

DOCUMENT

EO generic RAW and Lo specification

Prepared by	EOP-PEP Team
Reference	PE-TN-ESA-GS-586
Issue	1
Revision	3
Date of Issue	23-03-2026
Status	Approved/Applicable
Document Type	TN

Distribution **EOP-P**

APPROVAL

Title EO generic RAW and Lo specification	
Issue 1	Revision 3
Author EOP-PEP	Date 23-03-2026
Approved by	Date 23-03-2026
Michele Zundo	  Digitally signed by Michele.Zundo Date: 2026.10.19 18:34:04 +03'00'

CHANGE LOG

Reason for change	Issue	Revision	Date
new	1	0	22-02-2017
editorials and milliseconds separator	1	1	12-12-2019
Synchronisation with FORUM FFSTD Tailoring	1	2	17-04-2024
Clarification on Lo	1	3	23-03-2026

CHANGE RECORD

Issue 1	Revision 0		
Reason for change	Date	Pages	Paragraph(s)
new	22-02-2017	all	all

Issue 1	Revision 1		
Reason for change	Date	Pages	Paragraph(s)
Use of letter d instead of decimal dot	12-12-2019	10-11	all
correction from microseconds to milliseconds	12-12-2019	11	all
editorials	12-12-2019	all	all

Issue 1	Revision 2		
Reason for change	Date	Pages	Paragraph(s)
Update of FFST from 3.0 to 3.0.1	17-04-2024		
Update of category and descriptors for RAW	17-04-2024		Sec 5.1.3.1 and 5.1.3.2
Update of length of instance id for RAW shape	17-04-2024		Section 5.1



Issue 1		Revision 3	
Reason for change	Date	Pages	Paragraph(s)
Update of FFST from 3.0.1 to 3.2	23-03-2026		
Improved text in introduction	23-03-2026		1
Added reference to Lo E2ES vs Operational Format	23-03-2026		3.2
Added Lo figure	23-03-2026		3.2
Added reference to RAW CFDP PDUs files	23-03-2026		3 and 5



Table of contents:

1	INTRODUCTION.....	5
2	DOCUMENTS.....	6
2.1	Applicable Documents	6
2.2	Reference Documents	6
3	DATA CONTENT DEFINITION	7
3.1	RAW data.....	7
3.1.1	Space Packet RAW data files	7
3.1.2	Other RAW data files	7
3.2	Level 0 data	8
4	FILE STRUCTURE	9
4.1	Logical vs Physical Files	9
4.2	Header	9
4.3	Data Block	9
4.4	Packaging and Distribution of Files.....	9
5	FILE NAMING.....	9
5.1	Logical File Names	9
5.1.1	Mission ID	9
5.1.2	File Class	9
5.1.3	File Type.....	10
5.1.3.1	File Category	10
5.1.3.2	Product Semantic Descriptor.....	10
5.1.4	File Instance ID.....	12
5.1.4.1	File Instance ID for RAW or Level 0 data files.....	12
5.1.4.1.1	RAW files.....	12
5.1.4.1.2	Level 0 files.....	13



1 INTRODUCTION

The purpose of this technical note (TN) is to define a common, simple exchange format for RAW and Level 0 data files transporting the data exchanged on the S2G link.

The space-to-ground (S2G) interface is a critical technical and contractual interface for EO missions. Data transferred across this interface shall comply with the applicable technical standards and best practices at every layer, from the RF link to the data format.

Standardisation and harmonisation across all relevant onboard and ground subsystems and stakeholders are therefore essential. Onboard stakeholders include the satellite, instruments, mass memory, modulators and transmitters, and avionics in general. Ground stakeholders include EGSE, test-data generators, acquisition systems and front-end processors (FEPs) at ground stations, AIT and AIV teams, E2ES and processor-prototype teams, operational-processor teams, and commissioning teams. This common approach enables easy and efficient data exchange, supports the reuse of common tools, and avoids a proliferation of specifications.

While CCSDS and ECSS standards define the formatting of data units and the transport and coding layers, they do not define a common file format and naming convention for RAW and Level 0 data that preserves content bit-for-bit as transmitted over the S2G link.

This TN therefore defines a generic file-level baseline for this system-level interface, covering both file formats and naming conventions. Its objectives are to:

1. preserve content that is **bit-for-bit identical** to the data transmitted over the S2G link;
2. **avoid mission-specific** or vendor-specific **implementations**; and
3. introduce **zero or minimal overhead** and allow data to be operated on and **accessed directly at file level**, without additional or external software libraries.

The container format and file-naming logic for RAW and Level 0 data are based on the ESA Earth Observation File Format Standard [AD-1]. The data content is based on the applicable CCSDS and ECSS standards [AD-3] and [AD-4]. This baseline format is used by the E2ES and processor-prototype teams.

A Ground Segment subsystem or application may derive an internal variant from these baseline types, for example an operational Level 0 format with additional fields or annotations for legacy compatibility. The interface baseline shall comply with this TN, and the derivation process and technical means shall be documented to preserve an efficient exchange format.

This TN shall be used as an input to the mission-specific tailoring of the Filename and Format Standard.

2 DOCUMENTS

2.1 Applicable Documents

	Title	Reference	Issue
[AD-1]	Ground Segment File Format Standard	PE-TN-ESA-GS-001	3.2
[AD-2]	Handbook for EO XML and Binary Schemas	PE-TN-ESA-GS-121	1.7.1
[AD-3]	CCSDS Standards	all	
[AD-4]	ECSS Standards + Adoption Notes	all	

2.2 Reference Documents

	Title	Reference	Issue

3 DATA CONTENT DEFINITION

3.1 RAW data

3.1.1 *Space Packet RAW data files*

RAW data files are defined as data files containing pure binary CCSDS/PUS Space Packets:

- appended one after the other in the order as transmitted on the RF space to ground link
- without headers,
- without annotation or any other interleaved data
- without using container formats.

Any associated metadata e.g. GS Headers, information, annotation generated by EGSE or by the Ground Station is to be stored in a physical separated file and not within the RAW data file itself.

Each RAW data file can contain either all APIDs as generated on board or only a subset of these (e.g. es result of filtering performed by the EGSE or Ground Acquisition system). Each RAW data file contains data pertaining to a single CCSDS Virtual Channel

The size of each RAW data files is variable and can either be a single (generally corresponding to one acquisition) or chunked for convenience in blocks of smaller size (e.g. 100 Mbyte)

3.1.2 *Other RAW data files*

Although less common and used primarily for diagnostic it is possible to define RAW data containing other CCSDS data units in same way that the application layer (Space packets) structure have been defined. Therefore we have RAW data files containing CFDP PDUs, reduced PDUs, Transfer Frames and coded CADUs.

In these cases, the same principle will be used and the corresponding RAW files will contain a pure binary CCSDS/PUS PDUs, Transfer Frames or CADUs appended one after the other in the order as transmitted on the RF space to ground link without headers, without annotation or any other interleaved data and without using container formats.

3.2 Level 0 data

A Level 0 Data file consist of RAW data with Space Packets sorted (per SSC and Timestamp if applicable), duplicate and corrupted (as per CRC) removed.

Level 0 data is naturally split per data type based on specific mission needs and depending different user of that specific Level 0 file, for example different Level 0 data files could contain Space Packets grouped per APID, per VC, or the one correspondingly to HTKM, Measurement Modes, Calibration Modes, Instrument or Platform Ancillary, Navigation, etc.

Level 0 data file have a simplified standard structure in the context of the E2E Simulation (Header + data) while the operational format used in the operational ground segment follows a specific data model and possibly bespoke packaging/format (SAFE, NetCDF, etc).



Figure 1 Data Files



4 FILE STRUCTURE

4.1 Logical vs Physical Files

The content of [AD-1] fully applies to RAW and Level 0, although RAW data are not required to have an header.

4.2 Header

The content of [AD-1] fully applies to Level 0 but not to RAW

4.3 Data Block

The content of [AD-1] fully applies, noting that the file name extension, which will be “raw” for RAW files.

4.4 Packaging and Distribution of Files

The content of [AD-1] fully applies: baseline packaging and distribution is use of ‘zip’.

5 FILE NAMING

5.1 Logical File Names

The content of [AD-1] fully applies. The file naming convention for RAW/Level 0 products is therefore identified by the sequence of fields described here below:

<MMM>_<CCCC>_<TTTTTTTTTT>_<instance ID>

where:

<MMM>	= Mission ID (3 characters)
<CCCC>	= File Class (4 characters)
<TTTTTTTTTT>	= File Type (10 characters)
<instance ID>	= File Instance ID (28 characters for the RAW, while for Lo it is mission specific).

The above fields constitute the smaller set of information which ensures that each logical file name is unique, within the context of the Ground Segment.

5.1.1 Mission ID

The Mission ID consists of 3 characters, either uppercase letters, digits or underscores “_”.

5.1.2 File Class

The File Class element indicates the type of activity for which the file is used. It consists of 4 characters, either uppercase letters or digits.



The following File Class are identified:

- TEST for internal testing
- OGCA for on-ground calibration
- GSOV for ground segment overall validation, system level testing
- EGSE for data related to on-ground instrument validation usage
- OPER for operational processing
- NRTI for near-real time processing
- OFFL for offline processing
- RPRO for reprocessing

Other file classes could be defined if needed.

5.1.3 File Type

The File Type element identifies the product and consists of 10 characters, either uppercase letters, digits or underscores “_”.

For RAW and Level 0, the File Type can be subdivided into two sub-elements of respectively 4 and 6 characters, as follow:

`<TTTTTTTTTT> = <FFF><DDDDDDDD>`

where:

`<FFF>`: File Category
`<DDDDDDDD>`: Product Semantic Descriptor

5.1.3.1 File Category

The File Category element consists of 3 characters (3 uppercase letters, digits or underscores “_”).

For RAW, the following file categories have been identified:

- RAW for raw binary CCSDS data
- Lo_ for instrument level 0 instrument data files (science data products)

5.1.3.2 Product Semantic Descriptor

The Product Semantic Descriptor must be unique for a given File Type and be as descriptive as possible. It consists of 7 characters, either uppercase letters, digits or underscores “_”. (if possible, the first character should be an underscore “_”)

For RAW data files identified with the File Type element set to “RAW_”, the Product Semantic Descriptor is defined based on the RF link and CCSDS data type as follows:

- __XB_CAD: XB CADU data
- __SB_CAD: SB CADU data



- `_XB_TF_`: XB Transfer Frame data
- `_SB_TF_`: SB Transfer Frame data
- `_XB_SP_`: XB Space Packet data
- `_SB_SP_`: SB Space Packet data
- `_XB_PD_`: XB CFDP PDUs
- `_SB_PD_`: SB CFDP PDUs

NB: The semantic descriptors above covers the RF links more commonly used. In case other frequencies are used (e.g. K-Band or Optical Laser Link, etc) and if there is the danger of ambiguities, the Semantic descriptor should be tailored accordingly e.g. `_KB_SP_` etc..

For Level 0 data files identified with the File Type element set to “Lo_”, the Product Semantic Descriptor is defined based on the APID and data type as follows for the generic types while for observation and calibration modes are mission specific and need to be tailored per every mission also considering the case of missions with multiple payloads on board:

- `_ANCINS`: Instrument Ancillary data
- `_NAVATT`: NAVATT data
- `_ANCSAT`: Satellite platform ancillary data
- `_HKTM__`: HKTM data
- `_XXXXX1`: *Observation Data Type 1*
- `_XXXXX2`: *Observation Data Type 2*
- `_CALXX1`: *Calibration Data Type 1*
- `_CALXX2`: *Calibration Data Type 2*
- *etc*



5.1.4 File Instance ID

For RAW and Level 0, the following shapes are suggested for the File Instance ID.

The first one is used for RAW and Lo files containing observation and calibration data generated by XB Ground Acquisition Station or by the IDS, the second to Lo files containing ancillary and HKTM packet.

5.1.4.1 File Instance ID for RAW or Level 0 data files

5.1.4.1.1 RAW files

For RAW data file with the File Type set “RAW_” and semantic descriptor XB_SP__ or SB_SP__ the suggested File Instance ID consists of 28 characters, either uppercase letters, digits or underscores “_”, with the one of the following 3 shapes:

```
<APIDNNNN>_<YYYYMMDDTHHMMSSdmmm>
<VCID__NN>_<YYYYMMDDTHHMMSSdmmm>
<CADU____>_<YYYYMMDDTHHMMSSdmmm>
<CFDP_PDU>_<YYYYMMDDTHHMMSSdmmm>
<CFDPRPDU>_<YYYYMMDDTHHMMSSdmmm>
```

This suggested instance shape can be further tailored if there is the need e.g. to differentiate between the source of the data.

where:

<APIDNNNN>: Space Packet APID in decimal:

- 8 digits corresponding to the decimal APID. In case the file includes multiple APID the value shall be the string “MULTIPLE”

<VCID__NN>: Virtual Channel ID in decimal:

- 8 digits corresponding to the decimal Virtual Channel ID. In case the file includes multiple VCID the value shall be the string “MULTIPLE”
-

<CADU____>: Fixed string

<CFDP_PDU>: Fixed string when all CFDP PDUs downlinked are contained interleaved in the file (both data and directive PDUs)

<CFDPRPDU>: Fixed string when the content is only the CFDP Reduced PDUs required for closing the Class 2 protocol.



< YYYYMMDDTHHMMSSdmmm>: file generation (start) time, consisting of 19 characters, either uppercase letters or digits:

- 8 digits for the date: “YYYYMMDD”, year, month, day
- 1 uppercase T: “T”
- 6 digits for the time: “HHMMSS”, hour, minutes, seconds
- 1 lowercase “d” as decimal separator
- 3 digits for millisecond: “mmm”

5.1.4.1.2 *Level 0 files*

For Level 0 files the definition of instance ID is left to the mission specific tailoring of the Filename and Format Standard.