Argo data management

DOI: <http://dx.doi.org/10.13155/29825>

Argo auxiliary files format for the Coriolis DAC

Version 1.1

February 17th, 2021





**Argo auxiliary files format for the Coriolis DAC**

**Author:** Jean-Philippe Rannou.

**How to cite this document**

Jean-Philippe Rannou. **Argo auxiliary files format for the Coriolis DAC.** <http://dx.doi.org/10.13155/29825>

**Table of contents**

[1 Introduction 6](#_Toc484696567)

[2 Formats description 6](#_Toc484696568)

[2.1 Metadata file format 6](#_Toc484696569)

[2.1.1 Global attributes 6](#_Toc484696570)

[2.1.2 Dimensions and definitions 7](#_Toc484696571)

[2.1.3 General information on the meta-data file 7](#_Toc484696572)

[2.1.4 Float characteristics 8](#_Toc484696573)

[2.1.5 Configuration parameters 8](#_Toc484696574)

[2.1.5.1 Static configuration parameters 9](#_Toc484696575)

[2.1.5.2 Launch configuration parameters 9](#_Toc484696576)

[2.1.5.3 Modified configuration parameters 10](#_Toc484696577)

[2.1.6 Float sensor information 10](#_Toc484696578)

[2.1.6.1 Float parameter information 11](#_Toc484696579)

[2.1.7 Float calibration information 11](#_Toc484696580)

[2.2 Technical data file format 12](#_Toc484696581)

[2.2.1 Global attributes 12](#_Toc484696582)

[2.2.2 Dimensions and definitions 12](#_Toc484696583)

[2.2.3 General information on the technical data file 13](#_Toc484696584)

[2.2.4 Technical data 14](#_Toc484696585)

[2.2.4.1 Set of unique technical measurement 14](#_Toc484696586)

[2.2.4.2 Series of technical parameter measurements 15](#_Toc484696587)

[2.3 Profile file format 17](#_Toc484696588)

[2.3.1 Global attributes 17](#_Toc484696589)

[2.3.2 Dimensions 17](#_Toc484696590)

[2.3.3 General information on the profile file 18](#_Toc484696591)

[2.3.4 General information for each profile 18](#_Toc484696592)

[2.3.5 Measurements for each profile 21](#_Toc484696593)

[2.3.6 Calibration information for each profile 22](#_Toc484696594)

[2.3.7 History information for each profile 22](#_Toc484696595)

[2.4 Trajectory file format 25](#_Toc484696596)

[2.4.1 Global attributes 25](#_Toc484696597)

[2.4.2 Dimensions and definitions 25](#_Toc484696598)

[2.4.3 General information on the trajectory file 26](#_Toc484696599)

[2.4.4 General information on the float 26](#_Toc484696600)

[2.4.5 N\_MEASUREMENT dimension variable group 27](#_Toc484696601)

[2.4.6 N\_CYCLE dimension variable group 29](#_Toc484696602)

[2.4.7 History information 30](#_Toc484696603)

[2.5 File localization and naming 32](#_Toc484696604)

[2.5.1 Coriolis auxiliary files localization 32](#_Toc484696605)

[2.5.2 Coriolis auxiliary files naming convention 32](#_Toc484696606)

[2.5.2.1 Metadata file 32](#_Toc484696607)

[2.5.2.2 Technical data file 32](#_Toc484696608)

[2.5.2.3 Profile data files 32](#_Toc484696609)

[2.5.2.4 Trajectory data file 32](#_Toc484696610)

[3 Reference tables 33](#_Toc484696611)

[3.1 Reference table AUX\_1: data type 33](#_Toc484696612)

[3.2 Reference table AUX\_3: parameter code table 33](#_Toc484696613)

[3.2.1 Reference table AUX\_3a: sensor parameter code table 33](#_Toc484696614)

[3.2.2 Reference table AUX\_3b: technical parameter code table 33](#_Toc484696615)

[3.3 Reference Table AUX\_25: SENSOR 33](#_Toc484696616)

[3.4 Reference Table AUX\_26: SENSOR\_MAKER 34](#_Toc484696617)

[3.5 Reference Table AUX\_27: SENSOR\_MODEL 34](#_Toc484696618)

History of the document

|  |  |  |
| --- | --- | --- |
| Version | Date | Comment |
| 1.0 | 09/02/2017 | JP Rannou : creation of the document |
| 1.1 | 08/06/2017  17/02/2021 | JP Rannou : DATA\_CENTER added in TECH\_AUX format (to fit with Matlab decoder version ‘012b’)  FLOAT\_META\_DATA\_VALUE(N\_FLOAT\_META\_DATA, STRING4096) instead of FLOAT\_META\_DATA\_VALUE(N\_FLOAT\_META\_DATA, STRING512) |
|  |  |  |

Reference documents

|  |  |  |
| --- | --- | --- |
| Reference# | Title | Link |
| RD1 | Argo user’s manual | <http://dx.doi.org/10.13155/29825> |
|  |  |  |

# Introduction

This document describes the format of the NetCDF files produced by the Coriolis DAC to store auxiliary data.

Auxiliary data are transmitted by Argo floats (or computed from transmitted data) but are too specific or outside the framework of Argo to be stored in the Argo NetCDF files.

The Argo floats of the Coriolis DAC produce up to four NetCDF files types (depending on float type, version and configuration):

* A meta data file
* A technical data file
* Individual profile data files
* A trajectory data file

The Argo user’s manual [RD1] should be read and understood before the present one.

# Formats description

## Metadata file format

### Global attributes

global attributes:

:title = "Argo float auxiliary metadata file"

:institution = "CORIOLIS"

:source = "Argo float"

:history = "2017-02-09T08:53:53Z creation; 2017-02-09T09:06:47Z last update (coriolis float real time data processing)"

:references = " "

:user\_manual\_version = "1.0"

:Conventions = "CF-1.6 Coriolis-Argo-Aux-1.0"

|  |  |
| --- | --- |
| Global attribute name | Definition |
| title | A succinct description of what is in the dataset. |
| institution | Specifies where the original data was produced. |
| source | The method of production of the original data. If it was model-generated, source should name the model and its version, as specifically as could be useful. If it is observational, source should characterize it (e.g., "surface observation" or "radiosonde"). |
| history | Provides an audit trail for modifications to the original data. Well-behaved generic NetCDF filters will automatically append their name and the parameters with which they were invoked to the global history attribute of an input NetCDF file. We recommend that each line begin with a timestamp indicating the date and time of day that the program was executed. |
| references | Published or web-based references that describe the data or methods used to produce it. |
| user\_manual\_version | The version number of the user manual. |
| Conventions | The conventions supported by this file, blank separated. |

### Dimensions and definitions

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| DATE\_TIME | DATE\_TIME = 14 | This dimension is the length of an ASCII date and time value.  Date\_time convention is : YYYYMMDDHHMISS  YYYY : year  MM : month  DD : day  HH : hour of the day  MI : minutes  SS : seconds  Date and time values are always in universal time coordinates (UTC).  Examples : 20010105172834 : January 5th 2001 17:28:34 19971217000000 : December 17th 1997 00:00:00 |
| STRING4096  STRING1024  STRING512  STRING256  STRING128  STRING64  STRING32  STRING16  STRING8  STRING4 | STRING4096 = 4096  STRING1024 = 1024  STRING512 = 512  STRING256 = 256  STRING128 = 128  STRING64 = 64  STRING32 = 32  STRING16 = 16  STRING8 = 8  STRING4 = 4 | String dimensions. |
| N\_FLOAT\_META\_DATA | N\_FLOAT\_META\_DATA = <int value> | Number of additional miscellaneous meta data. |
| N\_STATIC\_CONFIG\_PARAM | N\_STATIC\_CONFIG\_PARAM = <int value> | Number of static configuration parameters. |
| N\_LAUNCH\_CONFIG\_PARAM | N\_LAUNCH\_CONFIG\_PARAM = <int value> | Number of pre-deployment or launch configuration parameters. |
| N\_CONFIG\_PARAM | N\_CONFIG\_PARAM = <int value> | Number of configuration parameters. |
| N\_SENSOR | N\_SENSOR = <int value> | Number of sensors mounted on the float and used to measure the parameters. |
| N\_PARAM | N\_PARAM = <int value> | Number of parameters measured or calculated for a pressure sample. |
| N\_MISSIONS | N\_MISSIONS = UNLIMITED | Number of missions. |

### General information on the meta-data file

This section contains information about the whole file.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| DATA\_TYPE | char DATA\_TYPE(STRING16)  DATA\_TYPE:long\_name = "Data type"  DATA\_TYPE:conventions = "Reference table AUX\_1"  DATA\_TYPE:\_FillValue = " " | This field contains the type of data contained in the file.  The list of acceptable data types is in the reference table AUX\_1. |
| FORMAT\_VERSION | char FORMAT\_VERSION(STRING4)  FORMAT\_VERSION:long\_name = "File format version"  FORMAT\_VERSION:\_FillValue = " " | File format version.  Example : «1.0» |
| DATE\_CREATION | char DATE\_CREATION(DATE\_TIME)  DATE\_CREATION:long\_name = "Date of file creation"  DATE\_CREATION:conventions = "YYYYMMDDHHMISS"  DATE\_CREATION:\_FillValue = " " | Date and time (UTC) of creation of this file.  Format : YYYYMMDDHHMISS  Example : 20011229161700 : December 29th 2001 16:17:00 |
| DATE\_UPDATE | char DATE\_UPDATE(DATE\_TIME)  DATE\_UPDATE:long\_name = "Date of update of this file"  DATE\_UPDATE:conventions = "YYYYMMDDHHMISS"  DATE\_UPDATE:\_FillValue = " " | Date and time (UTC) of update of this file.  Format : YYYYMMDDHHMISS  Example : 20011230090500 : December 30th 2001 09:05:00 |

### Float characteristics

This section contains the main characteristics of the float.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| PLATFORM\_NUMBER | char PLATFORM\_NUMBER(STRING8)  PLATFORM\_NUMBER:long\_name = "Float unique identifier"  PLATFORM\_NUMBER:conventions = "WMO float identifier : A9IIIII"  PLATFORM\_NUMBER:\_FillValue = " " | WMO float identifier. WMO is the World Meteorological Organization. This platform number is unique.  Example : 6900045 |
| FLOAT\_SERIAL\_NO | char FLOAT\_SERIAL\_NO(STRING32)  FLOAT\_SERIAL\_NO:long\_name = "Serial number of the float"  FLOAT\_SERIAL\_NO:\_FillValue = " " | This field should contain only the serial number of the float.  Example 1679 |
| DATA\_CENTRE | char DATA\_CENTRE(STRING2)  DATA\_CENTRE:long\_name = "Data centre in charge of float data processing"  DATA\_CENTRE:conventions = "Argo reference table 4"  DATA\_CENTRE:\_FillValue = " " | Code of the data centre in charge of the float data management.  The data centre codes are described in the Argo reference table 4.  Example : IF for Ifremer |

The following variables are used to store additional miscellaneous metadata that are part of float characteristics.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| FLOAT\_META\_DATA\_NAME | char FLOAT\_META\_DATA\_NAME(N\_FLOAT\_META\_DATA, STRING128)  FLOAT\_META\_DATA\_NAME:long\_name = "Name of miscellaneous float metadata"  FLOAT\_META\_DATA\_NAME:\_FillValue = " " | Name of miscellaneous float metadata. |
| FLOAT\_META\_DATA\_VALUE | char FLOAT\_META\_DATA\_VALUE(N\_FLOAT\_META\_DATA, STRING4096)  FLOAT\_META\_DATA\_VALUE:long\_name = "Value of miscellaneous float metadata"  FLOAT\_META\_DATA\_VALUE:\_FillValue = " " | Value of miscellaneous float metadata. |
| FLOAT\_META\_DATA\_DESCRIPTION | char FLOAT\_META\_DATA\_DESCRIPTION (N\_FLOAT\_META\_DATA, STRING1024)  FLOAT\_META\_DATA\_DESCRIPTION:long\_name = "Description of miscellaneous float metadata"  FLOAT\_META\_DATA\_DESCRIPTION:\_FillValue = " " | Description of miscellaneous float metadata. |

### Configuration parameters

This section describes the configuration parameters for a float. It is important to note that configuration parameters are float settings, not measurements reported by the float.

Configuration parameters may or may not be reported by a float.

Configuration parameter names are identified by the “CONFIG” prefix.

#### Static configuration parameters

Static configuration parameters are set before deployment; they could not be modified at sea once the float started its mission.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| STATIC\_CONFIG\_PARAMETER\_NAME | char STATIC\_CONFIG\_PARAMETER\_NAME(N\_STATIC\_CONFIG\_PARAM, STRING128)  STATIC\_CONFIG\_PARAMETER\_NAME:long\_name = ”Name of static parameter”  STATIC\_CONFIG\_PARAMETER\_NAME:\_FillValue = " " | Name of static configuration parameter. |
| STATIC\_CONFIG\_PARAMETER\_VALUE | char STATIC\_CONFIG\_PARAMETER\_VALUE(N\_STATIC\_CONFIG\_PARAM, STRING128)  STATIC\_CONFIG\_PARAMETER\_VALUE:long\_name = ”Value of static parameter”  STATIC\_CONFIG\_PARAMETER\_VALUE:\_FillValue = " " | Value of static configuration parameter. |
| STATIC\_CONFIG\_PARAMETER\_DESCRIPTION | char STATIC\_CONFIG\_PARAMETER\_DESCRIPTION(N\_STATIC\_CONFIG\_PARAM, STRING1024)  STATIC\_CONFIG\_PARAMETER\_DESCRIPTION:long\_name = ”Description of static parameter”  STATIC\_CONFIG\_PARAMETER\_DESCRIPTION:\_FillValue = " " | Description of static configuration parameter. |

#### Launch configuration parameters

Launch configuration parameters are set before deployment; they could be modified at sea once the float started its mission.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| LAUNCH\_CONFIG\_PARAMETER\_NAME | char LAUNCH\_CONFIG\_PARAMETER\_NAME(N\_LAUNCH\_CONFIG\_PARAM, STRING128)  LAUNCH\_CONFIG\_PARAMETER\_NAME:long\_name = ”Name of configuration parameter at launch”  LAUNCH\_CONFIG\_PARAMETER\_NAME:\_FillValue = " " | Name of configuration parameter at launch. |
| LAUNCH\_CONFIG\_PARAMETER\_VALUE | double LAUNCH\_CONFIG\_PARAMETER\_VALUE(N\_LAUNCH\_CONFIG\_PARAM)  LAUNCH\_CONFIG\_PARAMETER\_VALUE:long\_name = ”Value of configuration parameter at launch”  LAUNCH\_CONFIG\_ PARAMETER\_VALUE:\_FillValue = 999999 | Value of configuration parameter at launch. |
| LAUNCH\_CONFIG\_PARAMETER\_DESCRIPTION | char LAUNCH\_CONFIG\_PARAMETER\_DESCRIPTION(N\_LAUNCH\_CONFIG\_PARAM, STRING1024)  LAUNCH\_CONFIG\_PARAMETER\_DESCRIPTION:long\_name = ”Description of configuration parameter at launch”  LAUNCH\_CONFIG\_PARAMETER\_DESCRIPTION:\_FillValue = " " | Description of configuration parameter at launch. |

#### Modified configuration parameters

These configuration parameters are part of configuration parameters at launch that have been modified at sea during the float mission.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| CONFIG\_PARAMETER\_NAME | char CONFIG\_PARAMETER\_NAME(N\_CONFIG\_PARAM, STRING128)  CONFIG\_PARAMETER\_NAME:long\_name = ”Name of configuration parameter”  CONFIG\_PARAMETER\_NAME:\_FillValue = " " | Name of configuration parameter. |
| CONFIG\_PARAMETER\_VALUE | double CONFIG\_PARAMETER\_VALUE(N\_MISSIONS, N\_CONFIG\_PARAM)  CONFIG\_PARAMETER\_VALUE:long\_name = ”Value of configuration parameter”  CONFIG\_PARAMETER\_VALUE:\_FillValue = 999999 | Value of configuration parameter. |
| CONFIG\_MISSION\_NUMBER | int CONFIG\_MISSION\_NUMBER(N\_MISSIONS)  CONFIG\_MISSION\_NUMBER:long\_name = "Unique number denoting the missions performed by the float"  CONFIG\_MISSION\_NUMBER:conventions = "1...N, 1 : first complete mission"  CONFIG\_MISSION\_NUMBER:\_FillValue = 99999 | Unique number of the mission to which this parameter belongs. |
| CONFIG\_MISSION\_COMMENT | char CONFIG\_MISSION\_COMMENT(N\_MISSIONS, STRING256)  CONFIG\_MISSION\_COMMENT:long\_name = ”Comment on configuration”  CONFIG\_MISSION\_COMMENT:\_FillValue = " " | Comment on this configuration mission.  Example : “This mission follows a 1000 dbar meddie during parking” |

### Float sensor information

This section contains information about the sensors of the profiler.

|  |  |  |
| --- | --- | --- |
| **Name** | **Definition** | **Comment** |
| SENSOR | char SENSOR(N\_SENSOR, STRING32)  SENSOR:long\_name = "Name of the sensor mounted on the float"  SENSOR:conventions = "Reference table AUX\_25"  SENSOR:\_FillValue = " " | Names of the sensor mounted on the float. |
| SENSOR\_MAKER | char SENSOR\_MAKER(N\_SENSOR, STRING256)  SENSOR\_MAKER:long\_name = "Name of the sensor manufacturer"  SENSOR\_MAKER:conventions = "Reference table AUX\_26"  SENSOR\_MAKER:\_FillValue = " " | Name of the sensor manufacturer. |
| SENSOR\_MODEL | char SENSOR\_MODEL(N\_SENSOR, STRING256)  SENSOR\_MODEL:long\_name = "Type of sensor"  SENSOR\_MODEL:conventions = "Reference table AUX\_27"  SENSOR\_MODEL:\_FillValue = " " | Model of the sensor. |
| SENSOR\_SERIAL\_NO | char SENSOR\_SERIAL\_NO(N\_SENSOR, STRING16)  SENSOR\_SERIAL\_NO:long\_name = "Serial number of the sensor"  SENSOR\_SERIAL\_NO:\_FillValue = " " | Serial number of the sensor. |

#### Float parameter information

This section contains information about the parameters measured by the profiler or derived from profiler measurements.

|  |  |  |
| --- | --- | --- |
| **Name** | **Definition** | **Comment** |
| PARAMETER | char PARAMETER(N\_PARAM, STRING64)  PARAMETER:long\_name = "Name of parameter computed from float measurements”  PARAMETER:conventions = "Reference table AUX\_3a"  PARAMETER:\_FillValue = " " | Name of the parameter measured by float sensors or derived from float measurements. |
| PARAMETER\_SENSOR | char PARAMETER\_SENSOR(N\_PARAM, STRING128)  PARAMETER\_SENSOR:long\_name = "Name of the sensor that measures this parameter”  PARAMETER\_SENSOR:conventions = "Reference table AUX\_25"  PARAMETER\_SENSOR:\_FillValue = " " | Name of the sensor that measured the float parameter. |
| PARAMETER\_UNITS | char PARAMETER\_UNITS(N\_PARAM, STRING32)  PARAMETER\_UNITS:long\_name = "Units of value, accuracy and resolution of the parameter"  PARAMETER\_UNITS:\_FillValue = " " | Units of value, accuracy and resolution of the parameter. |
| PARAMETER\_ACCURACY | char PARAMETER\_ACCURACY(N\_PARAM, STRING32)  PARAMETER\_ACCURACY:long\_name = "Accuracy of the parameter"  PARAMETER\_ACCURACY:\_FillValue = " " | Accuracy of the parameter. |
| PARAMETER\_RESOLUTION | char PARAMETER\_RESOLUTION(N\_PARAM, STRING32)  PARAMETER\_RESOLUTION:long\_name = "Resolution of the parameter"  PARAMETER\_RESOLUTION:\_FillValue =" " | Resolution of the parameter returned by the sensor (note that this is not necessarily equivalent to the resolution of the parameter returned by the float through telemetry). |

### Float calibration information

This section contains information about the calibration of the profiler. The calibration described in this section is an instrumental calibration. The delayed mode calibration, based on a data analysis is described in the profile format.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Definition | Comment | |
| PREDEPLOYMENT\_CALIB \_EQUATION | char PREDEPLOYMENT\_CALIB\_EQUATION(N\_PARAM, STRING4096)  PREDEPLOYMENT\_CALIB\_EQUATION:long\_name = "Calibration equation for this parameter"  PREDEPLOYMENT\_CALIB\_EQUATION:\_FillValue = " " | Calibration equation for this parameter. | |
| PREDEPLOYMENT\_CALIB\_COEFFICIENT | char PREDEPLOYMENT\_CALIB\_COEFFICIENT(N\_PARAM, STRING4096)  PREDEPLOYMENT\_CALIB\_COEFFICIENT:long\_name = "Calibration coefficients for this equation"  PREDEPLOYMENT\_CALIB\_COEFFICIENT:\_FillValue = " " | Calibration coefficients for this equation. | |
| PREDEPLOYMENT\_CALIB\_COMMENT | char PREDEPLOYMENT\_CALIB\_COMMENT(N\_PARAM, STRING4096)  PREDEPLOYMENT\_CALIB\_COMMENT:long\_name = "Comment applying to this parameter calibration"  PREDEPLOYMENT\_CALIB\_COMMENT:\_FillValue = " " | | Comments applying to this parameter calibration. | |

## Technical data file format

### Global attributes

global attributes:

:title = "Argo float auxiliary technical data file"

:institution = "CORIOLIS"

:source = "Argo float"

:history = "2017-02-09T09:06:42Z creation; 2017-02-09T09:06:42Z last update (coriolis float real time data processing)"

:references = " "

:user\_manual\_version = "1.0"

:Conventions = "CF-1.6 Coriolis-Argo-Aux-1.0"

|  |  |
| --- | --- |
| Global attribute name | Definition |
| title | A succinct description of what is in the dataset. |
| institution | Specifies where the original data was produced. |
| source | The method of production of the original data. If it was model-generated, source should name the model and its version, as specifically as could be useful. If it is observational, source should characterize it (e.g., "surface observation" or "radiosonde"). |
| history | Provides an audit trail for modifications to the original data. Well-behaved generic NetCDF filters will automatically append their name and the parameters with which they were invoked to the global history attribute of an input NetCDF file. We recommend that each line begin with a timestamp indicating the date and time of day that the program was executed. |
| references | Published or web-based references that describe the data or methods used to produce it. |
| user\_manual\_version | The version number of the user manual. |
| Conventions | The conventions supported by this file, blank separated. |

### Dimensions and definitions

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| DATE\_TIME | DATE\_TIME = 14 | This dimension is the length of an ASCII date and time value.  Date and time values are always in universal time coordinates (UTC).  Date\_time convention is : YYYYMMDDHHMISS  YYYY : year  MM : month  DD : day  HH : hour of the day  MI : minutes  SS : seconds  Examples :  20010105172834 : January 5th 2001 17:28:34  19971217000000 : December 17th 1997 00:00:00 |
| STRING1024  STRING256  STRING64  STRING32  STRING8  STRING4  STRING2 | STRING1024 = 1024  STRING256 = 256  STRING64 = 64  STRING32 = 32  STRING8 = 8  STRING4 = 4  STRING2 = 2 | String dimensions. |
| N\_TECH\_AUX\_LABEL | N\_TECH\_AUX\_LABEL = <int value> | Number of auxiliary technical labels. |
| N\_TECH\_MEAS\_PARAM | N\_TECH\_MEAS\_PARAM = <int value> | Number of technical parameters providing series of measurements. |
| N\_TECH\_MEASUREMENT | N\_TECH\_MEASUREMENT = <int value> | Number of technical parameter measurements. |
| N\_TECH\_PARAM | N\_TECH\_PARAM = UNLIMITED | Number of technical parameters. |

### General information on the technical data file

This section contains information about the technical data file itself.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| PLATFORM\_NUMBER | char PLATFORM\_NUMBER(STRING8)  PLATFORM\_NUMBER:long\_name = "Float unique identifier"  PLATFORM\_NUMBER:conventions = "WMO float identifier : A9IIIII"  PLATFORM\_NUMBER:\_FillValue = " " | WMO float identifier.  WMO is the World Meteorological Organization.  This platform number is unique.  Example : 6900045 |
| DATA\_TYPE | char DATA\_TYPE(STRING32)  DATA\_TYPE:long\_name = "Data type"  DATA\_TYPE:conventions = "Reference table AUX\_1"  DATA\_TYPE:\_FillValue = " " | This field contains the type of data contained in the file.  The list of acceptable data types is in the reference table AUX\_1. |
| FORMAT\_VERSION | char FORMAT\_VERSION(STRING4)  FORMAT\_VERSION:long\_name = "File format version"  FORMAT\_VERSION:\_FillValue = " " | File format version  Example : «1.0» |
| REFERENCE\_DATE\_TIME | char REFERENCE\_DATE\_TIME(DATE\_TIME)  REFERENCE\_DATE\_TIME:long\_name = "Date of reference for Julian days"  REFERENCE\_DATE\_TIME:conventions = "YYYYMMDDHHMISS"  REFERENCE\_DATE\_TIME:\_FillValue = " " | Date of reference for julian days.  The recommended reference date time is “19500101000000” : January 1st 1950 00:00:00 |
| TECHNICAL\_MEASUREMENT\_PARAMETERS | char TECHNICAL\_MEASUREMENT\_PARAMETERS(N\_TECH\_MEAS\_PARAM, STRING64)  TECHNICAL\_MEASUREMENT\_PARAMETERS:long\_name = "List of available technical parameters for the station"  TECHNICAL\_MEASUREMENT\_PARAMETERS:conventions = "Reference table AUX\_3b"  TECHNICAL\_MEASUREMENT\_PARAMETERS:\_FillValue = " " | List of available technical parameters for the station. |
| DATA\_CENTRE | char DATA\_CENTRE(STRING2)  DATA\_CENTRE:long\_name = "Data centre in charge of float data processing"  DATA\_CENTRE:conventions = "Argo reference table 4"  DATA\_CENTRE:\_FillValue = " " | Code of the data centre in charge of the float data management.  The data centre codes are described in the Argo reference table 4.  Example : IF for Ifremer |
| DATE\_CREATION | char DATE\_CREATION(DATE\_TIME)  DATE\_CREATION:long\_name = "Date of file creation"  DATE\_CREATION:conventions = "YYYYMMDDHHMISS"  DATE\_CREATION:\_FillValue = " " | Date and time (UTC) of creation of this file.  Format : YYYYMMDDHHMISS  Example :  20011229161700 : December 29th 2001 16 :17 :00 |
| DATE\_UPDATE | char DATE\_UPDATE(DATE\_TIME)  DATE\_UPDATE:long\_name = "Date of update of this file"  DATE\_UPDATE:conventions = "YYYYMMDDHHMISS"  DATE\_UPDATE:\_FillValue = " " | Date and time (UTC) of update of this file.  Format : YYYYMMDDHHMISS  Example :  20011230090500 : December 30th 2001 09 :05 :00 |

### Technical data

Technical data of each profile can be reported by the float as:

* A set of unique technical measurements (at most one measurement of each technical parameter for each profile),
* A series of parameter measurements (for each profile).

#### Set of unique technical measurement

This section contains a set of technical data for each profile.

For each cycle, for each technical parameter, the name of the parameter and the value of the parameter are recorded.

The parameter name and its value are recorded as strings of 256 characters.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| TECHNICAL\_PARAMETER\_NAME | char TECHNICAL\_PARAMETER\_NAME(N\_TECH\_PARAM, STRING256)  TECHNICAL\_PARAMETER\_NAME:long\_name=”Name of technical parameter”  TECHNICAL\_PARAMETER\_NAME:\_FillValue = " " | Name of the technical parameter.  The list of technical parameter names is provided through the TECH\_AUX\_PARAM\_LABEL variable. |
| TECHNICAL\_PARAMETER\_VALUE | char TECHNICAL\_PARAMETER\_VALUE(N\_TECH\_PARAM, STRING256)  TECHNICAL\_PARAMETER\_VALUE:long\_name=”Value of technical parameter”  TECHNICAL\_PARAMETER\_VALUE:\_FillValue = " " | Value of the technical parameter.  Example :  "125049" |
| CYCLE\_NUMBER | int CYCLE\_NUMBER(N\_TECH\_PARAM)  CYCLE\_NUMBER:long\_name = "Float cycle number"  CYCLE\_NUMBER:conventions = "0...N, 0 : launch cycle (if exists), 1 : first complete cycle"  CYCLE\_NUMBER:\_FillValue = 99999 | Cycle number of the technical parameter.  Example : 157 |

Technical parameter names are identified by the “TECH” prefix, their descriptions are provided by the following variables.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| TECH\_AUX\_PARAM\_LABEL | char TECH\_AUX\_PARAM\_LABEL(N\_TECH\_AUX\_LABEL, STRING256)  TECH\_AUX\_PARAM\_LABEL:long\_name = "Name of auxiliary technical parameter label used by this float type and version"  TECH\_AUX\_PARAM\_LABEL:\_FillValue = " " | Name of auxiliary technical parameter label used by this float type and version. |
| TECH\_AUX\_PARAM\_DESCRIPTION | char TECH\_AUX\_PARAM\_DESCRIPTION(N\_TECH\_AUX\_LABEL, STRING1024)  TECH\_AUX\_PARAM\_DESCRIPTION:long\_name = "Description of auxiliary technical parameter label used by this float type and version"  TECH\_AUX\_PARAM\_DESCRIPTION:\_FillValue = " " | Description of auxiliary technical parameter label used by this float type and version. |

#### Series of technical parameter measurements

This section contains a series of technical parameter measurements for each profile.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| JULD | double JULD(N\_TECH\_MEASUREMENT)  JULD:long\_name = "Julian day (UTC) of each measurement relative to REFERENCE\_DATE\_TIME"  JULD:standard\_name = "time"  JULD:units = "days since 1950-01-01 00:00:00 UTC"  JULD:conventions = "Relative julian days with decimal part (as parts of day)"  JULD:resolution = <X>  JULD:\_FillValue = 999999.  JULD:axis = "T" | Julian day of the measurement.  The integer part represents the day, the decimal part represents the time of the measurement.  Date and time are in universal time coordinates.  The julian day is relative to REFERENCE\_DATE\_TIME.  Example :  18833.8013889885 : July 25 2001 19:14:00  <X>: these field depends on float type and version. |
| JULD\_STATUS | char JULD\_STATUS(N\_TECH\_MEASUREMENT)  JULD\_STATUS:long\_name=”Status of the date and time”  JULD\_STATUS:conventions = “Argo reference table 19"  JULD\_STATUS:\_FillValue = " " | Status flag on JULD date and time.  The flag scale is described in the Argo reference table 19.  Example:  2 : Value is transmitted by the float |
| JULD\_QC | char JULD\_QC(N\_TECH\_MEASUREMENT)  JULD\_QC:long\_name = "Quality on date and time"  JULD\_QC:conventions = "Argo reference table 2"  JULD\_QC:\_FillValue = " " | Quality flag on JULD date and time.  The flag scale is described in the Argo reference table 2.  Example :  1: the date and time seems correct. |
| JULD\_ADJUSTED | double JULD\_ADJUSTED(N\_TECH\_MEASUREMENT)  JULD\_ADJUSTED:long\_name = "Adjusted julian day (UTC) of each measurement relative to REFERENCE\_DATE\_TIME"  JULD\_ADJUSTED:standard\_name = "time"  JULD\_ADJUSTED:units = "days since 1950-01-01 00:00:00 UTC"  JULD\_ADJUSTED:conventions = "Relative julian days with decimal part (as parts of day)"  JULD\_ADJUSTED:resolution = <X>  JULD\_ADJUSTED:\_FillValue = 999999.  JULD:axis = "T" | Adjusted julian day of the measurement.  The integer part represents the day, the decimal part represents the time of the measurement.  Date and time are in universal time coordinates.  The julian day is relative to REFERENCE\_DATE\_TIME.  The date may be adjusted due to float clock drift or expert review.  Example :  18833.8013889885 : July 25 2001 19:14:00  <X>: these field depends on float type and version. |
| JULD\_ADJUSTED\_STATUS | char JULD\_ADJUSTED\_STATUS(N\_TECH\_MEASUREMENT  JULD\_ADJUSTED\_STATUS:long\_name=”Status of the JULD\_ADJUSTED date”  JULD\_ADJUSTED\_STATUS:conventions = “Argo reference table 19"  JULD\_ADJUSTED\_STATUS:\_FillValue = " " | Status flag on JULD date and time.  The flag scale is described in the Argo reference table 19.  Example:  2 : Value is transmitted by the float |
| JULD\_ADJUSTED\_QC | char JULD\_ADJUSTED\_QC(N\_TECH\_MEASUREMENT  JULD\_ADJUSTED\_QC:long\_name = "Quality on adjusted date and time"  JULD\_ADJUSTED\_QC:conventions = "Argo reference table 2"  JULD\_ADJUSTED\_QC:\_FillValue = " " | Quality flag on JULD\_ADJUSTED date and time.  The flag scale is described in the Argo reference table 2.  Example :  1 : the date and time seems correct. |
| CYCLE\_NUMBER\_MEAS | int CYCLE\_NUMBER\_MEAS(N\_TECH\_MEASUREMENT)  CYCLE\_NUMBER\_MEAS:long\_name = "Float cycle number of the measurement"  CYCLE\_NUMBER\_MEAS:conventions = "0...N, 0 : launch cycle, 1 : first complete cycle"  CYCLE\_NUMBER\_MEAS:\_FillValue = 99999 | Cycle number of the float for this series of measurements.  For one cycle number, there are usually several measurements received. This cycle number must match the profile cycle number.  Example : 17 for measurements performed during the 17th cycle of the float. |
| MEASUREMENT\_CODE | int MEASUREMENT\_CODE(N\_TECH\_MEASUREMENT)  MEASUREMENT\_CODE:long\_name = "Flag referring to a measurement event in the cycle"  MEASUREMENT\_CODE:conventions = "Argo reference table 15"  MEASUREMENT\_CODE:\_FillValue = 99999 | Flag for each event in the cycle which corresponds to the Argo reference table 15.  Example: 100 : All measurements made at start of descent to drift pressure. |
| <TECH\_PARAM> | float <TECH\_PARAM>(N\_TECH\_MEASUREMENT)  <TECH\_PARAM>:long\_name = "<X>"  <TECH\_PARAM>:standard\_name = "<X>"  <TECH\_PARAM>:\_FillValue = <X>  <TECH\_PARAM>:units = "<X>"  <TECH\_PARAM>:valid\_min = <X>  <TECH\_PARAM>:valid\_max = <X>  <TECH\_PARAM>:C\_format = "<X>"  <TECH\_PARAM>:FORTRAN\_format = "<X>" | <TECH\_PARAM> contains the original values of a technical parameter listed in the "code" column of the reference table AUX\_3b.  <X> : these fields are specified in the columns of the reference table AUX\_3b. |
| <TECH\_PARAM>\_QC | char <TECH\_PARAM>\_QC(N\_TECH\_MEASUREMENT)  <TECH\_PARAM>\_QC:long\_name = "quality flag"  <TECH\_PARAM>\_QC:conventions = "Argo reference table 2"  <TECH\_PARAM>\_QC:\_FillValue = " " | Quality flag applied on each <TECH\_PARAM> values.  The flag scale is specified in the Argo reference table 2. |

## Profile file format

### Global attributes

global attributes:

:title = "Argo float vertical profile auxiliary data"

:institution = "CORIOLIS"

:source = "Argo float"

:history = "2017-02-16T17:46:44Z creation; 2017-02-16T17:46:44Z last update (coriolis float real time data processing)"

:references = " "

:user\_manual\_version = "1.0"

:Conventions = "CF-1.6 Coriolis-Argo-Aux-1.0"

:featureType = "trajectoryProfileCoriolisAux"

|  |  |
| --- | --- |
| Global attribute name | Definition |
| title | A succinct description of what is in the dataset. |
| institution | Specifies where the original data was produced. |
| source | The method of production of the original data. If it was model-generated, source should name the model and its version, as specifically as could be useful. If it is observational, source should characterize it (e.g., "surface observation" or "radiosonde"). |
| history | Provides an audit trail for modifications to the original data. Well-behaved generic NetCDF filters will automatically append their name and the parameters with which they were invoked to the global history attribute of an input NetCDF file. We recommend that each line begin with a timestamp indicating the date and time of day that the program was executed. |
| references | Published or web-based references that describe the data or methods used to produce it. |
| user\_manual\_version | The version number of the user manual. |
| Conventions | The conventions supported by this file, blank separated. |
| featureType | The NetCDF CF feature type. |

### Dimensions

|  |  |  |
| --- | --- | --- |
| Name | Value | Definition |
| DATE\_TIME | DATE\_TIME = 14 | This dimension is the length of an ASCII date and time value.  Date\_time convention is : YYYYMMDDHHMISS  YYYY : year  MM : month  DD : day  HH : hour of the day  MI : minutes  SS : seconds  Date and time values are always in universal time coordinates (UTC).  Examples : 20010105172834 : January 5th 2001 17:28:34 19971217000000 : December 17th 1997 00:00:00 |
| STRING256  STRING64  STRING32  STRING16  STRING8  STRING4  STRING2 | STRING256 = 256  STRING64 = 64  STRING32 = 32  STRING16 = 16  STRING8 = 8  STRING4 = 4  STRING2 = 2 | String dimensions. |
| N\_PROF | N\_PROF = <int value> | Number of profiles contained in the file.  This dimension depends on the data set.  A file contains at least one profile.  There is no defined limit on the maximum number of profiles in a file. |
| N\_PARAM | N\_PARAM = <int value> | Maximum number of parameters measured or calculated for a pressure sample.  This dimension depends on the data set. |
| N\_LEVELS | N\_LEVELS = <int value> | Maximum number of pressure levels contained in a profile.  This dimension depends on the data set. |
| N\_CALIB | N\_CALIB = <int value> | Maximum number of calibrations performed on a profile.  This dimension depends on the data set. |
| N\_HISTORY | N\_HISTORY = UNLIMITED | Number of history records. |

### General information on the profile file

This section contains information about the whole file.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| DATA\_TYPE | char DATA\_TYPE(STRING16)  DATA\_TYPE:long\_name = "Data type"  DATA\_TYPE:conventions = "Reference table AUX\_1"  DATA\_TYPE:\_FillValue = " " | This field contains the type of data contained in the file.  The list of acceptable data types is in the reference table AUX\_1. |
| FORMAT\_VERSION | char FORMAT\_VERSION(STRING4)  FORMAT\_VERSION:long\_name = "File format version"  FORMAT\_VERSION:\_FillValue = " " | File format version.  Example : «1.0» |
| REFERENCE\_DATE\_TIME | char REFERENCE\_DATE\_TIME(DATE\_TIME)  REFERENCE\_DATE\_TIME:long\_name = "Date of reference for Julian days"  REFERENCE\_DATE\_TIME:conventions = "YYYYMMDDHHMISS"  REFERENCE\_DATE\_TIME:\_FillValue = " " | Date of reference for julian days.  The recommended reference date time is “19500101000000” : January 1st 1950 00:00:00 |
| DATE\_CREATION | char DATE\_CREATION(DATE\_TIME)  DATE\_CREATION:long\_name = "Date of file creation"  DATE\_CREATION:conventions = "YYYYMMDDHHMISS"  DATE\_CREATION:\_FillValue = " " | Date and time (UTC) of creation of this file.  Format : YYYYMMDDHHMISS  Example :  20011229161700 : December 29th 2001 16 :17 :00 |
| DATE\_UPDATE | char DATE\_UPDATE(DATE\_TIME)  DATE\_UPDATE:long\_name = "Date of update of this file"  DATE\_UPDATE:conventions = "YYYYMMDDHHMISS"  DATE\_UPDATE:\_FillValue = " " | Date and time (UTC) of update of this file.  Format : YYYYMMDDHHMISS  Example :  20011230090500 : December 30th 2001 09 :05 :00 |

### General information for each profile

This section contains general information on each profile.

Each item of this section has a N\_PROF (number of profiles) dimension.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| PLATFORM\_NUMBER | char PLATFORM\_NUMBER(N\_PROF, STRING8)  PLATFORM\_NUMBER:long\_name = "Float unique identifier"  PLATFORM\_NUMBER:conventions = "WMO float identifier : A9IIIII"  PLATFORM\_NUMBER:\_FillValue = " " | WMO float identifier.  WMO is the World Meteorological Organization.  This platform number is unique.  Example : 6900045 |
| PROJECT\_NAME | char PROJECT\_NAME(N\_PROF, STRING64)  PROJECT\_NAME:long\_name = "Name of the project"  PROJECT\_NAME:\_FillValue = " " | Name of the project which operates the profiling float that performed the profile.  Example : “GYROSCOPE” (EU project for ARGO program) |
| PI\_NAME | char PI\_NAME (N\_PROF, STRING64)  PI\_NAME:long\_name = "Name of the principal investigator"  PI\_NAME:\_FillValue = " " | Name of the principal investigator in charge of the profiling float.  Example : Yves Desaubies |
| STATION\_PARAMETERS | char STATION\_PARAMETERS(N\_PROF, N\_PARAM, STRING64)  STATION\_PARAMETERS:long\_name = "List of available parameters for the station"  STATION\_PARAMETERS:conventions = "Reference table AUX\_3a"  STATION\_PARAMETERS:\_FillValue = " " | List of parameters contained in this profile.  The parameter names are listed in reference table AUX\_3a. |
| CYCLE\_NUMBER | int CYCLE\_NUMBER(N\_PROF)  CYCLE\_NUMBER:long\_name = "Float cycle number"  CYCLE\_NUMBER:conventions = "0...N, 0 : launch cycle (if exists), 1 : first complete cycle"  CYCLE\_NUMBER:\_FillValue = 99999 | Float cycle number. |
| DIRECTION | char DIRECTION(N\_PROF)  DIRECTION:long\_name = "Direction of the station profiles"  DIRECTION:conventions = "A: ascending profiles, D: descending profiles"  DIRECTION:\_FillValue = " " | Type of profile on which measurement occurs.  A : ascending profile  D : descending profile |
| DATA\_CENTRE | char DATA\_CENTRE(N\_PROF, STRING2)  DATA\_CENTRE:long\_name = "Data centre in charge of float data processing"  DATA\_CENTRE:conventions = "Argo reference table 4"  DATA\_CENTRE:\_FillValue = " " | Code for the data centre in charge of the float data management.  The data centre codes are described in the Argo reference table 4.  Example : “ME" for MEDS |
| DC\_REFERENCE | char DC\_REFERENCE(N\_PROF, STRING32)  DC\_REFERENCE:long\_name = "Station unique identifier in data centre"  DC\_REFERENCE:conventions = "Data centre convention"  DC\_REFERENCE:\_FillValue = " " | Unique identifier of the profile in the data centre.  Data centres may have different identifier schemes.  DC\_REFERENCE is therefore not unique across data centres. |
| DATA\_STATE\_INDICATOR | char DATA\_STATE\_INDICATOR(N\_PROF, STRING4)  DATA\_STATE\_INDICATOR:long\_name = "Degree of processing the data have passed through"  DATA\_STATE\_INDICATOR:conventions = "Argo reference table 6"  DATA\_STATE\_INDICATOR:\_FillValue = " " | Degree of processing the data has passed through.  The data state indicator is described in the Argo reference table 6. |
| DATA\_MODE | char DATA\_MODE(N\_PROF)  DATA\_MODE:long\_name = "Delayed mode or real time data"  DATA\_MODE:conventions = "R : real time; D : delayed mode; A : real time with adjustment"  DATA\_MODE:\_FillValue = " " | Indicates if the profile contains real time, delayed mode or adjusted data.  R : real time data  D : delayed mode data  A : real time data with adjusted values |
| PARAMETER\_DATA\_MODE | char PARAMETER\_DATA\_MODE(N\_ PROF, N\_PARAM)  PARAMETER\_DATA\_MODE:long\_name = "Delayed mode or real time data"  PARAMETER\_DATA\_MODE:conventions = "R : real time; D : delayed mode; A : real time with adjustment"  PARAMETER\_DATA\_MODE:\_FillValue = " " | Describe the data mode of the individual parameter :  R : real time data  D : delayed mode data  A : real time data with adjusted values |
| PLATFORM\_TYPE | char PLATFORM\_TYPE(N\_PROF, STRING32)  PLATFORM\_TYPE:long\_name = "Type of float"  PLATFORM\_TYPE:conventions = "Argo reference table 23"  PLATFORM\_TYPE:\_FillValue = " " | Type of float listed in the Argo reference table 23.  Example: SOLO, APEX, PROVOR, ARVOR, NINJA |
| FLOAT\_SERIAL\_NO | char FLOAT\_SERIAL\_NO(N\_PROF, STRING32)  FLOAT\_SERIAL\_NO:long\_name = "Serial number of the float"  FLOAT\_SERIAL\_NO:\_FillValue = " " | Serial number of the float. |
| FIRMWARE\_VERSION | char FIRMWARE\_VERSION(N\_PROF, STRING32)  FIRMWARE\_VERSION:long\_name = "Instrument firmware version"  FIRMWARE\_VERSION:\_FillValue = " " | Firmware version of the float. |
| WMO\_INST\_TYPE | char WMO\_INST\_TYPE(N\_PROF, STRING4)  WMO\_INST\_TYPE:long\_name = "Coded instrument type”  WMO\_INST\_TYPE:conventions = "Argo reference table 8"  WMO\_INST\_TYPE:\_FillValue = " " | Instrument type from WMO code table 1770.  A subset of WMO table 1770 is documented in the Argo reference table 8.  Example :  846 : Webb Research float, Seabird sensor |
| JULD | double JULD(N\_PROF)  JULD:long\_name = "Julian day (UTC) of the station relative to REFERENCE\_DATE\_TIME"  JULD:standard\_name = "time"  JULD:units = "days since 1950-01-01 00:00:00 UTC"  JULD:conventions = "Relative julian days with decimal part (as parts of day)"  JULD:resolution = <X>  JULD:\_FillValue = 999999.  JULD:axis = "T" | Julian day of the profile.  The integer part represents the day, the decimal part represents the time of the profile.  Date and time are in Universal Time.  The julian day is relative to REFERENCE\_DATE\_TIME.  <X>: these field depends on float type and version.  Example :  18833.8013889885 : July 25 2001 19:14:00 |
| JULD\_QC | char JULD\_QC(N\_PROF)  JULD\_QC:long\_name = "Quality on date and time"  JULD\_QC:conventions = "Argo reference table 2"  JULD\_QC:\_FillValue = " " | Quality flag on JULD date and time.  The flag scale is described in the Argo reference table 2.  Example :  1: the date and time seems correct. |
| JULD\_LOCATION | double JULD\_LOCATION(N\_PROF)  JULD\_LOCATION:long\_name = "Julian day (UTC) of the location relative to REFERENCE\_DATE\_TIME"  JULD\_LOCATION:units = "days since 1950-01-01 00:00:00 UTC"  JULD\_LOCATION:conventions = "Relative julian days with decimal part (as parts of day)"  JULD\_LOCATION:resolution = <X>  JULD\_LOCATION:\_FillValue = 999999. | Julian day of the location of the profile.  The integer part represents the day, the decimal part represents the time of the profile.  Date and time are in Universal Time.  The julian day is relative to REFERENCE\_DATE\_TIME.  <X>: these field depends on float type and version.  Example :  18833.8013889885 : July 25 2001 19:14:00 |
| LATITUDE | double LATITUDE(N\_PROF)  LATITUDE:long\_name = "Latitude of the station, best estimate"  LATITUDE:standard\_name = "latitude"  LATITUDE:units = "degree\_north"  LATITUDE:\_FillValue = 99999.  LATITUDE:valid\_min = -90.  LATITUDE:valid\_max = 90.  LATITUDE:axis = "Y" | Latitude of the profile.  Unit : degree north  This field contains the best estimated latitude.  The latitude value may be improved in delayed mode.  The measured locations of the float are located in the trajectory file.  Example : 44.4991 : 44° 29’ 56.76’’ N |
| LONGITUDE | double LONGITUDE(N\_PROF)  LONGITUDE:long\_name = "Longitude of the station, best estimate"  LONGITUDE:standard\_name = "longitude"  LONGITUDE:units = "degree\_east"  LONGITUDE:\_FillValue = 99999.  LONGITUDE:valid\_min = -180.  LONGITUDE:valid\_max = 180.  LONGITUDE:axis = "X" | Longitude of the profile.  Unit : degree east  This field contains the best estimated longitude.  The longitude value may be improved in delayed mode.  The measured locations of the float are located in the trajectory file.  Example : 16.7222 : 16° 43’ 19.92’’ E |
| POSITION\_QC | char POSITION\_QC(N\_PROF)  POSITION\_QC:long\_name = "Quality on position (latitude and longitude)"  POSITION\_QC:conventions = "Argo reference table 2"  POSITION\_QC:\_FillValue = " " | Quality flag on position.  The flag on position is set according to (LATITUDE, LONGITUDE) quality.  The flag scale is described in the Argo reference table 2.  Example: 1: position seems correct. |
| POSITIONING\_SYSTEM | char POSITIONING\_SYSTEM(N\_PROF, STRING8)  POSITIONING\_SYSTEM:long\_name = "Positioning system"  POSITIONING\_SYSTEM:\_FillValue = " " | Name of the system in charge of positioning the float locations from the Argo reference table 9.  Examples : ARGOS |
| PROFILE\_<PARAM>\_QC | char PROFILE\_<PARAM>\_QC(N\_PROF)  PROFILE\_<PARAM>\_QC:long\_name = "Global quality flag of <PARAM> profile"  PROFILE\_<PARAM>\_QC:conventions = "Argo reference table 2a"  PROFILE\_<PARAM>\_QC:\_FillValue = " " | Global quality flag on the PARAM profile.  PARAM is among the STATION\_PARAMETERS.  The overall flag is set to indicate the percentage of good data in the profile as described in the Argo reference table 2a.  Example :  PROFILE\_TEMP\_QC = A : the temperature profile contains only good values  PROFILE\_PSAL\_QC = C : the salinity profile contains 50% to 75% good values |
| VERTICAL\_SAMPLING\_SCHEME | char VERTICAL\_SAMPLING\_SCHEME(N\_PROF, STRING256)  VERTICAL\_SAMPLING\_SCHEME:long\_name = "Vertical sampling scheme"  VERTICAL\_SAMPLING\_SCHEME:conventions = "Argo reference table 16"  VERTICAL\_SAMPLING\_SCHEME:\_FillValue = " " | Use vertical sampling scheme to differentiate and identify profiles from a single-cycle with different vertical sampling schemes.  See the Argo reference table 16. |
| CONFIG\_MISSION\_NUMBER | int CONFIG\_MISSION\_NUMBER(N\_PROF)  CONFIG\_MISSION\_NUMBER:long\_name = " Unique number denoting the missions performed by the float"  CONFIG\_MISSION\_NUMBER:conventions = "1...N, 1 : first complete mission"  CONFIG\_MISSION\_NUMBER:\_FillValue = 99999 | Unique number of the mission to which this profile belongs. |

### Measurements for each profile

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| <PARAM> | float <PARAM>(N\_PROF, N\_LEVELS)  <PARAM>:long\_name = "<X>"  <PARAM>:standard\_name = "<X>"  <PARAM>:\_FillValue = <X>  <PARAM>:units = "<X>"  <PARAM>:valid\_min = <X>  <PARAM>:valid\_max = <X>  <PARAM>:C\_format = "<X>"  <PARAM>:FORTRAN\_format = "<X>"  <PARAM>:resolution = <X> | <PARAM> contains the original values of a parameter listed in reference table AUX\_3a.  <X> : this field is specified in the reference table AUX\_3a. |
| <PARAM>\_QC | char <PARAM>\_QC(N\_PROF, N\_LEVELS)  <PARAM>\_QC:long\_name = "quality flag"  <PARAM>\_QC:conventions = "Argo reference table 2"  <PARAM>\_QC:\_FillValue = " " | Quality flag applied on each <PARAM> values.  The flag scale is specified in the Argo reference table 2. |
| <PARAM>\_ADJUSTED | float <PARAM>\_ADJUSTED(N\_PROF, N\_LEVELS)  <PARAM>\_ADJUSTED:long\_name = "<X>"  <PARAM>\_ADJUSTED:standard\_name = "<X>"  <PARAM>\_ADJUSTED:\_FillValue = <X>  <PARAM>\_ADJUSTED:units = "<X>"  <PARAM>\_ADJUSTED:valid\_min = <X>  <PARAM>\_ADJUSTED:valid\_max = <X>  <PARAM>\_ADJUSTED:C\_format = "<X>"  <PARAM>\_ADJUSTED:FORTRAN\_format = "<X>"  <PARAM>\_ADJUSTED:resolution= <X> | <PARAM>\_ADJUSTED contains the adjusted values derived from the original values of the parameter.  <X> : this field is specified in the reference table AUX\_3a.  When no adjustment is performed, the FillValue is inserted. |
| <PARAM>\_ADJUSTED\_QC | char <PARAM>\_ADJUSTED\_QC(N\_PROF, N\_LEVELS)  <PARAM>\_ADJUSTED\_QC:long\_name = "quality flag"  <PARAM>\_ADJUSTED\_QC:conventions = "Argo reference table 2"  <PARAM>\_ADJUSTED\_QC:\_FillValue = " " | Quality flag applied on each <PARAM>\_ADJUSTED values.  The flag scale is specified in the Argo reference table 2.  When no adjustment is performed, the FillValue is inserted. |
| <PARAM>\_ADJUSTED\_ERROR | float <PARAM>\_ADJUSTED\_ERROR(N\_PROF, N\_LEVELS)  <PARAM>\_ADJUSTED\_ERROR:long\_name = "Contains the error on the adjusted values as determined by the delayed mode QC process"  <PARAM>\_ADJUSTED\_ERROR:\_FillValue = <X>  <PARAM>\_ADJUSTED\_ERROR:units = "<X>"  <PARAM>\_ADJUSTED\_ERROR:C\_format = "<X>"  <PARAM>\_ADJUSTED\_ERROR:FORTRAN\_format = "<X>"  <PARAM>\_ADJUSTED\_ERROR:resolution= <X> | <PARAM>\_ADJUSTED\_ERROR  Contains the error on the adjusted values as determined by the delayed mode QC process.  <X> : this field is specified in the reference table AUX\_3a.  When no adjustment is performed, the FillValue is inserted. |

### Calibration information for each profile

This section contains calibration information for each parameter of each profile.

Each item of this section has a N\_PROF (number of profiles), N\_CALIB (number of calibrations), N\_PARAM (number of parameters) dimension.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| PARAMETER | char PARAMETER(N\_PROF, N\_CALIB, N\_PARAM, STRING64)  PARAMETER:long\_name = "List of parameters with calibration information"  PARAMETER:conventions = "Reference table AUX\_3a"  PARAMETER:\_FillValue = " " | Name of the calibrated parameter. The list of parameters is in the reference table AUX\_3a. |
| SCIENTIFIC\_CALIB\_EQUATION | char SCIENTIFIC\_CALIB\_EQUATION(N\_PROF, N\_CALIB, N\_PARAM, STRING256)  SCIENTIFIC\_CALIB\_EQUATION:long\_name = "Calibration equation for this parameter"  SCIENTIFIC\_CALIB\_EQUATION:\_FillValue = " " | Calibration equation applied to the parameter. |
| SCIENTIFIC\_CALIB\_COEFFICIENT | char SCIENTIFIC\_CALIB\_COEFFICIENT(N\_PROF, N\_CALIB, N\_PARAM, STRING256)  SCIENTIFIC\_CALIB\_COEFFICIENT:long\_name = "Calibration coefficients for this equation"  SCIENTIFIC\_CALIB\_COEFFICIENT:\_FillValue = " " | Calibration coefficients for this equation. |
| SCIENTIFIC\_CALIB\_COMMENT | char SCIENTIFIC\_CALIB\_COMMENT(N\_PROF, N\_CALIB, N\_PARAM, STRING256)  SCIENTIFIC\_CALIB\_COMMENT:long\_name = "Comment applying to this parameter calibration"  SCIENTIFIC\_CALIB\_COMMENT:\_FillValue = " " | Comment about this calibration  Example :  The sensor is not stable |
| SCIENTIFIC\_CALIB\_DATE | char SCIENTIFIC\_CALIB\_DATE (N\_PROF N\_CALIB, N\_PARAM, DATE\_TIME)  SCIENTIFIC\_CALIB\_DATE:long\_name = "Date of calibration"  SCIENTIFIC\_CALIB\_DATE:conventions = "YYYYMMDDHHMISS"  SCIENTIFIC\_CALIB\_DATE:\_FillValue = " " | Date of the calibration.  Example : 20011217161700 |

### History information for each profile

This section contains history information for each action performed on each profile by a data centre.

Each item of this section has a N\_HISTORY (number of history records), N\_PROF (number of profiles) dimension.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| HISTORY\_INSTITUTION | char HISTORY\_INSTITUTION(N\_HISTORY, N\_PROF, STRING4)  HISTORY\_INSTITUTION:long\_name = "Institution which performed action”  HISTORY\_INSTITUTION:conventions = "Argo reference table 4"  HISTORY\_INSTITUTION:\_FillValue = " " | Institution that performed the action.  Institution codes are described in the Argo reference table 4.  Example : ME for MEDS |
| HISTORY\_STEP | char HISTORY\_STEP(N\_HISTORY, N\_PROF, STRING4)  HISTORY\_STEP:long\_name = "Step in data processing"  HISTORY\_STEP:conventions = "Argo reference table 12"  HISTORY\_STEP:\_FillValue = " " | Code of the step in data processing for this history record. The step codes are described in the Argo reference table 12.  Example :  ARGQ : Automatic QC of data reported in real-time has been performed |
| HISTORY\_SOFTWARE | char HISTORY\_SOFTWARE (N\_HISTORY, N\_PROF, STRING4)  HISTORY\_SOFTWARE:long\_name = "Name of software which performed action"  HISTORY\_SOFTWARE:conventions = "Institution dependent"  HISTORY\_SOFTWARE:\_FillValue = " " | Name of the software that performed the action.  This code is institution dependent.  Example : WJO |
| HISTORY\_SOFTWARE\_RELEASE | char HISTORY\_SOFTWARE\_RELEASE(N\_HISTORY, N\_PROF, STRING4)  HISTORY\_SOFTWARE\_RELEASE:long\_name = "Version/release of software which performed action"  HISTORY\_SOFTWARE\_RELEASE:conventions = "Institution dependent"  HISTORY\_SOFTWARE\_RELEASE:\_FillValue = " " | Version of the software.  This name is institution dependent.  Example : «1.0» |
| HISTORY\_REFERENCE | char HISTORY\_REFERENCE (N\_HISTORY, N\_PROF, STRING64)  HISTORY\_REFERENCE:long\_name = "Reference of database"  HISTORY\_REFERENCE:conventions = "Institution dependent"  HISTORY\_REFERENCE:\_FillValue = " " | Code of the reference database used for quality control in conjunction with the software.  This code is institution dependent.  Example : WOD2001 |
| HISTORY\_DATE | char HISTORY\_DATE(N\_HISTORY, N\_PROF, DATE\_TIME)  HISTORY\_DATE:long\_name = "Date the history record was created"  HISTORY\_DATE:conventions = "YYYYMMDDHHMISS"  HISTORY\_DATE:\_FillValue = " " | Date of the action.  Example : 20011217160057 |
| HISTORY\_ACTION | char HISTORY\_ACTION(N\_HISTORY, N\_PROF, STRING4)  HISTORY\_ACTION:long\_name = "Action performed on data"  HISTORY\_ACTION:conventions = "Argo reference table 7"  HISTORY\_ACTION:\_FillValue = " " | Name of the action.  The action codes are described in the Argo reference table 7.  Example : QCF$ for QC failed |
| HISTORY\_PARAMETER | char HISTORY\_PARAMETER(N\_HISTORY, N\_PROF, STRING16)  HISTORY\_PARAMETER:long\_name = "Station parameter action is performed on"  HISTORY\_PARAMETER:conventions = "Reference table AUX\_3a"  HISTORY\_PARAMETER:\_FillValue = " " | Name of the parameter on which the action is performed. |
| HISTORY\_START\_PRES | float HISTORY\_START\_PRES(N\_HISTORY, N\_PROF)  HISTORY\_START\_PRES:long\_name = "Start pressure action applied on"  HISTORY\_START\_PRES:\_FillValue = 99999.f  HISTORY\_START\_PRES:units = "decibar" | Start pressure the action is applied to. |
| HISTORY\_STOP\_PRES | float HISTORY\_STOP\_PRES(N\_HISTORY, N\_PROF)  HISTORY\_STOP\_PRES:long\_name = "Stop pressure action applied on"  HISTORY\_STOP\_PRES:\_FillValue = 99999.f  HISTORY\_STOP\_PRES:units = "decibar" | Stop pressure the action is applied to. |
| HISTORY\_PREVIOUS\_VALUE | float HISTORY\_PREVIOUS\_VALUE(N\_HISTORY, N\_PROF)  HISTORY\_PREVIOUS\_VALUE:long\_name = "Parameter/Flag previous value before action"  HISTORY\_PREVIOUS\_VALUE:\_FillValue = 99999.f | Parameter or flag of the previous value before action.  Example : 2 (probably good) for a flag that was changed to 1 (good) |
| HISTORY\_QCTEST | char HISTORY\_QCTEST(N\_HISTORY, N\_PROF, STRING16)  HISTORY\_QCTEST:long\_name = "Documentation of tests performed, tests failed (in hex form)"  HISTORY\_QCTEST:conventions = "Write tests performed when ACTION=QCP$; tests failed when ACTION=QCF$"  HISTORY\_QCTEST:\_FillValue = " " | This field records the tests performed when ACTION is set to QCP$ (QC performed), the test failed when ACTION is set to QCF$ (QC failed).  The QCTEST codes are describe in the Argo reference table 11.  Example : 0A (in hexadecimal form) |

## Trajectory file format

### Global attributes

global attributes:

:title = "Argo float trajectory auxiliary data file"

:institution = "CORIOLIS"

:source = "Argo float"

:history = "2017-02-16T17:47:35Z creation; 2017-02-16T17:47:35Z last update (coriolis float real time data processing)"

:references = " "

:user\_manual\_version = "1.0"

:Conventions = “CF-1.6 Coriolis-Argo-Aux-1.0"

:featureType = "trajectoryCoriolisAux"

:comment\_on\_resolution = "<depends on float type and version>"

:comment\_on\_measurement\_code= "<depends on float type and version >"

|  |  |
| --- | --- |
| Global attribute name | Definition |
| title | A succinct description of what is in the dataset. |
| institution | Specifies where the original data was produced. |
| source | The method of production of the original data. If it was model-generated, source should name the model and its version, as specifically as could be useful. If it is observational, source should characterize it (e.g., "surface observation" or "radiosonde"). |
| history | Provides an audit trail for modifications to the original data. Well-behaved generic NetCDF filters will automatically append their name and the parameters with which they were invoked to the global history attribute of an input NetCDF file. We recommend that each line begin with a timestamp indicating the date and time of day that the program was executed. |
| references | Published or web-based references that describe the data or methods used to produce it. |
| user\_manual\_version | The version number of the user manual. |
| Conventions | The conventions supported by this file, blank separated. |
| featureType | The NetCDF CF feature type. |
| comment\_on\_resolution | Comment on parameter resolution. |
| comment\_on\_resolution | Comment on measurement codes. |

### Dimensions and definitions

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| DATE\_TIME | DATE\_TIME = 14 | This dimension is the length of an ASCII date and time value.  Date and time values are always in universal time coordinates (UTC).  Date\_time convention is : YYYYMMDDHHMISS  YYYY : year  MM : month  DD : day  HH : hour of the day  MI : minutes  SS : seconds  Examples :  20010105172834 : January 5th 2001 17:28:34  19971217000000 : December 17th 1997 00:00:00 |
| STRING64  STRING32  STRING16  STRING8  STRING4  STRING2 | STRING64 = 64  STRING32 = 32  STRING16 = 16  STRING8 = 8  STRING4 = 4  STRING2 = 2 | String dimensions. |
| N\_PARAM | N\_PARAM = <int value> | Maximum number of parameters measured or calculated for a pressure sample. |
| N\_MEASUREMENT | N\_MEASUREMENT = UNLIMITED | This dimension is the number of recorded locations, cycle timings and measurements of the file. |
| N\_CYCLE | N\_CYCLE = <int value> | Number of collected float cycles.  If all the cycles have been collected (i.e. if there are no missing cycles), it is the number of cycles performed by the float. In this particular case, as some floats begin cycle numbering at 0, others at 1, in the former, N\_CYCLE = max(CYCLE\_NUMBER) +1. In the latter, N\_CYCLE = max(CYCLE\_NUMBER)  Example : N\_CYCLE = 100 |
| N\_HISTORY | N\_HISTORY = <int value> | Maximum number of history records for a location. This dimension depends on the data set. |

### General information on the trajectory file

This section contains information about the whole file.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| DATA\_TYPE | char DATA\_TYPE(STRING16)  DATA\_TYPE:long\_name = "Data type"  DATA\_TYPE:conventions = "Reference table AUX\_1"  DATA\_TYPE:\_FillValue = " " | This field contains the type of data contained in the file.  The list of acceptable data types is in the reference table AUX\_1.  Example : Argo trajectory |
| FORMAT\_VERSION | char FORMAT\_VERSION(STRING4)  FORMAT\_VERSION:long\_name = "File format version"  FORMAT\_VERSION:\_FillValue = " " | File format version.  Example : "1.0" |
| REFERENCE\_DATE\_TIME | char REFERENCE\_DATE\_TIME(DATE\_TIME)  REFERENCE\_DATE\_TIME:long\_name = "Date of reference for Julian days"  REFERENCE\_DATE\_TIME:conventions = "YYYYMMDDHHMISS"  REFERENCE\_DATE\_TIME:\_FillValue = " " | Date of reference for julian days.  The recommended reference date time is "19500101000000" : January 1st 1950 00:00:00 |
| DATE\_CREATION | char DATE\_CREATION(DATE\_TIME)  DATE\_CREATION:long\_name = "Date of file creation "  DATE\_CREATION:conventions = "YYYYMMDDHHMISS"  DATE\_CREATION:\_FillValue = " " | Date and time (UTC) of creation of this file.  Format : YYYYMMDDHHMISS  Example :  20011229161700 : December 29th 2001 16 :17 :00 |
| DATE\_UPDATE | char DATE\_UPDATE(DATE\_TIME)  DATE\_UPDATE:long\_name = "Date of update of this file"  DATE\_UPDATE:conventions = "YYYYMMDDHHMISS"  DATE\_UPDATE:\_FillValue = " " | Date and time (UTC) of update of this file.  Format : YYYYMMDDHHMISS  Example :  20011230090500 : December 30th 2001 09 :05 :00 |

### General information on the float

This section contains general information on the float.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| PLATFORM\_NUMBER | char PLATFORM\_NUMBER(STRING8)  PLATFORM\_NUMBER:long\_name = "Float unique identifier"  PLATFORM\_NUMBER:conventions = "WMO float identifier : A9IIIII"  PLATFORM\_NUMBER:\_FillValue = " " | WMO float identifier.  WMO is the World Meteorological Organization.  This platform number is unique.  Example : "6900045" |
| PROJECT\_NAME | char PROJECT\_NAME(STRING64)  PROJECT\_NAME:long\_name = "Name of the project"  PROJECT\_NAME:\_FillValue = " " | Name of the project which operates the float that performed the trajectory.  Example : "GYROSCOPE" (EU project for ARGO program) |
| PI\_NAME | char PI\_NAME (STRING64)  PI\_NAME:long\_name = "Name of the principal investigator"  PI\_NAME:\_FillValue = " " | Name of the principal investigator in charge of the float.  Example : Yves Desaubies |
| TRAJECTORY\_PARAMETERS | char TRAJECTORY\_PARAMETERS(N\_PARAM,STRING64)  TRAJECTORY\_PARAMETERS:long\_name = "List of available parameters for the station"  TRAJECTORY\_PARAMETERS:conventions = "Reference table AUX\_3a"  TRAJECTORY\_PARAMETERS:\_FillValue = " " | List of parameters contained in this trajectory file.  The parameter names are listed in the reference table AUX\_3a. |
| DATA\_CENTRE | char DATA\_CENTRE(STRING2)  DATA\_CENTRE:long\_name = "Data centre in charge of float data processing"  DATA\_CENTRE:conventions = "Argo reference table 4"  DATA\_CENTRE:\_FillValue = " " | Code for the data centre in charge of the float data management.  The data centre codes are described in the Argo reference table 4.  Example : "ME" for MEDS |
| DATA\_STATE\_INDICATOR | char DATA\_STATE\_INDICATOR(STRING4)  DATA\_STATE\_INDICATOR:long\_name = "Degree of processing the data have passed through"  DATA\_STATE\_INDICATOR:conventions = "Argo reference table 6"  DATA\_STATE\_INDICATOR:\_FillValue = " " | Degree of processing the data has passed through.  The data state indicator is described in the Argo reference table 6. |
| PLATFORM\_TYPE | char PLATFORM\_TYPE(STRING32)  PLATFORM\_TYPE:long\_name = "Type of float"  PLATFORM\_TYPE:conventions = "Argo reference table 23"  PLATFORM\_TYPE:\_FillValue = " " | Type of float listed in the Argo reference table 23.  Example: SOLO, APEX, PROVOR, ARVOR, NINJA. |
| FLOAT\_SERIAL\_NO | char FLOAT\_SERIAL\_NO(STRING32)  FLOAT\_SERIAL\_NO:long\_name = "Serial number of the float"  FLOAT\_SERIAL\_NO:\_FillValue = " " | This field should contain only the serial number of the float. |
| FIRMWARE\_VERSION | char FIRMWARE\_VERSION(STRING32)  FIRMWARE\_VERSION:long\_name = "Instrument firmware version"  FIRMWARE\_VERSION:\_FillValue = " " | Firmware version of the float. |
| WMO\_INST\_TYPE | char WMO\_INST\_TYPE(STRING4)  WMO\_INST\_TYPE:long\_name = "Coded instrument type”  WMO\_INST\_TYPE:conventions = "Argo reference table 8"  WMO\_INST\_TYPE:\_FillValue = " " | Instrument type from WMO code table 1770.  A subset of WMO table 1770 is documented in the Argo reference table 8.  Example : 831 |

### N\_MEASUREMENT dimension variable group

This section describes the variables found in the N\_MEASUREMENT dimension variable group.

|  |  |  |
| --- | --- | --- |
| Name | definition | comment |
| JULD | double JULD(N\_MEASUREMENT)  JULD:long\_name = "Julian day (UTC) of each measurement relative to REFERENCE\_DATE\_TIME"  JULD:standard\_name = "time"  JULD:units = "days since 1950-01-01 00:00:00 UTC"  JULD:conventions = "Relative julian days with decimal part (as parts of day)"  JULD:resolution = <X>  JULD:\_FillValue = 999999.  JULD:axis = "T" | Julian day of the location (or measurement).  The integer part represents the day, the decimal part represents the time of the measurement.  Date and time are in universal time coordinates.  The julian day is relative to REFERENCE\_DATE\_TIME.  <X>: these field depends on float type and version.  Example :  18833.8013889885 : July 25 2001 19:14:00 |
| JULD\_STATUS | char JULD\_STATUS(N\_MEASUREMENT)  JULD\_STATUS:long\_name=”Status of the date and time”  JULD\_STATUS:conventions = “Argo reference table 19"  JULD\_STATUS:\_FillValue = " " | Status flag on JULD date and time.  The flag scale is described in the Argo reference table 19.  Example:  2 : Value is transmitted by the float |
| JULD\_QC | char JULD\_QC(N\_MEASUREMENT)  JULD\_QC:long\_name = "Quality on date and time"  JULD\_QC:conventions = "Argo reference table 2"  JULD\_QC:\_FillValue = " " | Quality flag on JULD date and time.  The flag scale is described in the Argo reference table 2.  Example :  1: the date and time seems correct. |
| JULD\_ADJUSTED | double JULD\_ADJUSTED(N\_MEASUREMENT)  JULD\_ADJUSTED:long\_name = "Adjusted julian day (UTC) of each measurement relative to REFERENCE\_DATE\_TIME"  JULD\_ADJUSTED:standard\_name = "time"  JULD\_ADJUSTED:units = "days since 1950-01-01 00:00:00 UTC"  JULD\_ADJUSTED:conventions = "Relative julian days with decimal part (as parts ofday)"  JULD\_ADJUSTED:resolution = <X>  JULD\_ADJUSTED:\_FillValue = 999999.  JULD:axis = "T" | Adjusted julian day of the location (or measurement).  The integer part represents the day, the decimal part represents the time of the measurement.  Date and time are in universal time coordinates.  The julian day is relative to REFERENCE\_DATE\_TIME.  The date may be adjusted due to float clock drift or expert review.  <X>: these field depends on float type and version.  Example :  18833.8013889885 : July 25 2001 19:14:00 |
| JULD\_ADJUSTED\_STATUS | char JULD\_ADJUSTED\_STATUS(N\_MEASUREMENT)  JULD\_ADJUSTED\_STATUS:long\_name=”Status of the JULD\_ADJUSTED date”  JULD\_ADJUSTED\_STATUS:conventions = “Argo reference table 19"  JULD\_ADJUSTED\_STATUS:\_FillValue = " " | Status flag on JULD date and time.  The flag scale is described in the Argo reference table 19.  Example:  2 : Value is transmitted by the float |
| JULD\_ADJUSTED\_QC | char JULD\_ADJUSTED\_QC(N\_MEASUREMENT)  JULD\_ADJUSTED\_QC:long\_name = "Quality on adjusted date and time"  JULD\_ADJUSTED\_QC:conventions = "Argo reference table 2"  JULD\_ADJUSTED\_QC:\_FillValue = " " | Quality flag on JULD\_ADJUSTED date and time.  The flag scale is described in the Argo reference table 2.  Example :  1 : the date and time seems correct. |
| CYCLE\_NUMBER | int CYCLE\_NUMBER(N\_MEASUREMENT)  CYCLE\_NUMBER:long\_name = "Float cycle number of the measurement"  CYCLE\_NUMBER:conventions = "0...N, 0 : launch cycle, 1 : first complete cycle"  CYCLE\_NUMBER:\_FillValue = 99999 | Cycle number of the float for this series of measurements, locations and timings. Some floats begin with a cycle 0 and some begin at cycle number 1. Cycle number is -1 for the float's launch and includes the time and location.  For one cycle number, there are usually several locations/measurement received. This cycle number must match the profile cycle number.  Example : 17 for measurements performed during the 17th cycle of the float. |
| CYCLE\_NUMBER\_ADJUSTED | int CYCLE\_NUMBER\_ADJUSTED (N\_MEASUREMENT)  CYCLE\_NUMBER\_ADJUSTED:long\_name = "Adjusted float cycle number of the measurement"  CYCLE\_NUMBER\_ADJUSTED:conventions = "0...N, 0 : launch cycle, 1 : first complete cycle"  CYCLE\_NUMBER\_ADJUSTED:\_FillValue = 99999 | Adjusted cycle number of the float for this series of measurements, locations and timings. Some floats begin with a cycle 0 and some begin at cycle number 1.  For one cycle number, there are usually several locations/measurement received. Sometimes cycle numbers are assigned erroneously and need to be corrected. This variable contains the corrected cycle numbers.  Example : 17 for measurements performed during the 17th cycle of the float. |
| MEASUREMENT\_CODE | int MEASUREMENT\_CODE (N\_MEASUREMENT)  MEASUREMENT\_CODE:long\_name = "Flag referring to a measurement event in the cycle"  MEASUREMENT\_CODE:conventions = "Argo reference table 15"  MEASUREMENT\_CODE:\_FillValue = 99999 | Flag for each event in the cycle which corresponds to Argo reference table 15.  Example: 100 : All measurements made at start of descent to drift pressure . Could be time, location, surface pressure, etc. |
| <PARAM> | float <PARAM>(N\_MEASUREMENT)  <PARAM>:long\_name = "<X>"  <PARAM>:standard\_name = "<X>"  <PARAM>:\_FillValue = <X>  <PARAM>:units = "<X>"  <PARAM>:valid\_min = <X>  <PARAM>:valid\_max = <X>  <PARAM>:C\_format = "<X>"  <PARAM>:FORTRAN\_format = "<X>" | <PARAM> contains the original values of a parameter listed in the "code" column of the reference table AUX\_3a.  <X> : these fields are specified in the columns of the reference table AUX\_3a. |
| <PARAM>\_QC | char <PARAM>\_QC(N\_MEASUREMENT)  <PARAM>\_QC:long\_name = "quality flag"  <PARAM>\_QC:conventions = "Argo reference table 2"  <PARAM>\_QC:\_FillValue = " " | Quality flag applied on each <PARAM> values.  The flag scale is specified in the Argo reference table 2. |
| <PARAM>\_ADJUSTED | float <PARAM>\_ADJUSTED(N\_MEASUREMENT)  <PARAM>\_ADJUSTED:long\_name = "<X>"  <PARAM>\_ADJUSTED:standard\_name = "<X>"  <PARAM>\_ADJUSTED:\_FillValue = <X>  <PARAM>\_ADJUSTED:units = "<X>"  <PARAM>\_ADJUSTED:valid\_min = <X>  <PARAM>\_ADJUSTED:valid\_max = <X>  <PARAM>\_ADJUSTED:comment = "<X>"  <PARAM>\_ADJUSTED:C\_format = "<X>"  <PARAM>\_ADJUSTED:FORTRAN\_format = "<X>"  <PARAM>\_ADJUSTED:resolution= <X> | <PARAM>\_ADJUSTED contains the adjusted values derived from the original values of the parameter.  <X> : these fields are specified in the columns of the reference table AUX\_3a.  When no adjustment is performed, the FillValue is inserted. |
| <PARAM>\_ADJUSTED\_QC | char <PARAM>\_ADJUSTED\_QC(N\_MEASUREMENT)  <PARAM>\_ADJUSTED\_QC:long\_name = "quality flag"  <PARAM>\_ADJUSTED\_QC:conventions = "Argo reference table 2"  <PARAM>\_ADJUSTED\_QC:\_FillValue = " " | Quality flag applied on each <PARAM>\_ADJUSTED values.  The flag scale is specified in the Argo reference table 2.  When no adjustment is performed, the FillValue is inserted. |
| <PARAM>\_ADJUSTED\_ERROR | float <PARAM>\_ADJUSTED\_ERROR(N\_MEASUREMENT)  <PARAM>\_ADJUSTED\_ERROR:long\_name = "Contains the error on the adjusted values as determined by the delayed mode QC process."  <PARAM>\_ADJUSTED\_ERROR:\_FillValue = <X>  <PARAM>\_ADJUSTED\_ERROR:units = "<X>"  <PARAM>\_ADJUSTED\_ERROR:C\_format = "<X>"  <PARAM>\_ADJUSTED\_ERROR:FORTRAN\_format = "<X>"  <PARAM>\_ADJUSTED\_ERROR:resolution= <X> | <PARAM>\_ADJUSTED\_ERROR contains the error on the adjusted values of the parameter.  <X> : these fields are specified in the columns of the reference table AUX\_3a.  When no adjustment is performed, the FillValue is inserted. |

### N\_CYCLE dimension variable group

This section contains information on the variables with dimension N\_CYCLE.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| CONFIG\_MISSION\_NUMBER | int CONFIG\_MISSION\_NUMBER(N\_CYCLE)  CONFIG\_MISSION\_NUMBER:long\_name = "Unique number denoting the missions performed by the float”  CONFIG\_MISSION\_NUMBER:conventions = "1...N, 1 : first complete mission"  CONFIG\_MISSION\_NUMBER:\_FillValue = 99999 | Unique number of the mission to which this cycle belongs. |
| CYCLE\_NUMBER\_INDEX | int CYCLE\_NUMBER\_INDEX(N\_CYCLE)  CYCLE\_NUMBER\_INDEX:long\_name = "Cycle number that corresponds to the current index"  CYCLE\_NUMBER\_INDEX:conventions = "0...N, 0 : launch cycle, 1 : first complete cycle"  CYCLE\_NUMBER\_INDEX:FillValue = 99999 | Cycle number of the float that corresponds to the information contained in the current index. This cycle number must match the profile cycle number, ensuring that the trajectory and profile with the same cycle number contain data from the same cycle.  Example: 17 means information for the 17th cycle of the float is contained in this index. |
| CYCLE\_NUMBER\_INDEX\_ADJUSTED | int CYCLE\_NUMBER\_INDEX\_ADJUSTED(N\_CYCLE)  CYCLE\_NUMBER\_INDEX\_ADJUSTED:long\_name = "Adjusted cycle number that corresponds to the current index"  CYCLE\_NUMBER\_INDEX\_ADJUSTED:conventions = "0...N, 0 : launch cycle, 1 : first complete cycle"  CYCLE\_NUMBER\_INDEX\_ADJUSTED:FillValue = 99999 | Corrected cycle number of the float that corresponds to the information contained in the current index. Errors may be found in CYCLE\_NUMBER\_INDEX variable which are corrected and contained in this variable. Example: 17 means information for the 17th cycle of the float is contained in this index. |
| DATA\_MODE | char DATA\_MODE(N\_CYCLE)  DATA\_MODE:long\_name = "Delayed mode or real time data"  DATA\_MODE:conventions = "R : real time; D : delayed mode; A : real time with adjustment"  DATA\_MODE:\_FillValue = " " | Indicates if the trajectory cycle contains real time, adjusted or delayed mode data. A delayed mode cycle means the positions, times, cycle number, pressure, temperature, and salinity (if measured) have been quality controlled. Additional parameters like oxygen may not be quality controlled. Floats often have delayed mode data only after they die, but can have both delayed mode and real time data. When this occurs, two trajectory files exist - a real time file ("R") with only real time data for all the cycles in the float record and a delayed mode file ("D") with both real time and delayed mode data for all the cycles that have been delayed mode quality controlled. Floats can be adjusted in real time with adjusted time values only in the JULD\_ADJUSTED variable and its associated \_STATUS and \_QC variables. This occurs when floats are corrected in real time for clock drift.  Examles :  'R' : real time data  'D' : delayed mode data  'A' : real time data with adjusted values |

### History information

This section contains history information for each action performed on each measurement.

|  |  |  |
| --- | --- | --- |
| Name | Definition | Comment |
| HISTORY\_INSTITUTION | char HISTORY\_INSTITUTION(N\_HISTORY, STRING4)  HISTORY\_INSTITUTION:long\_name = "Institution which performed action”  HISTORY\_INSTITUTION:conventions = "Argo reference table 4"  HISTORY\_INSTITUTION:\_FillValue = " " | Institution that performed the action.  Institution codes are described in the Argo reference table 4.  Example : "ME" for MEDS |
| HISTORY\_STEP | char HISTORY\_STEP(N\_HISTORY, STRING4)  HISTORY\_STEP:long\_name = "Step in data processing"  HISTORY\_STEP:conventions = "Argo reference table 12"  HISTORY\_STEP:\_FillValue = " " | Code of the step in data processing for this history record. The step codes are described in the Argo reference table 12.  Example :  "ARGQ" : Automatic QC of data reported in real-time has been performed |
| HISTORY\_SOFTWARE | char HISTORY\_SOFTWARE(N\_HISTORY, STRING4)  HISTORY\_SOFTWARE:long\_name = "Name of software which performed action"  HISTORY\_SOFTWARE:conventions = "Institution dependent"  HISTORY\_SOFTWARE:\_FillValue = " " | Name of the software that performed the action.  This code is institution dependent.  Example : "OW" |
| HISTORY\_SOFTWARE\_RELEASE | char HISTORY\_SOFTWARE\_RELEASE(N\_HISTORY, STRING4)  HISTORY\_SOFTWARE\_RELEASE:long\_name = "Version/release of software which performed action"  HISTORY\_SOFTWARE\_RELEASE:conventions = "Institution dependent"  HISTORY\_SOFTWARE\_RELEASE:\_FillValue = " " | Version of the software.  This name is institution dependent.  Example : "1.0" |
| HISTORY\_REFERENCE | char HISTORY\_REFERENCE(N\_HISTORY, STRING64)  HISTORY\_REFERENCE:long\_name = "Reference of database"  HISTORY\_REFERENCE:conventions = "Institution dependent"  HISTORY\_REFERENCE:\_FillValue = " " | Code of the reference database used for quality control in conjunction with the software.  This code is institution dependent.  Example : "WOD2001" |
| HISTORY\_DATE | char HISTORY\_DATE(N\_HISTORY, DATE\_TIME)  HISTORY\_DATE:long\_name = "Date the history record was created"  HISTORY\_DATE:conventions = "YYYYMMDDHHMISS"  HISTORY\_DATE:\_FillValue = " " | Date of the action.  Example : "20011217160057" |
| HISTORY\_ACTION | char HISTORY\_ACTION(N\_HISTORY, STRING4)  HISTORY\_ACTION:long\_name = "Action performed on data"  HISTORY\_ACTION:conventions = "Argo reference table 7"  HISTORY\_ACTION:\_FillValue = " " | Name of the action.  The action codes are described in the Argo reference table 7.  Example : "QCF$" for QC failed |
| HISTORY\_PARAMETER | char HISTORY\_PARAMETER(N\_HISTORY, STRING64)  HISTORY\_PARAMETER:long\_name = "Station parameter action is performed on"  HISTORY\_PARAMETER:conventions = "Reference table AUX\_3a"  HISTORY\_PARAMETER:\_FillValue = " " | Name of the parameter on which the action is performed. |
| HISTORY\_PREVIOUS\_VALUE | float HISTORY\_PREVIOUS\_VALUE(N\_HISTORY)  HISTORY\_PREVIOUS\_VALUE:long\_name = "Parameter/Flag previous value before action"  HISTORY\_PREVIOUS\_VALUE:\_FillValue = 99999. | Parameter or flag of the previous value before action.  Example :' 2' (probably good) for a flag that was changed to '1' (good) |
| HISTORY\_INDEX\_DIMENSION | char HISTORY\_INDEX\_DIMENSION(N\_HISTORY)  HISTORY\_INDEX\_DIMENSION:long\_name = "Name of dimension to which HISTORY\_START\_INDEX and HISTORY\_STOP\_INDEX correspond"  HISTORY\_INDEX\_DIMENSION:conventions = "C: N\_CYCLE, M: N\_MEASUREMENT"  HISTORY\_INDEX\_DIMENSION:\_FillValue = " " | Name of dimension to which HISTORY\_START\_INDEX and HISORY\_STOP\_INDEX correspond.  'C': N\_CYCLE  'M': N\_MEASUREMENT |
| HISTORY\_START\_INDEX | int HISTORY\_ START\_INDEX(N\_HISTORY)  HISTORY\_START\_INDEX:long\_name = "Start index action applied on"  HISTORY\_START\_INDEX:\_FillValue =  99999 | Start index the action is applied to. This index corresponds to N\_MEASUREMENT or N\_CYCLE, depending on the corrected parameter |
| HISTORY\_STOP\_INDEX | int HISTORY\_ STOP\_INDEX(N\_HISTORY)  HISTORY\_STOP\_INDEX:long\_name = "Stop index action applied on"  HISTORY\_STOP\_INDEX:\_FillValue =  99999 | Stop index the action is applied to. This index corresponds to N\_MEASUREMENT or N\_CYCLE, depending on the corrected parameter. |
| HISTORY\_QCTEST | char HISTORY\_QCTEST(N\_HISTORY, STRING16)  HISTORY\_QCTEST:long\_name = "Documentation of tests performed, tests failed (in hex form)"  HISTORY\_QCTEST:conventions = "Write tests performed when ACTION=QCP$; tests failed when ACTION=QCF$"  HISTORY\_QCTEST:\_FillValue = " " | This field records the tests performed when ACTION is set to QCP$ (QC performed), the test failed when ACTION is set to QCF$ (QC failed).  The QCTEST codes are describe in the Argo reference table 11.  Example : "0A" (in hexadecimal form) |

## File localization and naming

### Coriolis auxiliary files localization

**TBD**

### Coriolis auxiliary files naming convention

The produced auxiliary files comply with the following naming conventions.

#### Metadata file

<FloatID>\_meta\_aux.nc

where:

* <FloatID> is the float WMO number

Example: 4901802\_meta\_aux.nc

#### Technical data file

<FloatID>\_tech\_aux.nc

where:

* <FloatID> is the float WMO number

Example: 4901802\_tech\_aux.nc

#### Profile data files

<R/D><FloatID>\_<XXX><D>\_aux.nc

where:

* <FloatID> is the float WMO number,
* <R/D> indicates Real-Time data (R) or Delayed-Mode data (D),
* <XXX> is the cycle number,
* <D> is added for descending profile (default is ascending profile).

Example: R4901802\_012\_aux.nc

#### Trajectory data file

<FloatID>\_<R/D>traj\_aux.nc

where:

* <FloatID> is the float WMO number,
* <R/D> indicates Real-Time data (R) or Delayed-Mode data (D).

Example: 4901802\_Rtraj\_aux.nc

# Reference tables

## Reference table AUX\_1: data type

This table contains the list of acceptable values for DATA\_TYPE field of auxiliary files.

|  |
| --- |
| Name |
| Aux-Argo profile |
| Aux-Argo trajectory |
| Argo auxiliary meta-data |
| Argo auxiliary technical data |

## Reference table AUX\_3: parameter code table

### Reference table AUX\_3a: sensor parameter code table

The following table describes the defined sensor parameter codes for auxiliary files. Note that the codes of the Argo reference table 3 could also be present in the auxiliary files.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Parameter name | long\_name | cf standard\_name | Unit | valid\_min | valid\_max |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### Reference table AUX\_3b: technical parameter code table

The following table describes the defined technical parameter codes for auxiliary files.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Parameter name | long\_name | standard\_name | Unit | valid\_min | valid\_max |
| PUMP\_ACTION\_FLAG | Pump action flag | - | boolean | 0 | 1 |
| PUMP\_ACTION\_DURATION | Pump action duration | - | dsec | - | - |
| VALVE\_ACTION\_FLAG | Valve action flag | - | boolean | 0 | 1 |
| VALVE\_ACTION\_DURATION | Valve action duration | - | dsec | - | - |
|  |  |  |  |  |  |

## Reference Table AUX\_25: SENSOR

This table contains the list of the defined SENSOR for auxiliary files. Note that the SENSOR of the Argo reference table 25 could also be present in the auxiliary files.

|  |
| --- |
| SENSOR |
|  |
|  |
|  |
|  |
|  |

## Reference Table AUX\_26: SENSOR\_MAKER

This table contains the list of the defined SENSOR\_MAKER for auxiliary files. Note that the SENSOR\_MAKER of the Argo reference table 26 could also be present in the auxiliary files.

|  |  |
| --- | --- |
| SENSOR\_MAKER | Description |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Reference Table AUX\_27: SENSOR\_MODEL

This table contains the list of the defined SENSOR\_MODEL for auxiliary files. Note that the SENSOR\_MODEL of the Argo reference table 27 could also be present in the auxiliary files.

|  |  |
| --- | --- |
| SENSOR\_MODEL | Comment |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |