SALP

SALP Products Specification – Volume 20 : AltiKa / SARAL User Products

Prepared by :	J.P. DUMONT S. URIEN E. BRONNER	CLS CLS CNES	12/09/2012 p.o Bronner
Accepted by :	J. NOUBEL	CNES	12/09/2012
Approved by :	T.GUINLE	CNES	12/09/2012

Document ref : SALP-ST-M-EA-15839-CN	Issue :2	Update :2

For	DS2	DS4	DS5	DH2	TP	ENVISAT	JASON1	DCY	LTA-SIRAL
Application to									
For	SMM	SALP					JASON2		SARAL/AltiKa
Application to									X

Configuration controlled Document	YES	by : CCM SALP	Since : 30-01-2009

SUMMARY

Confidentiality : no Type :
Key words : SARAL User Products
Summary : This document is aimed at defining the AltiKa / SARAL User Products

DOCUMENT CHANGE RECORD

Issue	Update	Date	Modifications	Visa
1	**0**	January, 30th 2009	Creation	
1	**1**	June, 1st 2009	• Adding of associated quality flags for latitude, longitude, altimeter range, significant wave height and backscatter coefficient 1-Hz parameters for standard and expert datasets • Accounting for the ice-2 and sea-ice retracking algorithms in the processing • Accounting for a radiometer instrumental characterization data file (global attribute)	
1	**2**	July, 15th 2009	• Adding of 1-Hz and 40-Hz altimeter rain flags	
2	**0**	September 12th, 2011	• New orbit quality flag in the OGDR products (SALP-FT-7963) • Modification of the introduction of the document • Adding of the ice flag in SSHA products, update of the GOT tide model to GOT4.7 (DM SALP-FT-7963) and typo corrections	**E.BRONNER**
2	**1**	November 9th, 2011	• Quality flag = "orb_state_flag_rest" replaced by Quality flag = "orb_state_flag_rest or orb_state_flag_diode" + comments (SALP-FT-8343) • Microseconds (".mmmmmm") removed from the global attribute « history » (SALP-FT-8343) • Modification of the "tracker_diode_40hz:long_name" ('counter' removed from the field) (SALP-FT-8343) • New GOT 4.8 tidal model instead of GOT 4.7 (SALP-FT-8343) • New MSS_CNES_CLS-2011 mean sea surface instead of MSS CLS_01(SALP-FT-8343) • New MDT_CNES_CLS-2009 mean dynamic	**E.BRONNER**

			topography instead of MDT CLS_RIO 2005 (SALP-FT-8343) • Trailing_edge_variation_flag" used instead of "rain_flag" (SALP-FT-8343) • No editing strategy for SSHA calculation (SALP-FT-8343rev1) • New attributes "xref_gim_data", "xref_mog2D_data" (SALP-FT-8343) • Attribute "xref_orf_data" added in SSHA dataset (SALP-FT-8343) • New fields added in expertise dataset (map of valid estimates of the tracker ranges, 40-Hz ice-2 epoch, 40-Hz sea ice epoch, 40-Hz ice-2 amplitude, 40-Hz ice-2 mean amplitude, 40-Hz sea ice amplitude, 40-Hz ice-2 thermal noise, 40-Hz square of the off-nadir angle for rain flag computation, 40-Hz slope of the trailing edge for mispointing) (SALP-FT-8343) • Following flags removed : "alt_quality_flag", "rad_quality_flag", "geophysical_quality_flag", "alt_echo_type" (SALP-FT-8343) • Model wet tropospheric correction used in SSHA computation instead of radiometric wet tropospheric correction (SALP-FT-8343) • Parameter "atmos_corr_sig0" coded using short integer instead of byte (SALP-FT-8343rev1).	
2	**2**	February 1st, 2012	• Acquisition station name attribute (« acq_station_name ») removed (SALP-FT-8343rev2) • New fields added in expertise dataset (pri, off_nadir_angle_pf, signal_to_noise_ratio) (SALP-FT-8343rev2) • New comments for "tb_k" and "tb_ka" parameters (SALP-FT-8343rev2) • New meaning for the trailing edge variation flag (SALP-FT-8343rev2) • SSHA calculated even if GIM is at default value (SALP-FT-8343rev2) • Modification of typo + parameters order.	

ABBREVIATIONS

Sigle	Definition
AD	Applicable Documents
AGC	Automatic Gain Control
CAL	Calibration
CDL	Common Data Language
CF	Climate and Forecast convention
CLS	Collecte Localisation Satellites
CNES	Centre National d'Etudes Spatiales
COG	Center Of Gravity
DAD	Dynamic Auxiliary Data
DORIS	Doppler Orbitography and Radiopositioning Integrated by Satellite
ECMWF	European Centre for Medium-Range Weather Forecasts
FFT	Fast Fourier Transform
GDR	Geophysical Data Record
GPS	Global Positioning System
IGDR	Interim Geophysical Data Record
LPF	Low Pass Filter
LTM	Long Term Monitoring
MDS	Measurement Data Set
N/A	Not Applicable
NRT	Near Real Time
OFL	Off Line
OGDR	Operational Geophysical Data Record
POD	Precise Orbit Determination
POE	Precise Orbit Ephemeris
PTR	Point Target Response
RD	Reference Documents
RMS	Root Mean Square
SAD	Static Auxiliary Data
SALP	Service d'Altimétrie et Localisation Précise
SDR	Sensor Data Record
SGDR	Sensor Geophysical Data Record
SNR	Signal to Noise Ratio
SSALTO	Segment Sol ALTimétrie et Orbitographie
SSHA	Sea-Surface Height Anomaly
SWH	Significant WaveHeight
TBC	To Be Confirmed
TBD	To Be Defined
TEC	Total Electron Content
USO	Ultra Stable Oscillator
UTC	Universal Time Coordinate

APPLICABLE AND REFERENCE DOCUMENTS

Reference	Document title
ALK-SY1-SP-056-CNES	AD 1 ALTIKA/SARAL OPERATIONAL SERVICE SPECIFICATION
SALP-MU-M-OP-15984-CN	AD 2 ALTIKA/SARAL Products Handbook

TBC AND TBD LIST

TBC/TBD	Page	Brief description
TBC	12 and 13	ISRO data producer, email contact and product center names

CONTENTS

1. INTRODUCTION

The aim of this document is to define the SARAL Level 2 altimeter products specifications. It is applicable to the development of the processing module (SPA, TM_NRT) and of the other tools developed by 3 partners (BUFR convertor, …). Document AD 2 gives complementary information to users.

According to requirements from AD 1, three different data products shall be produced and distributed to the users:

1. Operational Geophysical Data Record (**OGDR**) produced in near real time
2. Interim Geophysical Data Record (**IGDR**) produced in 1 to 1.5 days
3. Geophysical Data Record (**GDR**) produced in 60 days

The first one is a NRT product. The other two are OFL products.

In addition to the native NetCdf format which is described in this document, a 1Hz BUFR-formatted dataset from the OGDR family (OGDR-BUFR) for distribution via the World Meteorological Organization (WMO) Global Tele-communication System (GTS) and EUMETCast is also generated. The BUFR format is described in AD 2.

Netcdf OGDR/IGDR/GDR products will have the same information and format. The only difference will be related to auxiliary data (orbit, meteo files, calibrations, …).

Taken into account Jason-1/2 heritage, products will be split into several data sets :

1. One "**reduced**" file, close to the current Jason-1/2 NRT-SSHA, limited to 1 Hz sampling
2. One "**standard**" file, close to the current Jason-1/2 I/GDR, containing 1 Hz and 40 Hz values
3. One "**expertise**" file, close to the current Jason-1/2 SGDR, containing 1 Hz, 40 Hz and waveforms values. This file will not be generated in NRT.

The following table shows the data sets available for each kind of product.

		Data set		
		Reduced	**Standard**	**Expertise**
Products	OGDR	X	X	
	IGDR	X	X	X
	GDR	X	X	X

Table 1 – Data set availability per product

An overview of the file format used for the data sets is given in section 2. Then the data sets are described from section 3 to section 6.

2. SARAL PRODUCTS OVERVIEW

2.1. NETCDF FORMAT AND CF CONVENTION

The NetCDF data format has been chosen to store the different data sets (one file per data set). This format is extremely flexible, self describing and has been adopted as a de-facto standard for many operational oceanography systems. What's more, the files will follow the Climate and Forecast NetCDF conventions CF-1.1 because these conventions provide a practical standard for storing.

2.2. THE NETCDF DATA MODEL

A NetCDF file contains dimensions, variables, and attributes, which all have both a name by which they are identified. These components can be used together to capture the meaning of data and relations among data fields in an array-oriented data set.

2.2.1. DIMENSIONS

A dimension may be used to represent a real physical dimension, for example, time, latitude, longitude, or height. A dimension might also be used to index other quantities (waveforms index for example). The following dimensions are used in the SARAL product files:

		Data set		
Dimension name	**Value**	**Reduced**	**Standard**	**Expertise**
time	Number of measurements in the file	**Yes**	**Yes**	**Yes**
meas_ind	**40** (number of elementary measurements)	No	**Yes**	**Yes**
wvf_ind	**128** (number of waveform samples)	No	No	**Yes**

Table 2 - Dimensions used in the SARAL data sets

2.2.2. VARIABLES

Variables are used to store the bulk of the data in a netCDF file. A variable represents an array of values of the same type. A scalar value is treated as a 0-dimensional array. A variable has a name, a data type, and a shape described by its list of dimensions specified when the variable is created. A variable may also have associated attributes, which may be added, deleted or changed after the variable is created.

A variable data type is one of a small set of netCDF types. In this document the variable types will be represented as follows:

Variable type	Description
char	characters
byte	8-bit data signed
short	16-bit signed integer
int	32-bit signed integer
float	IEEE single precision floating point (32 bits)
double	IEEE double precision floating point (64 bits)

Table 3 - netCDF variable type

2.2.3. COORDINATE VARIABLES AND AUXILIARY COORDINATE VARIABLES

A variable with the same name as a dimension is called a **coordinate variable**. It typically defines a physical coordinate corresponding to that dimension. In accordance with the Climate and Forecast conventions, we must declare a coordinate variable for each dimension. What's more, missing values are not allowed in coordinate variables and they must be strictly monotonic.

An **auxiliary coordinate variable** is a netCDF variable that contains coordinates data but is not a coordinate variable as defined above. Unlike coordinate variables, there is no relationship between the name of an auxiliary coordinate variable and the name(s) of its dimension(s).

2.2.4. ATTRIBUTES

NetCDF attributes are used to store data about the data (ancillary data or metadata), similar in many ways to the information stored in data dictionaries and schema in conventional database systems. Most attributes provide information about a specific variable. These are identified by the name of that variable, together with the name of the attribute.

Some attributes provide information about the data set as a whole. They are called **global attributes**.

The following table shows the variable attributes used in the SARAL product. There are no mandatory attributes.

Attribute	Description
_FillValue	A value used to represent missing or undefined data
add_offset	If present, this number is to be added to the date after it is read by an application. If both *scale_factor* and *add_offset* attributes are present, the date are first scaled before the offset is added.
calendar	Reference time calendar
comment	Miscellaneous information about the data or the methods used to produce it
coordinates	Identified auxiliary coordinates variables.
flag_meanings	Use in conjunction with *flag_values* to provide descriptive words or phrase for each flag value.
flag_values	Provide a list of the flag values. Use in conjunction with *flag_meanings*.

Attribute	Description
institution	Institution which provides the data
leap_second	UTC time at which a leap second occurs
long_name	A descriptive name that indicates a variable's content. This name is not standardized.
quality_flag	Name of the variable(s) (quality flag) representing the quality of the current variable
scale_factor	If present, the date are to be multiplied by this factor after the data are read by an application. See also *add_offset* attribute.
source	Data source (model features, or observation)
standard_name	A standard name that references a description of a variables content in the standard name table.
tai_utc_difference	Difference between TAI and UTC reference time
units	Unit of a variable's content. The value of this attribute must be a string that can be recognized by the UNIDATA's Udunits package.
valid_max	Largest theoretical valid value of a variable (this is not the maximum of actual data).
valid_min	Smallest theoretical valid value of a variable (this is not the minimum of actual data).

Table 4 - Variable's attributes

2.3. THE COMMON DATA LANGUAGE

The Common Data Language (CDL) will be used to describe the content of a data set.

The CDL is textual notation that described the netCDF object and it is human readable. The netCDF utility **ncdump** converts netCDF objects binary to CDL text. The netCDF utility **ncgen** creates netCDF binary file from CDL text file.

A CDL description of a netCDF data set takes the form:

```
netCDF name {
        dimension: …
        variables: …
        data: …
}
```

where the name is used only as a default in constructing file names by the ncgen utility. The CDL description consists of three optional parts, introduced by the keywords dimensions variables and data. NetCDF dimension declarations appear after the dimensions keyword, netCDF variables and attributes are defined after the variables keyword and variable data assignments appears after the data keyword. CDL statement are terminated by a semicolon. Spaces, tabs and newlines can be used freely for readability. Comments in CDL follow the characters '//' on any line.

Example :

```
netcdf example {
  dimensions:                                                  //  dimensions name are declared first
    time = 2680;

  variables:
    double time(time);                                         // variable <type> <name>(<dimension>)
        time:long_name = "time";                               // variable attributes
        time:units = "seconds since 2000-01-01 00:00:00.0";

    int lon(time);
        lon:long_name = "longitude";
        lon:standard_name = "longitude";
        lon:units = "degrees_east";
        lon:scale_factor = 1.0e-06;


     int alt(time);
        alt:long_name = "1 Hz altitude of satellite";
        alt:_FillValue = 2147483647;
        alt:units = "m";
        alt:add_offset = 1.30e+06;
        alt:scale_factor = 1.00e-04;
        alt:coordinates = "lon lat";

     byte surface_type(time);
        surface_type:long_name = "surface type";
        surface_type:_FillValue = 127b;
        surface_type:flag_values = 0b, 1b, 2b, 3b ;
        surface_type:flag_meanings = "ocean lake_enclosed_sea ice land";
        surface_type:coordinates = "lon lat";
        surface_type:comment = "Computed using a DTM2000 file: 0 = open oceans or semi-enclosed
seas; 1 = enclosed seas or lakes; 2 = continental ice; 3 = land. See SARAL User Handbook";
```

- time is a coordinate variable.
- surface_type is a flag fully described by the flag_meanings and flag_values attributes:

  surface_type = 0 -> ocean
  surface_type = 1 -> lake or enclosed sea
  surface_type = 2 -> ice
  surface_type = 3 -> land

  If surface_type is not computed, it will take the value 127 (_FillValue attribute).

- alt is *packed*. The data are stored in 32-bit integers (long). The value of the altitude of the satellite can be recovered using:

  $alt = (alt_{long} * scale\_factor) + add\_offset$

3. GLOBAL ATTRIBUTES

Global attributes are defined in the table below.

[xxx] refers to the name of the parameter. Those parameters are defined inside the TM_NRT and SPA processing modules.

			Data Set		
Attribute name	**Format**	**Description**	**Reduced**	**Standard**	**Expertise**
Conventions	String	netCDF convention followed. **[conventions]** This attribute should be set to **"CF-1.1"** to indicate that the file is compliant with the Climate and Forecast netCDF convention.	X	X	X
title	String	A descriptive title for the data set, built as follows: **[PO_PROD] – [Title_x],** leading to OGDR - Reduced dataset OGDR - Standard dataset IGDR - Reduced dataset IGDR - Standard dataset IGDR - Expertise dataset GDR - Reduced dataset GDR - Standard dataset GDR - Expertise dataset	X	X	X
institution	String	The name of the data producer (ex. CNES EUMETSAT or ISRO (TBC)): **[GA_INSTITUTION]**	X	X	X
source	String	The method of production of original data (model vs observational): **"radar altimeter"**	X	X	X
history	String	Product creation date and time **(**YYYY-MM-DD HH:MM:SS : creation**)**	X	X	X
contact	String	A text giving the primary contact for information about the data set (ex. CNES aviso@oceanobs.com, EUMETSAT ops@eumetsat.int, ISRO : TBC): **[contact]**	X	X	X
references	String	The version of the altimetric library used to produce the data set (ex: Level1 library = v3.1, Level2 Library = v2.3p1, Processing Pilot = v2.3): **[GA_REF]**	X	X	X

			Data Set		
Attribute name	**Format**	**Description**	**Reduced**	**Standard**	**Expertise**
processing_center	String	Name of the processing center (SALP, EUMPC or ISRO TBC): **[GA_PROC_CENTRE]**	X	X	X
reference_document	String	Name of the reference document describing the products (ex. SARAL/ALTIKA Products Handbook, SALP-MU-M-OP-15984-CN): **[reference_document]**	X	X	X
mission_name	String	Name of the mission (ex. "SARAL"): **[mission_name]"**	X	X	X
altimeter_sensor_name	String	Name of the altimeter sensor (ex "ALTIKA"): **[altimeter_sensor_name]**	X	X	X
radiometer_sensor_name	String	Name of the radiometer sensor (ex "ALTIKA_RAD"): **[radiometer_sensor_name]**	X	X	X
doris_sensor_name	String	Name of the DORIS sensor (ex "DGXX") **[doris_sensor_name]**	X	X	X
cycle_number	long	Cycle number: **[GA_CYCLE_NB]**	X	X	X
absolute_rev_number	long	Absolute number of revolution: **[GA_ABS_REV_NB]**	X	X	X
pass_number	long	Pass number in the cycle (relative pass number): **[GA_PASS_NB]**	X	X	X
absolute_pass_number	long	Absolute pass number (since the beginning of the mission): **[GA_ABS_PASS_NB]**	X	X	X
equator_time	String	UTC time of equator crossing (YYYY-MM-DD HH:MM:SS.mmmmmm): **[GA_EQ_TIME]**	X	X	X
equator_longitude	double	Longitude of equator crossing: **[GA_EQ_LON]**	X	X	X
first_meas_time	String	UTC Date of the first measurement of the data set (YYYY-MM-DD HH:MM:SS.mmmmmm)	X	X	X
last_meas_time	String	UTC Date of the last measurement of the data set (YYYY-MM-DD HH:MM:SS.mmmmmm)	X	X	X
xref_altimeter_level1	String	Name of the altimeter level 1.0 product: **[GA_ALT_L1]**	X	X	X

			Data Set		
Attribute name	**Format**	**Description**	**Reduced**	**Standard**	**Expertise**
xref_radiometer_level1	String	Name of the radiometer level 1.0 product: **[GA_RAD_L1]**	X	X	X
xref_altimeter_characterisation	String	Name of the altimeter characterisation data file: **[IF_CHAR_ALT]**	X	X	X
xref_radiometer_characterisation	String	Name of the radiometer characterisation data file: **[IF_CHAR_RAD]**	X	X	X
xref_altimeter_ltm	String	Name of the altimeter Long Term Monitoring data file: **[GA_LTM]**	X	X	X
xref_doris_uso	String	Name of the file containing the DORIS-derived USO frequency: **[GA_USO]**	X	X	X
xref_orbit_data	String	Name of the file containing the orbit ephemeris (not applicable to OGDRs products): **[GA_ORB]**	X	X	X
xref_pf_data	String	Name of the file containing the platform data (mispointing, distance antenna-COG): **[GA_PLA]**	X	X	X
xref_pole_location	String	Name of the file containing the pole location data: **[GA_POL]**	X	X	X
xref_gim_data	String	Name of the ionospheric correction file used to create the pass file: **[GA_GIM]**	X	X	X
xref_mog2d_data	String	Name of the MOG2D correction files used to create the pass file (not applicable to OGDRs products): **[GA_MOG2D]**	X	X	X
xref_orf_data	String	Name of the Orbit Revolution File used to create the pass file: **[GA_ORF]**	X	X	X
xref_meteorological_files	String	Name of the meteorological files used to create the pass file: **[GA_METEO]**	X	X	X
ellipsoid_axis	String	Semi-major axis of the reference ellipsoid (from SAD : "UniversalConstants.txt") **[ellipsoid_sm_axis]**	X	X	X

			Data Set		
Attribute name	**Format**	**Description**	**Reduced**	**Standard**	**Expertise**
ellipsoid_flattening	String	Flattening coefficient of the reference ellipsoid (from SAD : "UniversalConstants.txt") **[ellipsoid _flattening]**	X	X	X

4. DATA SETS

4.1. REDUCED DATA SET

```
netcdf reduced {
  dimensions:
    time = < number of measurements >;

  variables:
```

// Time Tag

```
double time(time);
       time:long_name = "time (sec. since 2000-01-01)";
       time:standard_name = "time";
       time:units = "seconds since 2000-01-01 00:00:00.0";
       time:calendar = "gregorian";
       time:tai_utc_difference = [GA_TAI_UTC_DIF];
       time:leap_second = [GA_LEAP_TIME];
       time:comment = "[tai_utc_difference] is the difference between TAI and UTC
reference time (seconds) for the first measurement of the data set. [leap_second] is the
UTC time at which a leap second occurs in the data set, if any. After this UTC time, the
[tai_utc_difference] is increased by 1 second"
```

// Location and surface type

```
int lat(time);
       lat:long_name = "latitude";
       lat:standard_name = "latitude";
       lat:units = "degrees_north";
       lat:scale_factor = 1.00e-06;
       lat:comment = "Positive latitude is North latitude, negative latitude is South
latitude. See SARAL User Handbook. Associated quality flag is orb_state_flag_diode for
the OGDR products, orb_state_flag_rest for the IGDR and GDR products";

int lon(time);
       lon:long_name = "longitude";
       lon:standard_name = "longitude";
       lon:units = "degrees_east";
       lon:scale_factor = 1.00e-06;
       lon:comment = "East longitude relative to Greenwich meridian. See SARAL User
Handbook. Associated quality flag is orb_state_flag_diode for the OGDR products,
orb_state_flag_rest for the IGDR and GDR products";

byte surface_type(time);
       surface_type:long_name = "surface type";
       surface_type:_FillValue = 127b;
       surface_type:flag_values = 0b, 1b, 2b, 3b ;
       surface_type:flag_meanings = "ocean lake_enclosed_sea ice land";
       surface_type:coordinates = "lon lat";
       surface_type:comment = "Computed using a DTM2000 file: 0 = open oceans or semi-
enclosed seas; 1 = enclosed seas or lakes; 2 = continental ice; 3 = land. See SARAL User
Handbook";


byte rad_surf_type(time);
       rad_surf_type:long_name = "radiometer surface type";
```

```
       rad_surf_type:_FillValue = 127b;
       rad_surf_type:flag_values = 0b, 1b;
       rad_surf_type:flag_meanings = "ocean land";
       rad_surf_type:coordinates = "lon lat";
       rad_surf_type:comment = "Set to 1 = land if the K-band (first channel) radiometer
land fraction is greater than a given threshold";
```

// Quality information

```
byte ecmwf_meteo_map_avail(time);
       ecmwf_meteo_map_avail:long_name = "ECMWF meteorological map availability";
       ecmwf_meteo_map_avail:_FillValue = 127b;
       ecmwf_meteo_map_avail:flag_values = 0b, 1b, 2b, 3b ;
       ecmwf_meteo_map_avail:flag_meanings = "2_maps_nominal 2_maps_degraded
1_map_extrapolated no_map";
       ecmwf_meteo_map_avail:coordinates = "lon lat";
       ecmwf_meteo_map_avail:comment = ”Possible values are: 0 meaning ’2 maps, nominal’
(six hours apart), 1 meaning ’2 maps, degraded’ (more than six hours apart), 2 meaning ’1
map, extrapolation used’, 3 meaning ’no map’";

byte trailing_edge_variation_flag(time);
       trailing_edge_variation_flag:long_name = "1 Hz trailing edge variation flag";
       trailing_edge_variation_flag:_FillValue = 127b;
       trailing_edge_variation_flag:flag_values = 0b, 1b;
       trailing_edge_variation_flag:flag_meanings = "non_short_scale_variation
short_scale_variation";
       trailing_edge_variation_flag:coordinates = "lon lat";
       trailing_edge_variation_flag:comment = "See SARAL User Handbook";

byte ice_flag(time);
       ice_flag:long_name = "ice flag";
       ice_flag:_FillValue = 127b;
       ice_flag:flag_values = 0b, 1b;
       ice_flag:flag_meanings = "no_ice ice";
       ice_flag:coordinates = "lon lat";
       ice_flag:comment = "See SARAL User Handbook";
```

// Orbit

```
int alt(time);
       alt:long_name = "1 Hz altitude of satellite";
       alt:standard_name = "height_above_reference_ellipsoid";
       alt:_FillValue = 2147483647;
       alt:units = "m";
       alt:add_offset = 8.000000e+05;
       alt:scale_factor = 1.00e-04;
       alt:coordinates = "lon lat";
       alt:comment = "Altitude of satellite above the reference ellipsoid. Associated
quality flag is orb_state_flag_diode for the OGDR products, orb_state_flag_rest for the
IGDR and GDR products";
```

// Altimeter range

```
int range(time);
       range:long_name = "1 Hz corrected altimeter range";
       range:standard_name = "altimeter_range”;
```

```
        range:_FillValue = 2147483647;
        range:units = "m";
        range:add_offset = 8.000000e+05;
        range:scale_factor = 1.00e-04;
        range:coordinates = "lon lat";
        range:comment = "All instrumental corrections included, i.e. distance antenna-COG
(cog_corr), USO drift correction (uso_corr), internal path correction
(internal_path_delay_corr), Doppler correction (doppler_corr), modeled instrumental
errors correction (modeled_instr_corr_range) and system bias";
```

// Altimeter range corrections

```
short model_dry_tropo_corr(time);
        model_dry_tropo_corr:long_name = "model dry tropospheric correction";
        model_dry_tropo_corr:standard_name =
"altimeter_range_correction_due_to_dry_troposphere";
        model_dry_tropo_corr:source = [mto_fields_source];
        model_dry_tropo_corr:institution = [mto_fields_institution];
        model_dry_tropo_corr:_FillValue = 32767s;
        model_dry_tropo_corr:units = "m";
        model_dry_tropo_corr:scale_factor = 1.00e-04;
        model_dry_tropo_corr:coordinates = "lon lat";
        model_dry_tropo_corr:comment = "Computed at the altimeter time-tag from the
interpolation of 2 meteorological fields that surround the altimeter time-tag. A dry
tropospheric correction must be added (negative value) to the instrument range to correct
this range measurement for dry tropospheric range delays of the radar pulse. See SARAL
User Handbook";

short rad_wet_tropo_corr(time);
        rad_wet_tropo_corr:long_name = "radiometer wet tropospheric correction";
        rad_wet_tropo_corr:standard_name =
"altimeter_range_correction_due_to_wet_troposphere";
        rad_wet_tropo_corr:source = [radiometer_sensor_name];
        rad_wet_tropo_corr:institution = [radiometer_sensor_institution];
        rad_wet_tropo_corr:_FillValue = 32767s;
        rad_wet_tropo_corr:units = "m";
        rad_wet_tropo_corr:scale_factor = 1.00e-04;
        rad_wet_tropo_corr:coordinates = "lon lat";
        rad_wet_tropo_corr:comment = "A wet tropospheric correction must be added
(negative value) to the instrument range to correct this range measurement for wet
tropospheric range delays of the radar pulse";

short iono_corr_gim(time);
        iono_corr_gim:long_name = "GIM ionospheric correction";
        iono_corr_gim:standard_name = "altimeter_range_correction_due_to_ionosphere";
        iono_corr_gim:institution = [GIM_institution];
        iono_corr_gim:_FillValue = 32767s;
        iono_corr_gim:units = "m";
        iono_corr_gim:scale_factor = 1.00e-04;
        iono_corr_gim:coordinates = "lon lat";
        iono_corr_gim:comment = "An ionospheric correction must be added (negative value)
to the instrument range to correct this range measurement for ionospheric range delays of
the radar pulse. See SARAL User Handbook";

short sea_state_bias(time);
        sea_state_bias:long_name = "sea state bias correction";
        sea_state_bias:standard_name =
"sea_surface_height_bias_due_to_sea_surface_roughness";
        sea_state_bias:source = [altimeter_ssb_source];
        sea_state_bias:institution = [altimeter_ssb_institution];
        sea_state_bias:_FillValue = 32767s;
        sea_state_bias:units = "m";
        sea_state_bias:scale_factor = 1.00e-04;
```

```
       sea_state_bias:coordinates = "lon lat";
       sea_state_bias:comment = "A sea state bias correction must be added (negative
value) to the instrument range to correct this range measurement for sea state delays of
the radar pulse. This element should not be used over land. See SARAL User Handbook";
```

// Significant waveheight

```
short swh(time);
       swh:long_name = "Corrected significant waveheight";
       swh:standard_name = "sea_surface_wave_significant_height";
       swh:_FillValue = 32767s;
       swh:units = "m";
       swh:scale_factor = 1.00e-03;
       swh:coordinates = "lon lat";
       swh:comment = "All instrumental corrections included, i.e. modeled instrumental
errors correction (modeled_instr_corr_swh) and system bias";
```

// Backscatter coefficient

```
short sig0(time);
       sig0:long_name = "Corrected backscatter coefficient";
       sig0:standard_name = "surface_backwards_scattering_coefficient_of_radar_wave";
       sig0:_FillValue = 32767s;
       sig0:units = "dB";
       sig0:scale_factor = 1.00e-02;
       sig0:coordinates = "lon lat";
       sig0:comment = "All instrumental corrections included, excepted the system bias,
i.e. AGC instrumental errors correction, internal calibration correction
(internal_corr_sig0), modeled instrumental errors correction (modeled_instr_corr_sig0)
and atmospheric attenuation (atmos_corr_sig0)"
```

// Geophysical parameters

```
int mean_sea_surface(time);
       mean_sea_surface:long_name = "mean sea surface height above reference ellipsoid";
       mean_sea_surface:source = [mean_sea_surface_source];
       mean_sea_surface:institution = [mean_sea_surface_institution];
       mean_sea_surface:_FillValue = 2147483647;
       mean_sea_surface:units = "m";
       mean_sea_surface:scale_factor = 1.00e-04;
       mean_sea_surface:coordinates = "lon lat";
       mean_sea_surface:comment = "See SARAL User Handbook";

int mean_topography(time);
       mean_topography:long_name = "mean dynamic topography above geoid";
       mean_topography:source = [mdt_source];
       mean_topography:institution = [mdt_institution];
       mean_topography:_FillValue = 2147483647;
       mean_topography:units = "m";
       mean_topography:scale_factor = 1.00e-04;
       mean_topography:coordinates = "lon lat";
       mean_topography:comment = "See SARAL User Handbook";

int bathymetry(time);
       bathymetry:long_name = "ocean depth/land elevation";
       bathymetry:source = [bathy_topo_source];
       bathymetry:institution = [bathy_topo_institution];
       bathymetry:_FillValue = 2147483647;
       bathymetry:units = "m";
       bathymetry:coordinates = "lon lat";
```

short inv_bar_corr(time);
inv_bar_corr:long_name = "inverted barometer height correction";
inv_bar_corr:standard_name = "sea_surface_height_correction_due_to_air_pressure_at_low_frequency";
inv_bar_corr:source = [mto_fields_source];
inv_bar_corr:institution = [mto_fields_institution];
inv_bar_corr:_FillValue = 32767s;
inv_bar_corr:units = "m";
inv_bar_corr:scale_factor = 1.00e-04;
inv_bar_corr:coordinates = "lon lat";
inv_bar_corr:comment = "Computed at the altimeter time-tag from the interpolation of 2 meteorological fields that surround the altimeter time-tag. See SARAL User Handbook";

short hf_fluctuations_corr(time);
hf_fluctuations_corr:long_name = "high frequency fluctuations of the sea surface topography";
hf_fluctuations_corr:standard_name = "sea_surface_height_correction_due_to_air_pressure_and_wind_at_high_frequency";
hf_fluctuations_corr:institution = [mog2d_institution];
hf_fluctuations_corr:_FillValue = 32767s;
hf_fluctuations_corr:units = "m";
hf_fluctuations_corr:scale_factor = 1.00e-04;
hf_fluctuations_corr:coordinates = "lon lat";
hf_fluctuations_corr:comment = "Provided as a correction to the inverted barometer correction (inv_bar_corr)";

int ocean_tide_sol1(time);
ocean_tide_sol1:long_name = "geocentric ocean tide height (solution 1)";
ocean_tide_sol1:standard_name = "sea_surface_height_amplitude_due_to_geocentric_ocean_tide";
ocean_tide_sol1:source = [ocean_tide_sol1_source];
ocean_tide_sol1:institution = [ocean_tide_sol1_institution];
ocean_tide_sol1:_FillValue = 2147483647;
ocean_tide_sol1:units = "m";
ocean_tide_sol1:scale_factor = 1.00e-04;
ocean_tide_sol1:coordinates = "lon lat";
ocean_tide_sol1:comment = "Includes the loading tide (load_tide_sol1) and equilibrium long-period ocean tide height (ocean_tide_equil). The permanent tide (zero frequency) is not included in this parameter because it is included in the geoid and mean sea surface (geoid, mean_sea_surface). See SARAL User Handbook";

short solid_earth_tide(time);
solid_earth_tide:long_name = "solid earth tide height";
solid_earth_tide:standard_name = "sea_surface_height_amplitude_due_to_earth_tide";
solid_earth_tide:source = [solid_earth_tide_source];
solid_earth_tide:_FillValue = 32767s;
solid_earth_tide:units = "m";
solid_earth_tide:scale_factor = 1.00e-04;
solid_earth_tide:coordinates = "lon lat";
solid_earth_tide:comment = "Calculated using Cartwright and Tayler tables and consisting of the second and third degree constituents. The permanent tide (zero frequency) is not included. See SARAL User Handbook";

short pole_tide(time);
pole_tide:long_name = "geocentric pole tide height";
pole_tide:standard_name = "sea_surface_height_amplitude_due_to_pole_tide";
pole_tide:source = [pole_tide_source];
pole_tide:_FillValue = 32767s;
pole_tide:units = "m";
pole_tide:scale_factor = 1.00e-04;
pole_tide:coordinates = "lon lat";
pole_tide:comment = "See SARAL User Handbook”;

// Environmental parameters

```
short wind_speed_alt(time);
       wind_speed_alt:long_name = "altimeter wind speed";
       wind_speed_alt:standard_name = "wind_speed";
       wind_speed_alt:_FillValue = 32767s;
       wind_speed_alt:units = "m/s";
       wind_speed_alt:scale_factor = 1.00e-02;
       wind_speed_alt:coordinates = "lon lat";
       wind_speed_alt:comment = "Should not be used over land. See SARAL User Handbook";

short rad_water_vapor(time);
       rad_water_vapor:long_name = "radiometer water vapor content";
       rad_water_vapor:standard_name = "atmosphere_water_vapor_content";
       rad_water_vapor:source = [radiometer_sensor_name];
       rad_water_vapor:institution = [radiometer_sensor_institution];
       rad_water_vapor:_FillValue = 32767s;
       rad_water_vapor:units = "kg/m^2";
       rad_water_vapor:scale_factor = 1.00e-01;
       rad_water_vapor:coordinates = "lon lat";
       rad_water_vapor:comment = "Should not be used over land";

short rad_liquid_water(time);
       rad_liquid_water:long_name = "radiometer liquid water content";
       rad_liquid_water:standard_name = "atmosphere_cloud_liquid_water_content";
       rad_liquid_water:source = [radiometer_sensor_name];
       rad_liquid_water:institution = [radiometer_sensor_institution];
       rad_liquid_water:_FillValue = 32767s;
       rad_liquid_water:units = "kg/m^2";
       rad_liquid_water:scale_factor = 1.00e-02;
       rad_liquid_water:coordinates = "lon lat";
       rad_liquid_water:comment = "Should not be used over land";
```

// Sea Surface height

```
short ssha(time);
       ssha:long_name = "sea surface height anomaly";
       ssha:standard_name = "sea_surface_height_above_sea_level";
       ssha:source = [altimeter_sensor_name];
       ssha:institution = [altimeter_sensor_institution];
       ssha:_FillValue = 32767s;
       ssha:units = "m";
       ssha:scale_factor = 1.00e-03;
       ssha:coordinates = "lon lat";
       ssha:comment = "= altitude of satellite (alt) - corrected altimeter range (range)
- gim ionospheric correction (iono_corr_gim) - model dry tropospheric correction
(model_dry_tropo_corr) - model wet tropospheric correction (model_wet_tropo_corr) - sea
state bias correction (sea_state_bias) - solid earth tide height (solid_earth_tide) -
geocentric ocean tide height solution 1 (ocean_tide_sol1) - geocentric pole tide height
(pole_tide) - inverted barometer height correction (inv_bar_corr) - high frequency
fluctuations of the sea surface topography (hf_fluctuations_corr for I/GDR off line
products only) - mean sea surface (mean_sea_surface). Calculated even if iono_corr_gim is
at default value";

}
```

4.2. STANDARD DATA SET

```
netcdf standard {
  dimensions:
    time = < number of measurements >;
    meas_ind = 40;

  variables:
```

// Time Tag

```
double time(time);
       time:long_name = "time (sec. since 2000-01-01)";
       time:standard_name = "time";
       time:units = "seconds since 2000-01-01 00:00:00.0";
       time:calendar = "gregorian";
       time:tai_utc_difference = [GA_TAI_UTC_DIF];
       time:leap_second = [GA_LEAP_TIME];
       time:comment = "[tai_utc_difference] is the difference between TAI and UTC
reference time (seconds) for the first measurement of the data set. [leap_second] is the
UTC time at which a leap second occurs in the data set, if any. After this UTC time, the
[tai_utc_difference] is increased by 1 second";

byte meas_ind(meas_ind);
       meas_ind:long_name = "elementary measurement index";
       meas_ind:units = "count";
       meas_ind:comment = "Set to be compliant with the CF-1.1 convention";

double time_40hz(time,meas_ind);
       time_40hz:long_name = "time 40 Hz (sec. since 2000-01-01)";
       time_40hz:standard_name = "time";
       time_40hz:_FillValue = 18446744073709551616.000000;
       time_40hz:units = "seconds since 2000-01-01 00:00:00.0";
       time_40hz:calendar = "gregorian";
       time_40hz:tai_utc_difference = [GA_TAI_UTC_DIF];
       time_40hz:leap_second = [GA_LEAP_TIME];
       time_40hz:comment = "[tai_utc_difference] is the difference between TAI and UTC
reference time (seconds) for the first measurement of the data set. [leap_second] is the
UTC time at which a leap second occurs in the data set, if any. After this UTC time, the
[tai_utc_difference] is increased by 1 second";
```

// Location and surface type

```
int lat(time);
       lat:long_name = "latitude";
       lat:standard_name = "latitude";
       lat:units = "degrees_north";
       lat:scale_factor = 1.00e-06;
       lat:quality_flag = "orb_state_flag_rest or orb_state_flag_diode";
       lat:comment = "Positive latitude is North latitude, negative latitude is South
latitude. See SARAL User Handbook. Associated quality flag is orb_state_flag_diode for
the OGDR products, orb_state_flag_rest for the IGDR and GDR products";

int lon(time);
       lon:long_name = "longitude";
       lon:standard_name = "longitude";
       lon:units = "degrees_east";
       lon:scale_factor = 1.00e-06;
```

```
        lon:quality_flag = "orb_state_flag_rest or orb_state_flag_diode";
        lon:comment = "East longitude relative to Greenwich meridian. See SARAL User
Handbook. Associated quality flag is orb_state_flag_diode for the OGDR products,
orb_state_flag_rest for the IGDR and GDR products";

int lon_40hz(time,meas_ind);
        lon_40hz:long_name = "40 Hz longitude";
        lon_40hz:standard_name = "longitude";
        lon_40hz:_FillValue = 2147483647;
        lon_40hz:units = "degrees_east";
        lon_40hz:scale_factor = 1.00e-06;
        lon_40hz:comment = "East longitude relative to Greenwich meridian. See SARAL User
Handbook";

int lat_40hz(time,meas_ind);
        lat_40hz:long_name = "40 Hz latitude";
        lat_40hz:standard_name = "latitude";
        lat_40hz:_FillValue = 2147483647;
        lat_40hz:units = "degrees_north";
        lat_40hz:scale_factor = 1.00e-06;
        lat_40hz:comment = "Positive latitude is North latitude, negative latitude is
South latitude. See SARAL User Handbook";

byte surface_type(time);
        surface_type:long_name = "surface type";
        surface_type:_FillValue = 127b;
        surface_type:flag_values = 0b, 1b, 2b, 3b ;
        surface_type:flag_meanings = "ocean lake_enclosed_sea ice land";
        surface_type:coordinates = "lon lat";
        surface_type:comment = "Computed using a DTM2000 file: 0 = open oceans or semi-
enclosed seas; 1 = enclosed seas or lakes; 2 = continental ice; 3 = land. See SARAL User
Handbook";


byte rad_surf_type(time);
        rad_surf_type:long_name = "radiometer surface type";
        rad_surf_type:_FillValue = 127b;
        rad_surf_type:flag_values = 0b, 1b;
        rad_surf_type:flag_meanings = "ocean land";
        rad_surf_type:coordinates = "lon lat";
        rad_surf_type:comment = "Set to 1 = land if the K-band (first channel) radiometer
land fraction is greater than a given threshold";
```

// Quality information and sensor status

// Quality flags for 1Hz altimeter data

```
byte qual_alt_1hz_range(time);
        qual_alt_1hz_range:long_name = "quality flag for 1 Hz altimeter data: range";
        qual_alt_1hz_range:_FillValue = 127b;
        qual_alt_1hz_range:flag_values = 0b, 1b;
        qual_alt_1hz_range:flag_meanings = "good bad";
        qual_alt_1hz_range:coordinates = "lon lat";

byte qual_alt_1hz_swh(time);
        qual_alt_1hz_swh:long_name = "quality flag for 1 Hz altimeter data: SWH";
        qual_alt_1hz_swh:_FillValue = 127b;
        qual_alt_1hz_swh:flag_values = 0b, 1b;
        qual_alt_1hz_swh:flag_meanings = "good bad";
        qual_alt_1hz_swh:coordinates = "lon lat";

byte qual_alt_1hz_sig0(time);
```

```
      qual_alt_1hz_sig0:long_name = "quality flag for 1 Hz altimeter data: backscatter
coefficient";
      qual_alt_1hz_sig0:_FillValue = 127b;
      qual_alt_1hz_sig0:flag_values = 0b, 1b;
      qual_alt_1hz_sig0:flag_meanings = "good bad";
      qual_alt_1hz_sig0:coordinates = "lon lat";

byte qual_alt_1hz_off_nadir_angle_wf(time);
      qual_alt_1hz_off_nadir_angle_wf:long_name = "quality flag for 1 Hz altimeter data:
      off nadir angle from waveforms";
      qual_alt_1hz_off_nadir_angle_wf:_FillValue = 127b;
      qual_alt_1hz_off_nadir_angle_wf:flag_values = 0b, 1b;
      qual_alt_1hz_off_nadir_angle_wf:flag_meanings = "good bad";
      qual_alt_1hz_off_nadir_angle_wf:coordinates = "lon lat";
```

// Quality flags for 1 Hz altimeter instrumental corrections

```
byte qual_inst_corr_1hz_range(time);
      qual_inst_corr_1hz_range:long_name = "quality flag for 1 Hz instrumental
correction: range";
      qual_inst_corr_1hz_range:_FillValue = 127b;
      qual_inst_corr_1hz_range:flag_values = 0b, 1b;
      qual_inst_corr_1hz_range:flag_meanings = "good bad";
      qual_inst_corr_1hz_range:coordinates = "lon lat";

byte qual_inst_corr_1hz_swh(time);
      qual_inst_corr_1hz_swh:long_name = "quality flag for 1 Hz instrumental correction:
SWH";
      qual_inst_corr_1hz_swh:_FillValue = 127b;
      qual_inst_corr_1hz_swh:flag_values = 0b, 1b;
      qual_inst_corr_1hz_swh:flag_meanings = "good bad";
      qual_inst_corr_1hz_swh:coordinates = "lon lat";

byte qual_inst_corr_1hz_sig0(time);
      qual_inst_corr_1hz_sig0:long_name = "quality flag for 1 Hz instrumental
correction: backscatter coefficient";
      qual_inst_corr_1hz_sig0:_FillValue = 127b;
      qual_inst_corr_1hz_sig0:flag_values = 0b, 1b;
      qual_inst_corr_1hz_sig0:flag_meanings = "good bad";
      qual_inst_corr_1hz_sig0:coordinates = "lon lat";
```

// Quality flags for 1 Hz radiometer data

```
byte qual_rad_1hz_tb_k(time);
      qual_rad_1hz_tb_K:long_name = "quality flag for 1 Hz radiometer data: K band
(channel 1) brightness temperature";
      qual_rad_1hz_tb_k:_FillValue = 127b;
      qual_rad_1hz_tb_k:flag_values = 0b, 1b;
      qual_rad_1hz_tb_k:flag_meanings = "good bad";
      qual_rad_1hz_tb_k:coordinates = "lon lat";

byte qual_rad_1hz_tb_ka(time);
      qual_rad_1hz_tb_ka:long_name = "quality flag for 1 Hz radiometer data: Ka band
(channel 2) brightness temperature";
      qual_rad_1hz_tb_ka:_FillValue = 127b;
      qual_rad_1hz_tb_ka:flag_values = 0b, 1b;
      qual_rad_1hz_tb_ka:flag_meanings = "good bad";
      qual_rad_1hz_tb_ka:coordinates = "lon lat";
```

// Altimeter state flags

```
byte alt_state_flag_acq_mode_40hz(time,meas_ind);
       alt_state_flag_acq_mode_40hz:long_name = "40 Hz altimeter state flag: acquisition
mode";
       alt_state_flag_acq_mode_40hz:_FillValue = 127b;
       alt_state_flag_acq_mode_40hz:flag_values = 0b, 1b, 2b;
       alt_state_flag_acq_mode_40hz:flag_meanings = "autonomous_acq/track
autonomous_DIODEacq/track DIODE+DEM/track";
       alt_state_flag_acq_mode_40hz:coordinates = "lon_40hz lat_40hz";
       alt_state_flag_acq_mode_40hz:comment = "0 = autonomous acquisition / tracking, 1 =
autonomous DIODE acquisition / tracking, 2 = DIODE + Digital Elevation Model tracking";

byte alt_state_flag_tracking_mode_40hz(time,meas_ind);
       alt_state_flag_tracking_mode_40hz:long_name = "40 Hz altimeter state flag:
tracking mode";
       alt_state_flag_tracking_mode_40hz:_FillValue = 127b;
       alt_state_flag_tracking_mode_40hz:flag_values = 0b, 1b;
       alt_state_flag_tracking_mode_40hz:flag_meanings = "earliest_detectable_part
median_tracking";
       alt_state_flag_tracking_mode_40hz:coordinates = "lon_40hz lat_40hz";
       alt_state_flag_tracking_mode_40hz:comment = "0 =earliest detectable part, 1 =
median tracking";
```

// Orbit state flags

```
byte orb_state_flag_diode(time);
       orb_state_flag_diode:long_name = "orbit state flag: OGDR products";
       orb_state_flag_diode:_FillValue = 127b;
       orb_state_flag_diode:flag_values = 0b, 1b, 2b, 3b, 4b, 5b, 6b, 7b, 8b, 9b;
       orb_state_flag_diode:flag_meanings = "From good quality (0) to bad quality (9)";
       orb_state_flag_diode:coordinates = "lon lat";
       orb_state_flag_diode:comment = "0 = Accurate orbit (0 - 5 cm radial), 1 = Good
orbit (5 - 10 cm radial), 2 = Moderate orbit (10 - 15 cm radial), 4-8 = Potentially
degraded orbit (> 15 cm radial), 9 = Degraded orbit (e.g., as during maneuver). Set to
default for off-line products";

byte orb_state_flag_rest(time);
       orb_state_flag_rest:long_name = "orbit state flag: restituted orbit";
       orb_state_flag_rest:_FillValue = 127b;
       orb_state_flag_rest:flag_values = 0b, 1b, 2b, 3b, 4b, 5b, 6b, 7b, 8b, 9b;
       orb_state_flag_rest:flag_meanings = "op_maneuver op_adjusted op_extrapolated
pre_adjusted pre_maneuver pre_interpolated_gap pre_extrapolated_L1 pre_extrapolated_L1S2
pre_extrapolated_S2 DIODE";
       orb_state_flag_rest:coordinates = "lon lat";
       orb_state_flag_rest:comment = "0 characterizes a mission operations orbit that is
computed during a maneuver period, 1 stands for an adjusted mission operations orbit, 2
stands for an extrapolated mission operations orbit, 3 stands for an adjusted
(preliminary/precise) orbit, 4 indicates that the (preliminary/precise) orbit is
estimated during a maneuver period, 5 indicates that the (preliminary/precise) orbit is
interpolated over a tracking data gap, 6 means that the (preliminary/precise) orbit is
extrapolated for a duration less than 1 day, 7 means that the (preliminary/precise) orbit
is extrapolated for a duration that ranges from 1 day to 2 days, 8 means that the
(preliminary/precise) orbit is extrapolated for a duration larger than 2 days, or that
the orbit is extrapolated just after a maneuver, 9 stands for the DORIS DIODE navigator
orbit. The nominal value is 3. Set to default for NRT products";
```

// Geophysical flags

```
byte ecmwf_meteo_map_avail(time);
       ecmwf_meteo_map_avail:long_name = "ECMWF meteorological map availability";
```

```
       ecmwf_meteo_map_avail:_FillValue = 127b;
       ecmwf_meteo_map_avail:flag_values = 0b, 1b, 2b, 3b;
       ecmwf_meteo_map_avail:flag_meanings = "2_maps_nominal 2_maps_degraded
1_map_extropolated no_map";
       ecmwf_meteo_map_avail:coordinates = "lon lat";
       ecmwf_meteo_map_avail:comment = "Possible values are: 0 meaning '2 maps, nominal'
(six hours apart), 1 meaning '2 maps, degraded' (more than six hours apart), 2 meaning '1
map, extrapolation used', 3 meaning 'no map'";

byte trailing_edge_variation_flag(time);
       trailing_edge_variation_flag:long_name = "1 Hz trailing edge variation flag";
       trailing_edge_variation_flag:_FillValue = 127b;
       trailing_edge_variation_flag:flag_values = 0b, 1b;
       trailing_edge_variation_flag:flag_meanings = "non_short_scale_variation
short_scale_variation";
       trailing_edge_variation_flag:coordinates = "lon lat";
       trailing_edge_variation_flag:comment = "See SARAL User Handbook";

byte trailing_edge_variation_flag_40hz(time,meas_ind);
       trailing_edge_variation_flag_40hz:long_name = "40 Hz trailing edge variation
flag";
       trailing_edge_variation_flag_40hz:_FillValue = 127b;
       trailing_edge_variation_flag_40hz:flag_values = 0b, 1b;
       trailing_edge_variation_flag_40hz:flag_meanings = "non_short_scale_variation
short_scale_variation";
       trailing_edge_variation_flag_40hz:coordinates = "lon_40hz lat_40hz";
       trailing_edge_variation_flag_40hz:comment = "See SARAL User Handbook";

byte ice_flag(time);
       ice_flag:long_name = "ice flag";
       ice_flag:_FillValue = 127b;
       ice_flag:flag_values = 0b, 1b;
       ice_flag:flag_meanings = "no_ice ice";
       ice_flag:coordinates = "lon lat";
       ice_flag:comment = "See SARAL User Handbook";
```

// Quality flags for interpolation

```
byte interp_flag_mean_sea_surface(time);
       interp_flag_mean_sea_surface:long_name = "mean_sea_surface interpolation flag";
       interp_flag_mean_sea_surface:_FillValue = 127b;
       interp_flag_mean_sea_surface:flag_values = 0b, 1b;
       interp_flag_mean_sea_surface:flag_meanings = "good bad";
       interp_flag_mean_sea_surface:coordinates = "lon lat";

byte interp_flag_mdt(time);
       interp_flag_mdt:long_name = "MDT interpolation flag";
       interp_flag_mdt:_FillValue = 127b;
       interp_flag_mdt:flag_values = 0b, 1b;
       interp_flag_mdt:flag_meanings = "good bad";
       interp_flag_mdt:coordinates = "lon lat";

byte interp_flag_ocean_tide_sol1(time);
       interp_flag_ocean_tide_sol1:long_name = "ocean tide solution 1 interpolation
flag";
       interp_flag_ocean_tide_sol1:_FillValue = 127b;
       interp_flag_ocean_tide_sol1:flag_values = 0b, 1b;
       interp_flag_ocean_tide_sol1:flag_meanings = "good bad";
       interp_flag_ocean_tide_sol1:coordinates = "lon lat";
       interp_flag_ocean_tide_sol1:comment = "0 = 4 points over ocean; 1 = less than 4
points";

byte interp_flag_ocean_tide_sol2(time);
```

```
       interp_flag_ocean_tide_sol2:long_name = "ocean tide solution 2 interpolation
flag";
       interp_flag_ocean_tide_sol2:_FillValue = 127b;
       interp_flag_ocean_tide_sol2:flag_values = 0b, 1b;
       interp_flag_ocean_tide_sol2:flag_meanings = "good bad";
       interp_flag_ocean_tide_sol2:coordinates = "lon lat";
       interp_flag_ocean_tide_sol2:comment = "0 = 4 points over ocean; 1 = less than 4
points";

byte interp_flag_meteo(time);
       interp_flag_meteo:long_name = "meteorological data interpolation flag";
       interp_flag_meteo:_FillValue = 127b;
       interp_flag_meteo:flag_values = 0b, 1b;
       interp_flag_meteo:flag_meanings = "good bad";
       interp_flag_meteo:coordinates = "lon lat";
       interp_flag_meteo:comment = "0 = interpolation from 4 points; 1 = interpolation
from less than 4 points";
```

// Orbit

```
int alt(time);
       alt:long_name = "1 Hz altitude of satellite";
       alt: standard_name = "height_above_reference_ellipsoid";
       alt:_FillValue = 2147483647;
       alt:units = "m";
       alt:add_offset = 8.000000e+05;
       alt:scale_factor = 1.00e-04;
       alt:coordinates = "lon lat";
       alt:quality_flag = "orb_state_flag_rest or orb_state_flag_diode";
       alt:comment = "Altitude of satellite above the reference ellipsoid. Associated
quality flag is orb_state_flag_diode for the OGDR products, orb_state_flag_rest for the
IGDR and GDR products";

int alt_40hz(time,meas_ind);
       alt_40hz:long_name = "40 Hz altitude of satellite";
       alt_40hz:standard_name = "height_above_reference_ellipsoid";
       alt_40hz:_FillValue = 2147483647;
       alt_40hz:units = "m";
       alt_40hz:add_offset = 8.000000e+05;
       alt_40hz:scale_factor = 1.00e-04;
       alt_40hz:coordinates = "lon_40hz lat_40hz";
       alt_40hz:comment = "Altitude of satellite above the reference ellipsoid";

short orb_alt_rate(time);
       orb_alt_rate:long_name = "1 Hz orbital altitude rate";
       orb_alt_rate:_FillValue = 32767s;
       orb_alt_rate:units = "m/s";
       orb_alt_rate:scale_factor = 1.00e-02;
       orb_alt_rate:coordinates = "lon lat";
       orb_alt_rate:comment = "The reference surface for the orbital altitude rate is the
combined mean_sea_surface/geoid surface. It is used to compute the Doppler correction on
the altimeter range (doppler_corr)";
```

// Altimeter range

```
int range(time);
       range:long_name = "1 Hz corrected altimeter range";
       range:standard_name = "altimeter_range”;
       range:_FillValue = 2147483647;
       range:units = "m";
       range:add_offset = 8.000000e+05;
       range:scale_factor = 1.00e-04;
```

```
        range:coordinates = "lon lat";
        range:quality_flag = "qual_alt_1hz_range";
        range:comment = "All instrumental corrections included, i.e. distance antenna-COG
(cog_corr), USO drift correction (uso_corr), internal path correction
(internal_path_delay_corr), Doppler correction (doppler_corr), modeled instrumental
errors correction (modeled_instr_corr_range) and system bias";

int range_40hz(time,meas_ind);
        range_40hz:long_name = "40 Hz corrected altimeter range";
        range_40hz:standard_name = "altimeter_range";
        range_40hz:_FillValue = 2147483647;
        range_40hz:units = "m";
        range_40hz:add_offset = 8.000000e+05;
        range_40hz:scale_factor = 1.00e-04;
        range_40hz:coordinates = "lon_40hz lat_40hz";
        range_40hz:comment = "All instrumental corrections included, i.e. distance
antenna-COG (cog_corr), USO drift correction (uso_corr), internal path correction
(internal_path_delay_corr), Doppler correction (doppler_corr), modeled instrumental
errors correction (modeled_instr_corr_range) and system bias";

byte range_used_40hz(time,meas_ind);
        range_used_40hz:long_name = "40 Hz flag for utilization in the computation of 1Hz
range";
        range_used_40hz:_FillValue = 127b;
        range_used_40hz:flag_values = 0b, 1b;
        range_used_40hz:flag_meanings = "yes no";
        range_used_40hz:coordinates = "lon_40hz lat_40hz";
        range_used_40hz:comment = "Map of valid points used to compute the 1-Hz altimeter
range";

short range_rms(time);
        range_rms:long_name = "RMS of the range";
        range_rms:_FillValue = 32767s;
        range_rms:units = "m";
        range_rms:scale_factor = 1.00e-04;
        range_rms:coordinates = "lon lat";
        range_rms:comment = "Compression of high rate elements is preceded by a detection
of outliers. Only valid high-rate values are used to compute this element";

byte range_numval(time);
        range_numval:long_name = "number of valid points for range";
        range_numval:_FillValue = 127b;
        range_numval:units = "count";
        range_numval:coordinates = "lon lat";
        range_numval:valid_min = 0b;
        range_numval:valid_max = 40b;
```

// Ocean retracking outputs

```
byte number_of_iterations(time,meas_ind);
        number_of_iterations:long_name = "40 Hz number of iterations of the ocean
retracking";
        number_of_iterations:_FillValue = 127b;
        number_of_iterations:units = "count";
        number_of_iterations:coordinates = "lon_40Hz lat_40Hz";
```

// Altimeter range corrections

```
int net_instr_corr_range(time);
        net_instr_corr_range:long_name = "net instrumental correction on the range";
```

net_instr_corr_range:_FillValue = 2147483647;
net_instr_corr_range:units = "m";
net_instr_corr_range:scale_factor = 1.00e-04;
net_instr_corr_range:coordinates = "lon lat";
net_instr_corr_range:quality_flag = "qual_inst_corr_1hz_range";
net_instr_corr_range:comment = "Sum of distance antenna-COG (cog_corr), USO drift correction (uso_corr), internal path correction (internal_path_delay_corr), Doppler correction (doppler_corr), modeled instrumental errors correction (modeled_instr_corr_range) and system bias";

short model_dry_tropo_corr(time);
model_dry_tropo_corr:long_name = "model dry tropospheric correction";
model_dry_tropo_corr:standard_name = "altimeter_range_correction_due_to_dry_troposphere";
model_dry_tropo_corr:source = [mto_fields_source];
model_dry_tropo_corr:institution = [mto_fields_institution];
model_dry_tropo_corr:_FillValue = 32767s;
model_dry_tropo_corr:units = "m";
model_dry_tropo_corr:scale_factor = 1.00e-04;
model_dry_tropo_corr:coordinates = "lon lat";
model_dry_tropo_corr:quality_flag = "interp_flag_meteo";
model_dry_tropo_corr:comment = "Computed at the altimeter time-tag from the interpolation of 2 meteorological fields that surround the altimeter time-tag. A dry tropospheric correction must be added (negative value) to the instrument range to correct this range measurement for dry tropospheric range delays of the radar pulse. See SARAL User Handbook”;

short model_wet_tropo_corr(time);
model_wet_tropo_corr:long_name = "model wet tropospheric correction";
model_wet_tropo_corr:standard_name = "altimeter_range_correction_due_to_wet_troposphere";
model_wet_tropo_corr:source = [mto_fields_source];
model_wet_tropo_corr:institution = [mto_fields_institution];
model_wet_tropo_corr:_FillValue = 32767s;
model_wet_tropo_corr:units = "m";
model_wet_tropo_corr:scale_factor = 1.00e-04;
model_wet_tropo_corr:coordinates = "lon lat";
model_wet_tropo_corr:quality_flag = "interp_flag_meteo";
model_wet_tropo_corr:comment = "Computed at the altimeter time-tag from the interpolation of 2 meteorological fields that surround the altimeter time-tag. A wet tropospheric correction must be added (negative value) to the instrument range to correct this range measurement for wet tropospheric range delays of the radar pulse. See SARAL User Handbook";

short rad_wet_tropo_corr(time);
rad_wet_tropo_corr:long_name = "radiometer wet tropospheric correction";
rad_wet_tropo_corr:standard_name = "altimeter_range_correction_due_to_wet_troposphere";
rad_wet_tropo_corr:source = [radiometer_sensor_name];
rad_wet_tropo_corr:institution = [radiometer_sensor_institution];
rad_wet_tropo_corr:_FillValue = 32767s;
rad_wet_tropo_corr:units = "m";
rad_wet_tropo_corr:scale_factor = 1.00e-04;
rad_wet_tropo_corr:coordinates = "lon lat";
rad_wet_tropo_corr:quality_flag = "qual_rad_1hz_tb_k and qual_rad_1hz_tb_ka";
rad_wet_tropo_corr:comment = "A wet tropospheric correction must be added (negative value) to the instrument range to correct this range measurement for wet tropospheric range delays of the radar pulse”;

short iono_corr_gim(time);
iono_corr_gim:long_name = "GIM ionospheric correction";
iono_corr_gim:standard_name = "altimeter_range_correction_due_to_ionosphere";
iono_corr_gim:institution = [GIM_institution];
iono_corr_gim:_FillValue = 32767s;
iono_corr_gim:units = "m";

```
       iono_corr_gim:scale_factor = 1.00e-04;
       iono_corr_gim:coordinates = "lon lat";
       iono_corr_gim:comment = "An ionospheric correction must be added (negative value)
to the instrument range to correct this range measurement for ionospheric range delays of
the radar pulse. See SARAL User Handbook";

short sea_state_bias(time);
       sea_state_bias:long_name = "sea state bias correction";
       sea_state_bias:standard_name =
"sea_surface_height_bias_due_to_sea_surface_roughness";
       sea_state_bias:source = [altimeter_ssb_source];
       sea_state_bias:institution = [altimeter_ssb_institution];
       sea_state_bias:_FillValue = 32767s;
       sea_state_bias:units = "m";
       sea_state_bias:scale_factor = 1.00e-04;
       sea_state_bias:coordinates = "lon lat";
       sea_state_bias:comment = "A sea state bias correction must be added (negative
value) to the instrument range to correct this range measurement for sea state delays of
the radar pulse. This element should not be used over land. See SARAL User Handbook";
```

// Significant waveheight

```
short swh(time);
       swh:long_name = "Corrected significant waveheight";
       swh:standard_name = "sea_surface_wave_significant_height";
       swh:_FillValue = 32767s;
       swh:units = "m";
       swh:scale_factor = 1.00e-03;
       swh:coordinates = "lon lat";
       swh:quality_flag = "qual_alt_1hz_swh";
       swh:comment = "All instrumental corrections included, i.e. modeled instrumental
errors correction (modeled_instr_corr_swh) and system bias";

short swh_40hz(time,meas_ind);
       swh_40hz:long_name = "40 Hz corrected significant waveheight";
       swh_40hz:standard_name = "sea_surface_wave_significant_height";
       swh_40hz:_FillValue = 32767s;
       swh_40hz:units = "m";
       swh_40hz:scale_factor = 1.00e-03;
       swh_40hz:coordinates = "lon_40hz lat_40hz";
       swh_40hz:comment = "All instrumental corrections included, i.e. modeled
instrumental errors correction (modeled_instr_corr_swh) and system bias";

byte swh_used_40hz(time,meas_ind);
       swh_used_40hz:long_name = "40 Hz flag for utilization in the computation of 1Hz
significant waveheight";
       swh_used_40hz:_FillValue = 127b;
       swh_used_40hz:flag_values = 0b, 1b;
       swh_used_40hz:flag_meanings = "yes no";
       swh_used_40hz:coordinates = "lon_40hz lat_40hz";
       swh_used_40hz:comment = "Map of valid points used to compute the 1-Hz significant
waveheight";

short swh_rms(time);
       swh_rms:long_name = "RMS of the significant waveheight";
       swh_rms:_FillValue = 32767s;
       swh_rms:units = "m";
       swh_rms:scale_factor = 1.00e-03;
       swh_rms:coordinates = "lon lat";
       swh_rms:comment = "Compression of high rate elements is preceded by a detection of
outliers. Only valid high-rate values are used to compute this element";

byte swh_numval(time);
```

```
       swh_numval:long_name = "number of valid points used to compute significant
waveheight";
       swh_numval:_FillValue = 127b;
       swh_numval:units = "count";
       swh_numval:coordinates = "lon lat";
       swh_numval:valid_min = 0b;
       swh_numval:valid_max = 40b;
```

// Significant waveheight corrections

```
short net_instr_corr_swh(time);
       net_instr_corr_swh:long_name = "net instrumental correction on significant
waveheight";
       net_instr_corr_swh:_FillValue = 32767s;
       net_instr_corr_swh:units = "m";
       net_instr_corr_swh:scale_factor = 1.00e-03;
       net_instr_corr_swh:coordinates = "lon lat";
       net_instr_corr_swh:quality_flag = "qual_inst_corr_1hz_swh";
       net_instr_corr_swh:comment = "Sum of modeled instrumental errors correction
(modeled_instr_corr_swh) and system bias";
```

// Backscatter coefficient

```
short sig0(time);
       sig0:long_name = "Corrected backscatter coefficient ";
       sig0:standard_name = "surface_backwards_scattering_coefficient_of_radar_wave";
       sig0:_FillValue = 32767s;
       sig0:units = "dB";
       sig0:scale_factor = 1.00e-02;
       sig0:coordinates = "lon lat";
       sig0:quality_flag = "qual_alt_1hz_sig0";
       sig0:comment = "All instrumental corrections included, excepted the system bias,
i.e. AGC instrumental errors correction, internal calibration correction
(internal_corr_sig0), modeled instrumental errors correction (modeled_instr_corr_sig0)
and atmospheric attenuation (atmos_corr_sig0)”;

short sig0_40hz(time,meas_ind);
       sig0_40hz:long_name = "40 Hz corrected backscatter coefficient";
       sig0_40hz:standard_name =
"surface_backwards_scattering_coefficient_of_radar_wave";
       sig0_40hz:_FillValue = 32767s;
       sig0_40hz:units = "dB";
       sig0_40hz:scale_factor = 1.00e-02;
       sig0_40hz:coordinates = "lon_40hz lat_40hz";
       sig0_40hz:comment = "All instrumental corrections included, excepted the system
bias, i.e. AGC instrumental errors correction, internal calibration correction
(internal_corr_sig0), modeled instrumental errors correction (modeled_instr_corr_sig0)
and atmospheric attenuation (atmos_corr_sig0)";

byte sig0_used_40hz(time,meas_ind);
       sig0_used_40hz:long_name = "40 Hz flag for utilization in the computation of 1Hz
backscatter coefficient";
       sig0_used_40hz:_FillValue = 127b;
       sig0_used_40hz:flag_values = 0b, 1b;
       sig0_used_40hz:flag_meanings = "yes no";
       sig0_used_40hz:coordinates = "lon_40hz lat_40hz";
       sig0_used_40hz:comment = "Map of valid points used to compute the 1-Hz backscatter
coefficient";

short sig0_rms(time);
       sig0_rms:long_name = "RMS of the backscatter coefficient";
       sig0_rms:_FillValue = 32767s;
```

```
        sig0_rms:units = "dB";
        sig0_rms:scale_factor = 1.00e-02;
        sig0_rms:coordinates = "lon lat";
        sig0_rms:comment = "Compression of high rate elements is preceded by a detection
of outliers. Only valid high-rate values are used to compute this element";

byte sig0_numval(time);
        sig0_numval:long_name = "number of valid points used to compute backscatter
coefficient";
        sig0_numval:_FillValue = 127b;
        sig0_numval:units = "count";
        sig0_numval:coordinates = "lon lat";
        sig0_numval:valid_min = 0b;
        sig0_numval:valid_max = 40b;
```

// Tracker AGC

```
short agc(time);
        agc:long_name = "Corrected AGC";
        agc:_FillValue = 32767s;
        agc:units = "dB";
        agc:scale_factor = 1.00e-02;
        agc:coordinates = "lon lat";
        agc:comment = "AGC is corrected for instrumental errors due to the imperfections
of the on-board attenuators";

short agc_rms(time);
        agc_rms:long_name = "RMS of the AGC";
        agc_rms:_FillValue = 32767s;
        agc_rms:units = "dB";
        agc_rms:scale_factor = 1.00e-02;
        agc_rms:coordinates = "lon lat";
        agc_rms:comment = "Compression of high rate elements is preceded by a detection of
outliers. Only valid high-rate values are used to compute this element";

byte agc_numval(time);
        agc_numval:long_name = "number of valid points used to compute AGC";
        agc_numval:_FillValue = 127b;
        agc_numval:units = "count";
        agc_numval:coordinates = "lon lat";
        agc_numval:valid_min = 0b;
        agc_numval:valid_max = 40b;
```

// Backscatter coefficient corrections

```
short net_instr_corr_sig0(time);
       net_instr_corr_sig0:long_name = "net instrumental correction on backscatter
coefficient";
       net_instr_corr_sig0:_FillValue = 32767s;
       net_instr_corr_sig0:units = "dB";
       net_instr_corr_sig0:scale_factor = 1.00e-02;
       net_instr_corr_sig0:coordinates = "lon lat";
       net_instr_corr_sig0:quality_flag = "qual_inst_corr_1hz_sig0";
       net_instr_corr_sig0:comment = "Sum of AGC instrumental errors correction, internal
calibration correction (internal_corr_sig0) and modeled instrumental errors correction
(modeled_instr_corr_sig0) - system bias not included";

short atmos_corr_sig0(time);
       atmos_corr_sig0:long_name = "atmospheric attenuation correction on backscatter
coefficient";
       atmos_corr_sig0:_FillValue = 32767s;
       atmos_corr_sig0:units = "dB";
       atmos_corr_sig0:scale_factor = 1.00e-02;
       atmos_corr_sig0:coordinates = "lon lat";
```

// Off nadir angle

```
short off_nadir_angle_wf(time);
       off_nadir_angle_wf:long_name = "square of the off nadir angle computed from
waveforms";
       off_nadir_angle_wf:_FillValue = 32767s;
       off_nadir_angle_wf:units = "degrees^2";
       off_nadir_angle_wf:scale_factor = 1.00e-04;
       off_nadir_angle_wf:coordinates = "lon lat";
       off_nadir_angle_wf:quality_flag = "qual_alt_1hz_off_nadir_angle_wf";

short off_nadir_angle_wf_40hz(time,meas_ind);
       off_nadir_angle_wf_40hz:long_name = "40 Hz square of the off nadir angle computed
from waveforms";
       off_nadir_angle_wf_40hz:_FillValue = 32767s;
       off_nadir_angle_wf_40hz:units = "degrees^2";
       off_nadir_angle_wf_40hz:scale_factor = 1.00e-04;
       off_nadir_angle_wf_40hz:coordinates = "lon_40hz lat_40hz";
```

// Brightness temperatures

```
short tb_k(time);
       tb_k:long_name = "K band (channel 1) main beam brightness temperature";
       tb_k:standard_name = "surface_brightness_temperature";
       tb_k:_FillValue = 32767s;
       tb_k:units = "K";
       tb_k:scale_factor = 1.00e-02;
       tb_k:coordinates = "lon lat";
       tb_k:quality_flag = "qual_rad_1hz_tb_k";
       tb_k:comments = "Brightness temperatures at altimeter time tags are computed using
a weighted averaging window";

short tb_ka(time);
       tb_ka:long_name = "Ka band (channel 2) main beam brightness temperature";
       tb_ka:standard_name = "surface_brightness_temperature";
       tb_ka:_FillValue = 32767s;
```

```
        tb_ka:units = "K";
        tb_ka:scale_factor = 1.00e-02;
        tb_ka:coordinates = "lon lat";
        tb_ka:quality_flag = "qual_rad_1hz_tb_ka";
        tb_ka:comments = "Brightness temperatures at altimeter time tags are computed
using a weighted averaging window";
```

// Geophysical parameters

```
int mean_sea_surface(time);
        mean_sea_surface:long_name = "mean sea surface height above reference ellipsoid";
        mean_sea_surface:source = [mean_sea_surface_source];
        mean_sea_surface:institution = [mean_sea_surface_institution];
        mean_sea_surface:_FillValue = 2147483647;
        mean_sea_surface:units = "m";
        mean_sea_surface:scale_factor = 1.00e-04;
        mean_sea_surface:coordinates = "lon lat";
        mean_sea_surface:quality_flag = "interp_flag_mean_sea_surface";
        mean_sea_surface:comment = "See SARAL User Handbook";

int mean_topography(time);
        mean_topography:long_name = "mean dynamic topography above geoid";
        mean_topography:source = [mdt_source];
        mean_topography:institution = [mdt_institution];
        mean_topography:_FillValue = 2147483647;
        mean_topography:units = "m";
        mean_topography:scale_factor = 1.00e-04;
        mean_topography:coordinates = "lon lat";
        mean_topography:quality_flag = "interp_flag_mdt";
        mean_topography:comment = "See SARAL User Handbook";

int geoid(time);
        geoid:long_name = "geoid height";
        geoid:standard_name = "geoid_height_above_reference_ellipsoid";
        geoid:source = [geoid_source];
        geoid:institution = [geoid_institution];
        geoid:_FillValue = 2147483647;
        geoid:units = "m";
        geoid:scale_factor = 1.00e-04;
        geoid:coordinates = "lon lat";
        geoid:comment = "Computed from the geoid model with a correction to refer the
value to the mean tide system i.e. includes the permanent tide (zero frequency). See
SARAL User Handbook";

int bathymetry(time);
        bathymetry:long_name = "ocean depth/land elevation";
        bathymetry:source = [bathy_topo_source];
        bathymetry:institution = [bathy_topo_institution];
        bathymetry:_FillValue = 2147483647;
        bathymetry:units = "m";
        bathymetry:coordinates = "lon lat";

short inv_bar_corr(time);
        inv_bar_corr:long_name = "inverted barometer height correction";
        inv_bar_corr:standard_name =
"sea_surface_height_correction_due_to_air_pressure_at_low_frequency";
        inv_bar_corr:source = [mto_fields_source];
        inv_bar_corr:institution = [mto_fields_institution];
        inv_bar_corr:_FillValue = 32767s;
        inv_bar_corr:units = "m";
        inv_bar_corr:scale_factor = 1.00e-04;
        inv_bar_corr:coordinates = "lon lat";
```

```
      inv_bar_corr:quality_flag = "interp_flag_meteo";
      inv_bar_corr:comment = "Computed at the altimeter time-tag from the interpolation
of 2 meteorological fields that surround the altimeter time-tag. See SARAL User
Handbook";

short hf_fluctuations_corr(time);
      hf_fluctuations_corr:long_name = "high frequency fluctuations of the sea surface
topography";
      hf_fluctuations_corr:standard_name =
"sea_surface_height_correction_due_to_air_pressure_and_wind_at_high_frequency";
      hf_fluctuations_corr:institution = [mog2d_institution];
      hf_fluctuations_corr:_FillValue = 32767s;
      hf_fluctuations_corr:units = "m";
      hf_fluctuations_corr:scale_factor = 1.00e-04;
      hf_fluctuations_corr:coordinates = "lon lat";
      hf_fluctuations_corr:comment = "Provided as a correction to the inverted barometer
correction (inv_bar_corr)";

int ocean_tide_sol1(time);
      ocean_tide_sol1:long_name = "geocentric ocean tide height (solution 1)";
      ocean_tide_sol1:standard_name =
"sea_surface_height_amplitude_due_to_geocentric_ocean_tide";
      ocean_tide_sol1:source = [ocean_tide_sol1_source];
      ocean_tide_sol1:institution = [ocean_tide_sol1_institution];
      ocean_tide_sol1:_FillValue = 2147483647;
      ocean_tide_sol1:units = "m";
      ocean_tide_sol1:scale_factor = 1.00e-04;
      ocean_tide_sol1:coordinates = "lon lat";
      ocean_tide_sol1:quality_flag = "interp_flag_ocean_tide_sol1";
      ocean_tide_sol1:comment = "Includes the loading tide (load_tide_sol1) and
equilibrium long-period ocean tide height (ocean_tide_equil). The permanent tide (zero
frequency) is not included in this parameter because it is included in the geoid and mean
sea surface (geoid, mean_sea_surface). See SARAL User Handbook";

int ocean_tide_sol2(time);
      ocean_tide_sol2:long_name = "geocentric ocean tide height (solution 2)";
      ocean_tide_sol2:standard_name =
"sea_surface_height_amplitude_due_to_geocentric_ocean_tide";
      ocean_tide_sol2:source = [ocean_tide_sol2_source];
      ocean_tide_sol2:institution = [ocean_tide_sol2_institution];
      ocean_tide_sol2:_FillValue = 2147483647;
      ocean_tide_sol2:units = "m";
      ocean_tide_sol2:scale_factor = 1.00e-04;
      ocean_tide_sol2:coordinates = "lon lat";
      ocean_tide_sol2:quality_flag = "interp_flag_ocean_tide_sol2";
      ocean_tide_sol2:comment = "Includes the loading tide (load_tide_sol2) and
equilibrium long-period ocean tide height (ocean_tide_equil). The permanent tide (zero
frequency) is not included in this parameter because it is included in the geoid and mean
sea surface (geoid, mean_sea_surface). See SARAL User Handbook";

short ocean_tide_equil(time);
      ocean_tide_equil:long_name = "equilibrium long-period ocean tide height";
      ocean_tide_equil:standard_name =
"sea_surface_height_amplitude_due_to_equilibrium_ocean_tide";
      ocean_tide_equil:source = [ocean_tide_eq_source];
      ocean_tide_equil:_FillValue = 32767s;
      ocean_tide_equil:units = "m";
      ocean_tide_equil:scale_factor = 1.00e-04;
      ocean_tide_equil:coordinates = "lon lat";
      ocean_tide_equil:comment = "This value has already been added to the two
geocentric ocean tide height values recorded in the product (ocean_tide_sol1 and
ocean_tide_sol2). The permanent tide (zero frequency) is not included in this parameter
because it is included in the geoid and mean sea surface (geoid, mean_sea_surface). See
SARAL User Handbook";
```

```
short ocean_tide_non_equil(time);
       ocean_tide_non_equil:long_name = "non-equilibrium long-period ocean tide height";
       ocean_tide_non_equil:standard_name =
"sea_surface_height_amplitude_due_to_non_equilibrium_ocean_tide";
       ocean_tide_non_equil:source = [ocean_tide_neq_source];
       ocean_tide_non_equil:institution = [ocean_tide_neq_institution];
       ocean_tide_non_equil:_FillValue = 32767s;
       ocean_tide_non_equil:units = "m";
       ocean_tide_non_equil:scale_factor = 1.00e-04;
       ocean_tide_non_equil:coordinates = "lon lat";
       ocean_tide_non_equil:comment = "This parameter is computed as a correction to the
parameter ocean_tide_equil. This value can be added to ocean_tide_equil (or
ocean_tide_sol1, ocean_tide_sol2) so that the resulting value models the total non
equilibrium ocean tide height. See SARAL User Handbook";

short load_tide_sol1(time);
       load_tide_sol1:long_name = "load tide height for geocentric ocean tide (solution
1)";
       load_tide_sol1:source = [tidal_loading_sol1_source];
       load_tide_sol1:institution = [tidal_loading_sol1_institution];
       load_tide_sol1:_FillValue = 32767s;
       load_tide_sol1:units = "m";
       load_tide_sol1:scale_factor = 1.00e-04;
       load_tide_sol1:coordinates = "lon lat";
       load_tide_sol1:comment = "This value has already been added to the corresponding
ocean tide height value recorded in the product (ocean_tide_sol1). See SARAL User
Handbook";

short load_tide_sol2(time);
       load_tide_sol2:long_name = "load tide height for geocentric ocean tide (solution
2)";
       load_tide_sol2:source = [tidal_loading_sol2_source];
       load_tide_sol2:institution = [tidal_loading_sol2_institution];
       load_tide_sol2:_FillValue = 32767s;
       load_tide_sol2:units = "m";
       load_tide_sol2:scale_factor = 1.00e-04;
       load_tide_sol2:coordinates = "lon lat";
       load_tide_sol2:comment = "This value has already been added to the corresponding
ocean tide height value recorded in the product (ocean_tide_sol2). See SARAL User
Handbook";

short solid_earth_tide(time);
       solid_earth_tide:long_name = "solid earth tide height";
       solid_earth_tide:standard_name = "sea_surface_height_amplitude_due_to_earth_tide";
       solid_earth_tide:source = [solid_earth_tide_source];
       solid_earth_tide:_FillValue = 32767s;
       solid_earth_tide:units = "m";
       solid_earth_tide:scale_factor = 1.00e-04;
       solid_earth_tide:coordinates = "lon lat";
       solid_earth_tide:comment = "Calculated using Cartwright and Tayler tables and
consisting of the second and third degree constituents. The permanent tide (zero
frequency) is not included. See SARAL User Handbook";

short pole_tide(time);
       pole_tide:long_name = "geocentric pole tide height";
       pole_tide:standard_name = "sea_surface_height_amplitude_due_to_pole_tide";
       pole_tide:source = [pole_tide_source];
       pole_tide:_FillValue = 32767s;
       pole_tide:units = "m";
       pole_tide:scale_factor = 1.00e-04;
       pole_tide:coordinates = "lon lat";
       pole_tide:comment = "See SARAL User Handbook";
```

```
short wind_speed_model_u(time);
       wind_speed_model_u:long_name = "U component of the model wind vector";
       wind_speed_model_u:standard_name = "wind_speed";
       wind_speed_model_u:source = [mto_fields_source];
       wind_speed_model_u:institution = [mto_fields_institution];
       wind_speed_model_u:_FillValue = 32767s;
       wind_speed_model_u:units = "m/s";
       wind_speed_model_u:scale_factor = 1.00e-02;
       wind_speed_model_u:coordinates = "lon lat";
       wind_speed_model_u:quality_flag = "interp_flag_meteo and ecmwf_meteo_map_avail";
       wind_speed_model_u:comment = "Computed at the altimeter time-tag from the
interpolation of 2 meteorological fields that surround the altimeter time-tag. See SARAL
User Handbook";

short wind_speed_model_v(time);
       wind_speed_model_v:long_name = "V component of the model wind vector";
       wind_speed_model_v:standard_name = "wind_speed";
       wind_speed_model_v:source = [mto_fields_source];
       wind_speed_model_v:institution = [mto_fields_institution];
       wind_speed_model_v:_FillValue = 32767s;
       wind_speed_model_v:units = "m/s";
       wind_speed_model_v:scale_factor = 1.00e-02;
       wind_speed_model_v:coordinates = "lon lat";
       wind_speed_model_v:quality_flag = "interp_flag_meteo and ecmwf_meteo_map_avail";
       wind_speed_model_v:comment = "Computed at the altimeter time-tag from the
interpolation of 2 meteorological fields that surround the altimeter time-tag. See SARAL
User Handbook";

short wind_speed_alt(time);
       wind_speed_alt:long_name = "altimeter wind speed";
       wind_speed_alt:standard_name = "wind_speed";
       wind_speed_alt:_FillValue = 32767s;
       wind_speed_alt:units = "m/s";
       wind_speed_alt:scale_factor = 1.00e-02;
       wind_speed_alt:coordinates = "lon lat";
       wind_speed_alt:comment = "Should not be used over land. See SARAL User Handbook";

short rad_water_vapor(time);
       rad_water_vapor:long_name = "radiometer water vapor content";
       rad_water_vapor:standard_name = "atmosphere_water_vapor_content";
       rad_water_vapor:source = [radiometer_sensor_name];
       rad_water_vapor:institution = [radiometer_sensor_institution];
       rad_water_vapor:_FillValue = 32767s;
       rad_water_vapor:units = "kg/m^2";
       rad_water_vapor:scale_factor = 1.00e-01;
       rad_water_vapor:coordinates = "lon lat";
       rad_water_vapor:quality_flag = "qual_rad_1hz_tb_k and qual_rad_1hz_tb_ka";
       rad_water_vapor:comment = "Should not be used over land";

short rad_liquid_water(time);
       rad_liquid_water:long_name = "radiometer liquid water content";
       rad_liquid_water:standard_name = "atmosphere_cloud_liquid_water_content";
       rad_liquid_water:source = [radiometer_sensor_name];
       rad_liquid_water:institution = [radiometer_sensor_institution];
       rad_liquid_water:_FillValue = 32767s;
       rad_liquid_water:units = "kg/m^2";
       rad_liquid_water:scale_factor = 1.00e-02;
       rad_liquid_water:coordinates = "lon lat";
       rad_liquid water:quality_flag = "qual_rad_1hz_tb_k and qual_rad_1hz_tb_ka";
       rad_liquid water:comment = "Should not be used over land";
```

// Ice-1 retracking

```
int ice1_range_40hz(time,meas_ind);
       ice1_range_40hz:long_name = "40 Hz altimeter range (ice-1 retracking)";
       ice1_range_40hz:standard_name = "altimeter_range";
       ice1_range_40hz:_FillValue = 2147483647;
       ice1_range_40hz:units = "m";
       ice1_range_40hz:add_offset = 8.000000e+05;
       ice1_range_40hz:scale_factor = 1.00e-04;
       ice1_range_40hz:coordinates = "lon_40hz lat_40hz";
       ice1_range_40hz:comment = "Distance antenna-COG (cog_corr), USO drift correction
(uso_corr) and internal path correction (internal_path_delay_corr) included";

short ice1_sig0_40hz(time,meas_ind);
       ice1_sig0_40hz:long_name = "40 Hz backscatter coefficient (ice-1 retracking)";
       ice1_sig0_40hz:standard_name =
"surface_backwards_scattering_coefficient_of_radar_wave";
       ice1_sig0_40hz:_FillValue = 32767s;
       ice1_sig0_40hz:units = "dB";
       ice1_sig0_40hz:scale_factor = 1.00e-02;
       ice1_sig0_40hz:coordinates = "lon_40hz lat_40hz";
       ice1_sig0_40hz:comment = "AGC instrumental errors correction and internal
calibration correction (internal_corr_sig0) included";

byte ice1_qual_flag_40hz(time,meas_ind);
       ice1_qual_flag_40hz:long_name = "40 Hz ice-1 retracking quality flag";
       ice1_qual_flag_40hz:_FillValue = 127b;
       ice1_qual_flag_40hz:flag_values = 0b, 1b;
       ice1_qual_flag_40hz:flag_meanings = "good bad";
       ice1_qual_flag_40hz:coordinates = "lon_40hz lat_40hz" ;
       ice1_qual_flag_40hz:comment = "ice-1 retracking quality flag";
```

//Sea- Ice retracking

```
int seaice_range_40hz(time,meas_ind);
       seaice_range_40hz:long_name = "40 Hz altimeter range (sea-ice retracking)";
       seaice_range_40hz:standard_name = "altimeter_range";
       seaice_range_40hz:_FillValue = 2147483647;
       seaice_range_40hz:units = "m";
       seaice_range_40hz:add_offset = 8.000000e+05;
       seaice_range_40hz:scale_factor = 1.00e-04;
       seaice_range_40hz:coordinates = "lon_40hz lat_40hz";
       seaice_range_40hz:comment = "Distance antenna-COG (cog_corr), USO drift correction
(uso_corr) and internal path correction (internal_path_delay_corr) included";

short seaice_sig0_40hz(time,meas_ind);
       seaice_sig0_40hz:long_name = "40 Hz backscatter coefficient (sea-ice retracking)";
       seaice_sig0_40hz:standard_name =
"surface_backwards_scattering_coefficient_of_radar_wave";
       seaice_sig0_40hz:_FillValue = 32767s;
       seaice_sig0_40hz:units = "dB";
       seaice_sig0_40hz:scale_factor = 1.00e-02;
       seaice_sig0_40hz:coordinates = "lon_40hz lat_40hz";
       seaice_sig0_40hz:comment = "AGC instrumental errors correction and internal
calibration correction (internal_corr_sig0) included";

byte seaice_qual_flag_40hz(time,meas_ind);
       seaice_qual_flag_40hz:long_name = "40 Hz sea-ice retracking quality flag";
       seaice_qual_flag_40hz:_FillValue = 127b;
       seaice_qual_flag_40hz:flag_values = 0b, 1b;
       seaice_qual_flag_40hz:flag_meanings = "good bad";
       seaice_qual_flag_40hz:coordinates = "lon_40hz lat_40hz" ;
       seaice_qual_flag_40hz:comment = "sea-ice retracking quality flag";
```

// Ice-2 retracking

```
int ice2_range_40hz(time,meas_ind);
       ice2_range_40hz:long_name = "40 Hz altimeter range (ice-2 retracking)";
       ice2_range_40hz:standard_name = "altimeter_range”;
       ice2_range_40hz:_FillValue = 2147483647;
       ice2_range_40hz:units = "m";
       ice2_range_40hz:add_offset = 8.000000e+05;
       ice2_range_40hz:scale_factor = 1.00e-04;
       ice2_range_40hz:coordinates = "lon_40hz lat_40hz";
       ice2_range_40hz:comment = "Distance antenna-COG (cog_corr), USO drift correction
(uso_corr) and internal path correction (internal_path_delay_corr) included";

short ice2_le_sig0_40hz(time,meas_ind);
       ice2_le_sig0_40hz:long_name = "40 Hz leading edge backscatter coefficient (ice-2
retracking)";
       ice2_le_sig0_40hz:standard_name =
"surface_backwards_scattering_coefficient_of_radar_wave";
       ice2_le_sig0_40hz:_FillValue = 32767s;
       ice2_le_sig0_40hz:units = "dB";
       ice2_le_sig0_40hz:scale_factor = 1.00e-02;
       ice2_le_sig0_40hz:coordinates = "lon_40hz lat_40hz";
       ice2_le_sig0_40hz:comment = "AGC instrumental errors correction and internal
calibration correction (internal_corr_sig0) included";

short ice2_sig0_40hz(time,meas_ind);
       ice2_sig0_40hz:long_name = "40 Hz backscatter coefficient (ice-2 retracking)";
       ice2_sig0_40hz:standard_name =
"surface_backwards_scattering_coefficient_of_radar_wave";
       ice2_sig0_40hz:_FillValue = 32767s;
       ice2_sig0_40hz:units = "dB";
       ice2_sig0_40hz:scale_factor = 1.00e-02;
       ice2_sig0_40hz:coordinates = "lon_40hz lat_40hz";
       ice2_sig0_40hz:comment = "AGC instrumental errors correction and internal
calibration correction (internal_corr_sig0) included";

short ice2_sigmal_40hz(time,meas_ind);
       ice2_sigmal_40hz:long_name = "40 Hz width of the leading edge (ice-2 retracking)";
       ice2_sigmal_40hz:_FillValue = 32767;
       ice2_sigmal_40hz:units = "m";
       ice2_sigmal_40hz:scale_factor = 1.00e-3;
       ice2_sigmal_40hz:coordinates = "lon_40hz lat_40hz";
       ice2_sigmal_40hz:comment = "The width of the leading edge corresponds to the so-
called composite sigma (SigmaL)";

int ice2_slope1_40hz(time,meas_ind);
       ice2_slope1_40hz:long_name = "40 Hz slope of the first part of the logarithm of
the trailing edge (ice-2 retracking)";
       ice2_slope1_40hz:_FillValue = 2147483647;
       ice2_slope1_40hz:units = "s-1";
       ice2_slope1_40hz:coordinates = "lon_40hz lat_40hz";

int ice2_slope2_40hz(time,meas_ind);
       ice2_slope2_40hz:long_name = "40 Hz slope of the second part of the logarithm of
the trailing edge (ice-2 retracking)";
       ice2_slope2_40hz:_FillValue = 2147483647;
       ice2_slope2_40hz:units = "s-1";
       ice2_slope2_40hz:coordinates = "lon_40hz lat_40hz";

short ice2_mqe_40hz(time,meas_ind);
       ice2_mqe_40hz:long_name = "40 Hz MQE (ice-2 retracking)";
       ice2_mqe_40hz:_FillValue = 32767s;
       ice2_mqe_40hz:units = "count";
       ice2_mqe_40hz:scale_factor = 1.00e-04;
```

```
        ice2_mqe_40hz:coordinates = "lon_40hz lat_40hz";
        ice2_mqe_40hz:comment = "Mean Quadratic Error between the waveforms samples and
the corresponding model samples built from the ice-2 retracking outputs";

byte ice2_qual_flag_40hz(time,meas_ind);
        ice2_qual_flag_40hz:long_name = "40 Hz ice-2 retracking quality flag";
        ice2_qual_flag_40hz:_FillValue = 127b;
        ice2_qual_flag_40hz:flag_values = 0b, 1b;
        ice2_qual_flag_40hz:flag_meanings = "good bad";
        ice2_qual_flag_40hz:coordinates = "lon_40hz lat_40hz" ;
        ice2_qual_flag_40hz:comment = "ice-2 retracking quality flag";
```

// Waveforms characteristics

```
short mqe_40hz(time,meas_ind);
        mqe_40hz:long_name = "40 Hz MQE (ocean retracking)";
        mqe_40hz:_FillValue = 32767s;
        mqe_40hz:units = "count";
        mqe_40hz:scale_factor = 1.00e-04;
        mqe_40hz:coordinates = "lon_40hz lat_40hz";
        mqe_40hz:comment = "Mean Quadratic Error between the waveforms samples and the
corresponding model samples built from the ocean retracking outputs";

short peakiness_40hz(time,meas_ind);
        peakiness_40hz:long_name = "40 Hz peakiness on waveforms";
        peakiness_40hz:_FillValue = 32767s;
        peakiness_40hz:units = "count";
        peakiness_40hz:scale_factor = 1.00e-03;
        peakiness_40hz:coordinates = "lon_40hz lat_40hz";
```

// Sea Surface height

```
short ssha(time);
        ssha:long_name = "sea surface height anomaly";
        ssha:standard_name = "sea_surface_height_above_sea_level";
        ssha:source = [altimeter_sensor_name];
        ssha:institution = [altimeter_sensor_institution];
        ssha:_FillValue = 32767s;
        ssha:units = "m";
        ssha:scale_factor = 1.00e-03;
        ssha:coordinates = "lon lat";
        ssha:comment = "= altitude of satellite (alt) - corrected altimeter range (range)
- gim ionospheric correction (iono_corr_gim) - model dry tropospheric correction
(model_dry_tropo_corr) - model wet tropospheric correction (model_wet_tropo_corr) - sea
state bias correction (sea_state_bias) - solid earth tide height (solid_earth_tide) -
geocentric ocean tide height solution 1 (ocean_tide_sol1) - geocentric pole tide height
(pole_tide) - inverted barometer height correction (inv_bar_corr) - high frequency
fluctuations of the sea surface topography (hf_fluctuations_corr for I/GDR off line
products only) - mean sea surface (mean_sea_surface). Calculated even if iono_corr_gim
is at default value";

}
```

4.3. EXPERTISE DATA SET

All the variables described for the GDR data set are available in SGDR. Below are given the data available only in the SGDR data set.

```
netcdf expertise {
  dimensions:
time = < number of measurements >;
meas_ind = 40;
wvf_ind = 128;

  variables:
```

// Time Tag

```
double time(time);
       time:long_name = "time (sec. since 2000-01-01)";
       time:standard_name = "time";
       time:units = "seconds since 2000-01-01 00:00:00.0";
       time:calendar = "gregorian";
       time:tai_utc_difference = [GA_TAI_UTC_DIF];
       time:leap_second = [GA_LEAP_TIME];
       time:comment = "[tai_utc_difference] is the difference between TAI and UTC
reference time (seconds) for the first measurement of the data set. [leap_second] is the
UTC time at which a leap second occurs in the data set, if any. After this UTC time, the
[tai_utc_difference] is increased by 1 second";

byte meas_ind(meas_ind);
       meas_ind:long_name = "elementary measurement index";
       meas_ind:units = "count";
       meas_ind:comment = "Set to be compliant with the CF-1.1 convention";

byte wvf_ind(wvf_ind);
       wvf_ind:long_name = "waveform index";
       wvf_ind:units = "count";
       wvf_ind:comment = "Set to be compliant with the CF-1.1 convention";

double time_40hz(time,meas_ind);
       time_40hz:long_name = "time 40 Hz (sec. since 2000-01-01)";
       time_40hz:standard_name = "time";
       time_40hz:_FillValue = 18446744073709551616.000000;
       time_40hz:units = "seconds since 2000-01-01 00:00:00.0";
       time_40hz:calendar = "gregorian";
       time_40hz:tai_utc_difference = [GA_TAI_UTC_DIF];
       time_40hz:leap_second = [GA_LEAP_TIME];
       time_40hz:comment = "[tai_utc_difference] is the difference between TAI and UTC
reference time (seconds) for the first measurement of the data set. [leap_second] is the
UTC time at which a leap second occurs in the data set, if any. After this UTC time, the
[tai_utc_difference] is increased by 1 second";
```

// Cf. GDR product

…/… [cf. section 4.2]

// Tracker range

```
int tracker_40hz(time,meas_ind);
       tracker_40hz:long_name = "40 Hz corrected tracker range";
       tracker_40hz:standard_name = "altimeter_range";
       tracker_40hz:_FillValue = 2147483647;
       tracker_40hz:units = "m";
       tracker_40hz:add_offset = 8.000000e+05;
       tracker_40hz:scale_factor = 1.00e-04;
       tracker_40hz:coordinates = "lon_40hz lat_40hz";
       tracker_40hz:comment = "Operating tracker ('Diode+DEM' or 'Median' or 'Earliest
Detectable Part' tracker). This includes the Distance antenna-COG (cog_corr), USO drift
correction (uso_corr) and internal path correction (internal_path_delay_corr). But not
the Doppler correction (doppler_corr), modeled instrumental errors correction
(modeled_instr_corr_range) and system bias";

byte tracker_used_40hz(time,meas_ind);
       tracker_used_40hz:long_name = "40 Hz flag for utilization in the computation of
1Hz tracker range";
       tracker_used_40hz:_FillValue = 127b;
       tracker_used_40hz:flag_values = 0b, 1b;
       tracker_used_40hz:flag_meanings = "yes no";
       tracker_used_40hz:coordinates = "lon_40hz lat_40hz";
       tracker_used_40hz:comment = "Map of valid points used to compute the 1-Hz
altimeter tracker range";

int tracker_diode_40hz(time,meas_ind);
       tracker_diode_40hz:long_name = "40 Hz tracker range from Diode+DEM";
       tracker_diode_40hz:standard_name = "altimeter_range";
       tracker_diode_40hz:_FillValue = 2147483647;
       tracker_diode_40hz:units = "m";
       tracker_diode_40hz:add_offset = 8.000000e+05;
       tracker_diode_40hz:scale_factor = 1.00e-04;
       tracker_diode_40hz:coordinates = "lon_40hz lat_40hz";
```

// Altimeter PRI

```
int pri_counter_40hz(time,meas_ind);
       pri_counter_40hz:long_name = "40 Hz pulse repetition interval counter [12.5 ns]";
       pri_counter_40hz:_FillValue = 2147483647;
       pri_counter_40hz:units = "count";
       pri_counter_40hz:coordinates = "lon_40hz lat_40hz";
       pri_counter_40hz:comment = "Pulse repetition interval counter with a resolution of
12.5 ns";
```

// Off-nadir angle

```
byte qual_alt_1hz_off_nadir_angle_pf(time);
       qual_alt_1hz_off_nadir_angle_pf:long_name = "quality flag for 1 Hz altimeter data:
off nadir angle from platform";
       qual_alt_1hz_off_nadir_angle_pf:_FillValue = 127b;
       qual_alt_1hz_off_nadir_angle_pf:flag_values = 0b, 1b;
       qual_alt_1hz_off_nadir_angle_pf:flag_meanings = "good bad";
       qual_alt_1hz_off_nadir_angle_pf:coordinates = "lon lat";


short off_nadir_angle_pf(time);
       off_nadir_angle_pf:long_name = "square of the off nadir angle computed from
platform data";
       off_nadir_angle_pf:_FillValue = 32767s;
       off_nadir_angle_pf:units = "degrees^2";
```

```
        off_nadir_angle_pf:scale_factor = 1.00e-04;
        off_nadir_angle_pf:coordinates = "lon lat";
        off_nadir_angle_pf:quality_flag = "qual_alt_1hz_off_nadir_angle_pf";


short off_nadir_angle_rain_40hz(time,meas_ind);
        off_nadir_angle_rain_40hz:long_name = "40 Hz square of the off nadir angle
computed from waveforms for rain flag computation";
        off_nadir_angle_rain_40hz:_FillValue = 32767s;
        off_nadir_angle_rain_40hz:units = "degrees^2";
        off_nadir_angle_rain_40hz:scale_factor = 1.00e-04;
        off_nadir_angle_rain_40hz:coordinates = "lon_40hz lat_40hz";
```

// Altimeter range corrections

```
int uso_corr(time);
        uso_corr:long_name = "USO frequency correction on altimeter range";
        uso_corr:_FillValue = 2147483647;
        uso_corr:units = "m";
        uso_corr:scale_factor = 1.00e-04;
        uso_corr:comment = "Correction of the USO frequency drift on the altimeter range";

int internal_path_delay_corr(time);
        internal_path_delay_corr:long_name = "Internal path delay correction on altimeter
range";
        internal_path_delay_corr:_FillValue = 2147483647;
        internal_path_delay_corr:units = "m";
        internal_path_delay_corr:scale_factor = 1.00e-04;
        internal_path_delay_corr:comment = "Internal calibration correction on the
altimeter range";

short modeled_instr_corr_range(time);
        modeled_instr_corr_range:long_name = "Modeled instrumental correction on altimeter
range";
        modeled_instr_corr_range:_FillValue = 32767s;
        modeled_instr_corr_range:units = "m";
        modeled_instr_corr_range:scale_factor = 1.00e-04;

short doppler_corr(time);
        doppler_corr:long_name = "Doppler correction on altimeter range";
        doppler_corr:_FillValue = 32767s;
        doppler_corr:units = "m";
        doppler_corr:scale_factor = 1.00e-04;

short cog_corr(time);
        cog_corr:long_name = "Distance antenna-COG correction on altimeter range";
        cog_corr:_FillValue = 32767s;
        cog_corr:units = "m";
        cog_corr:scale_factor = 1.00e-04;
```

// Significant waveheight corrections

```
short modeled_instr_corr_swh(time);
        modeled_instr_corr_swh:long_name = "Modeled instrumental correction on significant
waveheight";
        modeled_instr_corr_swh:_FillValue = 32767s;
        modeled_instr_corr_swh:units = "m";
        modeled_instr_corr_swh:scale_factor = 1.00e-03;
```

// Backscatter coefficient corrections

```
short internal_corr_sig0(time);
       internal_corr_sig0:long_name = "Internal calibration correction on backscatter
coefficient";
       internal_corr_sig0:_FillValue = 32767s;
       internal_corr_sig0:units = "dB";
       internal_corr_sig0:scale_factor = 1.00e-02;

short modeled_instr_corr_sig0(time);
       modeled_instr_corr_sig0:long_name = "Modeled instrumental correction on
backscatter coefficient";
       modeled_instr_corr_sig0:_FillValue = 32767s;
       modeled_instr_corr_sig0:units = "dB";
       modeled_instr_corr_sig0:scale_factor = 1.00e-02;
```

// Tracker AGC

```
short agc_40hz(time,meas_ind);
       agc_40hz:long_name = "40 Hz corrected AGC";
       agc_40hz:_FillValue = 32767s;
       agc_40hz:units = "dB";
       agc_40hz:scale_factor = 1.00e-02;
       agc_40hz:coordinates = "lon_40Hz lat_40Hz";
       agc_40hz:comment = "AGC is corrected for instrumental errors due to the
imperfections of the on-board attenuators";

short agc_corr_40hz(time,meas_ind);
       agc_corr_40hz:long_name = "40 Hz AGC correction";
       agc_corr_40hz:_FillValue = 32767s;
       agc_corr_40hz:units = "dB";
       agc_corr_40hz:scale_factor = 1.00e-02;
       agc_corr_40hz:coordinates = "lon_40Hz lat_40Hz";
```

// Scaling factors for Sigma0 evaluation

```
int scaling_factor_40hz(time,meas_ind);
       scaling_factor_40hz:long_name = "Scaling factor for backscatter coefficient";
       scaling_factor_40hz:_FillValue = 2147483647;
       scaling_factor_40hz:units = "dB";
       scaling_factor_40hz:scale_factor = 1.00e-02;
       scaling_factor_40hz:coordinates = "lon_40hz lat_40hz";
       scaling_factor_40hz:comment = "This scaling factor represents the backscatter
coefficient for a waveform amplitude equal to 1. It accounts for all the parameters of
the radar equation excepted the amplitude of the waveform. It is a raw value accounting
for AGC 40Hz correction and internal calibration correction. All other correction are not
applied (ie atmospheric attenuation, modeled instrumental errors correction and system
bias)";
```

// Ocean retracking outputs

```
int epoch_40hz(time,meas_ind);
       epoch_40hz:long_name = "Epoch (ocean retracking)";
       epoch_40hz:_FillValue = 2147483647;
       epoch_40hz:units = "s";
       epoch_40hz:scale_factor = 1.00e-15;
       epoch_40hz:coordinates = "lon_40hz lat_40hz";

int width_leading_edge_40hz(time,meas_ind);
       width_leading_edge_40hz:long_name = "Width of the leading edge (ocean
retracking)";
```

```
       width_leading_edge_40hz:_FillValue = 2147483647;
       width_leading_edge_40hz:units = "s";
       width_leading_edge_40hz:scale_factor = 1.00e-15;
       width_leading_edge_40hz:coordinates = "lon_40hz lat_40hz";
       width_leading_edge_40hz:comment = "The width of the leading edge corresponds to
the so-called composite sigma (SigmaC)";

int amplitude_40hz(time,meas_ind);
       amplitude_40hz:long_name = "Amplitude (ocean retracking) [FFT power unit]";
       amplitude_40hz:_FillValue = 2147483647;
       amplitude_40hz:units = "count";
       amplitude_40hz:scale_factor = 1.00e-06;
       amplitude_40hz:coordinates = "lon_40hz lat_40hz";

int thermal_noise_40hz(time,meas_ind);
       thermal_noise_40hz:long_name = "Thermal noise (ocean retracking) [FFT power
unit]";
       thermal_noise_40hz:_FillValue = 2147483647;
       thermal_noise_40hz:units = "count";
       thermal_noise_40hz:scale_factor = 1.00e-06;
       thermal_noise_40hz:coordinates = "lon_40hz lat_40hz";
```

//Sea-Ice retracking outputs

```
int seaice_epoch_40hz(time,meas_ind);
       seaice_epoch_40hz:long_name = "Epoch (sea-ice retracking)";
       seaice_epoch_40hz:_FillValue = 2147483647;
       seaice_epoch_40hz:units = "s";
       seaice_epoch_40hz:scale_factor = 1.00e-15;
       seaice_epoch_40hz:coordinates = "lon_40hz lat_40hz";

int seaice_amplitude_40hz(time,meas_ind);
       seaice_amplitude_40hz:long_name = "Amplitude (sea-ice retracking) [FFT power
unit]";
       seaice_amplitude_40hz:_FillValue = 2147483647;
       seaice_amplitude_40hz:units = "count";
       seaice_amplitude_40hz:scale_factor = 1.00e-06;
       seaice_amplitude_40hz:coordinates = "lon_40hz lat_40hz";
```

// Ice-2 retracking outputs

```
int ice2_epoch_40hz(time,meas_ind);
       ice2_epoch_40hz:long_name = "Epoch (ice-2 retracking)";
       ice2_epoch_40hz:_FillValue = 2147483647;
       ice2_epoch_40hz:units = "s";
       ice2_epoch_40hz:scale_factor = 1.00e-15;
       ice2_epoch_40hz:coordinates = "lon_40hz lat_40hz";

int ice2_amplitude_40hz(time,meas_ind);
       ice2_amplitude_40hz:long_name = "Amplitude (ice-2 retracking) [FFT power unit]";
       ice2_amplitude_40hz:_FillValue = 2147483647;
       ice2_amplitude_40hz:units = "count";
       ice2_amplitude_40hz:scale_factor = 1.00e-06;
       ice2_amplitude_40hz:coordinates = "lon_40hz lat_40hz";

int ice2_mean_amplitude_40hz(time,meas_ind);
       ice2_mean_amplitude_40hz:long_name = "Mean amplitude (ice-2 retracking) [FFT power
unit]";
       ice2_mean_amplitude_40hz:_FillValue = 2147483647;
```

```
       ice2_mean_amplitude_40hz:units = "count";
       ice2_mean_amplitude_40hz:scale_factor = 1.00e-06;
       ice2_mean_amplitude_40hz:coordinates = "lon_40hz lat_40hz";

int ice2_thermal_noise_40hz(time,meas_ind);
       ice2_thermal_noise_40hz:long_name = "Thermal noise (ice-2 retracking) [FFT power
unit]";
       ice2_thermal_noise_40hz:_FillValue = 2147483647;
       ice2_thermal_noise_40hz:units = "count";
       ice2_thermal_noise_40hz:scale_factor = 1.00e-06;
       ice2_thermal_noise_40hz:coordinates = "lon_40hz lat_40hz";

int ice2_slope_40hz(time,meas_ind);
        ice2_slope_40hz:long_name = "40 Hz slope of the logarithm of the trailing edge
for mispointing (ice-2 retracking)";
        ice2_slope_40hz:_FillValue = 2147483647;
        ice2_slope_40hz:units = "s-1";
        ice2_slope_40hz:coordinates = "lon_40hz lat_40hz";
```

// Waveforms characteristics

```
short signal_to_noise_ratio(time);
       signal_to_noise_ratio:long_name = "1 Hz signal-to-noise ratio (ocean retracking)";
       signal_to_noise_ratio:_FillValue = 32767s;
       signal_to_noise_ratio:units = "dB";
       signal_to_noise_ratio:scale_factor = 1.00e-02;
       signal_to_noise_ratio:coordinates = "lon lat";
```

// Waveforms

```
short waveforms_40hz(time,meas_ind,wvf_ind);
       waveforms_40hz:long_name = "Waveform samples";
       waveforms_40hz:_FillValue = 32767s;
       waveforms_40hz:units = "count";
       waveforms_40hz:comment = "Waveforms are not corrected for the Low Pass Filter
effects";

}
```

DIFFUSION

INTERNAL:

SENGENES Pierre	DCT/PO/AL
NOUBEL Jocelyne	DCT/PO/AL
PICOT Nicolas	DCT/PO/AL
GUINLE Thierry	DCT/ME/OT
BRONNER Emilie	DCT/ME/OT
QUEYRUT Olivier	DCT/PS/CMI
GARCIA Charlotte	DCT/PS/CMI
MENOT Frédéric	DCT/PS/CMI
STRZEPECK Aurélie	DCT/PS/CMI
GUILLOT Amandine	DCT/SI/IM

EXTERNAL:

S. D'ALESSIO	CLS/DOS
J.P. DUMONT	CLS/DOS
S. URIEN	CLS/DOS
G. DIBARBOURE	CLS/DOS
M. ABLAIN	CLS/DOS