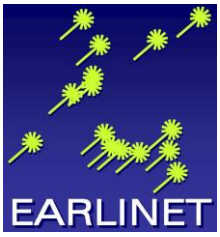

EARLINET Data File Structure

Francesco Amato, Giuseppe D'Amico, Lucia Mona,
Claudio Dema, Sergio Ciamprone
Consiglio Nazionale delle Ricerche
CNR-IMAA

Version 2.0 - 11/11/2019

Products

EARLINET aerosol profile data are collected at each station applying measurement procedures compliant with the recommendations provided by the ACTRIS aerosol profile community.
Currently, the EARLINET database contains quality assured aerosol extinction, backscatter and particle/volume depolarization ratio profiles. The main concept behind the current database is to provide aerosol lidar products at the best available resolution in both time and space.
The EARLINET database represents a comprehensive, quantitative, and statistically significant collection of data for the aerosol distribution on European scale. The data quality of the database is assured by intercomparisons at both instrument and algorithm levels.
It is an important source of data to contribute to the quantification of anthropogenic and biogenic emissions and concentrations of aerosol, quantification of their budgets, radiative properties and prediction of future trends. It can improve the understanding of physical and chemical processes related to the aerosols, their long-range transport and deposition, and their interaction with clouds.

All the EARLINET products are in **NetCDF** format(https://www.unidata.ucar.edu/software/netcdf/).
The NetCDF (Network Common Data Form) format, developed by the University Corporation for Atmospheric Research (UCAR), is particularly convenient for describing arrays and matrices of scientific data. Moreover, developers have made freely available online open source libraries and tools necessary to generate, read, view and manipulate files in NetCDF format.
In particular, libraries in C, C ++, Java and Fortran are available.
The NetCDF format is self-descriptive through a special header that describes the structure and contents of all other information in the file. Another strength of the NetCDF format is that it is independent of both the computer architecture and the operating system (platform independent) that is used. Is it also possible to use native compression features especially comfortable when dealing with large amounts of data.
EARLINET products (after the release of new/reshaped EARLINET database structure – June 2019) are compliant with **CF (Climate and Forecast) version 1.7** (http://cfconventions.org/Data/cf-conventions/cf-conventions-1.7/cf-conventions.html), that is the currently most recent release.
This convention defines metadata that provide a definitive description of what the data in each variable represents, and of the spatial and temporal properties of the data. The purpose of the CF conventions is to require conforming datasets to contain sufficient metadata that they are self-describing in the sense that each variable in the file has an associated description of what it represents, including physical units if appropriate, and that each value can be located in space and time. This enables users of data from different sources to decide which quantities are comparable, and facilitates building applications with powerful extraction, regridding, and display capabilities.
An important benefit of this convention is that it enables software tools to display data and perform operations on specified subsets of the data with minimal user intervention.

Products Naming convention

First of all, it has been set a new product filename convention. The filename of each EARLINET product is a string composed by 72 characters having the following format :

EARLINET_AerRemSen_ccc_Levzz_ttttt_YYYYMMDDHHMM_yyyymmddhhmm_vxx_qcyy.nc

The character **“_”** is used to separate different field in the filename. The extension of the file is always **“.nc”**. The number of fields composing the filename (excluding the extension) is always **9**. The following list describes, in the order, each field that composes the filename**.**

	Field	Length	Description
1st	*EARLINET*	8	This field always reports the string "EARLINET"
2nd	*AerRemSen*	9	This field means "Aerosol Remote Sensing"
3rd	*ccc*	3	This field reports a three digit code representing univocally an EARLINET station
4th	*Levzz*	5	This field specifies the level of the product. The levels are assigned to the product on the base of spe cific quality control procedure.
5th	*ttttt*	5	This field identifies the product type.
6th	*YYYYMMDDHHMM*	12	This is a date-time field and it provides the start date and time of the measurements contained in the product file. The time is UTC.
7th	*yyyymmddhhmm*	12	This is a date-time field and it provides the stop date and time of the measurements contained in the product file. The time is UTC.
8th	*vxx*	3	This field specifies the version of the product. The first characater'v' is always present. The next 2 characters are always numeric. (E.g. v01 identifies the version 1 of the file, v02 the version 2 and so on).
9th	*qcyy*	4	This field specifies the quality control version of the product. The suffix 'qc' is always present. The next 2 characters are always numeric. (E.g. qc01 identifies the QC version 0.1 of the file, qc0.2 the version 0.2 and so on).

The name to assign to each submitted product is generated by the datacenter automatically during the submission phase according to the above conventions. So the product submitted by the data originator will be always renamed. For traceability the original product name provided by the data originator is stored along the other product metadata.
The new products data file format has been enriched with both new variables and global attributes. Both variables and global attributes presence in a certain product are recorded into the database.

NetCDF file structure

The new products data file format has been enriched with both new variables and global attributes. The presence of each variables and/or global attributes contained in each product is are recorded into the database as product metadata.
The following tables show the name and the corresponding physical meaning of all the dimensions, variables and global attributes that an EARLINET optical product data file can contain. Products not compliant with these definitions will be not accepted by the datacenter.
It is important to note that the 3 variables **"quality_control_level"**, **"basic_quality_control"**, and **"advanced_quality_control"** are computed and written by the datacenter automatically and should not be provided by the data originator. The contents of these variables are set according to EARLINET quality control policy described in the document
New_Quality_Checks_description.pdf

Optical Product Dimensions

Name	Dimensions
time	number of profiles included in the time series
wavelength	number of wavelengths in the product
altitude	number of altitudes in the product
nv	ancillary dimension

Optical Product Variables

Name	Dimensions	Type	Description
time	time	double	time
time_bounds	time, nv	double	start, stop time for each optical product
altitude	altitude	double	height above sea level
wavelength	wavelength	float	wavelength of the transmitted laser pulse
longitude	-	float	longitude of station
latitude	-	float	latitude of station
station_altitude	-	float	station altitude above see level
zenith_angle	-	float	laser pointing angle with respect to the zenith
shots	-	int	accumulated laser shots
cirrus_contamination	-	byte	do the profiles contain cirrus layers?
cirrus_contamination_source	-	byte	How was cirrus_contamination obtained?
quality_control_level	-	int	Quality Control Level **ADDED BY DATACENTER**
basic_quality_control	-	int	Basic Quality Control **ADDED BY DATACENTER**
advanced_quality_control	-	int	Advanced Quality Control **ADDED BY DATACENTER**
backscatter	wavelength, time, altitude	double	aerosol backscatter coefficient
error_backscatter	wavelength, time, altitude	double	statistical uncertainty of aerosol backscatter
extinction	wavelength, time, altitude	double	aerosol extinction coefficient

error_extinction	wavelength, time, altitude	double	statistical uncertainty of aerosol extinction
vertical_resolution	wavelength, time, altitude	double	effective vertical resolution according to Pappalardo et al., appl. opt. 2004
lidarratio	wavelength, time, altitude	double	aerosol extinction-to-backscatter ratio
error_lidarratio	wavelength, time, altitude	double	statistical uncertainty of lidar ratio
aerosollayerheight	time	double	top of dust layer above sea level
mixinglayerheight	time	double	top of convective boundary layer above sea level
volumedepolarization	wavelength, time, altitude	double	volume linear depo larization ratio
error_volumedepolarization	wavelength, time, altitude	double	statistical uncertainty of volume linear depolarization ratio
particledepolarization	wavelength, time, altitude	double	aerosol linear depolarization ratio
error_particledepolarization	wavelength, time, altitude	double	statistical uncertainty of aerosol linear depolarization ratio
watervapormixingratio	wavelength, time, altitude	double	ratio of the mass of water vapor and the mass of dry air
error_watervapor	wavelength, time, altitude	double	statistical uncertainty of the water vapor mixing ratio
earlinet_product_type	-	int	Earlinet product type
user_defined_category	-	int	User defined category of the measurement
backscatter_calibration_value	wavelength	float	assumed backscatter-ratio value in calibration range
extinction_assumed_wavelength_dependence	wavelength	float	assumed wavelength dependence for extinction retrieval
error_retrieval_method	wavelength	byte	method used for the retrieval of uncertainties
backscatter_evaluation_method	wavelength	byte	method used for the backscatter retrieval

backscatter_calibration_range_search_algorithm	wavelength	byte	algorithm used for the search of the calibration_range
extinction_evaluation_algorithm	wavelength	byte	algorithm used for the extinction retrieval
raman_backscatter_algorithm	wavelength	byte	algorithm used for the retrieval of the Raman backscatter profile
atmospheric_molecular_calculation_source	-	byte	data source of the atmospheric molecular calculations
backscatter_calibration_range	wavelength, nv	float	altitude range where calibration was calculated
cloud_mask	wavelength, time, altitude	byte	cloud mask
backscatter_calibration_search_range	wavelength, nv	float	height range wherein calibration range is searched
elastic_backscatter_algorithm	wavelength	byte	0: Klett-Fernald, 1: iterative
__SkippedFraction	-	float	Fraction of skipped profiles within averaged period

Optical Product Global Attributes

Name	**Type**	**Description**
Conventions	string	Name of the conventions
title	string	A succinct description of what is in the dataset
source	string	The method of production of the original data
references	string	Published or web-based references that describe the data or methods used to produce it
history	string	Provides an audit trail for modifications to the original data
station_ID	string	Unique id associated to the station
location	string	lidar system location
system	string	Lidar system
institution	string	Specifies where the original data was produced
comment	string	Miscellaneous information

		about the data or methods used to produce it
measurement_ID	string	Unique id for the raw time series
measurement_start_datetime	string	measurement start datetime
measurement_stop_datetime	string	measurement stop datetime
PI	string	name of the Principal Investigator
PI_affiliation	string	Principal Investigator affiliation
PI_affiliation_acronym	string	Principal Investigator affiliation acronym
PI_address	string	Principal Investigator address
PI_phone	string	Principal Investigator phone
PI_email	string	Principal Investigator email
Data_Originator	string	name Data Originator
Data_Originator_affiliation	string	Data Originator affiliation
Data_Originator_affiliation_ acronym	string	Data Originator affiliation acronym
Data_Originator_address	string	Data Originator address
Data_Originator_phone	string	Data Originator phone
Data_Originator_email	string	Data Originator email
data_processing_institution	string	institution which has processed the product
hoi_system_ID	integer	HOI id assigned to the EARLINET system used to perform the raw measurements the product refer to.
hoi_configuration_ID	integer	HOI id assigned to the EARLINET system configuration used to perfom the raw measurements the product refer to.
scc_version	string	Single Calculus Chain version
scc_version_description	string	Single Calculus Chain version description
processor_name	string	Analysis Software
processor_version	string	Analysis software version
__file_format_version	string	file format version
input_file	string	raw data filename
overlap_correction_file	string	filename used to correct the overlap

NetCDF Variables declaration :

Here is listed the declaration of all the variables included in a NetCDF product data file sorted by type (double, float, int, byte) :

- **double variables :**

double time(time) ;
time:axis = "T" ;
time:bounds = "time_bounds" ;
time:calendar = "gregorian" ;
time:long_name = "time" ;
time:standard_name = "time" ;
time:units = "seconds since 1970-01-01T00:00:00Z" ;

double time_bounds(time, nv) ;

double altitude(altitude) ;
altitude:axis = "Z" ;
altitude:long_name = "height above sea level" ;
altitude:positive = "up" ;
altitude:standard_name = "altitude" ;
altitude:units = "m" ;

double backscatter(wavelength, time, altitude) ;
backscatter:ancillary_variables = "error_backscatter vertical_resolution" ;
backscatter:coordinates = "longitude latitude" ;
backscatter:long_name = "aerosol backscatter coefficient" ;
backscatter:plausibility = "parameter passed the EARLINET quality assurance." ;
backscatter:units = "m-1*sr-1" ;
backscatter:_FillValue = 9.96920996838687e+36 ;

double error_backscatter(wavelength, time, altitude) ;
error_backscatter:coordinates = "longitude latitude" ;
error_backscatter:long_name = "statistical uncertainty of aerosol backscatter" ;
error_backscatter:plausibility = "parameter passed the EARLINET quality assurance." ;
error_backscatter:units = "m-1*sr-1" ;
error_backscatter:_FillValue = 9.96920996838687e+36 ;

double extinction(wavelength, time, altitude) ;
extinction:ancillary_variables = "error_extinction vertical_resolution" ;
extinction:coordinates = "longitude latitude" ;
extinction:long_name = "aerosol extinction coefficient" ;
extinction:plausibility = "parameter passed the EARLINET quality assurance." ;
extinction:units = "m-1" ;
extinction:_FillValue = 9.96920996838687e+36 ;

double volumedepolarization(wavelength, time, altitude) ;
volumedepolarization:ancillary_variables = "error_volumedepolarization" ;
volumedepolarization:coordinates = "longitude latitude" ;
volumedepolarization:long_name = "volume linear depolarization ratio" ;
volumedepolarization:plausibility = "parameter not quality assured by EARLINET." ;
volumedepolarization:units = "1" ;
volumedepolarization:_FillValue = 9.96920996838687e+36 ;

double error_volumedepolarization(wavelength, time, altitude) ;
error_volumedepolarization:coordinates = "longitude latitude" ;
error_volumedepolarization:long_name = "statistical uncertainty of volume linear depolarization ratio" ;
error_volumedepolarization:plausibility = "parameter not quality assured by EARLINET." ;
error_volumedepolarization:units = "1" ;
error_volumedepolarization:_FillValue = 9.96920996838687e+36 ;

double particledepolarization(wavelength, time, altitude) ;
particledepolarization:ancillary_variables = "error_particledepolarization" ;
particledepolarization:coordinates = "longitude latitude" ;
particledepolarization:long_name = "aerosol linear depolarization ratio" ;
particledepolarization:plausibility = "parameter not quality assured by EARLINET." ;
particledepolarization:units = "1" ;
particledepolarization:_FillValue = 9.96920996838687e+36 ;

double error_particledepolarization(wavelength, time, altitude) ;
error_particledepolarization:coordinates = "longitude latitude" ;
error_particledepolarization:long_name = "statistical uncertainty of aerosol linear depolarization ratio" ;
error_particledepolarization:plausibility = "parameter not quality assured by EARLINET." ;
error_particledepolarization:units = "1" ;
error_particledepolarization:_FillValue = 9.96920996838687e+36 ;

double aerosollayerheight(time) ;
aerosollayerheight:coordinates = "longitude latitude" ;
aerosollayerheight:long_name = "top of dust layer above sea level" ;
aerosollayerheight:plausibility = "parameter not quality assured by EARLINET." ;
aerosollayerheight:units = "m" ;
aerosollayerheight:_FillValue = 9.96920996838687e+36 ;

double mixinglayerheight(time) ;
mixinglayerheight:coordinates = "longitude latitude" ;
mixinglayerheight:long_name = "top of convective boundary layer above sea level" ;
mixinglayerheight:plausibility = "parameter not quality assured by EARLINET." ;
mixinglayerheight:units = "m" ;
mixinglayerheight:_FillValue = 9.96920996838687e+36 ;

double particledepolarization(wavelength, time, altitude) ;
particledepolarization:ancillary_variables = "error_particledepolarization" ;
particledepolarization:coordinates = "longitude latitude" ;
particledepolarization:long_name = "aerosol linear depolarization ratio" ;
particledepolarization:plausibility = "parameter not quality assured by EARLINET." ;
particledepolarization:units = "1" ;
particledepolarization:_FillValue = 9.96920996838687e+36 ;

double error_watervapor(wavelength, time, altitude) ;
error_watervapor:standard_name = "humidity_mixing_ratio standard_error" ;
error_watervapor:coordinates = "longitude latitude" ;
error_watervapor:long_name = "statistical uncertainty of the water vapor mixing ratio" ;
error_watervapor:plausibility = "parameter not quality assured by EARLINET." ;
error_watervapor:units = "g/kg" ;
error_watervapor:_FillValue = 9.96920996838687e+36 ;

double lidarratio(wavelength, time, altitude) ;
lidarratio:ancillary_variables = "error_lidarratio" ;
lidarratio:coordinates = "longitude latitude" ;
lidarratio:long_name = "aerosol extinction-to-backscatter ratio" ;
lidarratio:plausibility = "parameter calculated from backscatter and extinction." ;
lidarratio:units = "sr" ;
lidarratio:_FillValue = 9.96920996838687e+36 ;

double error_lidarratio(wavelength, time, altitude) ;
error_lidarratio:coordinates = "longitude latitude" ;
error_lidarratio:long_name = "statistical uncertainty of lidar ratio" ;
error_lidarratio:plausibility = "parameter calculated from error_backscatter anderror_extinction." ;
error_lidarratio:units = "sr" ;
error_lidarratio:_FillValue = 9.96920996838687e+36 ;

double vertical_resolution(wavelength, time, altitude) ;
vertical_resolution:long_name = "effective vertical resolution according to Pappalardo et al., appl. opt. 2004" ;
vertical_resolution:units = "m" ;
vertical_resolution:_FillValue = 9.96920996838687e+36 ;

double assumed_particle_lidar_ratio(wavelength, time, altitude) ;
assumed_particle_lidar_ratio:ancillary_variables = "assumed_particle_lidar_ratio_error" ;
assumed_particle_lidar_ratio:long_name = "assumed particle lidar ratio value to use in elastic only backscatter retrieval" ;
assumed_particle_lidar_ratio:units = "sr" ;
assumed_particle_lidar_ratio:_FillValue = 9.96920996838687e+36 ;

double assumed_particle_lidar_ratio_error(wavelength, time, altitude) ;
assumed_particle_lidar_ratio_error:long_name = "error on assumed particle lidar ratio value to use in elastic only backscatter retrieval" ;
assumed_particle_lidar_ratio_error:units = "sr" ;
assumed_particle_lidar_ratio_error:_FillValue = 9.96920996838687e+36 ;

- **float variables :**

float wavelength(wavelength) ;
wavelength:long_name = "wavelength of the transmitted laser pulse" ;
wavelength:units = "nm" ;

float longitude ;
longitude:long_name = "longitude of station" ;
longitude:standard_name = "longitude" ;
longitude:units = "degrees_east" ;

float latitude ;
latitude:long_name = "latitude of station" ;
latitude:standard_name = "latitude" ;
latitude:units = "degrees_north" ;

float station_altitude ;
station_altitude:long_name = "station altitude above see level" ;
station_altitude:units = "m" ;
station_altitude:_FillValue = 9.96921e+36f ;

float zenith_angle ;
zenith_angle:long_name = "laser pointing angle with respect to the zenith" ;
zenith_angle:units = "degrees" ;
zenith_angle:_FillValue = 9.96921e+36f ;

float backscatter_calibration_range(wavelength, nv) ;
backscatter_calibration_range:long_name = "altitude range where calibration was calculated" ;
backscatter_calibration_range:units = "m" ;
backscatter_calibration_range:_FillValue = 9.96921e+36f ;

float backscatter_calibration_search_range(wavelength, nv) ;
backscatter_calibration_search_range:long_name = "altitude range wherein calibration range is searched" ;
backscatter_calibration_search_range:units = "m" ;
backscatter_calibration_search_range:_FillValue = 9.96921e+36f ;

float backscatter_calibration_value(wavelength) ;
backscatter_calibration_value:long_name = "assumed backscatter-ratio value in calibration range" ;
backscatter_calibration_value:units = "m-1*sr-1" ;
backscatter_calibration_value:_FillValue = 9.96921e+36f ;

float extinction_assumed_wavelength_dependence(wavelength) ;
extinction_assumed_wavelength_dependence:long_name = "assumed wavelength dependence for extinction retrieval" ;
extinction_assumed_wavelength_dependence:units = "1" ;
extinction_assumed_wavelength_dependence:_FillValue = 9.96921e+36f ;

- **int variables :**

int shots ;
shots:long_name = "accumulated laser shots" ;
shots:_FillValue = -2147483647 ;

int user_defined_category ;
user_defined_category:long_name = "User defined category of the measurement" ;
user_defined_category:_FillValue = -2147483647 ;
user_defined_category:valid_range = 0, 1023 ;
user_defined_category:flag_masks = 1, 2, 4, 8, 16, 32, 64, 128, 256, 512 ;
user_defined_category:flag_meanings = "cirrus climatology diurnal_cycles etna forest_fires photosmog rural_urban saharan_dust stratosphere satellite_overpasses" ;
user_defined_category:comment = "Those flags might have not been set in a homogeneous way. Before using them, contact the originator to obtain more detailed information on how these flags have been set." ;

int earlinet_product_type ;
earlinet_product_type:long_name = "Earlinet product type" ;
earlinet_product_type:_FillValue = -2147483647 ;
earlinet_product_type:valid_range = 1, 14 ;
earlinet_product_type:flag_values = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14 ;
earlinet_product_type:flag_meanings = "e0355 b0355 e0351 b0351 e0532 b0532 e1064 b1064 b0253 b0313 b0335 b0510 b0694 b0817" ;

- **int variables added by the system on the uploading phase:**

int quality_control_level ;
quality_control_level:long_name = "Quality Control Level" ;
quality_control_level:flag_values = 0, 1, 2 ;
quality_control_level:flag_meanings = "File_does_not_overcome_one_or_more_on_fly_quality_control File_does_overcome_all_on_fly_quality_control_but_fails_one_or_more_technical_quality_control File_does_overcome_all_technical_quality_control_and_physical_quality_control " ;
quality_control_level:version = "1.1" ;
quality_control_level:references = "https://www.earlinet.org/index.php?id=125" ;

int basic_quality_control ;
basic_quality_control:long_name = "Basic Quality Control" ;
basic_quality_control:valid_range = 0, 7 ;
basic_quality_control:flag_masks = 1, 2, 4 ;
basic_quality_control:flag_meanings = "Check_if_file_contains_data Check_for_Undefined_Variables_and_Global_Attributes Check_Coordinates_Consistency" ;
basic_quality_control:references = "https://www.earlinet.org/index.php?id=125" ;

int advanced_quality_control ;

advanced_quality_control:long_name = "Advanced Quality Control" ;

advanced_quality_control:valid_range = 0, 247 ;

advanced_quality_control:flag_masks = 1, 2, 4, 16, 32, 64, 128 ;

advanced_quality_control:flag_meanings = "Checks_for_Negative_Errors Negative_Peaks Check_on_AOD Check_on_LidarRatio Check_on_Volumedepolarization Check_on_Particledepolarization Check_on_Watervapormixingratio" ;

advanced_quality_control:references = "https://www.earlinet.org/index.php?id=125" ;

- **byte variables :**

byte cirrus_contamination ;
cirrus_contamination:long_name = "do the profiles contain cirrus layers?" ;
cirrus_contamination:_FillValue = -127b ;
cirrus_contamination:valid_range = 0b, 3b ;
cirrus_contamination:flag_values = 0b, 1b, 2b ;
cirrus_contamination:flag_meanings = "not_available no_cirrus cirrus_detected" ;

byte cirrus_contamination_source ;
cirrus_contamination_source:long_name = "how was cirrus_contamination obtained?" ;
cirrus_contamination_source:_FillValue = -127b ;
cirrus_contamination_source:valid_range = 0b, 3b ;
cirrus_contamination_source:flag_values = 0b, 1b, 2b ;
cirrus_contamination_source:flag_meanings = "not_available user_provided automatic_calculated" ;

byte error_retrieval_method(wavelength) ;
error_retrieval_method:long_name = "method used for the retrieval of uncertainties" ;
error_retrieval_method:_FillValue = -127b ;
error_retrieval_method:flag_values = 0b, 1b ;
error_retrieval_method:flag_meanings = "monte_carlo error_propagation" ;

byte backscatter_evaluation_method(wavelength) ;
backscatter_evaluation_method:long_name = "method used for the backscatter retrieval" ;
backscatter_evaluation_method:_FillValue = -127b ;
backscatter_evaluation_method:flag_values = 0b, 1b ;
backscatter_evaluation_method:flag_meanings = "Raman elastic_backscatter" ;

byte elastic_backscatter_algorithm(wavelength) ;
elastic_backscatter_algorithm:long_name = "0: Klett-Fernald, 1: iterative" ;
elastic_backscatter_algorithm:_FillValue = -127b ;
elastic_backscatter_algorithm:flag_values = 0b, 1b ;
elastic_backscatter_algorithm:flag_meanings = "Klett-Fernald iterative" ;

byte backscatter_calibration_range_search_algorithm(wavelength) ;
backscatter_calibration_range_search_algorithm:long_name = "algorithm used for the search of the calibration_range" ;
backscatter_calibration_range_search_algorithm:_FillValue = -127b ;
backscatter_calibration_range_search_algorithm:flag_values = 0b, 1b ;
backscatter_calibration_range_search_algorithm:flag_meanings = "minimum_of_signal_ratio minimum_of_elastic_signal" ;

byte extinction_evaluation_algorithm(wavelength) ;
extinction_evaluation_algorithm:long_name = "algorithm used for the extinction retrieval" ;
extinction_evaluation_algorithm:_FillValue = -127b ;
extinction_evaluation_algorithm:flag_values = 0b, 1b ;
extinction_evaluation_algorithm:flag_meanings = "weighted_linear_fit non-weighted_linear_fit" ;

byte raman_backscatter_algorithm(wavelength) ;

raman_backscatter_algorithm:long_name = "algorithm used for the retrieval of the Raman backscatter profile" ;

raman_backscatter_algorithm:_FillValue = -127b ;

raman_backscatter_algorithm:flag_values = 0b, 1b ;

raman_backscatter_algorithm:flag_meanings = "Ansmann via_backscatter_ratio" ;

byte atmospheric_molecular_calculation_source ;

atmospheric_molecular_calculation_source:long_name = "data source of the atmospheric molecular calculations" ;

atmospheric_molecular_calculation_source:_FillValue = -127b ;

atmospheric_molecular_calculation_source:flag_values = 0b, 1b, 2b, 3b, 4b ;

atmospheric_molecular_calculation_source:flag_meanings = "US_standard_atmosphere radiosounding ecmwf icon-iglo-12-13 gdas" ;

byte cloud_mask(time, altitude) ;

cloud_mask:long_name = "cloud mask" ;

cloud_mask:_FillValue = -127b ;

cloud_mask:valid_range = 0b, 7b ;

cloud_mask:flag_masks = 1b, 2b, 4b ;

cloud_mask:flag_meanings = "unknown_cloud cirrus_cloud water_cloud" ;

byte cloud_mask_type(time, altitude) ;

cloud_mask_type:long_name = "cloud mask type" ;

cloud_mask_type:_FillValue = -127b ;

cloud_mask_type:valid_range = 0b, 7b ;

cloud_mask_type:flag_masks = 1b, 2b, 4b ;

cloud_mask_type:flag_meanings = "no_cloudmask_available manual_cloudmask automatic_cloudmask" ;

CDL example file:

```
netcdf EARLINET_AerRemSen_pot_Lev01_e0355_201906131944_201906132129_v01_qc02 {
dimensions:
      time = 1 ;
      wavelength = 1 ;
      altitude = 245 ;
      nv = 2 ;
variables:
      double time(time) ;
            time:axis = "T" ;
            time:bounds = "time_bounds" ;
            time:calendar = "gregorian" ;
            time:long_name = "time" ;
            time:standard_name = "time" ;
            time:units = "seconds since 1970-01-01T00:00:00Z" ;
      double time_bounds(time, nv) ;

      double altitude(altitude) ;
            altitude:axis = "Z" ;
            altitude:long_name = "height above sea level" ;
            altitude:positive = "up" ;
            altitude:standard_name = "altitude" ;
            altitude:units = "m" ;

      float wavelength(wavelength) ;
            wavelength:long_name = "wavelength of the transmitted laser pulse" ;
            wavelength:units = "nm" ;

      float backscatter_calibration_value(wavelength) ;
            backscatter_calibration_value:long_name = "assumed backscatter-ratio
                                                value in calibration
range" ;
            backscatter_calibration_value:units = "m-1*sr-1" ;
            backscatter_calibration_value:_FillValue = 9.96921e+36f ;

      float extinction_assumed_wavelength_dependence(wavelength) ;
            extinction_assumed_wavelength_dependence:long_name = "assumed
                                          wavelength dependence for
                                          extinction retrieval" ;
            extinction_assumed_wavelength_dependence:units = "1" ;
            extinction_assumed_wavelength_dependence:_FillValue = 9.96921e+36f ;

      byte error_retrieval_method(wavelength) ;
            error_retrieval_method:long_name = "method used for the retrieval of
                                          uncertainties" ;
            error_retrieval_method:_FillValue = -127b ;
            error_retrieval_method:flag_values = 0b, 1b ;
            error_retrieval_method:flag_meanings = "monte_carlo
                                                error_propagation" ;

      byte backscatter_evaluation_method(wavelength) ;
            backscatter_evaluation_method:long_name = "method used for the
                                                backscatter retrieval" ;
            backscatter_evaluation_method:_FillValue = -127b ;
            backscatter_evaluation_method:flag_values = 0b, 1b ;
            backscatter_evaluation_method:flag_meanings = "Raman

elastic_backscatter" ;

      byte backscatter_calibration_range_search_algorithm(wavelength) ;
            backscatter_calibration_range_search_algorithm:long_name =
                  "algorithm used for the search of the calibration_range" ;
            backscatter_calibration_range_search_algorithm:_FillValue = -127b ;
            backscatter_calibration_range_search_algorithm:flag_values = 0b,1b ;
            backscatter_calibration_range_search_algorithm:flag_meanings =
                        "minimum_of_signal_ratio minimum_of_elastic_signal" ;
```

```
      byte extinction_evaluation_algorithm(wavelength) ;
            extinction_evaluation_algorithm:long_name = "algorithm used for the
                                                  extinction retrieval" ;
            extinction_evaluation_algorithm:_FillValue = -127b ;
            extinction_evaluation_algorithm:flag_values = 0b, 1b ;
            extinction_evaluation_algorithm:flag_meanings = "weighted_linear_fit
                                                  non-weighted_linear_fit" ;

      byte raman_backscatter_algorithm(wavelength) ;
            raman_backscatter_algorithm:long_name = "algorithm used for the
                        retrieval of the Raman backscatter profile" ;
            raman_backscatter_algorithm:_FillValue = -127b ;
            raman_backscatter_algorithm:flag_values = 0b, 1b ;
            raman_backscatter_algorithm:flag_meanings = "Ansmann
                                                  via_backscatter_ratio" ;

      float longitude ;
            longitude:long_name = "longitude of station" ;
            longitude:standard_name = "longitude" ;
            longitude:units = "degrees_east" ;

      float latitude ;
            latitude:long_name = "latitude of station" ;
            latitude:standard_name = "latitude" ;
            latitude:units = "degrees_north" ;

      float station_altitude ;
            station_altitude:long_name = "station altitude above see level" ;
            station_altitude:units = "m" ;
            station_altitude:_FillValue = 9.96921e+36f ;

      float zenith_angle ;
            zenith_angle:long_name = "laser pointing angle with respect to the
                                  zenith" ;
            zenith_angle:units = "degrees" ;
            zenith_angle:_FillValue = 9.96921e+36f ;

      int shots ;
            shots:long_name = "accumulated laser shots" ;
            shots:_FillValue = -2147483647 ;

      byte atmospheric_molecular_calculation_source ;
            atmospheric_molecular_calculation_source:long_name = "data source of
                              the atmospheric molecular calculations" ;
            atmospheric_molecular_calculation_source:_FillValue = -127b ;
            atmospheric_molecular_calculation_source:flag_values = 0b, 1b, 2b,
                                                                  3b, 4b ;
            atmospheric_molecular_calculation_source:flag_meanings =
            "US_standard_atmosphere radiosounding ecmwf icon-iglo-12-13 gdas" ;

      byte cirrus_contamination ;
            cirrus_contamination:long_name = "do the profiles contain cirrus
layers?" ;
            cirrus_contamination:_FillValue = -127b ;
            cirrus_contamination:valid_range = 0b, 3b ;
            cirrus_contamination:flag_values = 0b, 1b, 2b ;
            cirrus_contamination:flag_meanings = "not_available no_cirrus
                                          cirrus_detected" ;

      byte cirrus_contamination_source ;
            cirrus_contamination_source:long_name = "how was
                                          cirrus_contamination
obtained?" ;
            cirrus_contamination_source:_FillValue = -127b ;
            cirrus_contamination_source:valid_range = 0b, 3b ;
            cirrus_contamination_source:flag_values = 0b, 1b, 2b ;
            cirrus_contamination_source:flag_meanings = "not_available
                              user_provided automatic_calculated" ;
```

```
      int quality_control_level ;
            quality_control_level:long_name = "Quality Control Level" ;
            quality_control_level:flag_values = 0, 1, 2 ;
            quality_control_level:flag_meanings =
      "File_does_not_overcome_one_or_more_on_fly_quality_control
File_does_overcome_all_on_fly_quality_control_but_fails_one_or_more_technical_qu
ality_control
File_does_overcome_all_technical_quality_control_and_physical_quality_control
" ;
            quality_control_level:version = "1.1" ;
            quality_control_level:references =
"https://www.earlinet.org/index.php?id=125" ;

      int basic_quality_control ;
            basic_quality_control:long_name = "Basic Quality Control" ;
            basic_quality_control:valid_range = 0, 7 ;
            basic_quality_control:flag_masks = 1, 2, 4 ;
            basic_quality_control:flag_meanings = "Check_if_file_contains_data
Check_for_Undefined_Variables_and_Global_Attributes
Check_Coordinates_Consistency" ;
            basic_quality_control:references =
"https://www.earlinet.org/index.php?id=125" ;

      int advanced_quality_control ;
            advanced_quality_control:long_name = "Advanced Quality Control" ;
            advanced_quality_control:valid_range = 0, 247 ;
            advanced_quality_control:flag_masks = 1, 2, 4, 16, 32, 64, 128 ;
            advanced_quality_control:flag_meanings = "Checks_for_Negative_Errors
Negative_Peaks Check_on_AOD Check_on_LidarRatio Check_on_Volumedepolarization
Check_on_Particledepolarization Check_on_Watervapormixingratio" ;
            advanced_quality_control:references =
"https://www.earlinet.org/index.php?id=125" ;

      double backscatter(wavelength, time, altitude) ;
            backscatter:ancillary_variables = "error_backscatter
vertical_resolution" ;
            backscatter:coordinates = "longitude latitude" ;
            backscatter:long_name = "aerosol backscatter coefficient" ;
            backscatter:plausibility = "parameter passed the EARLINET quality
assurance." ;
            backscatter:units = "m-1*sr-1" ;
            backscatter:_FillValue = 9.96920996838687e+36 ;

      double error_backscatter(wavelength, time, altitude) ;
            error_backscatter:coordinates = "longitude latitude" ;
            error_backscatter:long_name = "statistical uncertainty of aerosol
backscatter" ;
            error_backscatter:plausibility = "parameter passed the EARLINET
quality assurance." ;
            error_backscatter:units = "m-1*sr-1" ;
            error_backscatter:_FillValue = 9.96920996838687e+36 ;

      double extinction(wavelength, time, altitude) ;
            extinction:ancillary_variables = "error_extinction
vertical_resolution" ;
            extinction:coordinates = "longitude latitude" ;
            extinction:long_name = "aerosol extinction coefficient" ;
            extinction:plausibility = "parameter passed the EARLINET quality
assurance." ;
            extinction:units = "m-1" ;
            extinction:_FillValue = 9.96920996838687e+36 ;

      double error_extinction(wavelength, time, altitude) ;
            error_extinction:coordinates = "longitude latitude" ;
            error_extinction:long_name = "statistical uncertainty of aerosol
extinction" ;
            error_extinction:plausibility = "parameter passed the EARLINET
quality assurance." ;
```

```
            error_extinction:units = "m-1" ;
            error_extinction:_FillValue = 9.96920996838687e+36 ;


      double vertical_resolution(wavelength, time, altitude) ;
            vertical_resolution:long_name = "effective vertical resolution
according to Pappalardo et al., appl. opt. 2004" ;
            vertical_resolution:units = "m" ;
            vertical_resolution:_FillValue = 9.96920996838687e+36 ;

      double lidarratio(wavelength, time, altitude) ;
            lidarratio:ancillary_variables = "error_lidarratio" ;
            lidarratio:coordinates = "longitude latitude" ;
            lidarratio:long_name = "aerosol extinction-to-backscatter ratio" ;
            lidarratio:plausibility = "parameter calculated from backscatter and
extinction." ;
            lidarratio:units = "sr" ;
            lidarratio:_FillValue = 9.96920996838687e+36 ;

      double error_lidarratio(wavelength, time, altitude) ;
            error_lidarratio:coordinates = "longitude latitude" ;
            error_lidarratio:long_name = "statistical uncertainty of lidar
ratio" ;
            error_lidarratio:plausibility = "parameter calculated from
error_backscatter and error_extinction." ;
            error_lidarratio:units = "sr" ;
            error_lidarratio:_FillValue = 9.96920996838687e+36 ;

      int earlinet_product_type ;
            earlinet_product_type:long_name = "Earlinet product type" ;
            earlinet_product_type:_FillValue = -2147483647 ;
            earlinet_product_type:valid_range = 1, 14 ;
            earlinet_product_type:flag_values = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
11, 12, 13, 14 ;
            earlinet_product_type:flag_meanings = "e0355 b0355 e0351 b0351 e0532
b0532 e1064 b1064 b0253 b0313 b0335 b0510 b0694 b0817" ;

      int user_defined_category ;
            user_defined_category:long_name = "User defined category of the
measurement" ;
            user_defined_category:_FillValue = -2147483647 ;
            user_defined_category:valid_range = 0, 1023 ;
            user_defined_category:flag_masks = 1, 2, 4, 8, 16, 32, 64, 128, 256,
512 ;
            user_defined_category:flag_meanings = "cirrus climatology
diurnal_cycles etna forest_fires photosmog rural_urban saharan_dust stratosphere
satellite_overpasses" ;
            user_defined_category:comment = "Those flags might have not been set
in a homogeneous way. Before using them, contact the originator to obtain more
detailed information on how these flags have been set." ;

      float backscatter_calibration_range(wavelength, nv) ;
            backscatter_calibration_range:long_name = "altitude range where
calibration was calculated" ;
            backscatter_calibration_range:units = "m" ;
            backscatter_calibration_range:_FillValue = 9.96921e+36f ;

      float backscatter_calibration_search_range(wavelength, nv) ;
            backscatter_calibration_search_range:long_name = "altitude range
wherein calibration range is searched" ;
            backscatter_calibration_search_range:units = "m" ;
            backscatter_calibration_search_range:_FillValue = 9.96921e+36f ;

      byte cloud_mask(time, altitude) ;
            cloud_mask:long_name = "cloud mask" ;
            cloud_mask:_FillValue = -127b ;
            cloud_mask:valid_range = 0b, 7b ;
            cloud_mask:flag_masks = 1b, 2b, 4b ;
            cloud_mask:flag_meanings = "unknown_cloud cirrus_cloud
water_cloud" ;
```

// global attributes:
:Conventions = "CF-1.7" ;
:title = "Profiles of aerosol optical properties" ;
:source = "Ground based LIDAR measurements" ;
:references = "Project website at http://www.earlinet.org" ;
:history = "2019-07-17T05:31Z : Assigned version 1\r\n 2019-07-17T05:31:00Z : File uploaded on Earlinet database\r\n 2019-06-14T09:50:28Z: elpp -d sccoperational -m 20190613po19 -c elpp.config; 2019-06-14T09:50:49Z: elda 20190613po19 -c elda.ini" ;
:station_ID = "pot" ;
:location = "Potenza, Italy" ;
:system = "MUSA" ;
:institution = "Consiglio Nazionale delle Ricerche - Istituto di Metodologie per l\'Analisi Ambientale (CNR-IMAA), Potenza - CNR-IMAA" ;
:resolution_evaluated = "" ;
:evaluation_method = "" ;
:input_parameters = "" ;
:comment = "" ;
:measurement_ID = "20190613po19" ;
:measurement_start_datetime = "2019-06-13T19:44:37Z" ;
:measurement_stop_datetime = "2019-06-13T21:29:27Z" ;
:PI = "Aldo Amodeo" ;
:PI_affiliation = "Consiglio Nazionale delle Ricerche - Istituto di Metodologie per l\'Analisi Ambientale " ;
:PI_affiliation_acronym = "CNR-IMAA" ;
:PI_address = "Contrada S.Loja, Zona Industriale - Tito Scalo I-85050 Potenza" ;
:PI_phone = "+39 0971 427263" ;
:PI_email = "aldo.amodeo@imaa.cnr.it" ;
:Data_Originator = "amato" ;
:Data_Originator_affiliation = "" ;
:Data_Originator_affiliation_acronym = "" ;
:Data_Originator_address = "" ;
:Data_Originator_phone = "" ;
:Data_Originator_email = "" ;
:data_processing_institution = "Consiglio Nazionale delle Ricerche - Istituto di Metodologie per l\'Analisi Ambientale (CNR-IMAA)" ;
:hoi_system_ID = 74 ;
:hoi_configuration_ID = 124 ;
:scc_version = "5.1.0" ;
:scc_version_description = "SCC vers. 5.1.0 (HiRELPP vers. 1.0.6, CloudMask vers. 1.1.8, ELPP vers. 7.0.5, ELDA vers. 3.3.9, ELIC vers. 1.0.3, ELQUICK vers. 1.0.3, ELDEC vers. 2.0.1)" ;
:processor_name = "ELDA" ;
:processor_version = "3.3.9" ;
:__file_format_version = "2.0" ;
:input_file = "20190613po19_0000470.nc" ;
:overlap_correction_file = "" ;

Mandatory Dimensions, Variables and Global attributes :

DIMENSIONS:	
time	MANDATORY
wavelength	MANDATORY
altitude	MANDATORY
nv	MANDATORY

VARIABLES:	
double time(time)	MANDATORY
double time_bounds(time, nv)	MANDATORY
double altitude(altitude)	MANDATORY
double backscatter(wavelength, time, altitude)	MANDATORY for b product
double error_backscatter(wavelength, time, altitude)	MANDATORY for b product if backscatter is defined
double extinction(wavelength, time, altitude)	MANDATORY for e product
double error_extinction(wavelength, time, altitude)	MANDATORY for e product if extinction is defined
double lidarratio(wavelength, time, altitude)	Computed by the database
double error_lidarratio(wavelength, time, altitude)	Computed by the database
float backscatter_calibration_range(wavelength, nv)	MANDATORY
float backscatter_calibration_value(wavelength)	MANDATORY
float backscatter_calibration_search_range(wavelength, nv)	MANDATORY
int quality_control_level	Added by the database
int technical_quality_control	Added by the database
int physical_quality_control	Added by the database
int earlinet_product_type	MANDATORY
byte atmospheric_molecular_calculation_source	MANDATORY
byte error_retrieval_method(wavelength)	MANDATORY
byte backscatter_evaluation_method(wavelength)	MANDATORY if backscatter is present
byte raman_backscatter_algorithm(wavelength)	MANDATORY if backscatter_evaluation_method = 0
byte elastic_backscatter_algorithm(wavelength)	MANDATORY if backscatter_evaluation_method = 1
byte extinction_evaluation_algorithm(wavelength)	MANDATORY if extinction is present
byte backscatter_calibration_range_search_algorithm(wavelength)	MANDATORY

GLOBAL ATTRIBUTES:		
NAME	TYPE	
Conventions	string	MANDATORY
title	string	MANDATORY
source	string	MANDATORY
references	string	MANDATORY
history	string	MANDATORY
station_ID	string	MANDATORY
location	string	MANDATORY
system	string	MANDATORY
institution	string	MANDATORY
comment	string	MANDATORY
measurement_start_datetime	string	MANDATORY
measurement_stop_datetime	string	MANDATORY
PI	string	MANDATORY
PI_affiliation	string	MANDATORY
PI_email	string	MANDATORY
Data_Originator	string	MANDATORY
Data_Originator_affiliation	string	MANDATORY
Data_Originator_email	string	MANDATORY
hoi_system_ID	integer	MANDATORY
hoi_configuration_ID	integer	MANDATORY

EARLINET product publication policy

When a product is submitted to the EARLINET database the following steps occur:

1. technical quality controls are executed to ensure the product is compliant mainly from technical point of view with the defined standard (for more details on all the checks implemented in the technical quality controls see ***New_Quality_Checks_description.pdf***)
2. advanced quality controls are executed to assess the quality from a physical point of view of the product (for more details on all the checks implemented in the advanced quality controls see ***New_Quality_Checks_description.pdf***)
3. The 3 variables “quality_control_level”, “basic_quality_control”, and “advanced_quality_control” are computed according to the results of the previous two steps and added into the product.
4. The product is renamed according to the filename conventions described earlier and definitively stored into the datacenter

Products not passing the step 1. are not accepted by the datacenter and a corresponding error message is shown to the data originator.
Products passing the step 1. but not passing the step 2. are accepted by the datacenter and labelled as ***Level 1*** *products*

Products passing both steps 1. and 2. are accepted by the datacenter and labelled as ***Level 2*** *products.*
Both Level 1 and Level 2 products are made public as soon as they are accepted by the datacenter. This means that a product once submitted and accepted by the datacenter (independently of the assigned level) cannot be deleted anymore.
It is possible to submit a new version of an already submitted product only in specific time window communicated by the datacenter administrator.

EARLINET product versioning

A specific Record Version Control system has been developed to allow multiple version of the same product. This is a primary and nececessary tool both for data originators and end-users. Indeed, sometimes it can happen that data originators may realize that something is wrong or not optimized in the products already uploaded ondatabase. Besides, if a new version of the retrieval algorithm is released, for example with a new SCC version release, products need to be re-analysed.
It must be remarked that *if the submitted product is a new version of an already uploaded product it will be accepted ONLY IF the submission is done in a specific time slot decided by the datacenter management group. Submission of new product versions will be NOT possible outside these specific time slots.*

EARLINET datacenter interfaces

1. https://upload.earlinet.org
   Interface the data originators should use to submit their product to the EARLINET datacenter (for more information see ***EARLINET_Uploading_Interface.pdf***)
2. https://data.earlinet.org
   Interface to browse, download, plot the products in the EALINET datacenter (for more information see ***EARLINET_Dataportal_Interface.pdf****)*

EARLINET stations code

code	location
ABY	Aberystwyth, United Kingdom
AKY	Antikythera, Geece
ARR	Andøya, Norway
ATZ	Athens, Greece
BGD	Belgrade, Serbia
BRC	Barcelona, Spain
BUJ	Burjassot, Spain
CBW	Cabauw, Netherlands
CLJ	Cluj-Napoca, Romania
COG	Belsk, Poland
CTN	Astrophysical Laboratory Catania, Italy
DUS	Dushanbe, Tajikistan
EVO	Evora, Portugal
FIK	Finokalia, Greece

GAR	Garmisch-Partenkirchen, Germany
GRA	Granada, Spain
HBB	Hamburg-Bergedorf, Germany
HBU	Hamburg, Germany
HPB	Observatory Hohenpeissenberg, Germany
INO	Bucharest, Romania
IPR	Ispra, Italy
JFJ	Jungfraujoch, Switzerland
KUH	Kuehlungsborn, Germany
KUO	Kuopio, Finland
LAQ	L'Aquila, Italy
LEI	Leipzig, Germany
LIM	Limassol, Cyprus
LIS	Lisbon, Portugal
LKP	Linkoping, Sweden
LLE	Lille, France
LMP	Lampedusa, Italy
MAS	Minsk, Belarus
MDR	Madrid, Spain
MEL	Melpitz, Germany
MUC	Maisach, Germany
MUN	Munich, Germany
NAP	Naples, Italy
NCL	Nicolosi, Serra La Nave Observatory, Italy
NEU	Neuchatel, Switzerland
OHP	Haute-Provence, France
PAY	Payerne, Switzerland
POT	Potenza, Italy
PUY	Clermont-Ferrand, France
RME	Roma-Tor Vergata, Italy
SAL	Lecce, Italy
SIR	Palaiseau, France
SOF	Sofia, Bulgaria
SPL	St. Petersburg, Russia
THE	Thessaloniki, Greece
UCC	Cork, Ireland
WAW	Warsaw, Poland