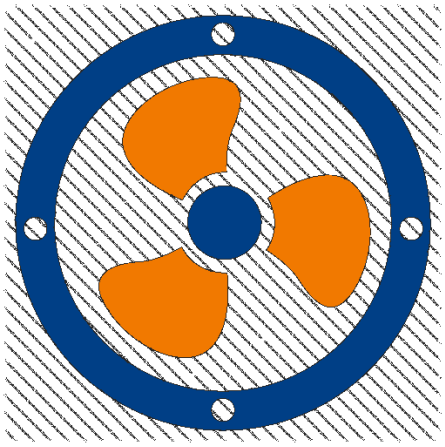


Data Exchange Component (DEC - 2025)
Software Reference Manual

October 12, 2024

Contents

1 Acronyms

Table 1.1: Acronyms

Acronym	**Text**
ADG	Auxiliary Data Gathering
ADP	Auxiliary Data Provider
API	Application Programming Interface
ARM	Advanced RISC Machines
ASTER	Advanced Spaceborne Thermal Emission & Reflection Radiometer
ASTGTM	ASTER Global Digital Elevation Model 1 arc second V3
ASTGTM_guide	ASTER Global Digital Elevation Model 1 arc second access guide
BULA	Bulletin A (IERS / quick EOP)
BULB	Bulletin B (IERS / EOP)
BULC	Bulletin C (IERS / leap-second)
CaaS	Container as a Service
CCSDS	Consultative Committee for Space Data Systems
CDM	Conjunction Data Message (SPCS)
CDM-CCSDS	Conjunction Data Message (CCSDS)
CELES	CelesTrak
CI/CD	Continuous Integration / Continuous Delivery
CLI	Command Line Interface
COAH	Copernicus Open Access Hub
COTS	Commercial Off The Shelf
CPOD	Copernicus Precise Orbit Determination
CPU	Central Processing Unit
CRD	Consolidated Ranging Data (ILRS)
CRD-V1	Consolidated Ranging Data format V1 (ILRS)
CSpOC	Combined Space Operations Center / former JSpOC (cf. SPCS)
CSV	Comma Separated Value
CVE	Common Vulnerabilities and Exposures
DE1	Deimos-1
DE2	Deimos-2
DEC	Data Exchange Component
DEM	Digital Elevation Model
DHUS	Data Hub Software
DIAS	Data and Information Access Services

ECDC	European Centre for Disease Prevention and Control
EDC	Eurolas Data Center
EO	Earth Observation
EOCFI	Earth Observation CFI Software
EOP	Earth Orientation Parameters
ERP	Earth Rotation Parameters
ESA	European Space Agency
ESA-FFS	ESA Earth Observation File Format Specification
F10.7	Solar Radio Flux index at 10.7 cm
FAQ	Frequently Asked Questions
FOSS	Free Open Source Software
FTP	File Transfer Protocol
FTPES	File Transfer Protocol Explicit TLS/SSL
FTPS	File Transfer Protocol SSL
GNSS	Global Navigation Satellite System
GSSC	GNSS Science Support Centre (ESA)
HGT	Height / SRTM file
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
I/F	Interface
IaaS	Infrastructure as a Service
ICD	Interface Control Document
IERS	International Earth Rotation Service
IERS-P	International Earth Rotation Service Paris Observatory
IGS	International GNSS Service
ILRS	International Laser Ranging Service
IoT	Internet of Things
ISO	International Standard Organisation
IVV	Integration Verification & Validation
JIT	Just In Time ruby compiler
JSON	Javascript Object Notation
LWM	Land Water Mask
MCS	Mission Control System
MCD12Q1	MODIS 6.0 Land Cover Quarterly Product
MOD09A1	MODIS 6.0 Land Surface Reflectance
MOD09A1_C61	MODIS 6.1 Land Surface Reflectance
MODIS	Moderate Resolution Imaging Spectroradiometer
MPS	Mission Planning System
MSI	Multi-Spectral Instrument (Sentinel-2)
NAOS	National Advanced Optical System
NASA	National Aeronautics and Space Administration
NASA-Ap	NASA space weather geomagnetic disturbance index
NASA-F10.7	NASA space weather solar radio 10.7 cm flux
NASA-CDDI-cURL	NASA CDDIS cURL login procedure
NASA-CDDIS	NASA Crustal Dynamics Data Information / Geodesy Data

NASA-LOGIN-CURL	NASA cURL login procedure
NASA-MSFC	NASA Marshall Space Flight Center / Earth Science Office
NATS	Neural Autonomic Transport System
NBULA	NASA Bulletin A
NBULC	NASA Bulletin C
NcML	NetCDF Markup Language
NetCDF	Network Common Data Form
NOAA	National Oceanic and Atmospheric Administration
OASIS	Organization for the Advancement of Structured Information Standards
OData	Open Data Protocol (OASIS)
OData-COAH	OData API for Copernicus DIAS
OpenSSL	Cryptography and SSL/TLS Toolkit
OS	Operative System
PRIP	Production Service Interface (delivery) Point (ESA Copernicus)
PSD	Product Specification Document
RFC	Request For Comments
RHEL	Red Hat Enterprise Linux
RPF	Reference Planning Facility
RSGA	Report and forecast of Solar and Geophysical Activity (NOAA)
RINEX	Receiver INdependent EXchange format
RINEX2	Receiver INdependent EXchange format V2
RINEX3	Receiver INdependent EXchange format V3
RINEX4	Receiver INdependent EXchange format V4
RVM	Ruby Version Manager
SATCAT	Satellite Catalogue
S1	Sentinel-1
S2	Sentinel-2
S3	Sentinel-3
S5P	Sentinel-5 Precursor
S1-PFS	Sentinel-1 Product Format Specification
S2-PFS	Sentinel-2 Product Format Specification
S3-LRR	Sentinel-3 Satellite Laser Retro Reflector
S3-SLR	Sentinel-3 Satellite Laser Ranging
S5P-PA	Sentinel-5P Products & Algorithms
SAP	Secure Access Point
SBOA	Sentinels Business Operations Analysis
SFL	Space-weather Forecast Long-term
SFS	Space-weather Forecast Short-term
SFTP	Secure File Transfer Protocol
SLR	Satellite Laser Ranging
SMTP	Simple Mail Transfer Protocol
SPCS	18th Space Control Squadron (space-track.org)
SPCS-API	API by space-track.org
SQL	Structured Query Language

SRTM	Shuttle Radar Topography Mission
SRTM_UG	SRTM User Guide
SSA	Space Situational Awareness
SSH	Secure Shell
SSL	Secure Socket Layer
SW	Software
TAI	Temps Atomique International
TCP	Transfer Control Protocol
TLE	Two Line Elements (Celestrak)
TLE-SPCS	Two Line Elements (SPCS)
TLS	Transport Layer Security
UAP	Unsecure Access Point
URI	Universal Resource Identifier
URL	Universal Resource Locator
UTC	Universal Time Coordinated
WebDAV	Web Distributed Authoring & Versioning
XML	Extensible Markup Language

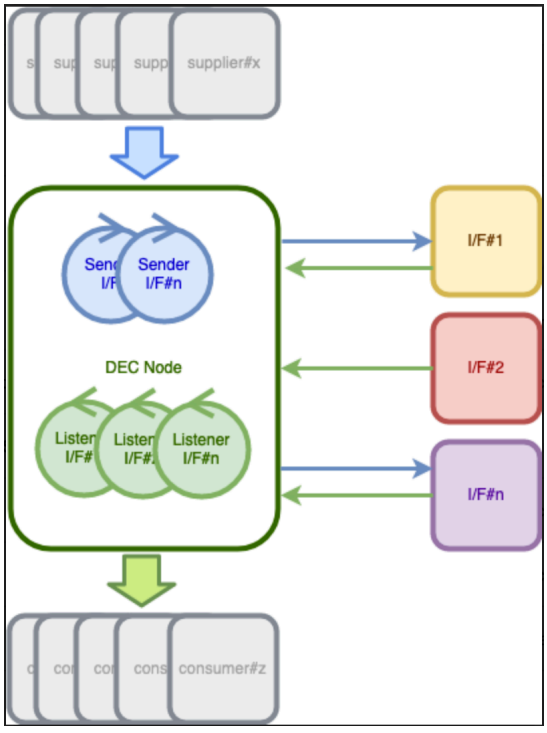
2 Introduction

2.1 Purpose & Scope

The Data Exchange Component (DEC) is a SW component to gather, transform, circulate, archive and publish files autonomously among different *interfaces* and *consumers*.

The scope of DEC SW usually lies on the different ICD defined for communication and exchange of data (generally *files*). As such, it relies on different COTS to delegate the implementation of various network protocols supported.

The DEC SW offers a command line interface to *pull* and *push* files towards the different configured interfaces.



2.2 Definitions

This section addresses a high level overview for the main design drivers that allows DEC to interface efficiently for the configured exchanges:

- *Interface* : this term beyond the general meaning is usually used to refer to a SW configuration item with such name which includes all the information to perform the circulations (e.g. hostname, credentials, protocol, etc).
- *Circulation* : this term generally refers to any exchange of files either for retrieval upon *pull* from or for distribution upon *push* to any configured interface.
- *Dissemination* : this term refers to the file-system local distribution into the Intray(s) of the files previously downloaded from a given interface as part of a *pull* circulation.
- *Download* : this term is used to refer to the actual retrieval of file from a given interface during a *pull* iteration.
- *Intray* : the final destination directory for downloaded files upon complete dissemination according to the rules.
- *Pull* : this term refers to the circulation operation for filtering and listing up to the actual download of files from a given interface according to the configured rules.
- *Push* : this term refers to the circulation operation to upload the available file(s) in some interface.
- *Rule* : this generic term refers to the configuration items for *pull* files, *disseminate* them, and *push* them.

2.3 Main Features

This section enumerates a high level overview for the main features that allows DEC to interface efficiently for the configured exchanges:

- *Automation* : the ability to perform unattended autonomous circulation operations.
- *Flexibility* : the ability to *pull* & *push* files from configurable interfaces and configurable circulation rules
- *Interface-Isolation* : every Interface is handled by separated isolated process avoiding any error propagation.
- *Robustness* : this is to perform “atomic” operations during file circulation ; the state for each operation is always known being network errors tolerant.
- *Performance* : parallelism of the circulation to fruit the available network bandwidth, support of file compression mechanisms to reduce the footprint of transfers and local dissemination for which duplication of files by *cp* or *mv* can be avoided by usage of *hardlinks*
- *Resiliency* : to *autonomously* recover from network errors, downtime and eventual glitches and resume operations during every iteration.

- *Zero-copy* : to avoid new copies in the file system during *dissemination* circulations.
- *Interfaces configuration management* : flexible CI/CD chain to create dedicated SW installation kit to carry specific desired interfaces configuration ready for operation.
- *Comprehensive traceability* : complete deterministic and unambiguous traceability along the entire workflow using codes within the log file of every operation.

3 New Installation

3.1 Install OS Dependencies

DEC SW is compatible with any POSIX OS. The information contained in this section aims to ensure that the ruby interpreter can be built at the target platform ; this approach aims for active obsolescence management allowing to make usage of recent stable versions avoiding lock-down by the OS distribution packages, which are usually very constraining in this respect. The SW packages are the very few which are requiring an OS administrator profile (i.e. root user) whilst DEC SW is recommended to live in the OS user environment which will execute it.

3.1.1 Debian / Ubuntu based distributions

The packages required for Linux distributions based on Debian package manager are listed below:

```
$> apt-get install curl
$> apt-get install libcurl4-openssl-dev
$> apt-get install git
$> apt-get install jq
$> apt-get install libssl-dev
$> apt-get install sshpass
$> apt-get install libsqlite3-dev
$> apt-get install sqlite3
$> apt-get install libpq-dev
$> apt-get install libxml2-utils
$> apt-get install ncftp
$> apt-get install p7zip-full
```

In order to generate the documentation in pdf, please install the following package: `$> apt-get install texlive-xe`

3.1.2 Red-Hat based distributions

The installation of packages with RHEL based distributions are constrained by the repositories configured ; official RHEL lack yum support for some of the above SW packages listed in previous section for Debian / Ubuntu based distributions.

The packages required for Linux distributions based on Red-Hat based distributions are listed below:

```
$> dnf groupinstall "Development Tools"
$> yum -y install perl-IPC-Cmd
$> yum install curl
$> yum install curl-devel
$> apt-get install sshpass
$> dnf install sqlite
$> apt-get install sqlite-devel
$> yum install -y openssl-devel
$> yum install -y libxml2
```

The following dependencies can be installed manually using rpm available at Nexus:

```
ncftp-3.2.5-18.el8.x86_64.rpm
p7zip-16.02-20.el7.x86_64.rpm
sshpass-1.09-4.el8.x86_64.rpm
```

```
$> yum install -y libxml2
```

3.1.3 macOS

The installation of SW dependencies in macOS can be resolved in different ways ; the tools which are not natively available with the OS or XCode development environment are installed using the package manager ”brew” ; openssl is however shipped with macOS but its underlying library libressl does not support some of the algorithms used, therefore it is required to install it with brew and ensure its execution prevails over the native version.

```
$> brew install jq
$> brew install ncftp
$> brew install openssl
$> brew install p7zip
```

3.2 Install Ruby

This section covers the install of *Ruby* interpreter and the *gem* dependencies needed by DEC SW. It is recommended that the *Ruby* interpreter is done locally for the Linux user who shall execute the DEC SW.

The *Ruby* interpreter can be obtained and installed in many different ways ; this manual describes how to install it local to the OS user which shall execute the DEC SW. Different runtime managers are addressed in the following sections to ease the installation and eventual futures updates in a seamless manner.

3.2.1 asdf

This section addresses how to install ruby based on asdf, which is a command-line tool run-time manager for different languages and development stacks, including ruby.

Download the tool:

```
$> git clone https://github.com/asdf-vm/asdf.git ~/.asdf --branch v0.14.1
```

Add the following entry to $HOME/.bash_profile

```
. $HOME/.asdf/asdf.sh
```

Then install / update the asdf ruby plug-in:

```
$> asdf plugin add ruby
```

```
$> asdf plugin update ruby
```

Now it is possible to select the ruby run-time version and easily install it for usage:

```
$> asdf list all ruby
```

```
1.8.5-p52
1.8.5-p113
1.8.5-p114
1.8.5-p115
1.8.5-p231
1.8.6
(...)
3.0.0-dev
3.0.0-preview1
3.0.0-preview2
3.0.0-rc1
3.0.0
3.0.1
3.0.2
3.0.3
3.1.0-dev
3.1.0-preview1
3.1.0
3.1.1
3.1.2
3.1.3
3.1.4
3.1.5
3.1.6
3.2.0-dev
(...)
```

Install the ruby run times specifying the version of choice, first to be downloaded and then to select it for local execution:

```
$> asdf install ruby 3.1.6
```

```
$> asdf local ruby 3.1.6; asdf reshim ruby
```

Now it is usually required to restart the shell upon installation to point the new ruby interpreter run-time.

3.2.2 RVM

This section addresses how to install ruby based on RVM, which is a command-line tool which allows you to easily install, manage, and work with multiple ruby environments from interpreters to sets of gems.

```
$> \curl -sSL https://get.rvm.io | bash -s stable --ruby
```

Alternatively to the latest ruby stable installation, the following command installs a specific version of ruby interpreter ; the version specified below corresponds to the one used for the unit tests and it is required as a mandatory precondition at installation time according to the gem file definition.

Different versions can be locally installed and select the one of choice at any time. However the following ruby version is recommended:

```
$> rvm install ruby-3.0
```

```
$> rvm --default use 3.0
```

3.3 Install DEC SW

In order to install DEC SW and the gems required, execute the following commands in the shell:

```
$> gem install aux_latest.gem
```

```
$> gem install dec_install_gemfile.gem
```

It is usually recommended to perform the installation of every DEC node with a dedicated installer, which already carries the desired configuration according to the defined interfaces and the desired behaviour ; this approach makes installations and configuration into the target environment almost instantaneously (e.g. maintenance in Operations). Every DEC node configuration can be kept under configuration control in order to build the dedicated installers.

The DEC installer naming file is generated to avoid ambiguity and bring information regarding the node they apply to. Below there are some few examples to illustrate the naming conventions to easily identify the installation kit for a given DEC node configuration.

```
$> gem install dec-1.0.32_s2_dec@s2boa-cloudferro.gem
```

```
$> gem install dec-1.0.33_s2_push_lisboa@e2espm-inputhub.gem
```

```
$> gem install dec-1.0.33_unit_tests@localhost.gem
```

Also the DEC installer can be customised to carry or avoid specific SW items, such as the testing tools (i.e. unit tests, interface tests, etc). The installation kit referred below as example has been customised for some Sentinel-2 project to carry the test tools, the OData tools and make usage of postgresql to persist the operations performed

```
$> gem install dec-1.0.33_s2_test_pg_odata@localhost.gem
```

As a very quick summary, users of the SW are encouraged to delegate the creation of the configuration by providing the requirements / interface documents to build dedicated installation packages for every node.

3.4 Uninstall DEC SW

In order to uninstall DEC SW, execute the following command in the shell:

```
$> gem uninstall dec
```

```
Remove executables:
decValidateConfig, decCheckConfig, decCheckSent, decConfigInterface2DB, decDeliverFiles, decGetFi

in addition to the gem? [Yn]
```

Press 'Y' key to remove the executables as well

3.5 FOSS Required

This section enumerates the FOSS which are used by DEC SW for exchange of file by some network protocol implementation, or file transformations associated to those exchanges.
This manual, which currently address component level information, does not address how to provision these COTS ; they can be obtained naturally with most OS distribution, downloaded with its native package manager, or manually installed. However it is noted that DEC containerized execution environments definition (IaaS) which already resolve every SW COTS dependencies out of the box are available, please do not hesitate in requesting information.

3.5.1 Databases

This section enumerates the different databases which can be used by DEC SW. Only *sqlite3* is *mandatory* to allow the execution of the entire set of *unit tests*. Below the different databases that have used at some deployment. It is recalled that it is possible to execute the DEC SW without any database by usage of flag *"–nodb"*.

- *sqlite3* : is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine
- *PostgreSQL* : object-relational database system with a strong reputation for reliability, feature robustness, and performance
- *MySQL* : a high performance, scalable database management system

3.5.2 Network Tools

This section enumerates the network tools which are used by DEC SW.

- *ncftp* : application programs implementing the File Transfer Protocol (FTP)
- *sftp* : application programs implementing the Secure File Transfer Protocol (SFTP)
- *sshpass*: application to handle the SSH password in non interactive mode (SFTP)
- *curl* : command line tool and library for transferring data with URLs (WebDAV)

3.5.3 File Compression Tools

This section enumerates the file compression tools which can be used by DEC SW.

- *7-zip* : is a file archiver with a high compression ratio ; name of the package can be *"p7zip"*
- *zip / unzip* : provide free, portable, high-quality versions of the *Zip* and *UnZip* compressor-archiver utilities
- *gzip* : The *gzip* reduces the size of the named files using Lempel–Ziv coding (LZ77)
- *compress* : The *compress* utility reduces the size of the named files by using adaptive Lempel-Ziv coding algorithm

3.5.4 File Transformation Tools

This section enumerates the file transformation tools which can be used by DEC SW.

- *xmllint* : is a command line XML parser which is part of the *libxml2* and libxml2-utils packages.
- *jq* : jq is a lightweight and flexible command-line JSON processor.

3.5.5 Encryption Tools

DEC SW makes usage of *openssl* toolkit to support the encryption of sensible configuration items. The version installed of openssl needs to support the algorithm PBKDF2.

3.6 Installation Verification

3.6.1 Verification with Unit Tests

This section describes how to execute the *unit tests*, which have been designed to be transparent and harmless in front of the potential different execution environments (i.e. development, integration, production), being their execution a simple and effective manner to verify the correct installation of the DEC SW.

The prerequisites to be able to successfully execute the *unit tests* are :

- an OS user *dectest*

- SFTP server running on *localhost* allowing login to *dectest* using the SSH keys
- FTP server running on *localhost* allowing login to *dectest* using the *password dectest*

```
$> decUnitTests
```

The results of the unit test should show no failures neither errors ; the execution time in a 2.66 GHz Intel Core 2 Duo is about 10 minutes approximately.

```
.
Finished in 590.39725 seconds.
--------------------------------------------------------------------------------
17 tests, 112 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notificati
100% passed
--------------------------------------------------------------------------------
0.03 tests/s, 0.19 assertions/s
```

3.6.2 Verification with Operational Interface

This section describes how to verify the correct installation of DEC SW by execution of a test with the IERS *operational* service, for which Internet connectivity is required for the FTP protocol. Note that it is not possible to ensure the connectivity availability by such service and sometimes test may fail by reply of *530 connect failed: Address already in use. No response from server.*

```
$> decUnitTests_IERS
```

```
.
Finished in 54.38868 seconds.
--------------------------------------------------------------------------------
2 tests, 25 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notification
100% passed
--------------------------------------------------------------------------------
0.02 tests/s, 0.33 assertions/s
```

In case of deployment without any database, the tests can be restricted to the ones which make usage of the *"–nodb"* execution option:

```
$> decUnitTests_IERS -n test_decGetFromInterface_NODB
```

```
.
Finished in 54.38868 seconds.
--------------------------------------------------------------------------------
1 tests, 18 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notification
100% passed
--------------------------------------------------------------------------------
0.02 tests/s, 0.23 assertions/s
```

3.7 DEC App Container

It is not strictly in the scope of this reference manual to describe the different integration and deployment strategies to make usage of the DEC SW. However it is considered valuable to document that the SW is ready for containerization (i.e. docker containers) and that such deployment strategy has been successfully used to design full standalone nodes including also standard TCP/IP servers to hub all the interface connections. In this respect there are available some dockerfiles IaaS to define the node SW dependencies in different execution environments (i.e. IVV, Production).

The files to be pushed or the files pulled to be made available to the end-peers are usually exchanged in the host which runs the container. Hence the execution of the DEC App container needs some mount point in the host ; this aspect is managed seamlessly by the different SW handling containers execution (i.e. docker, kubernetes, podman).

The DEC SW can be managed from the command line interface as it were natively installed in the host. As such the command reference of this manual can be used by just pointing to the DEC container by some alias definition invoking the execution for which the execution arguments are naturally appended:

```
alias decCheckConfig='podman exec dec decCheckConfig'
alias decConfigInterface2DB='podman exec dec decConfigInterface2DB'
alias decDeliverFiles='podman exec dec decDeliverFiles'
alias decGetFromInterface='podman exec dec decGetFromInterface'
alias decListDirUpload='podman exec dec decListDirUpload'
alias decListener='podman exec dec decListener'
alias decManageDB='podman exec dec decManageDB'
alias decNATS='podman exec dec decNATS'
alias decSend2Interface='podman exec dec decSend2Interface'
alias decStart='podman run --userns keep-id --env '\''USER'\'' --add-host=nl2-s-aut-srv-01:172.23
alias decStats='podman exec dec decStats'
alias decTestInterface_CelesTrak='podman exec -i dec decTestInterface_CelesTrak'
alias decTestInterface_NAOS_IVV-0500='podman exec -i dec decTestInterface_NAOS_IVV-0500'
alias decTestInterface_NASA='podman exec -i dec decTestInterface_NASA'
alias decValidateConfig='podman exec dec decValidateConfig'
```

4 Configuration

4.1 General

This section covers the general configuration of a DEC node which is defined in *dec_config.xml*.

4.1.1 Workflow

This section overviews the configuration items associated to the file-system directories used to prepare the circulations, temporal working directory or the directory in which DEC reports are placed.

- *SourceDir* : this is the directory from which the files are fetched to be circulated in *push* mode towards the configured interfaces according to the defined rules
- *GlobalOutbox* : this directory is owned by the DEC and it is a transient place to filter-out files which do not meet any circulation rule
- *ReportDir* : this directory is where the synthetic reports generated by DEC are placed upon circulation operations
- *TempDir* : this is the temporal working directory used by DEC

4.1.2 Inventory

This section describes the configuration items associated to the *Inventory* to record every performed circulation and to allow filtering of previously pulled files.

```
<Inventory>
   <Database_Adapter>sqlite3</Database_Adapter>
   <Database_Name>/tmp/dec_inventory</Database_Name>
   <Database_User>root</Database_User>
   <Database_Password>2mysql</Database_Password>
</Inventory>
```

The configuration items associated are the following below:

- *Database_Adapter* : this is the database used for recording the operations *unless* execution is performed with the *–nodb* execution flag is used ; values can be *"postgresql"*, *"sqlite3"*, etc driven by your database of choice

- *Database_Name* : this is a database name in which the circulation operations are recorded
- *Database_User* : this is the database user name
- *Database_Password* : this is the database user password

```
<Workflow>
   <SourceDir>/tmp/dec_delivery_root</SourceDir>
   <GlobalOutbox>/tmp/dec/outbox</GlobalOutbox>
   <ReportDir>/tmp/dec/in_basket_if_localhost_notsecure</ReportDir>
   <TempDir>/tmp/dec/tmp</TempDir>
</Workflow>
```

4.1.3 Circulation Reports

This section describes how to configure the DEC SW to generate the *circulation reports* containing information of the files pulled, pushed or which are available published by a given interface.

- *RetrievedFiles* : this report enumerates the files *pulled* from a given interface during a polling loop
- *DeliveredFiles* : this report enumerates the files *pushed* from a given interface during a polling loop
- *UnknownFiles* : this report enumerates the files available at the DownloadDir from a given interface which did not meet any rule of the *DownloadRules* during a polling loop iteration
- *EmergencyDeliveryFiles* : this report is specific of the RPF tailoring

The generated circulation reports are placed in directory location defined by the full path specified in the *ReportDir* configuration item:

```
<ReportDir>/dec/dec_reports</ReportDir>
```

4.1.3.1 Pulled Files

The DEC SW can generate a report enumerating the files which have been *pulled* from a given interface during a polling loop. The name of this *circulation report* is *"RetrievedFiles"*. Below there are the associated configuration items:

```
<Report Name="RetrievedFiles">
   <Enabled>true</Enabled>
   <Desc>List of Files Retrieved</Desc>
   <FileClass>OPER</FileClass>
   <FileType>DEC_R_PULL</FileType>
</Report>
```

4.1.3.2 Unknown Files

The DEC SW can generate a report enumerating the files which did not match any filtering rule for their retrieval from a given interface during a polling loop. The name of this *circulation report* is *"UnknownFiles"*. Below there are the associated configuration items:

```
<Report Name="UnknownFiles">
   <Enabled>true</Enabled>
   <Desc>List of unknown Files present</Desc>
   <FileClass>OPER</FileClass>
   <FileType>DECUNKNOWN</FileType>
</Report>
```

4.2 Interfaces

This section covers the *interfaces* configuration of a DEC node. The configuration of every interface is defined in the file *dec_interfaces.xml*.

- *TXRXParams* : configuration items regarding adjustments for the *pull* and *push* circulations
- *Enabled4Receiving* : *boolean* flag for the activation of the circulations in *push* mode
- *ImmediateRetries* : number of retries to be applied to recover eventual *upload* failures when *pushing* a file
- *LoopRetries* : number of loops to be applied to recover eventual *upload* failures when *pushing* a file
- *LoopDelay* : delay in *seconds* between two consecutive loops (if needed) for push
- *PollingInterval* : delay in *seconds* in between consecutive iterations for *pull* circulations
- *PollingSize* : maximum number of files to be *pulled* during an iteration
- *ParallelDownload* : number of simultaneous file downloads for pull circulations

4.2.1 TXRXParams

The *TXRXParams* configuration item defines the parameters which rule the circulation in pull and push mode.

- *Enabled4Sending* : *boolean* flag for the activation of the circulations in *pull* mode
- *Enabled4Receiving* : *boolean* flag for the activation of the circulations in *push* mode
- *ImmediateRetries* : number of retries to be applied to recover eventual *upload* failures when *pushing* a file
- *LoopRetries* : number of loops to be applied to recover eventual *upload* failures when *pushing* a file
- *LoopDelay* : delay in *seconds* between two consecutive loops (if needed) for push
- *PollingInterval* : delay in *seconds* in between consecutive iterations for *pull* circulations
- *PollingSize* : maximum number of files to be *pulled* during an iteration
- *ParallelDownload* : number of simultaneous file downloads for pull circulations

4.2.2 Server

The *Server* configuration item defines the parameters which rule the network protocol of choice selected to rule the file circulations either in *pull* and *push* mode.

- *Protocol* : unit tests are covering today *"FTP"*, *"FTPS"*, *"SFTP"*, *"WEBDAV"* - http and http(s) -, and *"LOCAL"* ; if configuration item is set to *"LOCAL"* then network configuration items such as *Hostname*, *Port*, *User* and *Pass* are not used
- *Hostname* : hostname of the interface
- *Port* : TCP port
- *User* : registered user for the server at the *Hostname* & *Protocol* server ; ; the value can optionally be kept encrypted using OpenSSL:

```
<User encrypted='true'>uiutw90k</User>
```

- *Pass* : password for authentication ; in case of secure *SecureFlag* set to true and the protocol *SFTP*,if this configuration item is empty, then the SSH keys associated to the OS user are used instead ; the value of this configuration item can optionally be kept encrypted using OpenSSL:

```
<Pass encrypted='true'>uiutw90OFb+JMH9Lk5OogQ==</Pass>
```

- *RegisterContentFlag* : *boolean* flag to activate tracking of the files available for *pulling* without effective download
- *RetrieveContentFlag* :*boolean* flag to activate the download of the files available for *pull circulation*
- *SecureFlag* : *boolean* flag to activate *encrypted* SSH communications
- *VerifyPeerSSL* : *boolean* flag to authenticate the server identity when using TLS/SSL communications
- *CompressFlag* : *boolean* flag to activate *compression* of the *encrypted* SSH communications
- *DeleteFlag* : *boolean* flag to delete the file upon successful circulation
- *PassiveFlag* : *boolean* flag to activate *the FTP passive mode* of the FTP data connections
- *CleanUpFreq* : defined in seconds used by the clean-up daemon if enabled to clean-up previously circulated files in *push mode*

4.2.3 DeliverByMailTo

List of email *Address* configuration items used to *push* files using SMTP.

4.2.4 Notify

The *Notify* configuration item serves to activate the generation and delivery of an email receipt carrying the name of the files which have been *pushed* towards a given *Interface*.

- *SendNotification* : *boolean* flag to activate the
- *To* : list of email *Address* configuration items which will receive the notification

```
<Notify>
        <SendNotification>true</SendNotification>
<To>
        <Address>mario.bros@gmail.com</Address>
        <Address>mario.draghi@gmail.com</Address>
        <Address>mario.cipollini@esa.int</Address>
</To>
</Notify>
```

4.2.5 Events

The *Events* configuration item host the activation and shell command associated to the event when circulating towards the interface.

```
<Events>
<Event Name="OnReceiveNewFilesOK"       executeCmd="echo :-)"/>
<Event Name="OnReceiveError"            executeCmd="echo :-("/>
<Event Name="NewFile2Intray"            executeCmd="echo %F"/>
</Events>
```

Below the possible *Event Name* values which can be configured:

- *OnSendOK* : it is raised upon a successful loop to push
- *OnSendNewFilesOK* : it is raised upon a successful *push* loop of file(s)
- *OnSendERROR* : it is raised if any file failed the *push* circulation
- *OnReceiveOK* : it is raised upon a successful loop for pull
- *OnReceiveNewFilesOK* : it is raised upon a successful *pull* loop of file(s)
- *OnReceiveNewFile* : it is raised every new file *pulled*
- *OnReceiveERROR* : it is raised if any *pull* of file has failed
- *OnTrackOK* : it is raised upon a successful loop to check availability pull
- *NewFile2Intray* : it is raised upon every new file locally disseminated after pulling it ; %f specifies the *filename* disseminated ; %F specifies the *full path filename* disseminated ; %d specifies the directory name in which the file has been disseminated

4.3 Pull circulations

The mechanism to *select*, *download* and locally *disseminate* files to a DEC node is referred to *pull mode*.
The configuration of the pull circulation rules is defined in *dec_incoming_files.xml*.

4.3.1 Interface pick-up points

The *Interface* configuration item defines the interfaces pick-up point(s) from which the exposed files are polled.

- *Name* : this is the configuration item *identifier* which is used to identify a given interface and refer to it in other sections of the configuration
- *LocalInbox* : this is the *local* directory in which files are initially placed upon successful download ; if no dissemination rule is applied, this directory becomes the final destination
- *DownloadDirs* : a list of *Directory* configuration items to define the *remote* directories of this interface from which the files will be pulled ; the attribute *DepthSearch* defines the directory sub-levels which are polled.
- *Switches* : *pull* configuration items to modulate the operations with respect the published files by the interface.

```
<ListInterfaces xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<Interface>
        <Name>LOCALHOST_SECURE</Name>
        <LocalInbox>/tmp/in_basket_if_localhost_secure</LocalInbox>
        <DownloadDirs>
                <Directory DepthSearch="0">/download1</Directory>
                <Directory DepthSearch="0">/download2</Directory>
        </DownloadDirs>
    <Switches>
           <DeleteDownloaded>true</DeleteDownloaded>
           <DeleteDuplicated>true</DeleteDuplicated>
           <DeleteUnknown>false</DeleteUnknown>
           <LogDuplicated>true</LogDuplicated>
           <LogUnknown>true</LogUnknown>
   </Switches>
</Interface>

<Interface>
        <Name>IERS</Name>
        <LocalInbox>/tmp/dec/if_iers_in_basket</LocalInbox>
        <DownloadDirs>
                <Directory DepthSearch="0">ser7</Directory>
        </DownloadDirs>
    <Switches>
```

```
        <DeleteDownloaded>false</DeleteDownloaded>
        <DeleteDuplicated>false</DeleteDuplicated>
        <DeleteUnknown>false</DeleteUnknown>
        <LogDuplicated>false</LogDuplicated>
        <LogUnknown>false</LogUnknown>
    </Switches>
</Interface>

</ListInterfaces>
```

4.3.1.1 Pull Switches

This sections describes the different pull switches which are defined interface-wise when retrieving files from it.

- *DeleteDownloaded* : file pulled are removed upon successful retrieval ; this option is usually disabled when fetching files from shared public interfaces.
- *DeleteDuplicated* : *duplicated* files found during a pull iteration are removed from the interface without retrieval ; this option is usually disabled when fetching files from shared public interfaces.
- *DeleteUnknown* : *unknown* files which do not meet any *download rule* are removed from the interface without retrieval ; this option is usually disabled when fetching files from shared public interfaces.
- *LogDuplicated* : *duplicated* files found from the interface during a pull iteration are logged as *[DEC_301]* warnings without retrieval ; this option is usually disabled when fetching files from shared public interfaces.
- *LogUnknown* : *unknown* files from the interface which do not meet any *download rule* are logged as *[DEC_320]* warnings ; this option is usually disabled when fetching files from shared public interfaces.

4.3.2 Download rules

This configuration item defines the filters to select the files to be pulled from the interface pick-up point(s).

- *File Type* attribute : it defines the file to be pulled, allowing the definition of wildcards
- *Description* : free text to capture a human friendly definition of the interface
- *FromList* : a list of *Interface* identifiers from which the file will be pulled

```
<DownloadRules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<File Type="tai-utc*">
        <Description>TAI UTC correlation</Description>
        <FromList>
                <Interface>IERS</Interface>
```

```
                <Interface>LOCALHOST_NOT_SECURE</Interface>
        </FromList>
</File>

<File Type="finals.a*">
        <Description>TAI UTC correlation consolidated</Description>
        <FromList>
                <Interface>IERS</Interface>
                <Interface>LOCALHOST_NOT_SECURE</Interface>
        </FromList>
</File>

<File Type="S2?_*">
        <Description>Sentinel-2 Ground Segment</Description>
        <FromList>
                <Interface>LOCALHOST_SECURE</Interface>
                <Interface>LOCALHOST_NOT_SECURE</Interface>
        </FromList>
</File>

</DownloadRules>
```

4.3.3 Dissemination rules

This configuration item defines the local Intrays and the dissemination rules to be applied upon successful pull of files.
Firstly it is defined the *ListIntrays Intray* configuration items:

- *Name* : this is the Intray *identifier* which is used to identify the final location the files pulled are placed
- *Directory* : this is the *local* directory associated to the in-tray
- *Execute* : this *optional* configuration item defines a command to be executed upon every file disseminated into the Intray as an *event*

```
<ListIntrays>

<Intray>
        <Name>S2ALL</Name>
        <Directory>/local_dissemination/S2ALL</Directory>
</Intray>

<Intray>
        <Name>S2A</Name>
        <Directory>/local/S2A</Directory>
        <Compress>7z</Compress>
</Intray>

<Intray>
```

```
        <Name>S2B</Name>
        <Directory>/local/S2B</Directory>
</Intray>

<Intray>
        <Name>GPS</Name>
        <Directory>/local/GPS</Directory>
</Intray>

</ListIntrays>
```

Then the rules for local dissemination of the pulled files into the in-tray(s) are defined. Only the first rule which matches the file name or wildcard defined in the *File Type* attribute is applied ; as such it is recommended to sort the rules ranking from the more restrictive ones.

- *Name* : this is the intray *identifier* which is used to identify the final location the files pulled are placed
- *HardLink* : this is a *boolean* flag to hardlink in the file-system a pulled file in case of definition of different in-trays for multiple dissemination into
- *ToList* : this is the list of *Intray* identifiers configuration item the defined *File Type* is *disseminated* into

```
<ListFilesDisseminated>

<File Type="tai-utc*">
        <HardLink>False</HardLink>
        <ToList>
                <Intray>GPS</Intray>
        </ToList>
</File>

<File Type="finals*">
        <HardLink>false</HardLink>
        <ToList>
                <Intray>GPS</Intray>
        </ToList>
</File>

<File Type="S2A_*.*">
        <HardLink>true</HardLink>
        <ToList>
                <Intray>S2A</Intray>
                <Intray>S2ALL</Intray>
        </ToList>
</File>

<File Type="S2B_*.*">
        <HardLink>true</HardLink>
        <ToList>
```

```
                <Intray>S2B</Intray>
                <Intray>S2ALL</Intray>
        </ToList>
</File>

<File Type="S2__*">
        <HardLink>False</HardLink>
        <ToList>
                <Intray>S2ALL</Intray>
        </ToList>
</File>

</ListFilesDisseminated>
```

4.4 Push circulations

The mechanism to fetch and upload files from a DEC node is referred to *push mode*.

The configuration of the pull circulation rules is defined in dec_ outgoing_files.xml.

4.4.1 Interface pick-up point

The *Interface* configuration item defines the interfaces pick-up point(s) from which the files are pushed into.

- *Name* : this is the configuration item *identifier* which is used to identify a given interface and refer to it in other sections of the configuration
- *LocalOutbox* : this is the *local* directory in which files are initially placed prior their upload ; a sub-directory is created with the name of the protocol in lowercase to temporarily host the files
- *UploadDir* : a list of *Directory* configuration items to define the *remote* directories of this interface where the files will be pushed into
- *UploadTemp* : a list of *Directory* configuration items to define the *remote* directories of this interface where the files will be temporarily pushed into in which the interface consumer is expected to not poll

```
<ListInterfaces xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

     <Interface>
        <Name>LOCALHOST_NOT_SECURE</Name>
        <LocalOutbox>$DEC_TMP/dec/out_basket_if_localhost_notsecure</LocalOutbox>
        <UploadDir>/tmp/dec_simul_if/localhost_notsecure/upload</UploadDir>
        <UploadTemp>/tmp/dec_simul_if/localhost_notsecure/upload/_TEMP_</UploadTemp>
     </Interface>

      <Interface>
             <Name>LOCALHOST_FTP_PASSIVE</Name>
         <LocalOutbox>$DEC_TMP/dec/out_basket_if_localhost_ftp_passive</LocalOutbox>
         <UploadDir>/tmp/tmp/dir1/</UploadDir>
         <UploadTemp>/tmp/tmp/dir2/</UploadTemp>
     </Interface>

</ListInterfaces>
```

4.4.2 Upload rules

TBW

4.4.3 Push Circulations

This section describes the specific environment variables used by DEC SW for *push circulations*.

- *DEC_DELIVERY_ROOT* : this is the source directory of the files to be *pushed*

4.5 Database

This section describes how to initialise the database configuration for DEC driven by some environment variable settings described below. It is however recalled that it is possible to execute the DEC *without* a database by usage of the flag *"–nodb"* which precludes some features such as control flow, throughput computation however allowing essential file data circulations.

4.5.1 Creation

This section describes how to initialise the database by creation of the tables and index used by DEC SW.
In order to create the database, execute the following command in the shell:

```
$> decManageDB --create-tables
```

In order to remove the database, execute the following command in the shell:

```
$> decManageDB --drop-tables
```

4.5.2 Configuration of Interfaces

This section describes how to configure within the database the interface identifiers defined in the configuration file *dec_interfaces.xml* :

```
$> decConfigInterfaceDB --process EXTERNAL
```

It is also possible to register the interface identifiers according to its name definition in *dec_interfaces.xml*:

```
<Interface Name="IERS">
```

The following command can be used to register an interface identifier in the database:

```
$> decConfigInterfaceDB --add IERS
```

4.5.3 Pushed Files

TBW

4.6 Environment Variables

The execution of DEC SW can be parameterized through the usage of environment variables which are loaded from the shell. Usage of these environment variables is considered *deprecated* since all the associated configuration items they refer to are covered within the configuration files.

The eventual definition and availability of the environment variable referred in this section would override the configuration parameters defined in the configuration files, therefore they are documented to mainly avoid their usage.

4.6.1 Database Variables

This section describes the environment variables used by DEC SW to define the database settings.

- *DEC_DB_ADAPTER* : this is the database used for recording the operations unless execution is performed with the *–nodb* execution flag is used ; values can be *"postgresql"*, *"sqlite3"*, etc driven by your database of choice
- *DEC_DATABASE_NAME* : this is a database name in which the circulation operations and file availability tracking are recorded
- *DEC_DATABASE_USER* : this is the database user name
- *DEC_DATABASE_PASSWORD* : this is the database user password

4.6.2 General Variables

This section describes the environment variables used by DEC SW to define the database settings.

- *DEC_CONFIG* : this is the directory in which the DEC config files are present
- *DEC_TMP* : this is the directory used for temporal / transient operation prior final availability in the data sinks
- *HOSTNAME* : this environment variable defines the current hostname which is used for mail authentication purposes

4.6.3 File Transfer for RPF

This section describes the specific environment variables used by DEC SW acting as File Transfer for the RPF. The configuration items associated are not covered by the DEC configuration files, therefore their definition is mandatory for the RPF execution environment.

- *FTPROOT* : this is the directory which is used to place the files which will be *pushed* as part of each ROP delivery
- *RPF_ARCHIVE_ROOT* : this is RPF archive root directory from which the files to be transferred are gathered
- *RPFBIN* : this is RPF binary directory in which the RPF tools invoked by the File-Transfer are placed (i.e. *put_report.bin*, *removeSchema.bin*, *write2Log.bin*)

4.7 Configuration Verification

This section describes how to verify the DEC configuration settled according to the previous sections.

To validate the XML files of the DEC configuration, there is an associated a DEC command line tool called *decValidateConfig* which verifies the syntax and semantics associated to configuration items.

Execute the following command in the shell to verify the configuration:

```
$> decValidateConfig --all
```

4.7.1 Check Interfaces

In order to check the entire configuration defined and the connectivity to the different configured interfaces execute the following command:

```
$> decCheckConfig --all
```

If your DEC deployment and configuration does not record the operations into the database (cf. execution with *"–nodb"* flag), alternatively execute the following command in the shell to verify the configuration:

```
$> decCheckConfig --all --nodb
```

5 Network Protocols Supported

5.1 Network Protocols for Pull

5.1.1 Introduction

This section introduces to the different network protocols & associated verbs which are supported for pull circulations:

- **Local**
- **SFTP**
- **FTP**
- **FTPS**
- **HTTP**
- **WebDAV**
- **OData**

5.1.2 Local

The files from the *DownloadDirs* are pulled using the OS file management operations. The files *"downloaded"* into the *LocalInbox* is perfomed using the POSIX hard-link operation whenever both configuration items point to the same file-system, otherwise a copy is perfomed.

5.1.3 SFTP

Secure File Transfer Protocol over SSH is supported. Authentication supported modes are passwordless with SSH keys and using the password.

If no password is configured in the interface definition, the interface handler relies on the identity management SSH keys configuration to perform password-less authentication before triggering batch / non-interactive pull requests to the interface. In this respect configuration-wise, the SSH keys are not included in the SW installation package and need to be set.

5.1.4 FTP

The files from the *DownloadDirs* are pulled using the File Transfer Protocol. It supports both PORT and PASV connections for the TCP data stream which carries the files pulled. Transfers are enforced in BINARY mode (i.e. ASCII transfer mode is not used). The FTP commands RETR and optionally DELE are used respectively to perform the file retrieval and its deletion in the server.

5.1.5 FTPS

Plain FTP and TLS / SSL Actually for FTPS the DEC SW implements two different handlers

5.1.6 HTTP(S)

The protocol HTTP(S) is not strictly a file based one, however files can be published using it and DEC SW is able to pull them. The handler works in two different modes driven by the *DownloadDirs* configuration, either by retrieval of a fixed URL, either using the URL to search for files for retrieval.

5.1.6.1 Fixed URL

The handler retrieves with HTTP GET the URL, which is defined by the *DownloadDirs* in the "Directory" item prepended by the *Server* Server *Hostname* configuration item; some data providers use this strategy to publish up-to-date data at a fixed location (e.g. below):

```
<Directory DepthSearch="0">/iers/bul/bulc/Leap_Second.dat</Directory>
```

5.1.6.2 Directory Emulation

The protocol handler detects the trailing symbol slash in the Directory configuration item defined as part of the *DownloadDirs*. Then it processes the HTML received ; the logic of processing is arbitrary which is to retrieve all HTML *'href'* anchors which are then nominally filtered according to the DEC configuration for such interface (cf. *DownloadRules*). This handler depends on the interface implementation since this is *not* protocol independent as the WebDAV extensions. Hence this handler needs to be verified case by case. In general this is a dirty solution when WebDAV cannot be available on top of HTTP.

5.1.7 WebDAV

The protocol WebDAV is built on top of HTTP with some specific extensions which allows to search and find items meeting some criteria. The WebDAV protocol handler uses the verb PROPFIND to obtain the items published for every Directory in *DownloadDirs* and then those are filtered according to the *DownloadRules*. The final retrieval of the selected items according to configuration is performed by the HTTP handler using the verb GET.

5.1.8 OData

The protocol OData is supported ; in particular for DIAS interfaces, which make available an API to query and obtain metadata or / and finally download the selected items driven by the previous queries. A dedicated client The protocol handler detects the trailing symbol slash in the Directory configuration item defined as part of the *decODataClient* is devoted for this kind of interfaces to perform selection by sensing time or availability time of the data.

5.1.9 NATS

5.1.9.1 Request-Reply

NATS request-reply pattern is supported ; DEC performs NATS request according to some API configuration and it processes the received reply.

5.1.9.2 Subscribe

NATS publish-subscribe pattern is supported ; DEC subscribes to some subjects according to some API configuration to asynchronously receive the messages for their processing.

5.2 Network Protocols for Push

5.2.1 LOCAL

PUT

5.2.2 SFTP

PUT

5.2.3 FTP

PUT

5.2.4 HTTP

PUT

5.2.5 WebDAV

The WebDAV extension of HTTP, which is defined in RFC4918 allows to rename files so that DEC can make them visible just upon complete upload.
Initially the file is uploaded into the *UploadTemp* using the HTTP verb PUT. Upon its successful upload the file is *moved* into the final *UploadDir* using the verb MOVE of the WebDAV extension.

6 Workflows Reference

6.1 Description

This section brings some typical workflow reference common to most use cases of the software.

6.2 Configuration workflow

This section comprises the typical actions to configure every DEC SW node. It assumes a correct installation of the DEC SW include the dependencies referred in this document. Configuration wise it is recommended to get an installation which already carries the definition of the workflows; however configuration can be changed in hot at any point following the information of this manual.
Every command verifies the execution environment to check whether the FOSS SW dependencies are available for usage, in case of missing any of them, the error message DEC_799 is raised. This explanation applies to every command and it is not duplicated in the document.

6.2.1 Validate Configuration

The DEC SW configuration items are spread out in different XML files. This steps aims to verify the syntax and semantic correctness of the such configuration as prerequisite of successful operations. This step is performed by decValidateConfig. This step is to be performed manually from the command line interface whenever is needed (e.g. new SW update carrying modified configuration).

6.2.1.1 Check XML Configuration

```
$> decValidateConfig -a
```

- DEC_002 : the configuration file verifies the settled XML schema
- DEC_798 : the configuration is not according to the settled XML schema

6.2.2 Create DEC Inventory

This step is required to persist and record the circulation operations in a database ; note that it is still possible to perform circulations without persistence of the operations by usage of the ”–nodb flag” by the different commands. Skip this step if the DEC SW was already previously

installed and it is a SW update.

Firstly the database tables are created using decManageDB ; then the interfaces defined in the XML configuration files are populated into the database using decConfigInterface2DB.

6.2.2.1 Drop DEC Inventory

```
$> decManageDB -d
```

6.2.2.2 Create DEC Inventory

```
$> decManageDB -c
```

DEC_000 : creation of the DEC DB / Inventory

6.2.2.3 Populate Interfaces into DEC Inventory

```
$> decConfigInterface2DB -p EXTERNAL
```

DEC_001 : Interface added into the DEC DB / Inventory

6.2.3 Check Interface Configuration

This step exploits the interface according to the configuration to check the availability of defined end-points (e.g. directories for file based protocols). It also covers the verification of the DEC/Inventory availability if used.

6.2.3.1 Check Interface Connectivity

```
$> decCheckConfig -e NASA_SFL
```

- DEC_003 : Interface is correctly declared in DEC/Inventory
- DEC_004 : Interface exchange point is reachable

6.3 Pull workflow

6.3.1 Manual Pull Workflow

This section covers the manual workflow by explicit CLI invoke to execute the pull workflow for a given interface.

6.3.1.1 List available data

```
$> decGetFromInterface -m CELESTRAK_SFS -l
```

- DEC_005 : Interface polling is started
- DEC_105 : File is available

- DEC_060 : Number of files available
- DEC_100 : Interface pull iteration completed

6.3.1.2 Pull available data

```
$> decGetFromInterface -m CELESTRAK_SFS
```

- DEC_005 : Interface polling is started
- DEC_060 : Number of files available
- DEC_110 : File is downloaded
- DEC_100 : Interface pull iteration completed

6.3.1.3 File conversion

```
$> auxConverter -m S3 -f /tmp/aux/20210514RSGA.txt -d /tmp/aux_out2
```

- AUX_001 : File has been converted

6.3.2 Automation Pull Workflow

This section cover the automation workflow to pull data from an interface. It leverages the mechanisms described in previous section regarding manual pull workflow.

6.3.2.1 Automation Listeners Start

```
$> decListener -a
```

DEC_006 : Starting automation listener for some interface

6.3.2.2 Automation Listeners Status

```
$> decListener -c
```

DEC_003 : Status of the automation listener for some interface

6.3.2.3 Automation Listeners Stop

```
$> decListener -S
```

- DEC_002 : Automation listener for some Interface is stopped
- DEC_603 : Automation listener for some Interface was not running

6.4 Push workflow

6.4.1 Retrieve from an archive

This step performed by command using decGetFiles4Transfer gathers every file which will be subject of circulation from SourceDir or a single directory source and places it into every interface LocalOutbox appending the sub-directory name according to the delivery protocol (i.e. ”sftp”, ”ftp”, etc).

6.4.1.1 Fetch Outgoing Files

```
$> decGetFiles4Transfer
```

- DEC_211 : File has been placed at the GlobalOutbox from the SourceDir
- DEC_212 : Fetched file has been removed in the SourceDir
- DEC_213 : File has been placed into the LocalOutbox out-tray.

7 Log Messages Reference

7.1 Information Messages

7.1.1 Scope

The information messages by DEC SW are meant in general for tracing the nominal performed circulations allowing consumers to monitor them.

7.1.2 Index of Messages

- [**DEC_000**] : DEC Database / Inventory $table created
- [**DEC_001**] : Interface added to DEC Inventory
- [**DEC_002**] : Valid schema check for configuration file
- [**DEC_003**] I/F ”interface”: Interface is correctly declared in the DEC Inventory
- [**DEC_004**] I/F ”interface”: Exchange point is reachable
- [**DEC_005**] I/F ”interface”: Polling Started
- [**DEC_006**] I/F ”interface”: Automation Listener Status Active
- [**DEC_007**] I/F ”interface”: Automation Listener Status Inactive
- [**DEC_050**] I/F ”interface”: Polling Completed / No file(s) available for pull
- [**DEC_060**] I/F ”interface”: Polling Completed / New file(s) available for pull
- [**DEC_100**] I/F ”interface”: Pull iteration completed for such interface
- [**DEC_105**] I/F ”interface”: ”filename” is available
- [**DEC_110**] I/F ”interface”: Downloaded ”filename” with size ”num_bytes” bytes
- [**DEC_111**] I/F ”interface”: Downloaded ”filename” with same md5 / duplication
- [**DEC_115**] Disseminated ”filename” into ”directory” Intray
- [**DEC_116**] Compressed ”filename” in ”method” at ”directory” Intray
- [**DEC_120**] I/F ”interface”: Deleting unknown file ”filename” available
- [**DEC_125**] I/F ”interface”: Deleting duplicated file ”filename” previously received

- [**DEC_126**] I/F "interface": Deleted downloaded file "filename"
- [**DEC_130**] I/F "interface": event "name" triggered
- [**DEC_131**] I/F "interface": event NewFile2Intray triggered
- [**DEC_135**] I/F "interface": "filename" pull report created
- [**DEC_210**] I/F "interface": File "filename" sent using "protocol"
- [**DEC_211**] File retrieved from SourceDir to the GlobalOutbox (prior its *push* circulation)
- [**DEC_212**] File removed from SourceDir upon DEC_211
- [**DEC_213**] File placed at the Interface LocalOutbox
- [**DEC_214**] File compressed at the interface LocalOutbox
- [**DEC_254**] Request of OData metadata to interface
- [**DEC_255**] Retry Request of OData URL
- [**DEC_257**] Metadata OData file created
- [**DEC_259**] Download request to an OData interface delivery point
- [**DEC_260**] Successful download from an OData interface delivery point
- [**AUX_001**] File conversion from source file source is performed
- [**NATS001**] I/F "interface": NATS Request performed
- [**NATS002**] I/F "interface": NATS Request is successful
- [**NATS003**] I/F "interface": NATS Subscribe to some queue is performed

7.1.3 [DEC_000] DEC Database / Inventory $table created

This message is generated by *decManageDB* when creating the DEC Inventory database and the associated tables which are necessary to be able to persist the records of the performed circulations.

```
[DEC_000] DEC Database / Inventory table interfaces created
[DEC_000] DEC Database / Inventory table tracked_files created
[DEC_000] DEC Database / Inventory table received_files created
[DEC_000] DEC Database / Inventory table sent_files created
```

7.1.4 [DEC_001] Interface added into DEC Inventory

This message is generated by *decConfigInterface2DB* when adding a new interface into the DEC Inventory database to be able to persist the records of the performed circulations.

7.1.5 [DEC_002] Node XML configuration is valid

This message is generated by *decValidateConfig* tool when checking the DEC configuration XML syntax and semantics across the different configuration files ; one message is generated for each configuration file.

```
[DEC_002] Valid schema check for dec_config
[DEC_002] Valid schema check for dec_interfaces
[DEC_002] Valid schema check for dec_incoming_files
[DEC_002] Valid schema check for dec_outgoing_files
[DEC_002] Valid schema check for ft_mail_config
[DEC_002] Valid schema check for dec_log_config
```

7.1.6 [DEC_003] Interface Correctly Declared in the Inventory

This message is generated by *decCheckConfig* tool when checking the availability of the configured interfaces *decCheckConfig* tool when checking the availability of the configured interfaces.

7.1.7 [DEC_004] Interface Exchange Point Reachable

This message is generated by *decCheckConfig* tool when checking the availability of the configured interfaces.

7.1.8 [DEC_005] Interface Polling Start

This message time tags the start of the polling operation to *pull* files from a given interface. The time to poll and filter the file to be pulled depends on the number of files published, the different rules configured for download or / and verification with the database of previous downloads ; measurements can be done with this message and its conclusion which is flagged by *[DEC_050]* or *[DEC_060]* message.

7.1.9 [DEC_006] Interface Automation Listener Active

This message time the automation listener status *Active* to *pull* files from a given interface.

7.1.10 [DEC_007] Interface Automation Listener Inactive

This message time the automation listener status *Inactive* to *pull* files from a given interface.

7.1.11 [DEC_050] Interface Polling Complete / No File(s) Available

This message is logged upon completion filtering the files published in the download interface and verification of eventual duplications in the database without any *new* file meeting the download rules. It can be used to measure the time required to perform the filtering operations.

7.1.12 [DEC_060] Interface Polling Complete / New File(s) Available

This message is logged upon completion filtering the files published in the download interface and verification of eventual duplications in the database reporting there are some *new* files for download. It can be used to measure the time required to perform the filtering operations.

7.1.13 [DEC_100] Interface Pull Iteration Completed

This message is logged upon the completion of the pull circulation loop.

7.1.14 [DEC_105] File Available

The file referenced in the message met any of the filtering rules defined in the configuration *dec_incoming_files.xml* and it is therefore available for download from the interface. This message is logged only when the *–list* mode is used, otherwise the files matching the rules are directly downloaded and logged accordingly.

7.1.15 [DEC_110] File Downloaded

The file referenced in the message has been downloaded from the interface and placed into the *LocalInbox* associated to such interface. If the same file is available again at the same interface, the DEC will detect a file duplication condition.

7.1.16 [DEC_111] File Downloaded with same MD5

The downloaded file from the interface referenced in the message has the same md5 with respect to the previous retrieval and it is handled as a file duplication condition.

7.1.17 [DEC_115] File Disseminated

The file referenced in the message met any of the and it has been moved from *LocalInbox* into the Intray(s) matching the first *dissemination rule*.

7.1.18 [DEC_116] File Compressed

The file referenced in the message has been compressed according to the *dissemination rule Compress* optional configuration item of the *Intray* entity.

7.1.19 [DEC_117] File Uncompressed

The file referenced in the message has been compressed according to the *dissemination rule Compress* optional configuration item of the *Intray* entity.

7.1.20 [DEC_120] Unknown File Deleted

According to the configuration item *DeleteUnknownFiles* defined in *dec_config.xml* configuration, a file available in the *pull* interface, which did not match any filtering rule defined in the configuration *dec_incoming_files.xml* has been deleted from the *DownloadDirs Directory* of such interface it was present.

7.1.21 [DEC_125] Duplicated File Deleted

According to the configuration item *DeleteDuplicatedFiles* defined in *dec_config.xml* configuration, a file available in the *pull* interface, which has been previously downloaded successfully such interface is considered duplicated and it is removed directly in download directory without downloading it.

7.1.22 [DEC_126] File Deleted

According to the configuration item *DeleteDownloaded* defined in *dec_incoming_files.xml* for every interface, the file which has been successfully *pulled* is deleted subsequently. This message can also be logged when deleting duplicated files.

7.1.23 [DEC_130] Event Triggered

The message traces some event which has been triggered for a given interface according to the configuration items defined in Events.

7.1.23.1 Event newfile2intray

This event is triggered upon successful polling completion (cf. DEC_060) which drove dissemination of some files (cf. DEC_115).

```
[DEC_130] LEAP I/F : event newfile2intray => auxConverter -f /S3/Leap_Second.dat
```

7.1.23.2 Event onreceivenewfilesok

This event is triggered upon successful polling completion (cf. DEC_060) which drove download of some files (cf. DEC_110).

```
[DEC_130] LEAP I/F : event onreceivenewfilesok triggered => echo %f RECEIVED
```

7.1.24 [DEC_131] Dissemination Command Execution

The message traces some event which has been triggered for a given interface according to the configuration item *"Execution"* defined in the *dissemination rules*.

7.1.25 [DEC_135] Report Generation of Files Pulled

The message traces the generation of the DEC report containing the files retrieved during a pull iteration from a given interface.

7.1.26 [DEC_144] Pull Statistics

This message is generated by decStats to supply some statistics regarding the file *pull* operations during the last "n" hours.
The internal of log messages is a JSON string referring to:

- **filename** : the name of the file pushed
- **size** : size of the file in Bytes
- **protocol** : protocol the file was pushed
- **date** : date of the circulation

7.1.27 [DEC_200] Push iteration to Interface completed successfully

This message traces the successful push circulation completion.

7.1.28 [DEC_203] Request to Push File to Interface started

This message traces the start of a given file push circulation towards an interface.

7.1.29 [DEC_205] Push Retry to Interface waiting LoopDelay

Push loop which has failed any circulation (cf. DEC_710) is waiting *LoopDelay* seconds to retry.

7.1.30 [DEC_206] Push Immediate Retry to Re-sending File to Interface waiting LoopDelay

Push loop which has failed any circulation (cf. DEC_710) is waiting *LoopDelay* seconds to retry.

7.1.31 [DEC_210] File Pushed to Interface

The file referenced in the message has been uploaded to the interface placed into its *UploadDir*.

7.1.32 [DEC_211] Push Setup / File at GlobalOutbox

The file has been placed in the GlobalOutbox for push circulations.

7.1.33 [DEC_212] Push Setup / File at Removed from SourceDir

The file has been removed from the SourceDir upon its previous availability at the GlobalOutbox for push circulation.

7.1.34 [DEC_213] Push Setup / File at Interface LocalOutbox

The file has been placed in the Interface LocalOutbox for push circulations.

7.1.35 [DEC_214] Push Setup / File Compressed at Interface LocalOutbox

The file has been compressed in the Interface LocalOutbox location for push circulations.

7.1.36 [DEC_235] Report Generation DeliveredFiles to Interface

This message traces the generation of the report which refers to the files pushed towards the configured interfaces.

7.1.37 [DEC_240] Push Email Notification to Interface

TBW / decNotify2Interface.

7.1.38 [DEC_241] Push Error Email Notification to Interface

TBW / decNotify2Interface.

7.1.39 [DEC_244] Push Statistics

This message is generated by decStats to supply some statistics regarding the file *push* operations during the last "n" hours.
The internal of the log message is a JSON string referring to:

- **filename** : the name of the file pushed
- **size** : size of the file in Bytes
- **protocol** : protocol the file was pushed
- **date** : date of the circulation


```
{"filename":"tai-utc.dat","size":61,"interface":"LOCAL","protocol":"FTP;","date":"2020-04-15 17:03:19 UTC"}
{"filename":"mario.jpg","size":32478,"interface":"LOCAL","protocol":"FTP;","date":"2020-04-15 17:03:19 UTC"}
```


7.1.40 [DEC_254] Request OData Metadata to Interface

Request of OData Metadata to an interface.

7.1.41 [DEC_255] Retry Request OData URL

Retry to request OData URL.

7.1.42 [DEC_257] OData Metadata File Created

Query reply from an OData service dumped to file.

7.1.43 [DEC_259] OData Download Request

Request to download using HTTP a product from an OData service.

7.1.44 [DEC_260] OData Download Successful

Product download successful from an OData service.

7.1.45 [AUX_001] Auxiliary Data Conversion

This message contains trace information regarding the file conversions from some source.

7.1.46 [NATS001] NATS Request Perfomed

This message contains trace information regarding the NATS request performed to an interface: server, subject and body.

7.1.47 [NATS002] NATS Request Successful

This message contains trace information regarding the NATS request reply.

7.1.48 [NATS003] NATS Subscribe Performed

This message contains trace information regarding the NATS subscribe message.

7.2 Warning Messages

7.2.1 Scope

The warning messages by DEC SW are meant in general for tracing not nominal situations which might eventually require some intervention.

7.2.2 Index of Messages

- **[DEC_301]** Detected duplicated file ”filename” already received from ”interface” I/F
- **[DEC_320]** Detected unknown file ”filename” available at ”interface” I/F
- **[DEC_330]** ”filename” is stuck in ”temporal” directory
- **[DEC_331]** ”filename” is stuck in LocalInbox directory
- **[DEC_401]** ”filename” was previously sent
- **[DEC_555]** ”filename” was previously Sent

7.2.3 [DEC_301] Duplicated File Available

The file reported is available again at the *pull* interface ; if the inventory is active, DEC retrieves just once a given file from each interface. This message will be reported during every polling iteration unless the *duplicated* file is removed from the interface.

7.2.4 [DEC_320] Unknown File Available

The file reported is available in the *pull* interface, but did not match any filtering rule defined in the configuration *dec_incoming_files.xml*. This message will be repeated every polling iteration unless the *unknown* file is removed from the interface.

7.2.5 [DEC_330] File Stuck in Temporal Directory

This warning is associated to error message *[DEC_620]* informing that the file was successfully downloaded and stuck in the temporal directory available for contingency recovery.

7.2.6 [DEC_340] Email notification is disabled for the Interface

This warning is obtained when triggering *decNotify2Interface* for an interface whose configuration has such notifications disabled (cf. flag *SendNotification*).

7.2.7 [DEC_401] File Previously Uploaded

The file reported has been found available again for upload towards an interface though it has been previously *pushed* there ; since the *push* iteration is executed with the ”once” flag, this file has been discarded for upload.

7.2.8 [DEC_501] Interface already present in DEC Inventory

This warning message is generated by *decConfigInterface2DB* when requesting addition of an interface which is already present in the DEC Inventory database.

7.2.9 [DEC_555] File Disseminated Exists

This message is raised when a previous file with same filename is found disseminating locally into an in-tray.

7.2.10 [DEC_556] Old File Removed

This message is raised upon detection of a previous file in the final intray to record its deletion prior the dissemination of the new one.

7.3 Error Messages

7.3.1 Scope

The error messages by DEC SW are meant in general for tracing problems encountered during one iteration which prevented the complete termination of any circulation rule.

7.3.2 Index of Messages

- [**DEC_600**] I/F ”interface”: Could not perform polling
- [**DEC_604**] I/F ”interface”: Exchange Point Not Reachable
- [**DEC_605**] I/F ”interface”: Pull Miss-Configuration
- [**DEC_610**] I/F ”interface”: Unable to connect to ”Hostname”
- [**DEC_611**] I/F ”interface”: reason of the connection failure
- [**DEC_612**] I/F ”interface”: Cannot reach directory defined in any of the the ”Download-Dirs”
- [**DEC_613**] I/F ”interface”: server message associated to the failure associated to the change of directory
- [**DEC_614**] I/F ”interface”: Cannot GET ”URL”
- [**DEC_615**] I/F ”interface”: Failed to get list of files / FTP passive mode is ”mode”
- [**DEC_620**] I/F ”interface”: Could not copy ”file” into local ”directory”
- [**DEC_625**] Dissemination failure of ”file” into intray ”directory” using mv command
- [**DEC_626**] Dissemination failure of ”file” into intray ”directory” using hard-links command
- [**DEC_627**] Compression / Uncompression failure of ”file” into intray ”directory” using hard-links command
- [**DEC_628**] Failure when executing command
- [**DEC_666**] I/F ”interface”: Could not download ”file”
- [**DEC_667**] OData I/F ”interface”: Could not download ”file”
- [**DEC_705**] I/F ”interface”: Push miss-configuration
- [**DEC_711**] Could not reach directory from *SourceDir* to fetch files for push
- [**DEC_713**] Could not place file in *LocalOutbox* to prepare files out-tray for push
- [**DEC_798**] Configuration problem
- [**DEC_799**] Integration problem
- [**AUX_600**] Cannot read specified file
- [**AUX_666**] Input file is not recognized

7.3.3 [DEC_600] Failed to Pull from Interface

This message is raised at the completion of the interface polling operation (i.e. *pull* circulation)when it failed. Reason should have been previously logged (e.g. DEC_610, DEC_611,etc). Polling elapse time can be computed by referring to the associated DEC_005 message.

7.3.4 [DEC_601] Failed to Start Listener for Interface

This message is raised when the interface commanded for *pull* circulation is miss-configured. It is usually explained by DEC_605 message.

7.3.5 [DEC_603] Listener for Interface was not running

This message is raised when the interface commanded to stop or re-start its listener was not previously switched-on.

7.3.6 [DEC_604] Interface Exchange Point Not Reachable

This message is generated by *decCheckConfig* tool when checking the availability of the configured interfaces. This message is the counter side of DEC_004 tracing that the check tool could not reach the exchange point, which may be related to miss-configuration or network unavailability.

7.3.7 [DEC_605] Pull Miss-Configuration

This message is raised when interface commanded for *pull* circulation is miss-configured.

7.3.7.1 Not existing interface in dec_interfaces.xml

```
[DEC_605] I/F Foo: such is not a configured interface
```

7.3.7.2 Inventory db is not configured / reachable

```
[DEC_605] I/F Foo: db inventory is not configured / reachable
```

7.3.7.3 Not registered interface in database

Polling is performed using the database but such interface has not been registered.

```
[DEC_605] I/F Foo: such is not a registered I/F in db
```

7.3.7.4 Pull flag is disabled in dec_interfaces.xml

```
[DEC_605] I/F IERS: Enabled4Receiving flag is disabled
```

7.3.7.5 I/F LocalInbox not defined in dec_incoming_files.xml

```
[DEC_605] I/F Foo: LocalInbox configuration item not found
```

7.3.8 [DEC_610] Unable to Connect to server

This error is raised when it is not possible to connect to ”Hostname” using the ”Protocol”.

7.3.9 [DEC_611] Pull Server Error Message

This error code can bring different messages associated to the connection failure [DEC_610] is reporting. Errors can driven by miss-configuration of the Server associated to such interface or its unavailability due to network problems.

7.3.9.1 Hostname is unknown

```
[DEC_611] I/F LOCALHOST_NOT_SECURE: getaddrinfo: nodename nor servname provided, or not known
```

7.3.9.2 Service not enabled by host

```
[DEC_611] I/F LOCALHOST_NOT_SECURE: Connection refused - connect(2) for 127.0.0.1:21
```

7.3.9.3 SSL/TLS not configured by server

```
[DEC_611] I/F LOCALHOST_FTPS: 500 This security scheme is not implemented
```

7.3.9.4 Failed authentication

The authentication failure message differs depending on the network protocol used for pull.

```
[DEC_611] I/F LOCALHOST_NOT_SECURE: 530 Login authentication failed
```

7.3.10 [DEC_612] Cannot Reach Interface Download Directory

Failed to reach one of the *Directory* of the ”DownloadDirs” of a given interface.

7.3.11 [DEC_613] Server Error Message

The information replied by the server when error condition [DEC_612] arises is usually meaningless to understand the root reason to limit the risks of security attacks. It usually is related to file permissions on the server or miss-configuration problems referring to a non existing directory.

```
[DEC_613] LOCALHOST I/F: 550 Can't change directory to /a/if/local/: No such file or directory
```

7.3.12 [DEC_614] Cannot Get URL

Failed to get the URL associated to one *Directory* of the ”DownloadDirs” of a given interface when using the HTTP protocol.

7.3.13 [DEC_615] Failed to list files with FTP

In practical terms, this is a failure when performing the FTP command list to obtain the list of files available in a given directory.

7.3.14 [DEC_620] Downloaded File not Copied into Final Dir

The file retrieved from the interface could not be copied into the final directory defined in the configuration. This is usually a problem of file permissions.

7.3.15 [DEC_625] File Move Dissemination Failure

This error is raised when DEC could not disseminate a file into a given *Intray* according to the *Dissemination rules* using a mv command. This is usually associated to file permission problems of the *Intray* directory.

7.3.16 [DEC_626] File Hard-Link Dissemination Failure

This error is raised when DEC could not disseminate a file into a given *Intray* according to the *Dissemination rules* using hard-links. This is usually associated to file permission problems of the *Intray* directory.

7.3.17 [DEC_627] File Dissemination Compress / Uncompress Failure

This error is raised when DEC could not compress / uncompress a file into a given *Intray* according to the *Dissemination rules*.

7.3.18 [DEC_628] Command Execution Failure

This error is raised when DEC tried to execute a command in sub-shell obtaining failure status.

7.3.19 [DEC_666] Download Failure

This error is raised when DEC could not download a file from a given interface.

7.3.20 [DEC_667] OData URI Download Failure

This error is raised when DEC could not download a URI file from some given OData interface delivery point.

7.3.21 [DEC_668] OData Metadata Download Failure

This error is raised when DEC could not download the OData metadata from some given OData interface delivery point.

7.3.22 [DEC_670] Deletion Failure

This error is raised when DEC could not delete from the interface a file which has been successfully downloaded (cf. *[DEC_110]*). For a given file from a specific interface this message is exclusive to *[DEC_126]*.

7.3.23 [DEC_700] Push Failure to Interface

This message is raised at the completion of the interface *push* operation, when it failed, upon completion of the retry loops (cf. TXRXParams configuration). Reason should have been previously logged (e.g. DEC_710, DEC_711,DEC_712).

7.3.24 [DEC_703] Push Iteration to Interface is already running

This message is raised upon execution of a new instance of decSend2Interface for a given interface when there is already a previous on-going execution for such given interface.

7.3.25 [DEC_705] Push Miss-Configuration for Interface

This message is raised when interface commanded for *push* circulation is miss-configured.

7.3.25.1 Not existing interface in dec_interfaces.xml

```
[DEC_705] Foo I/F: such is not a configured interface
```

7.3.25.2 Push flag is disabled in dec_interfaces.xml

```
[DEC_705] IERS I/F: Enabled4Sending flag is disabled
```

7.3.26 [DEC_710] Failed to send file to Interface

This message is raised when failure to *push* a file towards an interface.

7.3.27 [DEC_711] Cannot Reach SourceDir Sub-Directory

This error is raised when DEC could not reach a *SourceDir* sub-directory to fetch the selected files to *push* according to some *[Upload Rule* miss-configuration.

7.3.28 [DEC_712] Cannot Reach Interface Upload Directory

Failed to reach *UploadDir* for a given interface.

7.3.29 [DEC_713] File Hard-link Failure into LocalOutbox

This error is raised when hard-linking from the source directory to the local outbox.

7.3.30 [DEC_714] File Read Error in LocalOutbox

This error is raised when a file previously detected in the LocalOutbox for push is not found when reading it to extract its size. This is usually an integration problem of the data-flow and the component workflow.

7.3.31 [DEC_715] Push Server Error Message

Server error message associated to the *push* failure.

7.3.31.1 No Write Permission

```
[DEC_715] I/F LOCALHOST_FTPS: 553 Could not create file
```

7.3.32 [DEC_740] Failed to Email Push Success Notification to Interface

This message is raised when *[decNotify2Interface* failed to email the notification to inform about a *success push* iteration.

7.3.33 [DEC_741] Failed to Email Push Error Notification to Interface

This message is raised when *[decNotify2Interface* failed to email the notification to inform about a *failed push* iteration.

7.3.34 [DEC_742] Email Server Error

This message is the (SMTP) server error when sending notification emails.

7.3.34.1 Credentials Not Accepted

```
[ERROR] 2020-04-14 18:44:38 NODE_1.mail - [DEC_742] LOCALHOST_NOT_SECURE I/F: Username and Passwo
```

7.3.34.2 SMTP Server Not Known

```
[ERROR] 2020-04-14 18:49:42 NODE_1.mail - [DEC_742] LOCALHOST_NOT_SECURE I/F: getaddrinfo: nodena
```

7.3.34.3 Time-Out / Port Miss-configuration

```
[ERROR] 2020-04-14 18:56:31 NODE_1.mail - [DEC_742] LOCALHOST_NOT_SECURE I/F: execution expired
```

7.3.34.4 Need TLS / Secure Flag Miss-configuration

```
[ERROR] 2020-04-14 18:58:58 NODE_1.mail - [DEC_742] LOCALHOST_NOT_SECURE I/F: Must issue a STARTT
```

7.3.35 [DEC_750] Event Error

This message is logged when the command executed associated to some *Event* raised.

```
[ERROR] [DEC_750] I/F NOAA: EventManager failed execution of auxConverter / No such file or direc
```

7.3.36 [DEC_798] Configuration Error

This message is logged when the XML schema validation corresponding to a given configuration file is failing.

```
[DEBUG] NODE_1.checker - xmllint --schema /Users/mario/.frum/versions/3.1.0/lib/ruby/gems/3.1.0/g
[DEC_798] Fail schema check for dec_interfaces
```

7.3.37 [DEC_799] Integration Error

This message is logged when some unexpected error raised.

7.3.38 [AUX_600] Cannot access File

This message is logged when the file to be converted can not be reached from the specified path.

7.3.39 [AUX_666] No File conversion pattern found

This message is logged when the file to be converted does not meet any file converted pattern (i.e. the SW does not recognize the type of the input file).

7.3.39.1 Missing SW Dependency

Some SW dependency required by DEC was not found in $PATH. Refer to *section FOSS* to check the SW dependencies which DEC needs for its correct execution.

```
[DEC_799] Fatal Error : 7za not present in $PATH
[DEC_799] Fatal Error : jq not present in $PATH
[DEC_799] Fatal Error : xmllint not present in $PATH
[DEC_799] Fatal Error : sqlite3 not present in $PATH
[DEC_799] Fatal Error : curl not present in $PATH
[DEC_799] Fatal Error : ncftpget not present in $PATH
[DEC_799] Fatal Error : ncftpput not present in $PATH
[DEC_799] Fatal Error : sftp not present in $PATH
```

7.3.39.2 Database table not existing or reachable

This is an error when not finding the some DEC/Inventory table ; likely because it has not been created yet.

```
[DEC_799] Fatal Error : DEC/Inventory Could not find table 'interfaces'
```

7.3.39.3 Zero length file download

If an interfaces publishes a Zero length file, this can be retrieved though no data content is really dumped into the file-system (e.g. FTP protocol replies successful 226). This is an anomaly of the source interface.

7.3.39.4 Compress method for push not supported

This is a configuration error regarding the definition of the compression mechanism prior push operations.

```
[DEC_799] Fatal Error : compress method #{compress} not supported / check dec_outgoing_files.xml
```

7.3.39.5 Unexpected error when launching the listener for pull

This is an unexpected error when launching the listener for pull operations.

```
[DEC_799] I/F #{entity}: Error when launching Listener
```

7.3.39.6 File or Directory not existing or reachable

This is an error when trying to reach one file or directory.

```
[DEC_799] I/F #{entity}: No such file or directory @ dir_s_chdir - /volume1/dec/interfaces/dhus_c
```

7.4 Debug Messages

7.4.1 Scope

The debug messages by DEC SW are meant in general for troubleshooting purposes when undesired behaviour raises ; for instance it may be useful to activate it when rehearsing new configurations.

By default the log debug messages are not generated ; they can be optionally activated at the execution time of the different DEC commands by usage of the *–Debug* or *-D* execution flag supplied at the command line interface.

7.4.2 Index of Messages

- [**DEC_910**] I/F ”interface”: Filtering directory with files ”number” of items
- [**DEC_911**] I/F ”interface”: performance when filtering according to download rules
- [**DEC_912**] I/F ”interface”: performance when filtering according to database
- [**DEC_913**] I/F ”interface”: filtering file with database
- [**DEC_914**] I/F ”interface”: ”file” found in database
- [**DEC_915**] I/F ”interface”: ”file” not found in database
- [**DEC_941**] File dissemination command I
- [**DEC_942**] File dissemination command II
- [**DEC_951**] Removing ”file” in ”directory” which is ”LocalInbox”

7.4.3 [DEC_910] Filtering Number of Files in Download Directory

This message reports the number of files published in the ”DownloadDirs” which are about to be filtered according to the download rules and the database content.

7.4.4 [DEC_911] Filtering According to Download Rules

This message reports the CPU time resources used to perform the filtering operations.

7.4.5 [DEC_912] Filtering Number of Items with Database

This message traces the number of files which met the download rules during a given pull iteration and which are about to be filtered with respect to the database registry.

7.4.6 [DEC_913] Filtering File with Database

This message traces that a file found in the interface, which is matching the download rules, is about to be verified whether it has been previously received or not by checking its name reference in the database.

7.4.7 [DEC_914] File Found in Database

This message traces that a file found in the interface, which is matching the download rules, is already registered in the database referring to a previous successful download, and therefore such file will be filtered-out and it will not be pulled again.

7.4.8 [DEC_915] File Previously Not Recorded in Database

This message traces that a file found in the interface, which is matching the download rules, is not previously registered in the database, and therefore such file will be requested for download as part of the *pull* iteration.

7.4.9 [DEC_941] File Dissemination Command I

This message traces the file-system command used to perform the pulled file *first* dissemination from the Interface *"LocalInbox"* into the first *Intray* configured according to the *Dissemination Rules*.

7.4.10 [DEC_942] File Dissemination Command II

This message traces the file-system command used to perform the pulled file *remaining* dissemination(s) from the first *Intray* configured according to the *Dissemination Rules*.

7.4.11 [DEC_951] Removal from LocalInbox upon Dissemination

This message reports that the "file" has been removed from the "directory" "LocalInbox" upon a complete dissemination according to the configured rules.

8 Commands Reference

8.1 Command Line Interface

This section provides a brief description of the command line interface which is common to every DEC SW executable.

The command exit codes have been defined in order to chain them at will so it is possible to understand their execution correctness. In this respect the following exit codes are available:

- *exit 0* : the command was successfully executed .
- *exit 66* : the command was invoked with wrong parameters and it did not perform its workflow.
- *exit 99* : this command was executed and raised any problem when performing its workflow.

8.2 decValidateConfig

```
== Synopsis

This is a DEC command line tool that checks the validity of DEC configuration
files according to DEC's XSD schemas. This tool should be run everytime a
configuration change is performed.

-e flag:

With this option the Interfaces (Entities) configuration file (dec_interfaces.xml)
is validated using the schema interfaces.xsd

-g flag:

With the main DEC configuration file (dec_config.xml)
is validated using the schema dec_config.xsd

-i flag:

With this option the Incoming file-types configuration file (dec_incoming_files.xml)
is validated using the schema dec_incoming_files.xsd
```

```
-o flag:

With this option the Outgoing file-types configuration file (dec_outgoing_files.xml)
is validated using the schema dec_outgoing_files.xsd

-m flag:

With this option the DEC Mail configuration file (ft_mail_config.xml) is
validated using the schema ft_mail_config.xsd

-l flag:

With this option the DEC Logs configuration file (dec_log_config.xml) is
validated using the schema dec_log_config.xsd

-a flag:

This is the all flag, which performs all the checks described before.


== Usage
-a          Check all DEC configuration files
-g          Check DEC's general configuration file dec_config.xml
-e          Check the Entities Configuration file dec_interfaces.xml
-m          Check the mail configuration file ft_mail_config.xml
-i          Check the incoming file-types configuration file dec_incoming_files.xml
-i          Check the outgoing file-types configuration file dec_outgoing_files.xml
-X <dir>    eXtract the configuration into the specified directory
-N <label>  label of the node to be appended into the configuration
-h          shows this help
-v          shows version number
```

8.3 decCheckConfig

```
== Synopsis

This is a command line tool that checks the coherency of the DEC configuration.
DEC configuration is distributed amongst different XML files. The information set up
must be coherent. This tool ensures that all configuration critical elements are correct.
(All DEC config files must be placed in the $DEC_CONFIG directory).
So, run this tool everytime a configuration change is performed.

-e flag:

With this option the Interfaces (Entities) configuration placed in dec_interfaces.xml
is checked. As well it is checked the coherency between the dec_interfaces.xml
```

configuration file and the DEC Inventory (DEC Database).
(Note: if the network link to a given I/F is broken, the tool will not be able to connect and it will report a configuration error of this I/F).

-i flag:

With this option the Incoming file-types registered in the dec_incoming_files.xml are checked. Mainly what it is done is to check that the interfaces from a File is retrieved are configured in the dec_interfaces.xml file.

-m flag:

With this option the DEC Mail configuration placed in the ft_mail_config.xml is checked.

-s flag:

With this option the DCC Services configured in the dcc_services.xml file are checked. The check performed with this flag is that the executable set in the command of the service can be found in the $PATH environment variable.

-t flag:

With this option the In-Trays configured in the dec_incoming_files.xml file are checked.

-a flag:

This is the all flag, which performs all the checks described before.

== Usage

```
decCheckConfig [--nodb]
-a    checks all DEC configuration
-e    checks entities configuration in dec_interfaces.xml
--nodb no Inventory checks
-i    checks incoming file-types configured in dec_incoming_files.xml
-o    checks outgoing file-types configured in dec_outgoing_files.xml
-m    checks the mail configuration placed in ft_mail_config.xml
-t    checks the In-Trays configuration placed in dec_incoming_files.xml
-l    checks the log configuration
-h    it shows the help of the tool
-u    it shows the usage of the tool
-v    it shows the version number
-V    it performs the execution in Verbose mode
-D    it performs the execution in Debug mode
```

8.4 decCheckSent

```
== Synopsis

This is a DEC command line tool that lists the contents of the <Upload> directory
corresponding to a given interface configured in interfaces.xml

== Usage
decCheckSent -m <Interface_Name> [-t]
--mnemonic  <MNEMONIC> (mnemonic is case sensitive)
--temp      it shows the content of the <UploadTemp> directory
--Show      it shows all available I/Fs registered in the Inventory
--help      shows this help
--usage     shows the usage
--Debug     shows Debug info during the execution
--version   shows version number
```

8.5 decConfigInterface2DB

```
== Synopsis

This is a Data Exchange Component command line tool that synchronizes the Entities configuration
with DEC Inventory. It extracts all the I/Fs from the dec_interfaces.xml file and
inserts them in the DEC Inventory.

As well it allows to specify a new I/F mnemonic to be loaded into the DEC Inventory with
the "--add" command line option.

== Usage
decConfigInterfaceDB --add <MNEMONIC> | --process EXTERNAL
--add <MNEMONIC>     (mnemonic is case sensitive) add the specified Entity
--process EXTERNAL   process $DEC_CONFIG/dec_interfaces.xml
--Show               it shows all I/Fs already loaded in the DCC Inventory
--Verbose            execution in verbose mode
--version            shows version number
--help      shows this help
--usage     shows the usage
```

8.6 decDeliverFiles

8.7 decGetFiles4Transfer

```
== Synopsis

This is a DEC command line tool that retrieves files to be transferred (push)
from a source directory specified by $DEC_DELIVERY_ROOT environment variable.

Files present in the source directory are filtered according to the rules
defined in ft_outgoing_files.xml
=> wildcards such as  <File Type="S2A*">
=> children directories, such as <File Type="GIP_PROBA2">

Files gathered are finally placed into the directory defined in the
configuration <GlobalOutbox> present in dec_config.xml

-O flag:
The "ONCE" flag registers in the Inventory all the files sent. As well it checks
prior to the delivery whether a files has been previously sent or not to avoid
delivering it twice to the same Interface.

== Usage
decGetFiles4Transfer [-O] [-l]
--ONCE      The file is just sent once for each I/F
--list      list only
--help      shows this help
--usage     shows the usage
--Debug     shows Debug info during the execution
--version   shows version number
```

8.8 decGetFromInterface

```
decGetFromInterface

== Synopsis

This is a DEC command line tool that polls the I/Fs for retrieving
files of registered filetypes. As well It retrieves the I/F
exchange directory file content linked to a time-stamp.

-l flag:

With this option, only "List" of new availables files for Retrieving and Tracking is done.
This flag overrides configuration flags RegisterContentFlag RetrieveContentFlag in dec_interfaces
So Check ONLY of new Files is performed anyway.
```

```
-R flag:

With this option (Reporting), DEC Reports will be created (see dcc_config.xml).
Report files are initally placed in the Interface local inbox and
if configured in files2InTrays.xml disseminated as nominal retrieved file.

--del-unknown:

It overrides the dcc_config.xml configuration parameter DeleteUnknown and explicitly
commands for removal of unknown files not configured in ft_incoming_files.xml


== Usage
decGetFromInterface -m <MNEMONIC>  [-l] [--nodb]
--mnemonic  <MNEMONIC> (mnemonic is case sensitive)
--list      list only (not downloading and no ingestion)
--nodb      no Inventory recording
--no-intray skip step of delivery to intrays
--del-unknown it deletes remote files not configured in ft_incoming_files.xmls
--receipt   create only receipt file-list with the content available
--Report    create a Report when new files have been retrieved
--Show      it shows all available I/Fs registered in the DEC Inventory
--help      shows this help
--usage     shows the usage
--Debug     shows Debug info during the execution
--Unknown   shows Unknown files
--Benchmark shows Benchmark info during the execution
--version   shows version number
```

8.9 decODataClient

```
The decODataClient is a simple OData client to hit Copernicus Sentinel
servers to extract metadata and download the associated products.

DHUS queries:

Query by Sentinel-2 Datatake Identifier: which looks for the associated
published Complete Single Tile products

--query dhus:s1:S1B
--query dhus:s2:S2A
--query dhus:gnss:S1B
--query dhus:s2:GS2B_20200903T104429_018252_N02.14
--query dhus:s1:S1A:S1A_EW_GRDM
--query dhus:s1:S1A:45C1F
```

```
--query adgs:s2:S2B

--query dhus_s5p:s5:S5

PRIP: queries:

Query is so far targetting S2PRIP specifically.

--query prip:S2B_OPER_MSI_L0__DS_VGS2_20201028T163228_S20201028T144729_N02.09:MSI_L0__GR

The queries can be done also specifically by PDI:
decODataClient -u <user> -p <pass> -q prip:S2B_OPER_MSI_L0__DS_VGS1_20210126T095435_S20210126T080

== Usage
decODataClient
--user      <username>
--password  <password>
--time       2021-03-24T00:00:00.000 (DHuS format applied to the IngestionDate)
--delay     <hour delay> (decimal offset 0.5-0.25 is scaled to minutes)
--sensing   "2021-03-16T00:00:00.000,2021-03-20T00:00:00.000"
--creation  "2021-03-16T00:00:00.000,2021-03-20T00:00:00.000"
--H         <hours delay> (decimal offset 0.5-0.25 is scaled to minutes)
--Location  <full_path_dir>
--format    json | xml | csv
```

8.10 decListener

```
== Synopsis

This is a Data Exchange Component (DEC) command line tool
that manages the I/Fs listeners for data retrieval.

The DEC listeners automates the file pulling from the configured interface.
One listener is devoted for every interface configured.

The behaviour of the listener is driven by the settings defined
in the configuration file $DEC_CONFIG/dec_interfaces.xml

Alternatively the listener settings can be overriden with the command line options.

== Usage
decListener  --all [-R]| --mnemonic <MNEMONIC> --interval <seconds>
--all                 starts a listener for each I/Fs
--Reload              force a Restart of all listeners
--stop <MNEMONIC>     it stops of the listener for the given I/F
--Stop                it stops of all listeners
--check               it checks whether the listeners are running
--mnemonic <MNEMONIC> (mnemonic is case sensitive)
```

```
--interval              the frequency it is polled I/F given by MNEMONIC (in seconds)
--nodb                  no Inventory recording
--no-intray             skip step of delivery to intrays upon download
--help                  shows this help
--Debug                 shows Debug info during the execution
--version               shows version number
```

8.11 decManageDB

```
== Usage
decManageDB --create-tables | --drop-tables
--create-tables   create all minarc required tables
--drop-tables     drops all minarc tables
--rpf             selector to include reference planning tables
--Debug           shows Debug info during the execution
--help            shows this help
```

8.12 decNATS

```
== Synopsis

This is a command line tool that connects to NAOS MCS CCS5 using NATS, HTTP & SFTP.

> FSTATUS : Get the status of the CCS5 session
decNATS -n NAOS_IVV_MCS_NATS -FSTATUS
> FSTART  : Start a CCS5 session
decNATS -n NAOS_IVV_MCS_NATS -FSTART
> FSTOP   : Stop a CCS5 session
decNATS -n NAOS_IVV_MCS_NATS -FSTOP


F0. Get Session

F1. Manage G/S connections
CCS5.AutoPilot.NAOS.call, message: SleGroundStationMgm SVALS0 CONNECT

F2. Session Switch
NATS subject used by AUTO to issue the request: CCS5.AutoPilot.NAOS.switch
NATS body: none.

F3. Ingest MPS activities file
NATS subject used by AUTO to issue the request: CCS5.AutoPilot.NAOS.call
```

```
NATS body: Ingest <input URL> <target path>
<input URL>: path to local file (e.g. NFS mounted) or FTP URL for the file to be ingested, in the
<target path>: destination path for the ingested path (absolute or relative directory/file path).
Note that if the destination path ends with / (slash), it is treated as a directory, and the inge
Note that when using a relative path, the destination path will be relative to a root folder prec

decNATS -n NAOS_IVV_MCS_NATS -F3 -P "{\"filename\" :\"/tmp/2022-05-25T15:36:59.288624_PLA_SBA\",

F4. Trigger processing of MPS/FDS activities file
NATS subject used by AUTO to issue the request: CCS5.AutoPilot.NAOS.call
NATS body: ProcessActivityFile <PATH>
<PATH>: Path to the MPS/FDS activities XML file (corresponds to <target path> used in the ingest
Spacecraft activities file).

decNATS -n NAOS_IVV_MCS_NATS -F4 -P "{\"path\" :\"/CCS/VARIABLE/INPUTS/2022-05-25T15:36:59.288624
decNATS -n NAOS_IVV_MCS_NATS -F4 -P "{\"path\" :\"/CCS/VARIABLE/INPUTS/2022-05-27T09:14:38.449688

F5. Start dispatching a sequence of TCs created from an activities file
NATS subject used by AUTO to issue the request: CCS5.AutoPilot.NAOS.call
NATS body: UplinkActivityFile <filename>
<filename>: name of the activity file that was used to generate the TC sequence (without extensio
This request executes the TC sequence generated in F4 (above), starting the uplink of all TCs in
If COP1 is active, a command (either a PUS TC(11,4) carrier of a time-tagged TC or an immediate T
If COP1 is not active, the relevant stage for considering the TC uplinked is UV_GS_UPLINK (corres
After a successful uplink, the stack file is deleted, and the NATS reply message is simply "OK".

decNATS -n NAOS_IVV_MCS_NATS -F5 -P "{\"filename\" :\"2022-06-25T15:36:59\"}"

F6. Generate TM report (for FDS)
NATS subject used by AUTO to issue the request: CCS5.AutoPilot.NAOS.call
NATS body: HistoryReport <Report Type> <start time> <end time> <outputURL>
<Report Type>: GPS, THR, SCA, MSC (or other). This value identifies a report configuration file (
<start time> <endtime>: time range for the retrieval (e.g. last orbit), ISO format e.g. 2020-04-2
<outputURL>: destination of the final report.

decNATS -n NAOS_IVV_MCS_NATS -F6 -P "{\"type\" :\"GPS\", \"start\" : \"2022-07-06T01:00:00\" , \"
decNATS -n NAOS_IVV_MCS_NATS -F6 -P "{\"type\" :\"THR\", \"start\" : \"2022-07-06T01:00:00\" , \"

F7. Process Playback TM frames
NATS subject used by AUTO to issue the request: CCS5.AutoPilot.NAOS.call
NATS body: ReplayTmFrames <File path> <VC>
<File path> : TM File path (file absolute path, or relative to a preconfigured root folder for in
<VC> TM VC, filter for TM frames to replay (value: 0, 1, ...7) (optional argument, if not provide
The input TM file is a binary file of concatenated TM frames where each frame is 1115 bytes long
Frames are read out of the TM file and injected into CCS5 for processing. If synchronization is l
The process ends when all frames have been replayed, the caller (AUTO) is eventually notified wit


MessageBroker::subscribe -subject CCS5.AutoPilot.NAOS.call.ack referby ackMessage
```

```
trace add variable ackMessage write [lambda {var idx op} {
 syslog -ok "$tcl_name -> Received ACK message: [MessageBroker::getMessageBody $::ackMessage]"
}]


== Usage
decNATS -m <I/F SFTP> -n <I/F NATS> -F<function> -P {JSON parameters}
       --mnemonic  <MNEMONIC> (mnemonic is case sensitive)
       --Subject   <subject>
       --Body      <body>
       --nodb      no usage of the Inventory for recording operations
       --help      shows this help
       --usage     shows the usage
       --Debug     shows Debug info during the execution
       --version   shows version number
```

8.13 decNotify2Interface

```
== Usage
decNotify2Interface -m <MNEMONIC> --OK | --KO
--mnemonic  <MNEMONIC> (mnemonic is case sensitive)
--OK        notify success in the delivery to the I/F -f full_path_filelist list of files send
--KO        notify failure in the delivery to the I/F -f full_path_filelist list of files failed
--help      shows this help
--usage     shows the usage
--Debug     shows Debug info during the execution
--version   shows version number
```

8.14 decSend2Interface

```
== Synopsis

This is a DEC command line tool that deliver files to a given I/F in PUSH mode.
It delivers files using the configured protocols (s)ftp and email.
Files sent can be registered in an Inventory and the delivery date is set to the latest one.

This command can be used in order to send a given file just once
(for each delivery method: ftp, email) for a given Interface.
Use "-O" flag to enable this behaviour.

-R flag:

With this option (Report), a Report "List" with the new files sent is created.
```

```
This Report file is initally placed in the Interface local inbox.


== Usage
decSend2Interface -m <MNEMONIC> [-O] [--nodb]
--mnemonic  <MNEMONIC> (mnemonic is case sensitive)
--ONCE      The file is just sent once for that I/F
--AUTO      local outbox Automatic management
--loops <n> n is the number of Loop retries to achieve the Delivery
--delay <s> s seconds of delay between each Loop Retry
[60 secs by default if it is not specified]
--retries <r>  r is the number of retries on each Loop for each file
--Report    create a Report with the list of files delivered to the Interface
--list      list only (not downloading and no ingestion)
--Nomail    avoids mail notification to the I/F after successfully delivery
--Show      it shows all available I/Fs registered in the Inventory
--nodb      no usage of the Inventory for recording operations
--help      shows this help
--usage     shows the usage
--Debug     shows Debug info during the execution
--version   shows version number
```

8.15 decSmokeTests

8.16 decStats

```
== Synopsis

This is a DEC command line tool that shows file reception statistics

== Usage
decStats
--Hours <hours>            status of last n hours
--file <filename>          status of a given filename
--help                       shows this help
--usage                    shows the usage
--Debug                   shows Debug info during the execution
--version                  shows version number
```

8.17 auxConverter

```
== Synopsis

This is a Data Exchange Component (DEC) command line tool that manages the
```

```
file conversions / harmonisation according to every mission standard and
mission service providers such as ground stations.

This tool can be used manually from the command line or included in the DEC
configuration as part of any pull or push workflow.

For reference to the current file formant conversions and the auxiliary data
files supported, please execute the tool with the --version flag.

== Usage
auxConverter -m S3 -f <full_path_file> -d <dir>
    --mission <id>    mission identifier
      POD    - Copernicus Sentinel-3 Precise Orbit Determination
      S2     - Copernicus Sentinel-2
      S3     - Copernicus Sentinel-3
      NAOS   - NAOS mission
      KSAT   - Kongsberg Satellite Services
    --version         shows the version number / supported conversions
    --Debug           shows Debug info during the execution
    --help            shows this help
```

8.18 decUnitTests

8.19 decUnitTests_IERS

9 Test Reference

9.1 Introduction

This section describes the DEC SW tests, whose usage is mainly devoted to ensure the quality of every SW version released, but it also covers some dedicated tests to interface with some data providers.

The Unit Tests can be classified into two main categories, the ones devoted to test the general workflow of the SW and the ones which cover the different protocol handlers and their associated configurations within the SW.

The Unit Tests are the ones which are executed to validate every SW version; in opposition to the interface tests, in particular the ones exploiting public providers whose availability cannot be guaranteed to drive the release of the SW.

The definition of the unit testing environment is so far not covered; would there be some interest to execute the unit tests outside the development environment, this information can be formalised in this document.

The information about the test executions is naturally logged into files according to the configuration as any other operation performed by the SW. For the test units associated to the protocol handlers, every run with different test data is kept in different files for subsequent analysis if needed (e.g. test_pull_sftp_3_files_10GB_3Slots.log); the results of the test runner are usually sufficient for systematic testing purposes such as non-regression.

9.2 Unit Tests

9.2.1 Workflow

The testing of the DEC workflows and helpers are defined in a single test runner; these are independent with respect to the protocols used for the circulations. This section does not include information on every unit test but their whole execution since this is a pre-requisite to validate every SW version.

```
$> decUnitTests
```

9.2.2 Protocols

In general there is one test runner for every protocol handler which tests pull and push operations. Sometimes, specific protocol configuration are tested in additional test runners which also follow the convention for a specific test for pull (i.e. test_pull) and push (i.e. test_push) circulations respectively.

The tests are usually parametric in terms of: parallel transfers, number of files, size of the files, etc ; in this manner every test case is executed several times according the test parameters.

9.2.2.1 FTP

The FTP unit test performs the data transfers using the PORT mode in which the FTP client specifies the TCP ports opened for the server to connect to when pulling and pushing files with FTP GET and PUT respectively.

```
$> decUnitTests_FTP -n test_pull
```

```
$> decUnitTests_FTP -n test_pull
```

9.2.2.2 FTP Passive

There is a specific test runner for the FTP handler with the usual test_pull and test_push test cases to perform FTP GET and PUT requests respectively in FTP passive mode(cf. passive mode).

```
$> decUnitTests_FTP_PASSIVE -n test_pull
```

```
$> decUnitTests_FTP_PASSIVE -n test_push
```

9.2.2.3 FTPS Explicit

The FTPS protocol is the plain FTP application over the SSL/TLS which encrypts the TCP streams ; however the associated interface handler is different than the FTP one and therefore it is subject of dedicated unit tests. The unit tests make usage of a self-signed SSL certificate, hence the authenticity of the server is not verified to avoid the test failure and therefore the associated configuration is used accordingly (cf. VerifyPeerSSL).

```
$> decUnitTests_FTPS -n test_pull
```

```
$> decUnitTests_FTPS -n test_push
```

9.2.2.4 FTPS Implicit

Unlike the FTPS which is blah blah blah AUTH TLS TBW

9.2.2.5 HTTP

The HTTP unit test performs the data transfers using HTTP HEAD and GET requests for pull mode and HTTP PUT verb for push circulations respectively.

```
$> decUnitTests_HTTP -n test_pull
```

```
$> decUnitTests_HTTP -n test_push
```

9.2.2.6 LOCAL

The LOCAL protocol stands for circulation operations across the file-system, hence not implying explicitly any network transaction. Apriori the circulation operations are hardlinks; however the Interface Handler LOCAL is able to automatically failover operations performed across different file-systems, thus performing a copy in which the availability of the complete file is ensured at the end-point. The tests are performed in the same file-system hence verifying the hardlink circulation operations.

```
$> decUnitTests_LOCAL -n test_pull
```

```
$> decUnitTests_LOCAL -n test_push
```

9.2.2.7 SFTP

The unit tests for SFTP cover both pull and push circulations.

```
$> decUnitTests_SFTP -n test_pull
```

```
$> decUnitTests_SFTP -n test_push
```

9.2.2.8 SMTP Notifications

9.2.2.9 WebDAV

The WebDAV operations are performed on top of the base HTTP protocol. Pull operations are performed using the WebDAV extension PROPFIND request to look for elements according to the configuration. Push operations leverage the MOVE method to ensure the integrity of the push circulations end to end ; incomplete HTTP put could be perfectly handled by the end-peer but this mechanism introduces an additional verification step since it requires that the server has successfully dumped the received data from memory into a file.

```
$> decUnitTests_HTTP -n test_pull
```

9.2.2.10 NATS

The DEC SW includes a generic NATS client to perform requests and subscribe to messages (cf. decNATS). Such tool has been crafted in the framework of NAOS mission. The protocol NATS

is a message oriented application protocol whose implementation is delegated to well-proven libraries in the open source domain. For verification purposes, there is no dedicated unit test but an interface test which exploits decNATS and hence it verifies correct usage of NATS request and subscribe messages. The interface test is *decTestInterface_NATS_CCS5*.

9.3 Interface Tests

This sections brings a brief reference to some dedicated interface tests which may be leveraged. The principle is that the general DEC SW unit test capacities can be expanded and tailored with minimum effort to address dedicated interface tests which can also be exploited in IVV and Production environments.

Below the available test runners are listed; some are covered at the annex of this document when addressing the Public Data Providers. But there are also dedicated interface tests whose objective is to verify project dedicated exchanges:

- **decTestInterface_CelesTrak** : interface test exploiting CelesTrak APIs
- **decTestInterface_CloudFerro** : dec_s2_test_push_lisboa@e2espm-inputhub.gem node dedicated test of interfaces in the frame of the SBOA service to ESA-ESRIN for Copernicus
- **decTestInterface_ECDC** : Interface test exploiting some ECDC HTTP resources
- **decTestInterface_GNSS** : Interface test exploiting ESA GNSS server
- **decTestInterface_IERS** : Interface test exploiting ftp.iers.org server
- **decTestInterface_ILRS** : Interface test exploiting EUROLAS Data Center server
- **decTestInterface_OData_ADGS** : this is a functional interface test to verify the OData Server implementation DEC can support
- **decTestInterface_OData_DHUS** : Interface test exploiting OData API by Copernicus DIAS (OpenHub)
- **decTestInterface_NATS_CCS5** : Interface test exploiting NATS NAOS MCS / CCS5
- **decTestInterface_NASA** : Interface test exploiting NASA CDDIS & MSFC data
- **decTestInterface_NOAA** : Interface test exploiting NOAA Space Weather Prediction Center server
- **decTestInterface_S2PRIP** : Interface test exploiting internal ESA Copernicus Production Service for Sentinel-2 mission
- **decTestInterface_SCIHUB** : Interface test exploiting Copernicus DIAS (OpenHub) bulk catalogue export by HTTP in CSV files
- **decTestInterface_SPCS** : Interface test exploiting SPCS API for CDM

9.3.1 Celestrak

This interface test with Celestrak comprises the following data-flows:

- **TLE** : retrieval of TLE associated to different missions such as DE1, DE2, S2A, S2B updated on daily basis.
- **TCA** : retrieval of TLE catalogue of objects monitored, which is updated on a daily basis.
- **SFS** : retrieval of space-weather forecast short-term covering the ”Ap” and ”F10.7” indexes, covering 45 days generated on a daily basis.

9.3.2 NASA

This interface test with NASA comprises the following data-flows:

- **NBULA** : retrieval of NASA Bulletin A with Earth Rotation Parameters updated on weekly basis.
- **NBULC** : retrieval of NASA Bulletin C with TAI-UTC correlation and which is only updated by the leap second announcements ; latest is in 2017.
- **SFL** : retrieval of space-weather forecast long-term covering the ”Ap” and ”F10.7” indexes covering updated on monthly basis.

9.4 Integration Tests

There are some specific DEC tests which leverage the DEC CICD chain and testing infrastructure devoted for integration within projects. These tests are not generic but fairly straightforward to produce allowing to verify the interface integration among the service back-ends of the project. This information is included in this document for advise to ask for them if deemed valuable by any project.

A Configuration General

A.1 General Configuration

The general configuration file *dec_config.xml* carries the items to drive the DEC report generation and workflow of files prior their circulation ; additionally it carries the *Inventory* configuration items.

A.2 Example dec_config.xml

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

   <Project>
      <Name>Data Exchange Component</Name>
        <Id>DEC</Id>
        <Mission>UNIT TESTS</Mission>
        <SatPrefix>S2_</SatPrefix>
   </Project>

<Filters>
<IncomingFilters>
<Filter>*</Filter>
</IncomingFilters>
<OutgoingFilters>
<Filter>*</Filter>
</OutgoingFilters>
</Filters>

   <Options>
      <Upload>
          <DeleteSourceFiles>true</DeleteSourceFiles>
          <UploadFilePrefix>false</UploadFilePrefix>
          <UploadDirs>false</UploadDirs>
      </Upload>
      <Download>
         <DownloadDirs>false</DownloadDirs>
         <DeleteUnknownFiles>false</DeleteUnknownFiles>
         <DeleteDuplicatedFiles>true</DeleteDuplicatedFiles>
```

```
<!--
<LogUnknownFiles>false</LogUnknownFiles>
<LogDuplicatedFiles>false</LogDuplicatedFiles>
-->
         </Download>
</Options>


<Reports>

<!-- DDC delivery reports / push (decSend2Interface) -->

   <Report Name="DeliveredFiles">
      <Enabled>true</Enabled>
      <Desc>List of Files Delivered</Desc>
      <FileClass>OPER</FileClass>
      <FileType>DEC_F_SENT</FileType>
   </Report>

   <Report Name="EmergencyDeliveredFiles">
      <Enabled>false</Enabled>
      <Desc>List of Files Delivered in Emergency Mode</Desc>
      <FileClass>OPER</FileClass>
      <FileType>DEMERGENCY</FileType>
   </Report>

<!-- DCC Retrieval reports (decGetFromInterface) -->

   <Report Name="RetrievedFiles">
      <Enabled>true</Enabled>
      <Desc>List of Files Retrieved</Desc>
      <FileClass>OPER</FileClass>
      <FileType>DEC_F_RECV</FileType>
   </Report>

   <Report Name="UnknownFiles">
      <Enabled>true</Enabled>
      <Desc>List of unknown Files present</Desc>
      <FileClass>OPER</FileClass>
      <FileType>DECUNKNOWN</FileType>
   </Report>

</Reports>

<Workflow>
   <SourceDir>/tmp/dec_delivery_root</SourceDir>
   <GlobalOutbox>/tmp/dec/outbox</GlobalOutbox>
   <ReportDir>/tmp/dec/in_basket_if_localhost_notsecure</ReportDir>
   <TempDir>/tmp/dec/tmp</TempDir>
```

```
</Workflow>

<Inventory>
   <Database_Adapter>sqlite3</Database_Adapter>
   <Database_Name>/tmp/dec_inventory</Database_Name>
   <Database_User>root</Database_User>
   <Database_Password>2mysql</Database_Password>
</Inventory>

</Configuration>
```

B Configuration Interfaces

B.1 Interface Configuration

The general configuration file *dec_interfaces.xml* carries the items TBW TBW TBW.

B.2 Example dec_interfaces.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Interfaces xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <Interface Name="IERS">

     <Desc>IERS Interface</Desc>

     <TXRXParams>
         <Enabled4Sending>false</Enabled4Sending>
         <Enabled4Receiving>true</Enabled4Receiving>
         <ImmediateRetries>3</ImmediateRetries>
         <LoopRetries>2</LoopRetries>
         <LoopDelay unit="s">300</LoopDelay>
         <PollingInterval unit="s">20</PollingInterval>
         <PollingSize>470</PollingSize>
         <ParallelDownload>5</ParallelDownload>
     </TXRXParams>

     <Server>
        <Protocol>FTP</Protocol>
        <Hostname>ftp.iers.org</Hostname>
        <Port>21</Port>
        <User>anonymous</User>
        <Pass>guest</Pass>
        <RegisterContentFlag>false</RegisterContentFlag>
        <RetrieveContentFlag>true</RetrieveContentFlag>
        <SecureFlag>false</SecureFlag>
        <VerifyPeerSSL>false</VerifyPeerSSL>
        <CompressFlag>false</CompressFlag>
        <PassiveFlag>true</PassiveFlag>
        <CleanUpFreq Unit="s">5</CleanUpFreq>
```

```
      </Server>

      <DeliverByMailTo>
       <Address>mario.bross@gmail.com</Address>
      </DeliverByMailTo>

   <Notify>
  <SendNotification>false</SendNotification>
  <To>
<Address>mario.draghi@supereuro.com</Address>
   </To>
   </Notify>

   <Events>
      <Event Name="NewFile2InTray"          executeCmd="auxConverter -f %F" />
   </Events>

   <ContactInfo>
      <Name>Mr draghi</Name>
      <EMail>mario.draghi@supereuro.com</EMail>
      <Tel>0039-800 454 432</Tel>
      <Fax>0039-800 454 433</Fax>
     <Address>Citta Uova</Address>
   </ContactInfo>

   </Interface>

</Interfaces>
```

C Configuration Pull Circulations

C.1 General Configuration

The general configuration file *dec_incoming_files.xml* carries the items to drive pull circulations from the defined interfaces.

C.2 Example dec_incoming_files.xml

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

<Config>

   <ListInterfaces xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <Interface>
      <Name>ECDC</Name>
      <LocalInbox>/tmp/dec/in_basket_if_ecdc</LocalInbox>
      <DownloadDirs>
          <Directory DepthSearch="0">covid19/casedistribution/csv</Directory>
      </DownloadDirs>
      <Switches>
         <DeleteDownloaded>false</DeleteDownloaded>
         <DeleteDuplicated>false</DeleteDuplicated>
         <DeleteUnknown>false</DeleteUnknown>
         <LogDuplicated>false</LogDuplicated>
         <LogUnknown>false</LogUnknown>
         <MD5>false</MD5>
      </Switches>

   </ListInterfaces>

   <DownloadRules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

      <File Type="csv">
         <Description>Comma Separated Value</Description>
         <FromList>
            <Interface>ECDC</Interface>
         </FromList>
```

```
      </File>

 </DownloadRules>

   <DisseminationRules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <ListIntrays>

      <Intray>
   <Name>DIAS</Name>
         <Directory>/tmp/dec_final_dissemination/DIAS/</Directory>
      </Intray>

   </ListIntrays>

      <ListFilesDisseminated>

  <!--
          SORT Rules ranking from more restrictive ones
          Only the first rule matching the file pattern is applied
       -->

        <File Type="S2?_*OPENHUB*.csv">
           <HardLink>false</HardLink>
              <ToList>
                 <Intray>DIAS</Intray>
              </ToList>
        </File>

</Configuration>
```

D Configuration Push Circulations

D.1 General Configuration

The general configuration file *dec_outgoing_files.xml* carries the items to drive push circulations from the defined interfaces.

D.2 Example dec_outgoing_files.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<ListInterfaces>

<Interface>
<Name>PUSH_LISBOA</Name>
<LocalOutbox>/volumes/dec/interfaces/cloud_sboa.out</LocalOutbox>
<UploadDir>/s2pdgs_in</UploadDir>
<UploadTemp>/s2pdgs_in/_TEMP_</UploadTemp>
</Interface>

<Interface>
<Name>HTTP_FERRO</Name>
<LocalOutbox>/volumes/dec/interfaces/cloud_sboa.out.webdav</LocalOutbox>
<UploadDir>/</UploadDir>
<UploadTemp>/.tmp</UploadTemp>
</Interface>

<Interface>
<Name>TEST_HTTP_FERRO</Name>
<LocalOutbox>/volumes/dec/interfaces/cloud_sboa.out.webdav</LocalOutbox>
<UploadDir>/.tmp/test_final</UploadDir>
<UploadTemp>/.tmp/test_final/tmp</UploadTemp>
</Interface>


<Interface>
<Name>PUSH_FERRO</Name>
<LocalOutbox>/volumes/dec/interfaces/cloud_sboa.out</LocalOutbox>
```

```
<UploadDir>fromESRIN</UploadDir>
<UploadTemp>fromESRIN/_tmp_</UploadTemp>
</Interface>


</ListInterfaces>


<ListFiles>

<File Type="*MPL_ORBPRE*">
<Description>FOS Predicted Orbit File</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*MPL__NPPF_*">
<Description>Satellite Plan File</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*MPL_SP????*">
<Description>Station Plan File</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*REP_OPHKTM*">
<Description>HKTM PDI downlink orbit</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>
```

```
<File Type="S2?_OPER_REP_OPDPC*">
<Description>DPC Processing Reports</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="S2?_OPER_REP_ARC____????_??????????????_V??????????????_??????????????.EOF">
<Description>DPC Processing Reports</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="S2?_????_REP_OPDHUS_????_????????T??????.*">
<Description>DAGC DHUS Pick-up Point</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*REP_OPDAM?*">
<Description>DAM Ingestion Report</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*REP_OPDC__*">
<Description>DC Circulation Report</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
```

```
</File>

<File Type="*REP_OPAI__*">
<Description>DC Circulation Report</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*REP_PASS_*">
<Description>EISP / DFEP Report</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*SRA_EDRS_A*">
<Description>SRA File for EDRS_A</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

<File Type="*DEC_F_RECV*">
<Description>DEC pulled files</Description>
<ToList>
<Interface Name="PUSH_LISBOA"       Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="PUSH_FERRO"        Compress="NONE"   DeliveryMethods="FTP" />
<Interface Name="HTTP_FERRO"        Compress="NONE"   DeliveryMethods="HTTP" />
<Interface Name="TEST_HTTP_FERRO"   Compress="NONE"   DeliveryMethods="HTTP" />
</ToList>
</File>

</ListFiles>

</Config>
```

E Configuration Log

E.1 Log4r Configuration

The log capability by DEC SW are based on the *ruby* Log4r capabilities, which is a porting of the widely spread Java library Log4j.

The associated configuration file *dec_log_config.xml* is not DEC SW specific and it can be directly handled by *log4r* and can be similar to other components which use the same logging *Log4r* library.

E.1.1 Rolling File Outputter

This section recalls the Log4r *RollingFileOutputter* which is used by default in DEC SW.

- *type* : RollingFileOutputter
- *filename* : pull path filename
- *formatter* : PatternFormatter
- *trunc* : If true, deletes the existing log files, otherwise continues logging where it left off last time
- *maxsize* : Maximum size of every log file in bytes
- *max_backups* : Maximum number of prior log files kept
- *maxtime* : Maximum age of every log file in seconds

E.1.2 Server

The *Server* configuration item defines the parameters which rule the network protocol of choice selected to rule the file circulations either in *pull* and *push* mode.

E.2 Example dec_log_config.xml

```
<log4r_config>

   <pre_config>
      <parameter name="mainLoggerName" value="NODE_1"/>
   </pre_config>

   <outputter name="console"  level="INFO" >
      <type>StdoutOutputter</type>
      <formatter type="Log4r::PatternFormatter" pattern="[%5l] %d %c.#{moduleName} - %m">
      <date_pattern>%Y-%m-%d %H:%M:%S</date_pattern>
      </formatter>
   </outputter>

   <outputter name="dec_log" level="DEBUG">
      <type>RollingFileOutputter</type>
      <filename>/tmp/DEC.log</filename>
      <formatter type="PatternFormatter" pattern="[%5l] %d %c.#{moduleName} - %m">
      <date_pattern>%Y-%m-%d %H:%M:%S</date_pattern>
      </formatter>
      <trunc>false</trunc>
      <maxsize>10000000</maxsize>
      <max_backups>4</max_backups>
      <maxtime>2592000</maxtime>
   </outputter>

   <logger name="NODE_1" level="DEBUG" trace="false">
      <outputter>console</outputter>
      <outputter>dec_log</outputter>
   </logger>

</log4r_config>
```

F Public Data Providers

F.1 Introduction

This section includes some few open public data providers of auxiliary data used in different domains (earth observation, navigation, geodesy, space weather, public health, etc), for which there is already a DEC configuration available to exploit them. Any *public data provider* can be supported by DEC, but the ones described here have been selected to demonstrate the DEC SW capabilities considering that such interfaces have been used by several projects. As such, every provider is subject of a dedicated test interface which does not simulate the provider but it exploits the real interface. Also the selection of interfaces is driven by the nature of the different protocols used in order to have some variety.

The dedicated test interfaces aim at verifying the full capacity to exploit a given ADP, hence they comprise all the usual steps since configuration verification, listing up, data retrieval, local transformation up to the final local dissemination for consumption (i.e. pull circulations). It is noted that there are dedicated test interface project-wise to support both pull and push circulations but ADP, which are open, public and sometimes anonymous do not support push circulations by definition.

The information included in the following sections do not represent the full interface test suite for each data provider but it aims at showing some of the DEC natural capabilities whilst demonstrating the correct exploitation of those end-points.

F.2 CelesTrak

Celestrak publishes the NORAD satellite catalogue of TLE with a convenient public anonymous API.

F.2.1 Satellite TLE

The DEC SW queries the CelesTrak API to check its availability and request the spacecraft according to its SATCAT identifier (i.e. NORAD identifier).
Deimos satellites can be retrieved using CELES_D1 and CELES_D2 interface workflows, which are defined to retrieve the TLE of DE1 and DE2 spacecrafts, which are subsequently dumped into files and renamed accordingly for their unambiguous identification. Other missions such as Copernicus Sentinels have also been tested.

```
$> decTestInterface_CelesTrak
```

```
[ INFO] 2022-03-02 10:32:22 NODE_1.test - DECTestCaseCelesTrak::test_pull => START
[DEBUG] 2022-03-02 10:32:23 NODE_1.test - rm -f /tmp/dec/in_basket_if_celes/*
[DEBUG] 2022-03-02 10:32:23 NODE_1.test - decGetFromInterface -m CELES_D1 -l --nodb
[ INFO] 2022-03-02 10:32:25 NODE_1.pull - [DEC_005] I/F CELES_D1: Polling Started - List mode is true
[ INFO] 2022-03-02 10:32:26 NODE_1.pull - [DEC_105] I/F CELES_D1: File gp.php?CATNR=35681 is available
[ INFO] 2022-03-02 10:32:26 NODE_1.pull - [DEC_060] I/F CELES_D1: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-03-02 10:32:26 NODE_1.pull - [DEC_100] I/F CELES_D1: Pull iteration completed successfully
[DEBUG] 2022-03-02 10:32:28 NODE_1.test - decGetFromInterface -m CELES_D1 --nodb
[ INFO] 2022-03-02 10:32:30 NODE_1.pull - [DEC_005] I/F CELES_D1: Polling Started - List mode is false
[ INFO] 2022-03-02 10:32:31 NODE_1.pull - [DEC_060] I/F CELES_D1: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-03-02 10:32:32 NODE_1.pull - [DEC_110] I/F CELES_D1: gp.php?CATNR=35681 downloaded with size 168 bytes
[ INFO] 2022-03-02 10:32:32 NODE_1.pull - [DEC_130] I/F CELES_D1: event triggered onreceivenewfilesok =>
mv /tmp/dec/in_basket_if_celes/gp.php?CATNR=35681 /tmp/dec/in_basket_if_celes/"$(date +"%Y%m%d%H%M%S_DEIMOS-1.tle")"
[ INFO] 2022-03-02 10:32:32 NODE_1.pull - [DEC_130] I/F CELES_D1: event completed onreceivenewfilesok =>
mv /tmp/dec/in_basket_if_celes/gp.php?CATNR=35681 /tmp/dec/in_basket_if_celes/"$(date +"%Y%m%d%H%M%S_DEIMOS-1.tle")"
[ INFO] 2022-03-02 10:32:32 NODE_1.pull - [DEC_135] I/F CELES_D1: S2__OPER_DEC_F_RECV_2BOA_20220302T103232_V20220302T103232_20220302T103232_CELES_D1.xml pull report created
[ INFO] 2022-03-02 10:32:32 NODE_1.pull - [DEC_115] Intray TLE: 20220302103232_DEIMOS-1.tle disseminated into /tmp/dec_local_dissemination/TLE
[ INFO] 2022-03-02 10:32:32 NODE_1.pull - [DEC_100] I/F CELES_D1: Pull iteration completed successfully
[DEBUG] 2022-03-02 10:32:37 NODE_1.test - decGetFromInterface -m CELES_D2 -l --nodb
[ INFO] 2022-03-02 10:32:39 NODE_1.pull - [DEC_005] I/F CELES_D2: Polling Started - List mode is true
[ INFO] 2022-03-02 10:32:40 NODE_1.pull - [DEC_105] I/F CELES_D2: File gp.php?CATNR=40013 is available
[ INFO] 2022-03-02 10:32:40 NODE_1.pull - [DEC_060] I/F CELES_D2: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-03-02 10:32:40 NODE_1.pull - [DEC_100] I/F CELES_D2: Pull iteration completed successfully
[DEBUG] 2022-03-02 10:32:42 NODE_1.test - decGetFromInterface -m CELES_D2 --nodb
[ INFO] 2022-03-02 10:32:43 NODE_1.pull - [DEC_005] I/F CELES_D2: Polling Started - List mode is false
[ INFO] 2022-03-02 10:32:44 NODE_1.pull - [DEC_060] I/F CELES_D2: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-03-02 10:32:45 NODE_1.pull - [DEC_110] I/F CELES_D2: gp.php?CATNR=40013 downloaded with size 168 bytes
[ INFO] 2022-03-02 10:32:45 NODE_1.pull - [DEC_130] I/F CELES_D2: event triggered onreceivenewfilesok =>
mv /tmp/dec/in_basket_if_celes/gp.php?CATNR=40013 /tmp/dec/in_basket_if_celes/"$(date +"%Y%m%d%H%M%S_DEIMOS-2.tle")"
[ INFO] 2022-03-02 10:32:45 NODE_1.pull - [DEC_130] I/F CELES_D2: event completed onreceivenewfilesok =>
mv /tmp/dec/in_basket_if_celes/gp.php?CATNR=40013 /tmp/dec/in_basket_if_celes/"$(date +"%Y%m%d%H%M%S_DEIMOS-2.tle")"
[ INFO] 2022-03-02 10:32:45 NODE_1.pull - [DEC_135] I/F CELES_D2: S2__OPER_DEC_F_RECV_2BOA_20220302T103245_V20220302T103245_20220302T103245_CELES_D2.xml pull report created
[ INFO] 2022-03-02 10:32:45 NODE_1.pull - [DEC_115] Intray TLE: 20220302103245_DEIMOS-2.tle disseminated into /tmp/dec_local_dissemination/TLE
[ INFO] 2022-03-02 10:32:45 NODE_1.pull - [DEC_100] I/F CELES_D2: Pull iteration completed successfully
[DEBUG] 2022-03-02 10:32:50 NODE_1.test - grep ERROR /tmp/DEC000001.log
[ INFO] 2022-03-02 10:32:50 NODE_1.test - DECTestCaseCelesTrak::test_pull => END
```

F.2.2 CelesTrak TLE Catalogue

The DEC SW queries the CelesTrak API to obtain the TLE catalogue of active objects which they are monitoring.

```
$> decTestInterface_CelesTrak -n testTCA
```

```
[ INFO] 2022-08-29 09:00:30.185 nl2-u-moc-srv-01.test - DECTestCaseCelesTrak::testTCA START
[DEBUG] 2022-08-29 09:00:30.185 nl2-u-moc-srv-01.test - rm -f /tmp/dec/int/NS1_OPER_AUX_TCA____*
[DEBUG] 2022-08-29 09:00:30.195 nl2-u-moc-srv-01.test - rm -f /tmp/dec/in_basket_if_celes/tca/*
[DEBUG] 2022-08-29 09:00:30.202 nl2-u-moc-srv-01.test - decConfigInterface2DB -a CELESTRAK_TCA
[ INFO] 2022-08-29 09:00:31.235 nl2-u-moc-srv-01.db   - [DEC_001] I/F CELESTRAK_TCA: added to the DEC Inventory/Interface db
[ INFO] 2022-08-29 09:00:31.246 nl2-u-moc-srv-01.test - decCheckConfig -e CELESTRAK_TCA
[DEBUG] 2022-08-29 09:00:35.982 nl2-u-moc-srv-01.chck - [DEC_XXX] I/F CELESTRAK_TCA: File gp.php?GROUP=active&FORMAT=2le is available
[ INFO] 2022-08-29 09:00:35.989 nl2-u-moc-srv-01.chck - [DEC_004] I/F CELESTRAK_TCA: exchange point is reachable
[ INFO] 2022-08-29 09:00:36.570 nl2-u-moc-srv-01.chck - [DEC_003] I/F CELESTRAK_TCA: interface is correctly declared in DEC/Inventory
[ INFO] 2022-08-29 09:00:36.584 nl2-u-moc-srv-01.test - decGetFromInterface -m CELESTRAK_TCA -l
```

```
[ INFO] 2022-08-29 09:00:38.443 nl2-u-moc-srv-01.pull - [DEC_005] I/F CELESTRAK_TCA: Polling Started - List mode is true
[ INFO] 2022-08-29 09:00:41.998 nl2-u-moc-srv-01.pull - [DEC_105] I/F CELESTRAK_TCA: File gp.php?GROUP=active&FORMAT=21e is available
[ INFO] 2022-08-29 09:00:41.999 nl2-u-moc-srv-01.pull - [DEC_060] I/F CELESTRAK_TCA: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 09:00:41.999 nl2-u-moc-srv-01.pull - [DEC_100] I/F CELESTRAK_TCA: Pull iteration completed successfully
[ INFO] 2022-08-29 09:00:42.020 nl2-u-moc-srv-01.test - decGetFromInterface -m CELESTRAK_TCA
[ INFO] 2022-08-29 09:00:44.460 nl2-u-moc-srv-01.pull - [DEC_005] I/F CELESTRAK_TCA: Polling Started - List mode is false
[ INFO] 2022-08-29 09:00:47.859 nl2-u-moc-srv-01.pull - [DEC_060] I/F CELESTRAK_TCA: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 09:00:50.490 nl2-u-moc-srv-01.pull - [DEC_110] I/F CELESTRAK_TCA: gp.php?GROUP=active&FORMAT=21e downloaded with size 901984 bytes
[ INFO] 2022-08-29 09:00:50.546 nl2-u-moc-srv-01.pull - [DEC_115] Intray NAOS_IN: "gp.php?GROUP=active&FORMAT=21e" disseminated into /data/mocExternalInterfaces/tmp/naos/
[ INFO] 2022-08-29 09:00:50.547 nl2-u-moc-srv-01.pull - [DEC_130] I/F CELESTRAK_TCA: event triggered newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/"gp.php?GROU
[ INFO] 2022-08-29 09:00:50.858 nl2-u-moc-srv-01.auxc - [AUX_001] NS1_OPER_AUX_TCA____20220829T000000_21000101T000000_0001.TXT generated from gp.php?GROUP=active&FORMAT=21e
[ INFO] 2022-08-29 09:00:50.861 nl2-u-moc-srv-01.pull - [DEC_130] I/F CELESTRAK_TCA: event completed newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/"gp.php?GROU
[ INFO] 2022-08-29 09:00:50.913 nl2-u-moc-srv-01.pull - [DEC_100] I/F CELESTRAK_TCA: Pull iteration completed successfully
[ INFO] 2022-08-29 09:00:50.933 nl2-u-moc-srv-01.test - decGetFromInterface -m CELESTRAK_TCA
[ INFO] 2022-08-29 09:00:53.089 nl2-u-moc-srv-01.pull - [DEC_005] I/F CELESTRAK_TCA: Polling Started - List mode is false
[ INFO] 2022-08-29 09:00:56.571 nl2-u-moc-srv-01.pull - [DEC_060] I/F CELESTRAK_TCA: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 09:00:59.329 nl2-u-moc-srv-01.pull - [DEC_111] I/F CELESTRAK_TCA: gp.php?GROUP=active&FORMAT=21e downloaded is duplicated / same md5
[ INFO] 2022-08-29 09:00:59.379 nl2-u-moc-srv-01.pull - [DEC_100] I/F CELESTRAK_TCA: Pull iteration completed successfully
[ INFO] 2022-08-29 09:00:59.397 nl2-u-moc-srv-01.test - decStats
[ INFO] 2022-08-29 09:01:00.372 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"gp.php?GROUP=active&FORMAT=21e","interface":"CELESTRAK_TCA","date":"2022-08-29 07:00:50 UTC","protocol"
[ INFO] 2022-08-29 09:01:00.372 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"250.00 B/s","volume":"880.84 KiB"}
[ INFO] 2022-08-29 09:01:00.375 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[DEBUG] 2022-08-29 09:01:00.383 nl2-u-moc-srv-01.test - Verification of log trace: TCA conversion to NAOS mission
[DEBUG] 2022-08-29 09:01:00.384 nl2-u-moc-srv-01.test - grep AUX_001 /data/mocExternalInterfaces/log/DEC000001.log
[ INFO] 2022-08-29 09:00:50.858 nl2-u-moc-srv-01.auxc - [AUX_001] NS1_OPER_AUX_TCA____20220829T000000_21000101T000000_0001.TXT generated from gp.php?GROUP=active&FORMAT=21e
[DEBUG] 2022-08-29 09:01:00.388 nl2-u-moc-srv-01.test - Verification of log trace: Duplication of TCA
[DEBUG] 2022-08-29 09:01:00.388 nl2-u-moc-srv-01.test - grep DEC_111 /data/mocExternalInterfaces/log/DEC000001.log
[ INFO] 2022-08-29 09:00:59.330 nl2-u-moc-srv-01.pull - [DEC_111] I/F CELESTRAK_TCA: gp.php?GROUP=active&FORMAT=21e downloaded is duplicated / same md5
[ INFO] 2022-08-29 09:01:00.396 nl2-u-moc-srv-01.test - DECTestCaseCelesTrak::testTCA END
```

F.2.3 CelesTrak Space-weather

The DEC SW queries the CelesTrak interface to retrieve the resource SW-Last5Years.txt which is updated on daily basis carrying a 45-day forecast of daily ap and F10.7 index.

```
$> decTestInterface_CelesTrak -n testSFS
```

```
[ INFO] 2022-08-29 09:04:29.134 nl2-u-moc-srv-01.test - DECTestCaseCelesTrak::testSFS START
[DEBUG] 2022-08-29 09:04:29.134 nl2-u-moc-srv-01.test - decConfigInterface2DB -a CELESTRAK_SFS
[ INFO] 2022-08-29 09:04:30.155 nl2-u-moc-srv-01.db   - [DEC_001] I/F CELESTRAK_SFS: added to the DEC Inventory/Interface db
[ INFO] 2022-08-29 09:04:30.181 nl2-u-moc-srv-01.test - decCheckConfig -e CELESTRAK_SFS
[DEBUG] 2022-08-29 09:04:33.528 nl2-u-moc-srv-01.chck - [DEC_XXX] I/F CELESTRAK_SFS: File SW-Last5Years.txt is available
[ INFO] 2022-08-29 09:04:33.537 nl2-u-moc-srv-01.chck - [DEC_004] I/F CELESTRAK_SFS: exchange point is reachable
[ INFO] 2022-08-29 09:04:34.172 nl2-u-moc-srv-01.chck - [DEC_003] I/F CELESTRAK_SFS: interface is correctly declared in DEC/Inventory
[ INFO] 2022-08-29 09:04:34.187 nl2-u-moc-srv-01.test - decGetFromInterface -m CELESTRAK_SFS -l
[ INFO] 2022-08-29 09:04:36.285 nl2-u-moc-srv-01.pull - [DEC_005] I/F CELESTRAK_SFS: Polling Started - List mode is true
[ INFO] 2022-08-29 09:04:38.032 nl2-u-moc-srv-01.pull - [DEC_105] I/F CELESTRAK_SFS: File SW-Last5Years.txt is available
[ INFO] 2022-08-29 09:04:38.032 nl2-u-moc-srv-01.pull - [DEC_060] I/F CELESTRAK_SFS: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 09:04:38.032 nl2-u-moc-srv-01.pull - [DEC_100] I/F CELESTRAK_SFS: Pull iteration completed successfully
[ INFO] 2022-08-29 09:04:38.048 nl2-u-moc-srv-01.test - decGetFromInterface -m CELESTRAK_SFS
[ INFO] 2022-08-29 09:04:39.934 nl2-u-moc-srv-01.pull - [DEC_005] I/F CELESTRAK_SFS: Polling Started - List mode is false
[ INFO] 2022-08-29 09:04:41.712 nl2-u-moc-srv-01.pull - [DEC_060] I/F CELESTRAK_SFS: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 09:04:43.346 nl2-u-moc-srv-01.pull - [DEC_110] I/F CELESTRAK_SFS: SW-Last5Years.txt downloaded with size 308713 bytes
[ INFO] 2022-08-29 09:04:43.400 nl2-u-moc-srv-01.pull - [DEC_115] Intray NAOS_IN: SW-Last5Years.txt disseminated into /data/mocExternalInterfaces/tmp/naos/
[ INFO] 2022-08-29 09:04:43.401 nl2-u-moc-srv-01.pull - [DEC_130] I/F CELESTRAK_SFS: event triggered newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/SW-Last5Yea
[ INFO] 2022-08-29 09:04:43.720 nl2-u-moc-srv-01.auxc - [AUX_001] NS1_OPER_AUX_SFS____20220829T000000_20221012T000000_0001.TXT generated from SW-Last5Years.txt
[ INFO] 2022-08-29 09:04:43.723 nl2-u-moc-srv-01.pull - [DEC_130] I/F CELESTRAK_SFS: event completed newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/SW-Last5Yea
[ INFO] 2022-08-29 09:04:43.768 nl2-u-moc-srv-01.pull - [DEC_100] I/F CELESTRAK_SFS: Pull iteration completed successfully
[ INFO] 2022-08-29 09:04:43.786 nl2-u-moc-srv-01.test - decGetFromInterface -m CELESTRAK_SFS
[ INFO] 2022-08-29 09:04:45.677 nl2-u-moc-srv-01.pull - [DEC_005] I/F CELESTRAK_SFS: Polling Started - List mode is false
[ INFO] 2022-08-29 09:04:47.376 nl2-u-moc-srv-01.pull - [DEC_060] I/F CELESTRAK_SFS: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 09:04:49.003 nl2-u-moc-srv-01.pull - [DEC_111] I/F CELESTRAK_SFS: SW-Last5Years.txt downloaded is duplicated / same md5
[ INFO] 2022-08-29 09:04:49.047 nl2-u-moc-srv-01.pull - [DEC_100] I/F CELESTRAK_SFS: Pull iteration completed successfully
[ INFO] 2022-08-29 09:04:49.065 nl2-u-moc-srv-01.test - decStats
[ INFO] 2022-08-29 09:04:50.028 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"SW-Last5Years.txt","interface":"CELESTRAK_SFS","date":"2022-08-29 07:04:43 UTC","protocol":"HTTP","size"
[ INFO] 2022-08-29 09:04:50.028 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"85.00 B/s","volume":"301.48 KiB"}
[ INFO] 2022-08-29 09:04:50.031 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[DEBUG] 2022-08-29 09:04:50.039 nl2-u-moc-srv-01.test - Verification of log trace: SFS conversion to NAOS mission
[DEBUG] 2022-08-29 09:04:50.039 nl2-u-moc-srv-01.test - grep AUX_001 /data/mocExternalInterfaces/log/DEC000001.log
[ INFO] 2022-08-29 09:04:43.720 nl2-u-moc-srv-01.auxc - [AUX_001] NS1_OPER_AUX_SFS____20220829T000000_20221012T000000_0001.TXT generated from SW-Last5Years.txt
[DEBUG] 2022-08-29 09:04:50.043 nl2-u-moc-srv-01.test - Verification of log trace: Duplication of short-term weather forecast
[DEBUG] 2022-08-29 09:04:50.043 nl2-u-moc-srv-01.test - grep DEC_111 /data/mocExternalInterfaces/log/DEC000001.log
[ INFO] 2022-08-29 09:04:49.003 nl2-u-moc-srv-01.pull - [DEC_111] I/F CELESTRAK_SFS: SW-Last5Years.txt downloaded is duplicated / same md5
[ INFO] 2022-08-29 09:04:50.051 nl2-u-moc-srv-01.test - DECTestCaseCelesTrak::testSFS END
```

F.3 Copernicus Open Hub

The Copernicus Open Access Hub for EO Products of the Sentinels missions with fresh production.

F.3.1 Copernicus Precise Orbit Determination

The COAH publishes GNSS products belonging to the Sentinels.

```
 decODataClient -u ******** -p ******** -c "2022-03-07T00:00:00.000,2022-03-08T00:00:00.000" -q dhus:gnss:S1A -L /tmp -r
[ INFO] 2022-03-08 15:52:09 NODE_1.odata - [DEC_257] I/F DHUS: created DEC_OPER_AGNSS_S1A_ADGS_20220308T155208_V20220307T000000_20220308T000000_18_00.xml
[ INFO] 2022-03-08 15:52:09 NODE_1.odata - [DEC_259] I/F DHUS: downloading S1A_OPER_AUX_RESORB_OPOD_20220307T005502_V20220306T205322_20220307T001052 / 559.71 kB
##################################################################################################################################################################################################################################################
[ INFO] 2022-03-08 15:52:11 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S1A_OPER_AUX_RESORB_OPOD_20220307T005502_V20220306T205322_20220307T001052 / 559.71 kB / 0.39 MiB/s
[ INFO] 2022-03-08 15:52:11 NODE_1.odata - [DEC_259] I/F DHUS: downloading S1A_OPER_AUX_PROQUA_POD__20220307T021956_V20220304T235942_20220305T235941 / 1.57 MB
##################################################################################################################################################################################################################################################
[ INFO] 2022-03-08 15:52:11 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S1A_OPER_AUX_PROQUA_POD__20220307T021956_V20220304T235942_20220305T235941 / 1.57 MB / 5.49 MiB/s
[ INFO] 2022-03-08 15:52:11 NODE_1.odata - [DEC_259] I/F DHUS: downloading S1A_OPER_AUX_GNSSRD_POD__20220307T020223_V20220227T235946_20220228T235936 / 2.65 MB
##################################################################################################################################################################################################################################################

(...)

[ INFO] 2022-03-08 15:52:27 NODE_1.odata - [DEC_259] I/F DHUS: downloading S1A_OPER_AUX_RESORB_OPOD_20220307T200933_V20220307T163817_20220307T195547 / 559.57 kB
##################################################################################################################################################################################################################################################
[ INFO] 2022-03-08 15:52:28 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S1A_OPER_AUX_RESORB_OPOD_20220307T200933_V20220307T163817_20220307T195547 / 559.57 kB / 0.47 MiB/s
[ INFO] 2022-03-08 15:52:28 NODE_1.odata - [DEC_259] I/F DHUS: downloading S1A_OPER_AUX_RESORB_OPOD_20220307T221710_V20220307T181701_20220307T213431 / 559.91 kB
##################################################################################################################################################################################################################################################
[ INFO] 2022-03-08 15:52:29 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S1A_OPER_AUX_RESORB_OPOD_20220307T221710_V20220307T181701_20220307T213431 / 559.91 kB / 0.47 MiB/s
```

F.3.2 Sentinel-1

F.3.2.1 Selection by Catalogue Time

This test is demonstrating the selection and retrieval (i.e. pull circulation) of Sentinel-1 products (cf. S2-PFS) which have been published by the COAH during a certain time period.

```
$> decTestInterface_OData_DHUS -n test_download_s1a_updates
```

```
[DEBUG] 2022-02-21 18:30:31 NODE_1.test - DECTestInterfaceDHUS::test_download_s1a_updates START
[DEBUG] 2022-02-21 18:30:31 NODE_1.test - rm -f /tmp/S1A*
[DEBUG] 2022-02-21 18:30:32 NODE_1.test - decODataClient -u ******** -p ******** -c "2022-02-21T16:30:00.000,2022-02-21T16:32:00.000" -q dhus:s1:S1A -L /tmp -r
[ INFO] 2022-02-21 18:30:34 NODE_1.odata - [DEC_257] I/F DHUS: created DEC_OPER_OPDHUS_S1A_ADGS_20220221T183033_V20220221T163000_20220221T163200_1_0.xml
[ INFO] 2022-02-21 18:30:34 NODE_1.odata - [DEC_259] I/F DHUS: downloading S1A_IW_SLC__1SDV_20220221T153011_20220221T153024_042013_050121_4148 / 1.91 GB
##################################################################################################################################################################################################################################################

[ INFO] 2022-02-21 18:33:49 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S1A_IW_SLC__1SDV_20220221T153011_20220221T153024_042013_050121_4148 / 1.91 GB / 9.83 MiB/s
[DEBUG] 2022-02-21 18:33:49 NODE_1.test - DECTestInterfaceDHUS::test_download_s1a_updates END
teardown
.
Finished in 202.514565 seconds.
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
1 tests, 1 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
--------------------------------------------------------------------------------------------------------------
```

F.3.3 Sentinel-2

F.3.3.1 Selection by Catalogue Time

This test is demonstrating the selection and retrieval (i.e. pull circulation) of Sentinel-2 products (cf. S2-PFS) which have been published by the COAH during a certain time period.

```
$> decTestInterface_OData_DHUS -n test_download_s2a_updates
```

```
[DEBUG] 2022-02-18 10:17:28 NODE_1.test - DECTestInterfaceDHUS::test_download_s2a START
[DEBUG] 2022-02-18 10:17:28 NODE_1.test - rm -f /tmp/S2A*
[DEBUG] 2022-02-18 10:17:28 NODE_1.test - decODataClient -u ******** -p ******** -q dhus:s2:S2A --creation "2021-07-01T00:00:00.000,2021-07-01T00:05:03.000" -L /tmp -r
[ INFO] 2022-02-18 10:17:30 NODE_1.odata - [DEC_257] I/F DHUS: created DEC_OPER_OPDHUS_S2A_ADGS_20220218T101729_V20210701T000000_20210701T000503_3_0.xml
[ INFO] 2022-02-18 10:17:30 NODE_1.odata - [DEC_259] I/F DHUS: downloading S2A_MSIL1C_20210630T174911_N0301_R141_T12QUM_20210630T231156 / 103.90 MB
```

```
[ INFO] 2022-02-18 10:18:53 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S2A_MSIL1C_20210630T174911_N0301_R141_T12QUM_20210630T231156 / 103.90 MB / 1.26 MiB/s
[ INFO] 2022-02-18 10:18:53 NODE_1.odata - [DEC_259] I/F DHUS: downloading S2A_MSIL1C_20210630T174911_N0301_R141_T12SXB_20210630T231156 / 24.90 MB

[ INFO] 2022-02-18 10:19:13 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S2A_MSIL1C_20210630T174911_N0301_R141_T12SXB_20210630T231156 / 24.90 MB / 1.25 MiB/s
[ INFO] 2022-02-18 10:19:13 NODE_1.odata - [DEC_259] I/F DHUS: downloading S2A_MSIL1C_20210630T174911_N0301_R141_T13RBL_20210630T231156 / 14.25 MB

[ INFO] 2022-02-18 10:19:25 NODE_1.odata - [DEC_260] I/F DHUS: downloaded S2A_MSIL1C_20210630T174911_N0301_R141_T13RBL_20210630T231156 / 14.25 MB / 1.2 MiB/s
[DEBUG] 2022-02-18 10:19:25 NODE_1.test - DECTestInterfaceDHUS::test_download_s2a END
teardown
.
Finished in 121.409788 seconds.
1 tests, 1 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
0.01 tests/s, 0.01 assertions/s
End of DEC tests 4 DHUS
```

F.3.4 Sentinel-5P

F.3.4.1 Selection by Catalogue Time

This test is demonstrating the selection and retrieval (i.e. pull circulation) of Sentinel-5P NetCDF level-2 products (cf. S5P-PA) which have been published by the COAH during a certain time period.

```
$> decTestInterface_OData_DHUS -n test_download_s5p_updates
```

```
[DEBUG] 2022-03-08 15:36:51 NODE_1.test - DECTestInterfaceDHUS::test_download_s5p_updates START
[DEBUG] 2022-03-08 15:36:51 NODE_1.test - rm -f /tmp/S5*
[DEBUG] 2022-03-08 15:36:51 NODE_1.test - decODataClient -u ******** -p ******** -c "2022-03-08T13:36:00.000,2022-03-08T13:56:00.000" -q dhus_s5p:s5:S5P_NRTI_L2 -L /tmp
[ INFO] 2022-03-08 15:36:52 NODE_1.odata - [DEC_257] I/F DHUS_S5P: created DEC_OPER_OPDHUS_S5P_NRTI_L2_ADGS_20220308T153651_V20220308T133600_20220308T135600_13_00.xml
[DEBUG] 2022-03-08 15:36:57 NODE_1.test - decODataClient -u ******** -p ******** -c "2022-03-08T13:36:00.000,2022-03-08T13:56:00.000" -q dhus_s5p:s5:S5P_NRTI_L2 -L /tmp -r
[ INFO] 2022-03-08 15:36:59 NODE_1.odata - [DEC_257] I/F DHUS_S5P: created DEC_OPER_OPDHUS_S5P_NRTI_L2_ADGS_20220308T153658_V20220308T133600_20220308T135600_13_00.xml
[ INFO] 2022-03-08 15:36:59 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_AI_20220308T122157_20220308T122657_22801_02_020301_20220308T132819 / 1.18 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:37:03 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_AI_20220308T122157_20220308T122657_22801_02_020301_20220308T132819 / 1.18 MB / 0.3 MiB/s
[ INFO] 2022-03-08 15:37:03 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_AI_20220308T114157_20220308T114657_22800_02_020301_20220308T132516 / 1.68 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:37:10 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_AI_20220308T114157_20220308T114657_22800_02_020301_20220308T132516 / 1.68 MB / 0.23 MiB/s
[ INFO] 2022-03-08 15:37:10 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_AI_20220308T123157_20220308T123657_22801_02_020301_20220308T133043 / 17.73 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:37:45 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_AI_20220308T123157_20220308T123657_22801_02_020301_20220308T133043 / 17.73 MB / 0.51 MiB/s
[ INFO] 2022-03-08 15:38:26 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_AI_20220308T124657_20220308T125157_22801_02_020301_20220308T133531 / 18.46 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:39:06 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_AI_20220308T124657_20220308T125157_22801_02_020301_20220308T133531 / 18.46 MB / 0.45 MiB/s
[ INFO] 2022-03-08 15:39:06 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_AI_20220308T123657_20220308T124157_22801_02_020301_20220308T133503 / 17.58 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:39:46 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_AI_20220308T123657_20220308T124157_22801_02_020301_20220308T133503 / 17.58 MB / 0.44 MiB/s
[ INFO] 2022-03-08 15:39:46 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_AI_20220308T125657_20220308T130157_22801_02_020301_20220308T133755 / 18.30 MB
######################################################################## 100.0%curl: S

(...)

[ INFO] 2022-03-08 15:41:35 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_AI_20220308T125157_20220308T125657_22801_02_020301_20220308T133654 / 18.36 MB / 0.37 MiB/s
[ INFO] 2022-03-08 15:41:35 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_LH_20220308T113657_20220308T114157_22800_02_020301_20220308T132756 / 4.90 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:41:53 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_LH_20220308T113657_20220308T114157_22800_02_020301_20220308T132756 / 4.90 MB / 0.28 MiB/s
[ INFO] 2022-03-08 15:41:53 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_AI_20220308T124157_20220308T124657_22801_02_020301_20220308T133432 / 17.80 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:42:42 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_AI_20220308T124157_20220308T124657_22801_02_020301_20220308T133432 / 17.80 MB / 0.36 MiB/s
[ INFO] 2022-03-08 15:42:42 NODE_1.odata - [DEC_259] I/F DHUS_S5P: downloading S5P_NRTI_L2__AER_LH_20220308T114157_20220308T114657_22800_02_020301_20220308T132624 / 1.68 MB
######################################################################## 100.0%curl: S

[ INFO] 2022-03-08 15:42:49 NODE_1.odata - [DEC_260] I/F DHUS_S5P: downloaded S5P_NRTI_L2__AER_LH_20220308T114157_20220308T114657_22800_02_020301_20220308T132624 / 1.68 MB / 0.22 MiB/s
S5P_NRTI_L2__AER_LH_20220308T122157_20220308T122657_22801_02_020301_20220308T132926 / 1.19 MB / 0.26 MiB/s
[DEBUG] 2022-03-08 15:42:54 NODE_1.test - DECTestInterfaceDHUS::test_download_s5p_updates END
teardown
.
Finished in 368.136018 seconds.
1 tests, 2 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
0.00 tests/s, 0.01 assertions/s
End of DEC tests 4 DHUS
```

F.4 European Center for Disease Prevention and Control

The European Center for Disease Prevention and Control (ECDC) publishes some analytic regarding the Covid-19 pandemic distribution across the World. The interface is publishing a CSV resource containing the data using HTTP.

```
$> decTestInterface_ECDC
```

```
====================================
DEC Interface ECDC http / catalogue Unit Test Cases

decCheckConfig -e ECDC
[DEBUG] [DEC_XXX] I/F ECDC: File csv is available
[ INFO] [DEC_004] I/F ECDC: exchange point is reachable
[DEBUG] NODE_1.chck - Checking DEC/Inventory entries ...
[ INFO] NODE_1.chck - [DEC_003] I/F ECDC: interface is correctly declared in DEC/Inventory
decGetFromInterface --mnemonic ECDC -l
[ INFO] NODE_1.pull - [DEC_005] I/F ECDC: Polling Started - List mode is true
[ INFO] NODE_1.pull - [DEC_105] I/F ECDC: File csv is available
[ INFO] NODE_1.pull - [DEC_060] I/F ECDC: Polling Completed / New file(s) 1 available for pull
[ INFO] NODE_1.pull - [DEC_100] I/F ECDC: Pull iteration completed successfully
decGetFromInterface --mnemonic ECDC
[ INFO] NODE_1.pull - [DEC_005] I/F ECDC: Polling Started - List mode is false
[ INFO] NODE_1.pull - [DEC_060] I/F ECDC: Polling Completed / New file(s) 1 available for pull
[ INFO] NODE_1.pull - [DEC_110] I/F ECDC: csv downloaded with size 4307302 bytes
[ INFO] NODE_1.pull - [DEC_130] I/F ECDC: event triggered onreceivenewfilesok => mv /tmp/dec/in_basket_if_ecdc/csv /tmp/dec/in_basket_if_ecdc/"$(date +"%Y%m%d_covid-19.csv")"
[ INFO] NODE_1.pull - [DEC_130] I/F ECDC: event completed onreceivenewfilesok => mv /tmp/dec/in_basket_if_ecdc/csv /tmp/dec/in_basket_if_ecdc/"$(date +"%Y%m%d_covid-19.csv")"
[ INFO] NODE_1.pull - [DEC_135] I/F ECDC: S2__OPER_DEC_F_RECV_2BOA_20200715T110144_V20200715T110144_20200715T110144_ECDC.xml pull report created
[ INFO] NODE_1.pull - [DEC_100] I/F ECDC: Pull iteration completed successfully
.
Finished in 98.635273 seconds.
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
2 tests, 3 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
0.02 tests/s, 0.03 assertions/s
End of DECTestCase_Interface_ECDC
```

F.5 International Earth Rotation Service

F.5.1 Bulletins A & B

The IERS bulletins A & B with EOP data are grouped into the same interface test since they are retrieved from the same publishing point.

```
$> decTestInterface_IERS -n test_IERS_Bulletin_A_B
```

```
decGetFromInterface -m IERS -l --nodb
[ INFO] 2022-02-16 18:14:39 NODE_1.pull - [DEC_005] I/F IERS: Polling Started - List mode is true
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_105] I/F IERS: File finals.all is available
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_105] I/F IERS: File finals.data is available
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_105] I/F IERS: File finals2000A.data is available
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_105] I/F IERS: File gpsrapid.out is available
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_105] I/F IERS: File finals.daily is available
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_105] I/F IERS: File finals2000A.daily is available
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_105] I/F IERS: File gpsrapid.daily is available
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_060] I/F IERS: Polling Completed / New file(s) 7 available for pull
[ INFO] 2022-02-16 18:14:40 NODE_1.pull - [DEC_100] I/F IERS: Pull iteration completed successfully
decGetFromInterface -m IERS --nodb
[ INFO] 2022-02-16 18:14:44 NODE_1.pull - [DEC_005] I/F IERS: Polling Started - List mode is false
[ INFO] 2022-02-16 18:14:45 NODE_1.pull - [DEC_060] I/F IERS: Polling Completed / New file(s) 7 available for pull
[ INFO] 2022-02-16 18:14:46 NODE_1.pull - [DEC_110] I/F IERS: finals.daily downloaded with size 34028 bytes
[ INFO] 2022-02-16 18:14:46 NODE_1.pull - [DEC_110] I/F IERS: gpsrapid.out downloaded with size 936797 bytes
[ INFO] 2022-02-16 18:14:46 NODE_1.pull - [DEC_115] Intray GPS: gpsrapid.out disseminated into /tmp/dec_local_dissemination/GPS
[ INFO] 2022-02-16 18:14:46 NODE_1.pull - [DEC_130] I/F IERS: event triggered newfile2intray => echo NEW FILE TO INTRAY
NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:46 NODE_1.pull - [DEC_130] I/F IERS: event completed newfile2intray => echo NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:46 NODE_1.pull - [DEC_110] I/F IERS: finals.data downloaded with size 2147336 bytes
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_110] I/F IERS: finals2000A.data downloaded with size 2147336 bytes
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_110] I/F IERS: finals.all downloaded with size 3451680 bytes
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_110] I/F IERS: finals2000A.daily downloaded with size 34028 bytes
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_115] Intray S3_IN: finals2000A.daily disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_130] I/F IERS: event triggered newfile2intray => echo NEW FILE TO INTRAY
NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_130] I/F IERS: event completed newfile2intray => echo NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_110] I/F IERS: gpsrapid.daily downloaded with size 9029 bytes
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_115] Intray GPS: gpsrapid.daily disseminated into /tmp/dec_local_dissemination/GPS
[ INFO] 2022-02-16 18:14:47 NODE_1.pull - [DEC_130] I/F IERS: event triggered newfile2intray => echo NEW FILE TO INTRAY
NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event completed newfile2intray => echo NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_116] I/F IERS: finals.all compressed in 7z at with size 538436 bytes
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_115] Intray S2A: finals.7z disseminated into /tmp/dec_local_dissemination/S2A
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_115] Intray GPS: finals.all disseminated into /tmp/dec_local_dissemination/GPS
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event triggered newfile2intray => echo NEW FILE TO INTRAY
NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event completed newfile2intray => echo NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_115] Intray S2B: finals.all disseminated into /tmp/dec_local_dissemination/S2B
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event triggered newfile2intray => echo NEW FILE TO INTRAY
NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event completed newfile2intray => echo NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_115] Intray S2_7Z: finals.7z disseminated into /tmp/dec_local_dissemination/S2_7Z
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event triggered newfile2intray => echo NEW FILE TO INTRAY
NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event completed newfile2intray => echo NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_115] Intray S2ALL: finals.all disseminated into /tmp/dec_local_dissemination/S2ALL
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event triggered newfile2intray => echo NEW FILE TO INTRAY
NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event completed newfile2intray => echo NEW FILE TO INTRAY
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event triggered onreceivenewfilesok => echo \':-\)\'
':-)'
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_130] I/F IERS: event completed onreceivenewfilesok => echo \':-\)\'
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_135] I/F IERS: S2__OPER_DEC_F_RECV_2BOA_20220216T181448_V20220216T181448_20220216T181448_IERS.xml pull report created
[ INFO] 2022-02-16 18:14:48 NODE_1.pull - [DEC_100] I/F IERS: Pull iteration completed successfully

================================================
decTestInterface_IERS::teardown

decManageDB -d
-- drop_table(:received_files)
-> 0.0018s
-- drop_table(:tracked_files)
-> 0.0013s
-- drop_table(:sent_files)
-> 0.0013s
-- drop_table(:interfaces)
-> 0.0011s
.
Finished in 17.493844 seconds.
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
1 tests, 6 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
```

F.5.2 Bulletin C

The IERS bulletin C contains the leap-second information, which is retrieved from Paris Observatory hosted by SYRTE (cf. IERS-P). This test converts the BULC filename pulled into the Sentinel-3 file format conventions using the external component *auxConverter* tool.

```
$> decTestInterface_IERS -n test_IERS_S3_Bulletin_C
```

```
decGetFromInterface -m LEAP -l
[ INFO] 2022-02-16 18:19:14 NODE_1.pull - [DEC_005] I/F LEAP: Polling Started - List mode is true
[ INFO] 2022-02-16 18:19:14 NODE_1.pull - [DEC_105] I/F LEAP: File Leap_Second.dat is available
[ INFO] 2022-02-16 18:19:14 NODE_1.pull - [DEC_060] I/F LEAP: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-02-16 18:19:14 NODE_1.pull - [DEC_100] I/F LEAP: Pull iteration completed successfully
decGetFromInterface -m LEAP --nodb
[ INFO] 2022-02-16 18:19:21 NODE_1.pull - [DEC_005] I/F LEAP: Polling Started - List mode is false
[ INFO] 2022-02-16 18:19:21 NODE_1.pull - [DEC_060] I/F LEAP: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-02-16 18:19:21 NODE_1.pull - [DEC_110] I/F LEAP: Leap_Second.dat downloaded with size 1359 bytes
[ INFO] 2022-02-16 18:19:21 NODE_1.pull - [DEC_115] Intray S3_IN: Leap_Second.dat disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-16 18:19:21 NODE_1.pull - [DEC_130] I/F LEAP: event triggered newfile2intray => auxConverter -f /tmp/dec_local_dissemination/S3/Leap_Second.dat -d /tmp/dec/publish/s3/
S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3
[ INFO] 2022-02-16 18:19:22 NODE_1.pull - [DEC_130] I/F LEAP: event completed newfile2intray => auxConverter -f /tmp/dec_local_dissemination/S3/Leap_Second.dat -d /tmp/dec/publish/s3/
[ INFO] 2022-02-16 18:19:22 NODE_1.pull - [DEC_135] I/F LEAP: S2__OPER_DEC_F_RECV_2BOA_20220216T181922_V20220216T181922_20220216T181922_LEAP.xml pull report created
[ INFO] 2022-02-16 18:19:22 NODE_1.pull - [DEC_100] I/F LEAP: Pull iteration completed successfully
decGetFromInterface -m PUBLISH_S3 -l
[ INFO] 2022-02-16 18:19:29 NODE_1.pull - [DEC_005] I/F PUBLISH_S3: Polling Started - List mode is true
[ INFO] 2022-02-16 18:19:29 NODE_1.pull - [DEC_050] I/F PUBLISH_S3: Polling Completed / No file(s) available for pull
[ INFO] 2022-02-16 18:19:29 NODE_1.pull - [DEC_100] I/F PUBLISH_S3: Pull iteration completed successfully
decGetFromInterface -m PUBLISH_S3
[ INFO] 2022-02-16 18:19:31 NODE_1.pull - [DEC_005] I/F PUBLISH_S3: Polling Started - List mode is false
[ INFO] 2022-02-16 18:19:31 NODE_1.pull - [DEC_050] I/F PUBLISH_S3: Polling Completed / No file(s) available for pull
[ INFO] 2022-02-16 18:19:31 NODE_1.pull - [DEC_115] Intray FINAL_S3: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 disseminated into /tr
[ INFO] 2022-02-16 18:19:31 NODE_1.pull - [DEC_115] Intray ARCHIVE: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 disseminated into /tmp
[ INFO] 2022-02-16 18:19:31 NODE_1.pull - [DEC_100] I/F PUBLISH_S3: Pull iteration completed successfully
decGetFromInterface -m LEAP --nodb
[ INFO] 2022-02-16 18:19:33 NODE_1.pull - [DEC_005] I/F LEAP: Polling Started - List mode is false
[ INFO] 2022-02-16 18:19:33 NODE_1.pull - [DEC_060] I/F LEAP: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-02-16 18:19:34 NODE_1.pull - [DEC_110] I/F LEAP: Leap_Second.dat downloaded with size 1359 bytes
[ INFO] 2022-02-16 18:19:34 NODE_1.pull - [DEC_115] Intray S3_IN: Leap_Second.dat disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-16 18:19:34 NODE_1.pull - [DEC_130] I/F LEAP: event triggered newfile2intray => auxConverter -f /tmp/dec_local_dissemination/S3/Leap_Second.dat -d /tmp/dec/publish/s3/
S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3
[ INFO] 2022-02-16 18:19:34 NODE_1.pull - [DEC_130] I/F LEAP: event completed newfile2intray => auxConverter -f /tmp/dec_local_dissemination/S3/Leap_Second.dat -d /tmp/dec/publish/s3/
[ INFO] 2022-02-16 18:19:34 NODE_1.pull - [DEC_135] I/F LEAP: S2__OPER_DEC_F_RECV_2BOA_20220216T181934_V20220216T181934_20220216T181934_LEAP.xml pull report created
[ INFO] 2022-02-16 18:19:34 NODE_1.pull - [DEC_100] I/F LEAP: Pull iteration completed successfully
decGetFromInterface -m PUBLISH_S3 -l
[ INFO] 2022-02-16 18:19:41 NODE_1.pull - [DEC_005] I/F PUBLISH_S3: Polling Started - List mode is true
[ INFO] 2022-02-16 18:19:41 NODE_1.pull - [DEC_050] I/F PUBLISH_S3: Polling Completed / No file(s) available for pull
[ INFO] 2022-02-16 18:19:41 NODE_1.pull - [DEC_100] I/F PUBLISH_S3: Pull iteration completed successfully
decGetFromInterface -m PUBLISH_S3
[ INFO] 2022-02-16 18:19:43 NODE_1.pull - [DEC_005] I/F PUBLISH_S3: Polling Started - List mode is false
[ INFO] 2022-02-16 18:19:43 NODE_1.pull - [DEC_050] I/F PUBLISH_S3: Polling Completed / No file(s) available for pull
[ WARN] 2022-02-16 18:19:43 NODE_1.pull - [DEC_555_1] Intray FINAL_S3: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 duplicated already
[ WARN] 2022-02-16 18:19:43 NODE_1.pull - [DEC_556_1] Intray FINAL_S3: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 duplicated will be
[ INFO] 2022-02-16 18:19:43 NODE_1.pull - [DEC_115] Intray FINAL_S3: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 disseminated into /tr
[ WARN] 2022-02-16 18:19:43 NODE_1.pull - [DEC_555] Intray ARCHIVE: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 duplicated already ex
[ WARN] 2022-02-16 18:19:43 NODE_1.pull - [DEC_556] Intray ARCHIVE: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 duplicated will be de
[ INFO] 2022-02-16 18:19:43 NODE_1.pull - [DEC_115] Intray ARCHIVE: S3__GN_1_LSC_AX_19720101T000000_20221228T000000_20220216T000000____________________USN_O_NR_POD.SEN3 disseminated into /tmp
[ INFO] 2022-02-16 18:19:43 NODE_1.pull - [DEC_100] I/F PUBLISH_S3: Pull iteration completed successfully
grep ERROR /tmp/DEC000001.log

================================================
decTestInterface_IERS::teardown

decManageDB -d
-- drop_table(:received_files)
-> 0.0021s
-- drop_table(:tracked_files)
-> 0.0014s
-- drop_table(:sent_files)
-> 0.0012s
-- drop_table(:interfaces)
-> 0.0011s
.
Finished in 35.01586 seconds.
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
1 tests, 9 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
0.03 tests/s, 0.26 assertions/s
```

F.6 International GNSS Service

F.6.1 GNSS Broadcast Ephemeris

GNSS Broadcast ephemeris files RINEX2 merged GPS broadcast ephemeris file are pulled from an ESA server by the GSSC.

```
$> decTestInterface_GNSS
```

```
[DEBUG] 2022-02-24 12:07:57 NODE_1.test - DECTestCaseGNSS::setup
[DEBUG] 2022-02-24 12:07:58 NODE_1.test - DECTestCaseGNSS::test_pull => IGS noDB false
[DEBUG] 2022-02-24 12:08:00 NODE_1.test - decGetFromInterface -m IGS -l
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_005] I/F IGS: Polling Started - List mode is true
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0010.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0020.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0030.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0040.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0050.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0070.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0060.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0080.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0090.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0100.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0110.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0120.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0130.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0140.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0150.22n.gz is available

(...)


[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_105] I/F IGS: File brdc0550.22n.gz is available
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_060] I/F IGS: Polling Completed / New file(s) 55 available for pull
[ INFO] 2022-02-24 12:08:03 NODE_1.pull - [DEC_100] I/F IGS: Pull iteration completed successfully
[DEBUG] 2022-02-24 12:08:08 NODE_1.test - decGetFromInterface -m IGS
[ INFO] 2022-02-24 12:08:11 NODE_1.pull - [DEC_005] I/F IGS: Polling Started - List mode is false
[ INFO] 2022-02-24 12:08:11 NODE_1.pull - [DEC_060] I/F IGS: Polling Completed / New file(s) 55 available for pull
[ INFO] 2022-02-24 12:08:12 NODE_1.pull - [DEC_110] I/F IGS: brdc0010.22n.gz downloaded with size 58866 bytes
[ INFO] 2022-02-24 12:08:12 NODE_1.pull - [DEC_110] I/F IGS: brdc0020.22n.gz downloaded with size 57460 bytes
[ INFO] 2022-02-24 12:08:12 NODE_1.pull - [DEC_110] I/F IGS: brdc0030.22n.gz downloaded with size 58051 bytes
[ INFO] 2022-02-24 12:08:12 NODE_1.pull - [DEC_110] I/F IGS: brdc0040.22n.gz downloaded with size 58243 bytes
[ INFO] 2022-02-24 12:08:13 NODE_1.pull - [DEC_110] I/F IGS: brdc0050.22n.gz downloaded with size 58116 bytes
[ INFO] 2022-02-24 12:08:13 NODE_1.pull - [DEC_110] I/F IGS: brdc0070.22n.gz downloaded with size 57721 bytes

(...)

[ INFO] 2022-02-24 12:08:25 NODE_1.pull - [DEC_110] I/F IGS: brdc0530.22n.gz downloaded with size 57248 bytes
[ INFO] 2022-02-24 12:08:25 NODE_1.pull - [DEC_110] I/F IGS: brdc0540.22n.gz downloaded with size 57515 bytes
[ INFO] 2022-02-24 12:08:25 NODE_1.pull - [DEC_110] I/F IGS: brdc0550.22n.gz downloaded with size 29419 bytes
[ INFO] 2022-02-24 12:08:25 NODE_1.pull - [DEC_135] I/F IGS: S2__OPER_DEC_F_RECV_2BOA_20220224T120825_V20220224T120825_20220224T120825_IGS.xml pull report created
[ INFO] 2022-02-24 12:08:25 NODE_1.pull - [DEC_100] I/F IGS: Pull iteration completed successfully
[DEBUG] 2022-02-24 12:08:30 NODE_1.test - decStats -H 1
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0010.22n.gz","interface":"IGS","date":"2022-02-24 11:08:12 UTC","protocol":"FTP","size":58866}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0020.22n.gz","interface":"IGS","date":"2022-02-24 11:08:12 UTC","protocol":"FTP","size":57460}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0030.22n.gz","interface":"IGS","date":"2022-02-24 11:08:12 UTC","protocol":"FTP","size":58051}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0040.22n.gz","interface":"IGS","date":"2022-02-24 11:08:12 UTC","protocol":"FTP","size":58243}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0050.22n.gz","interface":"IGS","date":"2022-02-24 11:08:13 UTC","protocol":"FTP","size":58116}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0070.22n.gz","interface":"IGS","date":"2022-02-24 11:08:13 UTC","protocol":"FTP","size":57721}

(...)

[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0540.22n.gz","interface":"IGS","date":"2022-02-24 11:08:25 UTC","protocol":"FTP","size":57515}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"brdc0550.22n.gz","interface":"IGS","date":"2022-02-24 11:08:25 UTC","protocol":"FTP","size":29419}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_144] Pull stats: {"numFiles":55,"hours":1,"rate":"872.00 B/s","volume":"3.00 MiB"}
[ INFO] 2022-02-24 12:08:32 NODE_1.stat - [DEC_244] Push 1h: No files pushed
[DEBUG] 2022-02-24 12:08:32 NODE_1.test - decGetFromInterface -m IGS -l
[ INFO] 2022-02-24 12:08:34 NODE_1.pull - [DEC_005] I/F IGS: Polling Started - List mode is true
[ INFO] 2022-02-24 12:08:34 NODE_1.pull - [DEC_050] I/F IGS: Polling Completed / No file(s) available for pull
[ INFO] 2022-02-24 12:08:34 NODE_1.pull - [DEC_100] I/F IGS: Pull iteration completed successfully
[DEBUG] 2022-02-24 12:08:34 NODE_1.test - grep ERROR /tmp/DEC000001.log
[DEBUG] 2022-02-24 12:08:34 NODE_1.test - DECTestCaseGNSS::teardown
.
Finished in 37.71214 seconds.
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
1 tests, 6 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
0.03 tests/s, 0.16 assertions/s
[ INFO] 2022-02-24 12:08:34 NODE_1.test - End of DEC test for GNSS ADP interface
```

F.7 International Laser Ranging Service

The International Laser Ranging Service provides global satellite and lunar laser ranging data and their related products to support geodetic and geophysical research activities.

F.7.1 Sentinel-3

The Copernicus Sentinel-3 constellation are equipped with a laser retro reflector (cf. S3-LRR) to support altimetry science with laser ranging (cf. S3-SLR). Sentinel-3 ranging data in CRD is publicly published by Eurolas Data Center (cf. EDC).

```
$> decTestInterface_ILRS -n test_ILRS_S3
```

```
[DEBUG] 2022-02-21 11:07:43 NODE_1.test - DECTestCaseILRS::test_ILRS_S3 START
[DEBUG] 2022-02-21 11:07:43 NODE_1.test - decGetFromInterface -m ILRS -l
[ INFO] 2022-02-21 11:07:49 NODE_1.pull - [DEC_005] I/F ILRS: Polling Started - List mode is true
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_202201.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220101.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220102.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220103.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220104.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220105.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220106.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220107.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220108.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220109.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220110.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220111.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220112.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220113.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220114.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220115.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220116.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220117.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220118.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220119.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220120.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220121.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3a_20220122.npt is available

(...)

[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3b_20220219.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3b_20220220.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_105] I/F ILRS: File sentinel3b_20220221.npt is available
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_060] I/F ILRS: Polling Completed / New file(s) 108 available for pull
[ INFO] 2022-02-21 11:07:51 NODE_1.pull - [DEC_100] I/F ILRS: Pull iteration completed successfully
[DEBUG] 2022-02-21 11:07:53 NODE_1.test - decGetFromInterface -m ILRS
[ INFO] 2022-02-21 11:07:59 NODE_1.pull - [DEC_005] I/F ILRS: Polling Started - List mode is false
[ INFO] 2022-02-21 11:08:00 NODE_1.pull - [DEC_060] I/F ILRS: Polling Completed / New file(s) 108 available for pull
[ INFO] 2022-02-21 11:08:02 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3a_20220101.npt downloaded with size 46320 bytes
[ INFO] 2022-02-21 11:08:03 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3a_20220102.npt downloaded with size 35036 bytes
[ INFO] 2022-02-21 11:08:04 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3a_202201.npt downloaded with size 1321655 bytes
[ INFO] 2022-02-21 11:08:05 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3a_20220103.npt downloaded with size 47431 bytes
[ INFO] 2022-02-21 11:08:06 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3a_20220104.npt downloaded with size 18711 bytes
[ INFO] 2022-02-21 11:08:07 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3a_20220105.npt downloaded with size 54431 bytes
[ INFO] 2022-02-21 11:08:08 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3a_20220106.npt downloaded with size 45874 bytes

(...)

[ INFO] 2022-02-21 11:09:30 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3b_20220217.npt downloaded with size 67662 bytes
[ INFO] 2022-02-21 11:09:31 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3b_20220219.npt downloaded with size 37669 bytes
[ INFO] 2022-02-21 11:09:31 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3b_20220220.npt downloaded with size 18385 bytes
[ INFO] 2022-02-21 11:09:33 NODE_1.pull - [DEC_110] I/F ILRS: sentinel3b_20220221.npt downloaded with size 5605 bytes
[ INFO] 2022-02-21 11:09:33 NODE_1.pull - [DEC_130] I/F ILRS: event triggered onreceivenewfilesok => echo \':-\)\'
':-)'
[ INFO] 2022-02-21 11:09:33 NODE_1.pull - [DEC_130] I/F ILRS: event completed onreceivenewfilesok => echo \':-\)\'
[ INFO] 2022-02-21 11:09:33 NODE_1.pull - [DEC_135] I/F ILRS: S2__OPER_DEC_F_RECV_2BOA_20220221T110933_V20220221T110933_20220221T110933_ILRS.xml pull report created
[ INFO] 2022-02-21 11:09:33 NODE_1.pull - [DEC_100] I/F ILRS: Pull iteration completed successfully
[DEBUG] 2022-02-21 11:09:33 NODE_1.test - DECTestCaseILRS::test_ILRS_S3 END
```

F.8 NASA

NASA is a major public data provider covering several different domains such as earth observation / geophysical, navigation, space-weather, etc.

F.8.1 ASTER Global DEM

ASTER Global DEM (ASTGTM) are publicly available as part of NASA EOSDIS. DEC is able to perform bulk data retrievals in an unattended mode. For the sake of the test, DEC has been configured to retrieve the tiles metadata and quick-looks only.

```
$> decTestInterface_NASA -n testASTGTM
```

```
[ INFO] 2023-07-17 10:32:03.163 luuaplmds01.test - decGetFromInterface -m TEST_NASA_ASTGTM -l
[ INFO] 2023-07-17 10:32:08.162 luuaplmds01.pull - [DEC_005] I/F TEST_NASA_ASTGTM: Polling Started - List mode is true
[ INFO] 2023-07-17 10:34:02.702 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E006.1.jpg is available
[ INFO] 2023-07-17 10:34:02.703 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E006.zip.xml is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E009.1.jpg is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E009.zip.xml is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E010.1.jpg is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E010.zip.xml is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E011.1.jpg is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E011.zip.xml is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E012.1.jpg is available
[ INFO] 2023-07-17 10:34:02.704 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_N00E012.zip.xml is available
(...)
[ INFO] 2023-07-17 10:44:40.592 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_S83W157.1.jpg is available
[ INFO] 2023-07-17 10:44:40.592 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_ASTGTM: File ASTGTMV003_S83W157.zip.xml is available
[ INFO] 2023-07-17 10:44:40.592 luuaplmds01.pull - [DEC_060] I/F TEST_NASA_ASTGTM: Polling Completed / New file(s) 45824 available for pull
[ INFO] 2023-07-17 10:44:40.592 luuaplmds01.pull - [DEC_100] I/F TEST_NASA_ASTGTM: Pull iteration completed successfully
(...)
[ INFO] 2023-07-17 10:44:40.905 luuaplmds01.test - decGetFromInterface -m TEST_NASA_ASTGTM
[ INFO] 2023-07-17 10:44:45.122 luuaplmds01.pull - [DEC_005] I/F TEST_NASA_ASTGTM: Polling Started - List mode is false
[ INFO] 2023-07-17 10:46:42.401 luuaplmds01.pull - [DEC_060] I/F TEST_NASA_ASTGTM: Polling Completed / New file(s) 45824 available for pull
[ INFO] 2023-07-17 10:46:43.262 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E006.1.jpg downloaded with size 14925 bytes
[ INFO] 2023-07-17 10:46:43.688 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E010.1.jpg downloaded with size 113799 bytes
[ INFO] 2023-07-17 10:46:43.697 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E009.1.jpg downloaded with size 104509 bytes
[ INFO] 2023-07-17 10:46:44.480 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E011.1.jpg downloaded with size 102038 bytes
[ INFO] 2023-07-17 10:46:44.944 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E012.1.jpg downloaded with size 125479 bytes
[ INFO] 2023-07-17 10:46:46.196 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E013.1.jpg downloaded with size 182996 bytes
[ INFO] 2023-07-17 10:46:47.521 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E009.zip.xml downloaded with size 2851 bytes
[ INFO] 2023-07-17 10:46:47.688 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E010.zip.xml downloaded with size 2850 bytes
[ INFO] 2023-07-17 10:46:47.803 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E006.zip.xml downloaded with size 2849 bytes
[ INFO] 2023-07-17 10:46:48.186 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E012.zip.xml downloaded with size 2852 bytes
[ INFO] 2023-07-17 10:46:48.254 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E011.zip.xml downloaded with size 2852 bytes
[ INFO] 2023-07-17 10:46:48.758 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E014.1.jpg downloaded with size 129451 bytes
[ INFO] 2023-07-17 10:46:49.112 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E015.1.jpg downloaded with size 129684 bytes
[ INFO] 2023-07-17 10:46:49.240 luuaplmds01.pull - [DEC_110] I/F TEST_NASA_ASTGTM: ASTGTMV003_N00E014.zip.xml downloaded with size 2852 bytes
(...)
```

F.8.2 MODIS Products

MODIS products are publicly available as part of NASA EOSDIS. DEC is able to perform bulk data retrievals in an unattended mode.

```
$> decTestInterface_NASA -n testRFM
```

F.8.2.1 MOD09A1

MODIS L3 Surface Reflectance Product: HDF and XML metadata is managed ; the configuration also is polling the directories names in which the MOD09A1 products are published.

```
$> decTestInterface_NASA -n testRFM
```

```
[ INFO] 2023-07-17 10:02:27.007 luuaplmds01.test - decGetFromInterface -m TEST_NASA_RFM -l
[ INFO] 2023-07-17 10:02:31.728 luuaplmds01.pull - [DEC_005] I/F TEST_NASA_RFM: Polling Started - List mode is true
[ INFO] 2023-07-17 10:02:34.664 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_RFM: File MOD09A1.A2023169.h00v08.061.2023178033738.hdf is available
[ INFO] 2023-07-17 10:02:34.664 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_RFM: File MOD09A1.A2023169.h00v08.061.2023178033738.hdf.xml is available
[ INFO] 2023-07-17 10:02:34.664 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_RFM: File MOD09A1.A2023169.h00v09.061.2023178031908.hdf is available
```

```
[ INFO] 2023-07-17 10:02:34.664 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_RFM: File MOD09A1.A2023169.h00v09.061.2023178031908.hdf.xml is available
(...)
[ INFO] 2023-07-17 10:02:34.730 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_RFM: File 2023.06.10 is available
[ INFO] 2023-07-17 10:02:34.730 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_RFM: File 2023.06.18 is available
[ INFO] 2023-07-17 10:02:34.730 luuaplmds01.pull - [DEC_105] I/F TEST_NASA_RFM: File 2023.06.26 is available
[ INFO] 2023-07-17 10:02:34.730 luuaplmds01.pull - [DEC_060] I/F TEST_NASA_RFM: Polling Completed / New file(s) 771 available for pull
```

F.8.3 NASA Bulletin A

NASA CDDIS publishes Bulletin A Earth rotation parameters with a slight format different from IERS.

```
$> decTestInterface_NASA -n testNBULA
```

```
[ INFO] 2022-08-29 10:10:37.159 nl2-u-moc-srv-01.test - DECTestInterface_NASA_CDDIS::testNBULA START
[DEBUG] 2022-08-29 10:10:37.160 nl2-u-moc-srv-01.test - decConfigInterface2DB -a NASA_NBULA
[ INFO] 2022-08-29 10:10:38.162 nl2-u-moc-srv-01.db   - [DEC_001] I/F NASA_NBULA: added to the DEC Inventory/Interface db
[ INFO] 2022-08-29 10:10:38.172 nl2-u-moc-srv-01.test - decCheckConfig -e NASA_NBULA
[DEBUG] 2022-08-29 10:10:40.352 nl2-u-moc-srv-01.chck - [DEC_XXX] I/F NASA_NBULA: File finals2000A.data is available
[ INFO] 2022-08-29 10:10:40.358 nl2-u-moc-srv-01.chck - [DEC_004] I/F NASA_NBULA: exchange point is reachable
[ INFO] 2022-08-29 10:10:40.937 nl2-u-moc-srv-01.chck - [DEC_003] I/F NASA_NBULA: interface is correctly declared in DEC/Inventory
[ INFO] 2022-08-29 10:10:40.950 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_NBULA -l
[ INFO] 2022-08-29 10:10:42.770 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_NBULA: Polling Started - List mode is true
[ INFO] 2022-08-29 10:10:43.866 nl2-u-moc-srv-01.pull - [DEC_105] I/F NASA_NBULA: File finals2000A.data is available
[ INFO] 2022-08-29 10:10:43.866 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_NBULA: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:10:43.866 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_NBULA: Pull iteration completed successfully
[ INFO] 2022-08-29 10:10:43.882 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_NBULA
[ INFO] 2022-08-29 10:10:45.822 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_NBULA: Polling Started - List mode is false
[ INFO] 2022-08-29 10:10:46.910 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_NBULA: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:10:52.267 nl2-u-moc-srv-01.pull - [DEC_110] I/F NASA_NBULA: finals2000A.data downloaded with size 2184184 bytes
[ INFO] 2022-08-29 10:10:52.323 nl2-u-moc-srv-01.pull - [DEC_115] Intray NAOS_IN: finals2000A.data disseminated into /data/mocExternalInterfaces/tmp/naos/
[ INFO] 2022-08-29 10:10:52.324 nl2-u-moc-srv-01.pull - [DEC_130] I/F NASA_NBULA: event triggered newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/finals2000A.dat
[ INFO] 2022-08-29 10:10:52.655 nl2-u-moc-srv-01.auxc - [AUX_001] NS1_OPER_AUX_NBULA__20220826T000000_20230902T000000_0001.TXT generated from finals2000A.data
[ INFO] 2022-08-29 10:10:52.659 nl2-u-moc-srv-01.pull - [DEC_130] I/F NASA_NBULA: event completed newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/finals2000A.dat
[ INFO] 2022-08-29 10:10:52.698 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_NBULA: Pull iteration completed successfully
[ INFO] 2022-08-29 10:10:52.714 nl2-u-moc-srv-01.test - decStats
[ INFO] 2022-08-29 10:10:53.672 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"finals2000A.data","interface":"NASA_NBULA","date":"2022-08-29 08:10:52 UTC","protocol":"HTTP","size":218
[ INFO] 2022-08-29 10:10:53.673 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"606.00 B/s","volume":"2.08 MiB"}
[ INFO] 2022-08-29 10:10:53.676 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[ INFO] 2022-08-29 10:10:53.684 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_NBULA
[ INFO] 2022-08-29 10:10:55.821 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_NBULA: Polling Started - List mode is false
[ INFO] 2022-08-29 10:10:56.922 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_NBULA: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:10:59.668 nl2-u-moc-srv-01.pull - [DEC_111] I/F NASA_NBULA: finals2000A.data downloaded is duplicated / same md5
[ INFO] 2022-08-29 10:10:59.716 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_NBULA: Pull iteration completed successfully
[ INFO] 2022-08-29 10:10:59.732 nl2-u-moc-srv-01.test - decStats
[ INFO] 2022-08-29 10:11:00.967 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"finals2000A.data","interface":"NASA_NBULA","date":"2022-08-29 08:10:52 UTC","protocol":"HTTP","size":218
[ INFO] 2022-08-29 10:11:00.967 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"606.00 B/s","volume":"2.08 MiB"}
[ INFO] 2022-08-29 10:11:00.971 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[ INFO] 2022-08-29 10:11:00.982 nl2-u-moc-srv-01.test - DECTestInterface_NASA_CDDIS::testNBULA END
```

F.8.4 NASA Bulletin C

NASA CDDIS publishes Bulletin C leap second announcement with a slight format different from IERS.

```
$> decTestInterface_NASA -n testNBULC
```

```
[ INFO] 2022-08-29 10:28:42.405 nl2-u-moc-srv-01.test - DECTestInterface_NASA_CDDIS::testNBULC START
[DEBUG] 2022-08-29 10:28:42.405 nl2-u-moc-srv-01.test - decConfigInterface2DB -a NASA_NBULC
[ INFO] 2022-08-29 10:28:43.475 nl2-u-moc-srv-01.db   - [DEC_001] I/F NASA_NBULC: added to the DEC Inventory/Interface db
[ INFO] 2022-08-29 10:28:43.487 nl2-u-moc-srv-01.test - decCheckConfig -e NASA_NBULC
[DEBUG] 2022-08-29 10:28:46.214 nl2-u-moc-srv-01.chck - [DEC_XXX] I/F NASA_NBULC: File tai-utc.dat is available
[ INFO] 2022-08-29 10:28:46.257 nl2-u-moc-srv-01.chck - [DEC_004] I/F NASA_NBULC: exchange point is reachable
[ INFO] 2022-08-29 10:28:46.897 nl2-u-moc-srv-01.chck - [DEC_003] I/F NASA_NBULC: interface is correctly declared in DEC/Inventory
[ INFO] 2022-08-29 10:28:46.911 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_NBULC -l
[ INFO] 2022-08-29 10:28:48.871 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_NBULC: Polling Started - List mode is true
[ INFO] 2022-08-29 10:28:49.943 nl2-u-moc-srv-01.pull - [DEC_105] I/F NASA_NBULC: File tai-utc.dat is available
[ INFO] 2022-08-29 10:28:49.943 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_NBULC: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:28:49.944 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_NBULC: Pull iteration completed successfully
[ INFO] 2022-08-29 10:28:49.960 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_NBULC
[ INFO] 2022-08-29 10:28:51.963 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_NBULC: Polling Started - List mode is false
[ INFO] 2022-08-29 10:28:53.038 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_NBULC: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:28:54.169 nl2-u-moc-srv-01.pull - [DEC_110] I/F NASA_NBULC: tai-utc.dat downloaded with size 3321 bytes
[ INFO] 2022-08-29 10:28:54.238 nl2-u-moc-srv-01.pull - [DEC_115] Intray NAOS_IN: tai-utc.dat disseminated into /data/mocExternalInterfaces/tmp/naos/
[ INFO] 2022-08-29 10:28:54.239 nl2-u-moc-srv-01.pull - [DEC_130] I/F NASA_NBULC: event triggered newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/tai-utc.dat -d
[ INFO] 2022-08-29 10:28:54.582 nl2-u-moc-srv-01.auxc - [AUX_001] NS1_OPER_AUX_NBULC__20170101T000000_21000101T000000_0001.TXT generated from tai-utc.dat
[ INFO] 2022-08-29 10:28:54.586 nl2-u-moc-srv-01.pull - [DEC_130] I/F NASA_NBULC: event completed newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/tai-utc.dat -d
[ INFO] 2022-08-29 10:28:54.625 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_NBULC: Pull iteration completed successfully
[ INFO] 2022-08-29 10:28:54.643 nl2-u-moc-srv-01.test - decStats
```

```
[ INFO] 2022-08-29 10:28:55.630 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"tai-utc.dat","interface":"NASA_NBULC","date":"2022-08-29 08:28:54 UTC","protocol":"HTTP","size":3321,"m
[ INFO] 2022-08-29 10:28:55.630 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"0.00 B/s","volume":"3.24 KiB"}
[ INFO] 2022-08-29 10:28:55.633 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[ INFO] 2022-08-29 10:28:55.642 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_NBULC
[ INFO] 2022-08-29 10:28:57.541 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_NBULC: Polling Started - List mode is false
[ INFO] 2022-08-29 10:28:58.658 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_NBULC: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:28:59.768 nl2-u-moc-srv-01.pull - [DEC_111] I/F NASA_NBULC: tai-utc.dat downloaded is duplicated / same md5
[ INFO] 2022-08-29 10:28:59.806 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_NBULC: Pull iteration completed successfully
[ INFO] 2022-08-29 10:28:59.824 nl2-u-moc-srv-01.test - decStats
[ INFO] 2022-08-29 10:29:00.812 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"tai-utc.dat","interface":"NASA_NBULC","date":"2022-08-29 08:28:54 UTC","protocol":"HTTP","size":3321,"m
[ INFO] 2022-08-29 10:29:00.812 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"0.00 B/s","volume":"3.24 KiB"}
[ INFO] 2022-08-29 10:29:00.815 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[ INFO] 2022-08-29 10:29:00.828 nl2-u-moc-srv-01.test - DECTestInterface_NASA_CDDIS::testNBULC END
```

F.8.5 Space-weather F10.7 & Ap Index

NASA MSFC publishes Bulletin C leap second announcement with statistical index of solar radio 10.7 cm flux (F10.7), and the geomagnetic planetary index, Ap.

```
$> decTestInterface_NASA -n testSFL
```

```
[ INFO] 2022-08-29 10:35:33.742 nl2-u-moc-srv-01.test - DECTestInterface_NASA_CDDIS::testSFL START
[DEBUG] 2022-08-29 10:35:33.742 nl2-u-moc-srv-01.test - decConfigInterface2DB -a NASA_SFL
[ INFO] 2022-08-29 10:35:34.799 nl2-u-moc-srv-01.db   - [DEC_001] I/F NASA_SFL: added to the DEC Inventory/Interface db
[ INFO] 2022-08-29 10:35:34.810 nl2-u-moc-srv-01.test - decCheckConfig -e NASA_SFL
[DEBUG] 2022-08-29 10:35:36.846 nl2-u-moc-srv-01.chck - [DEC_XXX] I/F NASA_SFL: File jul2022f10_prd.txt is available
[ INFO] 2022-08-29 10:35:36.852 nl2-u-moc-srv-01.chck - [DEC_004] I/F NASA_SFL: exchange point is reachable
[ INFO] 2022-08-29 10:35:37.456 nl2-u-moc-srv-01.chck - [DEC_003] I/F NASA_SFL: interface is correctly declared in DEC/Inventory
[ INFO] 2022-08-29 10:35:37.468 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_SFL -l
[ INFO] 2022-08-29 10:35:39.543 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_SFL: Polling Started - List mode is true
[ INFO] 2022-08-29 10:35:40.386 nl2-u-moc-srv-01.pull - [DEC_105] I/F NASA_SFL: File jul2022f10_prd.txt is available
[ INFO] 2022-08-29 10:35:40.386 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_SFL: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:35:40.386 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_SFL: Pull iteration completed successfully
[ INFO] 2022-08-29 10:35:40.403 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_SFL
[ INFO] 2022-08-29 10:35:42.324 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_SFL: Polling Started - List mode is false
[ INFO] 2022-08-29 10:35:43.199 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_SFL: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:35:44.381 nl2-u-moc-srv-01.pull - [DEC_110] I/F NASA_SFL: jul2022f10_prd.txt downloaded with size 18431 bytes
[ INFO] 2022-08-29 10:35:44.446 nl2-u-moc-srv-01.pull - [DEC_115] Intray NAOS_IN: jul2022f10_prd.txt disseminated into /data/mocExternalInterfaces/tmp/naos/
[ INFO] 2022-08-29 10:35:44.447 nl2-u-moc-srv-01.pull - [DEC_130] I/F NASA_SFL: event triggered newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/jul2022f10_prd.t
[ INFO] 2022-08-29 10:35:44.768 nl2-u-moc-srv-01.auxc - [AUX_001] NS1_OPER_AUX_SFL____20220101T000000_20411001T000000_0001.TXT generated from jul2022f10_prd.txt
[ INFO] 2022-08-29 10:35:44.772 nl2-u-moc-srv-01.pull - [DEC_130] I/F NASA_SFL: event completed newfile2intray => auxConverter -m NAOS -f /data/mocExternalInterfaces/tmp/naos/jul2022f10_prd.t
[ INFO] 2022-08-29 10:35:44.816 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_SFL: Pull iteration completed successfully
[ INFO] 2022-08-29 10:35:44.832 nl2-u-moc-srv-01.test - decStats
[ INFO] 2022-08-29 10:35:45.811 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"jul2022f10_prd.txt","interface":"NASA_SFL","date":"2022-08-29 08:35:44 UTC","protocol":"HTTP","size":18
[ INFO] 2022-08-29 10:35:45.811 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"5.00 B/s","volume":"18.00 KiB"}
[ INFO] 2022-08-29 10:35:45.814 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[ INFO] 2022-08-29 10:35:45.821 nl2-u-moc-srv-01.test - decGetFromInterface -m NASA_SFL
[ INFO] 2022-08-29 10:35:47.673 nl2-u-moc-srv-01.pull - [DEC_005] I/F NASA_SFL: Polling Started - List mode is false
[ INFO] 2022-08-29 10:35:48.547 nl2-u-moc-srv-01.pull - [DEC_060] I/F NASA_SFL: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-08-29 10:35:49.624 nl2-u-moc-srv-01.pull - [DEC_111] I/F NASA_SFL: jul2022f10_prd.txt downloaded is duplicated / same md5
[ INFO] 2022-08-29 10:35:49.658 nl2-u-moc-srv-01.pull - [DEC_100] I/F NASA_SFL: Pull iteration completed successfully
[ INFO] 2022-08-29 10:35:49.675 nl2-u-moc-srv-01.test - decStats
[ INFO] 2022-08-29 10:35:50.680 nl2-u-moc-srv-01.stat - [DEC_144] Pull 1h: {"filename":"jul2022f10_prd.txt","interface":"NASA_SFL","date":"2022-08-29 08:35:44 UTC","protocol":"HTTP","size":18
[ INFO] 2022-08-29 10:35:50.680 nl2-u-moc-srv-01.stat - [DEC_144] Pull stats: {"numFiles":1,"hours":1,"rate":"5.00 B/s","volume":"18.00 KiB"}
[ INFO] 2022-08-29 10:35:50.683 nl2-u-moc-srv-01.stat - [DEC_244] Push 1h: No files pushed
[ INFO] 2022-08-29 10:35:50.696 nl2-u-moc-srv-01.test - DECTestInterface_NASA_CDDIS::testSFL END
```

F.9 NOAA

F.9.1 Report Solar Geophysical Activity

NOAA publishes solar and geophysical activity reports is a brief list of solar and geophysical events and indices including energetic solar flares, proton events, geomagnetic activity, and stratospheric warming alerts (cf. RSGA).

```
$> decTestInterface_NOAA
```

```
decGetFromInterface -m NOAA -l
[ INFO] 2022-02-15 13:57:09 NODE_1.pull - [DEC_005] I/F NOAA: Polling Started - List mode is true
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220101RSGA.txt is available
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220102RSGA.txt is available
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220103RSGA.txt is available
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220104RSGA.txt is available

(...)

[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220211RSGA.txt is available
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220212RSGA.txt is available
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220213RSGA.txt is available
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_105] I/F NOAA: File 20220214RSGA.txt is available
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_060] I/F NOAA: Polling Completed / New file(s) 45 available for pull
[ INFO] 2022-02-15 13:57:12 NODE_1.pull - [DEC_100] I/F NOAA: Pull iteration completed successfully
decGetFromInterface -m NOAA
[ INFO] 2022-02-15 13:57:20 NODE_1.pull - [DEC_005] I/F NOAA: Polling Started - List mode is false
[ INFO] 2022-02-15 13:57:23 NODE_1.pull - [DEC_060] I/F NOAA: Polling Completed / New file(s) 45 available for pull
[ INFO] 2022-02-15 13:57:25 NODE_1.pull - [DEC_110] I/F NOAA: 20220102RSGA.txt downloaded with size 2022 bytes
[ INFO] 2022-02-15 13:57:25 NODE_1.pull - [DEC_110] I/F NOAA: 20220101RSGA.txt downloaded with size 2121 bytes
[ INFO] 2022-02-15 13:57:25 NODE_1.pull - [DEC_115] Intray S3_IN: 20220102RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-15 13:57:25 NODE_1.pull - [DEC_115] Intray S3_IN: 20220101RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-15 13:57:26 NODE_1.pull - [DEC_110] I/F NOAA: 20220103RSGA.txt downloaded with size 1974 bytes
[ INFO] 2022-02-15 13:57:26 NODE_1.pull - [DEC_115] Intray S3_IN: 20220103RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-15 13:57:28 NODE_1.pull - [DEC_110] I/F NOAA: 20220104RSGA.txt downloaded with size 1924 bytes
[ INFO] 2022-02-15 13:57:28 NODE_1.pull - [DEC_110] I/F NOAA: 20220105RSGA.txt downloaded with size 1953 bytes
[ INFO] 2022-02-15 13:57:28 NODE_1.pull - [DEC_115] Intray S3_IN: 20220104RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-15 13:57:28 NODE_1.pull - [DEC_115] Intray S3_IN: 20220105RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/

(...)

[ INFO] 2022-02-15 13:58:07 NODE_1.pull - [DEC_110] I/F NOAA: 20220212RSGA.txt downloaded with size 1987 bytes
[ INFO] 2022-02-15 13:58:07 NODE_1.pull - [DEC_115] Intray S3_IN: 20220212RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-15 13:58:09 NODE_1.pull - [DEC_110] I/F NOAA: 20220213RSGA.txt downloaded with size 1973 bytes
[ INFO] 2022-02-15 13:58:09 NODE_1.pull - [DEC_115] Intray S3_IN: 20220213RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-15 13:58:09 NODE_1.pull - [DEC_110] I/F NOAA: 20220214RSGA.txt downloaded with size 1952 bytes
[ INFO] 2022-02-15 13:58:09 NODE_1.pull - [DEC_115] Intray S3_IN: 20220214RSGA.txt disseminated into /tmp/dec_local_dissemination/S3/
[ INFO] 2022-02-15 13:58:10 NODE_1.pull - [DEC_135] I/F NOAA: S2__OPER_DEC_F_RECV_2BOA_20220215T135810_V20220215T135810_20220215T135810_NOAA.xml pull report created
[ INFO] 2022-02-15 13:58:10 NODE_1.pull - [DEC_100] I/F NOAA: Pull iteration completed successfully
decGetFromInterface -m PUBLISH_S3
[ INFO] 2022-02-15 13:58:26 NODE_1.pull - [DEC_005] I/F PUBLISH_S3: Polling Started - List mode is false
[ INFO] 2022-02-15 13:58:26 NODE_1.pull - [DEC_050] I/F PUBLISH_S3: Polling Completed / No file(s) available for pull
[ INFO] 2022-02-15 13:58:26 NODE_1.pull - [DEC_100] I/F PUBLISH_S3: Pull iteration completed successfully


decStats -H 1
[ INFO] 2022-02-15 13:58:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"20220102RSGA.txt","interface":"NOAA","date":"2022-02-15 12:57:25 UTC","protocol":"FTP","size":2022}
[ INFO] 2022-02-15 13:58:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"20220101RSGA.txt","interface":"NOAA","date":"2022-02-15 12:57:25 UTC","protocol":"FTP","size":2121}
[ INFO] 2022-02-15 13:58:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"20220103RSGA.txt","interface":"NOAA","date":"2022-02-15 12:57:26 UTC","protocol":"FTP","size":1974}

(...)

[ INFO] 2022-02-15 13:58:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"20220213RSGA.txt","interface":"NOAA","date":"2022-02-15 12:58:09 UTC","protocol":"FTP","size":1973}
[ INFO] 2022-02-15 13:58:32 NODE_1.stat - [DEC_144] Pull 1h: {"filename":"20220214RSGA.txt","interface":"NOAA","date":"2022-02-15 12:58:09 UTC","protocol":"FTP","size":1952}
[ INFO] 2022-02-15 13:58:32 NODE_1.stat - [DEC_144] Pull stats: {"numFiles":45,"hours":1,"rate":"25.00 B/s","volume":"88.27 KiB"}
[ INFO] 2022-02-15 13:58:32 NODE_1.stat - [DEC_244] Push 1h: No files pushed

.
Finished in 103.289446 seconds.
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
1 tests, 6 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
0.01 tests/s, 0.06 assertions/s
End of DEC test for NOAA interface
```

F.10 SPCS / CSpOC

The SPCS publishes space situational awareness data in space-track.org using some RESTful API.

F.10.1 Conjunction Data Messages

The public conjunction data messages published by SPCS / CSpOC are retrieved in JSON using the available API and then dumped into a file for subsequent consumption.

```
[ INFO] 2022-03-08 12:47:17 NODE_1.test - DECTestCaseSPCS::test_pull => START
 /tmp/dec_local_dissemination/CDM/cdm_public.json
[DEBUG] 2022-03-08 12:47:18 NODE_1.test - decGetFromInterface -m SPCS -l --nodb
[ INFO] 2022-03-08 12:47:20 NODE_1.pull - [DEC_005] I/F SPCS: Polling Started - List mode is true
[ INFO] 2022-03-08 12:47:22 NODE_1.pull - [DEC_105] I/F SPCS: File cdm_public is available
[ INFO] 2022-03-08 12:47:22 NODE_1.pull - [DEC_060] I/F SPCS: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-03-08 12:47:22 NODE_1.pull - [DEC_100] I/F SPCS: Pull iteration completed successfully
[DEBUG] 2022-03-08 12:47:24 NODE_1.test - decGetFromInterface -m SPCS --nodb
[ INFO] 2022-03-08 12:47:26 NODE_1.pull - [DEC_005] I/F SPCS: Polling Started - List mode is false
[ INFO] 2022-03-08 12:47:28 NODE_1.pull - [DEC_060] I/F SPCS: Polling Completed / New file(s) 1 available for pull
[ INFO] 2022-03-08 12:48:44 NODE_1.pull - [DEC_110] I/F SPCS: cdm_public.json downloaded with size 9747797 bytes
[ INFO] 2022-03-08 12:48:44 NODE_1.pull - [DEC_115] Intray CDM: cdm_public.json disseminated into /tmp/dec_local_dissemination/CDM
[ INFO] 2022-03-08 12:48:44 NODE_1.pull - [DEC_135] I/F SPCS: S2__OPER_DEC_F_RECV_2BOA_20220308T124844_V20220308T124844_20220308T124844_SPCS.xml pull report created
[ INFO] 2022-03-08 12:48:44 NODE_1.pull - [DEC_100] I/F SPCS: Pull iteration completed successfully
[DEBUG] 2022-03-08 12:48:46 NODE_1.test - cat /tmp/dec_local_dissemination/CDM/cdm_public.json
[
  {
    "CDM_ID": "225135432",
    "CREATED": "2022-02-06 08:54:39.000000",
    "EMERGENCY_REPORTABLE": "Y",
    "TCA": "2022-02-06T11:10:31.625000",
    "MIN_RNG": "568",
    "PC": "0.0003684127",
    "SAT_1_ID": "177",
    "SAT_1_NAME": "THOR ABLESTAR DEB",
    "SAT1_OBJECT_TYPE": "DEBRIS",
    "SAT1_RCS": "SMALL",
    "SAT_1_EXCL_VOL": "5.00",
    "SAT_2_ID": "45128",
    "SAT_2_NAME": "COSMOS 2491 DEB *",
    "SAT2_OBJECT_TYPE": "DEBRIS",
    "SAT2_RCS": null,
    "SAT_2_EXCL_VOL": "5.00"
  },
  {
    "CDM_ID": "225135604",
    "CREATED": "2022-02-06 08:54:40.000000",
    "EMERGENCY_REPORTABLE": "Y",
    "TCA": "2022-02-08T15:19:39.697000",
    "MIN_RNG": "114",
    "PC": "0.0006129776",
    "SAT_1_ID": "426",
    "SAT_1_NAME": "THOR AGENA B R/B",
    "SAT1_OBJECT_TYPE": "ROCKET BODY",
    "SAT1_RCS": "LARGE",
    "SAT_1_EXCL_VOL": "5.00",
    "SAT_2_ID": "38716",
    "SAT_2_NAME": "FENGYUN 1C DEB",
    "SAT2_OBJECT_TYPE": "DEBRIS",
    "SAT2_RCS": "SMALL",
    "SAT_2_EXCL_VOL": "5.00"
  },
  {
    "CDM_ID": "225135691",
    "CREATED": "2022-02-06 08:54:40.000000",
    "EMERGENCY_REPORTABLE": "Y",
    "TCA": "2022-02-07T10:56:31.671000",
    "MIN_RNG": "87",
    "PC": "0.001533235",
    "SAT_1_ID": "560",
    "SAT_1_NAME": "THOR ABLESTAR DEB",
    "SAT1_OBJECT_TYPE": "DEBRIS",
    "SAT1_RCS": "SMALL",
    "SAT_1_EXCL_VOL": "5.00",
    "SAT_2_ID": "41827",
    "SAT_2_NAME": "FENGYUN 1C DEB",
    "SAT2_OBJECT_TYPE": "DEBRIS",
    "SAT2_RCS": "SMALL",
    "SAT_2_EXCL_VOL": "5.00"
  },

(...)

  {
    "CDM_ID": "249238682",
    "CREATED": "2022-03-08 01:31:15.000000",
    "EMERGENCY_REPORTABLE": "Y",
    "TCA": "2022-03-10T03:32:24.824000",
```

```
    "MIN_RNG": "424",
    "PC": "0.0001140841",
    "SAT_1_ID": "49691",
    "SAT_1_NAME": "COSMOS 1408 DEB",
    "SAT1_OBJECT_TYPE": "DEBRIS",
    "SAT1_RCS": "SMALL",
    "SAT_1_EXCL_VOL": "5.00",
    "SAT_2_ID": "32430",
    "SAT_2_NAME": "FENGYUN 1C DEB",
    "SAT2_OBJECT_TYPE": "DEBRIS",
    "SAT2_RCS": "SMALL",
    "SAT_2_EXCL_VOL": "5.00"
  },
  {
    "CDM_ID": "249238918",
    "CREATED": "2022-03-08 01:31:16.000000",
    "EMERGENCY_REPORTABLE": "Y",
    "TCA": "2022-03-09T20:46:07.657000",
    "MIN_RNG": "476",
    "PC": "0.000365506",
    "SAT_1_ID": "49710",
    "SAT_1_NAME": "COSMOS 1408 DEB",
    "SAT1_OBJECT_TYPE": "DEBRIS",
    "SAT1_RCS": "SMALL",
    "SAT_1_EXCL_VOL": "5.00",
    "SAT_2_ID": "43112",
    "SAT_2_NAME": "PSLV DEB",
    "SAT2_OBJECT_TYPE": "DEBRIS",
    "SAT2_RCS": "LARGE",
    "SAT_2_EXCL_VOL": "5.00"
  }
]
[ INFO] 2022-03-08 12:34:13 NODE_1.test - DECTestCaseSPCS::test_pull => END
.
Finished in 89.55922 seconds.
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
1 tests, 4 assertions, 0 failures, 0 errors, 0 pendings, 0 omissions, 0 notifications
100% passed
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
0.01 tests/s, 0.04 assertions/s
[ INFO] 2022-03-08 12:34:13 NODE_1.test - End of DEC test for SPCS
```

G Version History

This section includes the version history, which is used for reference of the SW this document applies to. The history brings the main updates associated to every version ; it is not meant to be exhaustive but to make every version recognisable.

The history does not include all the long heritage associated to this SW ; the starting point is the capacity to install / uninstall different versions seamlessly with the gem installer and therefore the need to quickly identify what is shipped within each version.

In order to check the version which is currently installed the gem command can be used or alternatively execute any DEC command line with the "-v" flag to obtain the information shown below.

```
$> gem list | grep dec

 dec (1.0.41.9)

$> decValidateConfig -v

    1.0.41 => Support to interfaces with custom scripts
    DEC env variables override the configuration attributes shipped with the installer
1.0.40 => MD5 config flag for pull to manage duplications (HTTP/FTPS)
    HTTP handler treats html navigation pages as directories for pull
1.0.39 => Support of SSH commands
    decNATS for NAOS updated
    decListener exit code always set to 0 if command is successful
1.0.38 => Fix on HTTP HEAD to use VerifyPeerSSL configuration
1.0.37 => New gem roman-numerals is required
    Support of SFTP authentication with password using sshpass
    Support of SFTP known_host checks disabled for the container build process
    Listener on status message code is fixed
1.0.36 => LocalDissemination config items required to drive automation decListener
1.0.35 => Management of duplications with MD5 for HTTP protocol
1.0.34 => NATS protocol support for NAOS MCS CCS5
1.0.33 => Update to use ruby 3.x series as interpreter
    User & Pass configuration items now can be kept encrypted
1.0.32 => HTTP handler updated to get only href anchors for SCIHUB
1.0.31 => Update of the OData client for DHUS to support S5P (s5phub)
1.0.30 => Generation of pull report files DEC_F_RECV when some file failed retrieval
```

```
1.0.29 => Enhancement to parallelise pull operations
    decODataClient updates:
    > ADGS support (AUXIP)
    > download products from DHUS/OpenHub
    > log messages 257, 259, 260 and 667
    first version of man pages shipped
    zero length files unpresent now raises error [DEC_799]
    InterfaceHandlerFTPS_Implicit now raises error when time-out
    public API now kept under ctc module
1.0.28 => CUC::CheckerProcessUniqueness rework
    decODataClient update to support DHUS/GNSS API
1.0.27 => curl --connect-timeout raised to 60 to absorb high latencies
    decODataClient updates:
    > new AVDHUS generated for CreationDate intervals
    > request retries implemented for DHUS
1.0.26 => Patch to delete sucessful download files in FTPS protocol:
    Update of the OData client for DHUS:
    > to query by sensing dates referred to ContentDate/Start
    > to define a configurable delay wrt the catalogue CreationDate
    > to request XML, JSON and CSV
1.0.25 => VerifyPeerSSL is used by HTTP handlers
    decODataClient support for DHUS / Sentinel-1
1.0.24 => Pull local dissemination chmod robustified for errors
1.0.23 => Local dissemination upon pull is safely compressed:
    Update of the OData client for S2PRIP to support download of PDI
1.0.22 => WebDAV verb MOVE for push does not carry time-out parameters
1.0.21 => Pull local dissemination chmod robustified for errors
1.0.20 => Support of WebDAV protocol verbs PUT + MOVE for push circulations
1.0.19 => FTPS Implicit mode (port 990) support for pull & push mode
    Pull local dissemination Compress 7z-x mode to decompress 7z files
    Escape special characters for user / password credentials
1.0.18 => Update of the OData client for DHUS & S2PRIP to support pagination
    OData client for DHUS to stream to the console the received XML
1.0.17 => Support to push parallelisation driven by ParallelDownload config
    OData client for DHUS supports pagination
1.0.16 => First version of the OData client for DHUS / PRIP for Sentinel-2
1.0.15 => Support to remote inventory / db different than localhost
    dec_config.xml Inventory config now includes Database_Host & Database_Port items
1.0.14 => New gems sys-filesystem and nokogiri are required
    Support of HTTP(S) protocol verb PUT for push circulations
    Support of HTTP(S) protocol verb GET for pull directories
    Support of HTTP(S) protocol verb HEAD for checking interface URL
    Migration of FTPS (explicit mode) protocol for pull circulations
    Migration of FTPS (explicit mode) protocol for push circulations
    dec_config.xml Inventory item added for database configuration
    dec_config.xml reshuffle of some configuration items
    dec_interfaces.xml configuration item DeleteFlag removed
    dec_incoming_files.xml config item Switches to handle duplication, unknowns, for each I/F
    dec_incoming_files.xml config item FileList replaced by DownloadRules
```

```
    dec_incoming_files.xml config item Execute command for each Intray available in Dissemination
1.0.13 => Support of Pull circulations using HTTP protocol for known URLs
    Support of authentication for HTTP(S) GET and DELETE verbs
    dec_interfaces.xml defines <VerifyPeerSSL> to validate the certificate
    log messages rationalisation and clean-up
    gem now includes the gemfile dependencies for their resolution at installation time
1.0.12 => Support of WebDAV / HTTP(S) protocol using verbs PROPFIND,GET & DELETE
    for pull mode (dec_incoming_files.xml)
    DEC RetrievedFiles report updated to Sentinels naming convention:
    dec_interfaces.xml replaces "FTPServer" with "Server" configuration item
1.0.11 => Migration to ActiveRecord 6
1.0.10 => new dec_config.xml deprecates dcc_config.xml & ddc_config.xml
    new dec_incoming_files.xml deprecates files2Intrays.xml & ft_incoming_files.xml
    new dec_outgoing_files.xml deprecates ft_outgoing_files.xml
    Earth Explorer / Earth Observation file-types are deprecated
    support to multiple log4r outputters
    unit tests updated to verify the PUSH mode to send files
1.0.9 => decUnitTests support batchmode to avoid prompting for confirmation
1.0.8 => decListener command line flags fixed
1.0.7 => decManageDB creates an index by filename for all tables
1.0.6 => decCheckConfig write checks for UploadDir/UploadDir for non secure FTP
1.0.5 => notify2Interface.rb fix sending mail to first address only
   decCheckConfig shipped in the gem
1.0.4 => decValidateConfig shipped with the required xsd schemas
1.0.3 => upgrade of rpf module to support ruby 2.x series
1.0.2 => commands triggered by reception events are now logged
1.0.1 => decStats -H <hours> has been integrated
1.0.0 => first version of the dec gem installer created
```

The auxiliary data converter tool is installed separately since it can be used as a standalone tool if precised.

```
$> gem list | grep aux

 aux (0.0.9.10)

$> auxConverter -v


DEC auxConverter Version: [0.0.9.10]

0.0.9 => ESA Sentinels SAFE format support
        ESA EOFFS format support
        NASA CDDIS IERS Bulletin A ASCII ser7.dat is supported
        Dynamic configuration parameters from env files
        Independent installation removing dependencies
0.0.8 => NASA ASTER Global DEM: ASTGTM
        NASA SRTMGL1 is supported
        USGS SRTMGL1 (geotiff) is supported / requires GDAL
        NASA MODIS products: MOD09A1
```

```
0.0.7 => IERS Bulletin C conversion for NAOS mission
0.0.6 => IERS Bulletin A XML is supported
         IERS Bulletin A ASCII updated for NAOS
0.0.5 => Celestrak CssiSpaceWeather Daily Prediction is supported
         Celestrak TCA (TLE catalogue) is supported
         Celestrak TLE (TLE mission)   is supported
         NASA MSFC Solar Flux (F10.7) / Geomagnetic disturbance (Ap) is supported
         NASA CDDIS Bulletin A / Earth Orientation Parameters is supported
         NASA CDDIS Bulletin C / TAI-UTC is supported
0.0.4 => NOAA Report Solar Geophysical Activity is supported
0.0.3 => IERS Bulletin A / Earth Orientation Parameters is supported
0.0.2 => IGS Broadcast Ephemeris Daily is supported
0.0.1 => IERS Bulletin C / TAI-UTC is supported
0.0.0 => first version of the aux installer created
```

H Docker Container DEC App

This section includes an example of a Dockerfile for a DEC node deployed as a standalone app. The definition carries the network authentication artefacts (e.g. SSH keys), which are not included as part of the DEC SW installer (cf. gem file).

```
##############################################################################
#
# Git:
#     Dockerfile.dec.s2.inputhub
#                           $Author: bolf$
#                           $Date: 2018-11-13T11:54:43+01:00$
#                           $Committer: bolf$
#                           $Hash: f3afa7c$
#
##############################################################################
#
# Dockerfile for DEC S2 INPUTHUB
#
# (Debian 10 Buster)
#
FROM ruby:2.6
# ================================================
RUN apt-get update && \
    apt-get install -y apt-utils && \
    apt-get install -y --no-install-recommends \
    net-tools \
    ftp \
    jq \
    libxml2 \
    libxml2-utils \
    xml-twig-tools \
    p7zip-full \
    sqlite3 \
    libsqlite3-dev \
    ncftp \
    curl \
    vim
# ------------------------------------------------
```

```
RUN apt-get install -y postgresql-client
# ------------------------------------------------
# ------------------------------------------------
RUN groupadd -g 10001 e2edc && \
    useradd -u 10001 -ms /bin/bash -g e2edc -d /home/e2edc e2edc
# ------------------------------------------------
COPY ./install/scripts/entrypoint_dec_s2_e2espm-inputhub_e2edc.sh /usr/bin/
RUN ln -s /usr/bin/entrypoint_dec_s2_e2espm-inputhub_e2edc.sh /
# ------------------------------------------------
RUN gem update
# ------------------------------------------------
COPY ./install/gems/dec_s2.gem .
RUN gem install dec_s2.gem
# ------------------------------------------------
# ------------------------------------------------
# Patch log4r
COPY ./install/patch/rollingfileoutputter.rb /usr/local/bundle/gems/log4r-1.1.10/lib/log4r/output
# ------------------------------------------------
# MKDIR p /volumes/dec
RUN mkdir -p /volumes/dec/log && \
    chown -R 10001:10001 /volumes/dec
# ------------------------------------------------
USER e2edc
RUN   mkdir -p /home/e2edc/.ssh
COPY --chown=10001:10001 ./config/ssh/* /home/e2edc/.ssh/
COPY --chown=10001:10001 ./config/ssh/e2espm.e2espm-esrin.private /home/e2edc/.ssh/id_rsa
COPY --chown=10001:10001 ./config/ssh/known_hosts_pdmcdag1.sentinel2.eo.esa.int /home/e2edc/.ssh/
ENV   HOSTNAME=dec GEM_HOME=/usr/local/bundle PATH="/usr/local/bundle/bin:${PATH}"
# "------------------------------------------------
ENTRYPOINT ["/usr/bin/entrypoint_dec_s2_e2espm-inputhub_e2edc.sh"]
```

I Frequently Asked Questions

I do not find any document change record, or document version, is this really a document ?

Answer: DEC documentation *is* code and it is treated accordingly ; as such, SW updates are never committed in the development repository without their documentation updates ensuring a proper alignment. The pdf document is generated with a rake task, and thus it propagates a generation time (i.e. daily basis). For document management purposes, the SW version it applies plus the document generation date is the general key.

How many Interfaces can be configured in every DEC node ?

Answer: There is no restriction in the number of Interfaces configured in a given node.

How many DEC nodes can be configured in the same host ?

Answer: Using the DEC gem, the restriction is associated to the ruby run-time environment; in the installation information it is strongly recommended to include the ruby run-time environment associated to each OS user who will execute the SW in opposition to the usage of the OS common run-times (e.g. /usr/local/lib /opt/lib, etc). In this respect the run-time execution isolation principle is met and different OS user with their app / services can execute different DEC nodes for their specific purposes.

Are the DEC SW dependencies somehow redundant or overalapping (e.g. curl vs. ncftp)

Answer: All DEC SW dependencies rely on maintained SW. The implementation of the different protocol handlers is independent, and therefore can rely on different SW dependencies. The handler of the plain FTP protocols is based on some initial legacy SW which has been updated to deal with obsolescence so the ncftp SW has been maintained for the time being despite the extensive usage of curl in other different protocol handlers.

How can I get the inbound throughput associated to pull circulations?

Answer: log message DEC_110 carries the size of every file pulled.

```
[ INFO] 2021-10-08 15:55:39 SBOA.pull - [DEC_110] I/F HUB: foo downloaded with size 107295 bytes
[ INFO] 2021-10-08 15:58:02 SBOA.pull - [DEC_110] I/F HUB: boo downloaded with size 1245 bytes
[ INFO] 2021-10-08 15:58:02 SBOA.pull - [DEC_110] I/F HUB: moo downloaded with size 17106 bytes
```

How can I get the outbound throughput associated to push circulations?

Answer: log message DEC_210 carries the size of every file pulled.

```
[ INFO] 2021-10-08 15:55:39 SBOA.push - [DEC_210] I/F NAV: finals.all UPLOADED / size 11 bytes
[ INFO] 2021-10-08 15:58:02 SBOA.push - [DEC_210] I/F NAV: tai-utc.dat UPLOADED / size 12 bytes
```

Can DEC support non file-based *pull* circulations ?

Answer: There is no restriction in the protocols supported, which can be enlarged by the implementation of dedicated protocol handlers. Standard protocols such as HTTP can publish data by URLs which are not necessarily liaised to files. Also there are other protocols specialising HTTP with dedicated API / end-points, which are not file-based (e.g. OASIS Open Data Protocol / OData). Upon performing the pull request and obtaining the data, then DEC flushes the retrieved data into files to feed the consumers.

Can DEC support non file-based *push* circulations ?

Answer: There is no restriction in the protocols supported, which can be enlarged by the implementation of dedicated protocol handlers. Standard protocols such as HTTP can receive data at URLs, which are not necessarily liaised to files by usage of POST requests ; these are tailored for the interface end-point definition (i.e. parameters part of the request).

This document does not include information about the resources comssumption or requirements needs by DEC SW ; is there any constraint ? *push* circulations ?

Answer: Such lack of information is a gap of this document which needs to be addressed. A quick reply is that the DEC SW needs for resources is really negligible. As example it is mentioned one use-case in the IoT domain using a low power ARM platform with 512MB RAM (cf. *raspberry pi*) to gather local meteorological data and systematically push it for consumption.

Where can the configuration files be found?

Answer: The location of the configuration files can be obtained using the following command using the Debug flag:

```
$> decValidateConfig -e -D
```

```
[DEBUG] xmllint --schema
/home/s2ops/.frum/ruby/gems/dec-1.0.24/code/dec/../../schemas/dec_interfaces.xsd
/home/s2ops/.frum/ruby/gems/dec-1.0.24/code/dec/../../config/dec_interfaces.xml --noout
```

Are SMTP - email notifications supported for pull circulations?

Answer: No, this functionality was historically implemented to alert the outgoing files towards external whereas pull is already under the control of the SW workflow (e.g. availability of a report file with the files pulled). If there is a use-case, it is fairly straight-forward to expand the functionality to support also email notifications for pull operations.

I do not find the interface tests tools referenced in the Annex - Public Data Providers.

Answer: If you are interested in the interface tests or / and the DEC SW associated configuration for their exploitation, please ensure that installation package carries them (i.e. in general the installation file carries the ”test” name) to ensure both test tools and configuration are included. Just ask for them ◡̈ ☺☻.

Pull data from SPCS / CSpOC seems slow?

Answer: space-track.org asks for a fair usage of their interface to control the bandwidth costs and the pull operations using cURL are limited by using ’–limit-rate 100K’ according

to their API recommendation.

How come this document is so ugly?

Answer: Documentation is handled as code in the repository. The document is compiled so far with no frills or appealing style templates but the essential in order to keep compatibility when converting to some content managers such as Wiki or Atlassian Confluence as example.