_type	Data feature type (point,station,swath,grid,...) as defined by Unidata CDM model : http://www.unidata.ucar.edu/software/netcdf-java/CDM/
keywords	The "keywords" attribute lists key words and phrases that are relevant to the dataset. Its use is highly recommended. The values in the list may be taken from a controlled list of keywords (e.g., the AGU Index list or the GCMD Science Keywords). If a controlled list is used, the "keywords_vocabulary" attribute may be used to identify the list.
keywords_vocabulary	The "keywords_vocabulary" attribute identifies the controlled list of keywords from which the values in the "keywords" attribute are taken. If you are following a guideline for the words/phrases in your "keywords" attribute, put the name of that guideline here. The use of this attribute

	is recommended and its value will be used by THREDDS to identify the vocabulary from which the keywords come.
standard_name_vocabulary	"NetCDF Climate and Forecast (CF) Metadata Convention”
project	The "project" attribute provides the name of the scientific project for which the data was created. The use of this attribute is recommended. (ex: OceanHeatFlux)
acknowledgement	A place to acknowledge various type of support for the project that produced this data.
license	Describe the restrictions to data access and distribution.
format_version	OceanHeatFlux format version (1.0)
history	The "history" attribute provides an audit trail for modifications to the original data. It should contain a separate line for each modification with each line including a timestamp, user name, modification name, and modification arguments. Its use is recommended and its value will be used by THREDDS as a history-type documentation. The "history" attribute is recommended by the NetCDF Users Guide and the CF convention.
publisher_name	Name of the distributing agency (Ifremer/CERSAT for OceanFlux)
publisher_url	URL of the distributing agency (http://cersat.ifremer.fr)
publisher_email	Email of help desk or contact point
creator_name	Name of the creating agency/person
creator_url	URL of the creating agency/person
creator_email	Email of help desk or contact point
processing_software	Name and version of the procesing software
processing_level	The "processing_level" attribute provides a textual description of the processing (or quality control) level of the data.
references	
nominal_latitude	Location of acquisition platform. Only used in case of a fixed

nominal_longitude	station (ex: moored buoy)
geospatial_lat_min geospatial_lat_max geospatial_lat_units	north/south latitude boundaries (units in "degrees")
geospatial_lon_min geospatial_lon_max geospatial_lon_units	east/west longitude boundaries (units in "degrees")
geospatial_vertical_min geospatial_vertical_max geospatial_vertical_units	depth/height boundaries (units in "meters above mean sea level"
time_coverage_start time_coverage_stop time_coverage_resolution	Start and end time of the product file data
platform_type	Type of platform (satellite, moored buoy,...)
sensor_type	Type of instrument (altimeter, accelerometer, anemometer,...)
sensor_description sensor_manufacturer sensor_part_number sensor_serial_number sensor_install_date sensor_height sensor_sampling_period sensor_sampling_rate sensor_calibration_date sensor_history	Instrument more specific information (especially for in situ instruments)

3.1.3.2 Specific attributes for satellite products

platform_name	Full name of the satellite (ex:Environmental Satellite)
platform_id	Identifier of the satellite (ex:Envisat)
sensor_id	Identifier of the sensor (ex: RA2)

3.1.3.3 Specific attributes for in situ data

wmo_id	WMO identifier, if any
buoy_network	Ex: NODC
station_name	Full name of the station (usually a location)
station_id	Identifier of the station in network
sea_floor_depth_below_sea_level	
site_elevation	

Example :

```
		:Conventions = "CF-1.6" ;
		:netcdf_version_id = "4.1.1 of Dec 22 2011 16:33:39 $" ;
		:date_created = "2012-05-02T05:25:SZ" ;
		:id = "51002" ;
		:naming_authority = "WMO" ;
		:wmo_id = "51002" ;
		:institution = "National Oceanographic Data Center" ;
		:institution_abbreviation = "NODC" ;
		:buoy_network = "NDBC" ;
		:title = "Buoy observation from NODC provided for GlobWave project" ;
		:summary =  ;
		:station_name =  ;
		:sea_floor_depth_below_sea_level = 5001.8 ;
		:site_elevation = 0. ;
		:cdm_feature_type = "station" ;
		:scientific_project = "GlobWave" ;
		:restrictions = "Restricted to Ifremer and GlobWave usage" ;
		:format_version = "2.0" ;
		:history = "1.0 : Processing to GlobWave netCDF format" ;
		:publisher_name = "Ifremer/Cersat" ;
		:publisher_url = "http://cersat.ifremer.fr" ;
		:publisher_email = "jfpiolle@ifremer.fr" ;
		:creator_url = "http://cersat.ifremer.fr" ;
		:creator_email = "jfpiolle@ifremer.fr" ;
		:date_modified =  ;
		:processing_software = "Globwave python lib v1.0" ;
		:references =  ;
		:data_source = "51002_201101" ;
		:nominal_latitude = 17.0941666666667 ;
		:nominal_longitude = -157.8075 ;
		:geospatial_lat_min = 17.0941666666667 ;
		:geospatial_lat_max = 17.0941666666667 ;
		:geospatial_lat_units = "degrees" ;
		:geospatial_lon_min = -157.8075 ;
		:geospatial_lon_max = -157.8075 ;
		:geospatial_lon_units = "degrees" ;
		:geospatial_vertical_min = 0. ;
```

```
		:geospatial_vertical_max = 0. ;
		:geospatial_vertical_units = "meters above mean sea level" ;
		:platform_type =  ;
		:sensor_type = "Directional accelerometer" ;
		:sensor_description =  ;
		:sensor_manufacturer =  ;
		:sensor_part_number =  ;
		:sensor_serial_number =  ;
		:sensor_install_date =  ;
		:sensor_height =  ;
		:sensor_sampling_period = 20. ;
		:sensor_sampling_rate = 1.70666666666667 ;
		:sensor_calibration_date =  ;
		:sensor_history =  ;
		:time_coverage_start = "2011-01-01T00:40:00Z" ;
		:time_coverage_stop = "2011-01-31T23:40:00Z" ;
```

4 THE OCEANHEATFLUX GRID

In order to combine or compare different parameters relevant for fluxes, it is necessary to remap the different fields on to the same spatial and temporal map. This capability will be implemented in OceanHeatFlux system and applied to any inputs considered in the project. The properties of this "OceanHeatFlux grid" were defined with respect to the available resolution of the various inputs.

The currently considered "OceanHeatFlux grid" has the following characteristics.

5. Plate Carree projection (regular lat/lon grid): this projection is not equal area and therefore introduces severe distortions towards high latitudes. However most of the inputs are provided on such projection so regridding to an equal area would not correct this. This choice may be revised at a later stage.
6. 1 degree resolution: this is the most common resolution of available fluxes climatologies.
7. -180/180° and -80/80° limits. The grid is centered on Greenwhich meridian. The provided lat/lon refer to the pixel's center.