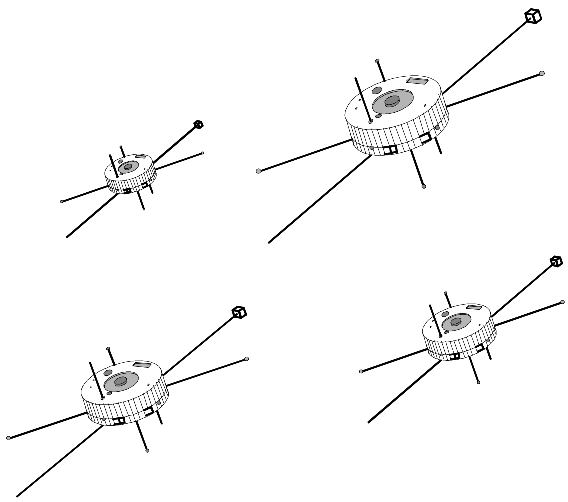
Directorate of Technical and Operational Support

European Space Operations Centre

CLUSTER Data Disposition System

Data Delivery Interface Document (DDID)



Document Reference:	CL-ESC-ID-2001
Document Status:	Issue 3
Prepared By:	ESA/ESOC
Date:	19 May 2000

Abstract:	This document defines the data delivery formats for the data delivered by ESOC to the scientific community for the CLUSTER mission. This includes the delivery formats for data available via the on-line network server and the off-line raw data media.	
Prepared by:	ESA/ESOC	
Reference:	CL-ESC-ID-2001	
Status:	Issue 3	
Approved:		
	G. Kerr (ESA Technical officer)	
	J. Dow (ESOC TOS-GFI)	P. Escoubet (Project Scientist/ESTEC)
Released:	M. Warhaut (Cluster GSM)	J. Ellwood (ESTEC/PK)
Distribution:	J. F. Kaufeler, E. Sorensen, J. Ellwood, ESTEC P. Escoubet, ESTEC M. Warhaut S. Matussi G. Kerr	J. Dow R. Corkill S. Pallaschke G. Gienger J. Wardill (cont...)

Distribution (cont):	K. Torkar IWF Institut für Weltraumforschung Inffeldgasse 12 A-8010 Graz Austria A. Balogh Imperial College London SW7 2BZ United Kingdom A. Fazakerley Mullard Space Science Lab. Holmbury St. Mary , Dorking Surrey RH5 6NT United Kingdom N.Cornilleau-Wehrlin CETP 10-12 Avenue de L'Europe F78140 Velizy France P.M.E. Decreau CNRS-LPCE 3A Av. de la Recherche Scientifique F-45071 Orleans Cedex 01 France H. Alleyne University of Sheffield, Dept. of Physics Mappin Street Sheffield S1 4DU United Kingdom H. Marquier (DGA/T/TI/PS/Tdbpi1501) CNES 18 Avenue Edouard Belin F-31401 Toulouse France M. Tatrallyay KFKI Budapest P.O. Box 49 H-1525 Budapest Hungary W. Mish NASA GSFC Greenbelt MD20771 U.S.A.	G. Paschmann MPI für Extraterrestrische Physik D-85740 Garching Germany H. Reme Centre d'Etude Spatial des Rayon. du CNRS B.P. 4346 F-31029 Toulouse France B. Wilken MPI Aeronomie D-37189 Katlenburg-Lindau Germany G. Gustafsson Swedish Institute of Space Physics S-75591 Uppsala Sweden D.A. Gurnett The University of Iowa Iowa City Iowa 52242-1479 U.S.A. Mr. C. Perry Rutherford Appleton Laboratory Chilton Didcot Oxon OX11 OQX United Kingdom A. Allner MPI für Extraterrestrische Physik D-85740 Garching Germany P.-A. Linquvist Div. of Plasma Physics S-10044 Stockholm Sweden H. Opgenoorth Swedish Inst. of Space Physics S-75591 Uppsala Sweden

Table of Contents

Sections

Tables and Figures

Document Control

Document Title:	CLUSTER Data Delivery Interface Document (DDID)		
Reference:	CL-ESC-ID-2001		
Issue	**Revision**	**Date**	**Reason for change**
0	1	10 Sept. 1991	First draft
1	0	25 August 1992	First issue reflecting internal comments received
2	0	21 January 1994	Included acronym list. Requested data limit set to 10 Mbytes per orbit. Access to system is 11 PIs plus JSOC. System architecture of on-line delivery added. VAX/VMS only used for examples, system is machine non-specific. CDDS s/w version included in acknowledgement. Experiment packets reconstituted to complete packets instead of delivery as telemetry frames, time-stamped correctly relevant to the reset pulse. Time-stamp calculation method explained. All times throughout document changed to comply with CCSDS Time-codes standard. DDS packet header shortened to remove unnecessary information: quality flags (data only delivered if good), ERT as no interest to PIs. All auxiliary data received same header as telemetry. Timeline for off-line delivery added. Figure of overall CD-ROM organisation added. The ability for the system to handle multiple versions of a CD-ROM and multiple CD-ROMs per day added. Cumulative index of CD-ROMs added to each CD-ROM as produced. All auxiliary data formats updated: short and long term orbit files (with FORTRAN access routine), short and long term event files, spacecraft attitude and spinrate file added, mission planning file deleted as not produced my MPS, telemetry mode log deleted as CDDS now reconstitutes experiment packets. Many minor edits, clarification and expansion for comprehensibility.

Document Title:	CLUSTER Data Delivery Interface Document (DDID)		
Reference:	CL-ESC-ID-2001		
Issue	**Revision**	**Date**	**Reason for change**
2	1	22 April 1994	Included R. Münch (H/OAD) as approval signature. Included J. Ellwood (ESTEC/PK) as release signature. DDS data delivery quota set to 150 Mbytes/day. The PI limit is now configurable. RDM delivery is within 3 weeks on generation of data on-board spacecraft. Support for FTAM network protocol dropped. New /AMOUNT request qualifier added. New /ZIP request qualifier added. Retransmission of files will occur a configurable number of times, in case of error. (cont.....)

			(.....cont.) The DDS is case insensitive except for the quoted target destination strings. CAB byte added to the DDS packet header. CD-ROM specification is stated as ISO 9660, level 1. CD-ROM filenaming convention changed. Support for more that 1 CD-ROM per day included. New version/reissue procedure included - only complete CD-ROMs are reissued in case of error. Housekeeping description file stream added for dynamic housekeeping description data. New CD-ROM cumulative index file format. Physical CD-ROM label defined. Section on use of data descriptions included. ADID table updated/expanded. Spacecraft platform housekeeping data added to availability list. All OAD file formats updated, data for spin phase calculation included Time calibration file format updated. Appendix on Orbit Determination and Reconstitution added. Appendix on Spin Phase and Spin Rate Reconstitution Method added. All dispositions on RIDs from review meeting on issue 2.0 included, many minor edits and clarification included.
2	2	9 May 1994	Distribution list updated. Acronym list updated. Added CD-ROM directory listing file for files changed on a reissued CD-ROM. Added comment regarding cumulative index not being updated on reissued CD-ROMs. Housekeeping Description File renamed to Housekeeping Parameter Definition File. Associated directories added to CD-ROM directory structure diagram and listing. Appendix containing CD-ROM directory and file listing updated. Several minor edits, clarification and expansion for comprehensibility. Appendix on Orbit Determination, Restitution and Prediction added.

2	3	7 June 1994	Updated distribution list to include data centres. Included reference to ZIP software and use. Method of timestamp calculation change to handle spacecraft anomaly Recall and recall/playback data stream types added. Time calibration flag added. OBT field in Time Cal. file change to 8 bytes. /SOFTWARE/ directory added to CD-ROM Minor editorial corrections for formal issue.

2	4	13 February 1995	Several minor edits, clarification and expansion for comprehensibility. Changes/corrections for ARs 1 to 50 as follows: Catalogue entry specifications updated. DDS packet header contents clarified. CD-ROM delivery during commissioning phase clarification. CD-ROM changed files list clarification. CD-ROM zero sized files clarification. CD-ROM data description files file naming convention updated. Cumulative index file format updated. HPD file clarification. ADID table; new ADID for offline catalogue entry added. ADID table; ADIDs for ASPOC and SC for NSD and BSD removed and explanation added. ADID table; "DES" corrected to "HPD". Support fax number given. CD-ROM filename list updated. LTOF and SATT formats clarified. LTEF, CMDH and TCAL formats updated. HPD file format updated and clarified. "Command history" and "CMDH" terminology made consistent. Bit and octet numbering conventions made more apparent. Spin phase definition clarified. Orbit determination compression method clarified. LTEF and STEF events lists updated. SFDU diagrams updated. Volume production information clarified. CD-ROM capacity and splitting updated/clarified. Maximum number of CDs per day set to four. Note regarding cumulative index file on reissued CD-ROMs. References made consistent. CD-ROM physical label format changed and volume label added. Note added regarding ground station field in CMDH DDS packet header corresponding to ground station field in packet data.

2	5	16 April 1996	Updated distribution list. Updated references. Clarified that a numerical IP address could be used as "Node" specification in a DDS request. Added the case "0" for "Unknown" in the "Ground Station ID" field of the DDS Packet Header format. Changed FD consolidation period from 5 to 6 days. Changed CD production baseline from 1 to 2 CDs/day. Updated the list of DDS error messages. Improved descriptive paragraphs of LTOF, LTEF and SATT sections. Added the case "**" for "Unknown" in the "Ground Station ID" field of the Command History File format.
2	6	26 January 2000	Updated distribution list Update orbit revolution numbering scheme (E.3) Inclusion of covariance matrix (E.10)
2	7	11 February 2000	Update calling sequence COVMAT (E.10)
3	0	19 May 2000	Table 5-6, Table A-1, Table E-1, Section E.3, Table E-9

1 INTRODUCTION

1.1 Purpose

This document defines the formats of all the data used in interfacing the CLUSTER Data Disposition System (CDDS) with the CLUSTER science community. The CDDS has the ability to deliver data to the Principal Investigators (PIs) via an on-line request service for quick-look purposes and will also provide an off-line prepared hard media copy of the complete data set received from the spacecraft for processing by the CLUSTER science community. These two functions require the definition of three data formats; the format of the data requests received by the CDDS from the PIs; the format of the data delivered in response to one of these requests; the format of the data files delivered off-line on the Raw Data Media (RDM). The formats and mechanisms required for PI participation in the commanding of the instruments are not covered by this document (see Ref. [7]).

The basic concepts and functionality of the CDDS are described in Ref. [1].

1.2 Structure of the Document

This document is structured as follows:

Section 2: The scope of the document.

Section 3: The data delivery requirements.

Section 4: A basic introduction to the SFDU concept.

Section 5: A detailed format description is given for each of the various types of data to be interchanged by the CLUSTER DDS. This is subdivided into three sub-sections:

(i) Format of requests received from the PIs;

(ii) Format of data sent to the PIs by the on-line system;

(iii) Format of the data sent to the PIs by the off-line system.

A number of appendices are included giving details of certain aspects of the Cluster DDS. These include:

Appendix A:	a complete list of the data available from the Cluster DDS;
Appendix E:	a complete description of all auxiliary and HPD data formats provided by ESOC;
Appendix F:	an outline of the SFDU concept. This appendix is limited to the SFDU features proposed for use in the CLUSTER DDS only; it is essential for the understanding of the data structures presented in Section 5;
Appendix G:	a definition of the bit and byte ordering conventions used for all delivered data;
Appendix H:	a description of the method used for orbit determination;
Appendix I:	a description of the spin phase and rate reconstitution method.

1.3 References

[1] "Cluster Data Distribution - A Concept Paper", Issue 2, March 1990, G. Zanoni-Aspes, ESOC/ECD/DPD/SCB/GZA/Note-05_4

[2] "Recommendation for Space Data System Standards: Standard Formatted Data Units -- Structure and Construction Rules", CCSDS 620.0-B-2, Blue Book, Issue 2, Consultative Committee for Space Data Systems, May 1992.

[3] "Report Concerning Space Data System Standards: Standard Formatted Data Units -- A Tutorial", CCSDS 621.0-G-1, Green Book, Issue 1, Consultative Committee for Space Data Systems, May 1992.

[4] "Recommendation for Space Data System Standards: Standard Formatted Data Units -- Control Authority Procedures", CCSDS 630.0-B-1, Blue Book, Issue 1, Consultative Committee for Space Data Systems, June 1993.

[5] "Recommendation for Space Data System Standards: ASCII Encoded English (CCSD0002)", CCSDS 643.0-B-1, Blue Book, Issue 1, Consultative Committee for Space Data Systems, November 1992.

[6] "Cluster Network Implementation Document", CL-ESC-ID-0002, W Dillen & R Hunter, Issue 2, 08 December 1994.

[7] "Cluster Command Request Interface Document (CRID)", CL-ESC-ID-0003, S Banfi & C Haddow, Issue 2.2, 3 July 1995.

[8] "Recommendation for Space Data System Standards: Time Code Formats", CCSDS 301.0-B-2, Blue Book, Issue 2, Consultative Committee for Space Data Systems, April 1990.

[9] "OBDH Bus Data Acquisition and Command Execution", CL-LAB-TN-0010, Issue 4A, G. Aranci/T. Aielli - Laben, 15 January 1993.

[10] Part 05 of "Experiment Interface Document (EID), Part C", CL-EST-RS-0002, Issue 1.2, 13 May 1992.

[11] "CLUSTER Experiment Interface Document (EID), Part A", CL-EST-RS-0002, Issue 2, 2 June 1993.

[12] "Recommendation for Space Data System Standards: Parameter Value Language Specification (CCSD0006)", CCSDS 641.0-B-1, Blue Book, Issue 1, Consultative Committee for Space Data Systems, May 1992.

[13]. "OAD Principles, Standards for Time and Coordinate Systems", E. M. Soop, May 1994.

[14] "Report Concerning Space Data System Standards: Parameter Value Language -- A Tutorial", CCSDS 641.0-G-1, Green Book, Issue 1, Consultative Committee for Space Data Systems, May 1992

[15] "Tools for E-mailing CSDS Documents. The ZIP and SHIP Programs", DS-MPA-TN-0002, Patrick W. Daly

1.4 Acronyms

ADID	Authority and Description Identifier
ADU	Application Data Unit
ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
ASPOC	Active Spacecraft Potential Control
BSD	Burst Science Data
CAB	Cluster Acquisition Byte
CAID	Control Authority Identifier
CCSDS	Consultative Committee for Space Data Systems
CD-ROM	Compact Disc - Read Only Memory
CDDS	Cluster Data Disposition System
CDPF	Cluster Data Processing Facility
CDS	CCSDS Day Segmented
CIS	Ion Spectrometry Experiment
CMDH	Command History File
CR	Carriage Return
CRID	Command Request Interface Document
CS	Computer Services
DAT	Digital Audio Tape
DDCS	Data Disposition Computer System
DDID	Data Delivery Interface Document
DDR	Data Description Record
DDS	Data Disposition System
DDU	Description Data Unit
DED	Data Entity Dictionary
DOY	Day of Year
DPD	Data Processing Department
DWP	Digital Wave Processing
ECD	ESA Computer Department
EDI	Electron Drift Instrument
EID	Experiment Interface Document
EDU	Exchange Data Unit
EM	Engineering Model
ERT	Earth Reception Time
EOF	End of File
ESOC	European Space Operations Centre
ESTEC	European Space Research and Technology Centre
EFW	Electric Field and Wave
FCSD	Flight Control Systems Department
FDS	Flight Dynamics System
FGM	Magnetic Field Investigation
FTP	File Transfer Protocol
GSM	Ground Segment Manager
GS	Ground Station
HKD	Housekeeping Data
HPD	Housekeeping Parameter Definition
ISO	International Organisation for Standardisation
JSOC	Joint Science Operations Centre
LSB	Least Significant Byte
LTEF	Long Term Event File
LTOF	Long Term Orbit File
LVO	Label Value Object
MCS	Mission Control System
MDS	Mission Dedicated Systems
MPS	Mission Planning System
MSB	Most Significant Byte
MSO	Most Significant Octet
N/A	Not applicable
NSD	Normal Science Data
OBDH	On-board Data Handler
OCC	Operations Control Centre
PEACE	Plasma Electron and Current Experiment
PB	Play-back
PI	Principal Investigator
PVL	Parameter Value Language
RA	Restricted ASCII
RAPID	Research with Adaptive Particle Imaging Detectors
RDM	Raw Data Media
ROM	Read Only Memory
RT	Real-time
RTSS	Real-time Systems Support
SATT	Spacecraft Attitude and Spin Rate
SC	Spacecraft
SCB	Spacecraft Control Branch
SCET	Spacecraft Event Time
SFDU	Standard Formatted Data Unit

STAFF	Spatio Temporal Analysis of Field Fluctuations
STEF	Short Term Event File
STOF	Short Term Orbit File
TBD	To be decided
TCAL	Time Calibration File
UTC	Universal Coordinated Time
VC	Virtual Channel
WBD	Wide Band Data
WEC	Wave Experiment Consortium

2 SCOPE

The aim of this document is to provide:

- Specifications of the format of the requests sent from the PIs to the CLUSTER DDS for the on-line retrieval of small quantities of quick-look data;
- Specification of the format of the requested data sent via networks from the CLUSTER DDS to the PIs. This includes the specification of:
    - The format of the request acknowledgement information;
    - The format of the relevant catalogue data;
    - The format of the requested science/auxiliary data itself;
    - The packaging of the above three elements to form the request response.
- Specification of the format of the data sent to the CLUSTER science community off-line on the Raw Data Media (RDM), i.e. CD-ROMs. This includes the specification of:
    - The technical specification of the RDM to be used;
    - The directory structure and file naming conventions to be used on the RDM;
    - The configuration control, version handling and indexing of the RDMs and each file on each RDM;
    - The format of the relevant catalogue data for each file on the RDM;
    - The format of the science/auxiliary data itself;
    - The packaging of the various elements above (i.e., catalogue, science data, data descriptions, etc.) into an SFDU structure; this shall include the packaging of all the data descriptions required to understand the data.
- Definition of the format of the full data catalogue;
- A basic understanding of the SFDU concept that is used for data packaging;

3 REQUIREMENTS

The are two separate areas of requirements for the data formats chosen. The first is for when the data is being delivered on-line to a PI, in response to a request from a PI, the second is when the data is being delivered off-line to the CLUSTER science community on an RDM. These two sets of requirements are discussed below.

3.1 On-line Data Delivery Requirements

The following requirements and assumptions are identified governing the formatting of data to be delivered on-line to the PIs:

1. Data shall be delivered to the PIs by file transfer.

2. Each transfer shall take place as a result of a PI request for data.

3. The system shall function in a client-server mode, that is it shall respond to requests, and not in an interactive mode. The CDDS shall act as the server.

4. A single request file from the PI may contain one or more request commands. A single request command may request:

    4.1 A single data stream from a single spacecraft, i.e. housekeeping or science data.

    4.2 A single data stream from a combination of 2, 3 or all 4 spacecraft.

    4.3 An auxiliary data file for a single spacecraft, i.e. orbit and attitude data, or time calibration data.

    4.4 An auxiliary data file for a combination of 2, 3 or all 4 spacecraft.

    4.5 A list of all data available via the CDDS system, i.e. a catalogue.

5. In the nominal case, the CDDS shall supply the PI with the following information, all within the same file transfer:

    5.1 An acknowledgement of his request.

    5.2 A catalogue entry, giving identification information about the data supplied; this shall include source/type identification, time period spanned by the data, etc.

    5.3 The requested data set itself.

6. In the non-nominal case, i.e. failure to satisfy the request, the items of 5.2 and 5.3 above shall not be supplied, and an error message shall be transferred in the acknowledgement (5.1).

7. The PI should receive the start of the requested data from the CDDS within 15 minutes after the CDDS receives the request. Obviously, this is subject to the data volume requested and system and network load, especially during a spacecraft pass phase.

8. Data shall be available on-line to the PIs for 10 days after acquisition from the spacecraft. This means that a request for data older than 10 days from the time that the request is submitted cannot be satisfied by the CDDS.

9. Each PI can have at most 12 requested data files queued on the CDDS at any one time. A requested data file is a file that results from a request; therefore, if a request command asks for a single data stream from all 4 spacecraft, this is deemed to result in 4 requested data files. When the PIs are submitting requests they must be aware of the number of data files that may result from any one request command, and not exceed the maximum limit, or else an error message shall be returned.

    9.1 If a single request command causes the number of requested data files to exceed the maximum number of requested data files permitted, then that single request command is rejected (**Note**: not the complete request file, just the single request command).

10. Due to performance considerations and network loading, the CDDS shall deliver, at maximum, 150 Mbytes of data in a single day (from 00:00:00 inclusive until 00:00:00 of the next day exclusive)

10.1 If a single requested data file would result in the total data delivered in a single day to exceed the maximum permitted, then the whole of that file shall not be delivered. Any further requests exceeding this limit shall be rejected by the CDDS, and a suitable error message returned.

10.2 There shall be a configurable split of the maximum data delivery quota between the various PIs and JSOC. The maximum quota per PI per day shall be decided by consultation between the PIs and the Project Scientist. The configuration shall be performed by software support at ESOC. See Ref. [6].

10.3 If a single requested data file from a PI would result in the total data delivered in a single day to exceed the maximum permitted for that PI, then the whole of that file shall not be delivered. Any further requests exceeding this limit shall be rejected by the CDDS, and a suitable error message returned.

11. The CDDS shall limit access to the data to the 11 PIs plus JSOC.

11.1 There is no restriction on which data a PI may access, i.e. all PIs may access all instrument data, plus auxiliary and catalogue data.

3.2 Off-line Data Delivery Requirements

The following requirements and assumptions are identified governing the formatting of data to be delivered off-line to the CLUSTER science community.

1. The data shall be delivered on a random access raw data media (RDM).

2. Each RDM shall contain at most a single complete days worth of data. The start of each day is defined as the first data packet time stamped at 00:00:00.00 time (midnight) inclusive or the first packet after this time, and the end of the day is defined as the latest packet time stamped before 00:00:00.00 time (midnight) of the next day.

3. If more than one RDM is needed to store a complete days worth of data, then the first RDM for that day will be filled until a predefined limit is reached, and

then a second RDM continued from that point onwards. There will therefore be a certain point in time (dictated by the predefined limit) that all the data streams will continue on the second RDM. This can of course be applied to a third or fourth RDM, there being maximum handling limit of four RDMs per day. Note that the average data reception profile will result in only a single RDM per day.

4. Only auxiliary data relevant to the same period as the science and housekeeping data shall be delivered on each RDM.

5. All data from all four spacecraft and all auxiliary data for each spacecraft shall be delivered together on RDM for each day.

6. All data on the RDM shall be accompanied by data format descriptions, these shall be contained on each RDM. This is so that each RDM is wholly self-describing. The overhead of repeating the data descriptions on each RDM is negligible given the anticipated capacity of the RDMs.

7. The data shall be delivered to the CLUSTER science community members on RDM within 3 weeks of generation on-board the spacecraft. That is, the oldest data on each RDM shall be no more than 3 weeks old.

4 INTRODUCTION TO STANDARD FORMATTED DATA UNITS

The driving requirement for the formatting of data on the CDDS is the need to handle a variety of packet sizes (e.g. the instrument data packets and the various packets of auxiliary data) and file sizes. The number of packets in the various files over a given time span can differ according to the planned operations of the corresponding instruments. For delivery of data on the RDM, the data must be delivered complete with data format descriptions so that each and every RDM is stand alone and does not require any external documentation or knowledge to understand the data. There must be a simple method of linking these data descriptions with the data that they describe. Furthermore, there must be no significant overhead imposed on the CLUSTER science community by packaging techniques or structuring of the delivered data, whether it be on-line or off-line delivery.

The particular technique proposed corresponds to an already published CCSDS Recommendation, see Ref. [2], defining structure and construction rules for the Standard Formatted Data Unit (SFDU).

The SFDU concept is applied to CLUSTERs on-line data delivery in a very simple way, so that the overhead to the PIs is minimal, and in the extreme situation, the PIs can treat the extra header fields involved as "spare" bytes.

For off-line data delivery on RDM, SFDUs are also used. They provide a method of packaging together and logically labelling (identifying) all the files on the RDM. The data files shall be accessible directly with no SFDU labelling overhead for those CLUSTER science community members who wish to access some particular data and already know exactly where that data is on the RDM. However, the SFDU packaging used for the data on the RDM shall allow science community members with no previous knowledge of the file structure being used to find and retrieve data and its relevant description.

An awareness of the SFDU concept shall help each science community member to build a more resilient data handling system and provide better visibility of data structures, particularly for common data and if needed, interchange of data between science community members. The SFDU concept also provides a technical and administrative framework for handling and exchanging of data descriptions.

Appendix F gives a brief description of the SFDU concept and the data packaging techniques used sufficiently enough to permit understanding of the SFDU implementation proposed in this document. This appendix must first be understood so that the data structures presented here can be followed. The interested reader is recommended to read Refs. [2] and [3] for a full description and tutorial on the SFDU concept.

5 CLUSTER DATA DELIVERY FORMATS

The two methods of data delivery for the CLUSTER mission (on-line and off-line) are quite independent of each other, and as such the data formats involved shall be described here separately. The data can be considered as coming from two separate systems. The first system is the on-line delivery system which involves the PIs submitting requests via file transfers to the CDDS and in return the CDDS delivering the requested data back to the PI. The second system is the off-line delivery system which involves their Operations Control Centre (OCC), which for this mission is the European Space Operations Centre (ESOC), delivering to each appropriate CLUSTER science community member the complete CLUSTER data set on RDM. **This system is totally independent from any interaction with the PIs**.

5.1 On-line Data Delivery

A simplified architecture of the Cluster Data Processing Facility (CDPF) is shown below in 5-1 Whilst the details of how ESOC receives, processes and handles the data is not of a direct interest to the PIs, and understanding of the overall procedure is deemed useful. The CDDS receives instrument data directly from the ground stations, whilst the auxiliary data comes separately from both the Mission Control System (MCS) and Flight Dynamics System (FDS). The PIs have no access to these latter two systems as they are protected by network bridges. The PIs can send files to a particular account on the CDDS machine in which the request and transfer of data as described below is coordinated. The on-line data delivery system works as a server/client model, with no interactive interface for the PIs, just the exchange of request files and resultant data transfers.

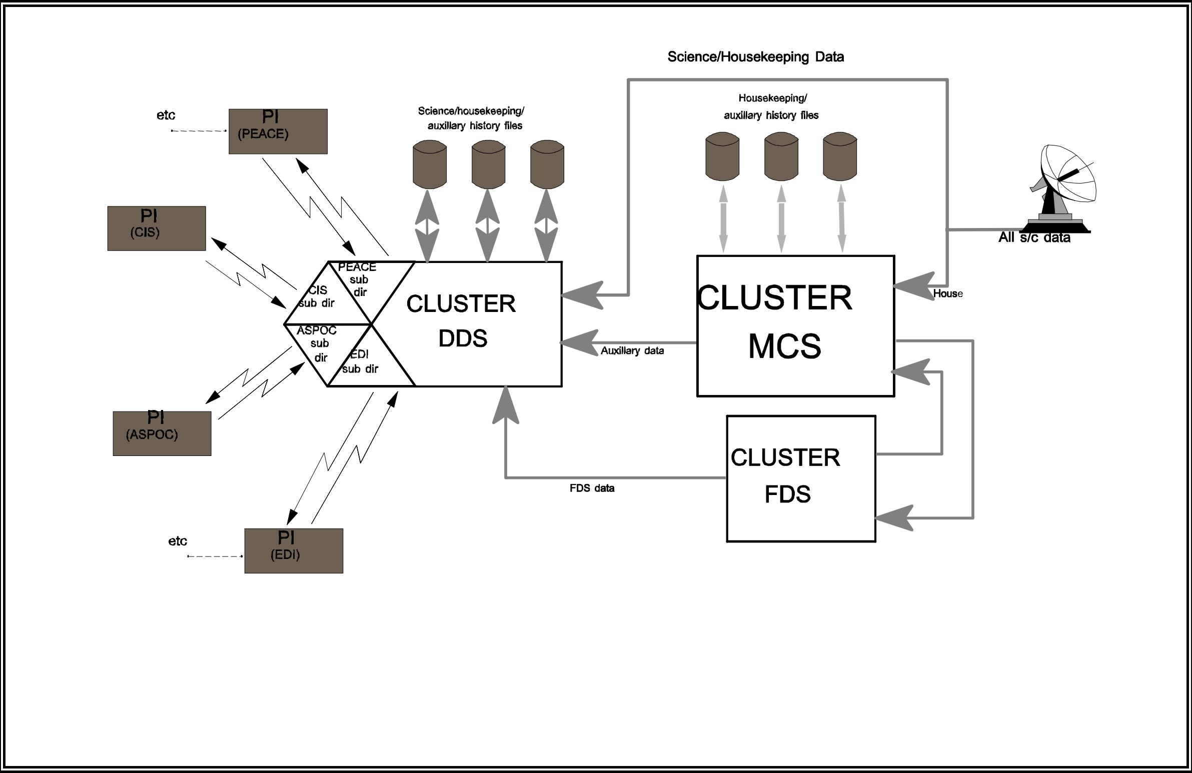


Figure 5.1 : Architecture of the Cluster Data Processing Facility

5.1.1 Communication

The CLUSTER DDS and the PIs shall communicate by the use of inter-computer file transfers. Each PI shall be assigned a separate directory on the CDDS machine, this shall be the only directory that any one PI has access to.

A request for data shall always be initiated by the PI. The PI shall copy a request file from their own network node to the assigned CDDS network node/directory. The CDDS shall process these requests in turn, and if valid, shall then copy the data sets requested to the destination specified in the request.

5.1.2 Requests from the PIs

5.1.2.1 Request Communication

This document covers in detail only the **format** of the data transferred between the PIs and the CDDS and visa-versa, and not the details of the communication protocols to be used (these are detailed in Ref. [6]). Therefore this section shall only outline the basic procedure for communication between the PIs to the CDDS. The examples given are based on a PI with a VAX/VMS machine communicating with the CDDS via DECnet. The CDDS shall support PIs communicating with the CDDS via **DECnet** and **FTP**. Even though all examples shown here use the VAX/VMS DECnet file transfer conventions, the procedure is very similar for FTP and so shall not be repeated.

For each data set sent from the CDDS to a PI, there must first be a request command from the PI. This request is received by the CDDS in the form of a request file copied into the PI specific directory allocated by ESOC on the host CDDS machine. For the full list of PI directories and privileges on the CDDS system see Appendix C.

The file transferred shall be formatted in ASCII, with each request command on a separate line. **A single request command shall not span more than one line**. A line is deemed to terminate at the first carriage return, line feed and/or end of record.

To deposit a data request file in the appropriate directory for a specific PI, the following command would be issued on the PIs machine when utilising the DECnet protocol:

$COPY PIRequestFile CDDSNode::[CDDSDirectory]CDDSRequestFile

Where:

PIRequestFile	**Specifies the filename of the request file on the PI machine**
CDDSNode	**Specifies the DECnet network node of the CDDS**
CDDSDirectory	**Specifies the PI directory on the CDDS machine**
CDDSRequestFile	**Specifies the file destination name on the CDDS computer of the request file.** This file shall always be named REQUEST.DDS

For this example, each of these parameters are in standard VAX/VMS format.

Note: For the CDDS to guarantee correct processing of the request file when receiving a file via the DECnet protocol, a file version number **must not** be specified as part of the REQUEST.DDS **filename.**

5.1.2.2 Request File Format

This section defines the request command that the PI shall put in the request file, REQUEST.DDS, that is transferred to the CDDS for processing. The PI has only one command available:

REQUEST

There may be a maximum of 12 REQUEST commands in a single REQUEST.DDS file. Each REQUEST command must be on a separate line in the request file. As well as a maximum of 12 REQUEST commands per request file, there is a further restriction that the request cannot result in a cumulation of more than 12 file transfers awaiting despatch to any one PI at a time.

The exact syntax of the REQUEST command is as follows:

```
REQUEST  [Qualifiers]  DataIdentifier  TargetSpec

Qualifiers:         /SC = 1
                    /SC = 2
                    /SC = 3
                    /SC = 4
                    /SC = ALL
                    /SINCE = TimeSpec
                    /BEFORE = TimeSpec
                    /AMOUNT = NumBytes
                    /ZIP
```

Figure 5-2 Format of the REQUEST Command

DataIdentifier:

The DataIdentifier specifies the unique data source that the PI is requesting. It has the following format:

DataSource.DataType

The punctuation mark is required to separate the two fields, these fields are:

DataSource	One of the data source mnemonics defined in Appendix A Typically a data source is a particular instrument, or auxiliary data source.
DataType	One of the data type mnemonics, also defined in Appendix A This is the sub-category of the data source.

Obviously not all combinations of DataSource and DataType are valid. An invalid combination shall be treated as a request error and reported back as such to the PI.

TargetSpec:

The TargetSpec specifies the name of the file on the PI machine which the requested data set shall be transferred to. The maximum length for the TargetSpec is 80 characters. For a PI destination that is using the DECnet protocol, a TargetSpec has the following format (the example below is valid for VAX/VMS):

NODE::DEVICE:[DIRECTORY]FILENAME.TYPE

The punctuation characters are required to separate the fields of the TargetSpec. The fields follow standard VAX VMS format as follows:

NODE	Network node name
DEVICE	Device name
DIRECTORY	Directory name
FILENAME	Filename
TYPE	File type

Note that when the VAX/VMS/DECnet protocol is being used for file transfers, to guarantee correct processing and delivery of data, there must be no version number specified in the TargetSpec.

For PI destinations that are not using DECnet on VAX/VMS machines, the TargetSpec must be enclosed in double quotes (i.e., ") and be of the following format:

"NODE:FileSpecification"

Where:

NODE	is the network node name. It can be expressed either as numerical IP address or host name. Numerical IP address is the preferred option.
FileSpecification	is a valid pathname/filename for the file on the specified node. For example, on a UNIX machine this might be /dir1/dir2/myfile.dat.

Note: For non VAX/VMS PI destination machines there must be only one colon (:) specified between the NODE and FileSpecification.

The name specified by the TargetSpec is not exactly the filename that will be used for transfers back to the PIs. As it is possible to request more than one spacecraft data with only one REQUEST command, the specified TargetSpec will be appended with the spacecraft number, either 01, 02, 03 or 04. For example, if the TargetSpec is:

EDINODE::[DIR1.DIR2]EDIDATA.NSD

and all four spacecraft data are requested (see below for request details), then there will be four files returned to the PI, with the following names:

EDINODE::[DIR1.DIR2]EDIDATA.NSD01
EDINODE::[DIR1.DIR2]EDIDATA.NSD02
EDINODE::[DIR1.DIR2]EDIDATA.NSD03
EDINODE::[DIR1.DIR2]EDIDATA.NSD04

The PIs must take this into account when they are specifying a TargetSpec. Note, the spacecraft number will be appended even if only one spacecraft data source is requested.

Since the CDDS has to support various communication protocols and PI machines, it shall not carry out any syntax checks on the supplied TargetSpec. Each PI will have to supply a default TargetSpec before the start of the mission, so that if there is an error or problem transferring a file to the destination specified in a request, an error message can be returned to the default destination. In the case of a transmission error, the CDDS will try and re-transmit the file a number of times (configurable by CDDS support at ESOC). If a PI does not receive either a reply or an error message or a default target reply within 2 hours, then they must assume that the request has failed totally and resubmit the request.

The REQUEST command and all qualifiers are case insensitive. If the protocol for communication being used is DECnet, then the filenames will automatically be converted to uppercase. If FTP is being used then the case of the TargetSpec specified in the quoted string of the request will be preserved and used "as is" for the returned filename.

Qualifiers:

/SC=1, /SC=2, /SC=3, /SC=4 and /SC=ALL, specify from which spacecraft the data type/source is being requested from. Any combination of these qualifiers may be used. The default, if none of these parameters are specified, is that all four spacecraft data will be requested (equivalent to specifying /SC=ALL). The PIs should note that even though only one REQUEST command is required to request up to four spacecraft data, this results in four file transfers from the CDDS to the PI and hence counts as 4 out of the maximum 12 permitted outstanding requests. If one REQUEST command should exceed the maximum 12 limit then all the resulting files from that one request command are rejected.

/SINCE=TimeSpec specifies that only data time stamped later than the specified TimeSpec shall be transferred. In the case where no /SINCE qualifier is specified, then the transfer shall start with the oldest data available for that particular data stream currently held on the CDDS.

The TimeSpec follows the CCSDS Time Codes Format Standard (Ref. [8]). The ASCII Time Code A format is used, this has the following format:

YYYY-MM-DDThh:mm:ssZ

Where each character is an ASCII character, which have the following meanings:

YYYY	The year with possible values 0001 - 9999
MM	The month with possible values 01 - 12
DD	The day of the month with possible values 01 - 28, 29 , 30 or 31 depending upon the month
T	Date-Time separator, always the ASCII character "T"
hh	The hour with possible values 00 - 23

mm	The minute with possible values 00 - 59
ss	The second with possible values 00 - 59
Z	A mandatory terminator, always the ASCII character "Z"

Note: The hyphens (-), colons (:), letter "T" and letter "Z" are used as specific sub-field separators. All sub-fields must contain leading zeros as necessary.

If a date is specified, but no time (e.g. 1996-03-12Z), then the time is assumed to be 00:00:00.

If a time is specified, but no date, then (e.g. 17:23:15Z), the date is assumed to be the date at the time the request is processed.

The intervening Date-Time separator (T) is required if both the date and time are specified, and omitted if either date or time also omitted.

If any trailing sub-fields are omitted from the date specification, then those particular sub-fields are assumed to equal the current date values when the request is processed.

If any trailing sub-fields are omitted from the time specification then they are assumed to be zero.

It is not permitted to omit leading sub-fields of the date or time specification.

The finest resolution that may be specified is the integer second, i.e., fractions of seconds may not be specified.

Confusion can occur especially when the time span requested crosses a year boundary, therefore, **it is strongly recommended that the full date and time format always be specified**.

/BEFORE=TimeSpec specifies that only data time stamped earlier than the particular TimeSpec shall be transferred. In the case where no /BEFORE qualifier is specified, then the data transfer shall end with the most recent data currently available for that particular data stream held on the CDDS at the time of processing of the request. The TimeSpec has the same specification as for when used in the /SINCE qualifier detailed above.

If, for a single request, the PI gives a /BEFORE qualifier that specifies a time that is earlier than that specified in a /SINCE qualifier, then a suitable error message shall be returned to the PI.

/AMOUNT=NumBytes specifies the maximum quantity of data that the PI wishes to be delivered in any single file. Note that this means if the /AMOUNT qualifier is used in a REQUEST command in conjunction with a qualifier(s) that specifies more than one spacecraft, then the quantity specified applies to each file for each spacecraft and not for the total for all spacecraft. The rational for having the /AMOUNT qualifier is so that a PI can more carefully control his permitted quota for on-line data delivery.

The NumBytes parameter is the maximum number of bytes specified in bytes (note, NOT Kbytes or Mbytes). If this number would result in the maximum quota for the day for that PI being exceeded, then an error is returned immediately.

In the situation where a /BEFORE and /AMOUNT qualifiers are both specified, then the data delivered will be up to the point of whichever qualifier is reached first. For example, if the /BEFORE value was 14:00:00, the /AMOUNT qualifier equalled 30000 and the experiment packet timestamped 12:37:45 made the resultant file delivered 30255 bytes, then the delivery would stop at that point and would not continue until the last packet before 14:00:00. Note that the limits on daily data quantities apply to the data before it has been optionally 'ZIPped' for transmission over the network (see next paragraph).

/ZIP specifies that the file to be delivered should be compressed prior to transmission over the network, using the commonly available 'ZIP' algorithm. Note, that the complete resultant file including acknowledgement, catalogue and data will compressed as a single file. Packages for 'UNZIPing' the received compressed file are readily available for all common hardware/software platforms, including UNIX, VMS, DOS and Mac (see Ref. [15], these packages shall not be supplied by ESOC).
If a file has been compressed before transmission to the PI, then the TargetSpec filename shall be appended with a "Z" character (after the appended spacecraft number).

It should be noted that the throughput benefits of using the ZIP option cannot be accurately assessed at present, as the degree of compression depends greatly upon the content of the data being compressed. Realistic science data is not currently available for testing.

The maximum length of the REQUEST command line is 1024 characters. Within these limits the REQUEST command string can contain any number of spaces and tabs between the various parameters. Between each REQUEST command in a single request file, there may be any number of white space characters (space, tab, carriage return, line feed). Any line that starts with an exclamation mark (!) in the first column shall be treated as a comment, and the remainder of that line ignored by the request processor. For example, the following is a valid request for data:

```
! This is an example data request file, that would be sent
! to ESOC from the EDI PI.

! Check on status of instruments on all 4 spacecraft today

REQUEST /SINCE=18:00Z /BEFORE=19:00Z EDI.HKD ENO::[DIR]4CRAFT.DT

! Get a sample of normal and burst science data from spacecraft 1

REQUEST /SINCE=1996-08-12Z /BEFORE=1996-08-13Z /SC=1 EDI.NSD
ENO::[DIR]NM.DT
REQUEST /SINCE=08:00:00Z /BEFORE=08:05:00Z /SC=1 /AMOUNT=10000
EDI.BSD ENO::[DIR]BT.DT
```

Figure 5-3 Example Request File from a PI to ESOC

From this request file the CDDS would transfer in return:

- Four files of housekeeping data, one from each spacecraft covering the period of 18:00 to 19:00 as of todays date, for the EDI instrument, as follows:

  ENO::[DIR]4CRAFT.DT01
  ENO::[DIR]4CRAFT.DT02
  ENO::[DIR]4CRAFT.DT03
  ENO::[DIR]4CRAFT.DT04

- One file of normal science data for spacecraft 1, covering the whole of the day of 12 August 1996, for the EDI instrument, as follows:

ENO::[DIR]NM.DT01

- One file of burst science data for spacecraft 1, covering five minutes from 08:00 to 08:05 as of today, for the EDI instrument, with a maximum file size of 10,000 bytes, as follows:

ENO::[DIR]BT.DT01

It should be noted that the CDDS will have to have write access to accounts on the PIs machine, and that the passwords for these accounts will be sent across the network as part of the transfer process. If DECnet is selected, then proxy accounts can be used which do not necessitate the transfer of passwords, but if FTP is selected then the transmission of passwords is unavoidable. The details of the communication processes and the security implications are addressed in Ref. [6].

5.1.3 Response from the CDDS

It is important to realise that the basic data delivery structure follows from the requirements stated in Section 3 and is not dependent upon any particular packaging technique. It follows from these requirements that each response to a successful request for data shall be in three parts within a single file (except when an error has occurred, see Section 5.1.3.4)

1. The **acknowledgement** of the request;

2. The **catalogue entry** for the data being requested;

3. The **requested data** itself.

Figure 5-4 (overleaf) shows an SFDU which reflects this structure, the first two parts of the reply (acknowledgement and catalogue information) are each in a separate Label-Value-Object (LVO), and the requested data is in a third LVO. The schematic of Figure 5-4 is the response to a request for all types of data available from the CDDS (i.e., either instrument data packets, housekeeping, or auxiliary data). It is built using the simplest SFDU packaging technique, Envelope Packaging. The envelope is an "outer" LVO with Class ID = Z (meaning the VALUE field can contain LVOs with any Class Identifier); and an Authority and Description Identifier (ADID) = CCSD0001 (meaning envelope packaging). This LVO contains the three components mentioned above, namely:

1. **<u>An LVO with Class ID = V</u>** carrying an Acknowledgement which contains a detailed breakdown of the original request and the response status;

2. **<u>An LVO with Class ID = K</u>** containing a catalogue entry for the requested data;

3. **<u>An LVO with Class ID = I</u>** containing the requested data set itself.

All the LVOs shall be delimited by ASCII length (i.e., LVOs with Version Identifier = 3 and Delimitation Identifier = A in their LABEL). The lengths of each of the LVOs shall not be fixed as it shall depend upon the volume of data transferred, which in turn depends upon what the PI requests. The actual length to be inserted in the LVO LABELs shall only be known after the VALUE fields have been generated by the system.

The ADIDs shown in Figure 5-4 are illustrative, and the Control Authority based at ESOC may assign different values from these examples.

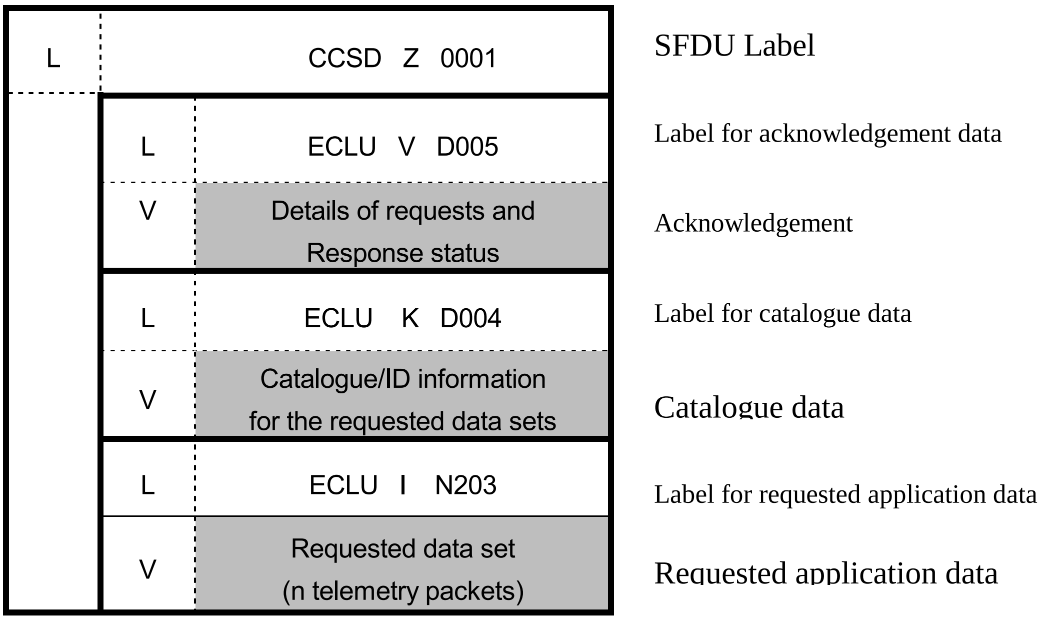


Figure 5-4: Schematic of On-line Data Delivery Format

5.1.3.1 The Acknowledgement

The acknowledgement constitutes the VALUE field of the Class V LVO. The VALUE field shall be expressed in Parameter Value Language (PVL, see Refs. [12] and [14]). This is a simple language where parameter names are assigned values, for example "SPACECRAFT_NAME=CLUSTER_1;" would assign the string "CLUSTER_1" to the parameter name "SPACECRAFT_NAME". The advantage of using PVL over a fixed position data format is that the values are intuitively understandable and self documenting to a human reader as they are written in pseudo-English. PVL also allows any amount of white space (space, carriage return, line-feed, tab, vertical tab) between any of the language elements, along with 'C' style comments, therefore the VALUE field can be laid out clearly and is easily human or machine interpretable. Finally, if there is a need to add any further information or parameters then this simply involves adding a new PVL statement to the VALUE field. Due to the flexible layout allowed by PVL this does not invalidate previous data**.**

The information that shall be contained in the acknowledgement are as follows:

- Details of the original file transfer request, including: qualifiers specified; full expansion and specification of the time span specified (or the assumed times if not specified); the filename of the request file(1); the data source and type specified (including spacecraft name), for auxiliary data pseudo data sources are defined (see Appendix A the full target file specification;
- Date and time at which the request was processed by the CDDS;
- Production information indicating the environment that this SFDU was created upon, including: computer, operating system and its version, SFDU generation software version, distribution method, etc;
- An error message string (this shall contain the string "NO ERRORS**"** for nominal processing with no errors).

<u>Note</u>: Each PI shall supply a default file specification to ESOC so that, if there is an error in the target file specification supplied in the request file, the CDDS can still inform the PI of the error by using the default file specification as the target file.

Specifically, the parameter names and possible values that are allowed in the

(1) For the CDDS the request filename should always be REQUEST.DDS**, but the filename is included in the acknowledgement so as to provide more robust error handling and for future expansion into a generic system where the request handler may deal with other types of files, possibly for other missions.**

acknowledgement are as follows:Table 5-1: Acknowledgement Specification

Parameter	Possible values
SPACECRAFT_NAME	CLUSTER_1, CLUSTER_2, CLUSTER_3, CLUSTER_4
DATA_SOURCE	See Appendix A
DATA_TYPE	See Appendix A
REQUEST_FILENAME	Quoted string
SINCE_TIME_SPECIFIED	Quoted /SINCE **string from command line**
SINCE_TIME_EXPANDED	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**)**
BEFORE_TIME_SPECIFIED	Quoted /BEFORE **string from command line**
BEFORE_TIME_EXPANDED	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**)**
MAXIMUM_BYTES_SPECIFIED	Integer numeric from command line or UNLIMITED **if not specified**
BYTES_DELIVERED	Integer numeric of actual delivered science data (note, ONLY Class I Value field size)
COMPRESSION	NONE **or** ZIP**, indicating compression used or not.**
TARGET_FILENAME_SPECIFIED	Quoted string
TARGET_FILENAME_EXPANDED	Quoted string including spacecraft number appended
TIME_PROCESSED	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**)**
COMPUTER	Quoted string, e.g. "DEC VAXStation 4000/90"
OPERATING_SYSTEM	Quoted string, e.g. "VAX VMS 5.5.2"
CDDS_SOFTWARE_VERSION	Quoted string, e.g. "2.1"
DISTRIBUTION_METHOD	FTP, DECNET
ERROR_MESSAGE	Quoted string

For example if the request file contained the request command as shown below in:

```
This is another example data request file, that would be sent
! to ESOC from the EDI PI.

REQUEST /SINCE=1995-08-25Z /BEFORE=19:00Z /SC=1 EDI.NSD
EDINODE::[MYDIR]MYFILE.DAT
```

Figure 5-5 Example of a Request File from a PI to ESOC

Then the acknowledgement that would be contained in the VALUE field of the Class V LVO transferred back to the PI would be as shown in Figure 5-6 (this is assuming that the time of processing is 1st September 1995 at 22:13:45):

```
/*****************************************************
****  Acknowledgment returned from the CLUSTER DDS ****
*****************************************************/

SPACECRAFT_NAME = CLUSTER_1 ;
DATA_SOURCE = EDI ;
DATA_TYPE = NSD ;

REQUEST_FILENAME = REQUEST.DDS ;

SINCE_TIME_SPECIFIED = "1995-08-25Z" ;
SINCE_TIME_EXPANDED = 1995-08-25T00:00:00Z ;

BEFORE_TIME_SPECIFIED = "19:00Z" ;
BEFORE_TIME_EXPANDED = 1995-09-01T19:00:00Z ;

MAXIMUM_BYTES_SPECIFIED = UNLIMITED ;
BYTES_DELIVERED = 8944 ;
COMPRESSION = NONE ;

TARGET_FILENAME_SPECIFIED = "EDINODE::[MYDIR]MYFILE.DAT" ;
TARGET_FILENAME_EXPANDED = "EDINODE::[MYDIR]MYFILE.DAT01" ;

TIME_PROCESSED = 1995-09-01T22:13:45Z ;

COMPUTER = "DEC VAXStation 4000/90" ;
OPERATING_SYSTEM = "VAX VMS 5.5.2" ;
CDDS_SOFTWARE_VERSION = "2.1";
DISTRIBUTION_METHOD = NETWORK;

ERROR_MESSAGE = "NO ERROR" ;
```

Figure 5-6 Example Acknowledgement Response

As can be seen, this format of acknowledgement is very clear to understand for the human reader and also very easy for a PVL parser to extract the relevant values for each of the parameter names.

5.1.3.2 The Catalogue Entry

The catalogue entry is the VALUE field of the Class K LVO. This contains information specific to the actual data transferred; obviously this can be different to the requested data depending upon what time stamped telemetry frames are available etc. The entry uses the same PVL language as for the acknowledgement, and contains the following pieces of information:

- The name of the spacecraft from which the data was received;
- The data source and data type; for auxiliary data pseudo data sources are defined (see appendix A);
- The ADID of the data description for the particular packets being transferred;
- The time stamp of the earliest packet in the requested data set;
- The time stamp of the latest packet in the requested data set;
- The number of packets that are being transferred in the requested data set.

Specifically the parameter names and possible values that are allowed in the acknowledgement are as shown in Table 5-2

Parameter	Possible values
SPACECRAFT_NAME	CLUSTER_1, CLUSTER_2 CLUSTER_3, CLUSTER_4
DATA_SOURCE	See Appendix A
DATA_TYPE	See Appendix A
ADID	Eight ASCII characters, e.g, ECLU0009**, see Appendix A**
EARLIEST_PACKET	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**)**
LATEST_PACKET	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**) Set to the same value as** EARLIEST_PACKET **if there is only one packet.**
NUMBER_OF_PACKETS	ASCII encoded decimal number, no leading zeros

Table 5-2 Catalogue Entry Specification

The PVL statements are grouped in a catalogue entry using the PVL aggregation construct:

```
BEGIN_OBJECT = CATALOGUE_ENTRY;
        catalogue information
END_OBJECT = CATALOGUE_ENTRY;
```

This is so that many catalogue entries can be supplied in the same LVO value field when a Master Catalogue is requested (see later).

For example, taking the previous request example the corresponding catalogue entry for the requested data could be (depending upon data availability):

```
/*************************************************
****  Catalogue entry from the CLUSTER DDS  *****
*************************************************/

BEGIN_OBJECT = CATALOGUE_ENTRY;

   SPACECRAFT_NAME = CLUSTER_1 ;
   DATA_SOURCE = EDI ;
   DATA_TYPE = NSD ;

   ADID = ECLU0009 ;

   EARLIEST_PACKET = 1995-08-25T01:12:32.23Z ;
   LATEST_PACKET = 1995-08-29T23:11:04:48Z ;

   NUMBER_OF_PACKETS = 1202 ;

END_OBJECT = CATALOGUE_ENTRY;
```

Figure 5-7: Example Catalogue Entry

5.1.3.3 The Requested Data

In each case the requested data set constitutes the VALUE field of the Class I LVO. A PI can request data from three sources: telemetry data; auxiliary data; catalogue data. Data from all three sources are requested in the same manner, i.e., with the REQUEST **command.**

5.1.3.3.1 Telemetry Data

Requested telemetry data shall appear as a set of unsegmented experiment or housekeeping packets with a given ADID and arranged in time ascending order. The ADID specified in the label of the delivered Class I LVO shall point to the data description of one complete packet for the particular data source/type. The number of these packets within the Class I LVO value field is indicated by the "NUMBER_OF_PACKETS**"** parameter in the catalogue entry (the accompanying Class

K LVO) for this transfer of data.

The experiment packets on-board the spacecraft are split into separate fixed length blocks and formatted into transfer frames for downlink to the ground and transfer to ESOC. These separate frames are then reconstituted back into complete experiment packets by the CDDS at ESOC and the complete experiment packets are delivered to the PIs, both for the on-line and off-line delivery formats. Depending upon whether the spacecraft is in normal data mode or burst data mode there will be either 10 or 62 frames respectively that make up one experiment packet, i.e., all the data between reset pulses.

The experiment packets will be timestamped with the Spacecraft Event Time (SCET). The SCET is the calibrated time of generation of the first bit in the associated packet. The time at which science data is obtained onboard is that of the preceding VC0 frame (housekeeping data); the first frame in a science packet sequence is indicated by the identification bit in the CAB byte being set, as indicated in Ref. [9]. Therefore, to obtain the correct timestamp for the experiment packet the following method will be used:

For the nominal case, i.e. a VC0 frame has been received which has a timestamp that satisfies the following:

$$T_{SC(1)} - T_{reset} < T_{VC0} < T_{SC(1)}$$

then the timestamp given to the science packet shall be;

$$T_{SCET} = T_{VC0}$$

where:

$\mathbf{T_{SC(1)}}$	is the timestamp of the first science (VC2/VC3) frame of the packet (i.e. the VC2/VC3 frame with the CAB bit set).
$\mathbf{T_{reset}}$	is the time interval of the reset pulse cycle (5.15222168 seconds as specified in Ref. [9]).
$\mathbf{T_{VC0}}$	is the time of the VC0 frame
$\mathbf{T_{SCET}}$	is the timestamp given to the science packet.

For the case where no such VC0 frame has been received (e.g. the frame was lost), then the time is extrapolated by using the VC0 frame which has the time closest to the missing VC0 frame as follows:

1. The time of the latest available VC0 frame with a timestamp earlier than $\mathbf{T_{SC(1)}}$ shall be obtained, this time is designated $\mathbf{T_{VC0-}}$ and the frame $\mathbf{F_{VC0-}}$

2. The time of the earliest available VC0 frame with a timestamp greater than $\mathbf{T_{SC(1)}}$ shall be obtained, this time is designated $\mathbf{T_{VC0+}}$ and the frame $\mathbf{F_{VC0+}}$

The following algorithm shall then be applied:

IF (frame F_{VC0-}) AND (frame F_{VC0+}) NOT available THEN

$\mathbf{T_{SCET} = T_{SC(1)}}$ *This is worst case contingency*

ELSE

IF ((frame $\mathbf{F_{VC0+}}$ NOT available) OR (($\mathbf{T_{SC(1)} - T_{VC0-}}$) ≤ ($\mathbf{T_{VC0+} - T_{SC(1)}}$))) THEN

$$T_{SCET} = T_{VC0-} + T_{reset} * INT [(T_{SC(1)} - T_{VC0-}) / T_{reset}]$$

ELSE

$$T_{SCET} = T_{VC0+} - T_{reset} * INT [(T_{VC0+} - T_{SC(1)} + T_{reset}) / T_{reset}]$$

END IF

END IF

From the above it follows that 3 “qualities” of time would be available, viz.:

1. **Actual** as obtained from the corresponding VC0 frame timestamp, this is the nominal case

2. **Extrapolated** as calculated from the nearest available VC0 frame timestamp.

3. **Contingency** no VC0 frames available, this is the worst case and should never occur.

It should be noted that if a PI requests data for which the corresponding VC0 data has not been received and then later, after the VC0 has been received, he requests the same data again both the timestamps and the time "quality" between the two data sets may be different. However, normally VC0 will be transferred from the ground station before the science data and thus this should not occur. In addition it will not affect the CD-ROM production because this is carried out some 5 days after the data has been received from the ground station and by this point all available data will have long since been transferred to ESOC.

The contents of each reconstituted packet of experiment data will be unprocessed, although there will be a small header attached to the beginning of each packet (the DDS Packet Header). This is necessary to include the SCET timestamp, data source identification, ground station received from, etc. These pieces of information are necessary either for the CLUSTER science community member to identify the source of the data packet or for ESOC to identify the source of anomalies and errors. For compactness this header does not use PVL to express the information in the header, but a bit and byte packed fixed format. The header has the format shown in Table 5-3

Byte	Bits	Field	Type	Description
0-7		SCET	CCSDS CDS time format	Packet timestamp, see below
8		Data source/type ID	8 bit integer	See Appendix A
9-11		Packet length	24 bit integer	Number of bytes within the data packet (does not include the size of this CDDS packet header)
12	0-3	Spacecraft ID	4 bit integer	1 = CLUSTER_1 2 = CLUSTER_2 3 = CLUSTER_3 4 = CLUSTER_4
	4-7	Ground station ID	4 bit integer	0 = Unknown 1 = Villafranca 2 = Kiruna 3 = Kourou 4 = Perth 5 = Malindi 6 = Canberra 15 = N/A
13		Data stream	8 bit integer	00 hex = RT VC0 02 hex = RT VC2 03 hex = RT VC3 40 hex = PB VC0 42 hex = PB VC2 43 hex = PB VC3 F0 hex = RE VC0 F2 hex = RE VC2 F3 hex = RE VC3 4F hex = RP VC0 E2 hex = RP VC2 E3 hex = RP VC3 FF hex = N/A
14	0-3	Time Quality	4 bit integer	0 = actual time 1 = extrapolated time 2 = contingency time
	4-7	Telemetry Acquisition Sequence ID (TASI)	4 bit integer	Range 0 -15 (0 = N/A) representing the current TASI as extracted from the CAB

(RT=real-time, PB=playback, RE=recall, RP=recall playback)

Table 5-3: DDS Packet Header Format

Spacecraft Event Time (SCET)

The Spacecraft Event Time (SCET) in each DDS packet header is expressed in the CCSDS Day Segmented (CDS) Time Code format (see Ref. [8]). This is a binary format, which is used in preference to an ASCII representation for compactness.

The CDS format consists of 8 bytes split into 3 segments, each segment represents the state of a right handed adjusted binary counter, cascaded with the adjacent counters, which rolls over at a modulo specified for each counter. The 3 segments (from MSB to LSB) are:

DAY	16 bits	a count of the number of days since the epoch January 1st 1958, starting at 0 (as defined in Ref. [8]).
ms of day	32 bits	a count of the number of milliseconds within the day.
μs of ms	16 bits	a count of the number of microseconds (μs) within the millisecond.

The CDS time is in UTC and leap seconds are accounted for.

Schematically the CDS format is shown in Figure 5-8

	DAY	**ms of day**	**μs of ms**
Segment width (bits)	< 16 >	< 32 >	< 16 >

Figure 5-8: Schematic of the CCSDS CDS Time Format

Therefore, schematically the delivered data product shall look like that depicted in Figure 5-9 (overleaf). Note the experiment packets can vary in length depending upon which acquisition mode the onboard data handler is in.

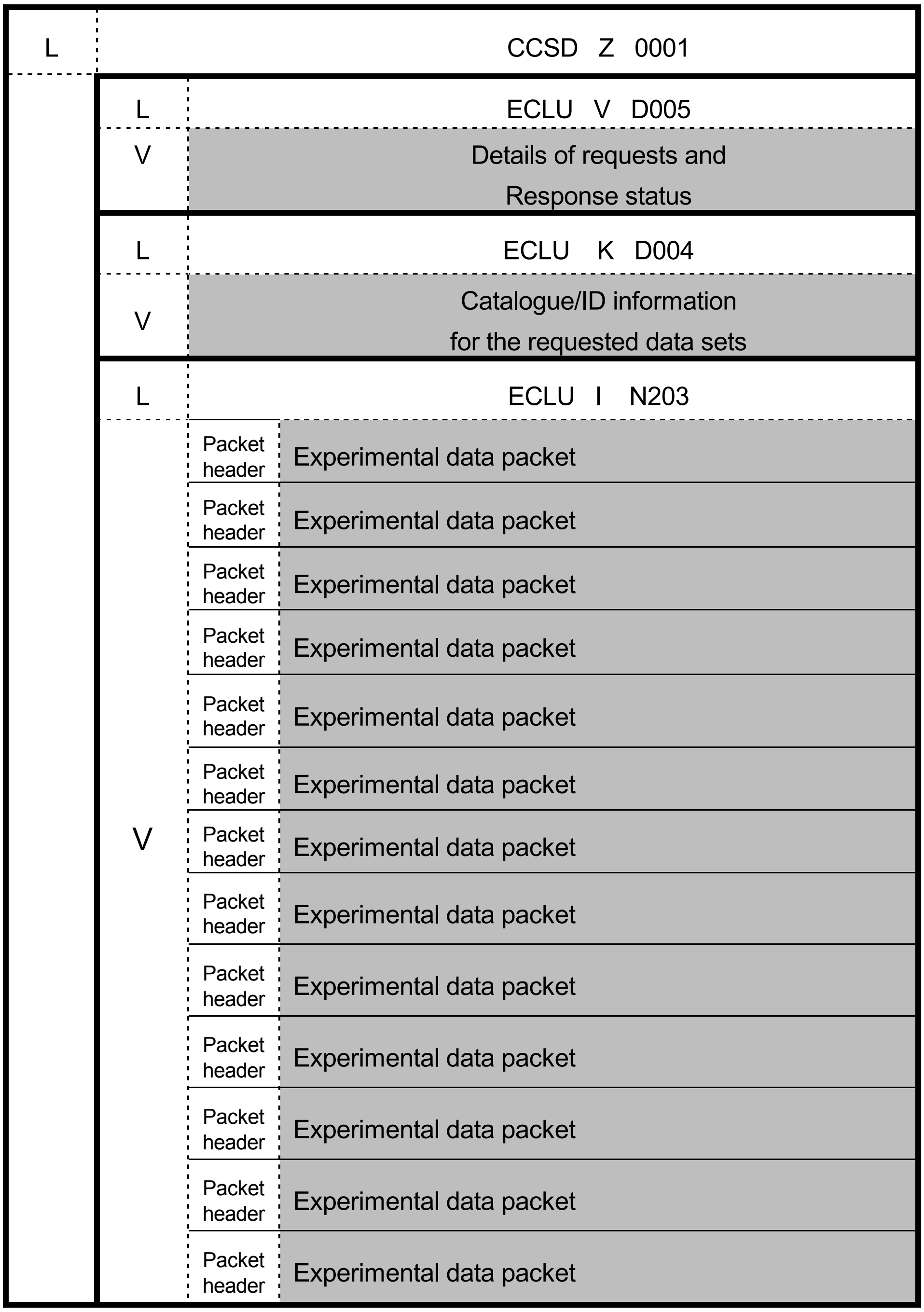


Figure 5-9: Schematic of On-line Data Delivery Format

5.1.3.3.2 Auxiliary Data

Auxiliary data is requested using the same mechanism as for telemetry data. The various files of auxiliary data available are also listed in Appendix A together with their corresponding data source/type mnemonics and ADID. They are referenced in the same manner as telemetry data streams, i.e., with a data source and data type mnemonic. Each auxiliary data entry (for example, a single event from the event file, a single command from the command history file, a set of Chebychev polynomials defining the orbit for a period, etc) is treated the same as an experiment packet from on-board the spacecraft, therefore it will also have the same header as that depicted in Table 5-3 except that the data stream and TASI fields will be set to the "N/A" (not applicable) value, and the time quality field to "actual time". The ground station will also be set to the "N/A" value for all auxiliary files apart from the command history file, where it will correspond to the ground station field in the packet data. For details of the format of the auxiliary data see Appendix E Each of the definitions in Appendix E define a single auxiliary data entry, a number of these (depending upon the timespan requested) will constitute the delivered auxiliary file.

With regard to the on-line retrieval of auxiliary data, Appendix E also describes the difference between "span valid" auxiliary files (such as the STOF, TCAL or HPD files) where a packets data is applicable from the time of that packet up to the time of the next packet in the file, and "point valid" auxiliary files (such as the STEF and CMDH files) where a packets data is applicable only at the packet time.

The catalogue entry for an auxiliary data source is the same as for the telemetry data sources.

5.1.3.3.3 Catalogue Data

As described in the earlier sections each delivery of a data set is accompanied by a catalogue entry. In order for PIs to know what data is available to be requested they must also have the capability to request availability catalogues, these are known as Master Catalogues.

A Master Catalogue request should not be confused with a single catalogue entry for a particular data source/type that is delivered with a requested data set. As described in Section 5.1.3.2 there is a catalogue entry with each data delivery (unless there is an error), this catalogue entry is for that particular data set only, and only covers the requested time span. Further to this, each PI shall also be able to request a Master Catalogue for all the data currently available on-line from the CDDS. When a PI asks for a Master Catalogue, he shall receive catalogue entries for all the available data streams. There is no restriction on the access of data for any PIs or JSOC, e.g. a PI may access data both from his own instrument and that of others. The format of a Master Catalogue is simply all the individual data stream catalogues concatenated together.

To request a Master Catalogue the PI follows the same procedure as for any other data request. The data source is defined as MASTER and the data type as CAT, to indicate a catalogue listing.

In a request for a Master Catalogue, the /SINCE and /BEFORE command qualifiers are still valid. This means that a PI can find out what data is available from all data streams during a particular period. If these command qualifiers are not supplied, then the response shall be a full listing of all the data streams currently held on the CDDS. Any other command qualifiers are also permitted, their meanings being the same as that stated earlier in this document.

A catalogue request, upon being successful, shall result in a file transfer back to the PI, the file having the same structure as shown in Figure 5-4 the catalogue entries that are being delivered shall be in the VALUE field of the Class I LVO; the Class K LVO within the delivered SFDU shall give details of what catalogue entries have been delivered. The catalogue parameters in the Class K LVO, FIRST_PACKET_TIME and the LAST_PACKET_TIME, have no meaning in this situation and therefore shall be a NULL string ("").

The reason for using the aggregation construct (BEGIN_OBJECT / END_OBJECT) around a single catalogue entry becomes obvious if one considers that a number of catalogue entries shall be contained within the single Class I LVO value field, and there would be no way to distinguish each catalogue entry from one another if it were not for this aggregation construct.

For example, a request file (REQUEST.DDS) with the following contents:

```
REQUEST MASTER.CAT EDINODE::[MYDIR]CATALOG.LIS
```

Would result in the catalogue entries, spanning the full length of time of all the data files currently stored on-line for all four spacecraft (Note: The reply would be 4 files, each one a catalogue for a single spacecraft), or, if the request was of the following format:

```
REQUEST /SINCE=1996-08-24 /SC=1 MASTER.CAT
EDINODE::[MYDIR]CATALOG.LIS
```

Then the same catalogue entries would be transferred, but the catalogue time stamps would be from the first packet time-stamp after the /SINCE time until the present time, and only the Master Catalogue for spacecraft 1 would be delivered.

Each catalogue entry shall have the same format as that described in Section 5.1.3.2 , except the NUMBER_OF_PACKETS parameter shall be a null string ("") to indicate an unknown quantity, as it's not possible due to performance reasons to count the number of packets available within any given time span.

5.1.3.4 Error Handling

The only time that the PI shall not receive an SFDU of the structure shown in Figure 5-4 is when an error has been detected during the processing of the request. When this occurs an SFDU containing only the acknowledgement LVO (Class V LVO) shall be transferred, the structure of which is shown in Figure 5-10 (overleaf).

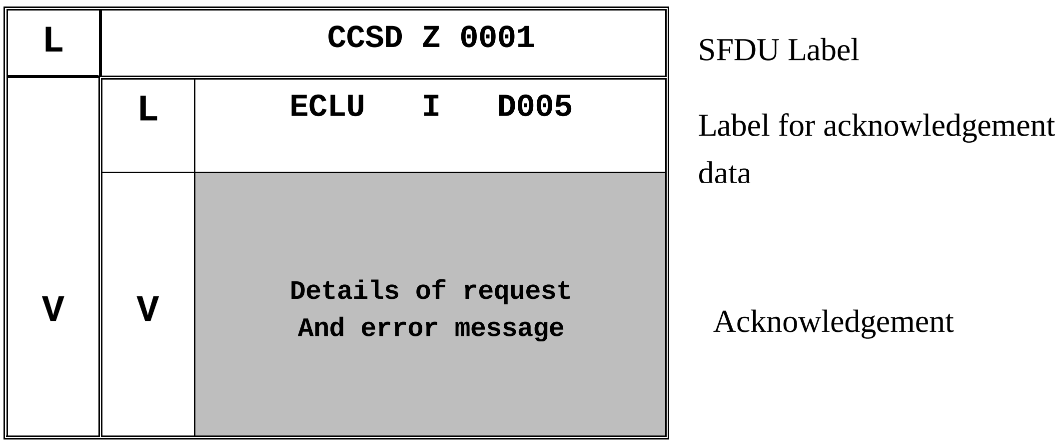


Figure 5-10: Schematic of Data Delivery Format Upon Error

An acknowledgement with an error message may also be returned from the monitoring process on the CDDS. This monitors when requests arrive, and how long it takes to respond to them. If after a defined time-out period the PI has not received the requested data then the PI must assume that either: the CDDS never received the request, system load was too great to process the request, or the requested data was lost in transfer. In this situation the data should be requested again. The overall time-out period is configurable by ESOC (in the range 0 - 32767 minutes); a value of

120 minutes is the default. The monitor process also monitors how many requests each PI has queued for processing by the CDDS, if this is equal to 12, then any new requests shall be deleted and an error message returned immediately.

The format of the acknowledgement LVO upon error is the same as for an error free request, although depending upon the error incurred not all the parameters can be given meaningful values. Any value which cannot be completed correctly shall be substituted by a null string (""). The ERROR_MESSAGE string shall be a suitable error message, these are listed in Appendix B.1 There may be situations when all the parameters of the acknowledgement are completed correctly, but there is still an error, e.g., in the case of a PI requesting a data source/type for which there is no data available during the requested time span, in this case the ERROR_MESSAGE string would show this. If there is no error at all then the ERROR_MESSAGE parameter shall be given the value "NO ERROR".

The ERROR_MESSAGE **string shall have the following format:**

"CLUSTER DDS ERROR-nn: error string"

Where nn is the error message number (see Appendix B.1).

5.2 Off-line Data Delivery

The off-line data delivery system is totally separate and asynchronous to the on-line data delivery system. The PIs have no input in the process to receive data, the system is wholly controlled by ESOC. The CLUSTER science community members shall receive the data on a raw data media (RDM). The chosen media is CD-ROM, formatted to ISO 9660 level 1 compliant. This means the CD-ROM uses the common "8.3" filenaming convention, and the disks are readable on all common platforms, e.g., UNIX, VMS, PC and Mac. Each disk shall contain the data from all data streams from all four spacecraft and also all the auxiliary data produced by ESOC. Further to this, each disk shall contain a copy of all the data descriptions required to understand the data on the disk. This means that the data on each disk shall be wholly self contained.

The CD-ROM for any one day must be delivered within 3 weeks of reception of the data by ESOC. Therefore the logistic and delivery timeline looks like that shown in 5-11. This allows 6 days for the auxiliary data such as orbit and attitude data to be consolidated before a master Write Once CD-R is produced. This master Write Once CD-R is then shipped to the CD-ROM manufacturer for mastering and reproduction. The CD-ROM manufacturer will then deliver the CD-ROMs directly to the appropriate science community members. During the commissioning phase of the mission, CD-ROMs will only be delivered to PIs and data centres (not co-investigators). Thereafter, during the routine phase, they shall be delivered to all PIs, co-Is and data centres.

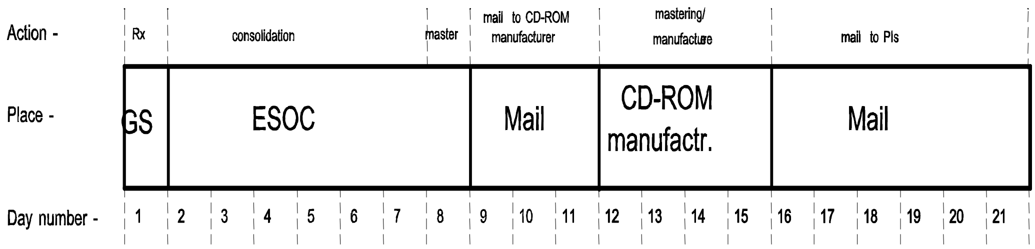


Figure 5-11Timeline for CD-ROM Production Logistics

The data shall be accessible on the CD-ROM by two methods, either the more conventional direct file access, i.e. each data stream shall be stored under a filename in a particular directory, and secondly the whole of the disk may be viewed as a single logical SFDU product. The SFDU product is formed by having an EDU file in the top level directory of the disk that uses the SFDU referencing technique to logically include all the other data files and data description files on the disk into a single SFDU product.

The conventional method of accessing the data will present no problems as long as the science community member knows what he is looking for, where it is on the disk, the filenaming convention etc. When he retrieves it he must already know what it actually means, e.g., the format, syntax and semantics of the data.

The SFDU approach has the advantage that nothing needs to be known about the disk in advance except the name of the top level "index" SFDU file. From this SFDU file the science community member can find all the data on the disk, the corresponding data descriptions and via the ADIDs can link the two together. So for long term archive access when documentation and knowledgeable people may not be available this assures the continued understanding of the data.

The CD-ROM is shown schematically in Figure 5-12 below. The conventional directory structuring and individual files are shown in black whilst the SFDU index file in the root directory of the CD-ROM and its file reference pointers are shown in grey.

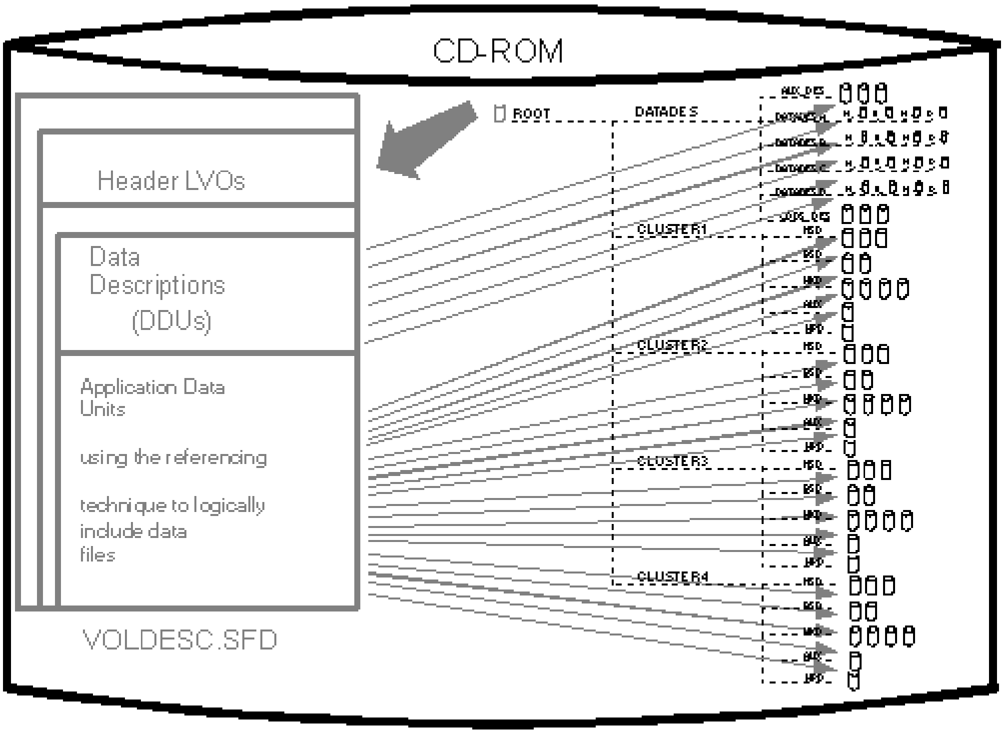


Figure 5-12 Physical and SFDU View of the CD-ROM

5.2.1 SFDU Structuring on the CD-ROM

The files on the CD-ROM shall be made to look like one logical SFDU product by using a single SFDU file in the root directory that shall then use the SFDU referencing technique to logically include all the other files on the disk. In other words, this top level SFDU file describes the whole volume of data, therefore it is called a VOLDESC file. The only filename required to be known is the name of this top level SFDU file.

The general structure of the VOLDESC file is shown in Figure 5-13 below. This is an EDU which contains:

1. Header LVOs which give details of the overall SFDU product, such as the volume production information, directory listing, cumulative index, etc. Each of these are items will be contained in separate LVO value fields.

2. A further EDU containing:

    - Description Data Units (DDUs)

    - Application Data Units (ADUs)

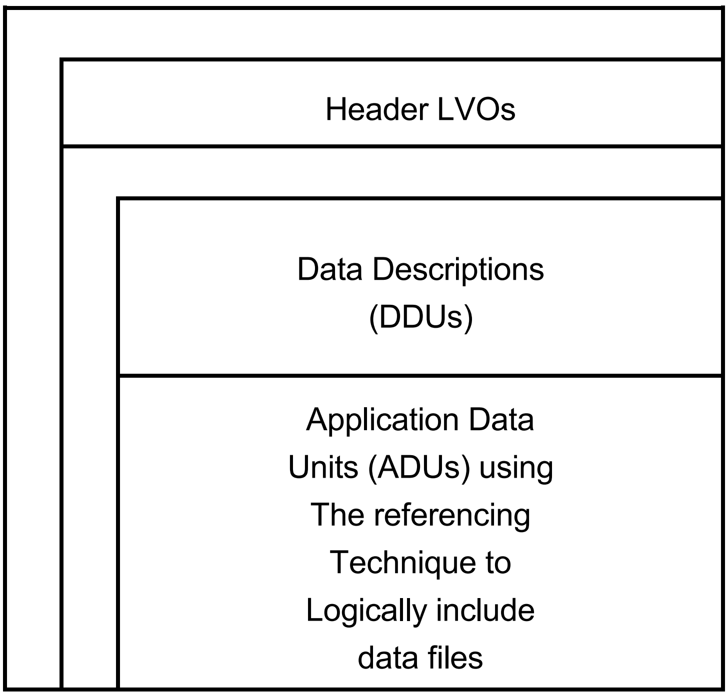


Figure 5-13: General Structure of the VOLDESC SFDU File

All these DDUs and ADUs use the referencing technique to point to the relevant data objects stored elsewhere on the CD-ROM, i.e., physically external to the VOLDESC SFDU file.

To demonstrate how the VOLDESC SFDU file is used for CLUSTER, Figure 5-13 shall gradually be expanded to include the details of each of the three areas shown (Header LVOs, Data Description Units and Application Data Units). The structure expanded in more detail is shown in Figure 5-14 .

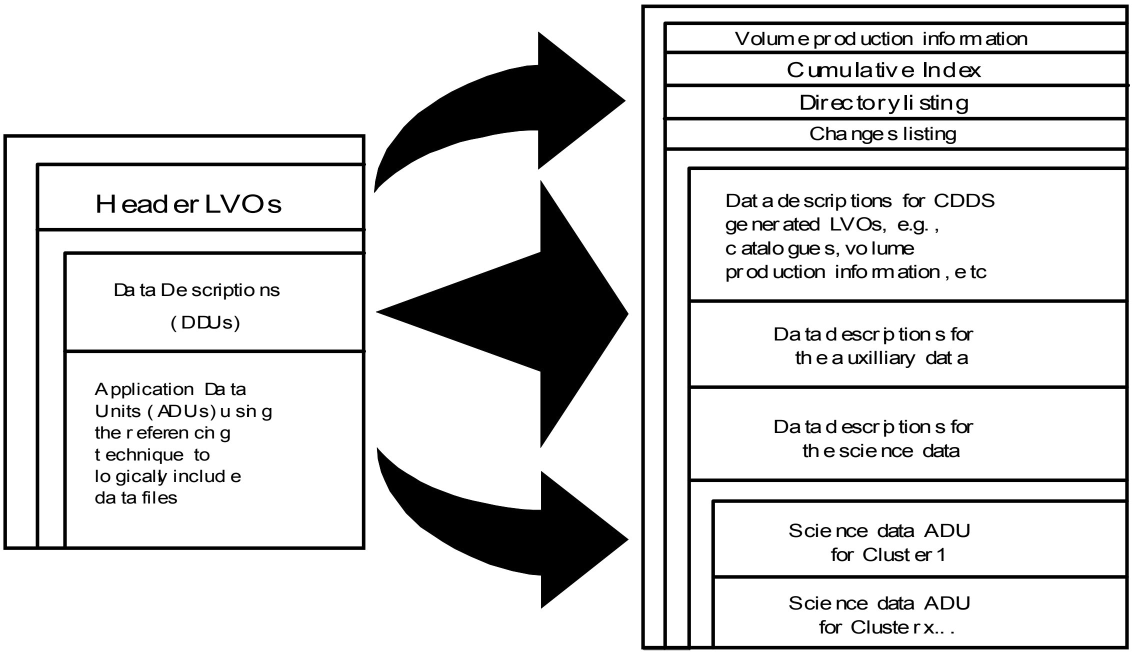


Figure 5-14Details of the 3 Main Elements of the VOLDESC SFDU File

The Header LVOs consist of data that pertains to the whole SFDU product, for example, the volume production information, supplementary data such as a directory listing of the whole CD-ROM, version and configuration information, etc. As the information in the header LVOs applies to all the data description and science data, all further LVOs are at a structure level lower than the header LVOs.

The data description LVOs consist of a sequence of DDUs. There is one DDU for each data description that is supplied on the CD-ROM. There are three categories of data descriptions; those describing the data formats used by the CD-ROM itself, e.g., the catalogue entries, the volume production information, etc.; those that describe the format of the auxiliary data produced by ESOC, e.g. time calibration data, orbit and attitude data; and those that describe the format of the instrument data from the spacecraft.

Following the header LVOs and the DDUs, there is the instrument data itself. Each stream of data generated by the spacecraft or ESOC is packaged in an Application

Data Unit (ADU), this is so that a catalogue entry giving catalogue details of the associated data file can be included within the ADU. This catalogue entry shall relate to the data file within the same ADU only.

Further detail of the LVOs involved in the structure shown in Figure 5-14 are shown Figure 5-15.

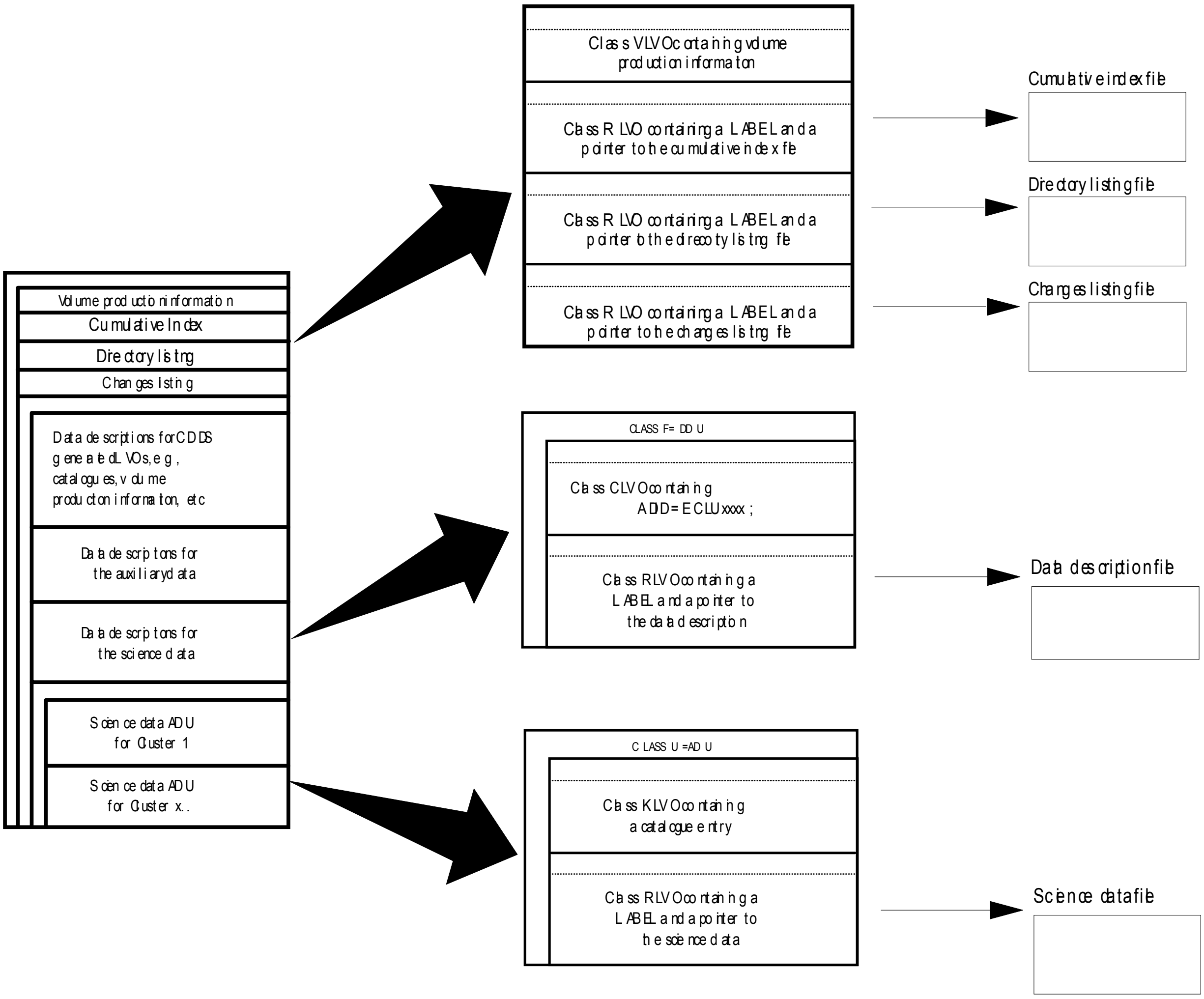


Figure 5-15 Details of the LVOs within the VOLDESC SFDU File

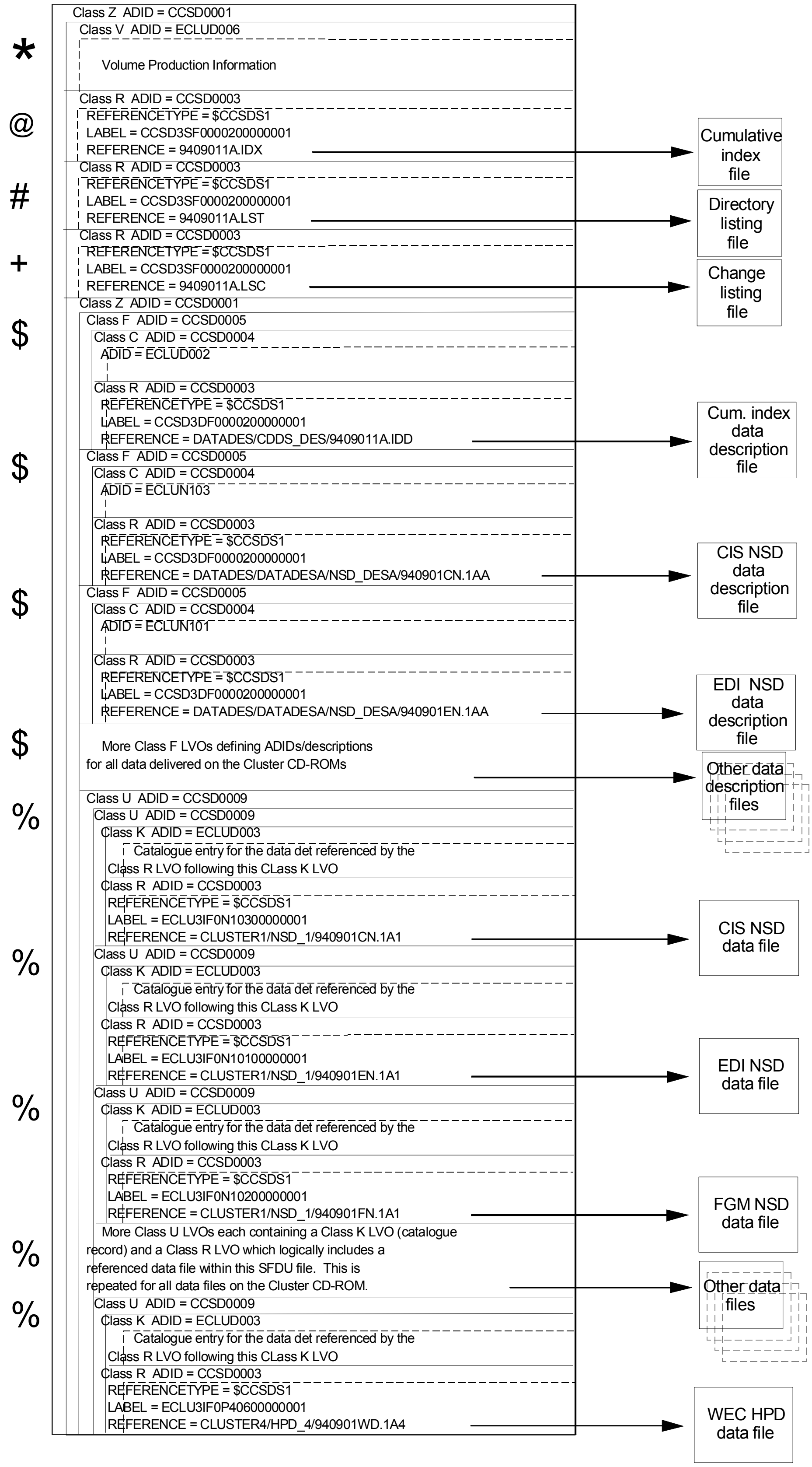
As can be seen, the VOLDESC file uses the file referencing service provided by the Class R LVO to logically include the various files in the overall SFDU. In the header LVOs, the directory structure listing and version control file are in files in the root directory of the volume, the Class R LVOs point to these files and hence they are logically included as part of the header information. For each Data Description Unit (DDU, signified by the Class F LVO), the Class C LVO is in-line in the VOLDESC file, this LVO assigns the ADID to the data description via the ADID=ECLUxxxx statement. The actual data description itself is contained in a disk file that is again logically included using the Class R referencing service. Finally each ADU, contains in-line a Class K LVO that contains the catalogue information for the associated data in the ADU, and another Class R LVO points to the data and logically includes it in the VOLDESC SFDU product.

In the VOLDESC file there is one DDU for each data description used on the CD-ROM, and then there is one ADU for each science/auxiliary data file stored on the CD-ROM. A schematic of this overall structure is shown in detail in Figure 5-16 . This figure includes the details of the statements used in the Class C and Class R LVO to provide the registration and referencing services respectively.

Referring to the symbols at the side of the SFDU shown in Figure 5-16 the various LVOs in the VOLDESC file have the following contents:

Class V LVO (*****): This LVO contains the volume production information for the whole SFDU, and therefore applies to the total contents of the CD-ROM. The volume production information is expressed in PVL, the valid parameters and their corresponding values are shown in Table 5-4

Figure 5-16 Detailed Schematic of the VOLDESC SFDU File



Parameter	Possible values
MISSION	CLUSTER
PREPARED_BY	"European Space Operations Centre, Robert Bosch Strasse 5, 64293 Darmstadt, Germany"
EARLIEST_PACKET	Time of earliest NSD, BSD or HKD on this CD-ROM. CCSDS format time (YYYY-MM-DDThh:mm:ssZ**). "not applicable" if only AUX/HPD on this CD-ROM.**
LATEST_PACKET	Time of latest NSD, BSD or HKD on this CD-ROM. CCSDS format time (YYYY-MM-DDThh:mm:ssZ**). "not applicable" if only AUX/HPD on this CD-ROM.**
TOTAL_DATA_QUANTITY	Integer number of bytes of data in all files referenced by the VOLDESC file, e.g., 12345 <bytes>
COMPUTER	Quoted string
OPERATING_SYSTEM	Quoted string
DDS_SOFTWARE_VERSION	Quoted string
PRODUCTION_TIME	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**)**

Table 5-4 VOLDESC Volume Production Information

Class R LVO (@): This LVO is used to reference an ASCII encoded English text file (as indicated by the ADID=CCSD0002; (see Ref. [5]). This text file is given a Class S LABEL (meaning Supplementary Data) by the referencing Class R LVOs LABEL statement. The contents of the referenced file is the cumulative index for the CD-ROM production (See Section 5.2.4) This file lists all the CD-ROMs that have been produced over the life of the mission until the present time. It also indicates any CD-ROMs that have been replaced by a corrected one, in the case of an erroneous CD-ROM issue. This version index is cumulative over the whole mission.

Class R LVO (+): This LVO is used to reference an ASCII encoded English text file (as indicated by the ADID=CCSD0002; (see Ref. [5]). This text file is given a Class S LABEL (meaning Supplementary Data) by the referencing Class R LVOs LABEL statement. This file will only have any content if the CD-ROM on which it is on is a reissue or a previously issued CD-ROM that was found to have an error. In this case, this file will list those files that are corrected for the reissue. This means the science community members know immediately if the corrected data concerns them and they should reprocess the CD-ROM. The contents of the referenced file is in the same format as the directory listing file (see next LVO). This shall show the full filenames with the directory name and the new file size

Class R LVO (#): This LVO also references an ASCII encoded English text file (as indicated by the ADID=CCSD0002; (see Ref. [5]). This text file is given a Class S LABEL (meaning Supplementary Data) by the referencing Class R LVOs LABEL statement, and the contents of the referenced file is a full directory listing of the CD-ROM. This shall show the directory structure and list of files in each directory with the file size.

Class F LVOs ($): There then follows a number of Class F LVOs, each of these LVOs is a Data Description Unit (DDU) and links one ADID with one data description. Nested within the Class F LVO is a Class C LVO followed by a Class R LVO. The Class C LVO has the PVL statement ADIDNAME=ECLUnnnn, this defines the ADID under which the data description following shall be referenced. The Class R LVO then uses the referencing technique to point to a data description file, this file is logically included within the Class F LVO and hence is associated with the named ADID. There is one Class F (each containing one Class C and one Class R) for each data description used on the CD-ROM.

Class U LVOs (%): The next Class U LVO contains many other Class U LVOs, these are all Application Data Units (ADUs), each of the lower level Class U LVOs contain a Class K LVO and a Class R LVO pair. The Class K LVO is the catalogue data for the corresponding data set pointed to by the Class R LVO. The Class R LVO is used to reference and hence logically include the data set file within the Class U. The LABEL statement in the Class R LVO contains an ADID that defines the data description to be used for understanding the data set. There is one nested Class U LVO for each data file that is stored on the CD-ROM**.**

The Class K LVOs VALUE field is a catalogue entry, this catalogue entry is written in PVL, the valid parameters and their values are shown in Table 5-5 (overleaf). As can be seen this is the same catalogue entry as for the on-line delivery situation, except that an extra parameter is included: CD-ROM_FILENAME**; the value of which is the referenced data files full filename including the directory path on the CD-ROM.**

The file pointed to by the Class R LVO has the same format as that defined in Section 5.1.3.3.1 That is the same format for science and auxiliary data is used for on-line delivery and off-line delivery.

Parameter	Possible values
SPACECRAFT_NAME	CLUSTER_1, CLUSTER_2 CLUSTER_3, CLUSTER_4
DATA_SOURCE	See Appendix A
DATA_TYPE	See Appendix A
ADID	Eight ASCII characters, e.g, ECLU0009
EARLIEST_PACKET	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**) Set to the string** "not applicable" **if the data file is empty for that day.**
LATEST_PACKET	CCSDS format time (YYYY-MM-DDThh:mm:ssZ**) Set to the string** "not applicable" **if the data file is empty for that day. Set to the same value as** EARLIEST_PACKET **if there is only one packet in the file on that day.**
NUMBER_OF_PACKETS	ASCII encoded decimal number, no leading zeros
CD-ROM_FILENAME	The full CD-ROM filename, including directory name

Table 5-5 Off-line Catalogue Entry Specification

As can be seen, in this way all the data files and data descriptions on the CD-ROM are logically included within the VOLDESC file, and the CD-ROM can be seen as a single SFDU product.

5.2.2 CD-ROM Directory Structure

The CD-ROM shall have a conventional hierarchical directory structure that divides the data into separate directories for each spacecraft and data type, the structure can be represented as shown in Figure 5-17

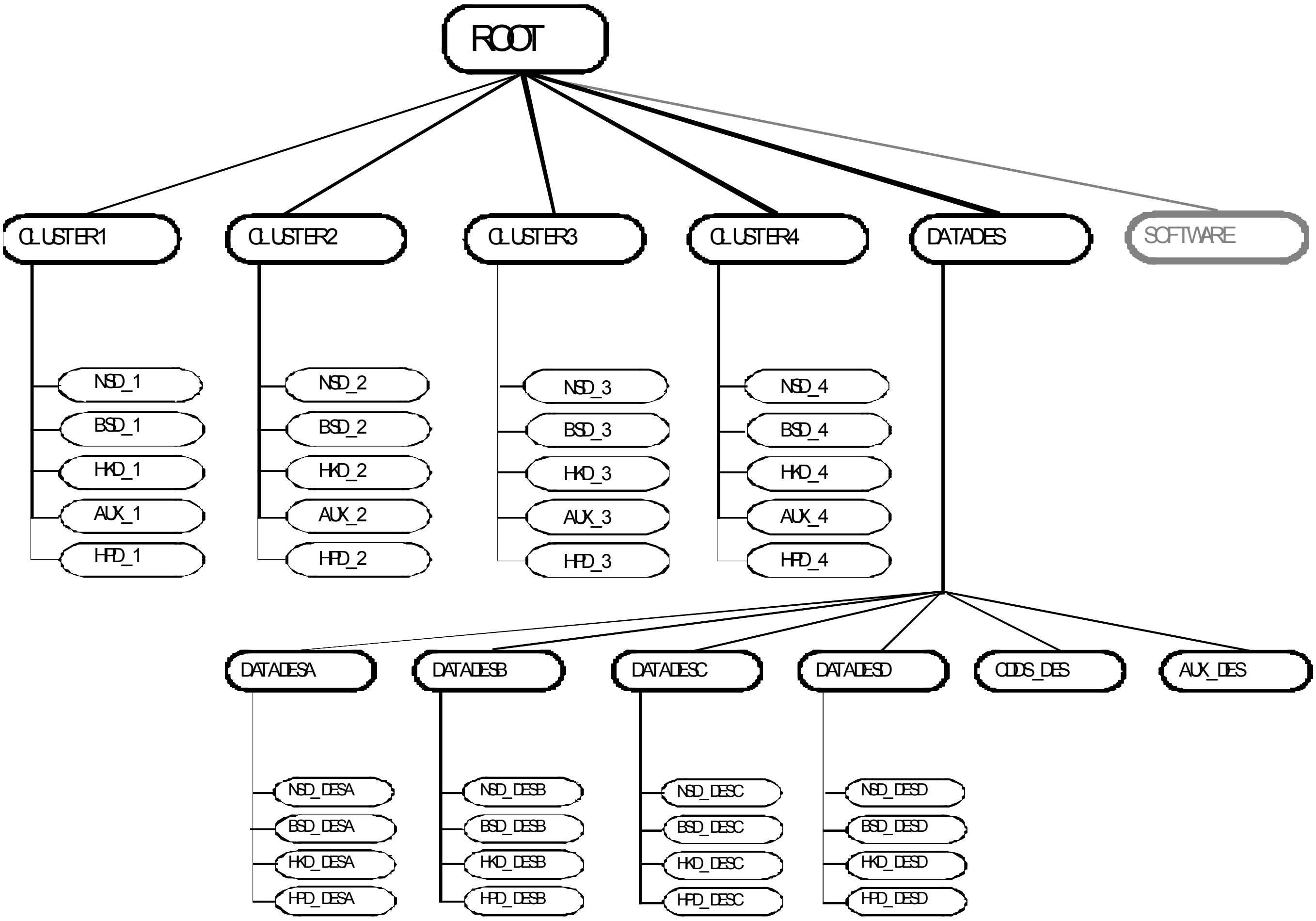


Figure 5-17: CD-ROM Directory Structure

The directory structure is self describing. The data generated by the spacecraft and ESOC is split into four sub-directories, one for each spacecraft, in each of these sub-directories there are five further sub-directories, these subdivide each spacecraft data into the Normal Science Data (NSD), the Burst Science Data (BSD), the Housekeeping Data (HKD), the Auxiliary Data (AUX) and the Housekeeping Parameter Definition Data (HPD) as generated by ESOC.

The DATADES sub-directory under the top root level contains files of all the data descriptions. Within the DATADES subdirectory there is a subdirectory for each spacecraft, each of these directories contain four subdirectories, each one containing the description files for each of the data types, NSD_DESx for Nomal Science Data description files, BSD_DESx for Burst Science Data description files, HKD_DESx for Housekeeping Data description files and HPD_DESx for Housekeeping Parameter Definition description files. The x indicates the spacecraft that they pertain to, where A is Cluster 1, B is Cluster 2, C is Cluster 3 and D is Cluster 4. There are separate directories for each spacecraft as due to changes or failures on different spacecraft the data description information may not be the same for each spacecraft.

Also under the DATADES directory there is an AUX_DES directory that contains descriptions of all auxiliary files. These files will always be identical in format for all spacecraft as they are centrally generated by ESOC on the ground and not directly by the spacecraft.

Furthermore, under the DATADES directory there is an CDDS_DES directory that contains descriptions of the files that are specific to the CDDS system and the CD-ROM, such as the directory listing file and the cumulative index files.

The SOFTWARE sub-directory contains software routines delivered by ESOC to assist in processing orbit data (see Appendix H)

Note that each CD-ROM shall contain all files even if the files have zero size. This may occur when there is no data for a particular date for a data source. Zero size files are included, as this makes clear that there was no data for the data source in the time range. If a file is completely missing then it immediately indicates a manufacturing error.

5.2.3 CD-ROM Filenaming Convention

All the files on the CD-ROM follow a particular filenaming convention, this is so that if a file should be copied from the CD-ROM to another media with no knowledge of which directory it came from, then the filename alone is enough to unambiguously identify the file. The filenames follow the ISO 9660 level 1 standard for CD-ROMs. This has a basic format of a directory path followed by an "8 dot 3" filename. Whilst this is quite restrictive, it does mean that the files and directory structure will transfer to any known random access media with no filename mapping required. The filenames have the following convention:

yymmddst.nvc

Where:	yy	=	last 2 digits of the year of the date of the first packet in the file (range 00 - 99).
	mm	=	month of the date of the first packet in the file (range 01 - 12).
	dd	=	day of the date of the first packet in the file (range 01 - 28,

		29, 30 or 31 depending upon the month).
s	=	single character code indicating the data source (see below).
t	=	single character code indicating the data type (see below).
n	=	the CD-ROM number within the particular day (range 1-4, first CD-ROM = 1). This field will be the same for every file on any single CD-ROM.
v	=	the version of the file in case of reissue (range A-Z, initial version = A).
c	=	single character code indicating the spacecraft (see below).

Table 5-6 gives a list of all the possible values for the above filename segments. For those files that are specific to the CD-ROM only and do not originate on the spacecraft or the control system, i.e. data description files, a spacecraft ID character of A, B, C, D or X is used in the extension**.**

There are a small number of files whose names do not follow the convention defined above, these are:

yymmddnv.LST	CD-ROM directory listing
yymmddnv.LSC	CD-ROM directory listing of files changed since last issue of this CD-ROM (empty for first issue)
yymmddnv.IDX	Cumulative index file for CD-ROM version control (see Section 5.2.4)
yymmddnv.LDD	Data description file for CD-ROM directory listings
yymmddnv.IDD	Data description file for cumulative index file
yymmddnv.CDD	Data description file for the catalogue LVO (Class K LVO) in the VOLDESC file
yymmddnv.VDD	Data description file for the volume production information LVO (Class V LVO) in the VOLDESC file
VOLDESC.SFD	Root SFDU file

Filename segment	Possible values	Notes
C	1, 2, 3, 4	Used to represent CLUSTER spacecraft 1, 2, 3 and 4 respectively
	A, B, C, D	Used to represent CLUSTER spacecraft 1, 2, 3 and 4 data description files respectively
	X	Used to indicate auxiliary data description files
S	A	ASPOC data files
	E	EDI data files
	F	FGM data files
	C	CIS data files
	P	PEACE data files
	R	RAPID data files
	W	WEC data files
	S	Spacecraft platform files, such as platform housekeeping data
	B	Short Term Orbit file
	T	Short Term Event file
	V	Covariance Matrix file
	G	Spacecraft Attitude and Spin Rate file
	L	Time calibration file
	I	Command history file
T	N	Normal science data
	B	Burst science data
	H	Housekeeping data
	D	Housekeeping Parameter Definition File
	A	Auxiliary data

Table 5-6 Filename Segment Values

For example, normal science data from CLUSTER spacecraft number 2, from the EDI instrument with the first packet from 27th April 1997, would reside in the directory:

/CLUSTER2/NSD_2/

relative to the root directory, and would have the filename:

970427EN.1A2

assuming that the file resided on the first CD-ROM produced for that day, and was the original CD-ROM, i.e., not a reissue.

Similarly, the data description file for this data source (if written on the 12th February 1995) shall reside in the directory:

/DATADES/DATADESB/NSD_DESB/

relative to the root directory, and would have the filename:

950212EN.1AB

Each CD-ROM has a capacity of approximately 650 Mbytes. Under the current baseline for data production, up to about 1 GByte of data per day will be generated. If the data produced in one day can fit in a single CD-ROM, then this should hold one whole days worth of data, starting at the first packet received after or including 00:00:00.00 (midnight) and finishing with the last packet received before 00:00:00.00 (exclusive) of the next day. If due to the data reception profile over a single orbit, more than 600 Mbytes of data is received within a single calender day, then a first CD-ROM for that day will be filled until the 600 Mbyte limit is reached, and a second CD-ROM continued from that point onwards. All data streams and relevant associated data will be on both CD-ROMs, the only difference being that there will be a certain point in time (dictated by the 600 Mbyte maximum), that all the streams will continue on the second CD-ROM. Of course this process may continue onto a third or fourth CD-ROM (the maximum being four CD-ROMs per day) or until the last packet timestamped before 00:00:00.00 (exclusive) of the next day. As stated above, each CD-ROM will also hold all the data descriptions required to "understand" the data.

With regard to the handling of "splitting" data across several CDs for a day, Appendix E also describes the difference between "span valid" auxiliary files (such as the STOF, TCAL or HPD files) where a packets data is applicable from the time of that packet up to the time of the next packet in the file, and "point valid" auxiliary files (such as the STEF and CMDH files) where a packets data is applicable only at the packet time.

See Appendix D for a full list of all files that shall be delivered on each CD-ROM and their position in the directory structure.

5.2.4 Version Control

Due to the fact that raw unprocessed data is being delivered, it is not possible to have a new version of such data, but what must be considered is the possibility of a fault in the manufacture of the CD-ROM, an incorrect selection of files being placed on the CD-ROM or a new release of auxiliary files due to an error in the CDPF. Therefore the system has been designed to handle new versions of files in the extraordinary situation that it may arise.

The present baseline for the system is that it shall generate 2 CD-ROM per day, another consideration is the possibility that more than 2 CD-ROM per day may be necessary, for example in the situation that the orbit period is changed, or the data reception profile is not even over an orbit, or a whole CD-ROM has to be reissued due a manufacturer error.

In the root directory of each CD-ROM there will be an index file. This file will list the full collection of Cluster CD-ROMs since launch. Against each CD-ROM title it shall indicate the time span that the CD-ROM covers (in case there is more than 1 CD-ROM produced in a single day), and also the issue indicator so that it is easy to see if an older CD-ROM has been replaced by a reissued one. This index file will be cumulative, i.e., it contains a record of all issued CD-ROMs up until the present date.

Furthermore, so that each CD-ROM is self identifying (should for example the CD-ROM be separated from its case), the cumulative index file on each CD-ROM shall have a unique filename that identifies the date of the data on the CD-ROM and the number of the CD-ROM within that day. The filename shall be of the format:

yymmddnv.IDX

Where:	yy	=	last 2 digits of the year of the date of the data on the CD-ROM (range 00 - 99)
	mm	=	month of the date of the data on the CD-ROM (range 01 - 12)
	dd	=	day of the date of the data on the CD-ROM (range 01 - 28, 29, 30 or 31 depending upon the month)
	n	=	the CD-ROM number within the particular day (range 1-9, first CD-ROM = 1).

v	=	the version indicator of the CD-ROM in the case of reissue (range A-Z, initial version = A).
IDX	=	fixed 3 character string, indicating that this is the CD-ROM identifier filename and index file

For example a cumulative index file named 9608122A.IDX, would indicate that the CD-ROM contained data from the 12th August 1996, and that this was the first issue of the second CD-ROM produced for that day.

Each entry in the cumulative index file contains: the disk identifier (i.e., cumulative index filename), start and end date/time of the science data, the number of the disk within the day and the total number of disks for the day, the version number and an indicator if the disk has been deleted (i.e., replace by a newer issue). The fields and their character positions within the each single line entry are defined in Table 5-7 below:

Field	Bytes	Description
CD-ROM ID	0-7	The identifier of this CD-ROM (the cumulative index filename minus the extension, e.g. 9512051A).
	8	Space character (ASCII 20h)
START TIME	9-28	The inclusive start time of the CD-ROM in CCSDS format (YYYY-MM-DDThh:mm:ssZ). For the first CD in the day the hh:mm:ss component will always be 00:00:00. For subsequent CDs within the day, start time will be the (inclusive) time of the split between the previous CD and this one.
	29	Space character (ASCII 20h)
END TIME	30-49	The exclusive end time of the CD-ROM in CCSDS format (YYYY-MM-DDThh:mm:ssZ). For the last CD in the day the hh:mm:ss component will always be 00:00:00 and the YYYY-MM-DD component will be the date of the day following the start time (since end time is exclusive). For previous CDs within the day, end time will be the (exclusive) time of the split between this CD and the next one.
	50	Space character (ASCII 20h)
CD-ROM NUMBER	51	The number of the CD-ROM within a single day.
	52	Space character (ASCII 20h)
TOTAL CD-ROMS	53	The total number of CD-ROMs in the single day.
	54	Space character (ASCII 20h)
VERSION	55	The version indicator for the CD-ROM, A being the initial version.
	56	Space character (ASCII 20h)
STATUS	57-66	A field used to indicate the status of the particular CD-ROM. This field is blank under normal circumstances or the string DELETED (right padded with three space characters - ASCII 20h) may be present if the CD-ROM has been replaced by a newer version.
	67	Linefeed character (ASCII 0Ah)

Table 5-7 Entry in the Cumulative Index File

Note that the data fields within the record are separated by space characters (ASCII 20h). The last byte in each entry contains a linefeed character (ASCII 0Ah). Each field value is left aligned within the field and padded on the right with spaces. The Cumulative Index File itself starts with a two line header (each line being terminated by a linefeed character - ASCII 0Ah).

The cumulative index file is sorted in reverse chronological order, that is the newest CD-ROM is listed at the start of the file. Any CD-ROM reissues are inserted in the list in the correct date order, just prior to the disk that they are replacing. The index file is maintained until the end of the mission.

There will be a fixed 2 line header to the file, naming the columns of information as shown in the example file in Figure 5-18 below.

```
ID          Start              End        n T V   Status
========-===================-===================-=-=-=-==========
9512091A 1995-12-09T00:00:00Z 1995-12-10T00:00:00Z 1 1 A
9512081A 1995-12-08T00:00:00Z 1995-12-09T00:00:00Z 1 1 A
9512072A 1995-12-07T13:14:15Z 1995-12-08T00:00:00Z 2 2 A
9512071A 1995-12-07T00:00:00Z 1995-12-07T13:14:15Z 1 2 A
9512061A 1995-12-06T00:00:00Z 1995-12-07T00:00:00Z 1 1 A
9512051B 1995-12-05T00:00:00Z 1995-12-06T00:00:00Z 1 1 B
9512051A 1995-12-05T00:00:00Z 1995-12-06T00:00:00Z 1 1 A DELETED
9512041A 1995-12-04T00:00:00Z 1995-12-05T00:00:00Z 1 1 A
9512033A 1995-12-03T20:10:05Z 1995-12-04T00:00:00Z 3 3 A
9512032A 1995-12-03T09:56:40Z 1995-12-03T20:10:05Z 2 3 A
9512031A 1995-12-03T00:00:00Z 1995-12-03T09:56:40Z 1 3 A
9512021B 1995-12-02T00:00:00Z 1995-12-03T00:00:00Z 1 1 B
9512021A 1995-12-02T00:00:00Z 1995-12-03T00:00:00Z 1 1 A DELETED
9512011A 1995-12-01T00:00:00Z 1995-12-02T00:00:00Z 1 1 A
9511311A 1995-11-31T00:00:00Z 1995-12-01T00:00:00Z 1 1 A
```

Figure 5-18: Example Cumulative Index File

Whenever a science community member wishes to access data from his collection of CD-ROMs, he must first consult the cumulative index on the most recent CD-ROM that he possesses, and search for the particular date and time that he is interested in and then from that entry he will see the relevant CD-ROM identifer and whether it has been replaced by a newer issue. The science community members must always use the cumulative index from the most recent CD-ROM. Reissued CD-ROMs will have a cumulative index file containing entries for all the CD-ROMs up to the date that the reissued CD-ROM was produced.

As already mentioned, the root directory will also contain a file showing a directory listing of the files that have been corrected for this issue of the CD-ROM. Science community members can consult this list of changed files to see if any of their data was affected by whatever problem caused the CD-ROM to be reissued.

The contents of this file will be similar to the directory structure show in Appendix D apart from the fact that only files corrected for this issue (and the directories containing them) will be listed. For the first issue of any CD-ROM, this file will be empty.

Note that when a multiple CD volume set is corrected, the times where the data is "split" across CDs may be different from the original volume set. The changed files list on each CD will contain only the corrected files for that CD, not files which have only changed due to their split time being different.

The data description of this changed listing file will be the same as that for the full CD-ROM directory listing file.

5.2.5 Physical CD-ROM Identification

Each CD-ROM shall have a printed label in the top surface of the CD-ROM. This label shall identify the physical CD-ROM for handling purposes. The format of the label shall be :

yymmdd_n_tv

where:	**yymmdd**	is the date of the science data on the CD-ROM
	n	is the number of the CD-ROM within the particular day
	t	is the total number of CD-ROMs within the particular day
	v	is the version number of the CD-ROM, (range A-Z, where A is the initial release)

The volume label of the CD-ROM will also use this format. For example, a CD-ROM from 12 December 1996 which is the second of three CD-ROMs, and is the initial issue, would have the following printed label and volume label:

961212_2_3A

5.2.6 Data Description Information

Following the CCSDS SFDU standard, each file of data delivered is tagged (either physically or via a pointer) with a label that includes an indicator of the data description information that clearly defines the format of the data. This indicator is called the Authority and Description Identifier or ADID. It indicates not only the particular description that applies to the attached data, but also the space agency that archives and maintains the description. In the case of the Cluster mission, all the raw data descriptions are stored at ESOC, therefore of the 8 characters allocated to the ADID, the first 4 (the Control Authority Identifier) are always "ECLU**". The remaining 4 characters are assigned by the Cluster mission as defined in Appendix A**

So that the data on the CD-ROMs is understandable without other external data sources or repeated access back to ESOC for the registered descriptions, each CD-ROM shall contain a full copy of the data description information that is available, for all the data delivered on that CD-ROM. This includes science data, housekeeping data and auxiliary data. For each of these sources of data the description data provided is of varying degrees of detail, depending upon what is available to ESOC.

For the science data, that is spacecraft Normal Science Data and Burst Science Data, the description provided is at a relatively high level, with no byte or bit level information. This is due to the fact that the inner details of the experiment packets lie in the domain of the PIs and not with ESOC, and ESOC has no method of access to this data. Therefore the data description shall simply state the following:

- Which experiment and spacecraft the data originates from;

- The institute and contact point that is responsible for the experiment and therefore the point of contact to get further details of the experiment data format.

- Any other more detailed information that can be provided by the PIs to explain the data format.

For the auxiliary data, i.e. that data which is provided directly by ESOC, such as orbit and attitude data, command history data, time calibration data etc., the full description of the data will be provided as an ASCII English text document. Essentially each auxiliary data file is described in Appendix E these descriptions will be registered at ESOC as Cluster ADIDs, and then copied onto each CD-ROM. No further information will be required to fully interpret this data.

The housekeeping data (both the platform and the experiments) is the most complex to describe completely. The reason for this being that for a fully meaningful description of the housekeeping data, not only the physical layout of the data within the delivered packets needs to be known, but also the calibration information that is stored within the operational control database at ESOC. Furthermore, the operational database is updated frequently as commissioning takes place, calibration information changes and various other operational tasks are undertaken. Obviously, with such a dynamically changing description, it is not possible to register this description under a single ADID that is static for the duration of the mission. The technique used to cope with this dynamically changing description is to treat the description information as another data stream which has to be interpreted so as to understand and process the actual housekeeping packets.

The conclusion as far as ADIDs are concerned, is that for the housekeeping packets they have an ADID which provides a basic description similar to the description for science data plus an explanation which specifies that the housekeeping parameter definition (HPD) file should be used to interpret/understand the information, e.g. ***"This is the housekeeping data for the PEACE experiment on-board spacecraft Cluster 1. The detailed description of the bytes and bits in this data stream are provided by the PEACE HPD file, the format of which is provided by ADID = ECLUxxxx".***

The HPD files are produced from an export of the operational database used by the control team at ESOC. This export shall be split into separate HPD files for each of the seven experiments and also one for the platform. This corresponds to the separate housekeeping data files that will be delivered. Each HPD file will contain the following information for each housekeeping parameter:

- Telemetry Acquisition Sequence ID;
- Parameter mnemonic;
- Start byte and bit position within the housekeeping packet;
- The length of the parameter in the packet bits;
- The data type of the parameter, e.g., integer, unsigned integer, boolean etc.;
- A short English description of the parameters meaning/use;
- The necessary calibration curve values/enumerated values for interpreting the parameter in a meaningful manner

The data description files for these eight HPD files will contain the description of the format of the files, for example, (an illustrative summary only): ***"This data is a description of the housekeeping data for the PEACE experiment on board the Cluster 1 spacecraft. It specifies in tabular form the telemetry acquisition sequence ID being used , the name of each parameter, the start byte and bit position, the length, the data type, an English description of the parameter and any necessary calibration information. The column positions are ......This information is required to interpret and understand the housekeeping data***

***produced by the PEACE experiment on board the Cluster 1 spacecraft, which is labelled with ADID = ECLUyyyy*"**. The exact format of the HPD files is defined in Appendix E.8

Every time that the operational database is changed, (possibly 2 or 3 times a week in the early stages of the mission), a new database export will be produced, and correspondingly eight new HPD file entries. Each of these entries will be treated as a single large packet of data and filed with a timestamp from when it is valid. It is valid until a new entry (operational database extraction) with a later time stamp is available.

For the science community members to process the housekeeping data, they must look at the timestamp of the housekeeping data in question, and then retrieve from the HPD file the entry with the most recent timestamp previous to this. This entry will describe the housekeeping packet in question.

Appendix A Data streams available and ADIDs

This appendix lists all the data streams that are available from the Cluster DDS on-line and also delivered on CD-ROM. When a PI requests a data stream on-line he specifies the data source, data type and spacecraft number. The data source and data type are listed in the first 2 columns of the table below, the data type, which indicates a general categorisation of the data content, must be one of the following:

MNEMONIC	Data Type
AUX	Auxiliary data
CAT	Catalogue data
HPD	Housekeeping parameter definition data
HKD	Housekeeping data
NSD	Normal science data
BSD	Burst science data

As has already been stated, each packet of data delivered has a DDS packet header attached to it. So as to be storage efficient, the mnemonic names for data source and data type are not used in the DDS packet header, but instead an 8 bit number is used to indicate each combination. This number is shown in the third column of the table.

All data delivered has a corresponding data description registered at ESOC, and delivered on each CD-ROM. The identifier of these data descriptions (called the ADID) is shown in the fifth column for each of the available data streams.

Note: There are no separate data streams for ASPOC normal science data and burst science data. This is due to the fact that all ASPOC data, both science and housekeeping, are all delivered within the housekeeping packets for the ASPOC experiment. Therefore there are no separate data descriptions for ASPOC NSD and BSD data as all the ASPOC data is described in the HKD description.

Note: The packet header IDs given for ADIDs ECLUD001 to ECLUD006 will never be used in a DDS header. They are there purely to maintain the one-to-one correspondence between packet header ID and ADID.

Data Source	Data Type	Packet header ID (8 bits)	Data Description Title	ADID
-	-	20	Cluster CD-ROM directory tree listing format	ECLUD001
-	-	21	Cluster CD-ROM cumulative index file format	ECLUD002
-	-	22	Catalogue format for Cluster off-line delivery	ECLUD003
-	-	23	Catalogue format for Cluster on-line delivery	ECLUD004
-	-	24	Acknowledgement format for Cluster on-line delivery	ECLUD005
-	-	25	Volume production information format for Cluster off-line delivery	ECLUD006
MASTER	CAT	0	Master catalogue of all Cluster data available	ECLUD007
LTOF	AUX	1	Long Term Orbit File Entry	ECLUA001
LTEF	AUX	2	Long Term Event File Entry	ECLUA002
STOF	AUX	3	Short Term Orbit File Entry	ECLUA003
STEF	AUX	4	Short Term Event File Entry	ECLUA004
SATT	AUX	5	Spacecraft Attitude and Spin Rate Entry	ECLUA005
TCAL	AUX	6	Time Calibration File Entry	ECLUA006
CMDH	AUX	7	Command History File Entry	ECLUA007
COVM	AUX	8	Covariance Matrix File Entry	
EDI	HPD	200	Housekeeping Parameter Definition File for EDI on Cluster 1	ECLUP101
FGM	HPD	201	Housekeeping Parameter Definition File for FGM on Cluster 1	ECLUP102
CIS	HPD	202	Housekeeping Parameter Definition File for CIS on Cluster 1	ECLUP103
PEACE	HPD	203	Housekeeping Parameter Definition File for PEACE on Cluster 1	ECLUP104
RAPID	HPD	204	Housekeeping Parameter Definition File for RAPID on Cluster 1	ECLUP105
WEC	HPD	205	Housekeeping Parameter Definition File for WEC on Cluster 1	ECLUP106
ASPOC	HPD	206	Housekeeping Parameter Definition File for ASPOC on Cluster 1	ECLUP107
SC	HPD	207	Housekeeping Parameter Definition File for Cluster 1 Platform	ECLUP108
EDI	HPD	208	Housekeeping Parameter Definition File for EDI on Cluster 2	ECLUP201
FGM	HPD	209	Housekeeping Parameter Definition File for FGM on Cluster 2	ECLUP202
CIS	HPD	210	Housekeeping Parameter Definition File for CIS on Cluster 2	ECLUP203
PEACE	HPD	211	Housekeeping Parameter Definition File for PEACE on Cluster 2	ECLUP204
RAPID	HPD	212	Housekeeping Parameter Definition File for RAPID on Cluster 2	ECLUP205
WEC	HPD	213	Housekeeping Parameter Definition File for WEC on Cluster 2	ECLUP206
ASPOC	HPD	214	Housekeeping Parameter Definition File for ASPOC on Cluster 2	ECLUP207
SC	HPD	215	Housekeeping Parameter Definition File for Cluster 2 Platform	ECLUP208
EDI	HPD	216	Housekeeping Parameter Definition File for EDI on Cluster 3	ECLUP301
FGM	HPD	217	Housekeeping Parameter Definition File for FGM on Cluster 3	ECLUP302
CIS	HPD	218	Housekeeping Parameter Definition File for CIS on Cluster 3	ECLUP303

Data Source	Data Type	Packet header ID (8 bits)	Data Description Title	ADID
PEACE	HPD	219	Housekeeping Parameter Definition File for PEACE on Cluster 3	ECLUP304
RAPID	HPD	220	Housekeeping Parameter Definition File for RAPID on Cluster 3	ECLUP305
WEC	HPD	221	Housekeeping Parameter Definition File for WEC on Cluster 3	ECLUP306
ASPOC	HPD	222	Housekeeping Parameter Definition File for ASPOC on Cluster 3	ECLUP307
SC	HPD	223	Housekeeping Parameter Definition File for Cluster 3 Platform	ECLUP308
EDI	HPD	224	Housekeeping Parameter Definition File for EDI on Cluster 4	ECLUP401
FGM	HPD	225	Housekeeping Parameter Definition File for FGM on Cluster 4	ECLUP402
CIS	HPD	226	Housekeeping Parameter Definition File for CIS on Cluster 4	ECLUP403
PEACE	HPD	227	Housekeeping Parameter Definition File for PEACE on Cluster 4	ECLUP404
RAPID	HPD	228	Housekeeping Parameter Definition File for RAPID on Cluster 4	ECLUP405
WEC	HPD	229	Housekeeping Parameter Definition File for WEC on Cluster 4	ECLUP406
ASPOC	HPD	230	Housekeeping Parameter Definition File for ASPOC on Cluster 4	ECLUP407
SC	HPD	231	Housekeeping Parameter Definition File for Cluster 4 Platform	ECLUP408
EDI	NSD	30	Normal science data for EDI on Cluster 1	ECLUN101
FGM	NSD	31	Normal science data for FGM on Cluster 1	ECLUN102
CIS	NSD	32	Normal science data for CIS on Cluster 1	ECLUN103
PEACE	NSD	33	Normal science data for PEACE on Cluster 1	ECLUN104
RAPID	NSD	34	Normal science data for RAPID on Cluster 1	ECLUN105
WEC	NSD	35	Normal science data for WEC on Cluster 1	ECLUN106
EDI	BSD	37	Burst science data for EDI on Cluster 1	ECLUB101
FGM	BSD	38	Burst science data for FGM on Cluster 1	ECLUB102
CIS	BSD	39	Burst science data for CIS on Cluster 1	ECLUB103
PEACE	BSD	40	Burst science data for PEACE on Cluster 1	ECLUB104
RAPID	BSD	41	Burst science data for RAPID on Cluster 1	ECLUB105
WEC	BSD	42	Burst science data for WEC on Cluster 1	ECLUB106
EDI	HKD	44	Housekeeping data for EDI on Cluster 1	ECLUH101
FGM	HKD	45	Housekeeping data for FGM on Cluster 1	ECLUH102
CIS	HKD	46	Housekeeping data for CIS on Cluster 1	ECLUH103
PEACE	HKD	47	Housekeeping data for PEACE on Cluster 1	ECLUH104
RAPID	HKD	48	Housekeeping data for RAPID on Cluster 1	ECLUH105

Data Source	Data Type	Packet header ID (8 bits)	Data Description Title	ADID
WEC	HKD	49	Housekeeping data for WEC on Cluster 1	ECLUH106
ASPOC	HKD	50	Housekeeping data for ASPOC on Cluster 1	ECLUH107
SC	HKD	51	Housekeeping data for Cluster 1 Platform	ECLUH108
EDI	NSD	70	Normal science data for EDI on Cluster 2	ECLUN201
FGM	NSD	71	Normal science data for FGM on Cluster 2	ECLUN202
CIS	NSD	72	Normal science data for CIS on Cluster 2	ECLUN203
PEACE	NSD	73	Normal science data for PEACE on Cluster 2	ECLUN204
RAPID	NSD	74	Normal science data for RAPID on Cluster 2	ECLUN205
WEC	NSD	75	Normal science data for WEC on Cluster 2	ECLUN206
EDI	BSD	77	Burst science data for EDI on Cluster 2	ECLUB201
FGM	BSD	78	Burst science data for FGM on Cluster 2	ECLUB202
CIS	BSD	79	Burst science data for CIS on Cluster 2	ECLUB203
PEACE	BSD	80	Burst science data for PEACE on Cluster 2	ECLUB204
RAPID	BSD	81	Burst science data for RAPID on Cluster 2	ECLUB205
WEC	BSD	82	Burst science data for WEC on Cluster 2	ECLUB206
EDI	HKD	84	Housekeeping data for EDI on Cluster 2	ECLUH201
FGM	HKD	85	Housekeeping data for FGM on Cluster 2	ECLUH202
CIS	HKD	86	Housekeeping data for CIS on Cluster 2	ECLUH203
PEACE	HKD	87	Housekeeping data for PEACE on Cluster 2	ECLUH204
RAPID	HKD	88	Housekeeping data for RAPID on Cluster 2	ECLUH205
WEC	HKD	89	Housekeeping data for WEC on Cluster 2	ECLUH206
ASPOC	HKD	90	Housekeeping data for ASPOC on Cluster 2	ECLUH207
SC	HKD	91	Housekeeping data for Cluster 2 Platform	ECLUH208
EDI	NSD	110	Normal science data for EDI on Cluster 3	ECLUN301
FGM	NSD	111	Normal science data for FGM on Cluster 3	ECLUN302
CIS	NSD	112	Normal science data for CIS on Cluster 3	ECLUN303
PEACE	NSD	113	Normal science data for PEACE on Cluster 3	ECLUN304
RAPID	NSD	114	Normal science data for RAPID on Cluster 3	ECLUN305
WEC	NSD	115	Normal science data for WEC on Cluster 3	ECLUN306

Data Source	Data Type	Packet header ID (8 bits)	Data Description Title	ADID
EDI	BSD	117	Burst science data for EDI on Cluster 3	ECLUB301
FGM	BSD	118	Burst science data for FGM on Cluster 3	ECLUB302
CIS	BSD	119	Burst science data for CIS on Cluster 3	ECLUB303
PEACE	BSD	120	Burst science data for PEACE on Cluster 3	ECLUB304
RAPID	BSD	121	Burst science data for RAPID on Cluster 3	ECLUB305
WEC	BSD	122	Burst science data for WEC on Cluster 3	ECLUB306
EDI	HKD	124	Housekeeping data for EDI on Cluster 3	ECLUH301
FGM	HKD	125	Housekeeping data for FGM on Cluster 3	ECLUH302
CIS	HKD	126	Housekeeping data for CIS on Cluster 3	ECLUH303
PEACE	HKD	127	Housekeeping data for PEACE on Cluster 3	ECLUH304
RAPID	HKD	128	Housekeeping data for RAPID on Cluster 3	ECLUH305
WEC	HKD	129	Housekeeping data for WEC on Cluster 3	ECLUH306
ASPOC	HKD	130	Housekeeping data for ASPOC on Cluster 3	ECLUH307
SC	HKD	131	Housekeeping data for Cluster 3 Platform	ECLUH308
EDI	NSD	150	Normal science data for EDI on Cluster 4	ECLUN401
FGM	NSD	151	Normal science data for FGM on Cluster 4	ECLUN402
CIS	NSD	152	Normal science data for CIS on Cluster 4	ECLUN403
PEACE	NSD	153	Normal science data for PEACE on Cluster 4	ECLUN404
RAPID	NSD	154	Normal science data for RAPID on Cluster 4	ECLUN405
WEC	NSD	155	Normal science data for WEC on Cluster 4	ECLUN406
EDI	BSD	157	Burst science data for EDI on Cluster 4	ECLUB401
FGM	BSD	158	Burst science data for FGM on Cluster 4	ECLUB402
CIS	BSD	159	Burst science data for CIS on Cluster 4	ECLUB403
PEACE	BSD	160	Burst science data for PEACE on Cluster 4	ECLUB404
RAPID	BSD	161	Burst science data for RAPID on Cluster 4	ECLUB405
WEC	BSD	162	Burst science data for WEC on Cluster 4	ECLUB406
EDI	HKD	164	Housekeeping data for EDI on Cluster 4	ECLUH401
FGM	HKD	165	Housekeeping data for FGM on Cluster 4	ECLUH402
CIS	HKD	166	Housekeeping data for CIS on Cluster 4	ECLUH403
PEACE	HKD	167	Housekeeping data for PEACE on Cluster 4	ECLUH404

Data Source	Data Type	Packet header ID (8 bits)	Data Description Title	ADID
RAPID	HKD	168	Housekeeping data for RAPID on Cluster 4	ECLUH405
WEC	HKD	169	Housekeeping data for WEC on Cluster 4	ECLUH406
ASPOC	HKD	170	Housekeeping data for ASPOC on Cluster 4	ECLUH407
SC	HKD	171	Housekeeping data for Cluster 4 Platform	ECLUH408

Table A-1: Data Stream ADID Summary

Appendix B CLUSTER DDS Error Handling

B.1 CLUSTER DDS Generated Error Messages

This appendix lists all the possible error messages which can be returned by the "**ERROR_MESSAGE**" parameter in an acknowledgement. They are all given as an ASCII string and are of the form:

CLUSTER DDS ERROR-nn: <String>

In the following list the error number, nn, is given and the error message.

NO ERROR

```
CLUSTER DDS ERROR-01: Illegal command specified.
CLUSTER DDS ERROR-02: Illegal target filename specified.
CLUSTER DDS ERROR-03: Illegal qualifier specified.
CLUSTER DDS ERROR-04: Too many parameters specified.
CLUSTER DDS ERROR-05: Illegal /BEFORE date/time format.
CLUSTER DDS ERROR-06: Illegal /SINCE date/time format.
CLUSTER DDS ERROR-07: /SINCE time greater than /BEFORE time.
CLUSTER DDS ERROR-08: Too many /BEFORE qualifiers specified.
CLUSTER DDS ERROR-09: Too many /SINCE qualifiers specified.
CLUSTER DDS ERROR-10: Unrecognised data source.
CLUSTER DDS ERROR-11: Unrecognised data type.
CLUSTER DDS ERROR-12: Illegal combination of data source/type.
CLUSTER DDS ERROR-13: Request would exceed permitted daily quota.
CLUSTER DDS ERROR-14: Maximum line length exceeded.
CLUSTER DDS ERROR-15: Maximum number of outstanding files exceeded.
CLUSTER DDS ERROR-16: No catalogue data within time requested.
CLUSTER DDS ERROR-17: No data packets available within time requested.
CLUSTER DDS ERROR-18: System timeout, try again later.
```

```
CLUSTER DDS ERROR-19: System unavailable, try again later.
CLUSTER DDS ERROR-20: Too many /AMOUNT qualifiers specified.
CLUSTER DDS ERROR-21: Illegal /AMOUNT value specified.
CLUSTER DDS ERROR-22: Illegal /SC= qualifier specified.
CLUSTER DDS ERROR-23: No data SOURCE.TYPE specified.
CLUSTER DDS ERROR-24: No target file name specified.
CLUSTER DDS ERROR-25: System resources exceeded, try again later.
CLUSTER DDS ERROR-26: System error occurred, try again later.
CLUSTER DDS ERROR-27: Request would exceed permitted system daily quota.
CLUSTER DDS ERROR-28: No access rights to requested data SOURCE.TYPE.
CLUSTER DDS ERROR-29: DDS access disabled.
```

B.2 : Contact Points in Case of Problems

This appendix defines the contact points and procedure for the PIs to follow in the case of problems with the CDDS. The line of reporting is shown in Figure B-1

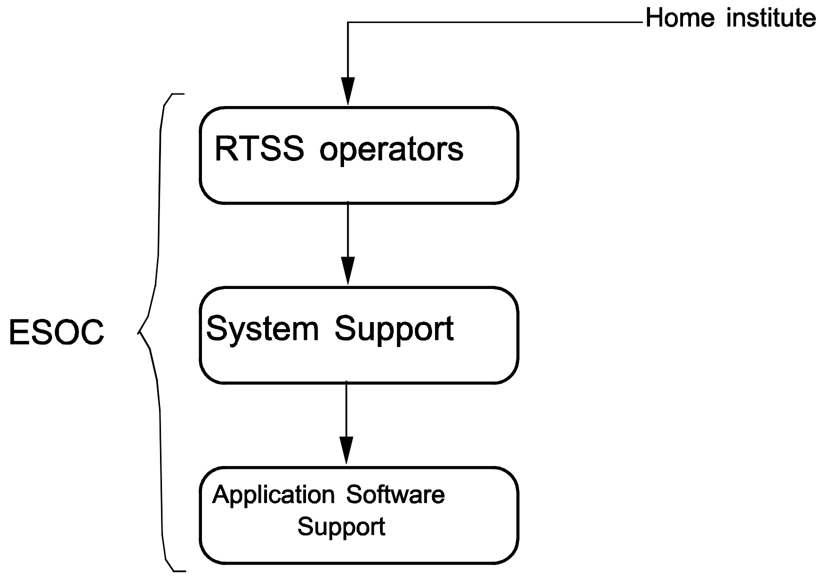


Figure B-1 :PI Contact Points in Case of CDDS Problems

If a PI has a problem with the CDDS which requires human contact with ESOC then they should telephone the RTSS operators. Information given to the RTSS operators must include the nature of the failure, probability of location of fault, time of occurrence, priority associated to the transmission of the data, etc. The RTSS operators will escalate the problem, if required, to System Support (CS) or Application

Support (DPD). ESOC will endeavour to respond to the failure within 1 hour.

The contact point at ESOC is:	RTSS Operators. Telephone extension 2249 or internationally (+49)-6151-902249. Fax number (+49)-6151-903061

Note: Fax may be used but it must be previously announced by a telephone call.

Appendix C : PI Directories

This appendix lists all the PIs and their corresponding path name of the directory that each PI has access to deposit the REQUEST.DDS files into on CDDS computer at ESOC. The directory path name specified in the third column of this table is the full directory name as far as the PI is concerned. The system will be set up at ESOC so that if for example the ASPOC PI deposits a file at the location ESOCNODE::ASPOC, then it will be in the correct directory**.**

Instrument	Principle investigator	CDDS directory
ASPOC	W. Riedler, Institut für Weltraumforschung, Graz, A	ASPOC
EDI	G. Paschmann, MPI für Extraterrestrische Physik, Garching, D	EDI
FGM	A. Balogh, Imperial College, London UK	FGM
CIS	H. Rème, Centre d'Etude Spatial des Rayonement, Toulouse, F	CIS
PEACE	A. Fazakerley, Mullard Space Science Laboratory, Holmbury, St. Mary, UK	PEACE
RAPID	B. Wilken, MPI für Aeronomie, Lindau/Harz, D	RAPID
WEC/STAFF	N. Cornilleau-Wehrlin, Centre de Recherche en Physique de l'Environment Terrestre et Planetaire, Velizy, F	STAFF
WEC/EFW	G. Gustafsson, Swedish Institute of Space Physics, Uppsala, S	EFW
WEC/WHISPER	P.M.E. Décréau, Laboratoire de Physique et Chemie de l'Environment, Orléans, F	WHISPER
WEC/WBD	D.A.Gurnett, University of Iowa, USA	WBD
WEC/DWP	H. Alleyne, University of Sheffield, UK	DWP
-	T. Dimbylow, Rutherford Appleton Laboratory, Oxford, UK	JSOC

Table C-1:PI Directories on the CDDS

Appendix D : Full CD-ROM Directory and File listing

This appendix list all files that shall be delivered on the CD-ROM to each appropriate CLUSTER science community member. For each file the full directory path name is also included.

VOLDESC.SFD
9409011A.IDX
9409011A.LSC
9409011A.LST

CLUSTER1/AUX_1/940901BA.1A1
CLUSTER1/AUX_1/940901GA.1A1
CLUSTER1/AUX_1/940901IA.1A1
CLUSTER1/AUX_1/940901LA.1A1
CLUSTER1/AUX_1/940901TA.1A1

CLUSTER1/BSD_1/940901CB.1A1
CLUSTER1/BSD_1/940901EB.1A1
CLUSTER1/BSD_1/940901FB.1A1
CLUSTER1/BSD_1/940901PB.1A1
CLUSTER1/BSD_1/940901RB.1A1
CLUSTER1/BSD_1/940901WB.1A1

CLUSTER1/HKD_1/940901AH.1A1
CLUSTER1/HKD_1/940901CH.1A1
CLUSTER1/HKD_1/940901EH.1A1
CLUSTER1/HKD_1/940901FH.1A1
CLUSTER1/HKD_1/940901PH.1A1
CLUSTER1/HKD_1/940901RH.1A1
CLUSTER1/HKD_1/940901SH.1A1
CLUSTER1/HKD_1/940901WH.1A1

CLUSTER1/HPD_1/940901AD.1A1
CLUSTER1/HPD_1/940901CD.1A1
CLUSTER1/HPD_1/940901ED.1A1
CLUSTER1/HPD_1/940901FD.1A1
CLUSTER1/HPD_1/940901PD.1A1
CLUSTER1/HPD_1/940901RD.1A1
CLUSTER1/HPD_1/940901SD.1A1
CLUSTER1/HPD_1/940901WD.1A1

CLUSTER1/NSD_1/940901CN.1A1
CLUSTER1/NSD_1/940901EN.1A1
CLUSTER1/NSD_1/940901FN.1A1
CLUSTER1/NSD_1/940901PN.1A1
CLUSTER1/NSD_1/940901RN.1A1
CLUSTER1/NSD_1/940901WN.1A1

CLUSTER2/AUX_2/94901BA.1A2
CLUSTER2/AUX_2/940901GA.1A2
CLUSTER2/AUX_2/940901IA.1A2
CLUSTER2/AUX_2/940901LA.1A2
CLUSTER2/AUX_2/940901TA.1A2

CLUSTER2/BSD_2/940901CB.1A2
CLUSTER2/BSD_2/940901EB.1A2
CLUSTER2/BSD_2/940901FB.1A2
CLUSTER2/BSD_2/940901PB.1A2
CLUSTER2/BSD_2/940901RB.1A2
CLUSTER2/BSD_2/940901WB.1A2

CLUSTER2/HKD_2/940901AH.1A2
CLUSTER2/HKD_2/940901CH.1A2
CLUSTER2/HKD_2/940901EH.1A2
CLUSTER2/HKD_2/940901FH.1A2
CLUSTER2/HKD_2/940901PH.1A2
CLUSTER2/HKD_2/940901RH.1A2
CLUSTER2/HKD_2/940901SH.1A2
CLUSTER2/HKD_2/94091WH.1A2

CLUSTER2/HPD_2/940901AD.1A2
CLUSTER2/HPD_2/940901CD.1A2
CLUSTER2/HPD_2/940901ED.1A2
CLUSTER2/HPD_2/940901FD.1A2
CLUSTER2/HPD_2/940901PD.1A2
CLUSTER2/HPD_2/940901RD.1A2
CLUSTER2/HPD_2/940901SD.1A2
CLUSTER2/HPD_2/940901WD.1A2

CLUSTER2/NSD_2/940901CN.1A2
CLUSTER2/NSD_2/940901EN.1A2
CLUSTER2/NSD_2/940901FN.1A2
CLUSTER2/NSD_2/940901PN.1A2
CLUSTER2/NSD_2/940901RN.1A2
CLUSTER2/NSD_2/940901WN.1A2

CLUSTER3/AUX_3/940901BA.1A3
CLUSTER3/AUX_3/940901GA.1A3
CLUSTER3/AUX_3/940901IA.1A3
CLUSTER3/AUX_3/940901LA.1A3
CLUSTER3/AUX_3/940901TA.1A3

CLUSTER3/BSD_3/940901CB.1A3
CLUSTER3/BSD_3/940901EB.1A3
CLUSTER3/BSD_3/940901FB.1A3
CLUSTER3/BSD_3/940901PB.1A3
CLUSTER3/BSD_3/940901RB.1A3
CLUSTER3/BSD_3/940901WB.1A3

CLUSTER3/HKD_3/940901AH.1A3
CLUSTER3/HKD_3/940901CH.1A3
CLUSTER3/HKD_3/940901EH.1A3
CLUSTER3/HKD_3/940901FH.1A3
CLUSTER3/HKD_3/940901PH.1A3
CLUSTER3/HKD_3/940901RH.1A3
CLUSTER3/HKD_3/940901SH.1A3
CLUSTER3/HKD_3/940901WH.1A3

CLUSTER3/HPD_3/940901AD.1A3
CLUSTER3/HPD_3/940901CD.1A3
CLUSTER3/HPD_3/940901ED.1A3
CLUSTER3/HPD_3/940901FD.1A3
CLUSTER3/HPD_3/940901PD.1A3
CLUSTER3/HPD_3/940901RD.1A3
CLUSTER3/HPD_3/940901SD.1A3
CLUSTER3/HPD_3/940901WD.1A3

CLUSTER3/NSD_3/940901CN.1A3
CLUSTER3/NSD_3/940901EN.1A3
CLUSTER3/NSD_3/940901FN.1A3
CLUSTER3/NSD_3/940901PN.1A3
CLUSTER3/NSD_3/940901RN.1A3
CLUSTER3/NSD_3/940901WN.1A3

CLUSTER4/AUX_4/940901BA.1A4
CLUSTER4/AUX_4/940901GA.1A4
CLUSTER4/AUX_4/940901IA.1A4
CLUSTER4/AUX_4/940901LA.1A4
CLUSTER4/AUX_4/940901TA.1A4

CLUSTER4/BSD_4/940901CB.1A4
CLUSTER4/BSD_4/940901EB.1A4
CLUSTER4/BSD_4/940901FB.1A4
CLUSTER4/BSD_4/940901PB.1A4
CLUSTER4/BSD_4/940901RB.1A4
CLUSTER4/BSD_4/940901WB.1A4

CLUSTER4/HKD_4/940901AH.1A4
CLUSTER4/HKD_4/940901CH.1A4
CLUSTER4/HKD_4/940901EH.1A4

CLUSTER4/HKD_4/940901FH.1A4
CLUSTER4/HKD_4/940901PH.1A4
CLUSTER4/HKD_4/94091RH.1A4
CLUSTER4/HKD_4/940901SH.1A4
CLUSTER4/HKD_4/940901WH.1A4

CLUSTER4/HPD_4/940901AD.1A4
CLUSTER4/HPD_4/940901CD.1A4
CLUSTER4/HPD_4/940901ED.1A4
CLUSTER4/HPD_4/940901FD.1A4
CLUSTER4/HPD_4/940901PD.1A4
CLUSTER4/HPD_4/940901RD.1A4
CLUSTER4/HPD_4/940901SD.1A4
CLUSTER4/HPD_4/940901WD.1A4

CLUSTER4/NSD_4/940901CN.1A4
CLUSTER4/NSD_4/940901EN.1A4
CLUSTER4/NSD_4/940901FN.1A4
CLUSTER4/NSD_4/940901PN.1A4
CLUSTER4/NSD_4/940901RN.1A4
CLUSTER4/NSD_4/940901WN.1A4

DATADES/AUX_DES/940901BA.1AX
DATADES/AUX_DES/940901GA.1AX
DATADES/AUX_DES/940901IA.1AX
DATADES/AUX_DES/940901LA.1AX
DATADES/AUX_DES/940901TA.1AX

DATADES/CDDS_DES/9409011A.CDD
DATADES/CDDS_DES/9409011A.IDD
DATADES/CDDS_DES/9409011A.LDD
DATADES/CDDS_DES/9409011A.VDD

DATADES/DATADESA/BSD_DESA/940901CB.1AA
DATADES/DATADESA/BSD_DESA/940901EB.1AA
DATADES/DATADESA/BSD_DESA/940901FB.1AA
DATADES/DATADESA/BSD_DESA/940901PB.1AA
DATADES/DATADESA/BSD_DESA/940901RB.1AA
DATADES/DATADESA/BSD_DESA/940901WB.1A
A

DATADES/DATADESA/HKD_DESA/940901AH.1AA
DATADES/DATADESA/HKD_DESA/940901CH.1AA
DATADES/DATADESA/HKD_DESA/940901EH.1AA
DATADES/DATADESA/HKD_DESA/940901FH.1AA
DATADES/DATADESA/HKD_DESA/940901PH.1AA
DATADES/DATADESA/HKD_DESA/940901RH.1AA
DATADES/DATADESA/HKD_DESA/940901SH.1AA
DATADES/DATADESA/HKD_DESA/940901WH.1A
A

DATADES/DATADESA/HPD_DESA/940901AD.1AA
DATADES/DATADESA/HPD_DESA/940901CD.1AA
DATADES/DATADESA/HPD_DESA/940901ED.1AA
DATADES/DATADESA/HPD_DESA/940901FD.1AA
DATADES/DATADESA/HPD_DESA/940901PD.1AA
DATADES/DATADESA/HPD_DESA/940901RD.1AA
DATADES/DATADESA/HPD_DESA/940901SD.1AA
DATADES/DATADESA/HPD_DESA/940901WD.1AA

DATADES/DATADESA/NSD_DESA/940901CN.1AA
DATADES/DATADESA/NSD_DESA/940901EN.1AA
DATADES/DATADESA/NSD_DESA/940901FN.1AA
DATADES/DATADESA/NSD_DESA/940901PN.1AA
DATADES/DATADESA/NSD_DESA/940901RN.1AA
DATADES/DATADESA/NSD_DESA/940901WN.1AA

DATADES/DATADESB/BSD_DESB/940901CB.1AB
DATADES/DATADESB/BSD_DESB/940901EB.1AB
DATADES/DATADESB/BSD_DESB/940901FB.1AB
DATADES/DATADESB/BSD_DESB/940901PB.1AB
DATADES/DATADESB/BSD_DESB/940901RB.1AB
DATADES/DATADESB/BSD_DESB/940901WB.1AB

DATADES/DATADESB/HKD_DESB/940901AH.1AB
DATADES/DATADESB/HKD_DESB/940901CH.1AB
DATADES/DATADESB/HKD_DESB/940901EH.1AB
DATADES/DATADESB/HKD_DESB/940901FH.1AB
DATADES/DATADESB/HKD_DESB/940901PH.1AB
DATADES/DATADESB/HKD_DESB/940901RH.1AB
DATADES/DATADESB/HKD_DESB/940901SH.1AB
DATADES/DATADESB/HKD_DESB/940901WH.1AB

DATADES/DATADESB/HPD_DESB/940901AD.1AB
DATADES/DATADESB/HPD_DESB/940901CD.1AB
DATADES/DATADESB/HPD_DESB/940901ED.1AB
DATADES/DATADESB/HPD_DESB/940901FD.1AB
DATADES/DATADESB/HPD_DESB/940901PD.1AB
DATADES/DATADESB/HPD_DESB/940901RD.1AB
DATADES/DATADESB/HPD_DESB/940901SD.1AB
DATADES/DATADESB/HPD_DESB/940901WD.1AB

DATADES/DATADESB/NSD_DESB/940901CN.1AB
DATADES/DATADESB/NSD_DESB/940901EN.1AB
DATADES/DATADESB/NSD_DESB/940901FN.1AB
DATADES/DATADESB/NSD_DESB/940901PN.1AB
DATADES/DATADESB/NSD_DESB/940901RN.1AB
DATADES/DATADESB/NSD_DESB/940901WN.1AB

DATADES/DATADESC/BSD_DESC/940901CB.1AC
DATADES/DATADESC/BSD_DESC/940901EB.1AC
DATADES/DATADESC/BSD_DESC/940901FB.1AC
DATADES/DATADESC/BSD_DESC/940901PB.1AC
DATADES/DATADESC/BSD_DESC/940901RB.1AC
DATADES/DATADESC/BSD_DESC/940901WB.1AC

DATADES/DATADESC/HKD_DESC/940901AH.1AC
DATADES/DATADESC/HKD_DESC/940901CH.1AC
DATADES/DATADESC/HKD_DESC/940901EH.1AC
DATADES/DATADESC/HKD_DESC/940901FH.1AC
DATADES/DATADESC/HKD_DESC/940901PH.1A
C
DATADES/DATADESC/HKD_DESC/940901RH.1A
C
DATADES/DATADESC/HKD_DESC/940901SH.1A
C
DATADES/DATADESC/HKD_DESC/940901WH.1
AC

DATADES/DATADESC/HPD_DESC/940901AD.1A
C
DATADES/DATADESC/HPD_DESC/940901CD.1A
C
DATADES/DATADESC/HPD_DESC/940901ED.1A
C
DATADES/DATADESC/HPD_DESC/940901FD.1A
C
DATADES/DATADESC/HPD_DESC/940901PD.1A
C
DATADES/DATADESC/HPD_DESC/940901RD.1A
C
DATADES/DATADESC/HPD_DESC/940901SD.1A
C
DATADES/DATADESC/HPD_DESC/940901WD.1
AC

DATADES/DATADESC/NSD_DESC/940901CN.1A
C
DATADES/DATADESC/NSD_DESC/940901EN.1A
C
DATADES/DATADESC/NSD_DESC/940901FN.1A
C
DATADES/DATADESC/NSD_DESC/940901PN.1A
C
DATADES/DATADESC/NSD_DESC/940901RN.1A
C
DATADES/DATADESC/NSD_DESC/940901WN.1
AC

DATADES/DATADESD/BSD_DESD/940901CB.1A
D
DATADES/DATADESD/BSD_DESD/940901EB.1A
D
DATADES/DATADESD/BSD_DESD/940901FB.1A
D
DATADES/DATADESD/BSD_DESD/940901PB.1A
D
DATADES/DATADESD/BSD_DESD/940901RB.1A
D
DATADES/DATADESD/BSD_DESD/940901WB.1
AD

DATADES/DATADESD/HKD_DESD/940901AH.1A
D
DATADES/DATADESD/HKD_DESD/940901CH.1A
D
DATADES/DATADESD/HKD_DESD/940901EH.1A
D
DATADES/DATADESD/HKD_DESD/940901FH.1A
D
DATADES/DATADESD/HKD_DESD/940901PH.1A
D
DATADES/DATADESD/HKD_DESD/940901RH.1A
D
DATADES/DATADESD/HKD_DESD/940901SH.1A

D

DATADES/DATADESD/HKD_DESD/940901WH.1A
D

DATADES/DATADESD/HPD_DESD/940901AD.1A
D

DATADES/DATADESD/HPD_DESD/940901CD.1A
D

DATADES/DATADESD/HPD_DESD/940901ED.1A
D

DATADES/DATADESD/HPD_DESD/940901FD.1A
D

DATADES/DATADESD/HPD_DESD/940901PD.1A
D

DATADES/DATADESD/HPD_DESD/940901RD.1A
D

DATADES/DATADESD/HPD_DESD/940901SD.1A
D

DATADES/DATADESD/HPD_DESD/940901WD.1A
D

DATADES/DATADESD/NSD_DESD/940901CN.1A
D

DATADES/DATADESD/NSD_DESD/940901EN.1A
D

DATADES/DATADESD/NSD_DESD/940901FN.1A
D

DATADES/DATADESD/NSD_DESD/940901PN.1A
D

DATADES/DATADESD/NSD_DESD/940901RN.1A
D

DATADES/DATADESD/NSD_DESD/940901WN.1A
D

SOFTWARE/DJ2000.FOR

SOFTWARE/JD2000.FOR

SOFTWARE/ORBIT.FOR

SOFTWARE/PR2000.FOR

Appendix E : CDDS Auxiliary Data Formats

This appendix outlines the formats of the Cluster auxiliary data provided by ESOC. Not all the auxiliary data files are available both on CD-ROM and on-line. The following table indicates which of the auxiliary files are available by each method. The table also indicates which files contain packets with a span of validity, as opposed to the packets being valid only at a particular point in time. Appendix E.9 explains fully the concept of "span valid" and "point valid" files.

File	App.	Avail. On-Line	On CD-ROM	Span Valid
Long Term Orbit File (LTOF)	E.1	Yes	No	Yes
Short Term Orbit File (STOF)	E.2	Yes	Yes	Yes
Long Term Event File (LTEF)	E.3	Yes	No	No
Short Term Event File (STEF)	E.4	Yes	Yes	No
Covariance Matrix File (COVN)	E.10	Yes	Yes	Yes
Spacecraft Attitude and Spin Rates (SATT)	E.5	Yes	Yes	Yes
Time Calibration File (TCAL)	E.6	Yes	Yes	Yes
Command History File (CMDH)	E.7	Yes	Yes	No
Housekeeping Parameter Definition (HPD)	E.8	Yes	Yes	Yes

Table E-1Auxiliary Data Availability

For the on-line delivery, auxiliary data is requested using the same mechanism as for telemetry data. The various files of auxiliary data available are also listed in Appendix A together with their corresponding data source/type mnemonics and ADID. They are referenced in the same manner as telemetry data streams, i.e., with a data source and data type mnemonic (the type always being AUX **or** HPD**).**

The following appendices each define the format of a single entry within any one auxiliary file, i.e., an orbit definition, an event, a time calibration, a spacecraft telecommand, etc.

Each auxiliary data entry is treated the same as an experiment packet from on-board the spacecraft, therefore it will also have the same DDS packet header as that depicted in Table 5-3, except that the data stream and TASI fields will be set to the "N/A" (not applicable) value. The time quality field will be set to actual time. The ground station field in the DDS packet header will also be set to the "N/A" value for all auxiliary files, apart from the command history file, where it will correspond to the ground station field in the packet data.

The catalogue entry for an auxiliary data source is the same as for the telemetry data sources.

E.1 : Long Term Orbit File (LTOF)

The LTOF will contain orbital state vectors for the remaining, a maximum of 4, major phases of the Cluster mission, that is, Cusp 1, Tail 1, Cusp 2 and Tail 2. There will be only one file for each spacecraft, covering the remaining phases. The file will be generated after each constellation manoeuvre phase.

The file is not intended for direct interpretation, but via access by the Fortran routine supplied (see Appendix F.5.1).

The content and format of the file is defined in the following tables. The shaded rows define the format and are followed by the parameter names in an unshaded row. The format definition in brackets follows the ANSI FORTRAN notation for format statements (e.g. A29 means 29 ASCII characters, 5X means 5 spaces, I2 means a 2 character denary integer and F7.2 means a 7 character fixed point number with 2 decimal characters).

For each period of time there is a header, the same as that used for telemetry packets as defined in Table 5-3 followed by a leader block as shown in Table E-2, this is further followed by a number of polynomial coefficient blocks as shown in Table E-3. The number of blocks depending upon the accuracy that is required. The SCET field in the DDS packet header has the same time value as the time in the **SRTTIM** field defined below (although the former is expressed in binary and the latter in ASCII), and this is used as the access key for the file.

S/C/ ID> (I3, X2)	<Predict or recon.> (A1, X2)	<Generation time> (A20, X2)	<Data start time> (A20, X2)	<Data end time> (A20)	LF (A1)
SCID	**PREREC**	**GENTIM**	**SRTTIM**	**ENDTIM**	**LF**

Rec ID> (I3)	<MJD start> (F12.6)	<MJD end> (F12.6)	<MJD orbit> (F15.9)	<Rev. num.> (F11.3)	<Semimajor axis> (F13.5)	<Mean motion> (F13.5)	LF (A1)

NREC	**DAYBEG**	**DAYEND**	**EPOCH**	**ORBIN**	**SMAXIS**	**OMOTIN**	**LF**

Rec ID> (I3)	<x-y-z of position vector> (3F11.3)	<x-y-z of velocity vector> (3F11.7)	<Absolute position> (F11.3)	LF (A1)
NREC	**XYZPOS(3)**	**XYZVEL(3)**	**RDIST**	**LF**

Table E-2 Definition of the Leader Block Within the Long Term Orbit File

Where:

SCID will contain the S/C body identification, this is an integer from 1 to 4 inclusive. The SCID shall be assigned before launch to each physical spacecraft, and remain the same throughout the mission.

PREREC a single character flag indicating if the data is predicted (P) **or reconstituted (R).**

GENTIM date and time when the leader block was written to the file in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

SRTTIM date and time of the start of the period for when the data is valid in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

ENDTIM date and time of the end of the period for when the data is valid in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

LF is a single line-feed character (ASCII $0A_{hex}$).

NREC an internally used record identifier.

DAYBEG Modified Julian Date 2000 (MJD 2000, i.e. the date 0.0 refers to the 1^{st} January 2000 at 0:00:00), of the start of the period for when the data is valid.

DAYEND Modified Julian Date 2000 (MJD 2000), of the end of the period for when the data is valid.

EPOCH Modified Julian Date 2000 (MJD 2000), of the epoch for the reference Kepler orbit.

ORBIN Revolution number for this epoch, counting start at perigee

SMAXIS Semimajor axis 'a', in km, of the reference Kepler orbit·

OMOTIN Inverse mean motion = 'a*sqrt(a/μ)' of the reference Kepler orbit in seconds/rad (μ = central Earth potential).

NREC an internally used record identifier.

XYZPOS(3) are the x-y-z components of the position vector in km of the reference Kepler orbit.

XYZVEL(3) are the x-y-z components of the velocity vector in km/s of the reference Kepler orbit.

RDIST is the absolute value 'r' of the position vector of the reference Kepler orbit in km.

Rec ID> (I3)	<Polynomial coefficient of x-y-z components of position vector> (3F11.3)	<Polynomial coefficients of x-y-z components of velocity vector> (3F11.7)	LF (A1)
RECID	**POLPOS(3)**	**POLVEL(3)**	**LF**

Table E-3 Definition of a Coefficient Block within the Long Term Orbit File

Where: **RECID** an internally used record identifier.

POLPOS(3) are the polynomial coefficients of the x-y-z components of the position vector in km of the reference Kepler orbit.

POLVEL(3) are the polynomial coefficients of the x-y-z components of the velocity vector in km/s of the reference Kepler orbit.

LF is a single line-feed character (ASCII $0A_{hex}$)

Depending upon the accuracy required the number of the coefficient blocks will vary between 0 and 10 inclusive.

The coordinate system is the Inertial Mean Geocentric Equatorial System of year J2000.0, with the x-axis towards the mean vernal equinox, the x-y plane coinciding with the mean equatorial plane and the z-axis toward north. Time and coordinate systems used for orbital operations at ESOC are described in Ref. [13].

E.2 : Short Term Orbit File (STOF)

The format of the short term orbit file is identical to that of the long term orbit file as defined in Appendix E.1 The difference between the files is the resolution and hence accuracy. In accordance with the mission baseline, the accuracy of the orbit determination is to within 5 km, further to this there may be an additional maximum error of 100 metres due to the use of the approximation method. The STOF will provide reconstituted data for the previous 10 days before generation, and 3.5 months of predicted orbit data from the day of generation. This file will also be generated at least once per week. This file will also contain from 0 to 10 (inclusive) coefficient blocks.

E.3 : Long Term Event File (LTEF)

The long term event file includes events that cover the remaining period of the mission from the time that it is generated, at a coarse resolution and accuracy. It shall be produced after each constellation manoeuvre, detailing events until the end of the mission.

The file consists of event entries, one entry per event. The events will be listed in chronological order. For each period of time there is a header, the same as that used for telemetry packets as defined in Table 5-3 followed by the event entry as defined below. The SCET field in the DDS packet header has the same time value as the time in the **EVTTIM** field defined below (although the former is expressed in binary and the latter in ASCII), and this is used as the access key for the file. The overall file consists of a number of these entries. There will be only one file for each spacecraft, covering the remaining phases. The shaded rows define the format and are followed by the parameter names in an unshaded row. The format definition in brackets follows the ANSI FORTRAN notation for format statements (e.g. A29 means 29 ASCII characters, 5X means 5 spaces, I2 means a 2 character denary integer and F7.2 means a 7 character fixed point number with 2 decimal characters).

Day of year> (I3)	<S/C ID> (X2, I2)	<Predict or recon.> (X2, A1)	<Event start time> (X2, A20)	<Duration of event> (X2, A10)	<Description of event> (X2,A40)	LF (A1)
DOY	**SCID**	**PREREC**	**EVTTIM**	**EVTDUR**	**EVTDES**	**LF**

Table E-4 Definition of Entry in Event File

Where:

DOY is the day of the year number.

SCID will contain the S/C body identification, this is an integer from 1 to 4 inclusive. The SCID shall be assigned before launch to each physical spacecraft, and remain the same throughout the mission.

PREREC a single character flag indicating if the data is predicted (P**) or reconstituted (**R**).**

EVTTIM is the start time for the event in CCSDS ASCII time format A (YYYY-MM-DDThh:mm:ssZ**).**

EVTDUR is the length of time of the event in the time subset of the CCSDS ASCII time code format A, except that the hours field can be an integer from 0 to 999 as this represents a time period, i.e. hhh:mm:ssZ. **This field may be blank (filled with spaces) if an event has no meaningful duration just an instance in time.**

EVTDES is a textual description of the event, described in full below.

LF is a single line-feed character (ASCII $0A_{hex}$).

EVTDES is a 40 character field containing a textual description of the event. Characters 1 to 36 contain the event description which in general has a fixed text. The four rightmost characters, 37 to 40, are treated as comments. The possible LTEF events are list in Table E-5 below. In the table the following abbreviations are used for variable fields within the event descriptions:

sss Three character ground station identifier. Possible values are "VIL" (Villafranca), "CAN" (Canberra) or “GDS” (Goldstone).

rrr Number of upcoming revolution. The second spacecraft pair will be reset to the orbit number of the first spacecraft pair at the beginning of the routine orbit of spacecraft pair 2. Three digits, left padded with zeroes.

dddd Duration of penumbra in minutes. Four digits left padded with zeroes.

pppp Percentage of sun occulted. Four digits left padded with zeroes.

Event String 0 1 2 3 4 1234567890123456789012345678901234567890	Duration Value Permitted
sss AOS EL=5	Yes
sss LOS EL=5	No
ECL EARTH START dddd	Yes
ECL EARTH END	No
ECL MOON START pppp	Yes
ECL MOON END	No
START REVOLUTION rrr	No
APOGEE CROSSING	No
35000 KM ASCEND	No
35000 KM DESCEND	No
ORBIT MAN START	Yes
ORBIT MAN END	No
ATTIT MAN START	Yes
ATTIT MAN END	No

Table E-5 List of Possible LTEF Events

An example of an event file is shown in Figure E-1 (note that the binary DDS packet header applied to the start of each event entry is not displayed here as it may consist of unprintable characters).

```
120  3  R  1996-08-25T12:23:42Z  005:11:23Z  ECL EARTH START          0010
120  3  R  1996-08-25T14:13:00Z              35000 KM DESCEND
120  3  R  1996-08-25T17:34:05Z              ECL EARTH END
121  3  R  1996-08-26T00:12:10Z  000:15:45Z  VIL AOS EL=5
121  3  P  1996-08-26T02:23:40Z  000:00:47Z  ATTIT MAN START
121  3  P  1996-08-26T02:24:27Z              ATTIT MAN END
```

Figure E-1 Example LTEF

E.4 : Short Term Event File (STEF)

The format of this file is the same as that for the long term event file, the only difference being the quantity of events, frequency of updates and that there are more types of events reported. This file will be generated at least once per week, with 10 days of reconstituted events for the previous 10 days and predicted events for the following 3.5 months. For the definition of the format of this file and table abbreviations refer to Appendix E-3. The possible events are listed in Table E-6

Event String 0 1 2 3 4 1234567890123456789012345678901234567890	Duration Value Permitted
sss AOS EL=0	Yes
sss AOS EL=5	Yes
sss SEL ANT N	No
sss SEL ANT S	No
sss LOS EL=5	No
sss LOS EL=0	No
sss ANT N>S START	Yes
sss ANT N>S END	No
sss ANT S>N START	Yes
sss ANT S>N END	No
sss SUN/EARTH START	Yes
sss SUN/EARTH END	No
sss LOW RATE START	Yes
sss LOW RATE END	No
ECL EARTH START dddd	Yes
ECL EARTH END	No
ECL MOON START pppp	Yes
ECL MOON END	No
START REVOLUTION rrr	No
APOGEE CROSSING	No
35000 KM ASCEND	No
35000 KM DESCEND	No
ORBIT MAN START	Yes
ORBIT MAN END	No
ATTIT MAN START	Yes
ATTIT MAN END	No

Table E-6 List of Possible STEF Events

E.5 : Spacecraft Attitude and Spin Rates File (SATT)

These spacecraft specific files contain actual and predicted data for the Spacecraft Attitude, including spin axis right ascension and declination and the spin rate and phase information for the specified validity times.

The processing intervals and generation times of this files are determined by the following rules:

1. For every day and for each eclipse, the following coverage period is used:

   a) If no Earth/Moon blinding is present within the first hour of coverage, the first hour of real-time telemetry will be processed.
   b) Else if Earth/Moon blinding is present within the first hour of coverage, the first hour of real-time telemetry without Earth/Moon blinding will be processed.
   c) Else if there is continuous Earth/Moon blinding during this coverage period, the first hour of VC1 dump telemetry without Earth/Moon blinding will be processed.
   d) Else if there is continuous Earth/Moon blinding in real-time telemetry and VC1 dump telemetry, no file will be produced.

2. For each manoeuvre, an interval ΔT of coverage following this manoeuvre will be processed.

The file will contain the mean values obtained within the processing interval.

The coverage of the file will be from the "validity start time" until the next planned manoeuvre. Nominally the file will contain 2 records: one which overwrites the previous days predicted values with the reconstituted data for this day, and a second which predicts the new values for the next day. In case of anomalies or other problems more then 2 records may be included, i.e., the previous 2 days, todays and the next days data.

For each entry there is a header, the same as that used for telemetry packets as defined in Table 5-3 followed by the SATT entry as defined below. The SCET field in the DDS packet header has the same time value as the time in the **VSTTIM** field defined below (although the former is expressed in binary and the latter in ASCII), and this is used as the access key for the file. The content and format of a record in the file is defined in the following table. The shaded rows define the format and are followed by the parameter names in an unshaded row. The format definition in brackets follows the ANSI FORTRAN notation for format statements (eg A29 means 29 characters, 5X means 5 spaces, I2 means a 2 character denary integer and F7.2 means a 7 character fixed point number with 2 decimal characters). All times are in CCSDS ASCII time format A - i.e. yyyy-mm-ddThh:mm:ss.ddZ **and are in UTC.**

S/C ID> (I2,1X)	<Predict or recon.> (A1,1X)	<Validity Start Time> (A20,1X)	<Validity End Time> (A20,1X)	<Right Ascension> (F6.2,1X)	<Declin'n> (F6.2,1X)
SCID	**PREREC**	**VSTTIM**	**VENTIM**	**SPRASC**	**SPDECL**

<Spin Rate> (F9.6,1X)	<Spin Phase at SPR> (F7.3,1X)	<c.o.m. shift> 3(F5.1,1X)	<Spin Axis Tilt Ψ2> (F5.2,1X)	<Spin Axis Tilt Ψ1> (F5.2,1X)	<Gen time> (A20)	LF (A1)
SPRATE	**SCPHAS**	**COMSHF**	**TPSI_2**	**TPSI_1**	**GENTIM**	**LF**

Table E-7: Definition of Entry in an SATT File

Where:

SCID will contain the S/C body identification, this is an integer from 1 to 4 inclusive. The SCID shall be assigned before launch to each physical spacecraft, and remain the same throughout the mission.

PREREC a single character flag indicating if any of the data in the entry is predicted (P**), and therefore will be replaced by definitive data eventually, or reconstituted (**R**) which means the data is definitive and will not change.**

VSTTIM will contain the time when the results start to be valid, in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

VENTIM will contain the time when the results validity ends, in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

SPRASC is the right ascension in degrees of the angular momentum vector in the Inertial Mean Geocentric Equatorial System of year J2000.0 (0^o to 360^o, maximum error of 0.25^o with 95% confidence and assuming 15 rpm of the spacecraft).

SPDECL is the declination in degrees of the angular momentum vector in the Inertial Mean Geocentric Equatorial System of year J2000.0 (-90^o to $+90^o$, maximum error of 0.25^o with 95% confidence and assuming 15 rpm of the spacecraft). The angular momentum vector coincides with the time average maximum principal inertia axis. When there is no nutation the angular momentum vector is in the same direction as the maximum principal axis.

SPRATE is the mean spin rate in rpm, only as a reference since instantaneous spin rate will be obtained from the Sun Reference Pulses in the telemetry.

SCPHAS is the spacecraft phase in degrees, at Sun Reference Pulse, including Sun Sensor elevation misalignment, spin axis tilt and other predictable delay (0^o to 360^o, maximum error of 0.25^o with 95% confidence and assuming 15 rpm of the spacecraft).

COMSHF is the centre of mass position, expressed in the Cluster body build frame (mm) (X_B, Y_B, Z_B).

TPSI_2 is the First Euler angle to rotate around Z_B axis (Y_A=2) to obtain the maximum principal inertia axis (in degrees).

TPSI_1 is the Second Euler angle to rotate around rotated Y_B axis (Y_A=1) to obtain the maximum principal inertia axis (in degrees).

GENTIM this will contain the time when the entry is generated, in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

LF is a single line-feed character (ASCII $0A_{hex}$).

An example of a Spacecraft Attitude and Spin Data File is shown below (the lines wrap in the figure, in reality there are only 2 lines):

```
1 R 1996-08-25T12:35:25Z 1996-08-26T16:15:31Z  273.45 -43.56 14.456789 ...
        ...333.723 777.9  -2.1   1.3  0.02 -0.05 1996-08-26T16:25:31Z
1 P 1996-08-26T16:15:31Z 1996-10-12T16:00:00Z  273.50 -43.55 14.457654 ...
        ...333.723 777.9  -2.1   1.3  0.02 -0.05 1996-08-26T16:25:31Z
```

Figure E-2 Example SATT File

E.6 : Time Calibration File (TCAL)

This file contains the relation between the SCET (Spacecraft Event Time) and the OBT (On-board Time Code), as calculated by the OCC in real-time. Since the OCC uses a linear fit algorithm, this file need only supply the triplet (OBT, SCET, TICK) on each activation of the time correlation function at the OCC. Upon each calibration there will be a discontinuity in the time calibration line; the reason that a time calibration is performed is so as to put that line back on the correct track. A time calibration value is valid until there is a new entry in the file with a newer OBT value.

For each time calibration entry there is a header, the same as that used for telemetry packets as defined in Table 5-3, followed by the time calibration entry as defined below. The SCET field in the DDS packet header is used as the access key for the file. The content and format of time calibration file entry is defined in the following table.

Field	Type and Length	Octet numbers	Description
OBT	56 bit integer	0-6	On-board time (as a binary counter)
TICK	64 bit integer	7-14	Tick value (as an integer number of pico-seconds)

Table E-8 Time Calibration File Format

E.7 : Command History File (CMDH)

For each command entry there is a DDS packet header, the same as that used for telemetry packets as defined in Table 5-3 followed by the command definition as defined below. The SCET field in the DDS packet header has the same time value as the execution time defined below (although the former is expressed in binary and the latter in ASCII), and this is used as the access key for the file. The ground station field in the DDS packet header corresponds to the ground station field in the CMDH packet data. The content and format of a command definition in the file is defined in the following table.

Byte	Len	Field Name	Description/Units

Byte	Len	Field Name	Description/Units
0-23	24	Execution Time	Format: **YYYY-MM-DDThh:mm:ss.dddZ**
24	1	Field separator	Space character
25-48	24	Uplink Time	Format: **YYYY-MM-DDThh:mm:ss.dddZ**
49	1	Field separator	Space character
50-58	9	Command Mnemonic	A text string
59	1	Field separator	Space character
60-68	9	Command Sequence Name	A text string
69	1	Field separator	Space character
70	1	Command Category	Possible values are: **D** for immediate command **T** for time-tagged command
71	1	Field separator	Space character
72	1	Command Source	Possible values are: **M** for manual stack **S** for command schedule **X** for macro-command expansion
73	1	Field separator	Space character
74-75	2	Ground Station ID	Possible values are: ** for Unknown **V1** for Villafranca-1 **V2** for Villafranca-2 **KA** for Kiruna **RE** for Redu **KO** for Kourou **PE** for Perth **MA** for Malindi **CA** for Canberra **GO** for Goldstone

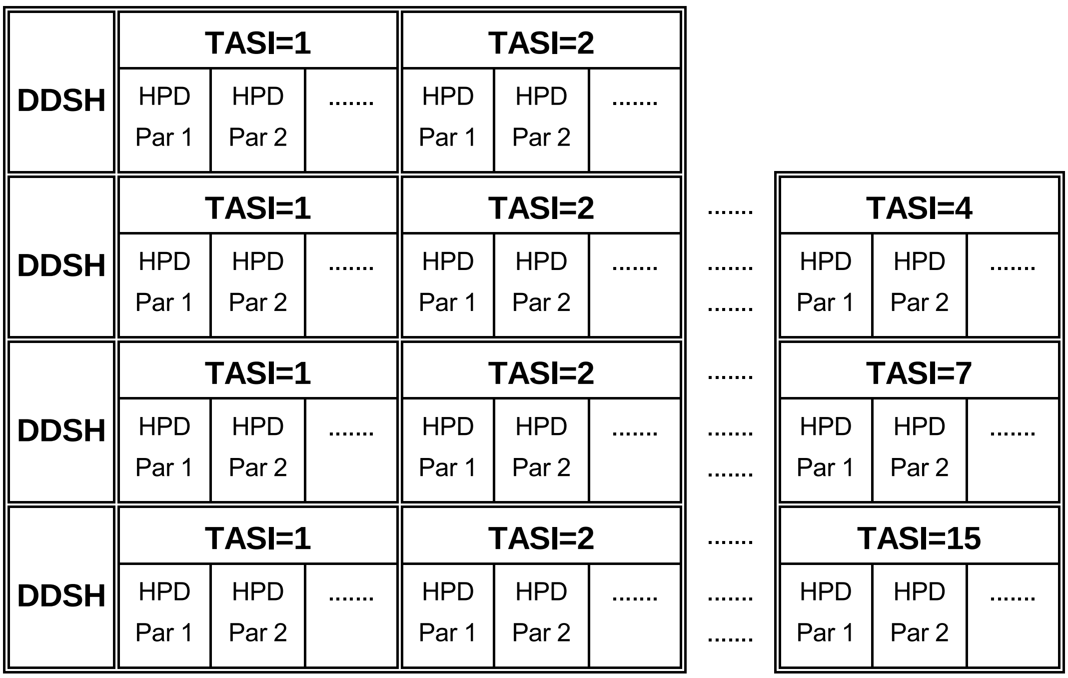
Byte	Len	Field Name	Description/Units
76	1	Field separator	Space character
77	1	Command Execution Verification	Possible values are: **0** for pending execution verification **1** for success **2** for indeterminate (i.e. no CEV in database; globally disabled; superseded in all datastream; deleted by a delete from on-board buffer command) **3** for failure (i.e. CEV failed; failure before CEV, e.g. uplink verification)
78	1	Field separator	Space character
79-94	16	Command Description	A text string
95	1	Field separator	Space character
96-97	2	Number of Parameters in Command	A 2 character ASCII decimal number that indicates how many parameters follow
98	1	Field separator	Space character
99-n	m	Command Parameter Data	For each parameter, its name, value and units are provided. They are of variable length, separated by a comma and terminated by a semicolon, e.g. **PAR1,15,Volt;PAR2,OFF,;**
n+1	1	Record terminator	A line feed character ($0A_{hex}$) to terminate the record.

Table E-9 Command History File Format

E.8 : Housekeeping Parameter Definition File (HPD)

The Housekeeping Parameter Definition data is exported from the operational telemetry database. It describes the format of this database export and is applicable for all experiment and platform housekeeping data.

Each entry in a housekeeping parameter definition file has two parts: firstly the standard DDS packet header, the same as that used for telemetry packets as defined in Table 5-3 followed by a number (the number depending upon the particular housekeeping stream) of parameter definition entries. The timestamp of the DDS packet header is the time that the housekeeping parameter definitions are valid from, until there is a newer entry. Following the DDS packet header, there are the housekeeping parameter definition entries for the Telemetry Acquisition Sequence IDs (TASIs) which were defined in the database at the time of the last export, in TASI order. The file structure is shown schematically below:



where: DDSH = DDS Packet Header
HPD = Housekeeping Parameter Definition

Figure E-3 Schematic Structure of Housekeeping Parameter Definition File

Each housekeeping parameter definition (HPD) is defined in the table below. The shaded row defines the formats of the parameter attributes named in the unshaded row. The format definition, in brackets follows the ANSI FORTRAN notation for format statements (e.g. A29 means 29 characters, 5X means 5 spaces, I2 means a 2 character denary integer and F7.2 means a 7 character fixed point number with 2 decimal characters).

TASI (A3,1X)	Mnemo'c (A6,1X)X	Descript'n (A16,1X)	Units (A4,1X)	Catagory (A1,1X)	Coding (A1,1X)	Width (A3,1X)	...
TASI	**MNEM**	**DESCR**	**UNIT**	**CATEG**	**CODIN**	**WIDTH**	**...**

	Byte (A5,1X)	Bit (A1,1X)	Occurance (A4,1X)	Groups (A4,1X)	Bits/gp (A5,1X)	Bytes/gp (A5,1X)	Cal. info (Ax)	LF (A1)
...	**OFFBY**	**OFFBI**	**NBOCC**	**NBGRP**	**LGOCC**	**LGGRP**	**CALINF**	**LF**

Table E-10: Format of a Housekeeping Parameter Definition

Where:

TASI is the Telemetry Acquisition Sequence Identifier. It is an integer from 0 to 15 inclusive.

MNEM is the short mnemonic name given to the parameter. It is an ASCII string.

DESCR is a short English description of the parameter meaning. It is an ASCII string.

UNIT are the engineering units of the parameter after calibration. It is an ASCII string (e.g. 'Volt').

CATEG is the parameter category. It is an ASCII character. Permissable values are:

- **N** = Numerical
- **S** = Status
- **T** = Text string

CODIN is the coding of the parameter. It is an ASCII character. Permissable values are:

- **A** = ASCII characters
- **B** = Bit pattern
- **I** = Signed integer value
- **U** = Unsigned integer value

0-9 = Real value ('0' = VAX real, others defined by mission)

WIDTH is the size of the parameter. It depends on the field **CATEG** such that, for:

CATEG = **N** the width in bits (integer from 1 to 32 inclusive).

CATEG = **S** the width in bits (integer from 1 to 8 inclusive).

CATEG = **T** the width in bytes (integer from 1 to 255 inclusive).

OFFBY is the location of the first occurrence of the parameter (or group of parameters) within the packet, in bytes, relative to the end of the DDS header (first byte after DDS header is byte 0). It is an integer from 0 to 65535 inclusive.

OFFBI is the starting bit of the parameter (or group of parameters) within the starting byte (first bit is bit 0). It is an integer from 0 to 7 inclusive.

NBOCC is the number of occurrences of the parameter in a group (if any). It is an integer from 1 to 9999 inclusive (equal to 1 if the parameter is simply-supercommutated).

NBGRP is the number of groups in a packet. It is an integer from 1 to 9999 inclusive (greater than 1 if the parameter is simply-supercommutated).

LGOCC is the number of bits between two parameter occurrences within a group. It is an integer from 0 to 32767 inclusive.

LGGRP is the number of bytes between two successive occurrences of groups. It is an integer from 0 to 32767 inclusive.

CALINF is the calibration information that may be necessary for the parameter. This may be either pairs of calibration points or pairs of enumerated status values depending on the field **CATEG**. It is an ASCII string of variable length depending on the number of calibration elements. This field may be empty.

For **CATEG** = **N**, calibration points pairs can be specified. Each pair is enclosed in brackets and separated from the next pair by a comma. The X and Y components of each pair are separated by a comma. There may be up to 32 calibration point pairs. Example: **(X1,Y1),(X2,Y2),(X3,Y3)**..., where the X's and the Y's are numerical values, that is, **(0,10),(3,20),(5,30)**....

For **CATEG** = **S**, enumerated status value pairs can be specified. Each pair is separated from the next pair by a comma. The A and B components of each pair are separated by an equal sign. There may be up to 255 enumerated status value pairs. Example: **A1=B1,A2=B2,A3=B3**..., where the As' are either a numerical value or a range (e.g. 1-5) and the B's are text strings (which could also represent a numeric value), that is, **0=OFF,1-5=Nominal,6=STOP....**

LF is a line feed character (ASCII 0Ah) to indicate the end of the calibration information field and as a separater from the next parameter definition.

To fully understand super-commutated parameters, consider the following definitions and example. Let **P** be a parameter contained in a packet (a packet is identified by the box in the pictures below), the following three cases are possible:

1. P is non supercommutated. It appears only once in the packet:

P

2. P is simply-supercommutated. It appears several time in the packet but as a single occurrence each time:

P	**P**	**P**

3. P is supercommutated. It appears several time in the packet with N occurrence each time (N=3 in the example below). The N occurrences are called "groups":

PPP	**PPP**	**PPP**	**PPP**

The figure below explains the terminology used for supercommutated parameters:

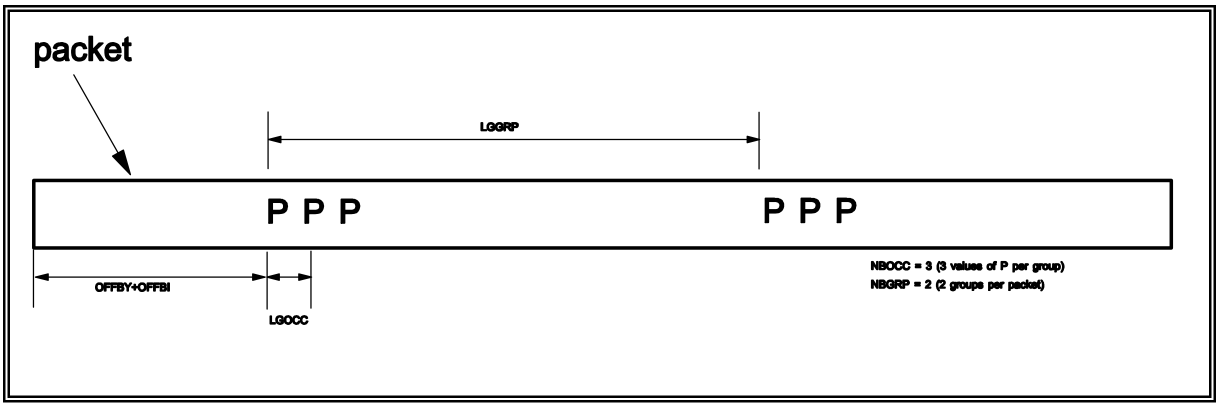


Figure E-4: Schematic Example of Supercommutated Parameters

E.9 : Span Valid and Point Valid Auxiliary and HPD File Handling

Table E.1 defines the auxiliary data files which are considered to be "span valid". That is, the packets within the files (either on the CD or delivered on-line) are valid from the packet time up to the time of the next packet in that file. Using the STOF as an example, if a CD-ROM was produced for 1996-04-18, and the orbit log has entries timestamped as follows:

1996-04-17T15:00:00Z	
1996-04-17T18:00:00Z	
1996-04-17T22:00:00Z	*
1996-04-18T06:00:00Z	*
1996-04-18T11:00:00Z	*
1996-04-18T17:00:00Z	*
1996-04-18T23:00:00Z	*
1996-04-19T04:00:00Z	
1996-04-19T10:00:00Z	

Then those entries marked with a * would be included as packets on the CD-ROM. The entry timestamped for 1996-04-17 would be included as this orbit entry is valid until the next entry and therefore covers the period from **1996-04-18T00:00:00Z** until **1996-04-18T05:59:59Z** for the day required**.**

With regard to on-line data delivery, the packet which is valid at the requested start time will also be included in the data returned by the DDS for these span valid files.

All HPD files are span valid, as are the STOF, SATT and TCAL. The STEF and CMDH files are "point valid", that is, the packets are only valid at a certain point in time (since events and commands happen at a certain point in time).

With regard to span valid packets for off-line data delivery, when the amount of data requires that there is more than one CD for the day, any currently (i.e. at the time of the split) valid packet will be repeated at the start of each CD, since they apply to the data on that CD. For example, if an HPD packet becomes valid halfway through the time range of the first of three CDs for a day, and is valid for the rest of that day, then it will be repeated at the start of the second and third CDs. This ensures that all the auxiliary and HPD data required to understand the science data is present on each CD.

E.10 : Covariance Matrix (COVM)

The evaluation of the spatial gradient of scientific parameters measured in the Earth's magnetosphere requires a good knowledge of the tetrahedron formed by the four Cluster satellites. The errors in the tetrahedron can be estimated from the covariance matrices of the individual spacecraft state vectors.

For completeness it should be mentioned that a 'mean' covariance matrix based on several consecutive orbit determinations will be provided.

The file is not intended for direct interpretation, but via access by a Fortran routine supplied, which is described below.

The content and format of the file is defined in the following tables. The shaded rows define the format and are followed by the parameter names in an unshaded row. The format definition in brackets follows the ANSI FORTRAN notation for format statements (e.g. A29 means 29 ASCII characters, 5X means 5 spaces, I2 means a 2 character denary integer and F7.2 means a 7 character fixed point number with 2 decimal characters).

S/C/ ID> (I3, X2)	<Predict or recon.> (A1, X2)	<Generation time> (A20, X2)	<Data start time> (A20, X2)	<Data end time> (A20)	LF (A1)
SCID	**PREREC**	**GENTIM**	**SRTTIM**	**ENDTIM**	**LF**
Rec ID> (I3)	<MJD orbit> (F15.9)				LF (A1)
NREC	**EPOCH**				**LF**
Rec ID> (I3)	<x-y-z of position and velocity vector> (3F11.3, 3F11.7)				LF (A1)
NREC	**STATE(6)**				**LF**
Rec ID> (I3)	<covariance matrix> (6(6D11.5, /))				LF (A1)
NREC	**COVMAE (6,6)**				**LF**

Table E-10 Definition of the Covariance Matrix File

Where: **SCID** will contain the S/C body identification, this is an integer from 1 to 4 inclusive. The SCID shall be assigned before launch to each physical spacecraft, and remain the same throughout the mission.

PREREC a single character flag indicating if the data is predicted (P) **or reconstituted (**R**).**

GENTIM date and time when the leader block was written to the file in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

SRTTIM date and time of the start of the period for when the data is valid in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

ENDTIM date and time of the end of the period for when the data is valid in CCSDS time code A format (YYYY-MM-DDThh:mm:ssZ**).**

LF is a single line-feed character (ASCII $0A_{hex}$).

NREC an internally used record identifier.

EPOCH Modified Julian Date 2000 (MJD 2000), of the epoch for the reference Kepler orbit.

STATE State vector (expressed in km and sec.)

NREC an internally used record identifier.

COVMAE Full covariance matrix (although symmetric) of the state vector. The matrix contains 36 parameters and is expressed in units of km and sec.

The co-ordinate system is the Inertial Mean Geocentric Equatorial System of year J2000.0, with the x-axis towards the mean vernal equinox, the x-y plane coinciding with the mean equatorial plane and the z-axis toward north. Time and co-ordinate systems used for orbital operations at ESOC are described in Ref. [1].

A subroutine is provided which calculates the covariance matrix of the state vector, i.e. cov x(t) for any required time t within the period of validity i.e. between SRTTIM and ENDTIM by pro-pagating the state vector (STATE) and the covariance matrix (COVMAE) which both refer to the time EPOCH. Unperturbed Keplerian motion is applied for the propagation. In fact, the covariance matrix at time t is given by the equation:

cov x(t) = Q times COVMAE times Q transposed,

where Q is a 6x6 matrix containing the partial derivatives of the state vector at time t with respect to the state vector STATE at the time EPOCH.

The calling sequence would be:

SUBROUTINE COVMA (IUNIT,T,COVMAT,IERR)

where
IUNIT I*4) = Logical FORTRAN unit number of covariance matrix file
T (R*8) = Input time in Modified Julian date (MJD2000)
COVMAT (R*8) = Resulting covariance matrix of state (array with a dimension of 6x6, units are km and sec) for the time T
IERR (I*4) = Error code
0 means no error
1 means T is too early
2 means T is too late
4 means IUNIT is out of range
7 means file is inconsistent.

Appendix F : Standard Formatted Data Units (SFDUs)

F.1 : Overview of the Standard Formatted Data Unit (SFDU) Concept

The Standard Formatted Data Unit (SFDU) concept provides standardised techniques for the automated packaging and interpreting of data products. It puts no constraint on the format of the user data, and can thus accommodate standard formats developed by other organisations or user communities. It operates in a heterogeneous environment.

The SFDU concept has been developed to address some of the problems associated with information interchange. It offers the following:

- A low overhead, internationally recognised data labelling scheme which permits self-identification of data objects;
- Standard techniques for providing complete and unambiguous data descriptions;
- Procedures for registration and administration of these data descriptions;
- Techniques for packaging labelled data objects into larger data products;
- Sufficient standardisation to allow the development of generic software to support the retrieval, access, parsing and presentation of SFDU data objects, while allowing those objects to have individual formats to satisfy particular application and user needs.

F.2 : SFDU Building Blocks - The Label-Value-Object

The basic SFDU building block is comprised of a LABEL field and a VALUE field, and is referred to as a Label-Value-Object (LVO). This structure is the fundamental structural element used to build SFDUs. It is shown in Figure F.1 . This basic element is deemed to be a stream of octets, the ASCII codes and octet and bit ordering used is described in Appendix G All data delivered by the CDDS will use these codes and octet/bit numbering scheme).

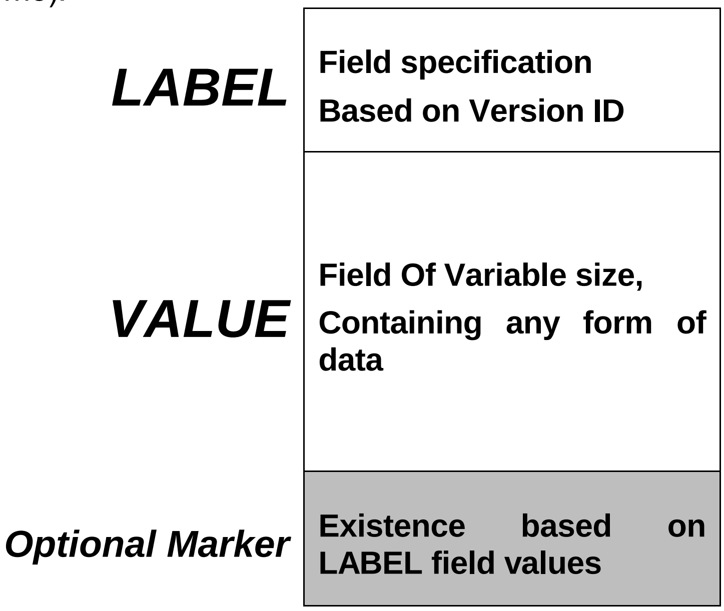


Figure F-1: LABEL-VALUE Encoding Structure

The VALUE field may contain any form of data that can be described by a user defined data description or by a CCSDS recognised data description. The method used to delimit this field, and a description of the data in this field, are identified through the associated LABEL sub-field.

The optional marker field is required by some delimitation techniques to delimit the VALUE field.

The SFDU LABEL has a fixed 20 octet length, which is split into a number of sub-fields as shown in Figure F-2

CAID (ADID 1 of 2)	Version ID	Class ID	Delim ID	Spare	DDID (ADID 2 of 2)	Delimitation Parameter	LABEL Sub field
0-3	4	5	6	7	8-11	12-19	Octet Number

Figure F-2: CCSDS LABEL Specification - Version ID = 1, 2 and 3

The **Version ID** sub-field dictates the format and meaning of the other sub-fields, in all the data structures defined in this document the Version ID is always equal to "**3**" (ASCII character "3"). (Version IDs "1" and "2" are for compatibility with older SFDU structures only)

For Version ID = "3", the other sub-fields have the following meaning and values.

- **Control Authority Identifier (CAID)**: The Control Authority Identifier contains the identifier of the organisation that has assigned the DDID to the information describing the VALUE field. This Control Authority Office has the responsibility for maintaining this data description information and supplying it to user. The CAID is the first part of the ADID.

- **Data Description Identifier (DDID)**: The Data Description Identifier contains the identifier of the data description information held at the Control Authority Office, as identified by the CAID. The DDID is the other part of the ADID.

- The combination of the CAID and the DDID is called the **Authority and Description Identifier (ADID)**. The ADID uniquely identifies the data description information that applies to the associated VALUE field.

- **Class ID**: The Class ID indicates the kind of data contained in the VALUE field following the LABEL. The Class IDs can be split into three basic categories, as shown in Figure F3. The Structure Classes handle the packaging of LVOs, the Service Classes provide CCSDS service mechanisms and the Data Classes contain the actual user data.

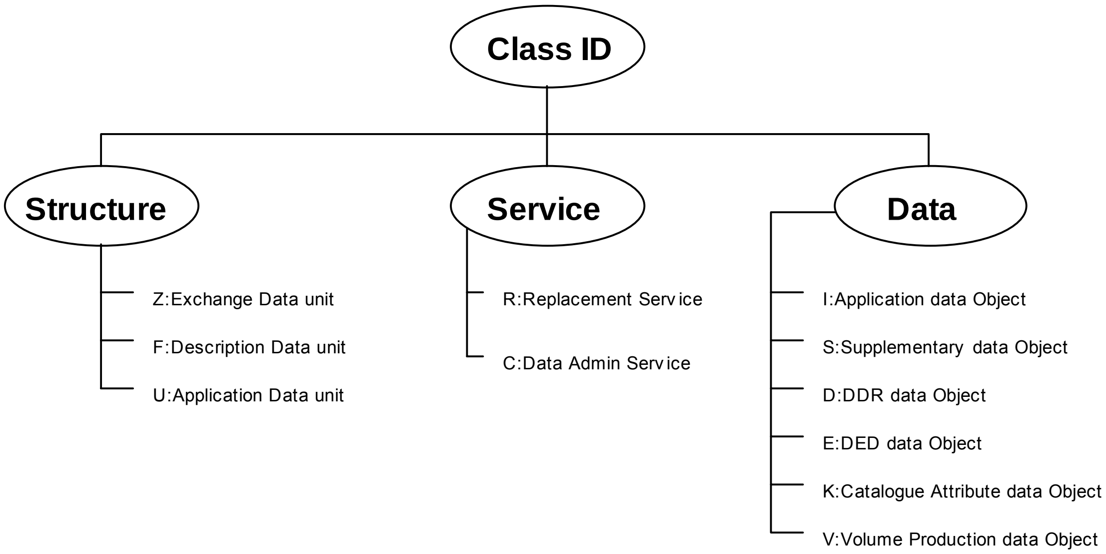


Figure F-3: SFDU Class ID Breakdown

- **Delimitation ID**: This field defines how the size of the VALUE field is determined. The possible values for this field as used in the SFDU structures defined in this document are (for a full list see Ref. [3]):

ASCII character "**A**" : This indicates that the length of the VALUE field is specified in the **Delimitation Parameter** sub-field in decimal ASCII, with leading zeros.

ASCII character "**F**" : This indicates that the VALUE field is complete when a single End-of-File is encountered in the input stream. The **Delimitation Parameter** sub-field must contain the ASCII string "**00000001**"

- **Delimitation Parameter:** This sub-field is used to provide any parameters that are required to complete the delimitation of the VALUE field. For example, in the two delimitation techniques used here, for Delimitation ID = "A", this field contains the VALUE field length represented by an eight octet decimal ASCII string, for Delimitation ID = "F" no further parameter is required as one may not have more than one EOF on random access media, therefore this field is set to the ASCII string "00000001".

- **Spare**: This is a spare octet which is set to the RA numeric character 0 **(zero).**

F.3 : Overview of CCSDS Defined ADIDs

There are two types of recognised ADIDs:

1. Those defined by the CCSDS, appearing in CCSDS Recommendations, and beginning with the four characters "CCSD". These are referred to as CCSDS ADIDs.

2. Those defined by SFDU users, typically through data description registration with Control Authority organisations (See Ref. [4]). These are referred to as non-CCSDS ADIDs.

While it is expected that most ADIDs shall be assigned by Control Authority Offices to data descriptions prepared by data producers, there are a number of standard descriptions that have been defined by the CCSDS for general use within the SFDU domain. These descriptions are assigned CCSDS ADIDs and they appear in CCSDS Recommendations. The CCSDS ADIDs that are used in this document are described in Table F1.

ADID	Usage
CCSD0001	The VALUE field contains one or more LVOs
CCSD0002	The VALUE field is expressed in ASCII Encoded English (See Ref. [5])
CCSD0003	VALUE field contains several "parameter=value" (PVL) statements that optionally label external data objects before logically including them in the current structure
CCSD0004	VALUE field contains one or more "parameter=value" (PVL) statements that identify a data description package and optionally reference other metadata objects
CCSD0005	VALUE field contains one or more LVOs, making up a Description Data Unit (DDU)
CCSD0006	The VALUE field is expressed in Parameter Value Language (PVL, see Ref. [9])
CCSD0009	VALUE field contains one or more LVOs, making up an Application Data Unit (ADU)

Table F-1 CCSDS Defined ADIDs

F.4 : SFDU Structuring

SFDU data products are constructed from the basic LVO in one of two ways. If the VALUE field of the LVO contains purely user data it is termed a "Simple LVO". If, on the other hand, the VALUE field of the LVO contains purely LVOs, it is termed a "Compound LVO".

SFDU products are always packaged in a special kind of Compound LVO called the Exchange Data Unit (EDU). Only EDUs may be interchanged between systems. Special types of Compound LVOs are also provided to package together application data (the Application Data Unit (ADU)) and data description data (the Description Data Unit (DDU)). The CCSDS defined categories of Simple and Compound LVOs, which vary depending upon the type of data or LVOs respectively that they contain, are detailed in the following sections.

F.4.1 : Simple LVOs

Data in a Simple LVO may be viewed as belonging to one of the following categories:

- Application data; that is the data which is of primary interest (typically measurements or data derived from measurements);

- Supplementary data; that is data that is considered to enhance the understanding of the associated data;

- Data description information, telling how the application data are formatted, including such details as size of the data fields, numerical or other representations used and the meanings of the fields;

- Data cataloguing and/or data production information, giving certain overall attributes of the data, for example, date of generation, instrument used, instrument location, general information about the way the data was collected, relayed or processed, etc .

Any one of these types of data may be contained in the VALUE field of a single LVO.

F.4.2 : Compound LVOs

Compound LVOs are LVOs which contain within their VALUE field a sequence of one or more LVOs, each of which can be a Simple or Compound LVO itself. LVOs that are contained in the VALUE field of a Compound LVO are deemed to be one "Structure Level" lower than that of the containing Compound LVO. If any of these contained LVOs are themselves a Compound LVO then they will themselves contain a sequence of LVOs; this sequence is at the next lower "Structure Level". This process may continue indefinitely leading to a succession of structure levels. This process is the way in which LVOs are nested. There are no rules dictating the number or order of Compound and Simple LVOs within a data product, except that there must be at least one Simple LVO at the lowest structure level of any Compound LVO (i.e., a Compound LVO cannot have a VALUE field of zero length).

There are three types of Compound LVOs; there is the Exchange Data Unit (EDU), and two particular structures which must be packaged within an EDU. These are the Application Data Unit (ADU), which explicitly does not contain any data description information, and the Description Data Unit (DDU), which can contain only data description information.

F.4.2.1 : Exchange Data Unit (EDU)

Typically an SFDU data product consists of not only the data *(e.g., an image, a set of measurement samples)*, but also all the supporting metadata that is needed to understand the data product. Any type of data may be contained within an Exchange Data Unit (EDU). SFDU data <u>MUST</u> be exchanged in the form of an EDU.

F.4.2.2 : Application Data Unit (ADU)

The purpose of an ADU is to package application data instances *(e.g., measurement samples)* together with any necessary ancillary data *(e.g., sampling rate)* and identification data *(e.g., catalogue information)*, and to explicitly exclude any data description information. In CLUSTERS data delivery, each day of data from each instrument, from each spacecraft shall be contained within its own ADU, with a relevant catalogue LVO.

F.4.2.3 : Description Data Unit (DDU)

A Description Data Unit is characterised as follows:

- It carries the description of a data object (typically syntactic information such as the format of a sample, and semantic information such as the name and units of the components of the sample);
- It explicitly links the data description to the data object to which it applies;
- It does not include any application data instances.

Each valid category of data is assigned a unique Authority and Data Description Identifier (ADID) by the Control Authority (see Ref. [4]), which shall be at ESOC. Control Authorities provide the important administrative function of assigning ADIDs and collecting and validating the associated data descriptions.

In the SFDU concept the ADID points to a Data Description, as illustrated in the general example of Figure F-4. In the very simple approach being followed for CLUSTER it is proposed that, in general, the data descriptions be expressed in <u>ASCII Encoded English</u>, i.e. no automated interpretation of data packets is foreseen.

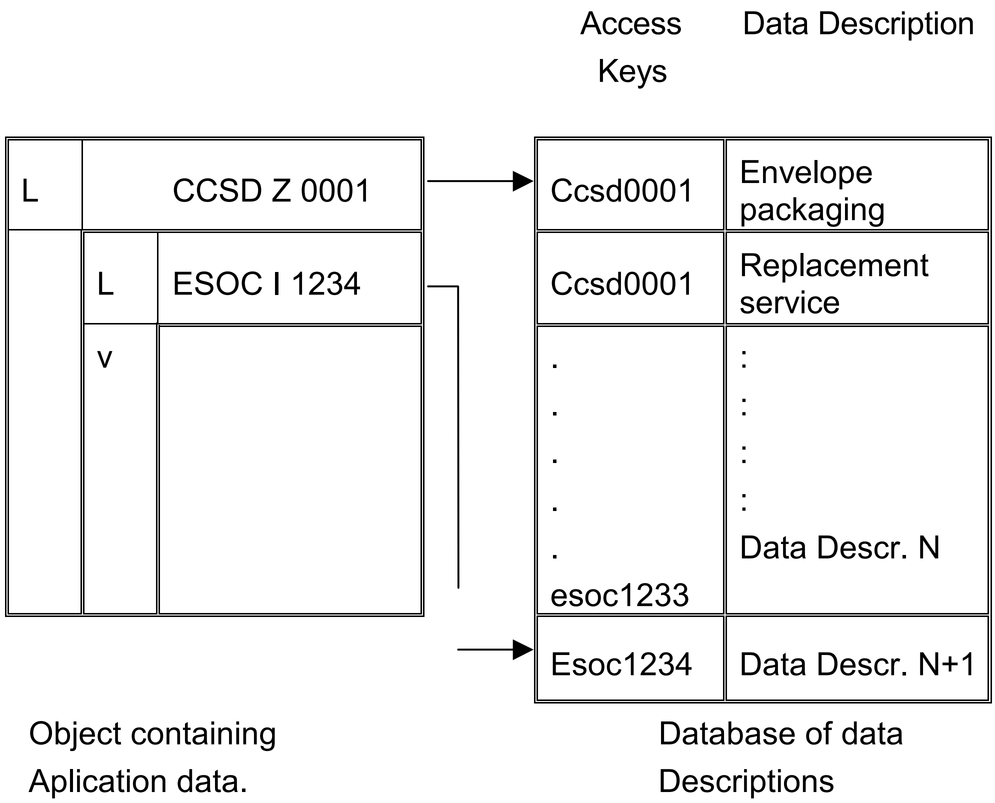


Figure F-4 Referencing of Data Descriptions

Thus ESOC, as Control Authority, shall have registered English language data descriptions of all data. Figure F-5 illustrates schematically what such a description looks like. The description is in another format of EDU. All these descriptions shall be provided on the CD-ROM with the science and auxiliary data, therefore the user shall have a wholly complete description of all data received.

<table>
<tr><td>L</td><td colspan="3">CCSD Z 0001</td></tr>
<tr><td rowspan="7">V</td><td>L</td><td colspan="2">CCSD F 0005</td></tr>
<tr><td rowspan="6">V</td><td>L</td><td>CCSD C 0004</td></tr>
<tr><td>V</td><td>ADIDNAME=ESOC1234;</td></tr>
<tr><td>L</td><td>CCSD D 0002</td></tr>
<tr><td>V</td><td>SYNTACTIC INFORMATION</td></tr>
<tr><td>L</td><td>CCSD E 0002</td></tr>
<tr><td>V</td><td>SYNTACTIC INFORMATION</td></tr>
</table>

Figure F-5 Example of a DDU

The main points to note are:

- The whole SFDU product is enclosed within a Class Z LVO (an EDU), indicating that the VALUE field can contain LVOs of any other class;

- The Class F LVO indicates that everything enclosed within it is part of a data description registration package. This is mandatory to register any data description package (Ref. [3]);

- The Class C LVO is used in its simplest form, and has a single Parameter Value Language (PVL) statement within its VALUE field, for example ADIDNAME=ESOC1234;**. This PVL statement specifies the ADID under which the data description shall be registered. White space is also allowed within the Class C LVO VALUE field;**

- The VALUE field of the Class D or Class E LVOs contain the actual data description. The ADID of this Class D is CCSD0002, this specifies that the VALUE field is represented in English ASCII Text. Each Class D LVO VALUE field shall therefore contain a conventional English language Interface Control Document.

As well as supplying these data description on the CD-ROM, the Control Authority, ESOC, shall maintain a data base of data descriptions. In the data base, each description shall be keyed on the ADID of the data it describes.

F.5 : EDU Structure Diagram

The four structures that have been illustrated in the previous section are the Simple Label-Value-Object (LVO), the Exchange Data Unit (EDU), the Application Data Unit (ADU) and the Description Data Unit (DDU). These structures may be packaged together as indicated in the structure diagram of **Error! Bookmark not defined.** (overleaf). Not all the components on the right of the **:=** have to be included in all EDUs, but at least one Simple LVO must be present. The packaging is hierarchical with the highest level object being an EDU.

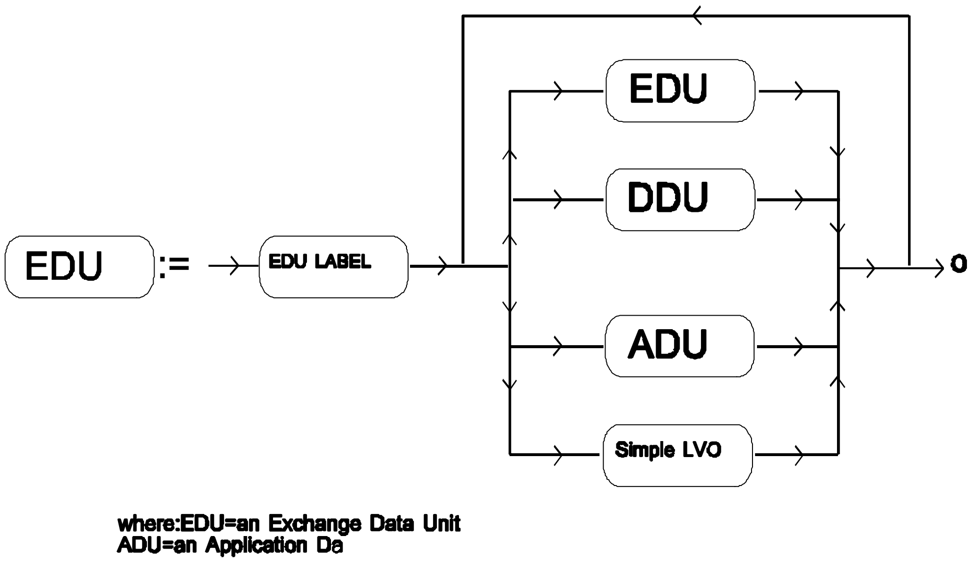


Figure F-6: Structure Diagram of an EDU

: Packaging Techniques

F.5.1 : Envelope Packaging

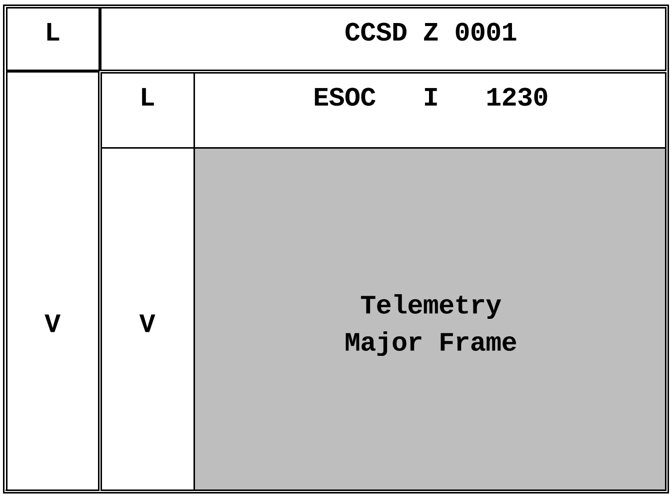


Figure F-7: Envelope Packaging

Envelope Packaging is the simplest form of SFDU packaging and is used when all the data/LVOs to be packaged reside in the same physical file. The idea is shown in F which shows the simplest case, in which an EDU contains one LVO of application data, which could, for example, be a major frame of telemetry data. A more elaborate example, showing the envelope packaging of a "bundle" of telemetry minor frames is shown in Figure F-7. Minor frames are bundled into an EDU for delivery of a partial major frame, so users can begin processing before the complete major frame is available.

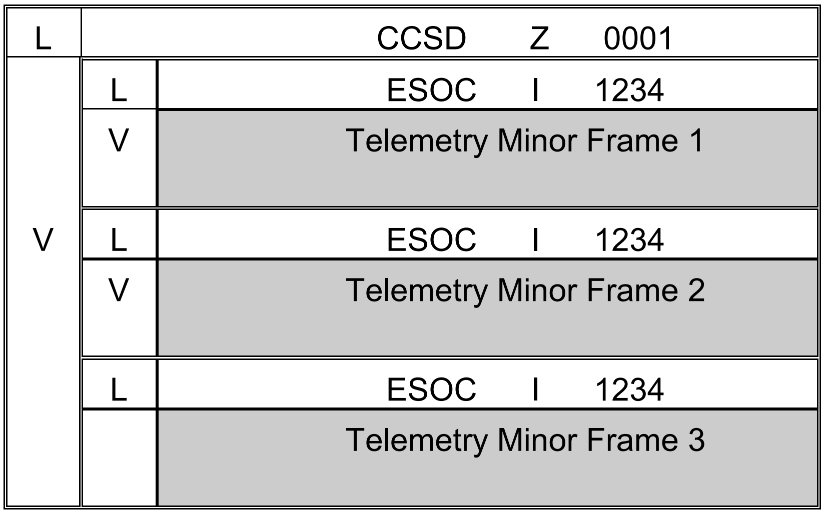


Figure F-8: More Complex Envelope Packaging

F.5.2 : Referencing Technique - the Replacement Service

The referencing technique is provided to include data units in a product even if those units are not stored contiguously with the rest of the product. This is realised by means of an LVO with Class ID = R, which contains the reference information in its VALUE field in the form of Parameter Value Language (PVL) statements (Ref. [5]). The concept is shown in F-9.

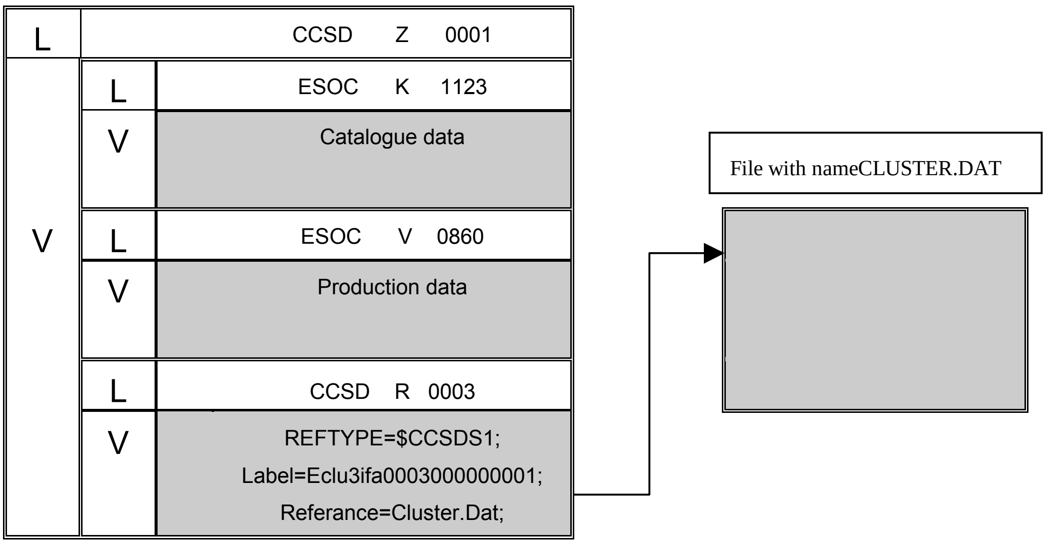


Figure F-9: The Replacement Service - Referencing an Unlabelled Data Object in a File

In this EDU, the data file (a sequence of telemetry frames) is stored on the same media, but not in the same file as the rest of the SFDU product which contains the associated catalogue and production data, the LVO with Class ID = R is envelope packaged in the EDU and references a file called CLUSTER.DAT. The PVL statements in the LVO with Class ID = R have the following meanings:

REFERENCETYPE=$CCSDS1: this statement defines the file naming convention for the **REFERENCE** statement. In the case of CLUSTER the CCSDS proposed canonical form, $CCSDS1, will be used.

LABEL=caid-v-c-d-s-ddid-xxxxxxxx: this statement provides a 20 octet string that is used to label the remote data file. The string has the same specification as for a standard LABEL.

REFERENCE=filename: this statement provides the actual referenced filename, the format of this depends upon the **REFERENCETYPE** parameter value.

Although there may be only one **REFERENCETYPE** statement in any one Class R LVO, and it must be the first statement, there may be any number of **LABEL** and **REFERENCE** statements, so that many files may be referenced from one Class R LVO.

Appendix G : Physical Data Nomenclature Conventions

G.1 : ASCII Codes Definition

This Appendix defines the ASCII code set that is used for SFDUs and also highlights the codes that comprise the Restricted ASCII character that is used in the LABEL field of an LVO.

A code is a correspondence between a symbol and a number of digits of a number system. The American Standard Code for Information Interchange (ASCII) is a seven-bit code also known as the USA Standard Code for Information Interchange (USASCII). The latest updated American National Standards Institute ANSI-X3 standard for this is ANSI X3.4-1977. This code has been incorporated into the ISO code of the same nature (ISO 646-1983) which includes other symbols and alphabets. Since the ISO code is an eight-bit code, the ASCII code is embedded in an eight-bit field in which the higher order bit is set to zero. The Restricted ASCII set of characters (denoted here by a * next to the code) is used by the CCSDS Recommendation. The primary reference to be used should be ISO 646-1983.

The ASCII and Restricted ASCII or RA codes are given in Table G-1. *(The code for each character (Char) is given in decimal (Dec), and hexadecimal (Hex)).*

Char	Dec	Hex		Char	Dec	Hex		Char	Dec	Hex	Char	Dec	Hex
NUL	0	00		*sp*	32	20		@	64	40	`	96	60
SOH	1	01		!	33	21	*	A	65	41	a	97	61
STX	2	02		"	34	22	*	B	66	42	b	98	62
ETX	3	03		#	35	23	*	C	67	43	c	99	63
EOT	4	04		$	36	24	*	D	68	44	d	100	64
ENQ	5	05		%	37	25	*	E	69	45	e	101	65
ACK	6	06		&	38	26	*	F	70	46	f	102	66
BEL	7	07		'	39	27	*	G	71	47	g	103	67
BS	8	08		(	40	28	*	H	72	48	h	104	68
HT	9	09		)	41	29	*	I	73	49	i	105	69
LF	10	0A		*	42	2A	*	J	74	4A	j	106	6A
VT	11	0B		+	43	2B	*	K	75	4B	k	107	6B
FF	12	0C		,	44	2C	*	L	76	4C	l	108	6C
CR	13	0D		-	45	2D	*	M	77	4D	m	109	6D
SO	14	0E		.	46	2E	*	N	78	4E	n	110	6E
SI	15	0F		/	47	2F	*	O	79	4F	o	111	6F
DLE	16	10	*	0	48	30	*	P	80	50	p	112	70
DC1	17	11	*	1	49	31	*	Q	81	51	q	113	71
DC2	18	12	*	2	50	32	*	R	82	52	r	114	72
DC3	19	13	*	3	51	33	*	S	83	53	s	115	73
DC4	20	14	*	4	52	34	*	T	84	54	t	116	74
NAK	21	15	*	5	53	35	*	U	85	55	u	117	75
SYN	22	16	*	6	54	36	*	V	86	56	v	118	76
ETB	23	17	*	7	55	37	*	W	87	57	w	119	77
CAN	24	18	*	8	56	38	*	X	88	58	x	120	78
EM	25	19	*	9	57	39	*	Y	89	59	y	121	79
SUB	26	1A		:	58	3A	*	Z	90	5A	z	122	7A
ESC	27	1B		;	59	3B		[	91	5B	{	123	7B
FS	28	1C		<	60	3C		\	92	5C	|	124	7C
GS	29	1D		=	61	3D		]	93	5D	}	125	7D

Char	Dec	Hex
RS	30	1E
US	31	1F

Char	Dec	Hex
>	62	3E
?	63	3F

Char	Dec	Hex
^	94	5E
_	95	5F

Char	Dec	Hex
~	126	7E
DEL	127	7F

Table G-1: ASCII and Restricted ASCII Codes

G.2 : Bit and Octet Numbering Conventions

In this document, the following convention is used to identify each octet (8-bit field) in an N-Octet field:

> The first octet in the field (to be transferred) will be drawn in the most left justified position and will be defined to be "Octet 0". The following octet will be defined as "Octet 1" and so on, up to "Octet N-1". When the field is used to express a numerical value, the Most Significant Octet (MSO), shall be the first octet of the field, *i.e., "Octet 0"*. The sequence of decreasing value will be Octet 1 to Octet N-1.

> According to the CCSDS convention, the Most Significant Bit (MSB) of any octet shall be the first bit transmitted and it shall be drawn in the most left justified position and designated as "Bit 0". The transmission sequence shall go from MSB to the Least Significant Bit (LSB).

Title of Appendix

CLUSTER ORBIT DETERMINATION, RESTITUTION AND PREDICTION

CLUSTER Document Identification Number

CL-ESC-ID-0001-Appendix H

OAD Document Identification Number

CLU-IA-Annex H

Prepared by:	S. Pallaschke, M. Soop
Status:	Issue 1
Date:	1994/05/09

Appendix H : Cluster Orbit Determination, Restitution and Prediction

H.1 : Introduction

The CLUSTER mission is designed to perform magnetospheric physics experiments which examine the Earth's magnetosphere in three dimensions using a cluster of four spacecraft in quasi-polar orbits. As a result of the mission analysis, the following orbit type has been selected:

perigee radius of 4.0 R_E
apogee radius of 19.6 R_E
inclination of 90°.

Within the two years of mission, various configurations of the CLUSTER satellites will be required, which obviously will necessitate slightly different orbits. The extremes for each individual orbital element within the entire mission are given below:

Orbital Element	Initially	Extreme Values	
Radius	4.0/19.6 R_E	6.4/17.2 R_E	2.2/21.3 R_E
Semi-major axis	75621.1 km	75000 km	76800 km
Eccentricity	.661017	.46	.81
Inclination	90°	85°	95°
Arg. of Perigee	345°	345°	25°
Period	57 hrs		

The CLUSTER satellites will be tracked from Redu, Belgium and Odenwald, Germany. As the two ground stations are located in the northern hemisphere, the satellites will only be visible during the first half of the revolution primarily.

Furthermore, since there will be 4 closely separated satellites but only two ground stations, it will not be possible to track each satellite for the total period of its visibility. In fact, during each complete pass, two of the satellites will be allocated to the Redu ground station while the other two satellites will be allocated to the Odenwald ground station. Since each antenna will be allocated to one satellite at any one time, the allocation will need to be scheduled to obtain adequate tracking of each of the four satellites. This means that, when considering individual satellites, several ground station tracking passes will be reduced in length due to station handover from one satellite to another, and some passes may even be lost completely.

The orbit determination accuracy that will be required for the mission is closely related to the intersatellite distances that will be used. These distances will depend upon which region of the magnetosphere is being investigated. There is a direct relationship between these intersatellite distances and the orbit. The tolerance of the satellite separation distances is

10 km for separation distances less than 1000 km and
1 % for separation distances greater than 1000 km.

The most stringent of these two requirements leads to an orbit determination accuracy of ± 5 km.

H.2 : Orbit Determination

The orbit determination procedure involves measuring the distance of a satellite from one or more ground stations, modelling the motion of both the satellite and the ground stations in an inertial coordinate system and then trying to determine the satellite state by some estimation process, usually a non-linear least squares procedure. These calculations are disturbed by several different factors, of which the most significant can be described as follows:

Firstly, measurements of the ground station to satellite distances are subject to bias and noise which arise from hardware errrors and atmospheric influences on the propagating tracking signals. Secondly, the modelling of the satellite is inexact due to dynamic modelling errors of the Solar radiation pressure and of the Earth's gravitational field. Then finally, there are the ground station dependent factors such as the station location accuracy and the tracking schedule.

The assumptions that were made for the orbit determination accuracy assessment are:

Spacecraft parameters.

Mass of satellite:	1150 kg
Cross sectional area:	4.5 m^2

Ground station parameters.

Station location accuracy:	2 m per component.

Tracking performance.

Range (bias:noise):	5 m : 3 m
Range-rate (bias:noise):	1 mm/s : 1 mm/s.

Modelling errors.

Uncertainty in Solar radiation pressure:	5 %

A further essential parameter to the orbit determination accuracy assessment is the tracking duration. The orbit determination will follow the general principle of processing entire revolutions. In order to obtain a good solution for a complete revolution, the tracking data of the following revolution and as well of the previous one are required. In fact this results in a 2.5 revolution tracking arc for the refined orbit determination of the middle revolution, as the ground coverage occurs primarily during the first half of the revolution because of the station location and orbit orientation. In case refined orbit determination is carried out for 2 revolutions, the tracking would be required over 3.5 revolutions.

Assuming a tracking interval of 2.5 or 3.5 revolutions, corresponding to about 6 days or 8.5 days respectively, a positional accuracy of about 2 km and an accuracy in semi-major axis of about 20 - 30 m could be achieved under nominal conditions. This achievable accuracy is well within the required limit of ± 5 km.

H.3 : Orbit Determination Procedure

Although the orbit determination will be based on tracking measurements accumulated over 2.5 or 3.5 revolutions, only the middle revolutions (1 or 2 respectively) will be considered as refined and final orbit determination. The orbit file will be overwritten from these middle revolutions onwards.

As mentioned before, the orbit determination accuracy will be in the order of a few kilometers for the satellite position. As the orbit update will always occur at perigee, the inconsistencies will be outside the scientific part of the revolution.

It is planned that the orbit determination is carried out twice per week. One orbit determination will cover 2.5 revolutions (i.e. refined orbit determination for 1 revolution) and the other one over 3.5 revolutions (i.e. refined orbit determination for 2 revolutions). With this sequence the 3 revolutions occurring within a 7 day period will be processed every week. However, the orbit determination will not always be performed on the same day, a slow sliding through the working week will be required. This procedure leads for the refined orbit determination to a maximum time delay of 7 days with the following breakdown:

4.75 days	2 revolutions
1.20 days	first half of subsequent revolution
1.00 days	processing.

The representation of the orbit data on the orbit file is defined in the next section. The approximation error of the reconstituted orbit will be less than 30 m for the position and less than 10 mm/s for the velocity.

In summary, the larger inconsistencies of a few km for the satellite position caused by the orbit determination will only occur at perigee, which is outside the scientific region of the orbit. A few very small inconsistencies of up to 30 m in satellite position will occur, which are caused by the orbit approximation method.

H.4 : Orbit Data Compression

For distributing orbital data outside ESOC's Flight Dynamics system, a special external orbit file representation has been designed. For eccentric orbits of the Cluster type, the combination of a Kepler orbit and Chebyshev polynomials provides a handy and flexible compromise between accuracy and degree of data compression.

The coordinate system is the mean geocentric equatorial system of J2000.0 and the time is Coordinated Universal Time UTC [Ref. HR.1]. Time is internally represented as a floating point number in Modified Julian Days, from the year 2000 (MD2000).

H.4.1 : Orbit Propagation

The orbit propagation and the calculation of the data for the Cluster orbit files are performed at ESOC in principle along the following lines. The same process applies independently to each one of the four Cluster satellites.

Orbits are expressed by means of a 6-dimensional state vector, i.e. the x-y-z components of the satellite position in km and velocity in km/s, and the corresponding epoch in MJD2000. The state vector is integrated from Newton's differential equation by a predictor/corrector multistep method of order 8 with about 150 steps per revolution. Time regularisation is applied to make the steps shorter around perigee and longer around apogee.

The integrated sequence of state vectors is stored at ESOC on an internally used file as the most accurate representation of the orbital evolution. For the compression to the external orbit file, the data is divided into compression intervals. The interval lengths and the number of state vectors in each can be varied arbitrarily, but for the Cluster routine phase orbit, about 4 intervals per revolution is adequate. When there is an orbit manoeuvre or a leapsecond in UTC, the time of the discontinuity is used as a boundary between two intervals.

H.4.2 : Compression Method

The orbit within each compression interval is represented by the data on one block of the external orbit file. Each block consists of between 3 and 13 records, but a block is the smallest unit that can be used for retrieving orbital information. The orbit file consists of a sequence of blocks, where the start time of the interval in each block equals the end time in the preceding block.

The compression is performed separately for each interval, which causes small discontinuities in the orbit at the interval boundaries, although less than the specified accuracy. The compression starts by the selection, near the interval centre, of a state vector and an epoch to define the reference Kepler orbit for the interval. From this state vector, an elliptic orbit is propagated by means of Stumpff functions and subtracted from all state vectors in the interval. The equations for the Keplerian propagation are given in Escobal, Methods of Orbit Determination [Ref. HR.2].

The remainders (Z), which are the differences between the actual orbit and the Kepler orbit, are fitted by least squares to a set of 10 coefficients (CU), one set for each of the 6 components of the state vector, of a 9th degree Chebyshev polynomial. The independent variable in the polynomial is the time (t) that has been normalised to -1 at the beginning and +1 at the end of the interval, i.e.

$$Z(t) = \sum_{n=0}^{9} CU_n T_n(t)$$

where: $T_o(t) = 1$

$T_1(t) = t$

$T_{n+1}(t) = 2t\, T_n(t) - T_{n-1}(t)$

The selected state vector, the polynomial coefficients and auxiliary data comprise one block on the orbit file according to the description given in Appendix E. Small coefficients are suppressed according to the accuracy required in the relevant mission phase, so the number of coefficients can vary between 0 and 10. The semimajor axis, the inverse mean motion and the absolute value of the position vector of the reference Kepler orbit is redundant information that is stored on the file in order to simplify the de-compression.

The revolution number for the epoch of the reference Kepler orbit is stored on the file. Its integer part is the number of revolutions since the mission start of the MOP orbit and the fractional part (counted from perigee to perigee) is obtained by dividing the mean anomaly at this epoch by 2*π or 360 degrees.

H.4.3 : External Users

There will be 4 different files, one per satellite. The same file lay-out and subroutine is used for the reconstituted orbit, the near predicted and the long term predicted orbit.

The orbit files for the Cluster satellites are distributed to the External Users with new updates according to an agreed schedule. The User may delete, split or concatenate parts of different file updates to compose a new orbit file, as long as the following constraints are considered:

- Only complete blocks shall be moved; a block shall never be split or modified;
- There must not be any time gap between the end of a block and the start of the next block of more than a few seconds;
- Time overlaps of subsequent blocks is allowed;

- Concatenation of blocks with different accuracy requirement is possible;
- There is only indirect protection against concatenation of blocks with different satellite numbers.

The record identification number on each record is coded such as to provide information of its position within a block and the total number of records in the block.

H.5 : Orbit Data De-compression

H.5.1 : File Read Routine

The orbit file can be read by a FORTRAN subroutine of about 200 statements, including comments. The calling sequence is shown below.

```
SUBROUTINE ORBIT(DAY,KODE,LFILE,IERROR,NSAT,X,REVNUM)
CP  ORBIT: Retrieval routine for compressed Cluster orbit
C
C   Input:
CI  DAY (R*8) = Modified Julian day, from 2000, for the state vector
CI  KODE (I*4) = number of components of state vector = dim. of array
C            X(); = 3 for S/C position, = 6 for position & velocity
CI  LFILE (I*4) = logical number of input data file
C   Output:
CO  IERROR (I*4) = return code: 0=no error, 1='DAY' too early, 2=too
C          late, 3=time gap in data, 4=wrong value of 'KODE',
C          5=file content inconsistent, 6=read error from data file
CO  NSAT (I*4) = satellite number; 1, 2, 3, 4
CO  X(KODE) (R*8) = spacecraft position, km (and velocity, km/s)
CO  REVNUM (R*8) = revolution number
```

H.5.2 : File Read Logic

For reading an orbit file, the User must assign a FORTRAN number to it, with different numbers allocated to the different Cluster satellites. This FORTRAN number shall be used for the parameter LFILE, and the satellite number is returned as the parameter NSAT. By verifying the latter, the User can check that a block from the correct orbit file has been read. The subroutine ORBIT does not explicitly open the orbit file.

At the first call of the subroutine ORBIT, the orbit file is read from the beginning until a block is found whose time interval contains the input value DAY. In case there is some time overlap between blocks, the first one that contains DAY is selected.

After a block is found, the de-compressing is performed as described below. The content of the block is kept inside the subroutine and is reused at the next call of ORBIT, if the new value of DAY lies within its time interval. If the new DAY lies beyond the end of the interval, the orbit file is read forward until a matching interval is found.

The orbit file is read from its beginning at a new call of ORBIT when:

- The new value of DAY lies before the start time of the interval for the last read block;
- The error return code IERROR was non-zero at the last call;
- The new call specifies a new file with a new number LFILE.

Because of the last mentioned property, the reading is inefficient if the User reads all the four satellite orbits in an inner loop of his program. In that case, the User should instead load the subroutine ORBIT four times under four different names, once per satellite, in order to save the last read block of each of the four orbits.

H.5.3 : De-Compression Method

The de-compression of the orbit data is performed as the inverse of the compression. The reference Kepler orbit is propagated as an ellipse by Stumpff functions from the reference epoch to the calling value of DAY. Since no closed analytical solution of the propagation exists, this is done by iteration until an accuracy of 14 digits has been achieved. The number of components (KODE) that is desired for the state vector can be set to any number between 1 and 6.

The required number of Chebyshev polynomials are calculated by the standard recursive formula, multiplied by the coefficients and added to the respective components of the state vector. The input time DAY is normalised to a dimension-less value between -1 and +1 for use as the independent variable in the polynomial. A long prediction of the orbit with low accuracy can be represented by the same lay-out of the orbit file and be read by the same ORBIT subroutine with the number of coefficients in the block set to zero.

The revolution number at DAY is obtained as follows. The time difference between DAY and the reference epoch is divided by the inversemean motion of the reference orbit and also divided by 2*pi and then added to the revolution number at the reference epoch. In this way the revolution number is represented as a linear function of time inside each data block but with a certain discontinuity at the block boundaries that is caused by the orbital perturbations.

H.5.4 : Error Returns

The error return codes are:

IERROR=0 no error;

IERROR=1 when the calling value of DAY is earlier than the start time of the first block on the file;

IERROR=2 when the calling value of DAY is later than the end time of the last block on the file;

IERROR=3 when there is a time gap of more than a few seconds between two successive blocks before a block that matches DAY has been found;

IERROR=4 when the input value of KODE is not 1, 2, 3, 4, 5 or 6.

IERROR=5 when the data on the orbit file is inconsistent. The following inconsistencies can be identified:

- If the record identification numbers do not match;
- If the begin and end times of a block interval are reverse;
- If and end-of-file appears before the first block has been read;
- If and end-of-file appears during the reading

of a block;

- If the iteration for the solution of the Kepler equation does not converge (can only happen when the data is corrupted).

IERROR=6 when the FORTRAN read statement gives an error.

H.6 : Coordinate Transformation

The CLUSTER orbit data will be given in the mean geocentric equatorial system of J2000.0.

The possibly required coordinate transformation to the geocentric Solar Magnetospheric (GSM) Coordinate System is given in the CLUSTER Interface Control Document between Flight Dynamics and Mission Analysis [Ref. H.R3].

It might also be required to transform the CLUSTER orbit data from the J2000.0 system to the mean of date one. This transformation is described in Ref. H.R1, chapter 7: Precession and the Mean-of-Date System. The rotation matrix for the precession of the Earth is calculated in the enclosed Orbit Library Subroutine PR2000, which should be applied in the following way:

$$R_{Mean\text{-}of\text{-}date}(i) = \sum_{j=1}^{3} P(i, j) \bullet R_{2000}(j)$$

```
SUBROUTINE PR2000(DAY,P)
CP  COMPUTES THE PRECESSION MATRIX P(3,3) FOR CONVERTING A VECTOR
C IN MEAN GEOCENTRIC EQUATORIAL SYSTEM OF 2000.0 TO MEAN-OF-DATE.
C REF: THE ASTRONOMICAL ALMANAC 1985 PAGE B18.
C
CINPUT:  DAY = MJD2000 = MOD. JULIAN DAY FOR THE MEAN-OF-DATE
SYSTEM
C           = MJD(1950) - 18262.0
C
COUTPUT: P(3,3) = PRECESSION MATRIX FOR THE TRANSFORMATION:
```

```
C     R(MEAN-OF-DATE) = P(,)*R(2000)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION P(3,3)
C
C CONVERT TO STANDARD EPOCH J2000.0 = 2000 JAN 1 AT 12:00:00
      T = DAY - 0.5D0
C
C  GZ=GREEK Z(A), ZA=Z(A), TH=THETA, ACCORDING TO THE REFERENCE.
C ORIGINAL, WITH TJC = (DAY-0.5D0)/36525.D0  IN JULIAN CENTURIES:
C     GZ = RAD*TJC*(0.6406161D0 + TJC*(839.D-7 + TJC*5.D-6))
C     ZA = GZ + RAD*TJC*TJC*(2202.D-7 + TJC*1.D-7)
C     TH = RAD*TJC*(0.5567530D0 - TJC*(1185.D-7 + TJC*116.D-7))
C
      GZ = T*(0.3061153D-6 + T*(0.10976D-14 + T*0.179D-20))
      ZA = GZ + T*T*(0.2881D-14 + T*0.358D-22)
      TH = T*(0.2660417D-6 - T*(0.1550D-14 + T*0.41549D-20))
C
      CGZ=DCOS(GZ)
      SGZ=DSIN(GZ)
      CZA=DCOS(ZA)
      SZA=DSIN(ZA)
      CTH=DCOS(TH)
      STH=DSIN(TH)
      P(1,1) = CGZ*CZA*CTH - SGZ*SZA
      P(1,2) = -SGZ*CZA*CTH - CGZ*SZA
      P(1,3) = -CZA*STH
      P(2,1) = CGZ*SZA*CTH + SGZ*CZA
      P(2,2) = -SGZ*SZA*CTH + CGZ*CZA
      P(2,3) = -SZA*STH
      P(3,1) = CGZ*STH
      P(3,2) = -SGZ*STH
      P(3,3) = CTH
      RETURN
      END
```

H.7 : Cluster Orbit Software

Orbit and Attitude information will be provided (as auxiliary data on each CD-ROM) for the processing of the CLUSTER Scientific Data. For the evaluation of the orbit data and their further processing the following four FORTRAN subroutines are made available by FCSD/OAD in the directory **/SOFTWARE/** on each CD-ROM:

JD2000.FOR Converts year, month, day, hour, minute, second to the Modified Julian Date MJD2000 with 0 on 2000/Jan/1 at 00:00:00.

DJ2000.FOR Converts Modified Julian Date MJD2000 to year, month, day, hour, minute, second

ORBIT.FOR Retrieval routine for compressed CLUSTER orbit file

PR2000.FOR Precession matrix of mean equatorial system of date relative to mean of J2000.0 (as given in The Astronomical Almanac 1985, page B18).

Please note that:

- the subroutines include READ statements that assume that the DDS packet header (see Table 5-4) is not present, therefore, before calling the subroutines the DDS packet header must be removed by the user.
- The subroutines have been developed on a SUN/UNIX environment. If a different environment is used then any adaptation and/or maintenance of the software is the responsibility of the user.

H.8 : References

HR.1 OAD Principles, Standards for Time and Coordinate Systems, May 1994

HR.2 ESCOBAL, Methods of Orbit Determinaton, Pages 112 - 114

HR.3 CLUSTER CL-ESC-ID-0606, Interface Control Document between Flight Dynamics and Mission Analysis, Chapter 6.1

Title of Appendix

CLUSTER
SPIN PHASE AND SPIN RATE RECONSTITUTION METHOD

CLUSTER Document Identification Number

CL-ESC-ID-0001-Appendix I

OAD Document Identification Number

CLU-IA-Annex I

Prepared by:	J.S. Llorente-Martinez
Edited by:	J.S. Llorente-Martinez
Reviewed by:	
Status:	Issue 1
Date:	1994/04/18
ESOC Technical Officer:	G. Gienger
Work Package:	WP 212581

Appendix I : CLUSTER SPIN PHASE AND SPIN RATE RECONSTITUTION METHOD

I.1 : Introduction

I.1.1 : Purpose

The main purpose of this Appendix. is to clarify the procedure for obtaining the Cluster SC and experiments spin phase and spin rate. A summary of the various aspects in the determination of the spin phase for the Cluster spacecraft is presented. The spin phase of the spacecraft is generated with two levels of accuracy, as well as the phase of any experiment. Finally the method for reduction of the random error of the spin period data as stated in Ref. IR.5. is reproduced and commented in view of the actual SRP accuracy.

I.1.2 : Scope

This appendix is intended to be used by any user of the Cluster TM who has to generate the spacecraft spin phase or spin rate, and it provides an explanation of the methods for calculation of both the spin phase of the spacecraft and the experiments. The spin rate calculation is also presented, and the user can select the level of accuracy of the obtained value.

I.1.3 : Definitions

I.1.3.1 : Definition of Spin Phase

After several reviews on the definitions and documentation the applicable definitions for the Spin Phase seems not clear.

In the Applicable Document IA.1, pg 29, the following definition is stated:

> ***The spin phase is the rotation angle of a fixed Spacecraft Meridian Plane w.r.t. the Sun Vector.***

The same definition is stated in Applicable Document IA.2. This definition does not state the axis with respect to which the fixed Spacecraft Meridian plane has to be rotated, and, for data delivery, a reference plane in the Spacecraft.

In Ref. IR.2 (Section A3.I), the following definition of the phase angle is used:

> ***(Instantaneous) spin phase: angle that the SC has "spun" through since the last Meridian Sun pulse was received.***

This second definition relates the phase with the passage of the Sun through the Sun Sensor Meridian Slit.

The spin phase should be defined to allow the transformation of any spacecraft fixed point or any fixed direction into inertial space at a given instant. In this situation and considering that the determined attitude is that of the maximum principal axis of inertia, the spin phase rotation should be defined around this axis, and the reference plane to define the spacecraft phase, to be a meridian plane with respect to this axis.

With this consideration, the phase of the SC that FD Attitude Determination will provide is:

> ***Rotation angle of the half-plane defined by the +Z_{SR} and +X_{SR} Spin Reference System axes (Section** *Error! Reference source not found.***), around the maximum principal axis of inertia (+Z_{SR}) from the time when the Sun direction was contained in this plane***.

After nutation and oscillations have been damped out, the actual spin axis, i.e. the angular velocity vector, and the angular momentum vector are aligned with the maximum principal axis of inertia.

A similar definition can be used for the phase of any point (position vector) or any direction (free vector) fixed in the Spacecraft:

> ***Rotation angle of the half-plane defined by the maximum principal axis of inertia and the particular point or the free vector (meridian semi-plane containing the SC point or the free vector), around the maximum principal axis of inertia from the time when the Sun direction was contained in this plane.***

The Sun Reference Pulse is used to determine the direction of the Sun in the spacecraft system. When this Sun Reference Pulse is generated the Sun is contained in the plane defined by the Sun Sensor Meridian Slit Plane. At this time the SC phase is $\_{SRP}$.

The phase of the Spacecraft when no dynamic modes are considered is defined by:

$$\varphi_{SC} = \varphi_{SRP} + \omega \bullet (t - T_{SRP}) \qquad \textbf{(EQ. 3)}$$

I.1.3.2 : Reference Systems

Three reference systems are used for determination of the Spacecraft phase:

I. Body-Build System, $\{X_B, Y_B, Z\}$ as defined in Applicable Docs. IA.1 and IA.2.

2. Attitude System, $\{X_A, Y_A, Z_A\}$ obtained as a rotation of the previous one, in order to have the Z_A axis aligned with the nominal spin axis:

- $X_A \rightarrow Y_B$
- $Y_A \rightarrow Z_B$
- $Z_A \rightarrow X_B$

$$[L_{A-B}] = \begin{bmatrix} 0\,1\,0 \\ 0\,0\,1 \\ 1\,0\,0 \end{bmatrix}$$

3. Spin Reference System, $\{X_{SR}, Y_{SR}, Z_{SR}\}$ aligned with the maximum principal inertia axis, suitable for description of the spin phase, and defined by:

- Z_{SR} : in the maximum principal inertia moment axis (near to the $+X_B$ axis)

- X_{SR} : normal to Z_{SR} in the plane defined by X_B-Y_B or X_A-Z_A

-Y_{SR} : completes the right handed frame

Translation to this system from the attitude system is obtained by the Euler rotation 2-1 (Ψ_2, Ψ_1), as shown in Figure I-1, and is expressed by:

$$[L_{SR-A}] = \begin{bmatrix} \cos\Psi_2 & 0 & -\sin\Psi_2 \\ \sin\Psi_2 \bullet \sin\Psi_1 & \cos\Psi_1 & \sin\Psi_1 \bullet \cos\Psi_2 \\ \sin\Psi_2 \bullet \cos\Psi_1 & -\sin\Psi_1 & \cos\Psi_1 \bullet \cos\Psi_2 \end{bmatrix} \qquad \textbf{(EQ. 2)}$$

NOTE: we can assume for all the systems the same origin. The displacement of the centre of mass is negligible when looking at the Sun, which is at infinity. When looking at neighbourhood a correction must be applied for this reason (section I.5)

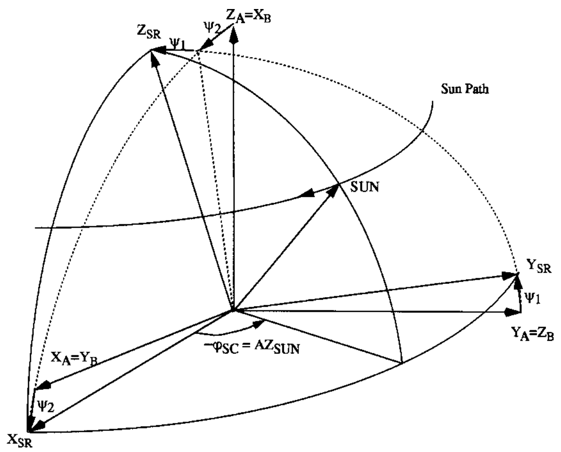


Figure I-1Tilt and Spacecraft Phase Definition

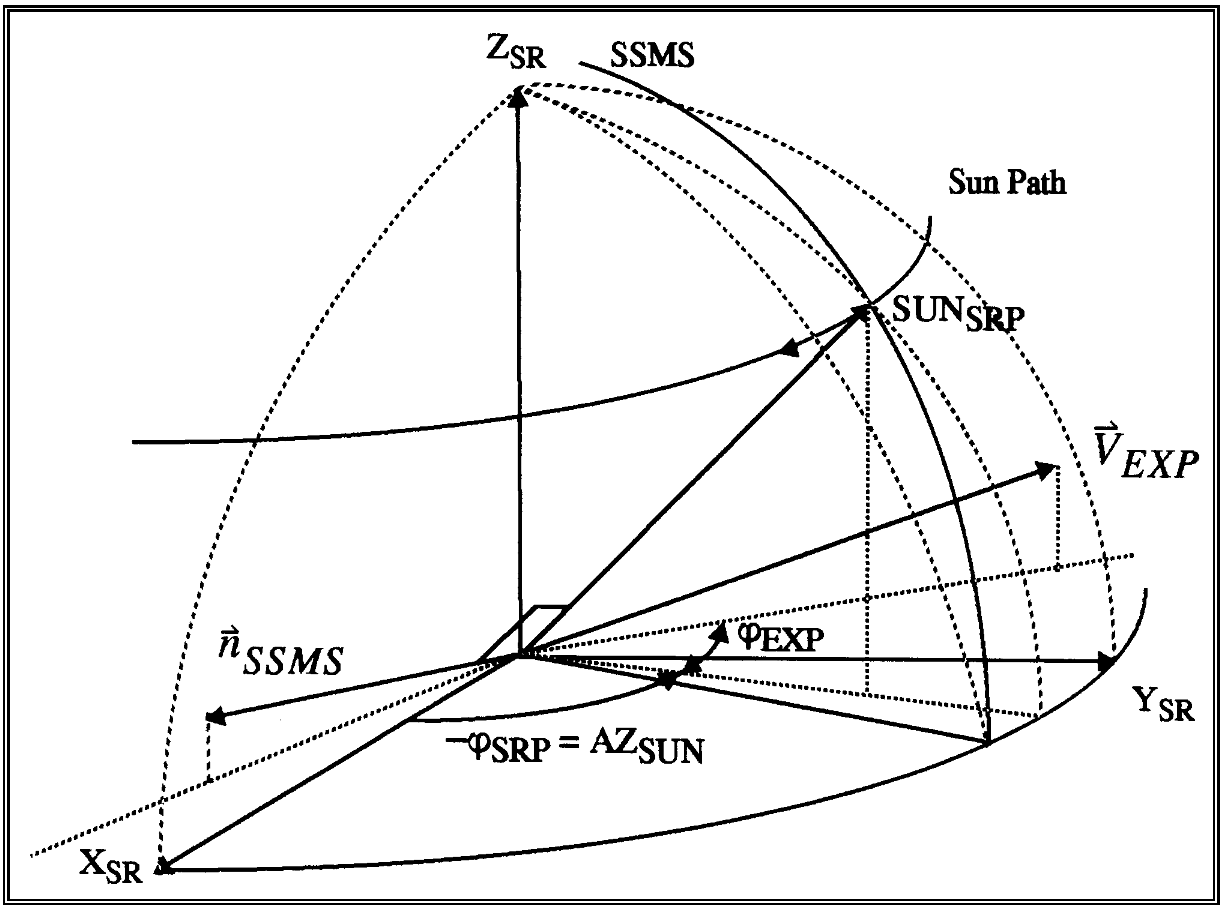


Figure I-2: Spacecraft and Experiment Phase at Sun Reference Pulse

I.1.4 : Applicable Documents

IA.l "CLUSTER System Design Specification". CL-DOR-DS-0001. Issue 7, 31-03-1993.

IA.2 "CLUSTER Interface Requirements Specification". CL-DOR-IS-0001. Issue 7A, 02-04-1993.

IA.3 "CLUSTER Flight Dynamics Support Requirements Compilation", CL-ESC-RS-0304 / OAD-CLU-RC, Issue 2, Jan. 1993.

I.1.5 : Reference Documents

IR.I "CLUSTER Quarterly Budget Report", CL-DOR-RP-0012, Issue 12, 1611-1993

IR.2 "CLUSTER ADCS Design Specification", CL-BAe-DS-843, Issue 3, May 1992

IR.3 "CLUSTER AOCMS Error Budget Derivations". CL-BAe-TN-922, Issue 2, 92-10-22.

IR.4 "CLUSTER UTC Time Correlation". CL-ESC-TN-0006, Issue 2, August 1992.

IR.5 L. Fraiture. "ATTITUDE ASPECTS OF CLUSTER IN THE ROUTINE PHASE". OAD Working Paper 438. February 1991

I.1.6 : Overview

After many considerations about the necessity of inclusion of the spin phase data in the Auxiliary Files to be provided by FD Attitude Determination system, this inclusion seems unnecessary considering the requirements on spin phase accuracy (even for non-nominal working of the system). See especially Applicable Docs. IA.I, IA.2, IA.3, and Ref. IR.I.

To be prepared for a possible contingency, the Project Scientist has required the delivery of some corrections and data to be applied to the experiment phases.

The spin phase concept used along the present document and to be used in future interfaces is clarified along section I.1.3.1 in order to have a correct interpretation of the results obtained with the presented algorithms.

With the stated definition of the spin phase, different ways of obtaining this spin phase of both the Spacecraft (defined a reference meridian plane) and any vector fixed to it are developed.

The evaluation of the Spacecraft phase is generated with two levels of accuracy, depending on the correction or not, of the predictable errors in the datations and the spin phase.

The spacecraft spin phase will be delivered by FD with the most accurate expression of section I.3 but this spin phase expression will be valid only while the Solar Aspect Angle and principal axis tilt is unchanged, otherwise the phase of the Sun Reference pulse (and the SC phase) will change. As far as the Solar Aspect Angle is very slowly changing, and the principal axis tilt depends only on the spacecraft mass properties, it is not foreseen that this phase will change significatively during a day, except for manoeuvres or eclipse conditions.

For determination of the phase of experiments the same approach is provided, and the user will be able to use both levels of accuracy from the algorithms stated in sections I.4 and I.5

Finally an evaluation of the Spin Phase error to be obtained from the actual Sun Sensor Meridian Slit datation accuracy has been presented. It is possible to obtain the required accuracy under nominal conditions and operational spin rate with two consecutive Sun Sensor datations with a confidence level of 99.73%.

Anyway a smoothed spin rate algorithm is included in order to obtain the spin rate from the TM with higher accuracy.

I.2 : Nominal Spacecraft Phase from the TM

As stated in the section I.1 the spacecraft phase is determined from the Sun Reference Pulse information plus the time since this event. The determination of the spacecraft phase for the nominal case is explained in this chapter. Nominal case is considered when the following errors in the generation of the measurements are not present:

- Sensor Meridian Slit is meridian (azimuth alignment to be considered)
- Spacecraft is rotating around the nominal spin axis.
- There are no dynamic modes in the spacecraft.
- Electronics do not introduce any additional error.

In this situation the spacecraft phase is directly:

$$(\varphi_{SRP})_0 = -AZ_{SSMS} + b_{SRP} = -AZ_n - \frac{\Pi}{2} + b_{SRP} \quad \textbf{(EQ. 3)}$$

$$(\varphi_{SC})_0 = -AZ_n - \frac{\Pi}{2} + b_{SRP} + \omega \bullet (t - T_{SRP}) \quad \textbf{(EQ. 4)}$$

Computation of this phase requires the following data:

- Sun Sensor Meridian Slit Alignment: (AZ_{SSMS}) or (AZ_n) as obtained from alignments and calibration. (AZ_n) delayed $\pi/2$ w.r.t (AZ_{SSMS}), expressed in any Attitude or Spin Reference System, since both are considered equal (tilt not considered).

- The predictable bias (b_{SRP}) will be added by Flight Dynamics to the obtained phase. This bias has been stated in Applicable Document IA.1, Fig. 5.11-1, obtained from Ref. IR.3. This error is derived from truncation and time delays and will change depending on the spin rate. The value in IR.3 is 3.8147+3µs, corresponding at 20 rpm to (0.00082^o+0.00046^o). In a general case the phase correction will be:

$$b_{SRP} = \left(\frac{1s}{131072} \, x \, \frac{1}{2} + 3 \, x \, 10^{-6} \, s \right) x \, 360^o \bullet \omega(r \;\; p \; s) \quad \textbf{(EQ. 5)}$$

- Phase rotated since last Sun Reference pulse:

  a) From the Spin Segment Clock Count (SSCC)

$$\omega \bullet (t - T_{SRP}) = n_{SSCC} \bullet \frac{360^o}{16384} \; \textbf{(EQ. 6)}$$

  b) From the datation of the Sun Reference Pulse T_{SRP}

  - Take the Sun Sensor Meridian Slit Datation before and after the required time, and compute:

$$\omega \bullet (t - T_{SRP}) = (HFC_t - HFC_{SRP}1) \bullet \frac{360^o}{HFC_{SRP}2 - HFC_{SRP}1} \quad \textbf{(EQ. 7)}$$

- HFC: S/C High Frequency Clock Count
- Subindex SRPI: related to first Sun Reference Pulse before the required phase
- Subindex SRP2: related to first Sun Reference Pulse after the required phase
- t (and subindex t): time at which the phase is required.

- The obtained phase will not be correlated with UTC time, but with SC clock (HFC). For correlation with UTC the use of Ref. IR.4 considerations will be required.

In a non-nominal situation this phase value will include errors coming from the assumptions stated at the beginning of this section:

a) Sun Sensor Meridian Slit tilt

b) Principal axis tilt

c) Dynamic modes

d) Attitude Uncertainty

e) Sun Reference Pulse delays

f) Ground Segment errors

g) Spin period calculation/interpolation (...)

Applicable Document IA.1 makes a systematic evaluation of most of these errors.

In the next chapter some of these errors are evaluated for correction of the spin phase.

I.3 : Corrections to the SRP Provided by Flight Dynamics

In the real case there will be several errors in the Spin Reference Pulse datation and consequently in the spacecraft spin phase, as stated in previous section. Some of these errors can be predicted, and corrected, especially the principal axis tilt, the Sun Sensor meridian slit tilt, and some spurious biases in the Sun Sensor datation.

The phase correction coming from the tilt (of the principal axis and the slit), are dependent on the Solar Aspect angle, and will be changing with it. FD will provide the SC phase of the SC at the Sun Reference Pulse, computed with the last obtained Solar Aspect Angle, and will include this value in the "SATT" file ("SCPHAS" column). The algorithms to derive this phase will be presented here below.

The effects to correct will be:

a) Sun Sensor Meridian Slit tilt

b) Principal axis tilt (Ψ_2, Ψ_1)

c) Sun Reference Pulse additional delays (_b_{SRP})

Sun Sensor Meridian Slit tilt and principal axis tilt can be grouped together, because the effect is similar, while unforeseen delays in SRP will be a single addition to the SC phase at SRP.

For Tilt corrections, the Spin phase is calculated in the Spin Reference System, where the expression of the phase is much simpler than in the Spacecraft frame. In order to express phase in this system, both Sun Sensor normal, and Sun direction are expressed in this frame.

The results obtained in this frame:

- In this frame (taking into account that Z_{SR} axis is the maximum principal inertia axis), the phase of the spacecraft is directly: $\_{sc} = -(AZ_S)_{SR}$

- When the Sun is in the plane of the Sun Sensor Meridian plane the relative situation of Sun w.r.t. spacecraft is: $(\vec{n}_{SSMS})_{SR} \bullet (\vec{d}_S)_{SR} = 0$ 8

- Once we have obtained the Sun Sensor Meridian Slit normal and the Sun direction in the Spin Reference System the condition can be expressed by:

$$\cos\delta_S \bullet \cos\delta_n \bullet (\cos AZ_S \bullet \cos AZ_n + \sin AZ_s \bullet \sin AZ_n) = -\sin\delta_S \bullet \sin\delta_n \qquad \textbf{(EQ. 8)}$$

$$AZ_S = AZ_n + \begin{cases} acos(-\tan\delta_S \bullet \tan\delta_n) \\ -acos(-\tan\delta_S \bullet \tan\delta_n) \end{cases}$$

Considering that with the defined slit normal the Sun must have approx. $\pi/2$ larger azimuth than the slit normal, the appropriate sign is "+" (*acos* taken from 0 to π)

$$AZ_S = AZ_n + acos(-\tan\delta_S \bullet \tan\delta_n) \qquad \textbf{(EQ. 9)}$$

The transformation from the Body-Build Reference System to the Spin Reference System is obtained with transformation matrix (EQ. 2) in the following way:

$$\{X\}_{(SR)} = [L_{SR-A}] \bullet [L_{A-B}] \bullet \{X\}_B \qquad \textbf{(EQ. 10)}$$

X_B is a generic free vector in the Body-Build Reference System (in this case the sun sensor slit normal)

When using these expressions the Sun Sensor Slit normal azimuth to use will be the obtained with the normal in the Spin Reference System:

$$AZ_n = \mathrm{ATAN2}((n_y)_{SR}, (n_x)_{SR}) \qquad \textbf{(EQ. 11)}$$

The required declinations are obtained from:

$$\tan\delta_S = \frac{S_Z}{\sqrt{1 - S_Z^{\,2}}} = \tan\left(\frac{\pi}{2} - SAA\right)$$

(EQ. 12)

$$\tan\delta_n = \frac{n_Z}{\sqrt{1 - n_Z^{\,2}}}$$

In order to apply these corrections it is required to know the principal axis tilt rotation angles (Ψ2 and Ψ1, as specified in SATT file, columns "TPSI_2" and "TPSI_1"), the Sun Sensors normal, and the Solar Aspect angle or the Sun direction.

The final expression of the phase for this case is similar to that obtained in Section I.2 with some modifications:

$$\varphi_{SC} = -(A\,Z_n\,)_{SR} - acos\,(-\tan\delta_S \bullet \tan\delta_n\,) + b_{SRP} + \_b_{SRP} + \omega \bullet (t - T_{SRP}\,) \qquad \textbf{(EQ. 13)}$$

$$\varphi_{SRP} = -(A\,Z_n\,)_{SR} - acos\,(-\tan\delta_S \bullet \tan\delta_n\,) + b_{SRP} + \_b_{SRP} \qquad \textbf{(EQ. 14)}$$

The phase obtained with this expression for the Sun Reference Pulse _SRP will be included in the "SATT" file ("SCPHAS" column). From this value the user will have to derive the experiment phase by means of the formulation given in the next two chapters.

I.4 : Nominal Spin Phase Reconstitution by the Experiments

Once defined the phase of the spacecraft as stated above, the direction of any vector in the spacecraft related to the Sun can be obtained related to this.

Any experiment at any time will have the following expression:

$$\varphi_{\mathrm{EXP}} = \varphi_{SC} + A\,Z_{\mathrm{EXP}} \qquad \textbf{(EQ. 15)}$$

The phase difference between spacecraft and experiment is determined by the

Azimuth difference in the Spin Reference system.

Considering for a first approximation the case when no tilts are present, the phase of the experiment is given by:

$$(\varphi_{\mathrm{EXP}})_0 = -AZ_{SSMS} + b_{SRP} + \omega \bullet (t - T_{SRP}) + AZ_{\mathrm{EXP}}$$

(EQ. 16)

$$(\varphi_{\mathrm{EXP}})_0 = (AZ_{\mathrm{EXP}} - AZ_{SSMS}) + b_{SRP} + \omega \bullet (t - T_{SRP})$$

Where the AZ_{EXP} and AZ_{SSMS} are measured in $\{X_A, Y_A, Z_A\}$ reference system

The evaluation of this phase at Sun Reference Pulse is easily obtained from that of the SC:

$$(\varphi_{\mathrm{EXP}})_0 = (\varphi_{SC})_0 + AZ_{\mathrm{EXP}}$$ **(EQ. 17)**

The corresponding phase at any time can be derived using the methods stated in (EQ. 6) or (EQ. 7), depending on the available information and required accuracy.

I.5 : Precise Spin Phase Reconstitution by the Experiments

As stated for the Spacecraft phase, for the experiments phase the precise spin phase reconstitution requires to consider the spacecraft principal axis tilt (additionally to the considerations stated above for the Sun Reference Pulse), and the centre of mass shift for the cases when the measurements are taken to the neighbourhood (not at infinity).

In this case the situation is also simplified using the Spin Reference Frame, where the phase of the experiment relative to the phase of the spacecraft is directly the azimuth of the experiment.

Under this consideration the first step is to translate the vector which phase the user wants to obtain:

$$\{\vec{V}\}_{SR} = [L_{SR-A}] \bullet [L_{A-B}] \begin{cases} \{\vec{R} - \vec{R}_{com}\}_B & \text{at neighbourhood} \\ \{\vec{V}\}_B & \text{at infinity} \end{cases}$$

And all the values are known:

- $[L_{SR-A}]$ and $[L_{A-B}]$ from section I.1.3.2 with $\Psi 2$ and $\Psi 1$ as specified in SATT file, columns "TPSI_2**" and "**TPSI_1**".**

- $\{\vec{V}\}_B$ 19is the direction at infinity of which the user wants to know its phase, expressed in Body-Build Reference System.

- $\{\vec{R}\}_B$ 20is the position at the neighbourhood that the user wants to know its phase, expressed in Body-Build Reference System.

- $\{\vec{R}_{com}\}_B$ 21is the centre of mass position, expressed in Body-Build Reference System, as specified in SATT file, columns "COMSHF (3)**".**

With the new vector it is obtained the azimuth:

$$AZ_{\mathrm{EXP}} = \mathrm{ATAN2}\left((\mathrm{V_y})_{\mathrm{SR}}, (\mathrm{V_x})_{\mathrm{SR}} \right) \quad \textbf{(EQ. 19)}$$

The phase of this element is obtained in the same way as stated in previous section:

$$\varphi_{\mathrm{EXP}} = \varphi_{SC} + AZ_{\mathrm{EXP}} \quad \textbf{(EQ. 20)}$$

I.6 : Spin Rate Data

I.6.1 : Instantaneous Spin Rate Data

The requirements on the accuracy of the spacecraft spin rate knowledge (accuracy better than 0.1 ms required) is higher than the accuracy that the Sun Sensor Meridian Slit datations can provide with two consecutive datation (requirements in (σ_{SRP} = 0.01°).

By means of the difference between two consecutive datations of the Sun Sensor Meridian Slit the user will obtain uncertainty (σ_T is the uncertainty in the random noise of the SRP datation):

$$\sigma = \sqrt{2} \bullet \sigma_T \quad \textbf{(EQ. 21)}$$

The biases are cancelled, and the obtained distribution is the addition of two normal distributions.

For a noise in SRP angle (1σ)=σ_{SRP} the maximum error (99.73% confidence) in the spin period is defined by:

$$\varepsilon = 3 \bullet \sqrt{2} \bullet (\sigma_{SRP})_{angle} \bullet \frac{1}{\omega} \quad \textbf{(EQ. 22)}$$

For a noise below requirements (σ_{SRP} = 0.01^o), at nominal spin rate (15 rpm), the maximum error is: 0.47 ms, out of requirements.

The actual error budget for random noise in SRP is 0.00184^o (1σ) (Applicable Doc. IA.1, Fig. 5.12-1.), and consequently the maximum error with a confidence level of 99.73%, at nominal spin rate (15 rpm) is: 0.0864 ms, inside requirements. Considering the range of the operational spin rate (+/- 10%), the error in the spin period will be always below 0.096 ms with a confidence level of 99.73%.

In this Applicable Document IA.1 the given errors are obtained for 20 rpm, but the only significative random error for the Sun Reference Pulse is that coming from the Sun Sensor and it is an error in phase (degrees), consequently this error can be taken independent from the spin rate.

I.6.2 : Smoothed Spin Rate Data

For any requirement of a better accuracy, or for spin rate determination under different conditions (lower spin rate, higher SRP random noise, ...), it will be required to process a group of n+1 SRP datations, which will provide the following spin period expression (Ref. IR.5):

$$T = \frac{t_n - t_0}{n}$$ **(EQ. 23)**

For this case the maximum error is reduced to:

$$\frac{3 \bullet \sqrt{2} \bullet \sigma_r}{n} = \frac{3 \bullet \sqrt{2} \bullet \sigma_{SRP}}{n} \bullet \frac{1}{\omega}$$ **(EQ. 24)**

and consequently the user will have to select the appropriate number of cycles to process in order to obtain the required accuracy.

As an example to obtain the required accuracy at 3 rpm (99.73% confidence), it will be required to use 5 consecutive cycles (6 consecutive datations).