CEOS ICF

Baseband Data Archive Interchange Format Description

Prepared by CEOS WGISS
Archive Task Team

THIS PAGE IS INTENTIONALLY LEFT BLANK

DOCUMENT STATUS SHEET

Issue	Date	Status
1.0	11 Aug 2002	Original Issue

THIS PAGE IS INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

1.1 PURPOSE OF THIS DOCUMENT 9
1.2 DOCUMENT OVERVIEW 9
1.3 REFERENCE DOCUMENTS 9
1.4 ABBREVIATIONS AND ACRONYMS 11
1.5 DEFINITIONS 12
2.1 RATIONALE 13
2.2 PRIMARY REQUIREMENTS 13
3.1 HARD-DISK ORGANIZATION (NORMATIVE) 16
3.2 MAGNETIC TAPE ENCAPSULATION (INFORMATIVE) 19
3.3 NETWORK TRANSFER AND INTEROPERABILITY (INFORMATIVE) 20
4.1 GENERAL DESCRIPTION 21
4.2 BASIC PRINCIPLES 22
4.3 SI UNITS AND REFERENCE SYSTEMS 23
4.4 GLOBAL DATA TYPES 25
4.5 METADATA FILE PROLOG 28
4.6 CEOS_ICF ROOT ELEMENT 28
4.7 PLATFORM ELEMENT 31
4.8 SENSOR ELEMENT 34
4.9 ORBIT ELEMENT 36
4.10 ELEMENT_SET ELEMENT 38
4.11 EPHEMERIS ELEMENT 41
4.12 ATTITUDE_SET ELEMENT 44
4.13 ATTITUDE ELEMENT 46
4.14 POSITION ELEMENT 49
4.15 VELOCITY ELEMENT 51
4.16 PROCESSING_LOG ELEMENT 54
4.17 PROCESSING ELEMENT 56
4.18 FACILITY ELEMENT 59
4.19 SYSTEM ELEMENT 61
4.20 SIGNAL_DATA ELEMENT 64
4.21 SUBSET ELEMENT 67
4.22 BLOCK ELEMENT 69
5.1 RATIONALE 80
5.2 MAPPING OF THE SIGNAL DATA RECORDS 80
5.3 SIGNAL DATA RECORDS PROPERTIES 82
5.4 PARTITIONING SIGNAL DATA FILE 83

THIS PAGE IS INTENTIONALLY LEFT BLANK

TABLE OF FIGURES

fig. 1 - Interchange between several facilities and their archive formats 13
fig. 2 - Component Diagram of CEOS ICF 16
fig. 3 – Class diagram of CEOS ICF Root Element 28
fig. 4 - Class diagram of Platform 31
fig. 5 - Class diagram of Sensor 34
fig. 6 - Class diagram of Orbit 36
fig. 7 - Class diagram of Element_set 38
fig. 8 - Class diagram of Ephemeris 41
fig. 9 - Class diagram of Attitude_set 44
fig. 10 - Class diagram of Attitude 46
fig. 11 - Class Diagram of Position 49
fig. 12 - Class Diagram of Velocity 51
fig. 13 - Class diagram of Processing_log 54
fig. 14 - Class diagram of Processing 56
fig. 15 - Class diagram of Facility 59
fig. 16 - Class diagram of System 61
fig. 17 - Class diagram of Signal_data 64
fig. 18 - Class diagram of subset 67
fig. 19 - Class Diagram of block 69
fig. 20 – Telemetry mapping 73
fig. 21 – Full Metadata File model 88

THIS PAGE IS INTENTIONALLY LEFT BLANK

1 INTRODUCTION

1.1 Purpose of this document

This document describes the "Baseband Archive Interchange Format" (CEOS ICF) recommended by the Committee on Earth Observation Satellites (CEOS). It will be used as a reference for CEOS ICF implementation. In order to facilitate the comprehension and the use of this new format, the strict specifications of structure and syntax of CEOS ICF are accompanied by a description of the objectives as well as scenarios and examples.

1.2 Document Overview

- Section 1 introduces the current document and identifies related resources;
- Section 2 presents CEOS ICF concepts;
- Section 3 describes the physical layout of the format;
- Section 4 specifies the CEOS ICF Metadata File format;
- Section 5 specifies the CEOS ICF Signal Data File format;
- Appendices print normative files related to CEOS ICF and informative samples.

1.3 Reference documents

This section lists the applied conventions and reference documents necessary to adequately comprehend the CEOS ICF specifications.

R-1 ATT Baseband paper
Baseband Data Concept
White Paper – May 6, 1999
Archive Task Team
Working Group on Information Systems and Services
Committee on Earth Observation Satellite

R-2 CEOS.WGISS.DS.TN01
Guidelines on Standard Formats and Data Description Languages
Version C Draft – September 3, 1997
Working Group on Information Systems and Services
Committee on Earth Observation Satellite
http://wgiss.ceos.org/documents/s&g/formguid.pdf

R-3 REC-xml-20001006
Extensible Markup Language (XML) 1.0 (Second Edition)
W3C Recommendation, Version 1.0 – October 6, 2000
World Wide Web Consortium
http://www.w3.org/TR/2000/REC-xml-20001006

R-4 CR-xmlschema-0-20010502
XML Schema Part 0: Primer
W3C Proposed Recommendation – May 2nd, 2001
World Wide Web Consortium
http://www.w3.org/TR/2001/PR-xmlschema-0-20010502/

R-5 CR-xmlschema-1-20010502
XML Schema Part 1: Structures
W3C Proposed Recommendation – May 2nd, 2001
World Wide Web Consortium
http://www.w3.org/TR/2001/PR-xmlschema-1-20010502/

R-6 CR-xmlschema-2-20010502
XML Schema Part 2: Datatypes
W3C Proposed Recommendation – May 2nd, 2001
World Wide Web Consortium
http://www.w3.org/TR/2001/PR-xmlschema-2-20010502/

R-7 PR-xlink-20001220
XML Linking Language (Xlink)
W3C Proposed Recommendation – December 20, 2000
World Wide Web Consortium
http://www.w3.org/TR/2000/PR-xlink-20001220

R-8 CR-xsl-20001121
Extensible Stylesheet Language (XSL)
W3C Candidate Recommendation – November 21, 2000
World Wide Web Consortium
http://www.w3.org/TR/1999/REC-xslt-19991116

R-9 REC-xslt-19991116
XSL Transformation (XSLT)
W3C Recommendation, Version 1.0 – November 16, 1999
World Wide Web Consortium
http://www.w3.org/TR/1999/REC-xslt-19991116

R-10 REC-xml-names
Namespaces in XML
W3C Recommendation – January 14, 1999
World Wide Web Consortium
http://www.w3.org/TR/1999/REC-xml-names-19990114

R-11 OMG-UMLv1.3
Unified Modeling Language (UML)
Version 1.3 – Mars 1, 2000
Object Management Group
ftp://ftp.omg.org/pub/docs/formal/00-03-01.pdf

R-12 ISO-8601
Data Elements And Interchange Formats
Information interchange
Representation of dates and times
First Edition – June 15, 1988
International Organization for Standardization
http://www.iso.ch/markete/8601.pdf

R-13 NIMA TR8350.2
Department of Defense
World Geodetic System 1984
Its Definition and Relationships with Local Geodetic Systems
Third Edition, Amendment 1 – January 3, 2000
National Imagery and Mapping Agency
ftp://164.214.2.65/pub/gig/tr8350.2/wgs84fin.pdf

R-14 BIPM SI 1998
The International System Of Units (SI)
Seventh Edition – 1998
Bureau International des Poids et Mesures
http://www.bipm.fr/pdf/si-brochure.pdf

R-15 RFC 2396
Uniform Resource Identifiers (URI) : Generic Syntax
Updates 1808, 1738 – August 1998
Network Working Group
http://www.ietf.org/rfc/rfc2396.txt

1.4 Abbreviations and Acronyms

This section controls the definition of all abbreviations and acronyms used within this document. Special attention has been paid to inherited abbreviations, acronyms and their definitions from international standards such as ISO, ANSI or ECSS.

API	Application Programming Interface
ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
ATT	Archive Task Team
BER	Bit Error Rate
CADU	Channel Access Data Unit (defined by synchronization codes)
CCF	Compatible Computer Format
CCSDS	Consultative Committee for Space Data Systems
CEOS	Committee on Earth Observation Satellites
CNES	Centre National d'Etudes Spatiales
CRC	Cyclic Redundancy Check
DEDSL	Data Entity Dictionary Specification Language
DLT	Digital Linear Tape
DPCM	Differential Pulse Code Modulation
EAF	Existing Archive Format
ECC	Error Correcting Code
ECSS	European Cooperation for Space Standardization
EO	Earth Observation
EOF	End Of File
EOR	End Of Record
EOT	End Of Tape
ESA	European Space Agency
FTP	File Transfer Protocol
GB	Gigabyte(s)
GSFC	Goddard Space Flight Center
HDDT	High Density Data Tape
CEOS ICF	Baseband Data Archive InterChange Format
ISO	International Organization for Standardization
IEEE	Institute of Electrical and Electronic Engineers
MDF	Metadata File
NORAD	North American Aerospace Defense Command
OBR	On-Board Recorder
PCD	Payload Control Data
PN	Pseudo-random noise

SDF	Signal Data File
SI	Système International d'Unités (International System of Units)
TBC	To Be Confirmed
TBD	To Be Defined
UML	Unified Modeling Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UTC	Universal Time Coordinated
VCDU	Virtual Channel Data Unit
WGISS	Working Group on Information Systems and Services
WRS	World Reference System
XML	Extensible Markup Language
XSL	Extensible Stylesheet Language
XSLT	Language for transforming XML documents

1.5 Definitions

This section fixes the definition of all common terms used within this document. Special attention has been paid to definitions inherited from international standards such as ISO, ANSI or ECSS.

element [R-5.Logical Structures] Each XML document contains one or more elements, the boundaries of which are either defined by a start-tag and end-tags, or for empty elements, by an empty-element tag. Each element has a type, identified by name, sometimes called "generic identifier" (GI), and may have a set of attribute specifications.

transmission format The transmission formats are those data structures, which are generated for the telemetry transfer.

2 CEOS ICF CONCEPTS

2.1 Rationale

The goal of the Baseband Data Archive Interchange Format (CEOS ICF) is to make data exchange possible between different Earth Observation archiving facilities, which do not employ the same processing systems.

CEOS ICF is an intermediate format that all archive centers can interpret. Facilities can export their archive data from their own format to CEOS ICF and conversely import foreign facilities data via CEOS ICF format (fig. 1).

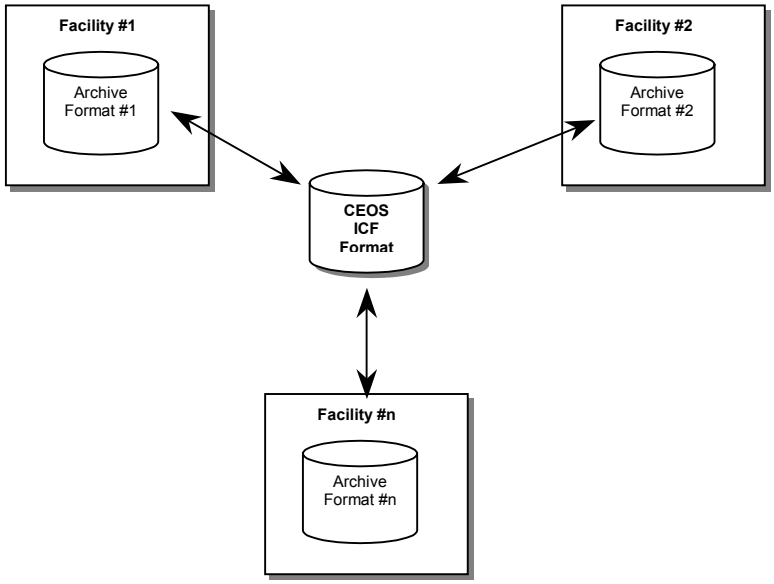


fig. 1 - Interchange between several facilities and their archive formats

The main benefit of CEOS ICF is to decrease the number of Earth Observation format translators by federating all the participant facilities around it. In order to exchange data formatted in a new Earth Observation archive format it is only required to develop an export translator to CEOS ICF format (conversely an import translator to access external Earth Observation data). Without using a common interchange format, n_A Earth Observation archive formats require the development of n_T translators:

$$n_T = \sum_{n=2}^{n_A} 2(n-1) = n_A(n_A - 1)$$

Using CEOS ICF the number of required translators is highly decreased to :

$$n_T = 2n_A << n_A(n_A - 1)$$

2.2 Primary Requirements

This section presents the important requirements constituting essential knowledge for the appropriate use of CEOS ICF. These features are necessary in order to reach the objectives defined in the preceding section.

Interchange

CEOS ICF allows interchanging archives between several facilities. It is thus not only able to represent sensor measurements and auxiliary data but also information necessary to their exploitation (origin, format, bounds, etc.). This is not to say that all the information related to measurements can be stored but at least those essential to their perfect identification and processing. This important concept makes it possible to answer the requirements "Multi-missions", "Multi-Sensors" and "Extendibility" described in the following sections.

Baseband Data Concept

CEOS ICF conforms to Baseband Data Concept [R-1] as defined by the Committee on Earth Observation Satellites (CEOS). The Archive Task Team (ATT) of the CEOS Working Group on Information Systems and Services (WGISS) has produced the Baseband Data Concept [R-1]. The main purpose of this concept is to keep the loss and destruction of original data to a minimum in order to preserve Earth Observation datasets for an indefinite period (long-term archive).

Baseband Interchange Unit

"Baseband Interchange Unit" concept has been created for the needs of CEOS ICF (see definition below). A CEOS ICF Archive contains a single "Baseband Interchange Unit" and only that.

📖 Definition: Baseband Interchange Unit (normative)

> A "Baseband Interchange Unit" is a contiguous sequence of measurements and related auxiliary data acquired during a single satellite pass over a continuous period of time.

According to the preceding definition, in order to represent several passes or several segments of acquisition it will be necessary to create as many CEOS ICF archives. This concept greatly decreases the effort required for the development of transcription tools and increases their robustness. There is therefore no multi-volume representation and thus no ambiguity concerning the number of Metadata Files or Signal Data Files nor any redundancy of information. CEOS ICF is always a coherent subset of measurements acquired in a continuous period of time.

📖 Property: Baseband Interchange Unit Subset (informative)

> A contiguous subset of "Baseband Interchange Unit" is a "Baseband Interchange Unit".

This property may be useful when an archive has to be cut for device space reasons or for constraints due to network transfer policy.

Multi-missions and multi-sensors

CEOS ICF does not assume any mission, sensor or specific parameter. Therefore CEOS ICF does not define attributes or entities like specific SPOT sensor modes, SPOT mirror steps or Landsat low/high gain switch status. However a maximum number of "generic" entities (even complex ones) have been defined and thus can be declared easily. For instance, sets of "ephemeris" vectors can be stored in common geodetic and time systems. Specific attributes/entities storage problems are solved by the requirement "Extendibility" defined here below.

Extendibility

CEOS ICF enables the definition of new entities and the extension of predefined entity definitions. That does not mean that the CEOS ICF definition can be modified but only that it is possible to extend it in order to convey very specific values. This capability guarantees that CEOS ICF will remain an interchange format for the maximum amount of missions, particularly in future missions where specific components are not yet known. The extension capability is based on a typical class inheritance mechanism well known in Object Oriented Modeling.

Extension definitions may be ensured by the organization responsible for the specific component. For example, the European Space Agency may be responsible for the ENVISAT specific attribute definition for CEOS ICF extension; CNES may be responsible for those concerning SPOT; CCSDS may be responsible of the satellite identification number, etc.

Hardware/Software and Operating System independence

CEOS ICF has been designed to be as independent as possible of hardware, software and operating systems. Suggestions and cautions are given in sections 3.1 and 3.2 below.

3 CEOS ICF FORMAT OVERVIEW

3.1 Hard-disk organization (normative)

CEOS ICF is primarily designed to be stored on hard disk or any other "file system" device. One CEOS ICF archive is nominally composed of two files - One "Metadata File" and one "Signal Data File". The following "UML Component Diagram" defines the way Signal Data File is aggregated to the Metadata File.

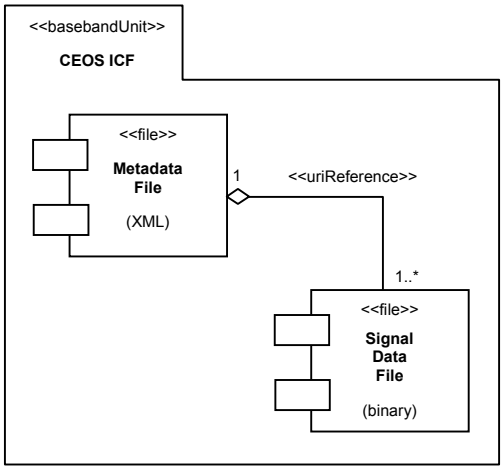


fig. 2 - Component Diagram of CEOS ICF

The format of each file is specified in sections 4 and 5 below. Due to the file system size limitations, CEOS ICF enables the partitioning of the Signal Data File into several subsets. This option is specified as well in the same sections 4 and 5.

File naming and location

📖 Definition: Metadata File name syntax (normative)

Metadata File name must respect the following grammatical rules:

```
filename   =  alphanum *(alphanum | mark) "." extension

alphanum   =  alpha | digit

alpha      =  "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" |
              "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" |
              "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z" |
              "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" |
              "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" |
              "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"

digit      =  "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
              "8" | "9"

mark       =  "-" | "_" | "."

extension  =  "xml" | "XML"
```

👍 Samples of correct Metadata File names (informative)

```
"METADATA.XML"            <- ISO9660 Compliant
"10.33.47.732.XML"
```

👎 Samples of bad Metadata File names (informative)

```
"Metadata of CEOS ICF"
".Xml"
"-Header-10.33.47.732.XML"
"A~Header.Xml"
```

There is no reason to create a new extension such as "mdf" or "icf" for CEOS ICF Metadata Files. A mandatory extension has been added in order to discriminate between Metadata Files and Signal Data Files. The "XML" extension was preferred due to the fact that the CEOS ICF Metadata File is primarily a standard XML file.

No specific rule is required for Signal Data Filename syntax except that it shall be identifiable by a URI reference [R–9]. No specific location is required for Metadata Files and Signal Data Files of CEOS ICF Archives. Maximum freedom has been allowed in order to avoid unnecessary operational constraints. Facility Managers may select a file base name compatible with their in-house configuration management or may separate Metadata Files and Signal Data Files location into several servers deployed on a network. It is however recommended where possible to:

- Set CEOS ICF files of the same archive in a single directory;
- Set only one CEOS ICF archive per directory;
- Use the following recommended naming rules;

Recommended naming rules (informative)

```
metafile    =  basename "." metaext
signalfile  =  basename [ sep subsetnum ] "." signalext

basename    =  mission sep datestart sep datestop

mission     =  alphanum *(alphanum) [ sep digit *(digit) ]

datestart   =  date
datestop    =  date

date        =  year month day hour minute second millisec
year        =  4(digit)
month       =  2(digit)
day         =  2(digit)
hour        =  2(digit)
minute      =  2(digit)
second      =  2(digit)
millisec    =  3(digit)

subsetnum   = "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

alphanum    =  alpha | digit

alpha       =  "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" |
               "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" |
               "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z" |
               "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" |
               "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" |
               "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"

digit       =  "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
               "8" | "9"

sep         =  "-"

metaext     =  "xml" | "XML"

signalext   =  "dat" | "DAT"
```

Sample of file naming and location (informative)

```
myhost:/usr/people/foo/ceos-icf-archive/
drwxrwxr-x 2 foo       1024 2001 .
drwxrwxr-x 3 foo       1024 2001 ..
-rw-rw-r-- 3 foo 2147482624 2001 ERS-1-19980225103347732-19980225102345876-1.dat
-rw-rw-r-- 3 foo     482624 2001 ERS-1-19980225103347732-19980225102345876-2.dat
-rw-rw-r-- 3 foo      20480 2001 ERS-1-19980225103347732-19980225102345876.xml
```

As explained in the following sections, other optional files can accompany the CEOS ICF archive. They should also be placed in the same directory with the same base name. Such files may be the metadata validation scheme (XML Schema), some style-sheets for metadata display (XSLT) or auxiliary files defined in any CEOS ICF extension.

File System Limitations (informative)

Many operating systems do not manage files bigger than 2 Gigabytes. For this reason, when the maximum file size of the destination facility is unknown, it is recommended to stay under the 2 Gigabytes limit and to partition archives as necessary. The partitioning of Signal Data Files is specified in sections 4 and 5 below. Note that in the previous file naming convention the “subsetnum” grammar production is the index of the partitioned Signal Data File.

3.2 Magnetic Tape encapsulation (informative)

CEOS ICF archives can be written on magnetic tape medium (DLT, Exabyte, etc) using "tape archiving tools". By "tape archiving tool" we mean a software tool able to encode and restore a subset of a file system on a tape. Encoding and restoring are done through the sequential stream of the tape drive mount point (device). The tape archiving tool must be able to encode and restore the entire file directory tree (generally processed recursively).

No special tool or tape archive format is obligatory. Facilities participants at each interchange should agree a common tool. This freedom is permitted in order to allow CEOS ICF to be stored by the maximum number of media and accessed by most operating systems. The following list gives an overview of the available software according to operating systems.

Tape archive tools (informative)

***UNIX**

tar	**UNIX standard tape archive**
gtar	GNU tar (**free** tar compatible tool)
EST BRU	(BSD, HP-UX, Linux, SCO, Solaris)
Legato Networker	(HP-UX, SCO, Solaris)
Cactus International Lone-Tar	(HP-UX, Linux, SCO)
Microlite Backup Edge	(Linux, SCO)
Yosemite TapeWare	(Linux)

***Microsoft Windows 95/98/2000**

Microsoft Windows Backup	**Microsoft Windows standard tape archive**
Retrospect	Dantz
Veritas Backup Exec	
NovaBackup	Novastor
NovaNET	Novastor
ArcserveIT	Computer Associates
Yosemite TapeWare	

***Microsoft Windows NT 4.0**

Microsoft NT Backup	**Microsoft Windows NT standard tape archive**
Retrospect	Dantz
Veritas Backup Exec	
NovaBackup	Novastor
NovaNET	Novastor
ArcserveIT	Computer Associates
Yosemite TapeWare	

***Mac OS**

Dantz Retrospect	**Apple standard backup system**

**The previous list does not purport to be exhaustive but only gives an overview of existing tools (free or off-the-shelf)*

However it is recommended as far as possible to encode tape archives in the tar format that is supported by a major part of operating systems used by Earth Observation facilities.

Hardware Compression

Many tape drives support data compression while transferring data. A compressed tape archive may not be readable by any other tape drive. Therefore it is recommended to use the less compressed mode (generally the default one) while writing a tape except in case of mutual agreement between facilities.

Tape archives limitations

As with file systems, tape archive tools may have a maximum file size limitation. It is recommended to maintain a maximum file size of 2 Gigabytes. In such cases it is not recommended to use the CEOS ICF partitioning mechanism to dispatch subsets to several tape archives. In fact this mechanism is based on "uriReference" type defined by XML Schema, which does not support tape resource identification. The multi-volume capability of the selected tape archive tool should therefore be used to maintain archive integrity.

File order

Some tape archive tools such as standard UNIX `tar` command do not assume any file order (generally file creation order). It is however recommended to force Metadata File storage first. This enables basic inspection or identification of archive contents without restoring them entirely.

3.3 Network transfer and interoperability (informative)

CEOS ICF archives are easy to transfer via the network using existing protocols such as ftp or http. Both metadata files as well as signal data files may be encapsulated in an archive pseudo system as described in section 3.2 above. However in order to accelerate signal data file recognition and interpretation through the network, the metadata file will be sent first if both files are not encapsulated in the same data archive (e.g. tar archive). The metadata file may be sent alone only when general information about an existing archive is required. In such case the URI reference mechanism between metadata files and signal data files will be used to reference the signal data file. This may occur in a distributed environment where several stations or facilities share the same archives avoiding signal data transfer.

4 METADATA FILE

4.1 General description

Metadata File of a CEOS ICF archive contains information related to the Baseband Interchange Unit. It describes the linked Signal Data File, identifies the physical components involved, such as the sensor or platform and provides administrative information regarding the processing facilities.

File format

CEOS ICF Metadata File is an XML document [R-3.]. XML documents are composed of storage units called "elements", which contain either parsed or unparsed data. Parsed data is made up of "character data" and "markup". Markup encodes a description of the document's storage layout and logical structure. XML provides a mechanism to impose constraints on the storage layout and logical structure. This format has been selected for several reasons but in particular:

- XML is supported by a wide variety of applications;
- It is easy to write programs which process XML documents;
- XML documents are human-legible and reasonably clear;
- XML documents terseness and structure can be described and validated.

CEOS ICF format assumes that Metadata File is "Well-formed" according to the document [R-3].

Namespace

All CEOS ICF markups are defined in a specific namespace. The goal of this namespace is to separate CEOS ICF "markup vocabulary" from any other defined elsewhere. See document [R-10] for detailed information about the interest and the use of XML namespaces. CEOS ICF namespace is identified by:

http://wgiss.ceos.org/ceos-icf-200208211

As for almost all others, CEOS ICF namespace has a URI syntax [R-6]. That does not assume correspondence to any file or data source. Interest is only focused on the uniqueness of this namespace identifier.

Validation

CEOS ICF Metadata File can be validated using the XML Schema provided in appendix A. XML Schema specifications provided by documents [R-4], [R-5] and [R-6] are recommendations of the World Wide Web Consortium. CEOS ICF XML Schema is identified by the following URI :

http://wgiss.ceos.org/ceos-icf/ceos-icf-20020811.xsd (TBC)

The targeted resource can be downloaded for validation of any occurrence of CEOS ICF Metadata File. According to XML Schema specifications [R-10] `xsi:schemaLocation` attribute can be used in the Metadata File to permit validation tools or XML processors to automatically retrieve the schema. Refer to document [R–10] for a complete definition of the `xsi:schemaLocation` attribute. The example below illustrates the use of this attribute within CEOS ICF Metadata File:

Sample of use of namespace and schema location within CEOS ICF (informative)

```
<ceos_icf
   xmlns="http://wgiss.ceos.org/ceos-icf-20020423"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://wgiss.ceos.org/ceos-icf-20020423
        http://wgiss.ceos.org/ceos-icf-20020423.xsd">
        …
</ceos_icf>
```

4.2 Basic principles

Markups naming convention

All markups of CEOS ICF Metadata Files respect the following rules:

- Characters are always lowercase
- Words composing the markup are separated by an underscore character
- Abbreviations and acronyms are only authorized when they are sets of initials

Sample of correct CEOS ICF markups (informative)

```
<ceos_icf>
<nssdc_identifier>
<sensor>
<element_set>
```

CEOS ICF extensions should respect the same markup naming convention.

Extendibility

As expected CEOS ICF Metadata File definition can be extended. "Extends" means adding information. In order to maintain CEOS ICF integrity it is not possible to override its definitions. The extendibility mechanism is useful to store information specific to the sensor or to the facility that is involved in the archive processing. For instance in a standard CEOS ICF archive, sensor declaration should be made as follows:

Sample of correct sensor instance (informative)

```
<ceos_icf
   xmlns="http://wgiss.ceos.org/ceos-icf-20020423"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://wgiss.ceos.org/ceos-icf-20020423
        http://wgiss.ceos.org/ceos-icf-20020423.xsd">
        …
        <sensor>
               <family_name>HRV</family_name>
               <number>2</number>
        </sensor>
        …
</ceos_icf>
```

Some specific parameters may be added without consequence even if not specified by CEOS ICF:

Correct sensor extension (informative)

```
<ceos_icf
   xmlns="http://wgiss.ceos.org/ceos-icf-20020423"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:spot="http://www.cnes.fr/spot"
   xsi:schemaLocation="http://wgiss.ceos.org/ceos-icf-20020423
        http://wgiss.ceos.org/ceos-icf-20020423.xsd">
        …
        <sensor>
               <family_name>HRV</family_name>
               <number>2</number>
               <spot:mirror_step value="23"/>
               <spot:sensor_mode>XI</spot:sensor_mode>
        </sensor>
        …
</ceos_icf>
```

On the other hand, as shown in the following example, removing a mandatory element or changing its definition or type is strictly forbidden:

Sample of bad sensor extension (informative)

```
<ceos_icf
   xmlns="http://wgiss.ceos.org/ceos-icf-20020423"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:spot="http://www.cnes.fr/spot"
   xsi:schemaLocation="http://wgiss.ceos.org/ceos-icf-20020423
        http://wgiss.ceos.org/ceos-icf-20020423.xsd">
        …
        <sensor>
               <number value="2"/>
               <spot:mirror_step value="23"/>
               <spot:sensor_mode>XI</spot:sensor_mode>
        </sensor>
        …
</ceos_icf>
```

The complete Metadata File description provided in the following sections may be useful for a perfect understanding of the previous sample.

4.3 SI units and reference systems

CEOS ICF assumes that all quantities shall be expressed in a standard unit system (SI). The selected standard is the SI defined by the Bureau International des Poids et Mesures (BIPM). The SI units meaning and extent is fully described in the standard document [R-14]. If quantities that are not listed below have to be used, their definition will come from document [R-14].

Quantity	Unit	Symbol	Definition
Length	*metre*	*m*	The *metre* is the length of the path traveled by light in vacuum during a time interval of 1/299792458 of a second.
Mass	*kilogram*	*kg*	The *kilogram* is the unit of mass; it is equal to the mass of the international prototype of the *kilogram*.
Time	*second*	*s*	The *second* is the duration of 9192631770 periods of the radiation corresponding to the transition between the two hyperfine levels of the ground state of the cesium 133 atom.
Plane Angle	*radian*	*rad*	The radian is a SI derived units that can be expressed in terms of SI base units as $m \cdot m^{-1} = 1$.

It is also assumed that without a special prompt all physical values shall be expressed in the following reference systems :

System name	Identifier	Description
World Geodetic System (1984)	WGS 84	WGS 84 is an international geodetic reference system for the Earth. It is mainly used for practical application of mapping, charting, geopositioning and navigation. It defines a coordinate system, fundamental and derived constants, the EGM96, the ellipsoidal (normal) gravity model and lists transformations to current local datum. WGS 84 is completely defined in the document [R-13] issued by the National Imagery and Mapping Agency (NIMA) of the United State Department of Defense (US DoD). Note : the WGS84 reference system is not an inertial reference system. Vectors (in particular velocities) have to be transformed if they are expressed in inertial reference systems. The transformation may be more or less complex according to the expected accuracy.
Universal Time Coordinated	UTC	UTC is a time scale maintained by the Bureau International des Poids et Mesures (named Bureau International de l'Heure before 1988) that forms the basis of a coordinated dissemination of standard frequencies and time signals [R-14].

4.4 Global Data Types

Global Data Types are defined to avoid repetitive specification in this document and for Metadata File validation. These definitions are valid throughout the overall document.

integer

`integer` data type is the one defined by the World Wide Web Consortium for the XML Schema requirement. A complete definition is available in the document [R-6].

📖 Definition [R-6] : `integer` (normative)

> `integer` is derived from decimal by fixing the value of scale to be 0. This result in the standard mathematical concept of the integer numbers. The value space of `integer` is the infinite set {…, -2, -1, 0, 1, 2, …}. The base type of `integer` is decimal.

positiveInteger

`positiveInteger` data type is the one defined by the World Wide Web Consortium for the XML Schema purpose. Complete definition is available in the document [R-6].

📖 Definition [R-6] : `positiveInteger` (normative)

> `positiveInteger` is derived from `nonNegativeInteger` by setting the value of `minInclusive` to be 1. This result in the standard mathematical concept of the positive integer numbers. The value space of `positiveInteger` is the infinite set {1,2,3,…}. The base type of `positiveInteger` is `nonNegativeInteger`.

dateTime

`dateTime` data type is the one defined by the World Wide Web Consortium for the XML Schema purpose. Complete definition is available in the document [R-6].

📖 Definition [R-6] : `dateTime` (normative)

> `dateTime` represents a specific instant of time. The values space of `dateTime` is the space of combinations of date and time of day values as defined by the **ISO-8601** standard [R-12].

As defined above, lexical and syntactical formats are completely specified by the ISO-8601 standard [R-12]. In order to facilitate the interchange we recommend to used the following representation that is fully compatible with ISO-8601 [R-12] :

`CCYY-MM-DDThh:mm:ss.sss`

Where :

- **`[C]`** represents a digit used in the thousands and hundreds components (the "century" component) of the time element "year" ;
- **`[Y]`** represents a digit used in the tens and units components of the time element "year" ;
- **`[M]`** represents a digit used in the time element "month" ;
- **`[D]`** represents a digit used in the time element "day" ;
- **`[T]`** is used as time designator to indicate the start of the representation of the time of the day in combined date and time of day expressions ;
- **`[h]`** represents a digit used in the time element "hour" ;
- **`[m]`** represents a digit used in the time element "minute" ;
- **`[s]`** represents a digit used in the time element "second" ;
- **`[.]`** Is used to separate integer part from decimal part of time element "second". The number of decimal digits is not fixed and can be extended as needed.

Sample of `dateTime` (informative)

```
1999-03-12T09:37:29.66545

2000-11-04T19:21

2001-01-01
```

string

`string` data type is the one defined by the World Wide Web Consortium for the XML Schema purpose. A complete definition is available in the document [R-6].

Definition [R-6]: `string` (normative)

> The `string` data type represents character strings in XML. The value space of `string` is the set of finite-length sequences of characters (as defined in [R–7]) that match the Char production from [R–7]. A character is an atomic unit of communication; it is not further specified except to note that every character has a corresponding Universal Code Set code point ([ISO 10646], [Unicode] and [Unicode3]), which is an integer.

double

`double` data type is the one defined by the World Wide Web Consortium for the XML Schema purpose. A complete definition is available in document [R-6].

📖 Definition [R-6]: `double` (normative)

> The `double` datatype corresponds to IEEE double-precision 64-bit floating point type [IEEE 754-1985]. The basic value space of `double` consists of the values *m* × *2^e*, where *m* is an integer whose absolute value is less than *2^53*, and *e* is an integer between -1075 and 970 inclusive. In addition to the basic value space described above, the value space of `double` also contains the following *special values*: positive and negative zero, positive negative infinity and not-a-number (i.e. NaN). The order-relation on `double` is: $x < y$ *if* $y - x$ is positive.

uriReference

`uriReference` data type is the one defined by the World Wide Web Consortium for the XML Schema purpose. A complete definition is available in document [R-6].

📖 Definition [R-6]: `uriReference` (normative)

> **uriReference** represents a Uniform Resource Identifier (URI) reference as defined in section 4 above [RFC 2396], as amended by [RFC 3732]. A `uriReference` can be an absolute or a relative `uriReference`, and may have an optional fragment identifier.

4.5 Metadata File Prolog

CEOS ICF Metadata File prolog is a standard XML declaration. It may contain XML declarations, comments, processing instructions and possible document type declaration. CEOS ICF does not specify any XML prolog but suggests using the simplest one:

CEOS ICF prolog (informative)

```
<?xml version="1.0"?>
```

4.6 CEOS_ICF root element

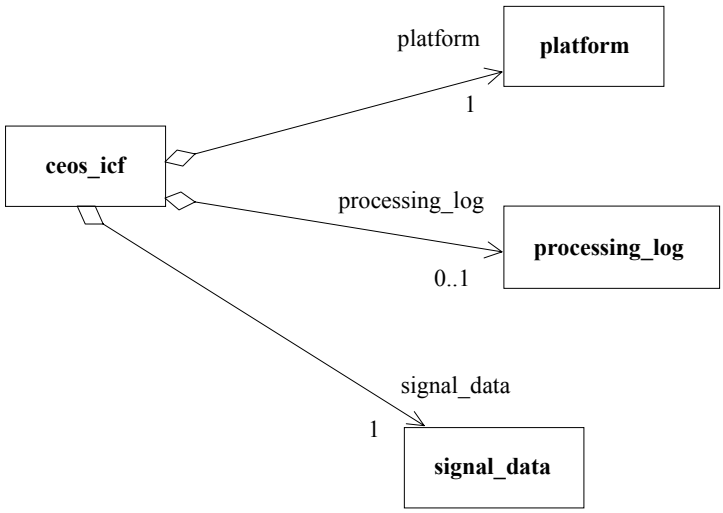


fig. 3 – Class diagram of CEOS ICF Root Element

Description

CEOS ICF Metadata File must contain one `ceos_icf` element and not more. This element is the root node where all the archive metadata are stored. A `ceos_icf` element contains metadata relative to a single Baseband Interchange Unit.

Sub-elements overview

platform	Type:	platform
	Minimum occurrence:	1
	Maximum occurrence:	1
	Description: The platform element contains the metadata relative to the platform that acquires the signal data. For instance it identifies the involved satellite, the affected sensor and some orbital elements. This element is mandatory and only one occurrence is expected. Platform has a complex type that is fully described in the section named "Platform Element".	
processing_log	Type:	processing_log
	Minimum occurrence:	0
	Maximum occurrence:	1
	Description: The processing_log element provides the operational events that have created or modified the current interchange archive. It details the facilities and systems (hardware as well as software) that have participated in the processing. Even if important for operation purposes, this element is not mandatory for perfect Signal Data File exploitation. processing_log element is of a complex type that is fully described in the section named "Processing_log Element".	
signal_data	Type:	signal_data
	Minimum occurrence:	1
	Maximum occurrence:	1
	Description: The signal_data element describes the CEOS ICF Signal Data File. For instance Signal Data File name, location, record size and record number are completely defined by this element. signal_data element is obviously mandatory and is expected only once. signal_data has a complex type that is fully described in the section named "Signal_data Element".	

XML Schema

📖 Fraction of schema for CEOS ICF root element (normative)

```
<xsd:element name="ceos_icf" minOccurs="1" maxOccurs="1">

    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="platform"
                         type="platform"
                         minOccurs="1"
                         maxOccurs="1"/>

            <xsd:element name="processing_log"
                         type="processing_log"
                         minOccurs="0"
                         maxOccurs="1"/>

            <xsd:element name="signal_data"
                         type="signal_data"
                         minOccurs="1"
                         maxOccurs="1"/>
        </xsd:sequence>
    </xsd:complexType>

</xsd:element>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

☝ Example of CEOS ICF root element (informative)

```
<?xml version="1.0"?>

<ceos_icf
    xmlns="http://wgiss.ceos.org/ceos-icf-20020423"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://wgiss.ceos.org/ceos-icf-20020423
        http://wgiss.ceos.org/ceos-icf-20020423.xsd">

    <platform>
        …
    </platform>
    <signal_data>
        …
    </signal_data>

</ceos_icf>
```

Note: the above "…" character sequences mean truncated parts

4.7 Platform Element

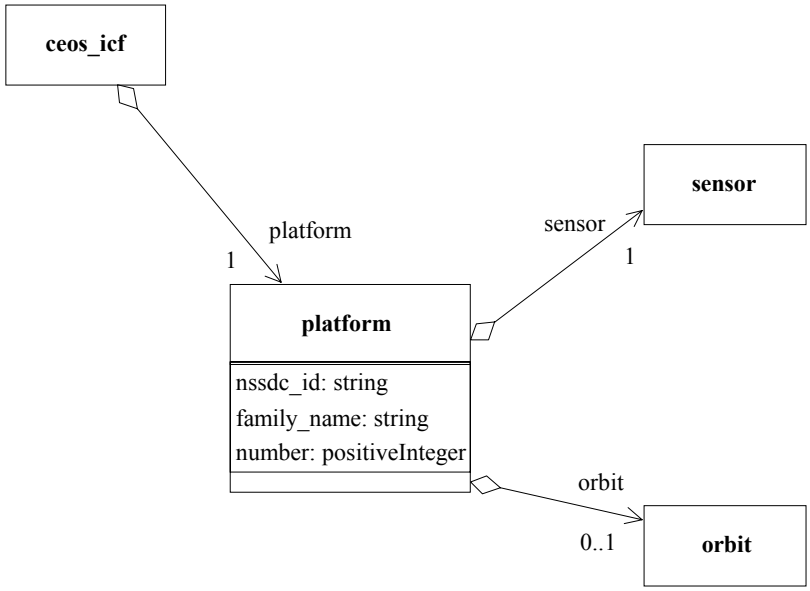


fig. 4 - Class diagram of Platform

Description

The `platform` identifies the system (satellite/aircraft) that acquired the data present in the Signal Data File. This element contains sub-elements that unequivocally identify the platform as well as those others identifying the specific sensor that acquired the data and its precise position of it in space.

Sub-elements overview

`nssdc_id`	Type:	`string`
	Minimum occurrence:	1
	Maximum occurrence :	1
	Description : The `nssdc_id` is the selected way to univocally identify a platform. The exact syntax and lexical space of this identifier is defined by the World Data Center for Satellite Information (WDC-SI), which depends on the NASA's National Space Science Data Center. For each platform, the exact identifier is available from the mission name at the address: http://nssdc.gsfc.nasa.gov/nmc/sc-query.html. This element is mandatory and cannot be repeated.	

family_name	Type:	string
	Minimum occurrence:	0
	Maximum occurrence :	1
	Description : The family_name is the mission name of the platform. This name is not a perfect identifier because it is usually not the expanded name of the mission. Therefore this element is only for printout purposes and legibility of the Metadata Data File. This element is not mandatory and cannot be repeated.	
number	Type:	positiveInteger
	Minimum occurrence :	0
	Maximum occurrence :	1
	Constraint	
	Description : The number is the sequence number of the platform in the platform family. The sequence number is already present in the nssdc_id element but is repeated here to make the Metadata File more human-readable. This element is not mandatory and cannot be repeated.	
sensor	Type	sensor
	Minimum occurrence:	1
	Maximum occurrence:	1
	Description: The sensor element identifies the sensor that acquired the data present in the Signal Data File. This element is mandatory and cannot be repeated. The precise definition of sensor type is provided in section 4.8 below.	
orbit	Type	orbit
	Minimum occurrence:	0
	Maximum occurrence:	1
	Description The orbit element provides information about the trajectory of the platform that acquired the data present in the Signal Data File. This element is not mandatory because all platforms do not provide orbital information in their telemetry stream. If include the orbit element cannot be repeated. The precise definition of orbit type is provided in section 4.9 below.	

XML Schema

📖 Fraction of schema for platform element (normative)

```
<xsd:complexType name="platform">
<xsd:sequence>

    <xsd:element name="nssdc_id" minOccurs="1" maxOccurs="1"
                 type="xsd:string"/>

    <xsd:element name="family_name" minOccurs="0" maxOccurs="1"
                 type="xsd:string"/>

    <xsd:element name="number" minOccurs="0" maxOccurs="1">
                 type="xsd:positiveInteger"/>

    <xsd:element name="sensor" minOccurs="1" maxOccurs="1"
                 type="sensor"/>

    <xsd:element name="orbit" minOccurs="0" maxOccurs="1"
                 type="orbit"/>

</xsd:sequence>
</xsd:complexType>
```

Note for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

✍ Example of Platform element (informative)

```
…
 <platform>
       <nssdc_id>1998-017A</nssdc_id>
       <family_name>SPOT</family_name>
       <number>4</number>

       <sensor>
             …
       </sensor>

       <orbit>
             …
       </orbit>
 </platform>
…
```

Note: the above "…" character sequences stand for truncated parts

4.8 Sensor Element

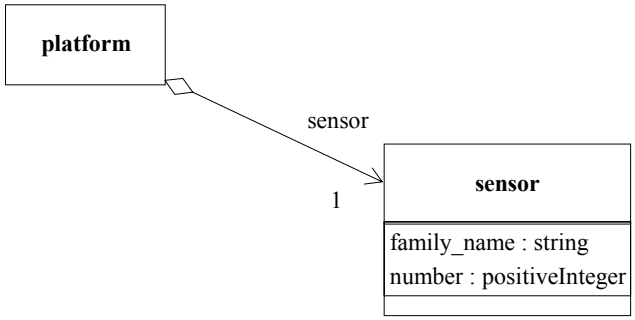


fig. 5 - Class diagram of Sensor

Description

The `sensor` identifies the instrument that acquired the data present in the Signal Data File. Conversely to the platform element, there is no absolute standard identifying or referencing the sensor. Therefore only the usual sensor family name is used to identify the instrument related to the Signal Data File.

Sub-elements overview

`family_name`	Type:	`string`
	Minimum occurrence:	1
	Maximum occurrence :	1
	Description : The `family_name` is the instrument name used to acquire the data present in the Signal Data File. This name is not a perfect identifier because it is usually not the expanded name of the sensor. However no other precise identifier has been found in the common standardization. This element is mandatory and cannot be repeated.	
`number`	Type:	`positiveInteger`
	Minimum occurrence :	0
	Maximum occurrence :	1
	Description : The `number` is the sequence number of the instrument in the platform. Both `sensor` sub-elements (i.e. `family_name, number`) are to be considered as the key to perfectly identify the sensor. This element is not mandatory and cannot be repeated.	

XML Schema

📖 Fraction of schema for sensor element (normative)

```
<xsd:complexType name="sensor">
<xsd:sequence>

    <xsd:element name="family_name" minOccurs="1" maxOccurs="1"
                 type="xsd:string"/>

    <xsd:element name="number" minOccurs="0" maxOccurs="1">
                 type="xsd:positiveInteger"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension , the documentation and comments have been deleted from the above schema

Example

☝ Example of sensor element (informative)

```
…
 <sensor>
       <family_name>HRVIR</family_name>
       <number>2</number>
 </sensor>
…
```

Note the above "…" character sequences stand for truncated parts

4.9 Orbit Element

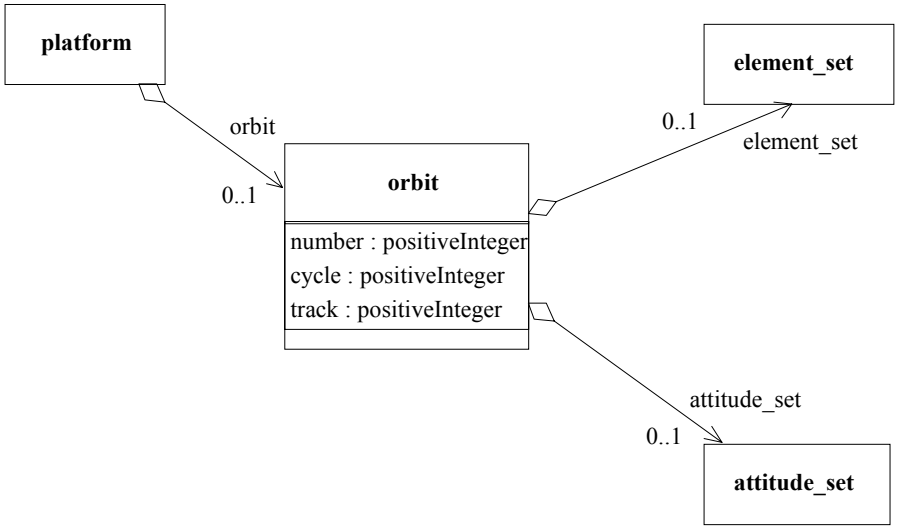


fig. 6 - Class diagram of Orbit

Description

The `orbit` element describes the trajectory of the platform. Usually the described orbit matches exactly the acquisition period of the data present in Signal Data File. However it can also describe a larger period but in any case it shall contain the acquisition period of the data present in the Signal Data File. Orbit is defined using ephemeris element sets and attitudes around the center of gravity of the platform.

Sub-elements overview

number	Type :	`positiveInteger`
	Minimum occurrence :	0
	Maximum occurrence :	1
	Description : The `number` is the orbit sequential number of the platform from its launch. Platforms do not necessarily have a periodical orbit. It is therefore not mandatory but cannot be repeated.	
cycle	Type :	`positiveInteger`
	Minimum occurrence :	0
	Maximum occurrence :	1
	Description : The `cycle` is the orbit cycle number of the platform from its launch. This element is not mandatory because all platforms do not necessarily have a cycling orbit. However, if provided it cannot be repeated.	

track	Type:	positiveInteger
	Minimum occurrence:	0
	Maximum occurrence:	1
	Description: The track is the number of the column in the World Reference System of the described platform. This element is not mandatory because all platforms do not necessarily have a reference system. However, if provided it cannot be repeated.	
element_set	Type:	element_set
	Minimum occurrence:	0
	Maximum occurrence:	1
	Description The element_set sub-element contains the ephemeris points related to the described platform and the data present within the linked Signal Data File. A precise definition of the element_set type is provided below. This element is not mandatory and cannot be repeated.	
attitude_set	Type:	attitude_set
	Minimum occurrence :	0
	Maximum occurrence :	1
	Description : The attitude_set sub-element contains the attitude angles related to the described platform and the data present is the linked Signal Data File. A precise definition of the attitude_set type is provided below. This element is not mandatory and cannot be repeated.	

XML Schema

📖 Fraction of schema for orbit element (normative)

```
<xsd:complexType name="orbit">
<xsd:sequence>
     <xsd:element name="number" minOccurs="0" maxOccurs="1">
                  type="xsd:positiveInteger"/>

     <xsd:element name="cycle" minOccurs="0" maxOccurs="1">
                  type="xsd:positiveInteger"/>

     <xsd:element name="track" minOccurs="0" maxOccurs="1">
                  type="xsd:positiveInteger"/>

     <xsd:element name="element_set" minOccurs="0" maxOccurs="1"
                  type="element_set"/>

     <xsd:element name="attitude_set" minOccurs="0" maxOccurs="1"
                  type="attitude_set"/>
</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension , the documentation and comments have been deleted from the above schema

Example

Example of Orbit element (informative)

```
…
 <orbit>
       <number>32455</number>
       <cycle>1298</cycle>
       <track>143</track>

       <element_set>
             …
       </element_set>

       <attitude_set type="angular_speed">
             …
       </attitude_set>
 </orbit>
…
```

Note: the above "…" character sequences stand for truncated parts

4.10 Element_set Element

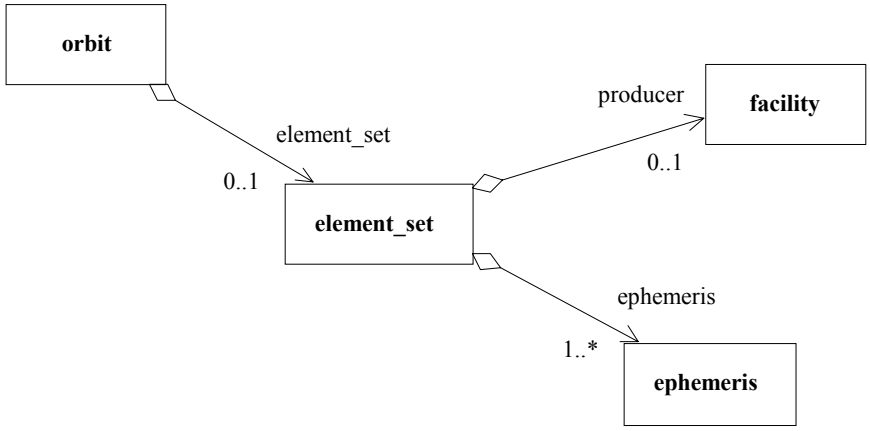


fig. 7 - Class diagram of Element_set

Description

The `element_set` is a container for the ephemeris points describing the platform trajectory. Ephemeris points may have been produced in several manners (onboard, at ground station, at mission center, etc.). Therefore the `element_set` provides a mechanism to identify the origin of the provided ephemeris points.

Sub-elements overview

producer	Type:	facility
	Minimum occurrence:	0
	Maximum occurrence :	1
	Description : The producer identifies the system (including the platform itself) or organism that provided the ephemeris points. Identifying the precise facility that has generated these ephemeris points gives an assessment about the quality of the measurements. A precise definition of the facility type is provided in the following sections. In particular this type enables identification of the involved systems (e.g. an organization, a location, as well as the software, hardware or firmware). This element is not mandatory but if provided it cannot be repeated.	
ephemeris	Type:	ephemeris
	Minimum occurrence:	1
	Maximum occurrence:	unbounded
	Description : The ephemeris element provides dynamic information (i.e. position and velocity) regarding the center of gravity of the described platform at a precise time instant. This element is mandatory and can be repeated. A complete definition of the ephemeris complex type is provided below.	

XML Schema

📖 Fraction of schema for Element_set element (normative)

```
<xsd:complexType name="element_set">
<xsd:sequence>

    <xsd:element name="producer" minOccurs="0" maxOccurs="1"
                 type="facility"/>

    <xsd:element name="ephemeris"
                 minOccurs="1"
                 maxOccurs="unbounded"
                 type="ephemeris"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension , the documentation and comments have been deleted from the above schema

Example

Example of Element_set element (informative)

```
…
  <element_set>

        <producer>
              …
        </producer>

        <ephemeris>
              …
        </ephemeris>

        <ephemeris>
              …
        </ephemeris>

  </element_set>
  …
```

Note: the above "…" character sequences stand for truncated parts

4.11 Ephemeris Element

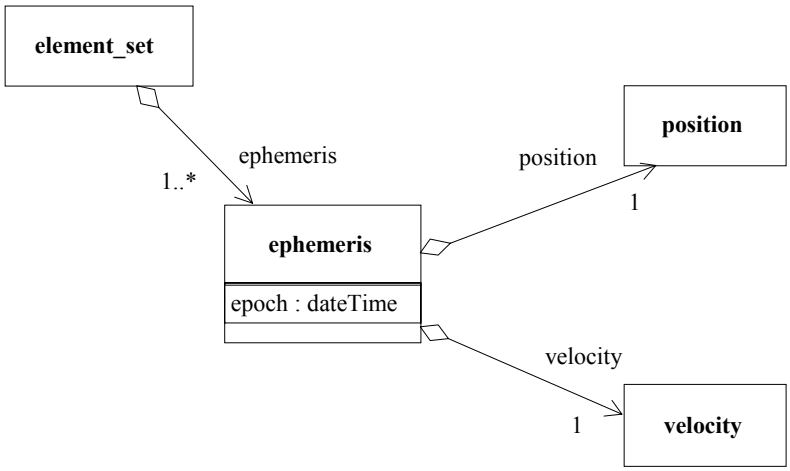


fig. 8 - Class diagram of Ephemeris

Description

The `ephemeris` element provides dynamic information (i.e. position and velocity) about the center of gravity of the described platform at a precise time instant. As described below in the sub-element section, the reference system used for the spatial and time location is the World Geodetic System 1984 [R–17] and the reference time of the `dateTime` simple type defined previously (i.e. UTC).

Sub-elements overview

epoch	Type :	`dateTime`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : The `epoch` defines the precise instant related to the dynamic information (i.e. position and velocity) provided by the current ephemeris. The time reference system is of the `dateTime` simple type (i.e. UTC). The time accuracy to be expected depends on the ephemeris `provider`. This element is mandatory and cannot be repeated.	

position	Type :	position
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : The position element provides the spatial position of the platform. The position measurement is strongly related to the epoch of the ephemeris element. The position is defined in the World Geodetic System 1984 [R-13] with the length unit system defined in section 4.3 above (i.e. metre). This element is mandatory and cannot be repeated. A complete definition of the position type is provided below. Note: WGS84 is not an inertial reference system. Positions have to be transformed if they derive from an inertial reference system. The transformation may be more or less complex according to the expected accuracy.	
velocity	Type:	velocity
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : As expected the velocity element provides the instantaneous speed of the platform. The velocity vector measurement is strongly related to the epoch of the ephemeris element. The velocity vector is defined in the World Geodetic System 1984 [R-13] with the velocity unit derived from the length and the time unit systems defined in section 4.3 (i.e. metre.s^{-1}). This element is mandatory and cannot be repeated. A complete definition of the velocity type is provided below. Note: the WGS84 reference system is not an inertial reference system. Velocities have to be transformed if they are expressed in an inertial reference system. The transformation may be more or less complex according to the expected accuracy.	

XML Schema

📖 Fraction of schema for Ephemeris element (normative)

```
<xsd:complexType name="ephemeris">
<xsd:sequence>

    <xsd:element name="epoch" minOccurs="1" maxOccurs="1"
                 type="xsd:dateTime"/>

    <xsd:element name="position" minOccurs="1" maxOccurs="1"
                 type="position"/>

    <xsd:element name="velocity" minOccurs="1" maxOccurs="1"
                 type="velocity"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

✍ Example of Element_set element (informative)

```
…
 <ephemeris>
       <epoch>1999-03-12T09:37:58.873645</epoch>
       <position>
             <x>3631079</x>
             <y>1109956</y>
             <z>6113720</z>
       </position>
       <velocity>
             <x>-0.271313</x>
             <y>-3.754622</y>
             <z>-3.754622</z>
       </velocity>
 </ephemeris>
…
```

Note: the above "…" character sequences stand for truncated parts

4.12 Attitude_set Element

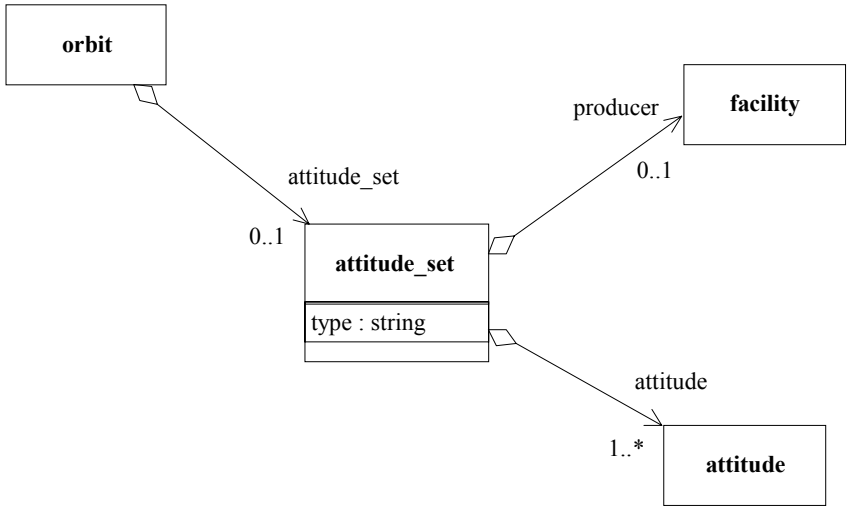


fig. 9 - Class diagram of Attitude_set

Description

The `attitude_set` is a receptacle for the attitude angles describing the platform position or movements around its center of gravity. Attitudes may have been produced or filtered in several ways (onboard, at ground station, at mission center, etc.). Therefore the `attitude_set` provides a mechanism to identify the origin of the attitudes.

Attributes overview

`type`	Type:	`string`
	Minimum occurrence:	1
	Maximum occurrence:	1
	Constraint	"`angular_velocity`" or "`absolute_rotation`"
	Description: Determines the type of the attitude measurements provided. All attitudes shall be of a homogeneous type in the overall attitude set. This attribute is mandatory and cannot be repeated.	

Sub-elements overview

producer	Type:	facility
	Minimum occurrence:	0
	Maximum occurrence :	1
	Description : The producer identifies the system (including the platform itself) or organism that provided the attitudes. Identifying the precise facility that has generated these attitudes provides an assessment of the quality of the measurements. A precise definition of the facility type is provided in the following sections. In particular this type enables identification of the involved systems (i.e. an organization, a location, as well as the software, hardware or firmware). This element is not mandatory but if provided it cannot be repeated.	
attitude	Type :	attitude
	Minimum occurrence:	1
	Maximum occurrence:	unbounded
	Description: The attitude element provides information about the rotation of the platform around its center of gravity at a precise time instant or time interval. Two types of attitudes can be provided: attitudes with absolute angles or attitudes with differential angles (i.e. angular velocity). A precise definition of attitude type is provided below. This element is mandatory and can be repeated as required.	

XML Schema

📖 Fraction of schema for Attitude_set element (normative)

```
<xsd:complexType name="attitude_set">

    <xsd:attribute name="type" minOccurs="1">
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="absolute_rotation"/>
            <xsd:enumeration value="angular_velocity"/>
        </xsd:restriction>
    </xsd:attribute>

    <xsd:sequence>

    <xsd:element name="producer" minOccurs="0" maxOccurs="1"
        type="facility"/>

    <xsd:element name="attitude" minOccurs="1"
        maxOccurs="unbounded" type="attitude"/>

    </xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

Example of attitude_set element (informative)

```
…
 <attitude_set type=”absolute_rotation”>
       <producer>
             …
       </producer>

       <attitude>
             …
       </attitude>

       <attitude>
             …
       </attitude>
 </attitude_set>
 …
```

Note: the above “…” character sequences stand for truncated parts

4.13 Attitude Element

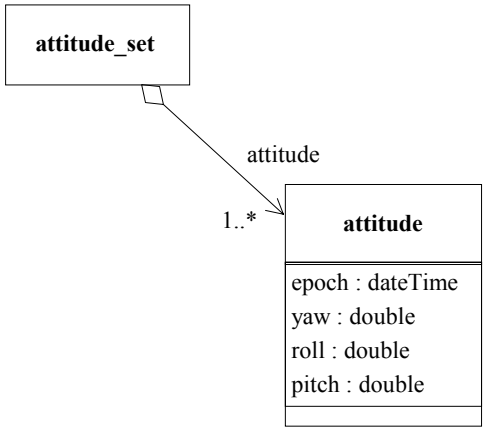


fig. 10 - Class diagram of Attitude

Description

The `attitude` element provides information about the rotation of the platform around its center of gravity at a precise time instant or time interval. Two types of attitudes can be provided: attitudes with absolute angles or attitudes with differential angles (i.e. angular velocity). The type of the provided attitude is determined by the `type` attribute of the `attitude_set` container. Being that the attitude reference system strongly depends on the platform, in particular on the inertial system used for the measurement, no default reference system is provided here. The attitude reference system to be

considered is the one specific to the described platform. Usually the attitude reference system is centered on the center of gravity of the platform and the spatial axes respectively are collinear to the position, the velocity vector. A third vector may complete the position and velocity vectors in order to form a direct referential. The direction of angles is also specific to the described platform. However the unit system is the one defined in the section 4.3 for angle quantities. The radian is therefore used for absolute angle values and radian.s^{-1} for angular velocities. The instant corresponding to the measurement is provided in the reference time system of the `dateTime` simple type defined previously (i.e. UTC).

Sub-elements overview

epoch	Type :	`dateTime`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : The `epoch` defines the precise instant related to the attitude provided by the current ephemeris. The time reference system is of the `dateTime` simple type (i.e. UTC). The time accuracy to be expected depends on the ephemeris `provider`. This element is mandatory and cannot be repeated.	
yaw	Type :	`double`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Absolute rotation or angular velocity around Yaw axis. Yaw axis is the one specific to the attitude reference system of the platform. This element is mandatory and cannot be repeated.	
roll	Type:	`double`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Absolute rotation or angular velocity around Roll axis. Roll axis is the one specific to the attitude reference system of the platform. This element is mandatory and cannot be repeated.	
pitch	Type:	`double`
	Minimum occurrence:	1
	Maximum occurrence:	1
	Description: Absolute rotation or angular velocity around Pitch axis. Pitch axis is the one specific to the attitude reference system of the platform. This element is mandatory and cannot be repeated.	

XML Schema

📖 Fraction of schema for attitude element (normative)

```
<xsd:complexType name="attitude">
<xsd:sequence>

     <xsd:element name="epoch" minOccurs="1" maxOccurs="1"
                  type="xsd:dateTime"/>

     <xsd:element name="yaw" minOccurs="1" maxOccurs="1"
                  type="xsd:double"/>

     <xsd:element name="roll" minOccurs="1" maxOccurs="1"
                  type="xsd:double"/>

     <xsd:element name="pitch" minOccurs="1" maxOccurs="1"
                  type="xsd:double"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

Example of attitude element (informative)

```
…
 <attitude>
       <epoch>1999-03-12T09:40:58.873645</epoch>
       <yaw>0.0355476</yaw>
       <roll>0.0862345</roll>
       <pitch>0.0034357</pitch>
 </attitude>
…
```

Note: the above "…" character sequences stand for truncated parts

4.14 Position Element

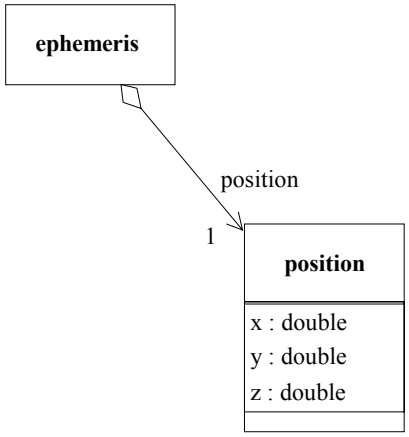


fig. 11 - Class Diagram of Position

Description

The `position` element provides the spatial location of a platform. The position vector is defined in the geocentric system described in the World Geodetic System 1984 [R-13] with the unit system derived from section 4.3 above (i.e. metre).

Sub-elements overview

x	Type :	`double`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Platform coordinate along X axis of the geocentric system defined in the World Geodetic System 1984 [R-13] expressed in the unit system derived from section 4.3 above (i.e. metre.s^{-1}). The X axis is defined by the intersection of the equator plane and the Greenwich meridian plane. X coordinates increase from earth center to the point with a 0 latitude and 0 longitude. This element is mandatory and cannot be repeated.	
y	Type :	`double`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Platform coordinate along the Y axis of the geocentric system defined in the World Geodetic System 1984 [R-13] expressed in the unit system derived from section 4.3 above (i.e. metre.s^{-1}). The Y axis derives from X and Z axis in order to make (X, Y, Z) a direct referential. This element is mandatory and cannot be repeated.	

z	Type :	double
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Platform coordinate Z axis of the geocentric system defined in the World Geodetic System 1984 [R-13] expressed in the unit system derived from section 4.3 above (i.e. metre.s^{-1}). The Z axis is the rotation axis of the earth. Z coordinates increase from the earth's center to the north pole. This element is mandatory and cannot be repeated.	

XML Schema

📖 Fraction of schema for position element (normative)

```
<xsd:complexType name="position">
<xsd:sequence>

     <xsd:element name="x" minOccurs="1" maxOccurs="1"
                  type="xsd:double"/>

     <xsd:element name="y" minOccurs="1" maxOccurs="1"
                  type="xsd:double"/>

     <xsd:element name="z" minOccurs="1" maxOccurs="1"
                  type="xsd:double"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension , the documentation and comments have been deleted from the above schema

Example

Example of position element instance (informative)

```
…
 <position>
       <x>3950385.000</x>
       <y>430304.100</y>
       <z>6001198.600</z>
 </position>
…
```

Note: the above "…" character sequences stand for truncated parts

4.15 Velocity Element

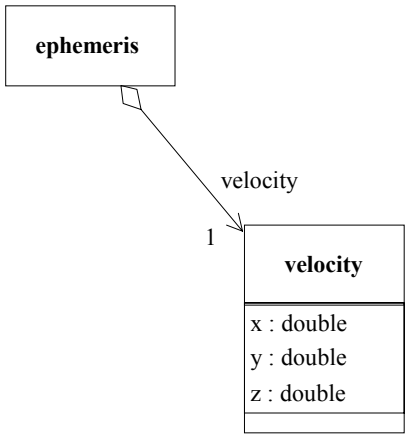


fig. 12 - Class Diagram of Velocity

Description

The `velocity` element provides the spatial motion of a platform. The velocity vector is defined in the geocentric system described in the World Geodetic System 1984 [R-13] with the speed unit system derived from section 4.3 above (i.e. metre.s^{-1}).

Sub-elements overview

x	Type :	`double`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Velocity coordinate along X axis of the geocentric system defined in the World Geodetic System 1984 [R-13] with the speed unit system derived from section 4.3 above (i.e. metre.s^{-1}). The X axis is defined by the intersection of the equator plane and the Greenwich meridian plane. X coordinates increase from earth center to the point with a 0 latitude and 0 longitude. This element is mandatory and cannot be repeated.	
y	Type :	`double`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Velocity coordinate along the Y axis of the geocentric system defined in the World Geodetic System 1984 [R-13] with the speed unit system derived from section 4.3 above (i.e. metre.s^{-1}). The Y axis derives from X and Z axis in order to make (X, Y, Z) a direct referential. This element is mandatory and cannot be repeated.	
z	Type :	`double`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Description : Velocity coordinate Z axis of the geocentric system defined in the World Geodetic System 1984 [R-13] with the speed unit system derived from section 4.3 above (i.e. metre.s^{-1}). The Z axis is the rotation axis of the earth. Z coordinates increase from the earth's center to the north pole. This element is mandatory and cannot be repeated.	

XML Schema

📖 Fraction of schema for velocity element (normative)

```
<xsd:complexType name="velocity">
<xsd:sequence>

    <xsd:element name="x" minOccurs="1" maxOccurs="1"
                 type="xsd:double"/>

    <xsd:element name="y" minOccurs="1" maxOccurs="1"
                 type="xsd:double"/>

    <xsd:element name="z" minOccurs="1" maxOccurs="1"
                 type="xsd:double"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension , the documentation and comments have been deleted from the above schema

Example

✍ Example of velocity element instance (informative)

```
…
 <velocity>
       <x>-0.271313</x>
       <y>-3.754622</y>
       <z>1.287588</z>
 </velocity>
…
```

Note: the above "…" character sequences stand for truncated parts

4.16 Processing_log Element

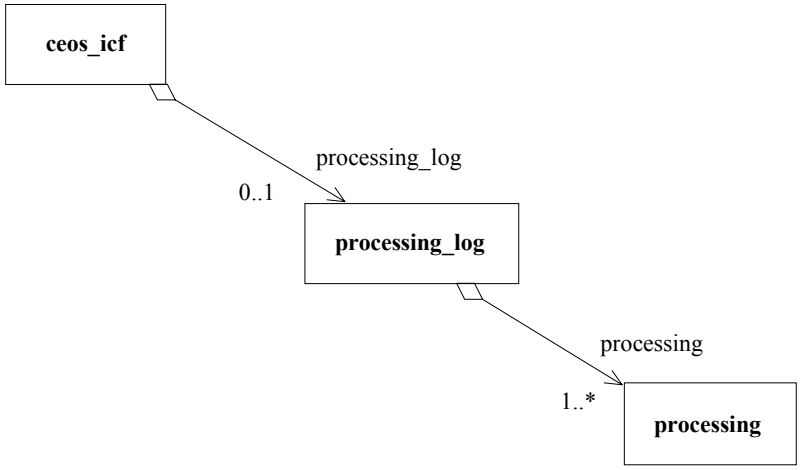


fig. 13 - Class diagram of Processing_log

Description

The `processing_log` element gathers history information enabling maintenance of the traceability of the described archive data through its main activities: receiving, pre-processing, archiving and exporting phases. Even if logged processing is dated, no assumption is made about the chronological order of the processing list.

Sub-elements overview

`processing`	Type:	`processing`
	Minimum occurrence:	1
	Maximum occurrence:	Unbounded
	Description: The `processing` elements are the logged activities applied to the present archive. At least one `processing` sub-element is expected.	

XML Schema

📖 Fraction of schema for processing_log element (normative)

```
<xsd:complexType name="processing_log">
<xsd:sequence>

    <xsd:element name="processing" minOccurs="1"
        maxOccurs="unbounded" type="processing"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

👍 Example of processing_log element (informative)

```
…
 <processing_log>
       <processing type="archiving">
             …
       </processing>
       <processing type="exporting">
             …
       </processing>
 </processing_log>
…
```

Note: the above "…" character sequences stand for truncated parts

4.17 Processing Element

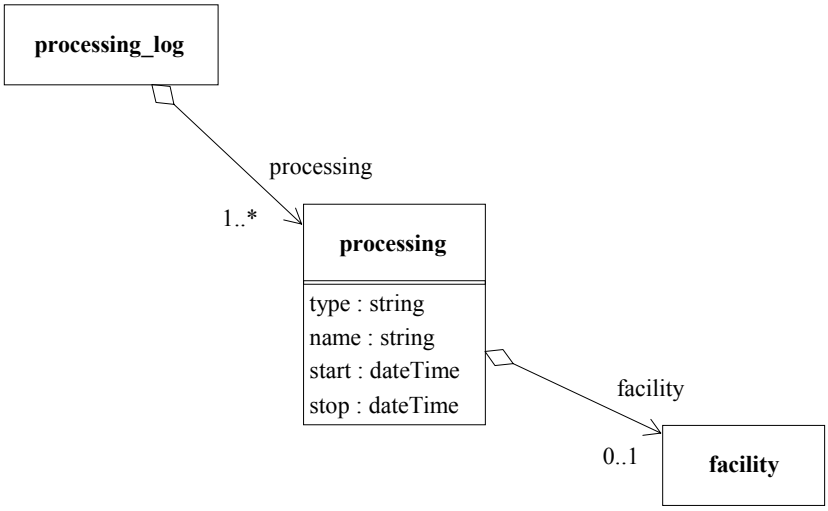


fig. 14 - Class diagram of Processing

Description

The `processing` element describes an activity applied to the present archive. The possible activities are grouped by the following categories: receiving, pre-processing, archiving and exporting phases. The `type` attribute of `processing` elements enables the identification of the related category. All processing activities are dated from their beginning to their ending.

Attributes overview

`type`	Type:	`string`
	Minimum occurrence:	1
	Maximum occurrence:	1
	Constraint	"`receiving`" or "`pre_processing`" or "`archiving`" or "`exporting`"
	Description: Determines the type of the processing provided. This attribute is mandatory and cannot be repeated.	

Sub-elements overview

`name`	Type:	`string`
	Minimum occurrence:	0

	Maximum occurrence :	1
	Description : Explicit name of the processing applied to the archive. This string is provided to make the Metadata File more legible. This element is not mandatory but if provided it will not be repeated.	
start	Type:	`dateTime`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Constraint	
	Description : Date corresponding to the beginning of the described processing. The date accuracy depends on the processing. For instance if the beginning of the described processing is entered by an operator it may have a different accuracy than if it had been written automatically by a software processor. This element is mandatory and will not be repeated.	
stop	Type:	`dateTime`
	Minimum occurrence :	0
	Maximum occurrence :	1
	Constraint	
	Description : Date corresponding to the ending of the described processing. As for the `start` element the date accuracy depends on the processing. This element is not mandatory but if provided, it will not be repeated. If `stop` date is not provided, the `start` will be considered as an overall processing date.	

XML Schema

📖 Fraction of schema for processing element (normative)

```
<xsd:complexType name="processing">

    <xsd:attribute name="type" minOccurs="1">
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="receiving"/>
            <xsd:enumeration value="pre_processing"/>
            <xsd:enumeration value="archiving"/>
            <xsd:enumeration value="exporting"/>
        </xsd:restriction>
    </xsd:attribute>

    <xsd:sequence>

        <xsd:element name="name" minOccurs="0" maxOccurs="1"
            type="xsd:string"/>

        <xsd:element name="start" minOccurs="1"
            maxOccurs="1" type="xsd:dateTime"/>

        <xsd:element name="stop" minOccurs="1"
            maxOccurs="1" type="xsd:dateTime"/>

    </xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

Example of processing element instance (informative)

```
…
 <processing type="exporting">
       <name>FRED archive -> CEOS ICF</name>
       <start>1999-03-12T09:38:12.354666</start>
       <stop>1999-03-12T09:47:52.223553</stop>
       <facility>
             <organization>ESA</organization>
             <site>FUCINO</site>
             <country>ITALY</country>
             <system type="software">
                   <name>TMFR</name>
                   <release>1.4.0</release>
             </system>
       </facility>
 </processing>
…
```

Note: the above "…" character sequences stand for truncated parts

4.18 Facility Element

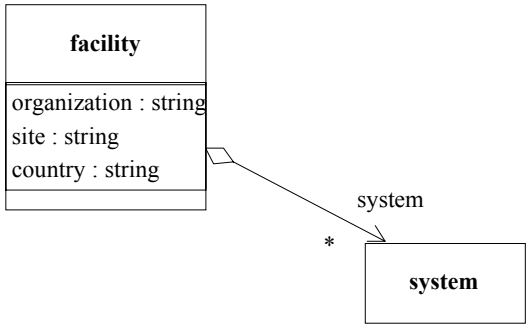


fig. 15 - Class diagram of Facility

Description

The `facility` identifies an organization as well as equipment. In particular this type allows naming the organization as well as locating it and describing the software, hardware or firmware systems used.

Sub-elements overview

`organization`	Type:	`string`
	Minimum occurrence:	0
	Maximum occurrence:	Unbounded
	Constraint	
	Description: Explicit name of the organization responsible for the facility. This name may be an agency name or a company name. This element is not mandatory and several organization names may be provided if all are responsible for the described facility. `organization` names are not restricted to a finite set of values and therefore this element cannot be considered as an identifier.	
`site`	Type:	`string`
	Minimum occurrence:	0
	Maximum occurrence:	1
	Constraint	
	Description: Geographical location of the facility. This element does not intend to describe the precise location as a postal address. However many facilities can be identified or geographical located with a usual name as Kiruna or Fucino receiving stations. This element is not mandatory but if provided, it will not be repeated. `site` names are not restricted to a finite set of values and therefore this element cannot be considered to be an identifier.	

country	Type:	string
	Minimum occurrence :	0
	Maximum occurrence :	1
	Constraint	
	Description : Country name in which the facility is located. This element is not mandatory but if provided, it will not be repeated. country names are not restricted to a finite set of values and therefore this element cannot be considered to be an identifier.	
system	Type:	system
	Minimum occurrence:	0
	Maximum occurrence:	Unbounded
	Constraint	
	Description: Describes the equipment used in the facility. Equipment can be software, hardware as well as firmware systems. system element is not mandatory and several of these can be provided. A detailed specification is provided in the section below.	

XML Schema

📖 Fraction of schema for facility element (normative)

```
<xsd:complexType name="facility">
<xsd:sequence>

     <xsd:element name="organization" minOccurs="0"
          maxOccurs="unbounded" type="xsd:string"/>

     <xsd:element name="site" minOccurs="0"
          maxOccurs="1" type="xsd:string"/>

     <xsd:element name="country" minOccurs="0"
          maxOccurs="1" type="xsd:string"/>

     <xsd:element name="system" minOccurs="0"
          maxOccurs="unbounded" type="system"/>

</xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

Example of facility element instance (informative)

```
…
 <facility>
       <organization>ESA</organization>
       <site>FUCINO</site>
       <country>ITALY</country>
       <system type="software">
             <name>TMFR</name>
             <release>1.4.0</release>
       </system>
 </facility>
…
```

Note: the above "…" character sequences stand for truncated parts

4.19 System Element

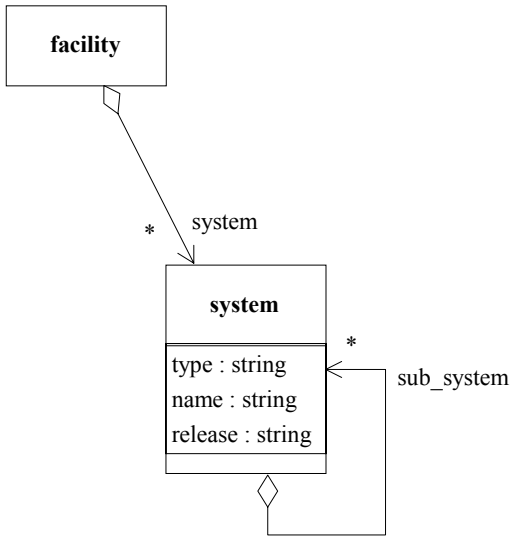


fig. 16 - Class diagram of System

Description

The `system` element type describes equipment: software, hardware or firmware systems. Each system can be broken down in several subsystems of the same or different types (see `sub_system` sub-element). `Type` attribute is used to select the type of system.

Attributes overview

type	Type:	string
	Minimum occurrence:	1
	Maximum occurrence:	1
	Constraint	"software" or "hardware" or "firmware"
	Description: Selects the type of the described system. This attribute is mandatory and cannot be repeated.	

Sub-elements overview

name	Type	string
	Minimum occurrence:	0
	Maximum occurrence :	1
	Constraint	
	Description : Explicit name of the described system. This element is mandatory and will not be repeated. System names are not restricted to a finite set of values and therefore this element cannot be considered as an identifier.	
release	Type:	string
	Minimum occurrence :	1
	Maximum occurrence :	0
	Constraint	
	Description : Version identifier of the system. Not all systems are maintained under configuration control therefore this element is not mandatory and the syntax is not restricted. This element will not be repeated several times.	
sub_system	Type:	system
	Minimum occurrence:	1
	Maximum occurrence:	Unbounded
	Constraint	
	Description: This element enables definition of equipment embedded in the present system. The definition of the sub_system element is the same as the present one. With this *system* it is possible to describe a workstation (hardware system) managed by a specific BIOS (firmware sub-system) and running a specific operating system (software sub-system). sub_system element is not mandatory but if provided, it can be repeated several times. Coherence between system and sub-system types is left to the writer who should not however define hardware as a sub-system of a software!	

XML Schema

📖 Fraction of schema for system element (normative)

```
<xsd:complexType name="system">

    <xsd:attribute name="type" minOccurs="1">
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="software"/>
            <xsd:enumeration value="hardware"/>
            <xsd:enumeration value="firmware"/>
        </xsd:restriction>
    </xsd:attribute>

    <xsd:sequence>

        <xsd:element name="name" minOccurs="1" maxOccurs="1"
            type="xsd:string"/>

        <xsd:element name="release" minOccurs="0"
            maxOccurs="1" type="xsd:string"/>

        <xsd:element name="sub_system" minOccurs="1"
            maxOccurs="unbounded" type="system"/>

    </xsd:sequence>
</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

Example of system element instance (informative)

```
…
 <system type="hardware">
       <name>Silicon Origin 2000</name>
       <sub_system type="software">
             <name>IRIX</name>
             <release>6.3</release>
       </sub_system>
       <sub_system type="hardware">
             <name>Pegasus HDDT</name>
             <release>4.45A</release>
             <sub_system type="firmware">
                   <name>Schlumberger HDDT-BIOS</name>
                   <release>2.1</release>
             </sub_system>
       </sub_system>
 </system>
…
```

Note: the above "…" character sequences stand for truncated parts

4.20 Signal_data Element

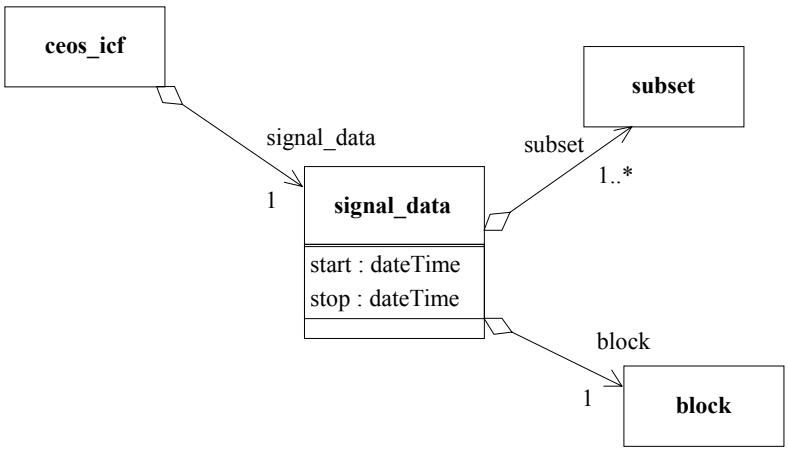


fig. 17 - Class diagram of Signal_data

Description

The `signal_data` element is one of the most important in the CEOS ICF Metadata File. It describes the physical location of the Signal Data File(s) and its internal structure. `signal_data` element also provides the start time and end time of the acquired data embedded in the Signal Data File.

According to this element the Signal Data Records (SDR) and their contents are completely defined. The SDR's are considered as an archive block that may be divided in sub-blocks. Each block describes the mapping from the transmission format structure to the one that can be found in the SDR's of the present archive. This very versatile mechanism enables the description of many archive data regardless of the acquisition platform.

Signal Data File respects however the definitions provided in the major section 5. We strongly recommend the reader as well as any other CEOS ICF users to pay special attention to section 5 for optimum comprehension of this `signal_data` element and sub-elements objectives and principles.

To overcome file system limitations (see in section 3.1 above), the Signal Data File can be partitioned in several contiguous subsets. All of them are referenced by `subset` elements.

Sub-elements overview

subset	Type:	`subset`
	Minimum occurrence:	1
	Maximum occurrence:	unbounded
	Constraint	
	Description: The Signal Data File can be partitioned in several contiguous subsets. `subset` element is mandatory and can be repeated several times in the `signal_data` element. The subsets shall be processed in their chronological order.	
start	Type:	`dateTime`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Constraint	
	Description : Beginning date of the data contained in the present archive. This date usually corresponds to the date of the first Signal Data Record embedded in the first subset. If this record contains several dates, the `start` element may be the oldest one in the chronological order. No interpolation or extrapolation is expected here. This element is mandatory and will not be repeated several times in the `signal_data` element.	
stop	Type:	`dateTime`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Constraint	
	Description : Ending date of the data contained in the present archive. This date usually corresponds to the date of the last Signal Data Record embedded in the last subset. If this record contains several dates, the `stop` element may be the most recent one in the chronological order. No interpolation or extrapolation is expected here. This element is mandatory and will not be repeated several times in the `signal_data` element.	

block	Type:	`block`
	Minimum occurrence :	1
	Maximum occurrence :	1
	Constraint	
	Description : This block element is the root element of the Signal Data Record description. It describes the precise mapping between the transmission format structure and the one that can be found in the Signal Data Record of the present archive. The complete definition of a `block` mapping type is provided in section 6. `block` element is mandatory and cannot repeated in the `signal_data` element.	

XML Schema

📖 Fraction of schema for `signal_data` element (normative)

```
<xsd:complexType name="signal_data">

    <xsd:sequence>

        <xsd:element name="subset" minOccurs="1"
            maxOccurs="unbounded" type="subset"/>

        <xsd:element name="start" minOccurs="1"
            maxOccurs="1" type="xsd:dateTime"/>

        <xsd:element name="stop" minOccurs="1"
            maxOccurs="1" type="xsd:dateTime"/>

        <xsd:element name="block" minOccurs="1"
            maxOccurs="1" type="block"/>

    </xsd:sequence>

</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

Example of `signal_data` element instance (informative)

```
…
 <signal_data>
       <subset source="ERS-1-1999-03-12T09:38:12.354666-1.dat"/>
       <subset source="ERS-1-1999-03-12T09:38:12.354666-2.dat"/>
       <start>1999-03-12T09:37:12.354666</start>
       <stop>1999-03-12T09:46:52.223553</stop>
       <block type=”composite”>
             …
       </block>
 </signal_data>
…
```

Note: the above "…" character sequences stand for truncated parts

4.21 Subset Element

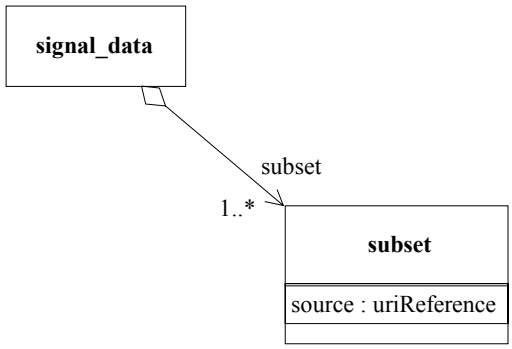


fig. 18 - Class diagram of subset

Description

The `subset` element references the partition subsets of the Signal Data File. The subset boundaries do not necessarily match those of the Signal Data Record. It is therefore not mandatory that a subset contains an integer number of Signal Data Records. It is important to keep in mind that the Signal Data File partitioning mechanism has been created to free from file system size limitation. There is therefore no particular relationship between subsets and Signal Data Records.

Files making up the subsets are referenced through the source attribute of the `subset` element. Source attribute is a uriReference and therefore enables a system of several locations, local, absolute or networked.

Attributes overview

source	Type:	uriReference
	Minimum occurrence:	1
	Maximum occurrence:	1
	Description: Referenced to the subset file composing the Signal Data File partition. This attribute is mandatory.	

XML Schema

📖 Fraction of schema for `subset` element (normative)

```
<xsd:complexType name="subset">

    <xsd:attribute name="source" minOccurs="1"
        type="xsd:uriReference"/>

</xsd:complexType>
```

Note: for optimum comprehension, the documentation and comments have been deleted from the above schema

Example

Example of `subset` element instance (informative)

```
…
 <subset source="ERS-1-1999-03-12T09:38:12.354666-1.dat"/>
…
```

Note: the above "…" character sequences stand for truncated parts

4.22 Block Element

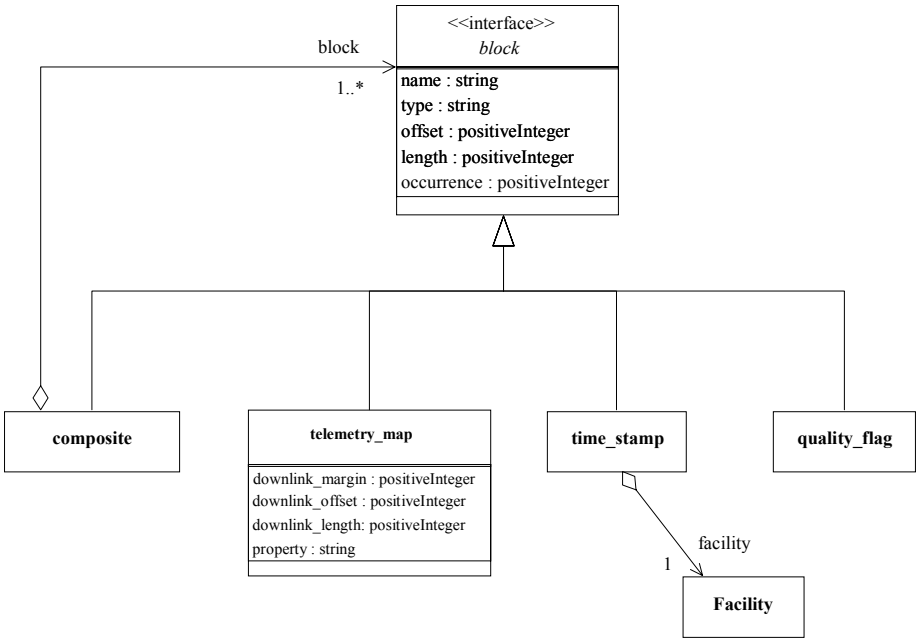


fig. 19 - Class Diagram of block

Description

The `block` element provides a physical description of the Signal Data Records of the ICF archive. These records match the smallest repeated pattern within the overall transmitted telemetry. Generally it corresponds to a frame or a major frame. It must be noticed that the smallest repeated pattern is not the smallest pattern that can be retrieved in the telemetry, which may be the byte, the format fields or the minor frame. The smallest repeated pattern is a data subset that has no higher structure definition in the telemetry. For instance the smallest repeated pattern for ERS platforms includes the frame 0 and the frames 1 to 28. For Landsat TM or ETM+ platforms it is the major frame.

In fact the Signal Data Record is a subset of this smallest repeated pattern: some parts may have been removed according to the general concepts of the CEOS ICF format (e.g. synch code); other parts extracted from the input archive may have been slightly modified to respect the Baseband concept that restricts the CEOS ICF format (e.g. frame synchronized, PN decoded, CRC code removed, byte aligned, etc.). It is also possible to add information not contained in the telemetry. For instance it is possible to tag the Signal Data Record with ground station time stamp or to flag corrupted records.

In order to reduce the Signal Data Record description to the minimum, it can be broken down into a tree of data blocks. The root block corresponds to a Signal Data Record itself. The blocks are “implemented” (i.e. UML interface implementation) into several types:

- composite block: breaks down all or part of the Signal Data Record into one or more sub-blocks;
- telemetry map block: provides a mapping from the telemetry format to the Signal Data Records of the present archive;
- time stamp block: provides time information regarding a particular facility or system;
- quality flag block: indicates if the Signal Data Record can be used or contains an error.

Complete descriptions of this block specialization are provided in the following sections.

Implementation mechanism

Because the block element is an "interface" it cannot be directly used in the Metadata. Only its "implementation" forms be used. The block implementations are defined in the Metadata File with the same name: "`block`". The implementation selection is performed through a "`type`" attribute that can be considered as an implementation switch. The use of a single `block` name has been preferred in order to simplify to the maximum the design and development of processing software.

📖 Use of block implementation (informative)

```
<block name="Signal Data Record" type="composite">
     <occurrence>117235</occurrence>

     <block name="Station time" type="time_stamp"/>

     <block type="quality_indicator">
          <length unit="bit">1</length>
     </block>

     <block type="telemetry_map">
          <length>1234</length>
          <downlink_offset>36</downlink_offset>
          <downlink_length>1234</downlink_length>
     </block>
</block>
```

Block interface

The `block` element provides the attributes and sub-elements common to all implementations. `block`'s attributes are:

name	Type:	`string`
	Minimum occurrence:	0
	Maximum occurrence :	1
	Constraint	
	Description : `Name` of the individualized block. A `block` generally corresponds to a specific field or group of fields in the telemetry format. Naming them increases the legibility aspect of the CEOS ICF Metadata File. Therefore the **`name` element will never be considered as a reference for the identification of a block in the telemetry format specification** ! This element is not mandatory but if provided, it shall not be repeated.	

type	Type:	string
	Minimum occurrence:	1
	Maximum occurrence:	1
	Constraint	"composite", "telemetry_map", "time_stamp", "quality_flag".
	Description: type attribute switches between the different implementation of blocks. This element is mandatory. *Note: the types are "Qualified names" as defined in the XML recommendation. If the CEOS ICF namespace is not the default on the Metadata File document, they shall be prefixed with the namespace tag (e.g. "icf:composite", "icf:telemetry_map", etc.). This mechanism is the same as the one used in the XML Schema built-in types (e.g. "xsd:positiveInteger", "xsd:string", etc.).*	

The block common sub-elements are:

offset	Type:	positiveInteger
	Minimum occurrence :	0
	Maximum occurrence :	1
	Constraint	
	Description : This element specifies the **byte/bit** offset of the present block in the Signal Data Record. If the present block is not the root block, the offset element refers to the parent block offset. If the parent block has several occurrences, the offset refers to the parent offset to which is added the parent length multiplied by the current occurrence index. This mechanism, even if complex to describe is very easy to implement on a computer. offset element is not mandatory and shall not be repeated in a block definition. If not provided the offset is defaulted to byte/bit immediately next to the last occurrence of previous sibling block (i.e. 0 if the current block is the first one). Because the Signal Data File generally does not contain gaps between blocks, the offset attributes are omitted. The unit of the offset value can be switched with a unit attribute that can be equal to "bit" or "byte". unit value is a "Qualified name" that shall be prefixed with the CEOS ICF namespace tag if CEOS ICF namespace is not the default one (e.g. "icf:bit", "icf:byte"). If the unit attribute is not provided it is defaulted to "byte".	

length	Type:	positiveInteger
	Minimum occurrence :	0
	Maximum occurrence :	1
	Constraint	
	Description : This element specifies the **byte/bit** length of the present block in the Signal Data Record. length element is not mandatory and shall not be repeated in a block definition. The default value depends on the implementation. For instance the default length of a composite block is the sum of the lengths (i.e. including occurrences) of the sub-blocks it contains. The unit of the length value can be switched with a unit attribute that can be equal to "bit" or "byte". unit value is a "Qualified name" that shall be prefixed with the CEOS ICF namespace tag if CEOS ICF namespace is not the default one (e.g. "icf:bit", "icf:byte"). If the unit attribute is not provided it is defaulted to "byte". *Note: the length is fixed among the occurrences of the described block.*	
occurrence	Type :	positiveInteger
	Minimum occurrence :	0
	Maximum occurrence :	1
	Constraint	
	Description : This element specifies the number of occurrences of the present described block in the Signal Data Record. This element is useful to reduce the description of the Signal Data Record in particular when the smallest repeated pattern is divided in a large number of minor frames. This element is not mandatory but shall be unique if provided. If not provided the occurrence is defaulted to 1.	

Composite block implementation

The composite block is a container of a sequence of sub-blocks. The default length of a composite block is the sum of the length of the blocks it contains. The sub-elements of a composite block are blocks:

block	Type:	block
	Minimum occurrence:	0
	Maximum occurrence:	unbounded
	Constraint	
	Description: This element enables the division of the present block in sub-blocks. These elements are not mandatory and can be provided several times. No assumptions are made about the possible overlapping or underlapping between sub-blocks.	

📖 Example of composite block (informative)

```
<block name="Signal Data Record" type="composite">
     <occurrence>117235</occurrence>

     <block type="telemetry_map">
          <length>6120</length>
          <downlink_length>6120</downlink_length>
          <downlink_offset>0</downlink_offset>
     </block>
</block>
```

In the previous example the length of the composite block (i.e. "Signal Data Record") is of 6120 bytes.

Telemetry map implementation

The "`telemetry_map`" implementation describes the mapping between a part of the Signal Data Record format and the transmission format structure (i.e. telemetry). This format is completely defined for each platform and stands as the unique reference to describe the contents of the Signal Data Record.

The example provided at the end of this section illustrates this mapping. We strongly recommend to the reader to refer to section 5 for optimum comprehension of the mapping of SDR's to the transmission format structures.

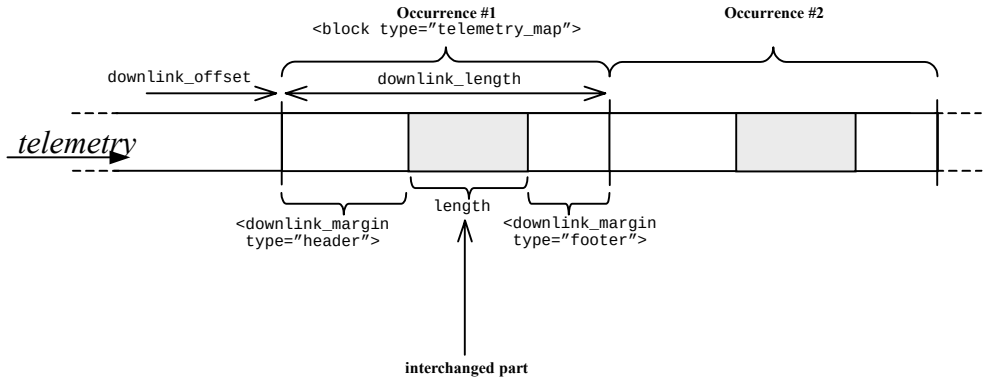


fig. 20 – Telemetry mapping

downlink_margin	Type:	positiveInteger
	Minimum occurrence:	0
	Maximum occurrence:	2
	Constraint	
	Description: This element specifies inner margins in the block that have been discarded from each occurrence of the mapped telemetry block. The margins before and after the interchanged part are respectively called "header" and "footer". Both can have different sizes expressed in bits or bytes. The margins are repeated in all the occurrences of the block, in opposition to the offset that is applied only once before the first occurrence. The type of margin can be switched using a `type` attribute than can be equal to `icf:header` or `icf:footer`. The margins are not mandatory and their values are defaulted to 0. The unit of the `downlink_margin` value can be switched with a `unit` attribute that can be equal to "`bit`" or "`byte`". `unit` value is a "Qualified name" that shall be prefixed with the CEOS ICF namespace tag if CEOS ICF namespace is not the default one (e.g. "`icf:bit`", "`icf:byte`"). If the `unit` attribute is not provided it is defaulted to "`byte`".	
downlink_offset	Type:	positiveInteger
	Minimum occurrence:	0
	Maximum occurrence:	1
	Constraint	
	Description: This element specifies the **bit/byte** offset of the present `block` in the transmission format structure. The offset is specified from the byte following the previous sibling block or from the first byte of the parent block if the current block is the first one. If the present block occurs several times the `downlink_offset` will refer to the offset of the first occurrence in the transmission format structure and will not be repeated among occurrences. The `downlink_offset` is not mandatory and is defaulted to 0 if not provided (i.e. the block therefore follows immediately its previous sibling). The unit of the `downlink_offset` value can be switched with a `unit` attribute that can be equal to "`bit`" or "`byte`". `unit` value is a "Qualified name" that shall be prefixed with the CEOS ICF namespace tag if CEOS ICF namespace is not the default one (e.g. "`icf:bit`", "`icf:byte`"). If the `unit` attribute is not provided it is defaulted to "`byte`".	

downlink_length	Type:	positiveInteger
	Minimum occurrence :	1
	Maximum occurrence :	1
	Constraint	
	Description : This element specifies the **bit/byte** length of the present block in the telemetry format. The length includes the header margin and the footer margin. The downlink_length has not default value and is therefore mandatory. However in a telemetry_map the length can be omitted: its default value is therefore equal to the downlink_length minus the margins. This rule is not always exact for instance when a decompression has been applied. The unit of the downlink_length value can be switched with a unit attribute that can be equal to "bit" or "byte". unit value is a "Qualified name" that shall be prefixed with the CEOS ICF namespace tag if CEOS ICF namespace is not the default one (e.g. "icf:bit", "icf:byte"). If the unit attribute is not provided it is defaulted to "byte".	
property	Type:	string
	Minimum occurrence :	0
	Maximum occurrence :	Unbounded
	Constraint	"compressed", "decompressed", "pn_encoded", "pn_decoded", "encrypted", "decrypted" {any other value expressed in another namespace}
	Description : The property element provides information about the transformation applied from the original telemetry. The Baseband data concept recommends applying a set of transformations (c.f. section 5 below). It may be however useful to allow departure from the Baseband data concept for very specific cases. According to the Baseband data concept the following defaults are considered: • decompressed • pn_decoded • decrypted The list of properties may be extended in future releases of this document. The existing values will however remain unchanged. Only new items may be added. property values are "Qualified names" that shall be prefixed with the CEOS ICF namespace tag if CEOS ICF namespace is not the default one (e.g. "icf:compressed", "icf:decompressed"). Thanks to this mechanism it is possible to declare transformations that are not specified within this document. For instance the particular DPCM decompression applied to Panchromatic data of SPOT may be declared: <property>cnes:dpcm_decompressed</property> where the cnes namespace tag differs from the CEOS ICF one.	

📖 Example of telemetry_map block (informative)

```
<block name="Channel 1 status" type="telemetry_map">

      <-- Description of the CEOS ICF Signal Data File -->

      <offset>2</offset>
      <length>2</length>

      <-- Corresponding subset in the telemetry -->

      <downlink_offset unit="bit">200</downlink_offset>
      <downlink_length unit="byte">25</downlink_length>

      <downlink_margin type="header" unit="byte">2
         </downlink_margin>
      <downlink_margin type="footer" unit="byte">1
         </downlink_margin>

      <-- Applied transformations -->

      <property>pn_decoded</property>
      <property>compressed</property> <-- Just for example -->

</block>
```

Time stamp implementation

The `time_stamp` block implementation allows inserting time information within a Signal Data Record. This mechanism enables the interchange of data acquired by satellites that do not provide time information in their telemetry. In such case the ground station time has to be used.

In CEOS ICF the origin of the time stamp is declared through the facility element (see section 4.18 above). The time information is expressed in MJD2000 (Modified Julian Day 2000) which is the decimal number of the day since January 1,2000 at 00:00 hours. It is encoded in binary with the following format:

MJD2000 (12 bytes)			
Desc.	Number of **days** elapsed since January 2000 00:00	Number of **seconds** elapsed since the beginning of that day	Number of **microseconds** elapsed since the last second
Type	Signed long integer	Unsigned long integer	Unsigned long integer
Length	4 bytes	4 bytes	4 bytes

The most significant bits of the long integers are written first. This format is the one used for the binary time encoding for ENVISAT, EUMETSAT MSG and METOP (EPS) level 0 (i.e. archive) formats. Because the length is fixed, the possible `length` declaration within the `block` will not be taken into account. The optional sub-element of the `time_stamp` implementation is:

facility	Type:	facility
	Minimum occurrence :	0
	Maximum occurrence :	1
	Constraint	
	Description : The facility that has generated this time stamp. This element is optional and cannot be repeated.	

📖 Example of time_stamp block (informative)

```
<block type="time_stamp">

   <offset>2</offset>

   <facility>
      <organization>ESA</organization>
      <site>FUCINO</site>
      <country>ITALY</country>
      <system type="software">
         <name>TMFR</name>
         <release>1.4.0</release>
      </system>
   </facility>
</block>
```

Quality flag implementation

The quality flag indicates whether the Signal Data Record can be used or not. As the semantic of the error can be very complex (missing frame, CRC error, PN decompression error, etc.), it is not kept in CEOS ICF Metadata File. It is however important to know that an error has been detected while or before processing this record.

It is not recommended (but possible) to declared/insert several quality indicators in the Signal Data Record. If a quality flag is declared/inserted it stands for the entire record, even if embedded in several levels of composite blocks.

The quality flag is a binary integer expressed from 1 bit to an number of bytes limited by the maximum integer size allowed by the processing hardware. It is strongly recommended to remain under the 8 bytes (64bits) and if possible under the 4 bytes that is nowadays the maximum allowed size of most of computer. The binary integer is encoded with the most significant bit first (MSB). This quality indicator has been designed purposely flexible in order to be compatible with the maximum amount of existing formats. In most cases it should not require any conversion from the telemetry or input archive format.

If the quality flag is equal to 0, the Signal Data Record is to be considered free of error (at least no error has been detected). Otherwise the record contains an error.

📖 Example of quality flag block (informative)

```
<block type="quality_flag">
     <length unit="bit">1</length>
</block>
```

XML Schema

📖 Fraction of schema for block element (normative)

```
<xsd:complexType name="block">

    <xsd:attribute name="name" minOccurs="0" type="xsd:string"/>

    <xsd:attribute name="type" minOccurs="1">
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="composite"/>
            <xsd:enumeration value="telemetry_map"/>
            <xsd:enumeration value="time_stamp"/>
            <xsd:enumeration value="quality_flag"/>
        </xsd:restriction>
    </xsd:attribute>

    <xsd:sequence>

        <xsd:element name="offset" minOccurs="0" maxOccurs="1"
                     type="xsd:positiveInteger"/>

        <xsd:element name="length" minOccurs="0" maxOccurs="1"
                     type="xsd:positiveInteger"/>


        <xsd:choice>

            <!-- Case of Composite implementation (nothing) -->

            <!-- Case of Telemetry map implementation -->

            <xsd:sequence>

                <xsd:element name="downlink_margin" minOccurs="0"
                             maxOccurs="2"
                             type="xsd:positiveInteger"/>

                <xsd:element name="downlink_offset" minOccurs="0"
                             maxOccurs="1"
                             type="xsd:positiveInteger"/>

                <xsd:element name="downlink_length" minOccurs="1"
                             maxOccurs="1"
                             type="xsd:positiveInteger"/>

                <xsd:element name="property" minOccurs="0"
                             maxOccurs="unbounded">
                     <xsd:restriction base="xsd:string>
                           <xsd:enumeration value="compressed"/>
                           <xsd:enumeration value="decompressed"/>
                           <xsd:enumeration value="pn_encoded"/>
                           <xsd:enumeration value="pn_decoded"/>
                           <xsd:enumeration value="encrypted"/>
                           <xsd:enumeration value="decrypted"/>
                     </xsd:restriction>
```

```
                </xsd:element>

            </xsd:sequence>

            <!-- Case of Time Stamp implementation -->

            <xsd:element name="facility" type="facility"
                         minOccurs="0" maxOccurs="1"/>

            <!-- Case of Quality Flag implementation (nothing) -->

        </xsd:choice>

    </xsd:sequence>

</xsd:complexType>
```

Note: for optimum comprehension , the documentation and comments have been deleted from the above schema

Example

Example of block element (informative)

```
…
  <signal_data>

        <subset source="SPOT-4-1999-03-12T09:38:12.dat"/>

        <start>1999-03-12T09:37:12.354666</start>
        <stop>1999-03-12T09:46:52.223553</stop>

        <block type="composite">

           <occurrence>6541</occurrence>

           <block name="FMT Id" type="telemetry_map">
              <length>2</length>
              <downlink_offset>17</downlink_offset>
              <downlink_length>2</downlink_length>
           </block>

           …

        </block>

  </signal_data>
…
```

Note: the above "…" character sequences stand for truncated parts

5 SIGNAL DATA FILE

5.1 Rationale

The signal data file is a binary file containing the telemetry data extracted from the input archive. The signal data file is broken down in signal data records of fixed length. Due to the fact that the CEOS ICF is dedicated to embed several telemetry, the signal data records have a dynamic layout. The precise contents of the signal data records are described in the Metadata File through the "`signal_data`" element and in particular with the "`block`" element.

5.2 Mapping of the signal data records

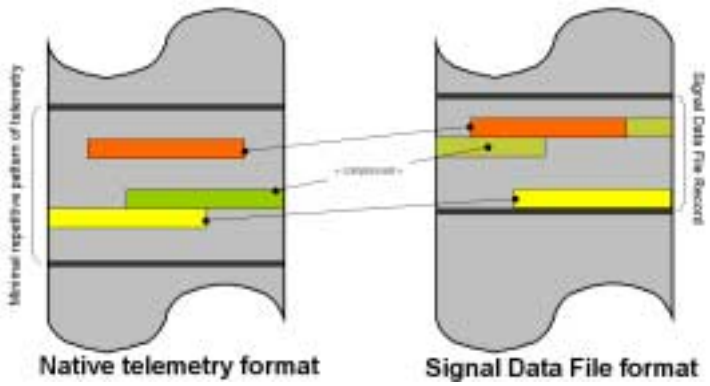
Need of dynamic layout

The signal data records of CEOS ICF have to embed telemetry data acquired from different platforms. Each of these platforms downlink the data in several formats. The scope of CEOS ICF signal data records is not to define a new generic format that is supposed to fit every telemetry format. On the contrary, CEOS ICF proposes dynamic records to which the contents are described in the Metadata File. The description is performed through a mapping mechanism between native telemetry and the current signal data records.

Mapping

The mapping mechanism breaks down the interchanged data into several blocks. Each block provides the address (e.g. position, length) in the native telemetry format and the corresponding one in the signal data records of the current CEOS ICF archive. The size and position of a block are free and will be selected according to the need of interchange.

The following diagram provides an example in which three blocks are interchanged. This diagram shows that the relative position of the block can be changed within CEOS ICF signal data records.



In this example, the sizes of the output records are independent from the input ones. In addition the output (CEOS ICF signal data records) contains unknown areas (gray areas in the signal data record) that are not described but however embedded. This functionality is useful to keep the record structure of an existing archive. For example, using this mapping mechanism it is possible to use a GERALD (SPOT archive format) signal data file as CEOS ICF signal data file with a minimum number of changes.

Example of mapping (informative)

This section is dedicated to the provision of detailed examples of `signal_data` declarations for SPOT HRVIR, ERS-2 SAR.

Example of SPOT 4 – HRVIR Signal Data File mapping (informative)

```
<signal_data>
   <subset source="SPOT-4-1999-03-12T09:38:12.354666-1999-03-12T09:47:52.223553.1.dat"/>
   <start>1999-03-12T09:37:12.354666</start>
   <stop>1999-03-12T09:46:52.223553</stop>

   <block type="composite">
      <occurrence>6541</occurrence>
      <block name="FMT Id" type="telemetry_map">
         <downlink_offset>17</downlink_offset>
         <downlink_length>2</downlink_length>
      </block>
      <block name="Channel 1 status" type="telemetry_map">
         <downlink_offset>25</downlink_offset>
         <downlink_length>2</downlink_length>
      </block>
      <block name="HRV1 mode" type="telemetry_map">
         <length>2</length>
         <downlink_offset>33</downlink_offset>
         <downlink_length>2</downlink_length>
      </block>
      <block name="Gains" type="telemetry_map">
         <downlink_offset>37</downlink_offset>
         <downlink_length>8</downlink_length>
      </block>
      <block name="Attitudes" type="telemetry_map">
         <downlink_offset>49</downlink_offset>
         <downlink_length>16</downlink_length>
      </block>
      <block name="HRV1 status" type="telemetry_map">
         <downlink_offset>193</downlink_offset>
         <downlink_length>8</downlink_length>
      </block>
      <block name="On board sequence time" type="telemetry_map">
         <downlink_offset>201</downlink_offset>
         <downlink_length>16</downlink_length>
      </block>
   </block>
</signal_data>
```

Example of ERS2 - Signal Data File mapping (informative)

```
<signal_data>
   <subset source="ERS-2-1999-03-12T09:38:12.354666-1999-03-12T09:47:52.223553.1.dat"/>
   <start>1999-03-12T09:37:12.354666</start>
   <stop>1999-03-12T09:46:52.223553</stop>

   <block name="Signal Data Record" type="composite">
       <occurrence>6541</occurrence>

       <!-- The first telemetry frame: here the offset has not been specified and
            is therefore defaulted to 0. In addition the IDHT Sync Code has been
            removed from the telemetry (i.e. downlink offset equal to 3) -->

       <block name="Frame 0" type="telemetry_map">
          <downlink_margin type=”header”>3</downlink_margin>
          <downlink_length>256</downlink_length>
       </block>

       <!-- The next telemetry frames: the length of the block is equal to 253 bytes
            because the IDHT Sync Code has been removed as for the previous Frame 0.
            The downlink footprint (i.e. downlink_length and downlink_offset) refers
            to the first occurrence of the telemetry record (i.e. Frame #1). -->

       <block name="Frame 1-28" type="telemetry_map">
          <occurrence>28</occurrence>

          <downlink_margin type=”header”>3</downlink_margin>
          <downlink_length>256</downlink_length>
       </block>
   </block>
</signal_data>
```

5.3 Signal Data Records properties

This section describes the common properties of the signal data records of a CEOS ICF Signal Data File. These properties are the default ones that are mainly inherited from the Baseband data concept. It is however possible to alter these recommended properties using the property element (see section 4.22 above).

4.5.2. No encryption

The CEOS ICF signal data records will not be encrypted.

4.5.3. CRC and Checksum

The signal data records are error free. All the error correction mechanisms will have been applied. As an example, the CRC error and checksum will have been processed to form a signal data record.

4.5.4. No compression

The signal data records will have been decompressed. For example, the imagery data of the SPOT-123-4 platforms are downlinked compressed using the DPCM compression algorithm. For such telemetry, the DPCM decompression will be processed for the CEOS ICF signal data records. Almost every archive adopts the same strategy and therefore such property is not a constraint.

4.5.5. PN decoded

The telemetry data will be PN-decoded as for almost all archive formats.

4.5.6. Data is frame synchronized

Loss of synchronization and corrupted bytes are replaced by default values but not deleted from the telemetry such as the records keep a fixed length and a sequential and continuous order. In the same manner, bit slip/bit insertion will be corrected. The "`quality_flag`" (see section 4.22 above) in the Metadata File will be used to flag the corrupted areas in the Signal Data File.

4.5.7. Chronological order

When data are time-reversed (case of playback from on-board recorders), bytes and bits will be reversed to restore a chronological acquisition.

4.5.8. Byte alignment

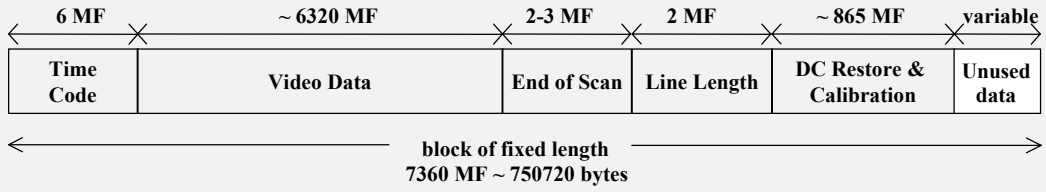
Bytes are aligned on the "`telemetry_map`" block listed in the Metadata File. However, within a "`telemetry_map`" block the original telemetry representation will be kept with the exception of changes made to comply with the other properties (e.g. decompression, PN decoding, etc.).

4.5.9. Signal data record

In order to allow a unique description of all the signal data records, the length of the records will match the smallest repeated pattern within the transmitted telemetry. In the case of variable length of this pattern (case of Landsat: variable number of minor frames in a major frame), the length of the signal data record will be oversized. Thanks to the CEOS ICF mapping system for the signal data records, the exceeding bytes will not be taken into account.

For example, the smallest repeated pattern within ERS telemetry has a length of 7424 bytes. Actually the smallest repeated pattern of ERS is composed of one OGRC HR format followed by 28 OGRC formats of data. Each of them has a fixed length of 256 bytes leading to a total size of 7424 bytes.

Example of oversized Landsat TM major frame (informative)



5.4 Partitioning Signal Data File

To cope with possible file size limitations, a partitioning mechanism has been introduced. This mechanism has been described in section 3.1. It enables to split the Signal Data File in several contiguous subsets. Each split data is called here a "subset" and represented in the metadata file by the "subset" element (see. 4.21).

The partitioning mechanism is ONLY dedicated to the physical split of the data. The Signal Data File (the BIU) is represented by the overall subsets.

It is therefore possible to rebuild the Signal Data File by concatenating the subsets from the first subset to the last one declared in the Metadata File. The way of declaring a subset is described in section 4.21.

If the Signal Data File has not been partitioned, it is composed of one subset.

APPENDIX A - XML SCHEMA (NORMATIVE)

```
<xsd:schema xmlns:xsd="http://www.w3c.org/2001/XMLSchema"
            xsd:targetNamespace="http://wgiss.ceos.org/ceos-icf-20020423">

   <xsd:complexType name="system">
      <xsd:attribute name="type" minOccurs="1">
         <xsd:restriction base="xsd:string">
            <xsd:enumeration value="software"/>
            <xsd:enumeration value="hardware"/>
            <xsd:enumeration value="firmware"/>
         </xsd:restriction>
      </xsd:attribute>
      <xsd:sequence>
         <xsd:element name="name" minOccurs="1" maxOccurs="1"
                      type="xsd:string"/>
         <xsd:element name="release" minOccurs="0" maxOccurs="1"
                      type="xsd:string"/>
         <xsd:element name="sub_system" minOccurs="1" maxOccurs="unbounded"
                      type="system"/>
      </xsd:sequence>
   </xsd:complexType>

   <xsd:complexType name="facility">
      <xsd:sequence>
         <xsd:element name="organization" minOccurs="0" maxOccurs="unbounded"
                      type="xsd:string"/>
         <xsd:element name="site" minOccurs="0" maxOccurs="1"
                      type="xsd:string"/>
         <xsd:element name="country" minOccurs="0" maxOccurs="1"
                      type="xsd:string"/>
         <xsd:element name="system" minOccurs="0" maxOccurs="unbounded"
                      type="system"/>
      </xsd:sequence>
   </xsd:complexType>

   <xsd:complexType name="processing">
      <xsd:attribute name="type" minOccurs="1">
         <xsd:restriction base="xsd:string">
            <xsd:enumeration value="receiving"/>
            <xsd:enumeration value="pre_processing"/>
            <xsd:enumeration value="archiving"/>
            <xsd:enumeration value="exporting"/>
         </xsd:restriction>
      </xsd:attribute>
      <xsd:sequence>
         <xsd:element name="name" minOccurs="0"   maxOccurs="1"
            type="xsd:string"/>
         <xsd:element name="start" minOccurs="1" maxOccurs="1"
                      type="xsd:dateTime"/>
         <xsd:element name="stop" minOccurs="1" maxOccurs="1"
                      type="xsd:dateTime"/>
      </xsd:sequence>
   </xsd:complexType>

   <xsd:complexType name="processing_log">
      <xsd:sequence>
         <xsd:element name="processing" minOccurs="1" maxOccurs="unbounded"
                      type="processing"/>
      </xsd:sequence>
   </xsd:complexType>
```

```
<xsd:complexType name="position">
   <xsd:sequence>
      <xsd:element name="x" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
      <xsd:element name="y" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
      <xsd:element name="z" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="velocity">
   <xsd:sequence>
      <xsd:element name="x" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
      <xsd:element name="y" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
      <xsd:element name="z" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="attitude">
   <xsd:sequence>
      <xsd:element name="epoch" minOccurs="1" maxOccurs="1"
                   type="xsd:dateTime"/>
      <xsd:element name="yaw" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
      <xsd:element name="roll" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
      <xsd:element name="pitch" minOccurs="1" maxOccurs="1"
                   type="xsd:double"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="attitude_set">
   <xsd:attribute name="type" minOccurs="1">
      <xsd:restriction base="xsd:string">
         <xsd:enumeration value="absolute_rotation"/>
         <xsd:enumeration value="angular_velocity"/>
      </xsd:restriction>
   </xsd:attribute>
   <xsd:sequence>
      <xsd:element name="producer" minOccurs="0" maxOccurs="1"
                   type="facility"/>
      <xsd:element name="attitude" minOccurs="0" maxOccurs="unbounded"
                   type="attitude"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ephemeris">
   <xsd:sequence>
      <xsd:element name="epoch" minOccurs="1" maxOccurs="1"
                   type="xsd:dateTime"/>
      <xsd:element name="position" minOccurs="1" maxOccurs="1"
                   type="position"/>
      <xsd:element name="velocity" minOccurs="1" maxOccurs="1"
                   type="velocity"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="element_set">
   <xsd:sequence>
      <xsd:element name="producer" minOccurs="0" maxOccurs="1"
                   type="facility"/>
      <xsd:element name="ephemeris" minOccurs="0" maxOccurs="unbounded"
                   type="ephemeris"/>
   </xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="orbit">
   <xsd:sequence>
      <xsd:element name="number" minOccurs="0" maxOccurs="1"
                   type="xsd:positiveInteger"/>
      <xsd:element name="cycle" minOccurs="0" maxOccurs="1"
                   type="xsd:positiveInteger"/>
      <xsd:element name="track" minOccurs="0" maxOccurs="1"
                   type="xsd:positiveInteger"/>
      <xsd:element name="element_set" minOccurs="0" maxOccurs="1"
                   type="element_set"/>
      <xsd:element name="attitude_set" minOccurs="0" maxOccurs="1"
                   type="attitude_set"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="sensor">
   <xsd:sequence>
      <xsd:element name="family_name" minOccurs="0" maxOccurs="1"
                   type="xsd:string"/>
      <xsd:element name="number" minOccurs="0" maxOccurs="1"
                   type="xsd:positiveInteger"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="platform">
   <xsd:sequence>
      <xsd:element name="nssdc_id" minOccurs="1" maxOccurs="1"
                   type="xsd:string"/>
      <xsd:element name="family_name" minOccurs="0" maxOccurs="1"
                   type="xsd:string"/>
      <xsd:element name="number" minOccurs="0" maxOccurs="1"
                   type="xsd:positiveInteger"/>
      <xsd:element name="sensor" minOccurs="1" maxOccurs="1"
                   type="sensor"/>
      <xsd:element name="orbit" minOccurs="0" maxOccurs="1"
                   type="orbit"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="block">
   <xsd:attribute name="name" minOccurs="0" type="xsd:string"/>
   <xsd:attribute name="type" minOccurs="1">
      <xsd:restriction base="xsd:string">
       <xsd:enumeration value="composite"/>
       <xsd:enumeration value="telemetry_map"/>
       <xsd:enumeration value="time_stamp"/>
       <xsd:enumeration value="quality_flag"/>
      </xsd:restriction>
   </xsd:attribute>

   <xsd:sequence>
    <xsd:element name="offset" minOccurs="0" maxOccurs="1"
                   type="xsd:positiveInteger"/>
    <xsd:element name="length" minOccurs="0" maxOccurs="1"
                   type="xsd:positiveInteger"/>
    <xsd:choice>
       <xsd:sequence>
          <xsd:element name="downlink_margin" minOccurs="0"
                       maxOccurs="2"
                type="xsd:positiveInteger"/>
          <xsd:element name="downlink_offset" minOccurs="0"
                       maxOccurs="1"
                type="xsd:positiveInteger"/>
          <xsd:element name="downlink_length" minOccurs="1"
                       maxOccurs="1"
                type="xsd:positiveInteger"/>
```

```
            <xsd:element name="property" minOccurs="0"
                         maxOccurs="xsd:unbounded">
                 <xsd:restriction base="xsd:string">
                    <xsd:enumeration value="compressed"/>
                    <xsd:enumeration value="decompressed"/>
                    <xsd:enumeration value="pn_encoded"/>
                    <xsd:enumeration value="pn_decoded"/>
                    <xsd:enumeration value="encrypted"/>
                    <xsd:enumeration value="decrypted"/>
                 </xsd:restriction>
            </xsd:element>
          </xsd:sequence>
          <xsd:element name="facility" minOccurs="0" maxOccurs="1"
                type="facility"/>
       </xsd:choice>
      </xsd:sequence>
   </xsd:complexType>

   <xsd:complexType name="signal_data">
      <xsd:sequence>
         <xsd:element name="subset" minOccurs="1" maxOccurs="unbounded"
                      type="subset"/>
         <xsd:element name="start" minOccurs="1" maxOccurs="1"
                      type="xsd:dateTime"/>
         <xsd:element name="stop" minOccurs="1" maxOccurs="1"
                      type="xsd:dateTime"/>
         <xsd:element name="block" minOccurs="1" maxOccurs="1"
                      type="block"/>
      </xsd:sequence>
   </xsd:complexType>

   <xsd:element name="ceos_icf" minOccurs="1" maxOccurs="1">
      <xsd:complexType>
         <xsd:sequence>
            <xsd:element name="platform"
                         type="platform"
                         minOccurs="1"
                         maxOccurs="1"/>
            <xsd:element name="processing_log"
                         type="processing_log"
                         minOccurs="0"
                         maxOccurs="1"/>
            <xsd:element name="signal_data"
                         type="signal_data"
                         minOccurs="1"
                         maxOccurs="1"/>
         </xsd:sequence>
      </xsd:complexType>
   </xsd:element>

</xsd:schema>
```

APPENDIX B - FULL METADATA FILE MODEL (INFORMATIVE)

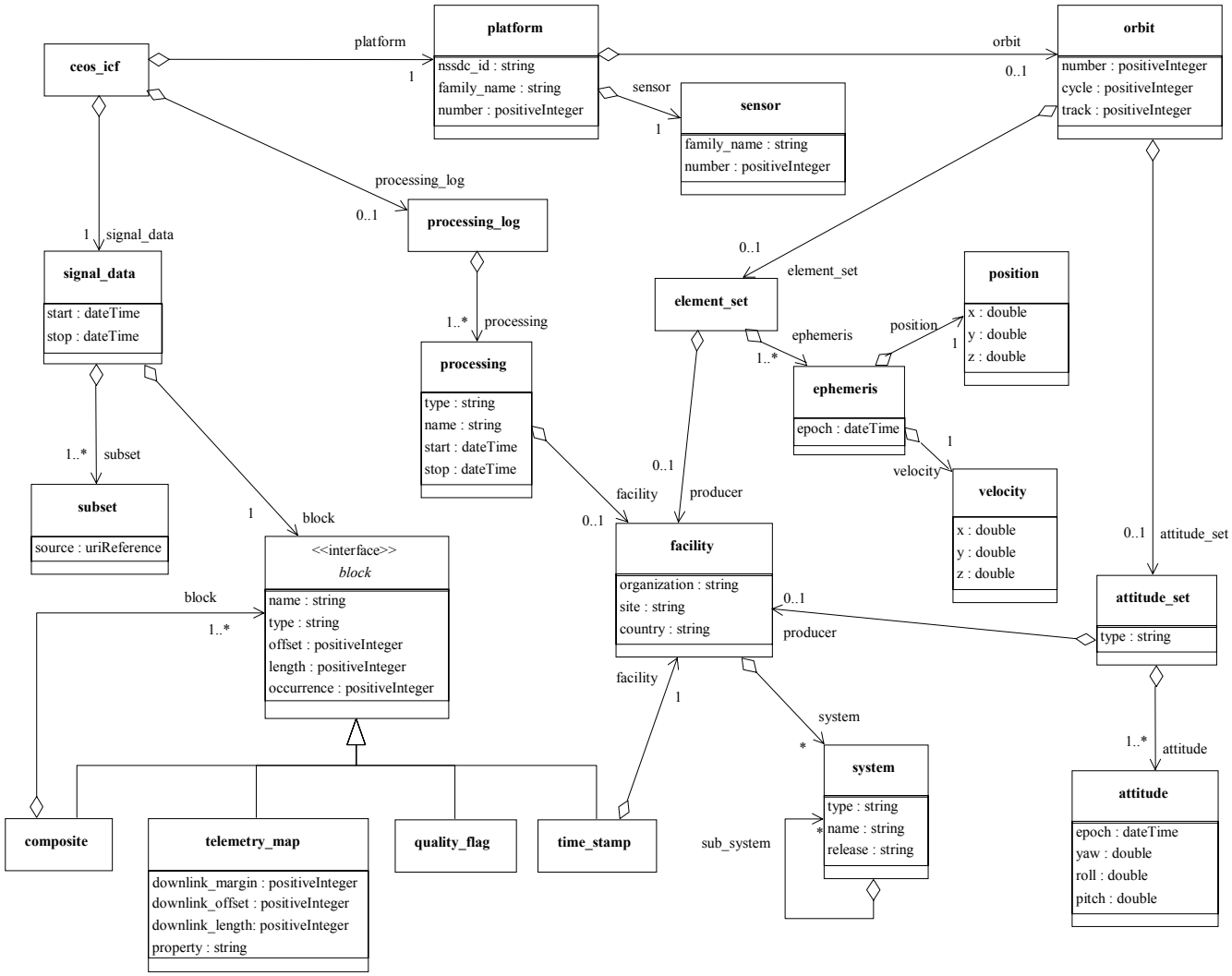


fig. 21 – Full Metadata File model

APPENDIX C - METADATA SAMPLE (INFORMATIVE)

```
<?xml version="1.0"?>

<ceos_icf xmlns="http://wgiss.ceos.org/ceos-icf-20020423">

   <platform>
      <nssdc_id>1998-017A</nssdc_id>
      <family_name>SPOT</family_name>
      <number>4</number>

      <sensor>
         <family_name>HRVIR</family_name>
         <number>2</number>
      </sensor>

      <orbit>
         <number>32455</number>
         <cycle>1298</cycle>
         <track>143</track>

         <element_set>
            <producer>
               <organization>CNES</organization>
               <site>CAP</site>
               <country>FRANCE</country>
               <system type="software">
                  <name>MADRAS</name>
                  <release>1.0</release>
               </system>
            </producer>

            <ephemeris>
               <epoch>1999-03-12T09:37:58.873645</epoch>
               <position>
                  <x>3631079</x>
                  <y>1109956</y>
                  <z>6113720</z>
               </position>
               <velocity>
                  <x>-0.271313</x>
                  <y>-3.754622</y>
                  <z>-3.754622</z>
               </velocity>
            </ephemeris>

            <ephemeris>
               <epoch>1999-03-12T09:38:58.873645</epoch>
               <position>
                  <x>3631079</x>
                  <y>1109956</y>
                  <z>6113720</z>
               </position>
               <velocity>
                  <x>-0.271313</x>
                  <y>-3.754622</y>
                  <z>-3.754622</z>
               </velocity>
            </ephemeris>
```

```
            <ephemeris>
               <epoch>1999-03-12T09:39:58.873645</epoch>
               <position>
                  <x>3631079</x>
                  <y>1109956</y>
                  <z>6113720</z>
               </position>
               <velocity>
                  <x>-0.271313</x>
                  <y>-3.754622</y>
                  <z>-3.754622</z>
               </velocity>
            </ephemeris>

            <ephemeris>
               <epoch>1999-03-12T09:40:58.873645</epoch>
               <position>
                  <x>3631079</x>
                  <y>1109956</y>
                  <z>6113720</z>
               </position>
               <velocity>
                  <x>-0.271313</x>
                  <y>-3.754622</y>
                  <z>-3.754622</z>
               </velocity>
            </ephemeris>
         </element_set>

         <attitude_set type="angular_speed">
            <producer>
               <organization>CNES</organization>
               <site>CAP</site>
               <country>FRANCE</country>
            </producer>

            <attitude>
               <epoch>1999-03-12T09:40:56.876876</epoch>
               <yaw>0.0862345</yaw>
               <roll>0.0862345</roll>
               <pitch>0.0862345</pitch>
            </attitude>
            <attitude>
               <epoch>1999-03-12T09:40:58.873645</epoch>
               <yaw>0.0862345</yaw>
               <roll>0.0862345</roll>
               <pitch>0.0862345</pitch>
            </attitude>
         </attitude_set>
      </orbit>
   </platform>

   <processing_log>
      <processing type="exporting">
         <name>Exporting from FRED archive</name>
         <start>1999-03-12T09:38:12.354666</start>
         <stop>1999-03-12T09:47:52.223553</stop>
         <facility>
            <organization>ESA</organization>
            <site>FUCINO</site>
            <country>ITALY</country>
            <system type="software">
               <name>TMFR</name>
               <release>1.4.0</release>
            </system>
         </facility>
      </processing>
   </processing_log>
```

```
   <signal_data>
      <subset source="SPOT-4-1999-03-12T09:38:12.354666-1999-03-12T09:47:52.223553.1.dat"/>
      <start>1999-03-12T09:37:12.354666</start>
      <stop>1999-03-12T09:46:52.223553</stop>

      <block type="composite">
         <occurrence>6541</occurrence>
         <block name="FMT Id" type="telemetry_map">
            <length>2</length>
            <downlink_offset>17</downlink_offset>
            <downlink_length>2</downlink_length>
         </block>
         <block name="Channel 1 status" type="telemetry_map">
            <length>2</length>
            <downlink_offset>25</downlink_offset>
            <downlink_length>2</downlink_length>
         </block>
         <block name="HRV1 mode" type="telemetry_map">
            <length>2</length>
            <downlink_offset>33</downlink_offset>
            <downlink_length>2</downlink_length>
         </block>
         <block name="Gains" type="telemetry_map">
            <length>8</length>
            <downlink_offset>37</downlink_offset>
            <downlink_length>8</downlink_length>
         </block>
         <block name="Attitudes" type="telemetry_map">
            <length>16</length>
            <downlink_offset>49</downlink_offset>
            <downlink_length>16</downlink_length>
         </block>
         <block name="HRV1 status" type="telemetry_map">
            <length>8</length>
            <downlink_offset>193</downlink_offset>
            <downlink_length>8</downlink_length>
         </block>
         <block name="On board sequence time" type="telemetry_map">
            <length>16</length>
            <downlink_offset>201</downlink_offset>
            <downlink_length>16</downlink_length>
         </block>
      </block>
   </signal_data>

</ceos_icf>
```