Consultative Committee for Space Data Systems

DRAFT RECOMMENDATION FOR SPACE DATA SYSTEM STANDARDS

Specification for the Formal Definition and Transfer Phase of a Producer-Archive Interface

CCSDS 651.1-W-01

DRAFT White BOOK

September 2004

AUTHORITY

Issue:	White book, Issue 1
Date:	September 2004
Location:	

(WHEN APPROVED FOR PUBLICATION AS A RECOMMENDATION, THE FOLLOWING TEXT WILL APPEAR)

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorisation of CCSDS Recommendations is detailed in Reference [E1], and the record of Agency participation in the authorisation of this document can be obtained from the CCSDS Secretariat at the address below.

This Recommendation is published and maintained by:

CCSDS Secretariat
Program Integration Division (Code-OI)
National Aeronautics and Space Administration
Washington, DC 20546, USA

STATEMENT OF INTENT

(WHEN APPROVED FOR PUBLICATION AS A RECOMMENDATION, THE FOLLOWING TEXT WILL APPEAR)

The Consultative Committee for Space Data Systems (CCSDS) is an organisation officially established by the management of the member space Agencies. The committee meets periodically to address data system problems that are common to all participants and to formulate sound technical solutions to these problems. Inasmuch as participation in the CCSDS is completely voluntary, the results of the committee are termed **Recommendations** and are not considered binding on any Agency.

This **Recommendation** is issued by, and represents the consensus of, the CCSDS Plenary body. Agency endorsement of **Recommendations** is entirely voluntary. Endorsement, however, indicates the following understandings:

- Whenever an Agency establishes a CCSDS-related **standard**, this **standard** will be in accordance with the relevant **Recommendation**. Establishing such a **standard** does not preclude other provisions which an Agency may develop.

- Whenever an Agency establishes a CCSDS-related **standard**, the Agency will provide other CCSDS member Agencies with the following information:

    — The **standard** itself.

    — The anticipated date of initial operational capability.

    — The anticipated duration of operational service.

- Specific service arrangements shall be made via memorandum of agreement. Neither this **Recommendation** nor any ensuing **standard** is a substitute for a memorandum of agreement.

No later than five years from its date of issuance, this **Recommendation** will be reviewed by the CCSDS to determine whether it should: (1) remain in effect without change; (2) be changed to reflect the impact of new technologies, new requirements, or new directions; or (3) be retired or cancelled.

In those instances when a new version of a **Recommendation** is issued, existing CCSDS-related Agency **standards** and implementation are not negated or deemed to be non-CCSDS compliant. It is the responsibility of each Agency to determine when such **standards** or implementation are to be modified. Each Agency is, however, strongly encouraged to direct planning of its new **standards** and implementations towards the later versions of this **Recommendation**.

FOREWORD

(WHEN THIS RECOMMENDATION IS FINALIZED, IT WILL CONTAIN THE FOLLOWING FOREWORD:)

This Recommendation is a technical Recommendation providing the XML implementation for the Abstract Syntax defined in the *DEDSL—Abstract Syntax* Recommendation (see reference [1]) in order to provide a computer processable standardisation of the expression of the semantic information which is to be carried with data.

Through the process of normal evolution, it is expected that expansion, deletion or modification to this document may occur. This Recommendation is therefore subject to CCSDS document management and change control procedures defined in reference [B1]. Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

Questions relative to the contents or status of this document should be addressed to the CCSDS Secretariat at the address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- British National Space Centre (BNSC)/United Kingdom.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Etudes Spatiales (CNES)/France.
- Deutsche Forschungsanstalt für Luft- und Raumfahrt e.V. (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- National Aeronautics and Space Administration (NASA)/USA.
- National Space Development Agency of Japan (NASDA)/Japan.
- Russian Space Agency (RSA)/Russian Federation.

Observer Agencies

- Australian Space Office (ASO)/Australia.
- Austrian Space Agency (ASA)/Austria.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- Centro Tecnico Aeroespacial (CTA)/Brazil.
- Chinese Academy of Space Technology (CAST)/China.
- Communications Research Laboratory (CRL)/Japan.
- Danish Space Research Institute (DSRI)/Denmark.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Federal Service of Scientific, Technical & Cultural Affairs (FSST&CA)/Belgium.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Industry Canada/Communications Research Centre (CRC)/Canada.
- Institute of Space and Astronautical Science (ISAS)/Japan.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- MIKOMTEK: CSIR (CSIR)/Republic of South Africa.
- Korea Aerospace Research Institute (KARI)/South Korea.
- Ministry of Communications (MOC)/Israel.
- National Oceanic & Atmospheric Administration (NOAA)/USA.
- National Space Program Office (NPSO)/Taiwan.
- Swedish Space Corporation (SSC)/Sweden.
- United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

Document	Title	Date	Status/Remarks
CCSDS 651.1-W-0.1	Recommendation for Space Data System Standards: Producer-Archive Interface implementation	September 2004	Original Draft

CONTENTS

Section Page

TMRENVOIPAGE

1 INTRODUCTION

1.1 PURPOSE AND SCOPE

The purpose of this recommendation is to provide a standard method for formally define the digital information objects to be transferred by an information Producer to an Archive and for effectively transferring these objects in the form of **Submission Information Packages (SIPs).**

This recommendation fits into the context defined by:

- The 'Producer Archive Interface Methodology Abstract Standard' recommendation (PAIMAS) (See reference [1]).
- The 'Reference Model for an Open Archival Information System' recommendation (OAIS) (See reference [2]).

The PAIMAS Recommendation (See Reference [1]) defines a methodology based on the four following phases: Preliminary, Formal Definition, Transfer, Validation.

This Recommendation applies specifically to implementation of the Formal Definition Phase and the Transfer Phase.

This Recommendation should help in the automation and the implementation of the validation phase.

The proposed method may however be used, to some extent for the Preliminary Phase.

This Recommendation does not exclude other PAIMAS implementation Recommendations.

1.2 APPLICABILITY

The implementation method defined in this Recommendation applies both to information Producers and to Archives to which this information is to be transferred. These Archives are assumed to comply with the OAIS Reference Model.

It may be applied to Archives responsible for preserving both physical objects and digital objects.

This Recommendation assumes that both the Producer and the Archive agree on the methods and particularities related to their context..

1.3 RATIONALE

This recommendation aims to overcome the many difficulties encountered during transactions between information Producers and the Archives.

Concerning the Formal Definition Phase, this Recommendation should enable:

- the Producer to have a very precise, unambiguous definition of the different digital objects to be produced, of the form and possibly the order in which they should be delivered,
- the Archive to be sure that the digital objects which are to be transferred to it will enable it to build Archival Information Packages which have all of the characteristics defined in the OAIS Reference Model,
- the respective Managers of the Producer and the Archive to be fully aware of all details of their commitments in terms of hardware, human and financial resources.

Concerning the Transfer phase, this Recommendation should enable a high degree of automation and verification of the transfer process.

1.4 CONFORMANCE

To be completed.

1.5 DOCUMENT STRUCTURE

To be completed later.

1.6 DEFINITIONS

1.6.1 ACRONYMS AND ABBREVIATIONS

This sub-section defines the acronyms and abbreviations which are used throughout this Recommendation:

AIP	Archival Information Package
ASCII	American Standard Code for Information Interchange
CCSDS	Consultative Committee for Space Data Systems
CDO	Complementary Data Object
DED	Data Entity Dictionary
DEDSL	Data Entity Dictionary Specification Language
DO	Data Object
DTD	Document Type Definition
EAD	Encoded Archival Description
EAST	Enhanced Ada Subset
ID	Identifier
ISO	International Organization for Standardization
MOT	Model of Objects to be Transferred
OAIS	Open Archival Information System
PAIMAS	Producer Archive Interface Methodology Abstract Standard
PDI	Preservation Description Information

PDF	Portable Document Format
RM	Reference Model
SIP	Submission Information Package
UML	Unified Modelling Language
XFDU	XML Formatted Data Units
XML	eXtensible Markup Language

1.6.2 GLOSSARY OF TERMS

Following is a short glossary of the OAIS terminology indispensable for this document. The terminology used is fully defined in references [2] and [1], except the definitions printed in italics. Only brief definitions are provided here. This terminology does not seek to replace existing terminology in the various domains related to archiving. Each domain should be able to apply this methodology while retaining their specific terminology.
When first used in the text, the terms defined in the terminology are shown in bold.

Archival Information Package: An Information Package, consisting of the Content Information and the associated Preservation Description Information (PDI), which is preserved within an OAIS.

Archive: An organization that intends to preserve information for access and use by a Designated Community.

Consumer: The role played by those persons, or client systems, who interact with OAIS services to find preserved information of interest and to access that information in detail. This can include other OAISs, as well as internal OAIS persons or systems.

Content Data Object: The Data Object, that together with associated Representation Information, is the original target of preservation.

Content Information: The set of information that is the primary target for preservation. It is an Information Object comprised of its Content Data Object and its Representation Information. An example of Content Information could be a single table of numbers representing, and understandable as, temperatures, but excluding the documentation that would explain its history and origin, how it relates to other observations, etc.

Data Dictionary: A formal repository of terms used to describe data.

Data Entity Dictionary (DED): A collection of semantic definitions of various data entities, together with a few mandatory and optional attributes about the collection as a whole. Data Entity Dictionaries may pertain to a single product, i.e., all the data entities within a single product are described in a corresponding single dictionary, or the Data Entity Dictionary may be a discipline-oriented dictionary that holds a number of previously defined data entity definitions which may be used by data designers and users as references.

Data Object: Either a Physical Object or a Digital Object.

Data Submission Session: A delivered set of media or a single telecommunications session that provides data to an OAIS. The Data Submission Session format/contents is based on a data model negotiated between the OAIS and the Producer in the Submission Agreement. This data model identifies the logical constructs used by the Producer and how these are represented on each media delivery or in the telecommunication session.

Descriptor (OR Object Descriptor): This is an information unit for describing a set of characteristics for a given Data Object. A Descriptor may come in the form of a model; in this case, it is used to identify the object category which can then be described as well as the attributes for this description. The Descriptor may have the state of an instance; in this case, it describes an object instance and supplies the attribute values defined in the Descriptor model.

Designated Community : An identified group of potential Consumers who should be able to understand a particular set of information. The Designated Community may be composed of multiple user communities.

EAST: The EAST language is a CCSDS and ISO norm. EAST offers means to describe the syntax of a data file, including:

- the fields in which it can be decomposed;
- structure (simple or composite);
- type (integer, real, enumerated, array, record, list);
- range (min value, max value);
- coding (ASCII, binary);
- location (rank, length);
- optionality (mandatory or not and, if not, presence condition);
- eventually, variable dimension (for arrays).
-

Fixity Information: The information which documents the authentication mechanisms and provides authentication keys to ensure that the Content Information Object has not been altered in an undocumented manner.

Identifier : An XML CDATA, that designates something. (from DEDSL)

Information: Any type of knowledge that can be exchanged. In an exchange, it is represented by data. An example is a string of bits (the data) accompanied by a description of how to interpret a string of bits as numbers representing temperature observations measured in degrees Celsius (the Representation Information).

Information Object: A Data Object together with its Representation Information.

Information Package: The Content Information ans associated Preservation Description Information which is needed to aid in the preservation of the Content Information.

Ingest: The OAIS entity that contains the services and functions that accept Submission Information Packages from Producers, prepares Archival Information Packages for storage, and ensures that Archival Information Packages and their supporting Descriptive Information become established within the OAIS.

Meta-data: Data about the content, the quality, condition and other characteristics of the data (from the FGDC Standards Reference Model, reference [**Erreur! Source du renvoi introuvable.**

Model : A data entity described independently from any instance in a data product and corresponding to a re-usable data entity definition from which other data entities may inherit the attributes and apply some specialization rules. (from DEDSL)

Packaging Information: The information that is used to bind and identify the components of an Information Package. For example, it may be the ISO 9660 volume and directory information used on a CD-ROM to provide the content of several files containing Content Information and Preservation Description Information.

Preservation Description Information (PDI): The information which is necessary for adequate preservation of the Content Information and which can be categorized as Provenance, Reference, Fixity, and Context Information.

Producer: The role played by those persons or client systems who provide the information to be preserved. This can include other OAISs or internal OAIS persons or systems.

Producer-Archive Project: A Producer-Archive Project is a set of activities and the means used by the information Producer as well as the Archive to ingest a given set of information into the Archive.

Representation Information: The information that maps a Data Object into more meaningful concepts. An example is the ASCII definition that describes how a sequence of bits (i.e., a Data Object) is mapped into a symbol.

Slip : A data object that conveys information related to another data object which is transferred between two entities. This information should (or shall ?) allow the addressee to identify, locate and if needed process the various transferred bit sequences which compose this data object.

Submission Agreement: The agreement reached between an OAIS and the Producer that specifies a data model for the Data Submission Session. This data model identifies format/contents and the logical constructs used by the Producer and how they are represented on each media delivery or in a telecommunication session.
In the framework of this abstract methodology, the Submission Agreement will also deal with other aspects such as validation, change management and schedule.

Submission Information Package (SIP): An Information Package that is delivered by the Producer to the OAIS for use in the construction of one or more AIPs.

Transfer: The act involved in a change of physical custody of SIPs. This definition is derived from the International Council on Archives [ICA] Dictionary on Archival Terminology

The terms '**class**', '**association**', **and** '**aggregation**' refer to UML terminology.

1.7 REFERENCES

The following documents contain provisions (through references within this text) which constitute provisions of this Recommendation. At the time of the publication the indicated editions were valid. All documents are subject to revision, and users of this Recommendation are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently available CCSDS Recommendations.

[1] *Producer Archive Interface methodology Abstract Standard* Recommendation for Space Data System Standards, CCSDS 651.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, May 2004.

[2] *Reference Model for an Open Archival Information System (OAIS)* Recommendation for Space Data System Standards, CCSDS 650.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, January 2002.

[3] *Data Entity Dictionary Specification Language (DEDSL)—Abstract Syntax (CCSD0011)*. Recommendation for Space Data System Standards, CCSDS 647.1-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, June 2001.

[4] *Data Entity Dictionary Specification Language (DEDSL)— XML/DTD Syntax (CCSD0013)*. Recommendation for Space Data System Standards, CCSDS 647.3-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, January 2002.

[5] Extensible Markup Language (XML) *1.0 (Third Edition) - W3C Recommendation 6February 2004. http://www.w3.org/TR/2004/REC-xml-20040204 /*

[6] XML Schema specification, part 1 (structure) and part 2 (data types) - W3C Recommendation 2 May 2001 http://www.w3.org/XML/Schema#dev

2 GENERAL FRAMEWORK

2.1 THE PRODUCER-ARCHIVE INTERFACE METHODOLOGY

The general context is that of the transfer of digital objects from a data Producer to an Archive. The methodology for defining, specifying, performing and validating this transfer is defined in the PAIMAS Recommendation (see Reference [1]).

The term 'Producer' designates the persons and systems which supply the Archive with information to be preserved.
The Archive is an OAIS Archive. The main responsibility of an Archive is to preserve a set of information and to make this available, in an intelligible and usable form, to a defined Designated Community.

Both Producer and Archive are assumed to be involved in a **Producer-Archive Project**: A Producer-Archive Project refers to both a set of activities and the means used by the information Producer as well as the Archive to ingest a given set of information into the Archive.

The Producer-Archive interactions, in a given Producer-Archive project, consist of four different phases:

- The Preliminary Phase, also known as a pre-ingest or pre-accessioning phase, includes the initial contacts between the Producer and the Archive and any resulting feasibility studies, preliminary definition of the scope of the project, a draft of the **Submission Information Package** (SIP) definition and finally a draft **Submission Agreement**.
- The Formal Definition Phase includes completing the SIP design with precise definitions of the digital objects to be delivered, completing the Submission Agreement with precise, contractual **transfer** conditions such as restrictions on access and establishing the delivery schedule.
- The Transfer Phase performs the actual transfer of the SIP from the Producer to the Archive and the preliminary processing of the SIP by the Archive, as defined in the agreement.
- The Validation Phase includes the actual validation processing of the SIP by the Archive and any required follow-up action with the Producer. Different systematic or in-depth levels of validation may be defined. Validations may be performed after each delivery, or later, depending on the validation constraints.

Each phase is carried out in chronological order. However, the transfer phase may overlap the validation phase.
Each phase is divided into a number of sub-phases that must also be carried out in chronological order.
Each of these sub-phases is made up of one or more action tables. The action tables and the actions can be carried out in any order.

2.2 THE FORMAL DEFINITION PHASE

The Formal Definition Phase consists of 3 sub-phases:

- An organisational sub-phase during which the Producer and the Archive have to define together the work organisation and scheduling, the documents to be produced and the points which require more in-depth analysis.

- A formal definition sub-phase which is the most crucial part of the phase. It may be broken down into two parts:
  - A complete and precise definition of the data to be transferred and the way they are organised in the form of a SIP. This aspect will be formally specified in this recommendation.
  - A set of complementary points on contractual and legal aspects, the transfer schedule, validation conditions, etc. which will be covered by one or more written documents.

- Another sub-phase during which the two parties will draw up and approve a 'Submission Agreement' defining the data to be transferred, the conditions for the transfer (for instance a specification of the Data Submission Session) and the validation.

2.3 THE TRANSFER PHASE

The Transfer phase consists of two sub-phases:

- A sub-phase for implementing tests in order to validate the whole transfer chain.

- A sub-phase for performing the transfer itself.

Packaging of the data to be transformed in SIPs is also specified in this Recommendation in section 5 'Transfer Phase Specification'. The purpose at this stage is to be able to check that the objects delivered are those which are expected and that they comply with what was defined during the Formal Definition Phase.

3. GENERAL PRINCIPLES FOR IMPLEMENTING THE FORMAL DEFINITION AND TRANSFER PHASES

3.1 OBJECTIVES

There is a twofold objective which may be summed up as follows:

During the Formal Definition Phase: develop a Model for the data to be transferred later (Model of Data Objects to be Transferred – MOT), which is sufficiently precise to meet the Producer's and the Archive's needs:

- For the Producer, 'sufficiently precise' means that he knows clearly which digital objects he will have to produce, that he knows how to do it and with what means.
- For the Archive, 'sufficiently precise' means that it knows that it will be capable of creating the **Archival Information packages** (AIPs) in compliance with the OAIS Model using the Submission Information Packages (SIPs) to be transmitted to it (possibly with additional digital objects from other Producers).

In addition, it should be possible:

- to use the Model with software for automation, checking and validation purposes,
- to represent the Model visually in a way that is easy for human beings to understand,
- to adapt the Model for the most varied contexts, disciplines and organisations.

During the Transfer Phase: the means used should enable checking:

- that any digital object transferred is an expected object in the previously defined Model,
- that this object complies with the characteristics defined in the Model.

3.2 TYPOLOGY OF INFORMATION CATEGORIES

3.2.1 OBJECTS AND COLLECTIONS

We assume that it is not necessary for the Producer to know and understand the information model and the typology of the OAIS information categories in detail, for instance **Content Information, Representation Information, Preservation Description Information**, etc. Indeed, it is the Archive's task to create AIPs from the SIPs transferred and thus to establish the suitable link between a given object coming from the Producer and any particular information category in the AIP within which this object will be inserted.

Our initial suggestion is to use a simplified terminology which may be changed and enriched later if necessary, according to needs, context and domain. Here we shall simply make a distinction between the main Data Objects to be preserved, which we shall call **Data Objects** and other objects which may be related to them and which will be called **Complementary Data Objects.**

The complementary Data Objects may, for example, be syntactical descriptors for Data Objects, written in EAST language, data dictionaries describing the semantics for information contained in the Data Objects, documentary information, documentary references, etc.

Data Objects and Complementary Data Objects may be grouped together in sets which will be called respectively **Collections** and **Complementary Collections**. These collections will make it possible to organise all of the objects to be delivered into an easily understandable set.

There may be a lot of Data Objects grouped together in a Collection. For space science data, it is not rare to find object collections of several tens or hundreds of thousands of objects. On the other hand, the Complementary Data Objects may be unique objects (for instance a syntactical description of Data Objects in a given structure within a Collection) or limited in number (publications for a specific space mission for example). In this case, the role of the Complementary Object Collections will be rather to clarify the Model of Objects to be Transferred by regrouping and structuring the information, especially if the Model in question is a complex one. Such a collection might group together very different types of objects.

A collection will itself consist either of collections of lower levels (sub-collections), or objects (Data Objects or Complementary Data Objects).

The concept of **Complementary Data Object** is in no way restrictive. It makes it easier to organise the information within an easily understandable global Model. It is not compulsory. It may for instance be possible, if this appears to be a good idea, to group a science observation data file and a syntactical description file as one composite Data Object without needing Complementary Data Objects.

This typology is somewhat pragmatic and is not altogether indispensable. Unlike the OAIS Information Model, it is not intended for general or universal use. It will always be possible, for a Producer-Archive Project, to modify the terminology defined here, to complete it for complex situations or to simplify it.

3.2.2 OBJECTS AND SIPS

Data objects, in the widest meaning of the word (including Complementary Data Objects) consist of one or more digital objects, in other words, one or more bit sequences.

In this recommendation, Data Objects refer to their information content as well as to the technical characteristics of the corresponding bit sequences: size of these sequences, fixity status, grouping together of different sequences within a ‘tar’ type file, compression of bit sequences , etc.

The data are transferred between the Producer and the Archive in the form of SIPs. These SIPs are simply groups of Data Objects which are transferred in the same packet.

These Data Objects may be grouped together for very different reasons:

- It may be necessary for the Archive to have several Data Objects made available within the same package to be able to create the corresponding AIPs,
- It might be useful to optimise the transfer: if the Data Objects are very small, a decision may be taken to group several of these objects together in a single package so as not to have too many packages,
- Etc.

We shall consider each Data Object as a whole, which means in particular that the different bit sequences of which it is made up may not be separated into elements transferred in separate SIPs.

3.3 FROM FORMAL PHASE TO TRANSFER PHASE

The global formalisation process for the Formal Definition Phase and the Transfer Phase is shown in figure 1.

One of the objectives of the Formal Definition Phase is to define the Model of digital Objects to be Transferred (MOT). This MOT is used to identify and describe the objects to be transferred and must thus

define in a sufficiently clear and precise way what the Data Objects created by the Producer within a Producer-Archive Project will be. The section 3.4 on the Model of Objects to be Transferred describes precisely how to build an MOT based on Descriptors.

The possible grouping or groupings of objects within SIPs will be dealt with in another stage during the Formal Definition Phase.

The MOT is not an information organisation Model within the Archive. It should simply enable the two partners to agree on the content on the information to be transferred and should facilitate the way this transfer is done.

At this stage of negotiation between the Producer and the Archive, some characteristics of the objects to be delivered may not be known. However, both the Archive and the Producer should each have the necessary information for defining and approving this Model. The objects and the SIPs will be characterised in the following way.

During the Formal Definition Phase:

- Each digital object category to be transferred is described by a Descriptor **instance.** The Descriptor contains a set of information for describing an object and for defining its relations with other objects within the MOT. We use the term 'instance' as this instance will be derived from a Descriptor Model.

- Each object will be transferred in the form of an elementary package. This elementary package will be accompanied by a **Slip** at the time of transfer. This Slip is derived from a Model. The Descriptor instance refers to the Slip Model which characterises the content of the SIP at the time of delivery more precisely. This Slip Model identifies the indispensable complementary information for effective retrieval of bit sequences within each elementary package: these are object identifiers, the location of objects in the MOT, the reference to data containers, etc.

- Finally, the elementary packages within SIPs are grouped together by means of the SIP Descriptor instance which makes it possible to clearly identify the different SIP categories.

During the Transfer Phase:

- SIPs are used to group one or more elementary packages containing objects, described by the same Descriptor or by different Descriptors, in a single package. Each elementary package is accompanied by the Slip instance which corresponds to it.

Descriptor instances and Slip instances are created from Models adapted to the project. These Models are themselves derived from generic Models described in this document. They may be specialised with respect to the particular domain in question. This is why we distinguish between:

- 'Generic Models' which are part of this standard (we then refer to a 'metamodel'),
- 'Domain generic Models' which have been derived from previous ones using the specialisation process described in section 4 of PAIMAS,
- and 'Project specific Models' which, whenever specialisation proves necessary, are negotiated by the Archive and the Producer during the Formal Definition Phase.

All of the Descriptors are used to build the Model of Objects to be transferred, which is the basis for

negotiating a Submission Agreement.
Descriptors and Slips are described in more detail in the following sections.

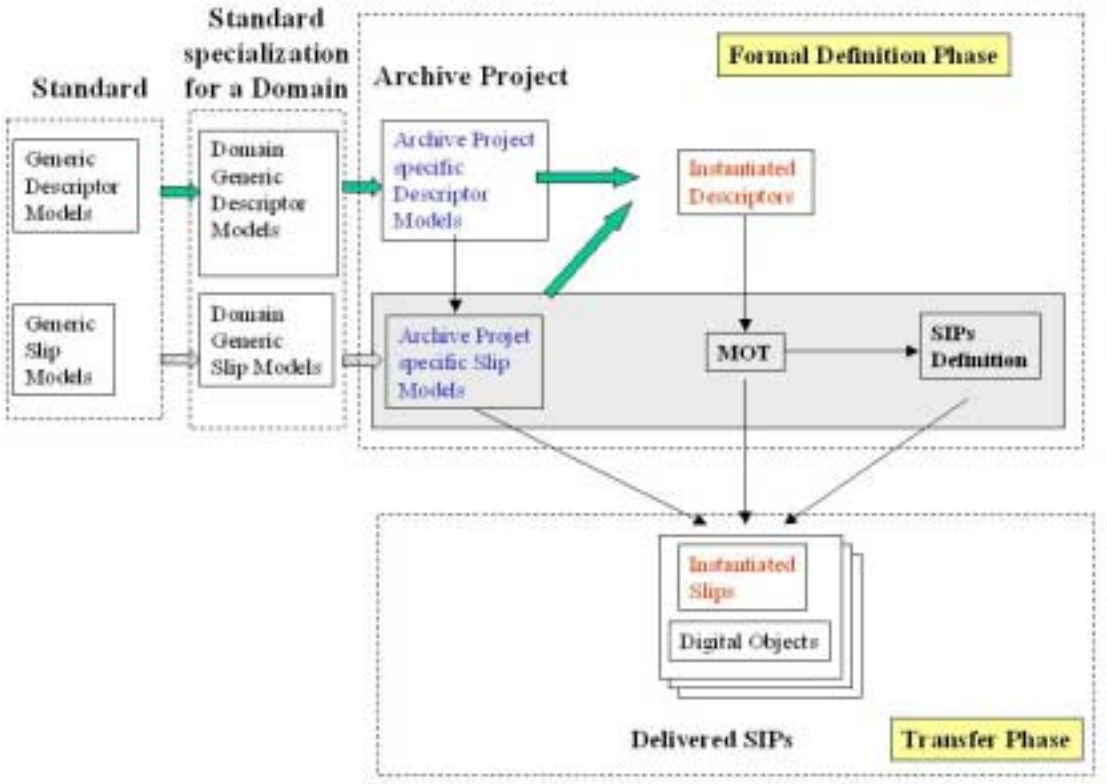

Figure 1: from Formal Definition Phase to Transfer Phase

3.4 MODEL OF OBJECTS TO BE TRANSFERRED (MOT)

3.4.1 APPROACH

The development of the digital object Model to be created by the Producer and transferred to the Archive is a difficult task.
On the one hand, this Model must be sufficiently clear and precise for both parties to be able to make a formal commitment in the framework of the 'Producer-Archive project'.
On the other hand, the idea is not to produce significant information content even before the Submission Agreement has been reached.

To describe a Collection, it would for instance be suitable:

- to define in writing what this Collection would include.
- to give this Collection a unique identifier and an explicit name.
- to specify what the parent Collection is.
- To specify that the Collection description will take the form of metadata in compliance with the ISO19115 Standard.
- etc.

But this does not, however, mean creating the metadata file, ahead of time, during the Formal Definition

Phase.

To describe the objects in this Collection, it would for instance be suitable to:

- give the essential information on the content of these objects: for instance, to indicate that the data contain the calibrated magnetic field vector expressed in a given inertial coordinate system;
- specify if necessary the spatial or temporal resolution;
- indicate whether the file format will be binary or ASCII or whether this format will comply with a specific standard format;
- give average quantitative indications of the size of the files;
- specify that a syntactical description will be created in EAST language;
- specify that a dictionary complying with the DEDSL Recommendation and describing the semantics of the fields will be made;
- etc.

This does not mean, however, creating the EAST description and the semantic description during the Formal Definition Phase.

However, in another context, the Archive might require the complete semantic description before any submission agreement.

We can see through these examples that the boundary between supplying sufficiently precise and clear information in the Model of Objects to be transferred and the supply of metadata during the Transfer Phase may vary and that both partners in the project will have to agree on the level of detail required.

The Modelling method proposed is thus very flexible and can be adapted to varying levels of detail required for given projects.

Moreover, in this Recommendation, we deliberately do not tie the description of the object (object Descriptor Instance) to the way in which this object is transferred in an elementary package (elementary package Slip Model). A given Slip Model may in fact be applied to several object categories. It is thus a good idea to separate the Slip Model from the object Descriptor.

3.4.2 THE DESCRIPTORS

The Model of Objects to be Transferred (MOT) gives a complete and overall view of all digital objects to be transferred as part of the project. These objects are described by ‘Descriptors’.

The Model will thus be structured into **Descriptor** instances. By Descriptor instance, we mean the supply of a set of information which makes it possible to build the MOT. The content of a Descriptor instance may vary greatly according to the Projects and the information available during the negotiation. These Descriptor instances are built from Descriptor Models adapted for the project. These project specific Models are themselves derived from domain specific Models and /or generic specific Models defined in this recommendation. This is why each Archive Project should create specialised versions of the proposed generic Models by adding, eliminating or modifying the attributes of these Models (cf figure 1). This process will be shown through the examples in the Annex.

Each Descriptor Model consists of a certain number of attributes and each of these attributes has a meaning, a syntax and a defined occurrence.

Descriptor’s attributes may be divided into four main groups:

- attributes for identifying the type of object described (DESCRIPTOR_ID), for referring to it in an understandable way (LABEL) and for defining the number of times it occurs (OBJECT_OCCURRENCE)

- attributes for defining the relations between the object described and the other Model objects:
  - The object is tied to the higher level node in the MOT (PARENT_COLLECTION).
  - The transversal links specify the type of relation between a Complementary Data Object and the objects to which it is tied (RELATION_CO_CDO) and the identifier of the objects in question (links to RELATED_DESCRIPTOR_ID),

- The description of the object content:
  - CONTENT: textual and qualitative description.
  - FORMAT, SIZE: qualitative and quantitative information on the objects.
  - Applicable metadata standards and models: DESCRIPTION_STANDARD, METADATA_MODEL, METADATA_STANDARD.

- APPLICABLE_SLIP gives the name of the Slip Model to be delivered; it will make it possible to specify in detail the content of the elementary package correspondent to the described object. Several Descriptors may reference the same Slip Model.

The attributes are placed in the following order: first, attributes which are common to all the Descriptors (in the logical order previously described), then attributes which are specific to each Descriptor (by alphabetical order). To make it easier to read, the latter are displayed in grey boxes.

Annex A.1 gives an example of the way generic Models are applied to make specialised Models then to their instances in the context of Space Physics.

3.4.2.1 Generic Descriptor Models

We have chosen here to define object Descriptor Models a priori for each of the four information categories defined in the section ‘Typology of information categories’:

Name of the information objects class	**Name of the corresponding Descriptor model**
Data Object	DO_DESCRIPTOR
Complementary Data Object	COMPLEMENTARY_DO_DESCRIPTOR
Collection	COLLECTION_DESCRIPTOR
Complementary Collection	COMPLEMENTARY_COLLECTION_ DESCRIPTOR

We are not referring at this stage to a particular implementation choice.

DO_DESCRIPTOR **Generic Model of a Data Object Descriptor**		
Attribute	**Meaning**	**Attribute occurrence**
DESCRIPTOR_ID	Identifier of Data Object Descriptor	1..1
OBJECT_OCCURRENCE	Number of DATA Objects described by this DESCRIPTOR_ID	1..N
LABEL	Extensive name of the Data Object	1..1
PARENT_COLLECTION	Identifier of Collection Descriptor to which this object belongs	1..1
CONTENT	Explanatory text specifying the content and main characteristics of the object (e.g. magnetic field, etc.)	0..1
SIZE	Estimated volume of an object and unit	0..1
APPLICABLE_SLIP	Name of slip model to be applied	1..1
DESCRIPTION_STANDARD	Enumerated list (EAST, DED, etc.). Used to specify the data description standard to be applied.	0..N
METADATA_MODEL	Enumerated list (XML Schema-name, etc). Used to specify the 'catalogue' data characterising the object: for instance in the form of an XML document specified by an XML Schema	0..N
OBJECT_FORMAT	Format of objects in the collection (PNG, PDF, CDF, Flat Binary, Flat ASCII, …)	0..1

- OBJECT_OCCURRENCE: A Descriptor instance may in fact describe a unique object (this is often the case for 'Collection' objects, OBJECT_OCCURRENCE = 1) or a representative of a category containing a certain number of objects (for example, a Data Object within a collection, OBJECT_OCCURRENCE > 1). The role of the OBJECT_OCCURRENCE attribute is to specify this aspect. The occurrence may not be known ahead of time (value 1..n).
- DESCRIPTION_STANDARD: description standard applying to the type (syntactical or semantic description).
- METADATA_MODEL: used to specify metadata characterising each individual Data Object (for instance XML file containing the start and end times, the geographic coordinates of the image, the orbit number for each Data Object, etc.).

COMPLEMENTARY_DO_DESCRIPTOR **Generic model of a Complementary Data Object Descriptor**		
Attribute	**Meaning**	**Attribute occurrence**
DESCRIPTOR_ID	Identifier of Complementary Data Object Descriptor	1..1
OBJECT_OCCURRENCE	Number of Complementary Data Objects having the DESCRIPTOR_ID type	1..N
LABEL	Extensive name of the Complementary Data Object	1..1
PARENT_COLLECTION	Identifier of the Collection Descriptor to which this object belongs	1..1
CONTENT	Explanatory text specifying the content and main characteristics of the Complementary Data Object	0..1
SIZE	Estimated volume of an object and unit	0..1
APPLICABLE_SLIP	Name of Slip Model to be applied	1..1
DESCRIPTION_STANDARD	Enumerated list (EAST, EAD, etc.). Used to specify the data description standard to be applied.	0..N
METADATA_MODEL	Enumerated list (XML Schema, etc.). Used to specify the 'catalogue' data characterising the object in the form of an XML Schema for instance.	0..N
RELATED_DESCRIPTOR_ID	Descriptor identifier for the Data Object Collection or Data Object to which the CDO refers.	1..N
RELATION_DO_CDO	Type of relation between the CDO in question and DO collection (or the type of DO).	0..1

- RELATION_DO_CDO: the relation between the Complementary Data Object and the RELATED_DESCRIPTOR_ID is used to characterise the kind of relation between the Complementary Data Object and the descriptor identified by the RELATED_DESCRIPTOR_ID. For instance if the Complementary Data Object is an EAST file, the type of the relation might be a 'syntactical description'.

COLLECTION_DESCRIPTOR **Generic model of a Data Object Descriptor**		
Attribute	**Meaning**	**Attribute occurrence**
DESCRIPTOR_ID	Identifier of the Collection Descriptor	1..1
OBJECT_OCCURRENCE	Number of Collections described by the Descriptor	1..N
LABEL	Extensive name of the Collection	1..1
PARENT_COLLECTION	Descriptor identifier of the parent Collection (possibly ROOT)	1..1
CONTENT	Explanatory text specifying content and main characteristics of the collection	0..1
SIZE	Total estimated volume and unit	0..1
APPLICABLE_SLIP	Name of Slip Model to be applied, or NONE	1..1
METADATA_STANDARD	Enumerated list (DIF, SPASE, DUBLIN_CORE, EAD, ISO_19115, etc.). Used to specify the metadata standard to be applied	1..1

- The CONTENT and SIZE attributes are optional attributes which may not be relevant in the context of a given project and which are likely to be modified or eliminated.
- SIZE: for a collection of collections, this attribute may be omitted (to the extent that the size of the terminal collections is known).
- APPLICABLE_SLIP: for a collection of collections, there may not be a corresponding delivery during a transfer stage and hence no applicable Slip: the attribute value is then NONE.

COMPLEMENTARY_COLLECTION_DESCRIPTOR **Generic model of a Complementary Data Object Collection Descriptor**		
Attribute	**Meaning**	**Attribute occurrence**
DESCRIPTOR_ID	Descriptor identifier for the Complementary Data Object Collection	1..1
OBJECT_OCCURRENCE	Number of Complementary Collections described by this Descriptor	1..N
LABEL	Extensive name of the Complementary Data Object Collection	1..1
PARENT_COLLECTION	Descriptor identifier of the higher level Collection (possibly ROOT)	1..1
CONTENT	Explanatory text specifying the content and main characteristics of the Collection	0..1
SIZE	Estimated volume of an object and unit	0..1
APPLICABLE_SLIP	Name of Slip Model to be applied, or NONE	1..1
METADATA_STANDARD	Enumerated list (DUBLIN_CORE, EAD, etc.). Used to specify the metadata standard to be applied	0..1
RELATED_DESCRIPTOR_ID	Descriptor identifier of the Data Objects	1..N

	collection or the Data Object Descriptor to which the CDO collection refers	
RELATION_DO_CDO	Type of relation between the CDO collection and the DO collection (or the type of DO)	0..1

- The CONTENT and SIZE attributes are optional attributes which may not be relevant in the context of the project and which are likely to be modified or eliminated.
- APPLICABLE_SLIP: for this Descriptor, there may be no digital object delivered during the transfer phase. The attribute value will then be NONE.
- RELATION_DO_CDO: the relation between the Complementary Data Object collection and the RELATED_DESCRIPTOR_ID is used to characterise the kind of relation between the Complementary Data Object Collection and the descriptor identified by the RELATED_DESCRIPTOR_ID. For instance, if the Complementary Data Object is an EAST file, the type of relation might be a 'syntactical description'.
- SIZE: for a collection of collections, this attribute may not be specified (to the extent that the size of the terminal collections is known).

3.4.2.2 Specialisation of generic Descriptor Models

The previous generic Models contain a minimum set of attributes. They may be specialised to achieve the applicable level desired by the Producer-Archive project:

- Specialisation of a domain generic Model from a generic Model towards (see section 4 of PAIMAS),
- Domain generic specialisation for creating a Project Archive Model (other intermediary levels may be considered if necessary).

To do this, the standard generic Model may undergo the following modifications:

- Addition of an attribute.
- Elimination of an attribute.
- Modification of existing attributes (content, occurrence, name of the attribute to adapt to the project terminology, syntax, etc.).

3.4.2.3 Descriptor Model instances

The creation of Descriptor instances is definitely the most crucial task during the Formal Definition Phase since it is at this stage that we clearly define the objects to be transferred. This creation allows for the MOT to be built.

This task will consist for example of:

- identifying all of the project data Collections.
- organising these Collections in a hierarchical tree structure.
- defining a Descriptor instance for each of the Collections.
- defining a Data Object Descriptor for the objects in each Collection.
- etc.

Figure 2 shows an example of a Descriptor instance.

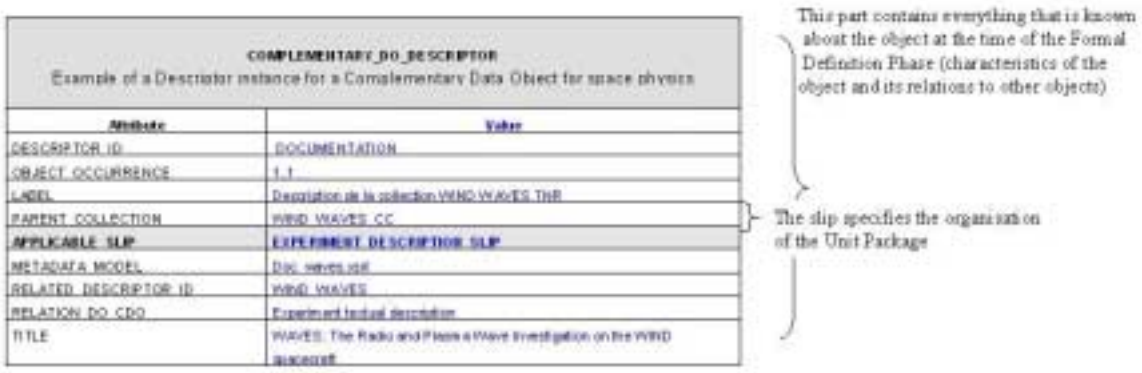

Figure 2: Structure of a Descriptor instance

3.4.3 DELIVERY SLIPS

Each object will be transferred in an elementary package to be delivered with a Slip when it is transferred. The Slip is the 'packaging' of the delivered object. This packaging is described by a Model referenced by the APPLICABLE_SLIP attribute in the MOT object Descriptors. There may thus be several types of Slips depending on the type of objects delivered. This Recommendation proposes a generic Model which can be adapted.

These Slip Models will be defined during the Formal Definition Phase and instantiated during the Transfer Phase.

Annex A.2 gives an example of the application of the generic Slip Model to specialised Models and then to their instances in a context of Space Physics.

3.4.3.1 Generic Slip Model

Initially, we have a single Slip Model for each of the four categories identified (see the section 3.2 on 'Typology of information categories'):

Name of information objects class	Name of corresponding Slip Model
Data Object	TRANSFERRED_OBJECT_SLIP
Complementary Data Object	
Collection	
Complementary Collection	

The Slip Model includes a certain number of attributes and each of these attributes has a meaning, syntax and a defined occurrence.

We are deliberately ignoring here the technology used to implement them: DEDSL- XML/DTD Syntax (see Reference [4]), XML Schema (see reference [6]), etc.

As was the case for the Descriptors, the generic Model provided in this standard may be adapted to make a specialised Model (see section 3.4.3.2).

The previously identified Slip Model uses the TRANSFERRED_METADATA_FILES and TRANSFERRED_DO_FILES attributes. These are composite attributes of the TRANSFERRED_FILES_DESCRIPTION type defined below.

TRANSFERRED_FILES_DESCRIPTION **Composite attribute for file transfer**		
Attribute	**Meaning**	**Attribute occurrence**
TRANSFERRED_FILE	Name of the file to be transferred	1..N
DO_EXTRACTION_SOFTWARE	Name of software used to reconstitute the Data Object from different files	0..1
DO_EXTERNAL_REFERENCE	External reference of files constituting this Data Object if these files are not included during transfer but deposited in a shared space.	0..N

TRANSFERRED_OBJECT_SLIP **Generic Slip Model**		
Attribute	**Meaning**	**Attribute occurrence**
SLIP_MODEL_ID	*Slip type identifier. This attribute takes a constant value in the model..*	*1..1*
OBJECT_ID	Object identifier (Data Object, Complementary Data Object, Collections)	1..1
DESCRIPTOR_ID	Identifier of the Object Descriptor to which the object refers. This attribute is used to locate this object in the MOT.	1..1
TRANSFERRED_DO_FILES	Data file(s) which make up this object. This attribute is of the TRANSFERRED_FILES_DESCRIPTION type.	0..N
LAST_OBJECT	Indicator specifying that it is the last object (among the Data Objects, Complementary Data Objects or Collections corresponding to the same MOT node).	0..1
TRANSFERRED_METADATA_FILES	Metadata file(s) accompanying this object. This attribute is of the TRANSFERRED_FILES_DESCRIPTION type.	0..N

The SLIP_MODEL_ID **constant** (in italics) is used to characterise the Slip Model. It takes its value from a set of possible names made up of all the Slip Models for the Archive-Project in question. This constant is known during the Formal Definition Phase and is used to identify the different SLIP models and the attributes corresponding to them.

LAST_OBJECT: This attribute is useful if the number of objects in a collection is not known ahead of

time and is not defined in the MOT. When it is used, a sequence management rule has to be specified for transferring the objects (case in which, during transfer, the last object would not arrive last).

3.4.3.2 Specialisation of generic Slip Model

This generic Slip Model may be used as defined but it can also be used as a basis for building one or more specific Slip Models adapted to the Domain for any particular Producer-Archive Project. This is why each project should make a specialised Model from the proposed Model.

This Model may be modified by adding or eliminating attributes, modifying existing attribute characteristics (change of name occurrence, etc.). It is also possible to define new object categories if the Archive Project believes it to be useful.

A few specialisation examples are given in annex A.2.

3.4.3.3 Slip Model instances

An instantiated Slip is delivered with each object.

Each Slip instance refers to one and only one object Descriptor identifier (defined by a DESCRIPTOR_ID attribute), which is used to tie the object to an MOT node.

A digital object may be transferred in different ways: a Data Object corresponding to a large file may be broken down into several smaller files depending on the network bandwidth. On the other hand, it might be aggregated with other objects within a 'tar' type file. The Slip then specifies the way the object should be reconstituted. The Slip's role is to provide information for reconstituting the Data Object from the different data containers used for the transfer.

3.4.4 MODEL DEVELOPMENT

For a given Producer-Archive project, the Model of Objects to be Transferred is developed during the Formal Definition Phase which consists of 3 work stages:

- Development of the dictionary for project Descriptor Models which may have been specialised for the project in question,
- Development of the Slip Model dictionary,
- Development of Descriptor instances describing what is to be delivered during the Archive Project.

This is a fairly simplified view because when developing instances, it is always possible, depending on the needs, to modify, complete or specialise Model dictionaries.

The development of the MOT assumes that the Producer and the Archive have done their iterative groundwork.

The formal representation built from the previously defined Models and Instances is described in detail in section 4.

3.4.5 MODEL REPRESENTATION

The Model of Objects to be Transferred should enable the Producer and the Archive to have a clear view and understanding of the objects to be transferred. This Model should also enable them to use automatic processing for monitoring the project and validating deliveries, whereas the implementation of the Model in computer language (XML for example) does not allow for visual displays which are easily understandable to the human eye.

The MOT as previously defined is a tree structure to which all of the Descriptor instances are linked (there are no orphan elements). This tree structure consists of descending hierarchical relations between Collections of Collections, Collections of objects and terminal objects, as well as transversal relations between Complementary Objects and Data Objects (or respective Collections). Finally, each node of the tree structure, whether this involves Collections or objects is defined by information contained in its corresponding Descriptor.
The MOT may thus be represented with the following levels of detail:

- Hierarchical tree structure representation.
- Hierarchical tree structure representation and transversal relations.
- Complete description of each Model node.

Moreover, it is interesting that the graphic representation of the MOT is used to distinguish between the different object categories (Collection, Data Object, etc.) and the number of times each object occurs, as is the case in the example shown in figure 3.

Figure 2 below represents in the form of a tree structure the project described in annex A:

- The WIND_WAVES node is a Collection of Data Object collections,
- The WIND_WAVES_TNR_L2 node is a Collection of Data Objects,
- The WIND_WAVES_CC node is a Collection of Complementary Data Objects,
- The WIND_WAVES_TNR_L2, EAST_DESCRIPTION and DOCUMENTATION nodes are leaves of the tree corresponding to Data Objects or Complementary Data Objects.
- The dotted lines show transversal relations (defined by the RELATION_DO_CDO attribute in the corresponding Descriptors). In the Descriptors, the choice was made to describe the relation in the CDOs and not in the Collections or DOs that the CDOs describe.

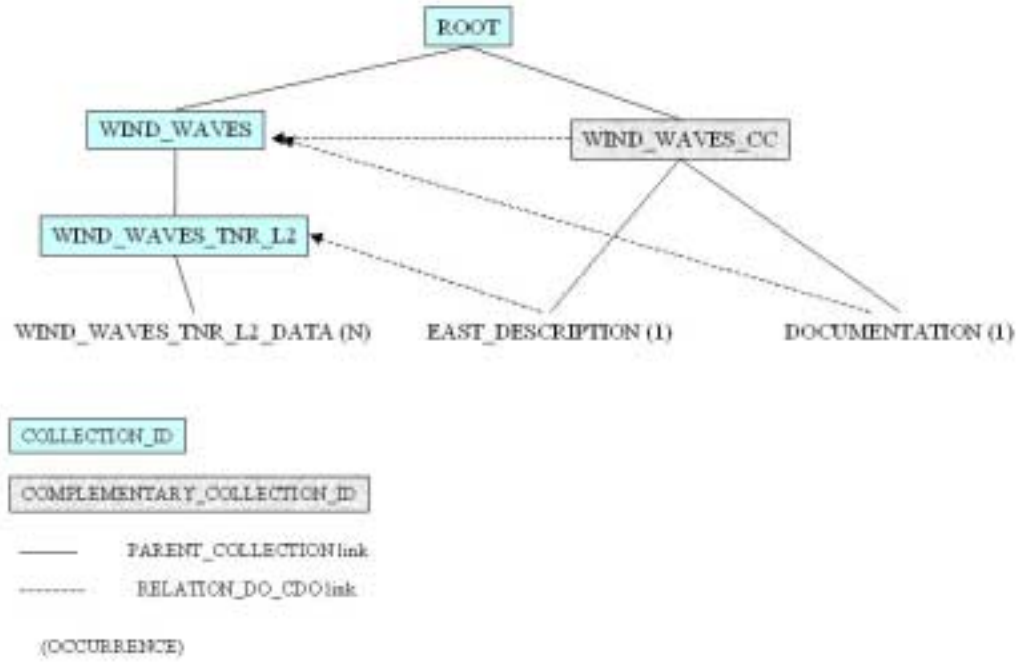

Figure 3: Hierarchical tree structure representation of transversal relations

Figure 4 is an example of a form used to describe a node. One might imagine a set of formulas written in HTML describing each of the nodes in the MOT. Instead of the attribute, one might also imagine a more 'user-friendly' description (short description of the attribute).

COMPLEMENTARY_DO_DESCRIPTOR_SPACE_PHYSICS

DESCRIPTOR_ID	DOCUMENTATION
OBJECT_OCCURRENCE	1..1
LABEL	Description de la collection WIND WAVES TNR
PARENT_COLLECTION	WIND_WAVES_CC
APPLICABLE_SLIP	EXPERIMENT_DESCRIPTION_SLIP
METADATA_MODEL	Doc_waves.xsd
RELATED_DESCRIPTOR_ID	WIND_WAVES
RELATION_DO_CDO	Experiment textual description
TITLE	WAVES: The Radio and Plasma Wave Investigation on the WIND spacecraft

Figure 4: Description of DOCUMENTATION node

3.5 SIPs

The SIP is a coherent group of elementary packages in a package to be transferred.

A SIP may contain several objects of the same type or of different types. The rules for grouping objects within an SIP must be defined during the Formal Definition Phase. These rules are then used to define the different SIP categories for the project. These categories are built from the following simple model:

SIP_DESCRIPTOR Generic SIP Model		
Attribute	**Meaning**	**Attribute occurrence**
SIP_MODEL_ID	SIP type identifier	1..1
DESCRIPTOR_ID	Object Descriptor identifier contained in the SIP.	1..N

From this Model, the different SIP categories may be defined during the following phase.

SIP_DESCRIPTOR SIP descriptor instance		
Attribute	**value**	**occurrence**
SIP_MODEL_ID	Sip_type1	
DESCRIPTOR_ID	D1	1..10
DESCRIPTOR_ID	D2	0..1

Figure 5: Structure of a SIP Descriptor instance

In the above example, it is specified that the SIP of the SIP_type1 category shall consist of:

- 1 to 10 elementary object packages described in the MOT by the D1 Descriptor,
- 0 or 1 elementary object packages described in the MOT by the D2 Descriptor.

Annex A.3 gives an example of the building of SIPs in the context of Space Physics.

Apart from specific procedures to be used for renewing transfer of an object which has already been transferred (for instance following observation of an anomaly in the initial object), we have assumed that an object identified in the MOT is only transferred once. The Producer and Archive thus have to ensure that the SIP definitions are coherent from this point of view.

3.6 SEQUENCING CONSTRAINTS

If there is no sequencing constraint, the SIPs may be transferred independently of each other in any order.

It may be necessary to specify in the Submission Agreement that a given digital object must be transmitted before or after another. For instance, in the case of a Collection of files of scientific data all having the same syntactical structure described with EAST language, the archive might want to systematically verify the conformance of each file in the collection in relation to the EAST description of the files in this Collection. In this case, the EAST description should be sent before the data files.
The sequencing systems may be complex but it appears clear to us that the sequencing constraints for a Producer-Archive project should remain simple.
This is why we shall only define here a limited number of possibilities for expressing sequencing constraints, given that the specialisation mechanism allows for more complex cases to be dealt with.

The existing constraints between two objects A and B may be totally independent of the existing constraints between two other objects C and D. This has led us to define the concept of **constraints group** as being the set of objects related to each other by a set of dependent sequencing constraints.
The example referred to above concerning an EAST syntactical descriptor can in practice apply to several Collections of distinct Data Objects, each having its own EAST description and belonging to the same Producer-Archive Project. We may thus define several independent constraint groups:

Constraint group Groupe_1:
The EAST description of the files in the A Collection must be delivered before the files in this Collection.
Constraint group Groupe_2:
The EAST description of the files in the B Collection must be delivered before the files in this Collection.

To express these constraints, we shall use two distinct attributes in the SIP Model, to define the sequencing constraints at the Descriptor level:

Attribute name	Meaning	Value syntax	Occurrence
TIME_CONSTRAINT_GROUP	define the groups	identifier	0..1
SERIAL_NUMBER_IN_CONSTRAINT	define the constraints within the same group	integer	0..1

The use of these two attributes is optional.

If TIME_CONSTRAINT_GROUP is defined in a SIP Descriptor, then
SERIAL_NUMBER_IN_CONSTRAINT is compulsory for this SIP Descriptor.
If there is no time constraint, they are omitted.
If there are time constraints, but a unique group, TIME_CONSTRAINT_GROUP may be omitted.
If two SIPs SIP1 and SIP2 belong to the same group with:
SERIAL_NUMBER_IN_CONSTRAINT = 1 for SIP1 and
SERIAL_NUMBER_IN_CONSTRAINT = 2 for SIP2,
this means that the digital object(s) corresponding to SIP1 must be transferred before the digital object(s) corresponding to SIP2.
If 3 SIPs SIP1, SIP2 and SIP3 belong to the same group with:
SERIAL_NUMBER_IN_CONSTRAINT = 1 for SIP1,
SERIAL_NUMBER_IN_CONSTRAINT = 2 for SIP2 and
SERIAL_NUMBER_IN_CONSTRAINT = 2 for SIP3,
this means that the digital object(s) corresponding to SIP1 must be transferred before the digital object(s) corresponding to SIPs, SIP2 and SIP3 and that there is no constraint between the digital object(s) of SIP2 and SIP3.

Example:
This example defines two groups of sequencing constraints which are independent of each other.
For each group, it specifies:
- that the EAST Descriptor must be transferred first,
- that the objects corresponding to the Collection descriptions (metadata describing the collection) and the Data Objects for these collections can then be transferred in any order for each collection.

-

SIP Descriptor instance (Collection)	
SIP_MODEL_ID	SIP1
DESCRIPTOR_ID	Collection_1
TIME_CONSTRAINT_GROUP	Group_1
SERIAL_NUMBER_IN_CONSTRAINT	2

SIP Descriptor instance (Collection)	
SIP_MODEL_ID	SIP2
DESCRIPTOR_ID	Collection_2

SIP Descriptor instance (Data Objects, Collection 1)	
Attribute	**Value**
SIP_MODEL	SIP3
DESCRIPTOR_ID	do_collection_1
TIME_CONSTRAINT_GROUP	Group_1
SERIAL_NUMBER_IN_CONSTRAINT	2

SIP Descriptor instance (Data Objects, Collection 2)	
Attribute	**Value**
SIP_MODEL	SIP4
DESCRIPTOR_ID	do_collection_2
TIME_CONSTRAINT_GROUP	Group_2
SERIAL_NUMBER_IN_CONSTRAINT	2

SIP Descriptor instance (complementary_do_descriptor)	
Attribute	**Value**
SIP_MODEL	SIP5
DESCRIPTOR_ID	EAST_descriptor_collection_1
TIME_CONSTRAINT_GROUP	Groupe_1
SERIAL_NUMBER_IN_CONSTRAINT	1

SIP Descriptor instance (complementary_do_descriptor)	
Attribute	**Value**
SIP_MODEL	SIP6
DESCRIPTOR_ID	EAST_descriptor_collection_2
TIME_CONSTRAINT_GROUP	Groupe_2
SERIAL_NUMBER_IN_CONSTRAINT	1

4 FORMAL PHASE DETAILED SPECIFICATION

In this section, we propose a particular implementation of the concepts described previously.

4.1 CONSTRUCTION OF MODEL DICTIONARIES

Construction of Descriptor Models and generic Slip Models in the form of XML schemas.

Method for specialising them and examples.

Example to be proposed in annex based on examples in annex A.

4.2 MODEL OF OBJECTS TO BE TRANSFERRED

Construction of Descriptor Instances in the form of XML documents based on specialised schemas defined in the previous stage.

This is the final stage for building the Model of objects to be transferred and of the representation of this Model according to needs.

4.3 FORMAL AGREEMENT

5. TRANSFER PHASE SPECIFICATION

Define how the SIPs may be transferred using an XFDU packaging for reconstituting objects.

Assigning to each XFDU package an ordinal number so as to be able to control the sequencing of packages and if necessary the compliance with sequencing constraints.

Definition of transmission sessions.

Monitoring of transfer over time: definition of control panel.

Identification of validation features offered by this packaging.

6. MANAGING MODIFICATIONS

This means all modifications which occur once the Submission Agreement has been approved.

There are two categories of modifications:

- Category 1: The MOT modifications: modification of Model dictionaries and Descriptor instances. These modifications should be done in compliance with the rules defined in the PAIMAS. The rules for managing these modifications may moreover have been specified in the Submission Agreement.

These modifications concern for instance:

  - addition or elimination of a Collection of Data Objects (or Complementary Data Objects).
  - addition or elimination of types of Data Objects (or Complementary Data Objects).
  - modification of Collection or object instances which have already been defined.
  - addition, elimination or modification of elements or attributes of dictionary elements and the consequences on the MOT negotiated for the Archive Project.

- Category 2: modifications related to an error or an anomaly in the production or during the transfer of object: if an SIP contains an erroneous object, it should be possible, by setting precise management rules, to deliver this SIP again.

7. TOOLS

The implementation suggested in this document, based on the XML Schemas can be used to build the MOT. However, the practical use of this implementation by an Archive Project involves setting up a certain number of tools for this implementation (this list is not complete):

The Formal Definition Phase:

- Tools for creating specialised dictionaries from the generic Models proposed (a tool such as XML Spy can be used to specialise an XML Schema),
- Tools for generating Descriptor instances, for instance by means of forms made from the schemas,
- Tool producing a graphic and written representation of the MOT which can be understood by the Producer and the Archive

The Transfer Phase:

- Tools for generating Slip instances,
  - Software generation when there are many instances,
  - Generation by forms for the other cases,

- Tool for validating the SIPs received: in compliance with the MOT,
- Tool for monitoring project progress for the Producer and for the Archive, as well as for expected objects and imposed sequencing constraints,
- etc.

The Producer and the Archive should identify the needs for tools at the outset and should evaluate development efforts if the tools have to be developed or adapted.

ANNEX A : EXAMPLES

(This annex **is not** part of the Recommendation)

In the following examples, the cells in the tables corresponding to the modified attributes appear with gray background. For each category of information, the specialized Model followed by one or more instances is given.
The Models appear in blue, their instances in red.

A.1 EXAMPLES OF MODELS AND INSTANCES OF DESCRIPTORS

Here are given some examples of Models and instances of specialized Descriptors in the context of Space Physics.

A.1.1 EXAMPLE OF COLLECTION DESCRIPTOR

In this example the COLLECTION_DESCRIPTOR specialization consists of:

- Adding 2 mandatory attributes "MISSION" and "EXPERIMENT",
- The enumerated of possible metadata standards,
- The modification of the occurrence for the attribute CONTENT made mandatory.

COLLECTION_DESCRIPTOR **Model of a Data Object Collection Descriptor for Space Physics**		
Attribute	Meaning	Attribute occurrence
DESCRIPTOR_ID	Collection identifier	1..1
OBJECT_OCCURRENCE	Number of Collections described by DESCRIPTOR_ID	1..N
LABEL	Extensive name of the Collection	1..1
PARENT_COLLECTION	Identifier of the higher level Collection (possibly root)	1..1
CONTENT	Explanatory text specifying the content and main characteristics of the Collection	1..1
SIZE	Estimated volume for the whole Collection	0..1
APPLICABLE_SLIP	Name of Slip Model to be applied, or NONE	1..1
MISSION	Name of the space mission	1..1
EXPERIMENT	Name of the experiment	1..1
METADATA_STANDARD	Enumerated list (DIF, SPASE, CAA or NONE) SPASE : Space Physics Archive Search and Extract CAA : Cluster Active Archive	1..1

The collection below is a terminal collection containing a set of Data Objects. All the attributes are defined.

Example of instance of COLLECTION_DESCRIPTOR for the Thermal Noise Receiver level 2 data of the WAVES experiment in WIND mission	
Attribute	**Value**
DESCRIPTOR_ID	WIND_WAVES_TNR_L2

OBJECT_OCCURRENCE	1
LABEL	Collection of the data of level 2 (gauged) of part TNR of the Waves experiment of mission WIND
PARENT_COLLECTION	WIND_WAVES
CONTENT	These data contain the high resolution gauged spectra produced, for each analyzed frequency band, by the Thermal Noise Receiver (TNR) part of the WAVES experiment. The dat start is November 1994
SIZE	200..400 gigabytes
APPLICABLE_SLIP	SPACE_PHYSICS_COLLECTION_SLIP
MISSION	WIND
EXPERIMENT	WAVES
METADATA_STANDARD	DIF

The collection below is a Collection of Collections. This is why its size is not defined (its size is that of the child Collection WIND_WAVES_TNR_L2 defined above). In the same way there is no associated METADATA_STANDARD or APPLICABLE_SLIP (no object will be delivered during the Transfer Phase for this Collection).

Example of instance of COLLECTION_DESCRIPTOR for the Thermal Noise Receiver level 2 data of the WAVES experiment, WIND mission	
Attribute	**Value**
DESCRIPTOR_ID	WIND_WAVES
OBJECT_OCCURRENCE	1
LABEL	Collection of WIND WAVES data Collections
PARENT_COLLECTION	ROOT
CONTENT	Contains the WIND_WAVES_TNR_L2 Collection
APPLICABLE_SLIP	NONE
MISSION	WIND
EXPERIMENT	WAVES
METADATA_STANDARD	NONE

A.1.2 EXAMPLE OF DATA_OBJECT DESCRIPTOR

The DO_DESCRIPTOR specialization consists of:

- The enumerated list of DO_DESCRIPTION_MODEL,
- The number of possible standards for data description,
- The modification of the occurrences of the attributes CONTENT et SIZE that are mandatory.

DO_DESCRIPTOR **Specific model of a Data Object Descriptor for Space Physics**		
Attribute	Meaning	Attribute occurrence
DESCRIPTOR_ID	Identifier of the Data Object	1..1
OBJECT_OCCURRENCE	Number of Data Objects described by this Descriptor	1..N

LABEL	Extensive name of representative of Data Object	1..1
PARENT_COLLECTION	Identifier of Collection Descriptor to which this object belongs	1..1
CONTENT	Explanatory text specifying the content and main characteristics of the object (e.g. magnetic field, etc.)	0..1
SIZE	Estimated volume of an object and unit	1..1
APPLICABLE_SLIP	Name of Slip Model to be applied	1..1
DESCRIPTION_STANDARD	Enumerated list (EAST, DEDSL, etc.). Used to specify the data description standard to be applied.	1..2
METADATA_MODEL	Enumerated list (XMLSchema-name, etc). Used to specify the 'catalogue' data characterising the object: for instance in the form of an XML document specified by an XML Schema	1..1
OBJECT_FORMAT	Format of objects in the Collection (PNG, PDF, CDF, Flat Binary, Flat ASCII, …)	1..1

Example of Descriptor instance for a level 2 Data Object of WAVES experiment on board WIND satellite (Space Physics)	
Attribute	**Value**
DESCRIPTOR_ID	WIND_WAVES_TNR_L2_DATA
OBJECT_OCCURRENCE	N
LABEL	Day file of Thermal Noise receiver data of WAVES experiment, WIND mission
PARENT_COLLECTION	WIND_WAVES_TNR_L2
CONTENT	Each data file containe: - Date and time - data of the Automatic gain control - high resolution spectra for each frequency band observed. The unit used for the spectra is the microV2/Hz Each file covers one day of data from 0 to 24 hours.
SIZE	10..40 megabytes
APPLICABLE_SLIP	SPACE_PHYSICS _SLIP
DESCRIPTION_STANDARD	EAST, DEDSL
METADATA_MODEL	CDPP_DO.xsd
OBJECT_FORMAT	Flat_binary

A.1.3 EXAMPLE OF COMPLEMENTARY COLLECTION DESCRIPTOR

- No model specialization here, the generic Model is directly used.

COMPLEMENTARY_COLLECTION_DESCRIPTOR **The Model of Complementary Data Object Collection Descriptor for Space Physics (the same as the generic one)**		
Attribute	Meaning	Attribute occurrence

DESCRIPTOR_ID	Descriptor identifier for the Complementary Data Object collection	1..1
OBJECT_OCCURRENCE	Number of Complementary Collections described by this Descriptor	1..N
LABEL	Extensive name of the Complementary Data Object collection	1..1
PARENT_COLLECTION	Descriptor identifier of the higher level Collection (possibly ROOT)	1..1
CONTENT	Explanatory text specifying the content and main characteristics of the Collection	0..1
SIZE	Estimated volume of an object and unit	0..1
APPLICABLE_SLIP	Name of Slip Model to be applied, or NONE	1..1
METADATA_STANDARD	Enumerated list (DUBLIN_CORE, EAD, etc.). Used to specify the metadata standard to be applied	0..1
RELATED_DESCRIPTOR_ID	Descriptor identifier of the Data Objects Collection or the Data Object Descriptor to which the CDO collection refers	1..N
RELATION_DO_CDO	Type of relation between the CDO Collection and the DO collection (or the type of DO)	0..1

COMPLEMENTARY_COLLECTION_DESCRIPTOR
Example of instance for a Complementary Data Object Collection Descriptor for Space Physics

Attribute	Value
DESCRIPTOR_ID	WIND_WAVES_CC
OBJECT_OCCURRENCE	1
LABEL	WIND WAVES documents and descriptive
PARENT_COLLECTION	ROOT
CONTENT	Documents, publications, syntactic and semantic descriptions relating to WAVES experiment and the various Collections of data resulting from this experiment
APPLICABLE_SLIP	NONE
RELATED_DESCRIPTOR_ID	WIND_WAVES
RELATION_DO_CDO	Associated documentation

A.1.4 EXAMPLE OF COMPLEMENTARY DATA OBJECT DESCRIPTOR

The COMPLEMENTARY_DO_DESCRIPTOR specialization consists of:

- The addition of TITLE attribute,
- Suppression of CONTENT and LABEL attributes.

COMPLEMENTARY_DO_DESCRIPTOR
Model of Complementary Data Object Descriptor fos Space Physics

Attribute	Meaning	Attribute occurrence
DESCRIPTOR_ID	Information category: enumerated list (EAST_DESCRIPTION, DED, DOCUMENTATION, REFERENCE)	1..1

OBJECT_OCCURRENCE	Number of Complementary Data Objects having the DESCRIPTOR_ID type	1..N
PARENT_COLLECTION	Identifier of Collection Descriptor to which this object belongs	1..1
SIZE	Estimated volume of an object and unit	0..1
APPLICABLE_SLIP	Name of Slip Model to be applied	1..1
METADATA_MODEL	Enumerated list (XML Schema, etc.). Used to specify the 'catalogue' data characterising the object in the form of an XML Schema for instance.	0..1
RELATED_DESCRIPTOR_ID	Descriptor identifier for the Data Object collection or Data Object to which the CDO refers.	1..N
RELATION_DO_CDO	Type of relation between the CDO in question and DO collection (or the type of DO).	0..1
TITLE	Title if it's a document	0..1

COMPLEMENTARY_DO_DESCRIPTOR **Example of instance for a Complementary Data Object Descriptor for Space Physics**	
Attribute	**Value**
DESCRIPTOR_ID	EAST_DESCRIPTION
OBJECT_OCCURRENCE	1
PARENT_COLLECTION	WIND_WAVES_CC
APPLICABLE_SLIP	EAST_SLIP
RELATED_DESCRIPTOR_ID	WIND_WAVES_TNR_L2_DATA
RELATION_DO_CDO	Syntactic file description

The Doc_waves.xsd file associated with the DOCUMENTATION type makes it possible to specify a set of information associated with this type of description like the author, the reference towards an abstract, the version, the number of pages.

COMPLEMENTARY_DO_DESCRIPTOR **Example of instance for a Complementary Data Object Descriptor for Space Physics**	
Attribute	**Value**
DESCRIPTOR_ID	DOCUMENTATION
OBJECT_OCCURRENCE	1
PARENT_COLLECTION	WIND_WAVES_CC
APPLICABLE_SLIP	EXPERIMENT_DESCRIPTION_SLIP
METADATA_MODEL	Doc_waves.xsd
RELATED_DESCRIPTOR_ID	WIND_WAVES
RELATION_DO_CDO	Experiment textual description
TITLE	WAVES: The Radio and Plasma Wave Investigation on the WIND spacecraft

Représentations:

The following figure 2 shows a view of Descriptors defined and instantiated during the Formal Definition Phase: hierarchical and descriptive links, categories of objects (blue, gray, framed or not), Descriptors and the files which will be delivered during the Transfer Phase.

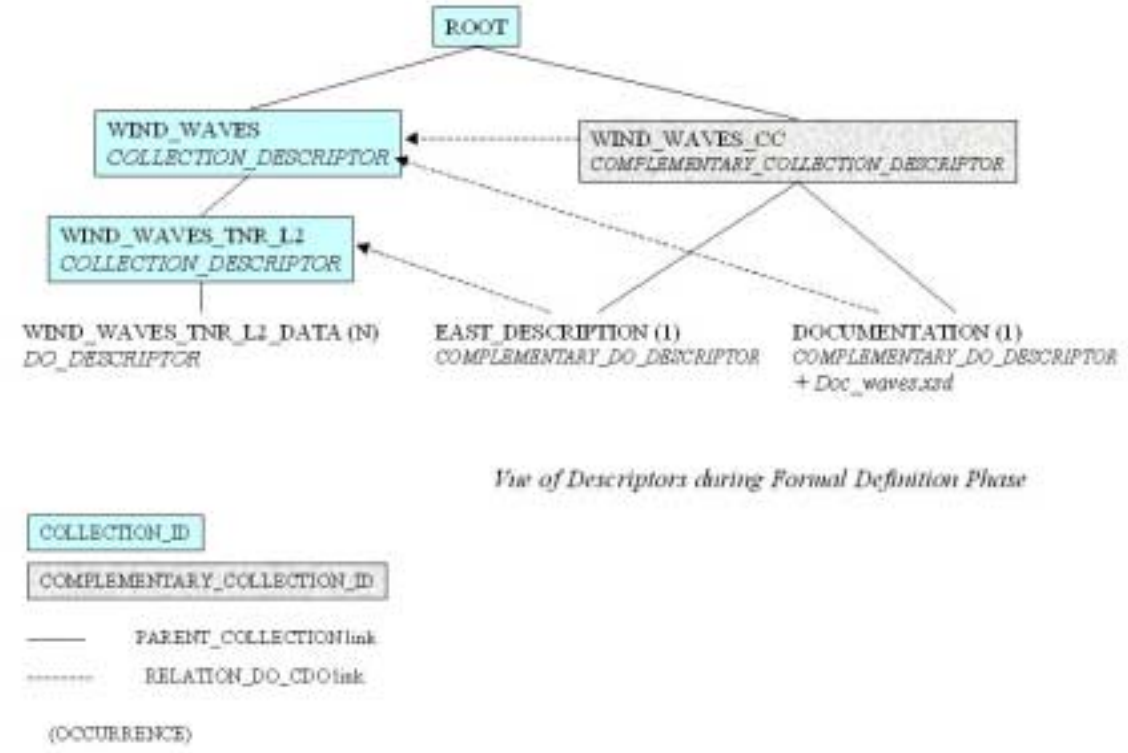

Figure 5: view of the graph of the Descriptor instances

A.2 EXAMPLES OF MODELS AND INSTANCES OF SLIPS

Let us take the case of the Space Physics data. The examples below supplement the examples provided with Descriptors (see section 3.4.3).

The Models of Slips defined for Space Physics during the Formal Definition Phase, are then instantiated during the Transfer Phase for each delivery of object or description of collection.

In the following example, 4 specific models of Slips have been defined: TNR_COLLECTION, TNR_L2_DO, TNR_L2_CDO_EAST and TNR_L2_CDO_DOC. Without the presence of attribute "METADATA_FILE " for the delivery of the document, it could have been planned to have a single Slip of delivery of the 2 complementary objects: TNR_L2_CDO instead of the two last.

A.2.1 EXAMPLE OF SPECIALIZED COLLECTION SLIP

The name of the attribute METADATA_FILE has been changed to SPASE_METADATA in order to use the domain terminology, the occurrence is 1.

SPACE_PHYSICS_COLLECTION_ SLIP **Modèle of specialized Slip of a Data Objects Collection for Space Physics**			
Attribute		Meaning	Attribute occurrence
SLIP_MODEL_ID		*TNR_COLLECTION*	*1..1*
OBJECT_ID		Collection Identifier	1..1
DESCRIPTOR_ID		Identifier of the Collection Descriptor to which the object refers	1..1
SPASE_METADATA_FILES			1..1
	SPASE_METADATA_FILE	Name of the metadata file Consistent with to the data model of SPASE consortium (Space Physics Archive Search and Extract).	1..1

In our case, there is only one collection of the type WIND_WAVES_TNR_L2, that's why the value of OBJECT_ID is identical for the two attributes.

Example of instance of Slip for a description of Data Object Collection for Space Physics SPACE_PHYSICS_COLLECTION_SLIP	
Attribute	**Value**
SLIP_MODEL_ID	*TNR_COLLECTION*
OBJECT_ID	WIND_WAVES_TNR_L2
DESCRIPTOR_ID	WIND_WAVES_TNR_L2
SPASE_METADATA_FILES	
SPASE_METADATA_FILE	Spase_wind_waves.xml

The specialization defined here for Space Physics can, if necessary, being specialized for such or such mission or scientific experiment.

A.2.2 EXAMPLE OF SPECIALIZED DATA OBJECT SLIP

Each Data Object consists of a file and only one transferred with its Slip.
There is no external metadata file. On the other hand, it was agreed to provide start and end time of the data in the Slip.

The Model of Slip for the Data Objects of this Collection could thus be:

SPACE_PHYSICS_ SLIP Model of Slip for a Data Object for Space Physics		
Attribute	**Meaning**	**Attribute occurrence**
SLIP_MODEL_ID	*TNR_L2_DO*	*1..1*
OBJECT_ID	Object identifier	1..1
DESCRIPTOR_ID	Identifier of the Object Descriptor to which the object refers	1..1
TRANSFERRED_DO_FILES	Data file(s) which make up this object.	1..1
LAST_OBJECT	Indicator specifying that it is the last object of the Collection	0..1
START_DATE	Start date for the Data Object	1..1
STOP_DATE	End date for the Data Object	1..1

These instances of Slip can for example take the following form:

Example of instance of Slip for a Data Object for Space Physics SPACE_PHYSICS_SLIP	
Attribute	**Value**
SLIP_MODEL_ID	*TNR_L2_DO*
OBJECT_ID	Wind_waves_tnr_l2_20040610
DESCRIPTOR_ID	WIND_WAVES_TNR_L2_DATA
TRANSFERRED_DO_FILES	TRANSFERRED_FILE = Wind_waves_tnr_l2_20040610.dat
LAST_OBJECT	FALSE
START_DATE	2004/06/10-00 :00 :00
STOP_DATE	2004/06/10-23 :59 :59

A.2.3 EXAMPLE OF SPECIALIZED COMPLEMENTARY DATA OBJECT SLIPS

Underneath 2 models of specialized Slips (an EAST file, a textual document).

The value of the DESCRIPTOR_ID is fixed at EAST or DOCUMENTATION. The occurrence of the DO_DIGITAL_CONTAINER is 1.There is no external file of reference.

EAST_SLIP **Model of specialized Slip for an EAST file for Space Physics**		
Attribute	Meaning	Attribute occurrence
SLIP_MODEL_ID	*TNR_L2_CDO_EAST*	*1..1*
OBJECT_ID	Data Object identifier	1..1
DESCRIPTOR_ID	EAST_DESCRIPTION	1..1
TRANSFERRED_DO_FILES	Name(s) of the data file(s) which make up this object	1..1

Example of instance of Slip for a Complementary Data Object for Space Physics **EAST_SLIP**	
attribut	**valeur**
SLIP_MODEL_ID	*TNR_L2_CDO_EAST*
OBJECT_ID	WIND_WAVES_TNR_L2.EAST
DESCRIPTOR_ID	EAST_DESCRIPTION
TRANSFERRED_DO_FILES	TRANSFERRED_FILE = WIND_WAVES_TNR_L2.EAST

EXPERIMENT_DESCRIPTION_SLIP **Model of specialized Slip for a document file for Space Physics**		
Attribute	Meaning	Attribute occurrence
SLIP_MODEL_ID	*TNR_L2_CDO_DOC*	*1..1*
OBJECT_ID	Data Object identifer	1..1
DESCRIPTOR_ID	DOCUMENTATION	
METADATA_FILE	Name of the data file to be transferred	1..1
TRANSFERRED_DO_FILES	Name(s) of the data file(s) which make up this object	1..1

Example of instance of Slip for a Complementary Data Object for Space Physics **EXPERIMENT_DESCRIPTION_SLIP**	
attribut	**valeur**

SLIP_MODEL_ID	*TNR_L2_CDO_DOC*
OBJECT_ID	WIND_WAVES_22000_0387
DESCRIPTOR_ID	DOCUMENTATION
TRANSFERRED_METADATA_FILES	TRANSFERRED_FILE = Wind_waves_22000.xml
TRANSFERRED_DO_FILES	TRANSFERRED_FILE = WIND_WAVES_22000_0387.PDF

The following figure 3 shows the delivery of 2 Slips during the Transfer Phase. Each transmitted Slip makes it possible to point to the object of the model which represents it: the delivered Spase_wind_waves.xml file is attached to the node WIND_WAVES, the Wind_waves_tnr_l2_20040610 file is the first Data Object delivered from the collection WIND_WAVES_TNR_L2 whose representative is WIND_WAVES_TNR_L2_DATA.

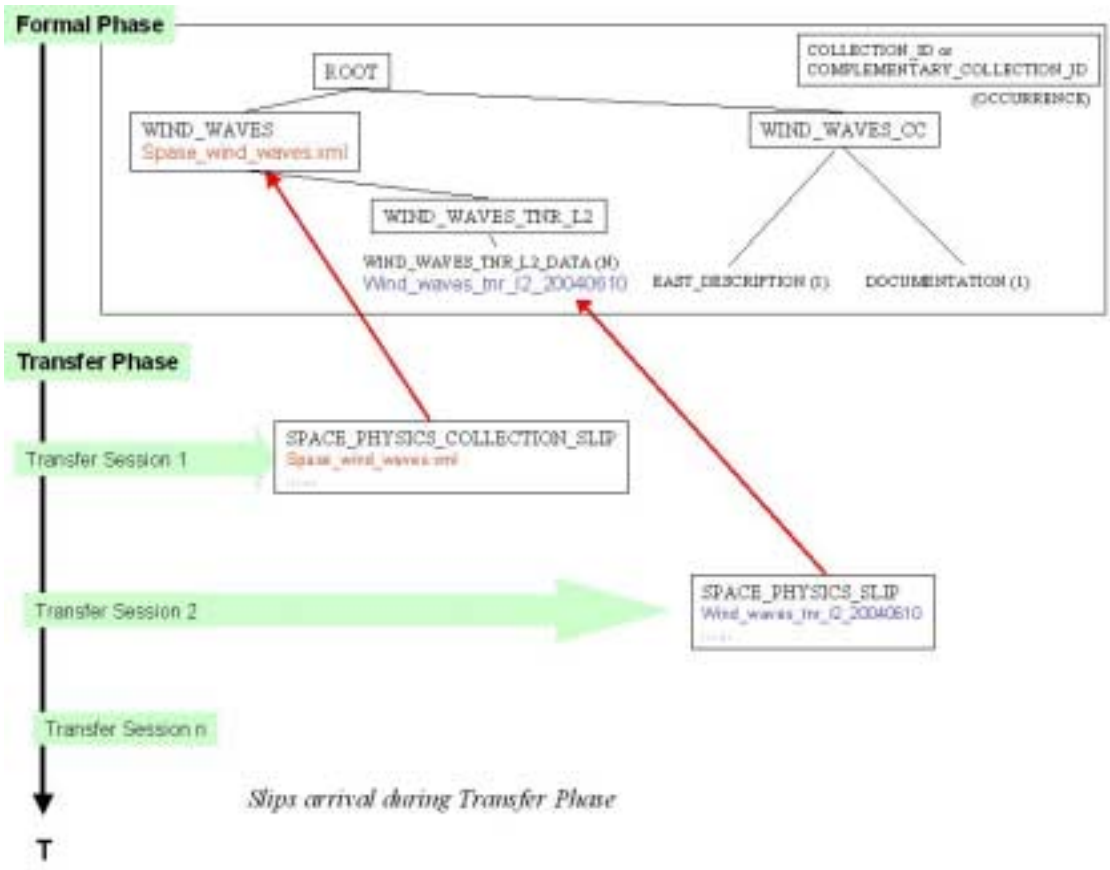

Figure 6

A.3 EXAMPLES OF MODELS AND INSTANCESOF SIPS

In this example, the objects to be delivered are gathered in 3 SIPs:

- SIP1 : Collection,
- SIP2 : complementary objects EAST file and document,
- SIP3 : Data Objects.

Concerning the scheduling constraints, it is imposed that the SIP2 is delivered before any delivery of SIP3. There is no constraint on SIP1.

SIP_MODEL		
Attribute	**Meaning**	**Attribute occurrence**
SIP_MODEL_ID	*SIP1*	*1..1*
DESCRIPTOR_ID	Identifier of the Object Descriptor to which the object refers in the SIP.	1..N

Example of SIP1 instance	
Attribute	**Value**
SIP_MODEL_ID	*SIP1*
DESCRIPTOR_ID	WIND_WAVES_TNR_L2 (1..1)

SIP_MODEL		
Attribute	**Meaning**	**Attribute occurrence**
SIP_MODEL_ID	*SIP2*	*1..1*
DESCRIPTOR_ID	Identifier of the Object Descriptor to which the object refers in the SIP.	1..N
TIME_CONSTRAINT_GROUP	Defines the groups	1..1
SERIAL_NUMBER_IN_CONSTRAINT	Defines the constraints within the same group	1..1

Exampleof SIP2 instance	
Attribute	**Value**
SIP_MODEL_ID	*SIP2*
DESCRIPTOR_ID	EAST_DESCRIPTION
DESCRIPTOR_ID	DOCUMENTATION
TIME_CONSTRAINT_GROUP	Groupe_1
SERIAL_NUMBER_IN_CONSTR	1

AINT	

SIP_MODEL		
Attribute	**Meaning**	**Attribute occurrence**
SIP_MODEL_ID	*SIP3*	*1..1*
DESCRIPTOR_ID	Identifier of the Object Descriptor to which the object refers in the SIP.	1..N
TIME_CONSTRAINT_GROUP	Defines the groups	1..1
SERIAL_NUMBER_IN_CONSTRAINT	Defines the constraints within the same group	1..1

Exemple d'instance SIP3	
Attribute	**Value**
SIP_MODEL_ID	*SIP3*
SIP_CONTENT	WIND_WAVES_TNR_L2_DATA
TIME_CONSTRAINT_GROUP	Groupe_1
SERIAL_NUMBER_IN_CONSTRAINT	2

ANNEX B : INFORMATIVE REFERENCES

(This annex **is not** part of the Recommendation)

This annex provides a list of references that may be valuable to the user of this Recommendation as background material or to provide implementation guidelines for using this Recommendation.

[B1] D*raft recommandation pour els XFDU*