# Use Cases

**[TOPIC – Part to be merged back to 651x2g0-[1-5,8]-core.docx document]**

## CoRoT – End of Mission Bulk Transfer

### Context and Benefits

This use case deals with the transfer of a full set of science and auxiliary data acquired by the CoRoT space mission. The science data is in an unprocessed form that is referred to as ‘level 0’, or sometimes as ‘raw’ data. The auxiliary data includes data about the spacecraft and instrument status, such as currents and temperatures, that is referred to as ‘housekeeping’ data. It also includes data from instrument calibration runs. The setup of this use case was contemporary to the actual transfer of CoRoT data from the acquisition centers to the long-term archiving center at CNES after the end of the mission. It served as a PAIS demonstration of capabilities for the bulk transfer of medium-large data sets.

CoRoT is a space astronomy mission devoted to the study of the variability with time of a stars brightness, with an extremely high accuracy (100 times better than from the ground), on very long durations (up to 150 days) and a very high duty cycle (more than 90%). The original scientific objectives were focused on the study of stellar pulsations (asteroseismology) and the detection of small exoplanets. However, the data collected are now feeding many domains of stellar physics. The mission was led by CNES in association with four French laboratories, and seven participating countries and agencies (Austria, Belgium, Brazil, Germany, Spain, and the ESA Science Programme). It was launched on December 27, 2006 by a Soyuz Rocket, from Baikonur. The mission lasted almost six years (the nominal three years duration and a three years extension) and has observed more than 160,000 stars. It stopped sending data suddenly on November 2, 2012.

This use case provides an example of PAIS configurations for bulk transfers and especially highlights the following benefits:

* Standardization associated with Transfer Objects and SIPs allows the Archive to efficiently recognize, ingest and manage the different types of data (e.g., housekeeping and scientific). This also facilitates data dissemination..
* Control of the sequencing of the transferred SIPs, such as requiring that the housekeeping data must be transferred prior to the level 0 data, facilitates efficient Archive management functions;
* Validation by the Archive of fixity, sizes, and occurrences can be performed starting from the first SIP received and continued during the overall period of transfer, rather than only at the very end of the transfer process. This provides increased efficiency and effectiveness;
* The declarative model provided by the PAIS descriptors provides semantics that can help the Archive to classify the input SIPs and to minimize the metadata extraction activity.

This use case does not cover the transfer of documents or representational information about the content of the transferred files. That information was to be transferred through a separate means. The transfer of metadata, documentation and the relationships with the actual data is covered by the ISEE and POLDER use cases, see sections 6.1 and 6.2 respectively.

### Objects to be Transferred

On the Producer side, the CoRoT repository of Level 0 data and accompanying auxiliary data is composed of about 460,000 files representing a total size of about 3.2 Tb. The file size spans from 1 Kb to 1.2 Gb with an average value of 1.5 Mb.

The logical layout of the repository is depicted in the following **Figure 6-1** below. The logical layout, also known as static layout, represents the hierarchy of directory and file “types” with their cardinalities. It does not represent the full list of directory and file instances.



**Figure 6-1: CoRoT Repository – Logical Layout**

Once again, the CoRoT data to be transferred consists of auxiliary (housekeeping and calibration) and Level 0 data.

The housekeeping data are those dedicated to the Level 0 data only and are filed under an N0\_HK folder of the repository. The N0 is an abbreviation of “Niveau 0” meaning “Level 0” in French, and the HK stands for Housekeeping. The housekeeping data is then distributed in 20 distinct series of parameters as currents, temperatures, etc. The folders of “HK SERIE” type depicted in the above figure can take the following names: FRACTIOPPS1, FRACTIOPPS2, LATCHEDOBT, …. Those folders contain a variable number of files formatted according to the Flexible Image Transport System (FITS) specifications (a common space science format).

The Level 0 data is filed in an N0 folder containing 28 sub-folders corresponding to the CoRoT observation “runs” that represent continuous observations of 20 to 150 days. The run folder names follow the pattern RUN{NN}\_{CODE} where {NN} is a counter and the {CODE} is a non-null string whose definition will not be detailed here e.g. RUN12\_LRC03. The counter starts with ‘03’. A separate run named CALIBRATION was used for calibration and validation of the CoRoT telescope. It has the same structure and content types as the other runs.

Each run folder is further subdivided into subfolders called datasets corresponding to different types of data or different processing levels. A dataset is a set of files in FITS format that are archived and compressed in tar/gz format. A run may not contain all datasets but most of them include a majority of the datasets defined in the following table:

Table -: CoRoT Level 0 Datasets

| **Dataset** | **Total Size** | **Max.File Size** | **File Number** | **Title** |
| --- | --- | --- | --- | --- |
| AN0\_BKGROUND | 16.8 Gb | 392 Mb | 190 | Asteroseismology Background |
| AN0\_ECARTO\_AFPS | 6.8 Gb | 404 Mb | 43 | Ecartometry Fine Pointing mode |
| AN0\_ECARTO\_ARPS | 66.5 Mo | 14 Mb | 38 | Ecartometry Rough Pointing mode |
| AN0\_ECARTO\_Undefined | 1.1 Mb | 0.2 Mb | 60 | Ecartometry mode non-detected |
| AN0\_FULLIMAGE | 549 Mb | 29.6 Mb | 76 | Asteroseismology Full Image |
| AN0\_FULLWINDOW | 317.3 Mb | 35.8 Mb | 36 | Asteroseismology Full Window |
| **AN0\_IMAGETTE**➊ | **82.80 Gb** | **1.3 Gb** | **194** | **Asteroseismology imagette** |
| AN0\_MASK | 164 Kb | 2 Kb | 141 | Asteroseismology templates |
| AN0\_OFFSET | 6.62 Gb | 268 Mb | 58 | Asteroseismology Offset |
| AN0\_STARWIND | 15.85 Gb | 185 Mb | 216 | Asteroseismology channel |
| AN0\_THRESHOLDIMAGE | 65.7 Gb | 11 Mb | 16 | Asteroseismology Threshold Image |
| EN0\_BKGROUND\_MONOCHROM | 1.37 Gb | 1.2 Mb | 5544 | Monochr. Exoplanet Bkgd. |
| EN0\_BKGROUND\_SAMPLEM | 3.04 Gb | 7.6 Mb | 1623 | Monochr. Oversampled Exoplanet Bkgd. |
| EN0\_BRIGHT\_PIX\_32 | 1.64 Gb | 5.3 Mb | 1372 | Exoplanet sky bkg. impacted pixels 32 |
| EN0\_BRIGHT\_PIX\_512 | 710 Mb | 768 Kb | 4116 | Exoplanet sky bkg. impacted pixels 512 |
| EN0\_FULLIMAGE | 1.71 Gb | 111 Mb | 74 | Exoplanet Full Image |
| EN0\_FULLWINDOW | 282.2 Mb | 663 Kb | 6898 | Exoplanet Full Window |
| EN0\_IMAGETTE | 72.64 Gb | 168 Mb | 1015 | Exoplanet imagette |
| EN0\_OFFSET\_SAMPLEM | 1.08 Gb | 32 Mb | 74 | Offset Exoplanet Oversampled Mono. |
| EN0\_STARWIND\_CHROM | 17.34 Gb | 969 Mb | 45987 | Chromatic Exoplanet Observation |
| EN0\_STARWIND\_MONOCHROM | 28.3 Gb | 567 Mb | 120143 | Mono. Exoplanet Observation |
| **EN0\_STARWIND\_SAMPLEC**➋ | **101.7 Gb** | **6.5 Mb** | **41917** | **Chromatic Overspld. Exoplanet Obs.** |
| EN0\_STARWIND\_SAMPLEM | 9.48 Gb | 3.7 Mb | 7777 | Mono. Oversampled Exoplanet Obs. |
| EN0\_TEMPLATE | 6.08 Mb | 1 Kb | 6658 | Exoplanet Templates |

The scientific or engineering nature of the datasets will not be further described in this use case. Their understanding could have helped to establish a more detailed model of transfer, for example by describing relationships between the datasets. However, the above table provides the size and count properties that are significant for the design of the transfer model. For example, the total size of the EN0\_STARWIND\_SAMPLEC ➋ exceeds 100 Gb and shows that it is probably not a good idea to define CoRoT datasets as the smallest unit for the transfer model. A modeling down to the file level would allow the transfer of packets of more reasonable sizes. At the file level, the table shows that the maximum size of the individual files is 1.3 Gb for the AN0\_IMAGETTE ➊ which is the smallest maximum Transfer Object size that the model can impose for this dataset since the PAIS does not allow a file to span over multiple Transfer Objects or SIPs. So the maximum file size is an interesting property used to define the transfer. If this value is not reasonable, it would have been necessary to consider a change, for example by slicing some of the big files into chunks of acceptable sizes.

The following **Figure 6-2** shows a partial view of the physical layout of the CoRoT repository. Due to the large number of files, it is not possible to depict all of them, but the figure provides actual examples of folder and file names for CoRoT runs, datasets, or housekeeping series.



**Figure 6-2: CoRoT Repository Physical Layout**

### Model of Objects for Transfer and SIP Constraints

The transfer of a full run as a single SIP is not practical because some may exceed 100Gb. The transfer model built splits the runs into subparts of less than 4 Gb, but with the assurance that each transferred SIP deals with only one run and only one type of dataset e.g. RUN01 and AN0\_BACKGROUND. However, a single run may require multiple SIPs to be fully transferred, depending of the total size of the dataset.

#### MOT

The XML descriptors are provided in annex XX.

The Model of Objects for Transfer, which is a hierarchical tree, can be summarized as follows:

* One root collection “CoRoT-N0”.
* A first Transfer Object Type “COROT-N0-RUN-PRODUCT-SET” represents the payload stream of CoRoT N0 products. There can be an unlimited number of objects of this type in the transfer project but each has a limited size of 4 Gb. It is made up of:
	+ A “COROT-N0-Run” Group Type. This Group Type assures that a “COROT-N0-RUN-PRODUCT-SET” object contains data dealing with one and only one RUN.
	+ A “COROT-N0-Product-Type” Group Type. This Group Type assures that a “COROT-N0-RUN-PRODUCT-SET” object contains data of the same set e.g. AN0\_BACKGROUND.



**Figure 6-3: CoRoT Model of Objects For Transfer**

* A second Transfer Object Type “COROT-N0-HK-SET” represents the payload stream of CoRoT HK auxiliary data. There can be an unlimited number of objects of this type in the transfer project with no constraint on size. It contains;
	+ A “COROT-N0-HK-Type” Group Type. This Group Type assures that a “COROT-N0-HK-SET” object contains data dealing with one and only one series e.g. FRACTIOPPS1.

Figure 6-4 below is a snapshot of the MOT viewed through the CNES prototype. It shows the Collections and the Transfer Objects levels as described in this section.



**Figure 6-4: CoRoT Model of Objects For Transfer**

#### SIP Constraints

The SIP Constraints XML document is provided in Annex D4.

This test case defines two SIP Content Types, one for each of the categories of data:

* **SIP-COROT-N0-PRODUCT-SET**: authorizing only one COROT-N0-RUN-PRODUCT-SET Transfer Object per SIP of this type. It corresponds to one homogeneous set of dataset of the same type accumulated up to a maximum of 4 Gb;
* **SIP-COROT-N0-HK-SET**: authorizing only one COROT-N0-HK-SET Transfer Object per SIP of this type. It corresponds to one homogeneous set of FITS files accumulated without limit of count or size.

This test case also defines a sequence order imposing the housekeeping data to be transferred first or at least before any N0 product.

Currently SIP Constraints are created manually via an XML editing tool or a text editing tool.

### SIPs

#### SIPs Generation

In this case, the SIPs are generated from the CoRot repository via the SIP Builder software, see section 7.2, with a configuration file provided in. The generated SIPs are 'XFDU PAIS SIP Conformant' as defined in the PAIS BB.

173 SIPs have been generated with the first 20 SIPs conveying Housekeeping auxiliary data followed by 153 packages of N0 products. More SIPs should have been generated from the CoRoT repository but for the demonstration the process has been voluntarily limited to the first nine runs i.e. RUN01 to RUN09.

#### SIPs Contents

The first series of SIPs are, as expected, of SIP-COROT-N0-HK type illustrated by the abstract tree below:

├── **COROT-N0-SIP-0001.zip <──── First HK SIP**

│   ├── N0\_HK

│   │   └── FRACTIOPPS1

│   │   ├── HK\_FRACTIOPPS1\_P\_P\_20070101T080503\_20070117T235951.fits

│   │   ├── HK\_FRACTIOPPS1\_P\_P\_20070118T000023\_20070402T235948.fits

│   │   ├── ...

│   │   ├── HK\_FRACTIOPPS1\_P\_P\_20120705T000009\_20121001T235932.fits

│   │   └── HK\_FRACTIOPPS1\_P\_P\_20121001T000004\_20121103T235941.fits

│   └── xfdumanifest.xml

│

├── **COROT-N0-SIP-0002.zip**

│   ├── N0\_HK

│   │   └── FRACTIOPPS2

│   │   ├── HK\_FRACTIOPPS2\_P\_P\_20070101T080503\_20070117T235951.fits

│   │   ├── HK\_FRACTIOPPS2\_P\_P\_20070118T000023\_20070402T235948.fits

│   │   ├── ...

│   │   ├── HK\_FRACTIOPPS2\_P\_P\_20120705T000009\_20121001T235932.fits

│   │   └── HK\_FRACTIOPPS2\_P\_P\_20121001T000004\_20121103T235941.fits

│   └── xfdumanifest.xml

│

├── **COROT-N0-SIP-0003.zip**

│   ├── N0\_HK

│   │   └── LATCHEDOBT

│   │   ├── HK\_LATCHEDOBT\_P\_P\_20070101T080431\_20070117T235951.fits

│   │   ├── HK\_LATCHEDOBT\_P\_P\_20070118T000023\_20070402T235948.fits

│   │   ├── ...

│   │   ├── HK\_LATCHEDOBT\_P\_P\_20120705T000009\_20121001T235932.fits

│   │   └── HK\_LATCHEDOBT\_P\_P\_20121001T000004\_20121103T235941.fits

│   └── xfdumanifest.xml

│

├── ...

│

└── **COROT-N0-SIP-0020.zip** **<──── Last HK SIP**

    ├── N0\_HK

    │   └── ZIZM2GC

    │   ├── HK\_ZIZM2GC\_P\_P\_20070226T103801\_20070402T235956.fits

    │   ├── HK\_ZIZM2GC\_P\_P\_20070402T000004\_20070509T235954.fits

    │   ├── ...

    │   ├── HK\_ZIZM2GC\_P\_P\_20120705T000001\_20121001T235956.fits

    │   └── HK\_ZIZM2GC\_P\_P\_20121001T000004\_20121103T235957.fits

    └── xfdumanifest.xml

The following tables show snippets of the Manifest xfdumanifest.xml file of the first SIP containing the first housekeeping series. The extracts are focused on the PAIS extension elements and are generally stripped of XFDU elements. These latter are reported only when they bring value to the example and are explicitly prefixed by xfdu to avoid confusion, although that may not be the case in the actual manifest file. The complete Manifest document is provided in section E1 of Annex E.

Table 6-: SIP-COROT-N0-HK SIP Manifest – Header

| **Element** | **Content** |
| --- | --- |
| **sipGlobalInformation** ➊ |  |
|  sipID | **COROT-N0-SIP-0001** |
| producerSourceID | CNES |
|  producerArchiveProjectID | COROT-N0 |
|  sipContentTypeID | **SIP-COROT-N0-HK** |
| sipSequenceNumber | 1 |

The Header contains general information associated to the whole package: identifier of SIP (created during SIP construction , unicity to be checked among the Producer-Archive Project), type of SIP (checked against the SIP constraints where it has been defined). The Producer Source ID and the Producer-Archive Project ID make links with the MOT where they have been defined (the Producer-Archive Project ID is the ID of the root node in the MOT).

Table -: SIP-COROT-N0-HK SIP Manifest – Information Package Map

|  |  |
| --- | --- |
| **sipTransferObject** ➋ |  |
| descriptorID | **COROT-N0-HK** |
|  transferObjectID | COROT-N0-HK-0001 |
|  lastTransferObjectFlag | FALSE |
|  **sipTransferObjectGroup** ➌ |  |
|  associatedDescriptorGroupTypeID | COROT-N0-HK-GROUP |
|  transferObjectGroupInstanceName | **N0-HK/FRACTIOPPS1** |
|  **sipDataObject** |  |
|  associatedDescriptorDataID | **COROT-N0-HK-DATA** |
|  xfdu:dataObjectPointer |  |
|  ➍ @dataObjectID | **DO-COROT-N0-HK-DATA-0001** |
|  **sipDataObject** |  |
|  associatedDescriptorDataID | COROT-N0-HK-DATA |
|  xfdu:dataObjectPointer |  |
|   @dataObjectID | **DO-COROT-N0-HK-DATA-0029** |

The Information Package Map describes the hierarchical content of the package by making links with the MOT through the Descriptor, Descriptor Group Types, Descriptor Data IDs. These IDs are checked against the MOT for conformity with the expected Data Objects. The Transfer Object ID is inserted during SIP building, and identifies the Transfer Object. This ID should be kept in a log, for potential update or deletion (this is the lowest delivery granule). Transfer Object Group Instance Name ?

The Information Package Map also points towards the physical Data Objects in the Data Object section through the Data Object Pointers.

Table -: SIP-COROT-N0-HK SIP Manifest – Data Object Section

|  |  |
| --- | --- |
| **xfdu:dataObject** |  |
|  @ID | **DO-COROT-N0-HK-DATA-0001** |
|  **byteStream** |  |
|  **fileLocation** |  |
|  @locatorType | URL |
|  @href **N0\_HK/FRACTIOPPS1/HK\_FRACTIOPPS1\_P\_P\_20070101T080503\_20070117T235951.fits** |
|  **Checksum** | d41d8cd98f00b204e9800998ecf8427e |
|  @checksumName | MD5 |
| **xfdu:dataObject** |  |
|  @ID | **DO-COROT-N0-HK-DATA-0029** |
|  **byteStream** |  |
|  **fileLocation** |  |
|  @locatorType | URL |
|  @href **N0\_HK/FRACTIOPPS1/HK\_FRACTIOPPS1\_P\_P\_20121001T000004\_20121103T235941.fits** |
|  **Checksum** | d41d8cd98f00b204e9800998ecf8427e |
|  @checksumName | MD5 |

The Data Object Section contains the physical location of the Data Objects as described in the Information Package Map. This is also the place to indicate checksums or file sizes.

The second series of SIPs are, as expected, of SIP-COROT-N0-RUN type illustrated by the abstract tree below:

├── **COROT-N0-SIP-0021.zip <──── First N0 for ‘RUN’ data SIP**

│   ├── N0

│   │   └── RUN03\_IRA01

│   │   └── AN0\_BKGROUND

│   │   ├── 79.tar.gz

│   │   ├── 80.tar.gz

│   │   ├── 81.tar.gz

│   │   ├── 82.tar.gz

│   │   └── 83.tar.gz

│   └── xfdumanifest.xml

│

├── **COROT-N0-SIP-0022.zip**

│   ├── N0

│   │   └── RUN03\_IRA01

│   │   └── AN0\_ECARTO\_AFPS

│   │   ├── 0000000116.tar.gz

│   │   └── 0000000223.tar.gz

│   └── xfdumanifest.xml

│

├── **COROT-N0-SIP-0023.zip**

│   ├── N0

│   │   └── RUN03\_IRA01

│   │   └── AN0\_ECARTO\_ARPS

│   │   ├── 0000000116.tar.gz

│   │   └── 0000000223.tar.gz

│   └── xfdumanifest.xml

│

├── ...

│

└── **COROT-N0-SIP-0173.zip <──── Last N0 SIP**

    ├── N0

    │   └── RUN09\_SRC02

    │   └── EN0\_TEMPLATE

    │   ├── 0.tar.gz

    │   ├── 100.tar.gz

    │   ├── ...

    │   ├── 98.tar.gz

    │   └── 99.tar.gz

    └── xfdumanifest.xml

SIP-021 to SIP-0173 contains the scientific level 0 data. The tabulated snippets represent the 3 nested repositories containing the data grouped in the form of tar limited in size (as specified in the MOT).

#### SIPs Ingestion

TOPIC – In this case, the SIPs are submitted and ingested by the CNES Prototype, see section 7.2.

TOPIC – The SIP Prototype main validation and ingestion steps:

1. Read project MOT and SIP Constraints
2. **For each SIP received**
	1. Inflate input ZIP
	2. Open XFDU Manifest
	3. Check project ID
	4. Check SIP Type ID
	5. Check SIP sequence number; in some cases the sequence number could be implemented in the SIP file name.
	6. **For each Content Unit annotated as Transfer Object**
		1. Check that Transfer Object Type ID is allowed for the current SIP Type
		2. Check that last Transfer Object of this type has not already been ingested
		3. Check maximum occurrence with respect to a project global counter for each Transfer Object type
		4. Update project global counter for the Transfer Object type
		5. **For each Group Type**
			1. Check maximum occurrence of this Group in the parent Group or Transfer Object type
			2. **For each Data Object Type**
				1. Check maximum occurrence of this Data Object in the parent Group
				2. **For each Data Object File**

Check maximum occurrence of this Data Object File in the current Data Object

Check Data Object file presence according to the URL

Verify file size

Compute/update and check maximum size of this Transfer Object

Verify file checksum

Verify that this file was not already ingested

Compute internal file path according to the name preservation rule of the current Transfer Object type

Copy/move file to the internal file path

Register file as ingested

* + - * 1. Check minimum occurrence of Data Object File(s) in this Data Object
			1. Check minimum occurrence of Data Object(s) in this Group
		1. Check minimum occurrence of Group(s) in this Transfer Object
	1. Check minimum size of this Transfer Object
	2. Increment a counter for this Transfer Object type
	3. Cleanup SIP and temporary inflated files
	4. If flagged as last Transfer Object
		1. Check that the counter for this Transfer Object type reaches or exceed the minimum occurrence
		2. Store that the last Transfer Object of this type has been received
1. Close project

In this case the Archive internal repository is a reconstruction of the original CoRoT repository on the Producer side. The 3 examples below show the progressive construction of the repository on the Archive side after ingestion.

Example of Archive internal repository after ingestion of the first SIP of CoRoT Housekeeping data.

└── **N0\_HK**

    └── **FRACTIOPPS1** **<──── From the first HK SIP**

   ├── HK\_FRACTIOPPS1\_P\_P\_20070101T080503\_20070117T235951.fits

    ├── HK\_FRACTIOPPS1\_P\_P\_20070118T000023\_20070402T235948.fits

    ├── ...

    ├── HK\_FRACTIOPPS1\_P\_P\_20120705T000009\_20121001T235932.fits

    └── HK\_FRACTIOPPS1\_P\_P\_20121001T000004\_20121103T235941.fits

Example of Archive internal repository after ingestion of second SIP of CoRoT Housekeeping data.

└── N0\_HK

   ├── FRACTIOPPS1

   │ ├── HK\_FRACTIOPPS1\_P\_P\_20070101T080503\_20070117T235951.fits

    │ ├── HK\_FRACTIOPPS1\_P\_P\_20070118T000023\_20070402T235948.fits

    │ ├── ...

    │ ├── HK\_FRACTIOPPS1\_P\_P\_20120705T000009\_20121001T235932.fits

    │ └── HK\_FRACTIOPPS1\_P\_P\_20121001T000004\_20121103T235941.fits

    └── **FRACTIOPPS2** **<──── From the second HK SIP**

    ├── HK\_FRACTIOPPS2\_P\_P\_20070101T080503\_20070117T235951.fits

    ├── HK\_FRACTIOPPS2\_P\_P\_20070118T000023\_20070402T235948.fits

    ├── ...

    ├── HK\_FRACTIOPPS2\_P\_P\_20120705T000009\_20121001T235932.fits

    └── HK\_FRACTIOPPS2\_P\_P\_20121001T000004\_20121103T235941.fits

Example of Archive internal repository after ingestion of first SIP of CoRoT Level 0 data.

├── N0\_HK

│   ├── FRACTIOPPS1

│  │ ├── HK\_FRACTIOPPS1\_P\_P\_20070101T080503\_20070117T235951.fits

│   │ ├── HK\_FRACTIOPPS1\_P\_P\_20070118T000023\_20070402T235948.fits

│   │ ├── ...

│   │ ├── HK\_FRACTIOPPS1\_P\_P\_20120705T000009\_20121001T235932.fits

│   │ └── HK\_FRACTIOPPS1\_P\_P\_20121001T000004\_20121103T235941.fits

│   ├── FRACTIOPPS2

│   │ ├── HK\_FRACTIOPPS2\_P\_P\_20070101T080503\_20070117T235951.fits

│   │ ├── HK\_FRACTIOPPS2\_P\_P\_20070118T000023\_20070402T235948.fits

│   │ ├── ...

│   │ ├── HK\_FRACTIOPPS2\_P\_P\_20120705T000009\_20121001T235932.fits

│   │ └── HK\_FRACTIOPPS2\_P\_P\_20121001T000004\_20121103T235941.fits

│ │

│ └── … **<──── Other HK folders not represented here for brevity**

│

└── **N0**

 └── **RUN03\_IRA01**

 └── **AN0\_BKGROUND** **<──── From the first N0 SIP**

 ├── 79.tar.gz

 ├── 80.tar.gz

 ├── 81.tar.gz

 ├── 82.tar.gz

 └── 83.tar.gz

1. CoRoT Use Case – Descriptors

This annex contains the PAIS XML descriptors of the CoRoT use case (see section 6.4.3).

* 1. CoRoT N0 – Collection Descriptor

<?xml version="1.0" encoding="UTF-8"?>

<**collectionDescriptor** xmlns="urn:ccsds:schema:pais:1">

 <identification>

 <descriptorModelID>CCSD0015</descriptorModelID>

 <descriptorModelVersion>1.0</descriptorModelVersion>

 <**descriptorID**>**COROT-N0**</**descriptorID**>

 </identification>

 <description>

 <collectionTitle>CoRoT N0 Collection</collectionTitle>

 <collectionDescription>

Collection of CoRoT N0 data

 </collectionDescription>

 </description>

 <relation>

 <**parentCollection**>**NONE**</**parentCollection**>

 </relation>

</**collectionDescriptor**>

* 1. CoRoT N0 Products – Transfer Object Descriptor

<?xml version="1.0" encoding="UTF-8"?>

<**transferObjectTypeDescriptor** xmlns="urn:ccsds:schema:pais:1">

 <identification>

 <descriptorModelID>CCSD0014</descriptorModelID>

 <descriptorModelVersion>V1.0</descriptorModelVersion>

 <**descriptorID**>**COROT-N0-RUN** </**descriptorID**>

 <producerSourceID>CNES</producerSourceID>

 </identification>

 <description>

 <transferObjectTypeTitle>

 CoRoT N0 RUN

 </transferObjectTypeTitle>

 <transferObjectTypeDescription>

 A set of CoRoT N0 Dataset of the same type and belonging

 to a single Run.

 </transferObjectTypeDescription>

 <**transferObjectTypeOccurrence**>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxUnknown**/>

 </**transferObjectTypeOccurrence**>

 <**transferObjectTypeSize**>

 <**maxSize**>**4**</**maxSize**>

 <**unitsType**>**GB**</**unitsType**>

 </**transferObjectTypeSize**>

 </description>

 <relation>

 <**parentCollection**>**COROT-N0**</**parentCollection**>

 </relation>

 <groupType>

 <**groupTypeID**>**COROT-N0-RUN**</**groupTypeID**>

 <groupTypeDescription>

 A group denoting a single Run.

 </groupTypeDescription>

 <groupTypeStructureName>directory</groupTypeStructureName>

 <**groupTypeOccurrence**>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxOccurrence**>**1**</**maxOccurrence**>

 </**groupTypeOccurrence**>

 <groupType>

 <**groupTypeID**>**COROT-N0-DATASET-GROUP**</**groupTypeID**>

 <groupTypeDescription>

 A group of CoRoT N0 Dataset of the same type.

 </groupTypeDescription>

 <groupTypeStructureName>directory</groupTypeStructureName>

 <**groupTypeOccurrence**>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxOccurrence**>**1**</**maxOccurrence**>

 </**groupTypeOccurrence**>

 <dataObjectType>

 <**dataObjectTypeID**>**COROT-N0-DATASET**</**dataObjectTypeID**>

 <dataObjectTypeDescription>

 A CoRoT N0 Dataset

 </dataObjectTypeDescription>

 <**dataObjectTypeOccurrence**>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxUnknown**/>

 </**dataObjectTypeOccurrence**>

 </dataObjectType>

 </groupType>

 </groupType>

</**transferObjectTypeDescriptor**>

* 1. CoRoT Houskeeping Data (HK) – Transfer Object Descriptor

<?xml version="1.0" encoding="UTF-8"?>

<**transferObjectTypeDescriptor** xmlns="urn:ccsds:schema:pais:1">

 <identification>

 <descriptorModelID>CCSD0014</descriptorModelID>

 <descriptorModelVersion>V1.0</descriptorModelVersion>

 <**descriptorID**>**COROT-N0-HK** </**descriptorID**>

 <producerSourceID>CNES</producerSourceID>

 </identification>

 <description>

 <transferObjectTypeTitle>CoRoT N0 - HK </transferObjectTypeTitle>

 <transferObjectTypeDescription>

 A set of CoRoT N0 Housekeeping data

 </transferObjectTypeDescription>

 <**transferObjectTypeOccurrence**>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxUnknown**/>

 </**transferObjectTypeOccurrence**>

 </description>

 <relation>

 <**parentCollection**>**COROT-N0**</**parentCollection**>

 </relation>

 <groupType>

 <**groupTypeID**>**COROT-N0-HK-GROUP**</**groupTypeID**>

 <groupTypeDescription>

 A group type for CoRoT N0 Housekeeping data

 </groupTypeDescription>

 <groupTypeStructureName>directory</groupTypeStructureName>

 <**groupTypeOccurrence**>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxOccurrence**>**1**</**maxOccurrence**>

 </**groupTypeOccurrence**>

 <dataObjectType>

 <**dataObjectTypeID**>**COROT-N0-HK-DATA**</**dataObjectTypeID**>

 <dataObjectTypeDescription>

 A CoRoT N0 housekeeping data

 </dataObjectTypeDescription>

 <**dataObjectTypeOccurrence**>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxUnknown**/>

 </**dataObjectTypeOccurrence**>

 </dataObjectType>

 </groupType>

</**transferObjectTypeDescriptor**>

* 1. CoRoT – SIP Constraints

<?xml version="1.0" encoding="UTF-8"?>

<**sipConstraints** xmlns="urn:ccsds:schema:pais:1">

 <**producerArchiveProjectID**>**COROT-N0**</**producerArchiveProjectID**>

 <!-- SIPs of COROT N0 RUNs -->

 <sipContentType>

 <**sipContentTypeID**>**SIP-COROT-N0-RUN**</**sipContentTypeID**>

 <authorizedDescriptor>

 <**descriptorID**>**COROT-N0-RUN** </**descriptorID**>

 <occurrence>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxOccurrence**>**1**</**maxOccurrence**>

 </occurrence>

 </authorizedDescriptor>

 </sipContentType>

 <!-- SIPs of COROT N0 housekeeping (HK) data -->

 <sipContentType>

 <**sipContentTypeID**>**SIP-COROT-N0-HK** </**sipContentTypeID**>

 <authorizedDescriptor>

 <**descriptorID**>**COROT-N0-HK** </**descriptorID**>

 <occurrence>

 <**minOccurrence**>**1**</**minOccurrence**>

 <**maxOccurrence**>**1**</**maxOccurrence**>

 </occurrence>

 </authorizedDescriptor>

 </sipContentType>

 <!-- Constraints: force HK before RUNs -->

 <sipSequencingConstraintGroup>

 <**groupName**>**CoRoT N0**</**groupName**>

 <constraintItem>

 <**sipContentTypeID**>**SIP-COROT-N0-HK** </**sipContentTypeID**>

 <**constraintSerialNumber**>**1**</**constraintSerialNumber**>

 </constraintItem>

 <constraintItem>

 <**sipContentTypeID**>**SIP-COROT-N0-RUN**</**sipContentTypeID**>

 <**constraintSerialNumber**>**2**</**constraintSerialNumber**>

 </constraintItem>

 </sipSequencingConstraintGroup>

</**sipConstraints**>

1. CoRoT Use Case – Examples of XFDU Manifests

This annex contains examples of SIP XFDU Manifests extracted from the CoRoT use case (see section 6.4.3).

* 1. First SIP of Housekeeping Series

<?xml version="1.0" encoding="UTF-8"?>

<**xfdu:XFDU** xmlns:pais="urn:ccsds:schema:pais:1"

 xmlns:xfdu="urn:ccsds:schema:xfdu:1">

 <**packageHeader** ID="COROT-N0-SIP-0001">

 <volumeInfo>

 <specificationVersion>1.0</specificationVersion>

 </volumeInfo>

 <environmentInfo>

 <extension>

 <**pais:sipGlobalInformation**>

 <**pais:sipID**>**COROT-N0-SIP-0001**</pais:sipID>

 <pais:producerSourceID>CNES</pais:producerSourceID>

 <**pais:producerArchiveProjectID**>**COROT-N0**↵

</pais:producerArchiveProjectID>

 <**pais:sipContentTypeID**>**SIP-COROT-N0-HK**↵

 </pais:sipContentTypeID>

 <pais:sipSequenceNumber>1</pais:sipSequenceNumber>

 </pais:sipGlobalInformation>

 </extension>

 </environmentInfo>

 </packageHeader>

 <**informationPackageMap**>

 <xfdu:contentUnit>

 <extension>

 <**pais:sipTransferObject**>

 <**pais:descriptorID**>**COROT-N0-HK**</pais:descriptorID>

 <**pais:transferObjectID**>**COROT-N0-HK-0001**↵

 </pais:transferObjectID>

 <pais:lastTransferObjectFlag>↵

 FALSE</pais:lastTransferObjectFlag>

 </pais:sipTransferObject>

 </extension>

 <xfdu:contentUnit>

 <extension>

 <**pais:sipTransferObjectGroup**>

 <**pais:associatedDescriptorGroupTypeID**>

 **COROT-N0-HK-GROUP**↵</pais:associatedDescriptorGroupTypeID>

 <**pais:transferObjectGroupInstanceName**>↵

 **N0\_HK/FRACTIOPPS1**</pais:transferObjectGroupInstanceName>

 </pais:sipTransferObjectGroup>

 </extension>

 <xfdu:contentUnit>

 <extension>

 <**pais:sipDataObject**>

 <**pais:associatedDescriptorDataID**>↵

 **COROT-N0-HK-DATA**</pais:associatedDescriptorDataID>

 </pais:sipDataObject>

 </extension>

 <**dataObjectPointer** dataObjectID="**DO-COROT-N0-HK-DATA-0001**"/>

 </xfdu:contentUnit>

 <xfdu:contentUnit>

 <extension>

 <**pais:sipDataObject**>

 <**pais:associatedDescriptorDataID**>↵

 **COROT-N0-HK-Data**</pais:associatedDescriptorDataID>

 </pais:sipDataObject>

 </extension>

 <**dataObjectPointer** dataObjectID="**DO-COROT-N0-HK-DATA-0029**"/>

 </xfdu:contentUnit>

 </xfdu:contentUnit>

 </xfdu:contentUnit>

 </informationPackageMap>

 <**dataObjectSection**>

 <**dataObject** ID="**DO-COROT-N0-HK-DATA-0001**">

 <byteStream size="0">

 <**fileLocation** locatorType="URL"

 href="**file:N0\_HK/FRACTIOPPS1/↵**

 **HK\_FRACTIOPPS1\_P\_P\_20070101T080503\_20070117T235951.fits**"/>

 <checksum checksumName="MD5">↵

 d41d8cd98f00b204e9800998ecf8427e</checksum>

 </byteStream>

 </dataObject>

 <**dataObject** ID="**DO-COROT-N0-HK-DATA-0029**">

 <byteStream size="0">

 <fileLocation locatorType="URL"

 href="**file:N0\_HK/FRACTIOPPS1/↵**

 **HK\_FRACTIOPPS1\_P\_P\_20121001T000004\_20121103T235941.fits**"/>

 <checksum checksumName="MD5">↵

 d41d8cd98f00b204e9800998ecf8427e</checksum>

 </byteStream>

 </dataObject>

 </dataObjectSection>

</xfdu:XFDU>

* 1. First SIP of Level 0 Datasets

<?xml version="1.0" encoding="UTF-8"?>

<**xfdu:XFDU** xmlns:pais="urn:ccsds:schema:pais:1"

 xmlns:xfdu="urn:ccsds:schema:xfdu:1">

 <**packageHeader** ID="COROT-N0-SIP-0021">

 <volumeInfo>

 <specificationVersion>1.0</specificationVersion>

 </volumeInfo>

 <environmentInfo>

 <extension>

 <**pais:sipGlobalInformation**>

 <**pais:sipID**>**COROT-N0-SIP-0021**</pais:sipID>

 <pais:producerSourceID>CNES</pais:producerSourceID>

 <**pais:producerArchiveProjectID**>**COROT-N0**

 </pais:producerArchiveProjectID>

 <**pais:sipContentTypeID**>**SIP-COROT-N0-RUN**

 </pais:sipContentTypeID>

 <pais:sipSequenceNumber>21</pais:sipSequenceNumber>

 </pais:sipGlobalInformation>

 </extension>

 </environmentInfo>

 </packageHeader>

 <**informationPackageMap**>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipTransferObject**>

 <**pais:descriptorID**>**COROT-N0-RUN**

 </pais:descriptorID>

 <**pais:transferObjectID**>**COROT-N0-RUN-0001**

 </pais:transferObjectID>

 <pais:lastTransferObjectFlag>FALSE

 </pais:lastTransferObjectFlag>

 </pais:sipTransferObject>

 </extension>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipTransferObjectGroup**>

 <**pais:associatedDescriptorGroupTypeID**>**COROT-N0-RUN**

 </pais:associatedDescriptorGroupTypeID>

 <**pais:transferObjectGroupInstanceName**>**N0/RUN03\_IRA01**

 </pais:transferObjectGroupInstanceName>

 </pais:sipTransferObjectGroup>

 </extension>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipTransferObjectGroup**>

 <**pais:associatedDescriptorGroupTypeID**>

 **COROT-N0-DATASET-GROUP**

 </pais:associatedDescriptorGroupTypeID>

 <**pais:transferObjectGroupInstanceName**>**AN0\_BKGROUND**

 </pais:transferObjectGroupInstanceName>

 </pais:sipTransferObjectGroup>

 </extension>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipDataObject**>

 <**pais:associatedDescriptorDataID**>**COROT-N0-DATASET**

 </pais:associatedDescriptorDataID>

 </pais:sipDataObject>

 </extension>

 <**dataObjectPointer**

 dataObjectID="**DO-COROT-N0-DATASET-0001**"/>

 </xfdu:contentUnit>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipDataObject**>

 <**pais:associatedDescriptorDataID**>**COROT-N0-DATASET**

 </pais:associatedDescriptorDataID>

 </pais:sipDataObject>

 </extension>

 <**dataObjectPointer**

 dataObjectID="**DO-COROT-N0-DATASET-0002**"/>

 </xfdu:contentUnit>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipDataObject**>

 <**pais:associatedDescriptorDataID**>**COROT-N0-DATASET**

 </pais:associatedDescriptorDataID>

 </pais:sipDataObject>

 </extension>

 <**dataObjectPointer**

 dataObjectID="**DO-COROT-N0-DATASET-0003**"/>

 </xfdu:contentUnit>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipDataObject**>

 <**pais:associatedDescriptorDataID**>**COROT-N0-DATASET**

 </pais:associatedDescriptorDataID>

 </pais:sipDataObject>

 </extension>

 <**dataObjectPointer**

 dataObjectID="**DO-COROT-N0-DATASET-0004**"/>

 </xfdu:contentUnit>

 <**xfdu:contentUnit**>

 <extension>

 <**pais:sipDataObject**>

 <**pais:associatedDescriptorDataID**>**COROT-N0-DATASET**

 </pais:associatedDescriptorDataID>

 </pais:sipDataObject>

 </extension>

 <**dataObjectPointer**

 dataObjectID="**DO-COROT-N0-DATASET-0005**"/>

 </xfdu:contentUnit>

 </xfdu:contentUnit>

 </xfdu:contentUnit>

 </xfdu:contentUnit>

 </informationPackageMap>

 <**dataObjectSection**>

 <**dataObject** ID="**DO-COROT-N0-DATASET-0001**">

 <byteStream size="0">

 <fileLocation locatorType="URL"

 href="**file:N0/RUN03\_IRA01/AN0\_BKGROUND/79.tar.gz**"/>

 <checksum

 checksumName="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>

 </byteStream>

 </dataObject>

 <**dataObject** ID="**DO-COROT-N0-DATASET-0002**">

 <byteStream size="0">

 <fileLocation locatorType="URL"

 href="**file:N0/RUN03\_IRA01/AN0\_BKGROUND/80.tar.gz**"/>

 <checksum

 checksumName="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>

 </byteStream>

 </dataObject>

 <**dataObject** ID="**DO-COROT-N0-DATASET-0003**">

 <byteStream size="0">

 <fileLocation locatorType="URL"

 href="**file:N0/RUN03\_IRA01/AN0\_BKGROUND/81.tar.gz**"/>

 <checksum

 checksumName="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>

 </byteStream>

 </dataObject>

 <**dataObject** ID="**DO-COROT-N0-DATASET-0004**">

 <byteStream size="0">

 <fileLocation locatorType="URL"

 href="**file:N0/RUN03\_IRA01/AN0\_BKGROUND/82.tar.gz**"/>

 <checksum

 checksumName="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>

 </byteStream>

 </dataObject>

 <**dataObject** ID="**DO-COROT-N0-DATASET-0005**">

 <byteStream size="0">

 <fileLocation locatorType="URL"

 href="**file:N0/RUN03\_IRA01/AN0\_BKGROUND/83.tar.gz**"/>

 <checksum

 checksumName="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>

 </byteStream>

 </dataObject>

 </dataObjectSection>

</xfdu:XFDU>

1. CoRoT Use Case – SIP Builder Configuration File

This annex contains an example of SIP Builder software configuration file for the generation of XFDU PAIS Conformant SIPs as described in output of CoRoT use case (see section 6.4.4.1).

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="urn:fr:gael:schema:ccsds:pais:sip-builder:1">

 <descriptors>

 <descriptor file="corot-pais-transfer-object-run.xml"/>

 <descriptor file="corot-pais-transfer-object-hk.xml"/>

 <descriptor file="corot-pais-sip-constraints.xml" />

 </descriptors>

 <collectors baseDirectory="../../../test-data/cnes-corot-tds-20140506">

 <!-- N0 Products -->

 <collector typeId="COROT-N0-RUN">

 <include>N0/RUN0.\*</include>

 </collector>

 <collector typeId="COROT-N0-DATASET-GROUP">

 <include>(A|E)N0.\*</include>

 </collector>

 <collector typeId="COROT-N0-DATASET">

 <include>.\*tar.gz</include>

 </collector>

 <!-- N0 HK Data -->

 <collector typeId="COROT-N0-HK-GROUP">

 <include>N0\_HK/.\*</include>

 </collector>

 <collector typeId="COROT-N0-HK-DATA">

 <include> .\*fits</include>

 </collector>

 </collectors>

</project>