NL-SCIA-DC Data Extractors

Richard M. van Hees
Netherlands Institute for Space Research

April 2, 2012

Contents

1 Introduction

This document describes the GOME and Sciamachy data extractors, originally developed for the Netherlands SCIAMACHY Data Center (NL-SCIA-DC). The software is developed to access the official GOME and SCIAMACHY level 0, 1b and 2 NRT data products as distributed by DLR and ESRIN.

SEE ALSO

More info about this software package can be found on the web-page of the Netherlands SCIAMACHY data center at NL-SCIA-DC.

2 GOME_LV1 Data Extractor

Name:

gome_lv1 – NL-SCIA-DC GOME level 1b (NRT product) extractor

Syntax:

gome_lv1 [OPTIONS] INFILE

Purpose:

Read ESA/DLR GOME level 1 products, extract subsets, optionally calibrate the science data, and write in a flexible binary format (HDF5) or ASCII

Description:

The NL-SCIA-DC GOME level 1b (NRT product) processor extracts data from a GOME level 1b product, and generates a HDF5 file. Optionally the extractor can dump, in human readable form, the contents of each PDS data set to a separate ASCII files.

Options and Parameters:

-help print out a message listing of the available options (alternatively: **-h**)

-version print out version number and copyright information, and exit (alternatively: **-V**)

-silent do not display any error messages

-show_param display setting of command-line parameters; no output generated

-check check inputfile by reading it; no output generated

-meta write (in ASCII format) NL-SCIA-DC meta-Database information

-ascii write output in ASCII format, generate for each PDS data set a separate file (see appendix B)

-sql ingest product into NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-remove delete product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-replace delete/ingest product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-hdf5 generate a HDF5 file *[default]*

-compress compress all datasets in the HDF5-file

-o *file_name* alternative name for the output HDF5 file and/or ASCII files. No filename extension has to be given: the extension for the HDF5 file is ".h5", and for the ASCII files the abbreviation of the PDS data set written (alternatively: **–output=***file_name*)

–time ***start_date start_time [end_date] end_time*** extract ground pixels within a given time-window. The end of the time window can be specified by *end_time* only, if the *start_date* is equal to the *end_date*. The following formats for date and time are recognised:

- **DD-MMM-YYYY, YYYY-MMM-DD** where MMM is one out of “JAN”, “FEB”, “APR”, ...
- **DD-MM-YYYY, YYYY-MM-DD** where MM is one out of “01”, “02”, “03”, ...
- **HH:MM:SS.SSS**

–region=*lat_min,lat_max,lon_min,lon_max* Geo-location given as the range in latitude and longitude (degrees)

-no_mds do not extract any Measurement Data Set records

-pmd extract PMD data

-no_pmd do not extract PMD data

-no_pmd_geo do not write geo-locations of the extracted PMD ground pixels

-nadir extract Nadir MDS records

-no_nadir do not extract Nadir MDS records

-moon extract Moon MDS records

-no_moon do not extract Moon MDS records

-sun extract Sun MDS records

-no_sun do not extract Sun MDS records

–ipixel=*min:max:step[,min:max:step]* extract ground pixels, by pixel number (for example: -p 100,125,350), or one or more pixel number ranges: between *min* and *max*, increasing the pixel number by *step*. Several ranges may be specified. Negative numbers are taken relative to the last pixel, “*” is a short cut for the last pixel. Note the index of the first pixel is 0 (zero)!

–xpixel=*[ECWB]* extract specific ground pixels: **E**ast, **C**enter, **W**est, **B**ackscan (default: all)

–band*[=1a,1b,2a,2b,3,4]* extract/write spectral band data (default: extract all)

-blind extract blind pixel data records

-stray extract stray-light data records (spectral bands: 1a, 1b, 2a)

–wave=*min,max* select spectral band data within the wavelength range between min and max

–cal_nadir*[=LAFSNPIU]* apply the default “**gdp**” calibration steps (*LAFSNPIU*), or only specific calibration steps as specified by a single key character. (see below for explanation) The default behavior is to write the spectral band data without any calibration

–cal_moon*[=LAFSNIU]* apply the default “**gdp**” calibration steps (*LAFSNIU*), or only specific calibration steps as specified by a single key character. (see below for explanation) The default behavior is to write the spectral band data without any calibration

–cal_sun*[=LAFSNBIU]* apply the default “**gdp**” calibration steps (*LAFSNBIU*), or only specific calibration steps as specified by a single key character. (see below for explanation) The default behavior is to write the spectral band data without any calibration.

–cal_pmd*[=LPIU]* apply the default “**gdp**” calibration steps (*LPIU*), or only specific calibration steps as specified by a single key character. (see below for explanation) The default behavior is to write the spectral band data without any calibration

Table 1: Listing of all implemented calibration options, including a brief description.

OPTIONS	SHORT DESCRIPTION
L	correct for fixed-pattern noise and leakage current
A	correct band 1a for cross-talk from the Peltier coolers in the Focal Plane Assembly
F	correct signals for pixel-to-pixel (fixed pattern) variations in the instrument response
S	correct signals for stray-light (uniform and spectral ghost components)
N	normalize the signals to 1 second detector integration time
B	divide signals from Sun measurements by the BSDF
P	Polarization sensitivity correction
I	absolute radiance calibration (Watt/(s cm^2sr nm))
U	unit conversion: Watt/(s cm^2sr nm) to photons/(s cm^2sr nm)
a	compute albedos instead of radiances
j	correct radiance jumps (*not implemented*)
d	correct for degradation (*not implemented*)

Examples:

1. Extract GOME level 1 data within a time window, output to 81022081.lv1.h5:

```
gome_lv1 --time 21-MAY-1998 12:00:00.0 22-MAY-1998 12:00:00.0 81022081.lv1
```

2. Extract the first 100 pixels and the last 100 pixels from a GOME level 1 scan:

```
gome_lv1 --ipixel=0:99:1,-100:*:1 81022081.lv1
```

Remarks:

Bug: ASCII output of PMD data is unreadable

ToDo: implement KNMI polarization correction

ToDo: implement correction for radiation jumps

ToDo: implement correction for degradation

3 GOME_LV2 Data Extractor

Name:

gome_lv2 – NL-SCIA-DC GOME level 2 (NRT product) extractor

Syntax:

gome_lv2 [OPTIONS] INFILE

Purpose:

Read ESA/DLR GOME level 2 products, extract subsets, and write in a flexible binary format (HDF5) or ASCII.

Description:

The NL-SCIA-DC GOME level 2 (NRT product) processor extracts data from a GOME level 2 product, and generates a HDF5 file. Optionally the extractor can dump, in human readable form, the contents of each PDS data set to a separate ASCII files.

Options and Parameters:

-help print out a message listing of the available options (alternatively: **-h**)

-version print out version number and copyright information, and exit (alternatively: **-V**)

-silent do not display any error messages

-show_param display setting of command-line parameters; no output generated

-check check inputfile by reading it; no output generated

-meta write (in ASCII format) NL-SCIA-DC meta-Database information

-ascii write output in ASCII format, generate for each PDS data set a separate file (see appendix B)

-sql ingest product into NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-remove delete product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-replace delete/ingest product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-hdf5 generate a HDF5 file *[default]*

-compress compress all datasets in the HDF5-file

-o *file_name* alternative name for the output HDF5 file and/or ASCII files. No filename extension has to be given: the extension for the HDF5 file is ".h5", and for the ASCII files the abbreviation of the PDS data set written (alternatively: **–output=***file_name*)

–time *start_date start_time [end_date] end_time* extract ground pixels within a given time-window. The end of the time window can be specified by *end_time* only, if the *start_date* is equal to the *end_date*. The following formats for date and time are recognized:

- **DD-MMM-YYYY, YYYY-MMM-DD** where MMM is one out of "JAN", "FEB", "APR", ...
- **DD-MM-YYYY, YYYY-MM-DD** where MM is one out of "01", "02", "03", ...
- **HH:MM:SS.SSS**

–region=*lat_min,lat_max,lon_min,lon_max* Geo-location given as the range in latitude and longitude (degrees)

–ipixel=*min:max:step[,min:max:step]* extract ground pixels, by pixel number (for example: -p 100,125,350), or one or more pixel number ranges: between *min* and *max*, increasing the pixel number by *step*. Several ranges may be specified. Negative numbers are taken relative to the last pixel, "*" is a short cut for the last pixel. Note the index of the first pixel is 0 (zero)!

–cloud=*min,max* cloud cover range [0,1] *(not implemented yet)*

–sunz=*min,max* Solar zenith angle range *(not implemented yet)*

Examples:

1. Extract gome level 2 data within a time window, output to 81022081.lv2.h5:

```
gome_lv2 --time 21-MAY-1998 12:00:00.0 22-MAY-1998 12:00:00.0 81022081.lv2
```

Remarks:

None

4 SCIA_NL0 Data Extractor

Name:

scia_nl0 – NL-SCIA-DC Sciamachy level 0 (NRT product) extractor

Syntax:

scia_nl0 [OPTIONS] INFILE

Purpose:

Read Envisat Sciamachy NRT level 0 products, extract subsets, and write in a flexible binary format (HDF5) or ASCII.

Description:

The NL-SCIA-DC Sciamachy level 0 (NRT product) processor extracts data from a Sciamachy level 0 product, and generates a HDF5 file. Optionally the extractor can dump, in human readable form, the contents of each PDS data set to a separate ASCII files.

Options and Parameters:

-help print out a message listing of the available options (alternatively: **-h**)

-version print out version number and copyright information, and exit (alternatively: **-V**)

-silent do not display any error messages

-show_param display setting of command-line parameters; no output generated

-check check inputfile by reading it; no output generated

-meta write (in ASCII format) NL-SCIA-DC meta-Database information

-info-cwd write a database with info-records in your current working directory. for fast future access to the input-file. The info-file contains, in binary format, a description of the MDS records in the file, as the PDS DSD records, but extended with the following parameters: packet_id, category, state_id, num_chan, chan_id, mjd, bcps, clusters, co_adding, and clus_len

-info-db same as **-info-cwd**, only the info-reocords will be stored in a database in the directory `DATA_DIR` (see `config/Rules_NADC.mk`)

-ascii write output in ASCII format, generate for each PDS data set a separate file (see appendix B)

-sql ingest product into NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-remove delete product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-replace delete/ingest product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-hdf5 generate a HDF5 file *[default]*

-compress compress all datasets in the HDF5-file

-o *file_name* alternative name for the output HDF5 file and/or ASCII files. No filename extension has to be given: the extension for the HDF5 file is ".h5", and for the ASCII files the abbreviation of the PDS data set written (alternatively: **–output=***file_name*)

–time ***start_date start_time [end_date] end_time*** extract ground pixels within a given time-window. The end of the time window can be specified by *end_time* only, if the *start_date* is equal to the *end_date*. The following formats for date and time are recognized:

- **DD-MMM-YYYY, YYYY-MMM-DD** where MMM is one out of "JAN", "FEB", "APR", ...
- **DD-MM-YYYY, YYYY-MM-DD** where MM is one out of "01", "02", "03", ...
- **HH:MM:SS.SSS**

-no_mds do not extract any Measurement Data Set records

-aux extract Auxiliary MDS records

-no_aux do not extract Auxiliary MDS records

-det extract Detector MDS records

-no_det do not extract Detector MDS records

-pmd extract PMD MDS records

-no_pmd do not extract PMD MDS records

–state=*[1,2,...,64]* extract/write data of MDS with selected states (using IDs)

–chan*[=1,2,...,8]* extract/write data of selected spectral bands (default: extract all)

Environment variables

NO_CLUSTER_CORRECTION check and correct cluster definitions as found in the Detector MDS. Implemented for orbit number higher than 4150.

Examples:

1. Extract only science Detector MDS records from a Sciamachy level 0 product and generate a HDF5 file:

```
scia_nl0 -hdf5 -det \
SCI_NL__0PNPDK20020823_103445_000060802008_00452_02510_0900.N1
```

2. Extract science Detector MDS records and PMD data from a Sciamachy level 0 product and generate a HDF5 file:

```
scia_nl0 -hdf5 -aux \
SCI_NL__0PNPDK20020823_103445_000060802008_00452_02510_0900.N1
```

Remarks:

This program has been shown to be very robust. A test on our Sciamachy level 0 dataset showed that the program failed in only 5 out of 42,000 products, while converting the products to HDF5, and none while dumping the data to ASCII. All consolidated products could be processed without problems. Known reason for failure are the presences of non-chronological MDS records.

5 SCIA_NL1 Data Extractor

Name:

scia_nl1 – NL-SCIA-DC Sciamachy level 1b/1c (NRT product) extractor

Syntax:

scia_nl1 [OPTIONS] INFILE

Purpose:

Read ESA/DLR Sciamachy NRT level 1b/1c products, extract subsets, optionally calibrate the science data, and write in L1c format, flexible binary format (HDF5) or ASCII.

Description:

The NL-SCIA-DC Sciamachy level 1b/1c (NRT product) processor extracts data from a Sciamachy level 1b product or level 1c child product, and generates a file in scial1c format. Optionally the extractor can generate a HDF5, or (ASCII) dump, in human readable form, the contents of each PDS data set to a separate ASCII files.

Options and Parameters:

-help print out a message listing of the available options (alternatively: **-h**)

-version print out version number and copyright information, and exit (alternatively: **-V**)

-silent do not display any error messages

-show_param display setting of command-line parameters; no output generated

-check check inputfile by reading it; no output generated

-meta write (in ASCII format) NL-SCIA-DC meta-Database information

-pds generate a Payload Data Segment (PDS) 1c file *[default]*. The use of this option implies **-lv1c**.

-pds_1b generate a Payload Data Segment (PDS) 1b file. Note only selection on MDS categories: *nadir, limb, occultation and monitor* is implemented (see below).

-ascii write output in ASCII format, generate for each PDS data set a separate file (see appendix B)

-sql ingest product into NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-remove delete product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-replace delete/ingest product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-hdf5 generate a HDF5 file

-compress compress all datasets in the HDF5-file

-o *file_name* alternative name for the output HDF5 file and/or ASCII files. No filename extension has to be given: the extension for the HDF5 file is ".h5", and for the ASCII files the abbreviation of the PDS data set written (alternatively: **–output=***file_name*)

–time ***start_date start_time [end_date] end_time*** extract ground pixels within a given time-window. The end of the time window can be specified by *end_time* only, if the *start_date* is equal to the *end_date*. The following formats for date and time are recognized:

- **DD-MMM-YYYY, YYYY-MMM-DD** where MMM is one out of "JAN", "FEB", "APR", ...
- **DD-MM-YYYY, YYYY-MM-DD** where MM is one out of "01", "02", "03", ...
- **HH:MM:SS.SSS**

–region=*lat_min,lat_max,lon_min,lon_max* Geo-location given as the range in latitude and longitude (degrees)

-no_gads do not extract any Global Annotation Data Sets

-no_ads do not extract any Annotation Data Sets

-no_aux0 do not extract level 0 Auxiliary ADS records

-no_pmd0 do not extract level 0 PMD ADS records

-no_mds do not extract any Measurement Data Set

-no_pmd do not extract PMD MDS records (level 1c)

-no_polV do not extract fractional polarization MDS (level 1c)

-limb extract Limb MDS records

-no_limb do not extract Limb MDS records

-moni extract Monitoring MDS records

-no_moni do not extract Monitoring MDS records

-nadir extract Nadir MDS records

-no_nadir do not extract Nadir MDS records

-occ extract Occultation MDS records

-no_occ do not extract Occultation MDS records

–cat=*[1,2,...,64]* extract/write data of MDS with selected categories (using IDs)

–state=*[1,2,...,64]* extract/write data of MDS with selected states (using IDs)

–chan*[=1,2,...,8]* extract/write data of selected spectral bands (default: extract all)

–clus=*[1,2,...,64]* extract/write data of selected clusters (default: extract all)

–cal*[=0,1,2,3,4,5,6,7,8,9,M,p,P,E,N]* apply the specified calibration steps, or (without further specification) perform a calibration to radiances following the ATBD (**–cal=atbd**). A listing of all calibration options, including a brief description, are given in Table 2. A more detailed description of the calibration algorithms are given in appendix C, and the auxiliary data files necessary for the NADC calibration are described in appendix D. Note that no calibration is applied when this option is not given. The use of this option implies **-lv1c**.

-lv1c force the HDF5 output to contain the same information and structure as the output produced by the ESA SciaL1C processor, equivalent to **–cal=none**

–cal_pmd*[=L]* apply spectral calibration on PMD MDS

Environment variables

SCIA_CORR_PET The Pixel Exposure Time of the Epitaxx readouts (channel 6–8) should be corrected by 1.18125e-3 sec, see for documentation `TN-SCIA-0000DO/19, 10.03.1999`. This correction will not be applied when the environment variable “SCIA_CORR_LOS” is set to zero. The default is to apply this correction.

SCIA_CORR_LOS the values of the level 1b line-of-sight zenith angles are always larger than zero, and the azimuth angle jumps with 180 degrees while scanning through nadir. Setting the environment variable “SCIA_CORR_LOS” to not zero will modify the output by: removing the jump in the azimuth angles and returns negative zenith angles, when the original azimuth angle was larger than 180 degree. The default is “SCIA_CORR_LOS=1”, which is different than the output from the EnviView-tool *SciaL1C*.

SCIA_MFACTOR_DIR give path to directory with auxiliary files for m-factor correction. Default is to look in directory ”**DATA_DIR**/m-factor_06.00”.

SCIA_CORR_L1C correct radiances in L1c product, using multiplication factors read from an auxiliary file (= value of “SCIA_CORR_L1C”). This option only works in combination with option “–cal=p”.

Examples:

1. Calibrate (= Mem/Lin,Dark,PPG/Etalon,Stray,Wavelength,Polarisation,Radiance,BDPM) a L1b product and generate a L1c product:

```
scia_nl1 --cal=0,1,2,3,4,5,6,7,9 \
     SCI_NL__1PPLRA20040601_155040_000056992027_00212_11789_1967.N1
```

2. Extract only nadir measurements from a Sciamachy level 1b product and generate an HDF5 file:

```
scia_nl1 -hdf5 -no_limb -no_moni -no_occ \
     SCI_NL__1PPLRA20040601_155040_000056992027_00212_11789_1967.N1
```

   or

```
scia_nl1 -hdf5 -nadir \
     SCI_NL__1PPLRA20040601_155040_000056992027_00212_11789_1967.N1
```

3. Extract measurements with State ID 8,32 (nad08 and limb05) from a Sciamachy level 1b product and generate a L1C file:

```
scia_nl1 --state=8,32 \
     SCI_NL__1PPLRA20040601_155040_000056992027_00212_11789_1967.N1
```

4. Extract all nadir measurements from a L1b product and write all (G)ADS and nadir MDS records to a L1b product:

```
scia_nl1 -pds_1b -nadir -o nadir.N1\
     SCI_NL__1PPLRA20040601_155040_000056992027_00212_11789_1967.N1
```

5. Multiply all pixel values with multiplication factors found in the file *./arbitrary_slit_inhomogeneity.ascii*, where the original L1b product is calibrated using the ESA scial1c:

```
export SCIA_CORR_L1C=./arbitrary_slit_inhomogeneity.ascii
scial1c -pmd -fracpol -type nadir -cal 0 \
     SCI_NL__1PPLRA20040601_155040_000056992027_00212_11789_1967.N1
scia_nl1 --cal=N -o arbitrary_slit.N1\
     SCI_NL__1PPLRA20040601_155040_000056992027_00212_11789_1967.N1.child
```

Remarks:

Note: limb measurements are corrected by the *limb dark measurement*, as described in the ATBD.

Note: the calibration algorithms are implemented in single precision, therefore, small differences compared to the scial1c output are observed (at the 1e-4 level)

Note: for unknown reason, geolocation calculation of the sub-satellite point differs from scial1c output

ToDo: error estimation on science data does not include polarisation and Mueller matrix errors

6 SCIA_NL2 Data Extractor

Name:

scia_nl2 – NL-SCIA-DC Sciamachy level 2 (NRT product) extractor

Syntax:

scia_nl2 [OPTIONS] INFILE

Purpose:

Read Envisat Sciamachy NRT level 2 products, extract subsets, and write in a flexible binary format (HDF5) or ASCII

Description:

The NL-SCIA-DC Sciamachy level 2 (NRT product) processor extracts data from a Sciamachy level 2 product, and generates a HDF5 file. Optionally the extractor can generate a dump, in human readable form, the contents of each PDS data set to a separate ASCII files.

Options and Parameters:

-help print out a message listing of the available options (alternatively: **-h**)

-version print out version number and copyright information, and exit (alternatively: **-V**)

-silent do not display any error messages

-show_param display setting of command-line parameters; no output generated

-check check inputfile by reading it; no output generated

-meta write (in ASCII format) NL-SCIA-DC meta-Database information

-ascii write output in ASCII format, generate for each PDS data set a separate file (see appendix B)

-hdf5 generate a HDF5 file *[default]*

-compress compress all datasets in the HDF5-file

-o *file_name* alternative name for the output HDF5 file and/or ASCII files. No filename extension has to be given: the extension for the HDF5 file is ".h5", and for the ASCII files the abbreviation of the PDS data set written (alternatively: **–output=***file_name*)

-no_gads do not extract any Global Annotation Data Sets

-no_ads do not extract any Annotation Data Sets

-no_mds do not extract any Measurement Data Set records

-cld extract Cloud/Aerosol MDS records

-no_cld do not extract Cloud/Aerosol MDS records

-bias extract BIAS MDS records

-no_bias do not extract BIAS MDS records

-doas extract DOAS MDS records

-no_doas do not extract DOAS MDS records

Examples:

Remarks:

None

7 SCIA_OL2 Data Extractor

Name:

scia_ol2 – NL-SCIA-DC Sciamachy level 2 (Off-line product) extractor

Syntax:

scia_ol2 [OPTIONS] INFILE

Purpose:

Read ESA/DLR Sciamachy Offline level 2 products, extract subsets, write in a flexible binary format (= HDF5) or ASCII

Description:

The NL-SCIA-DC Sciamachy level 2 (Off-line product) processor extracts data from a Sciamachy level 2 product, and generates a HDF5 file. Optionally the extractor can generate a dump, in human readable form, the contents of each PDS data set to a separate ASCII files.

Options and Parameters:

-help print out a message listing of the available options (alternatively: **-h**)

-version print out version number and copyright information, and exit (alternatively: **-V**)

-silent do not display any error messages

-show_param display setting of command-line parameters; no output generated

-check check inputfile by reading it; no output generated

-meta write (in ASCII format) NL-SCIA-DC meta-Database information

-ascii write output in ASCII format, generate for each PDS data set a separate file (see appendix B)

-sql ingest product into NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-remove delete product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-replace delete/ingest product from NL-SCIA-DC meta/pixel database (*PostrgreSQL*)

-hdf5 generate a HDF5 file *[default]*

-compress compress all datasets in the HDF5-file

-o *file_name* alternative name for the output HDF5 file and/or ASCII files. No filename extension has to be given: the extension for the HDF5 file is ".h5", and for the ASCII files the abbreviation of the PDS data set written (alternatively: **–output=***file_name*)

-no_gads no Global Annotation Data Sets extracted

-no_ads no Annotation Data Sets extracted

-no_mds do not extract any Measurement Data Set records

-cld extract Cloud/Aerosol MDS records

-no_cld do not extract Cloud/Aerosol MDS records

-bias extract BIAS MDS records

-no_bias do not extract BIAS MDS records

-doas extract DOAS MDS records

-no_doas do not extract DOAS MDS records

Examples:

Remarks:

None

A Description of the PostrgreSQL ingest

Configuration parameters for the PostrgreSQL server are read from the XML configuration file `nadc.config.xml`. This file maybe put in your home directory or in the correct working directory. The contents of the file should look like this:

```
<?xml version="1.0" ?>
<configuration>
  <metatables>
    <server
      host = "localhost"
      user = "nadc_admin"
      passwd = "*****"
      />
  </metatables>
</configuration>
```

Alternatively, you can use the environment variables **NADC_DB_HOST**, **NADC_DB_USER**, and **NADC_DB_PASSWD** to define the PostrgreSQL server settings. Note that these environment variable have preference above the XML configuration file.

B Description of the ASCII dump format

This document describes the ASCII output produced by the NADC data extractors. The ASCII output is a compromise between human readability and the aim to create an output which is well structured to be processed by text-parsers such as awk, Perl, Python, etc.

B.1 Syntax Definition

Every line contains the following elements:

- a field number, as defined in the Input/Output Data Definition Documents from ESA/DLR (*Field*)
- a data field description (*Comments*)
- the value of the data record in case of a scalar, or in case of an array a string `array[xdim]` (for an one dimensional array), or `array[xdim,ydim]` (for a two-dimensional array). The data values of an array are printed in the next line(s): `xdim` values on a line by `ydim` lines.

These fields are separated by a field separator, by default a “|”.

C Description of the Sciamachy calibration options

In this section the implemented algorithm of the different calibration steps are given, the technical background is given in the ATBD. The described calibration steps have to be applied in successive order, except for the masking of the bad/dead pixels.

C.1 Memory correction

C.1.1 ATBD (*option: 0m*)

The value for the memory correction for each Reticon detector readout is calculated by the level 0 to 1 processor and stored in the level 1b files (= MDS → Cluster data → Rsig(c)). The following equations are applied for the non-coadded and coadded readouts, respectively:

$$\begin{aligned} S_{i,j} &= S_{i,j} - 2 \cdot MEC_{i,j} \\ S_{i,j} &= S_{i,j} - 2 \cdot coaddf_j \cdot MEC_{i,j} \end{aligned}$$

C.1.2 NADC (*option: 0m+*)

The memory correction of the Reticon detectors is implemented as a function of the previous readout. This relation is measured during the Optec 5 measurements and validated on-flight during Sodap. Lookup tables which gives the relation between the measured signal and the correction are derived from these measurements. The correction is set to zero after a detector reset, i.e. start of a new state. The coadded signals are scaled by the PMD signal as described in the ATBD. The following equations are applied for the non-coadded and coadded readouts, respectively:

$$\begin{aligned} S_{i,j} &= S_{i,j} - MEC_{i-1,j} \\ S_{i,j} &= S_{i,j} - (coaddf_j - 1) \cdot MEC_{i,j} - MEC_{i-1,j} \end{aligned}$$

C.2 non-Linearity correction

The non-linearity correction is a function of the Epitaxx detector signal. This relation is measured during the Optec 5 measurements and validated on-flight during Sodap. Lookup tables which gives the relation between the measured signal and the correction are derived from these measurements.

C.2.1 NADC (*option: 0l*)

The value for the non-Linearity correction is stored in the level 1b files (= MDS → Cluster data → Esig(c)). The values of the non-linearity correction are scaled to 8-bit values. The following equations are applied for the non-coadded and coadded readouts, respectively:

$$\begin{aligned} S_{i,j} &= S_{i,j} - \text{scale\_factor}_j \cdot (nLC_{i,j} - \text{offset}_j) \\ S_{i,j} &= S_{i,j} - coaddf_j \cdot \text{scale\_factor}_j \cdot (nLC_{i,j} - \text{offset}_j) \end{aligned}$$

	offset	scale_factor
channel 6	-102	1.25
channel 7	+126	1.5
channel 8	+126	1.5

C.2.2 NADC (*option: 0l+*)

This correction assumes continuous sources, using the mean value for observations with co-adding factor larger than 1. The following equations are applied for the non-coadded and coadded readouts, respectively:

$$\begin{aligned} S_{i,j} &= S_{i,j} - nLC_{i,j} \\ S_{i,j} &= S_{i,j} - coaddf_j \cdot nLC_{i,j} \end{aligned}$$

C.3 Dark Current correction

C.3.1 Shot Noise

$$\varepsilon\_shot_{ij} = \sqrt{f_{Coad,ij} \cdot \sigma^2_{E,ij}(\varphi_{orb}) + \left| \frac{S_{ij} - f_{Coad,ij} \cdot FPN_{ij}}{e_j} \right|}$$

C.3.2 ATBD (Nadir)

$$\begin{aligned} S_{dark,ij} &= f_{Coad,ij} \cdot (FPN_{ij} + (LV_{Var,ij}(\varphi_{orb}) + DS_{Stray,ij}(\varphi_{orb})) \cdot t_{PET,ij}) \\ \varepsilon\_S_{dark,ij} &= \sqrt{f_{Coad,ij}} \cdot \sigma\_FPN_{ij} + f_{Coad,ij} \cdot t_{PET,ij} \cdot (\sigma\_LC_{0,ij} + \sigma\_LCVar, ij + \sigma\_DS_{Stray,ij}) \end{aligned}$$

C.3.3 ATBD (Limb)

$$\begin{aligned} S_{dark,ij} &= S_{dark(250km),ij} \\ \varepsilon\_S_{dark,ij} &= 2 \cdot \varepsilon\_shot_{ij} \end{aligned}$$

C.4 Pixel-to-Pixel Gain correction and Etalon correction

C.4.1 ATBD (*options: 3 & 4*)

$$\begin{aligned} S_{DP,ij} &= S_{dark,ij}/PPG_{ij}/ETN_{ij} \\ \varepsilon\_S_{DP,ij} &= \sqrt{\varepsilon\_S^2_{dark,ij} + (\delta_{ppg,ij} \cdot S_{DP,ij})^2} \end{aligned}$$

C.5 Solar Straylight correction

C.5.1 ATBD (*option: 2*)

$$\begin{aligned} S_{stray,ij} &= \text{scaling\_factor}_{ij} \cdot \text{straylight\_coded}_{ij}/10. \\ S_{DPS,ij} &= S_{DP,ij} - S_{stray,ij} \\ \varepsilon\_S_{DPS,ij} &= \sqrt{\varepsilon\_S^2_{DP,ij} + (\delta_{stray,ij} \cdot S_{stray,ij})^2} \end{aligned}$$

C.6 Wavelength calibration

C.6.1 ATBD (*option: 5*)

C.7 Polarisation Sensitivity correction

C.7.1 ATBD (*option: 6*)

The polarization correction factor and its accuracy:

$$\begin{aligned} c_{ij} &= 1/(1 + \mu 2^D_{ij} \cdot Q + \mu 3^D_{ij} \cdot U) \\ \Delta\_c_{ij} &= \sqrt{(\mu 2^D_{ij} \cdot \varepsilon\_Q_{ij})^2 + (\mu 3^D_{ij} \cdot \varepsilon\_U_{ij})^2} \end{aligned}$$

C.8 Radiance Sensitivity correction

C.8.1 ATBD (*option: 7*)

$$
\begin{aligned}
S_{cal,ij} &= S_{DPS,ij} \cdot c_{ij} / (M_1)_{ij,\alpha} \\
\varepsilon\_S_{ij} &= \sqrt{\varepsilon\_S_{DPS,ij}^2 + \varepsilon\_shot_{ij}^2 + 0.25} \\
\Delta\_S_{ij} &= \sqrt{\left(\frac{\varepsilon\_S_{ij}}{S_{DPS,ij}}\right)^2 + (\varepsilon\_M_1)_{ij}^2 + \left(\frac{\Delta\_c_{ij}}{c_{ij}}\right)^2}
\end{aligned}
$$

C.9 Divide with Solar spectrum

C.9.1 ATBD (*option: 8*)

C.9.2 NADC (*option: 8+*)

C.10 Bad/Dead Pixel mask

C.10.1 ATBD (*option: 9*)

C.10.2 NADC (*option: 9+*)

D Auxiliary calibration data sets

All auxiliary calibration datasets are search (first) in the current directory and (secondly) in the directory `DATA_DIR`. The value for `DATA_DIR` is set in the configuration file `config/Rules_NADC.mk`, and thus fixed during the compilation of nadc_tools.

D.1 Memory correction dataset

The memory correction values are written from a HDF5 file with a lookup table containing the correction value as function of measured signal. The name of the file: "MEMcorr.h5", its content is described in the table below:

Dataset	Type	dimensions
MemTable	float	5, 65536

D.2 Non Linearity correction dataset

The non-linearity correction values are written from a HDF5 file with a lookup table containing the correction value as function of measured signal. The name of the file: "NLcorr.h5", its content is described in the table below:

Dataset	Type	dimensions
CurveIndex	signed char	8, 1024
nLinTable	float	15, 65536

D.3 Dark current correction datasets

The analog offset and dark current values are read from a binary file: Little-endian, one file per orbit. Given are several parameters in the DarkCurrent records, see table below: Applicable files are searched in the directory: `DATA_DIR/DarkCurrent`.

Dataset	Type	dimensions
Orbit	long	1
MagicNumber	long	1
Saa	long	1
Time	double	1
Tobm	float	1
Tdet	float	8
QualityNumber	long	1
AnalogOffset	float	8192
DarkCurrent	float	8192
AnalogOffsetError	float	8192
DarkCurrentError	float	8192
ChiSquare	float	8192

The orbital variation of the darksignal is read from a binary file: Little-endian, one file per orbit. Given are the correction, only for channel 8 for 72 orbit phases, starting at orbit phase 0. The format of the file binary, containing one float array of dimension: 1024, 72 Applicable files are searched in the directory: `DATA_DIR/OrbitalVariation/Transmission`.

D.4 Radiance sensitivity correction dataset

The Radiance sensitivity values are written from a HDF5 file. The name of the file: "rsp_patch.h5", its content is described in the table below:

Dataset	Type	dimensions
KEY_DATA_FILE	string	62
/RSPL/azimuth	float	115
/RSPL/elevation	float	115
/RSPL/sensitivity	float	115, 8192
/RSPN/azimuth	float	17
/RSPN/elevation	float	17
/RSPN/sensitivity	float	17, 8192
/RSPO/azimuth	float	45
/RSPO/elevation	float	45
/RSPO/sensitivity	float	45, 8192

D.5 Dead and Bad pixel mask datasets

The bad pixel mask is read from an ASCII file: one file per orbit. Each file contains a table with one column and 8192 rows holding an integer flag: 0 = ok, 1 = Bad. Comment-lines are indicated with an #, as first character. Applicable files are searched in the directory: `DATA_DIR/SmoothMask/ASCII`.

E About NADC tools

E.1 Purpose and Limitations

The probably most important part of the NL-SCIA-DC software (nadc_tools) are the C-libraries, which contain functions to read Sciamachy data products into memory, dump the data as ASCII output or write the data in PDS or HDF5 format. The software is written in ANSI-C (and even assembler), and coded in such a way that you (an experienced (C-) programmer) can easily understand the code. The data extractors, included in this software package, are developed for the NL-SCIA-DC and give a good example of how to use the libraries.

The software distributed in this package contains libraries, written in ANSI-C (mostly POSIX compliant except for some ISO C99 extensions) and an IDL wrapper library (using `CALL_EXTERNAL`) with IDL functions to read GOME and Sciamachy data. The IDL interface requires sharable object libraries. Building shared libraries is easy on the supported platforms, however, do not forget to set the environment variable `LD_LIBRARY_PATH`.

This software package is written and maintained by Richard van Hees (SRON), and distributed under the GNU General Public License.

E.2 Acknowledgement

In case (part of) this dataset is used for a publication and essential to the work and the results, an offer of co-authorship to Richard van Hees (SRON) and /or Ralph Snel (SRON) is highly appreciated. Please do not hesitate to contact us in an early stage for necessary support.

In all other cases SRON should be properly acknowledged. Such an acknowledgement could state: "We acknowledge SRON for providing the nadc_tools and/or patched Sciamachy level 1b data products".

E.3 DISCLAIMER

We at SRON have developed nadc_tools as part of our commitment towards the verification and further improvement of the calibration of Sciamachy data. This software is developed for in-house usage, and generously shared with you WITHOUT ANY WARRANTY. We will try to help you with any problems, but only on a "best effort" basis. In order to continuously improve this facility, feedback from users is highly appreciated.

Note that this software package is also not supported in any way by ESA or DLR, although the nadc_tools extractors and libraries mimic some of the functionality of the gdp and EnviView toolbox.

The software package is in no way intended to replace the official data processor and should not be treated as such. Especially it is not meant to (and cannot) produce official data products. These have to be derived with ESA approved tools such as EnviView that can be downloaded for free from ESA.

It is your own responsibility to verify the results you produce with this package against official data products. If you find discrepancies, please inform SRON (not ESA or DLR) at once. Any information you give will help us to improve our software. We will relay to ESA all information to improve the official data processor.

E.4 COPYRIGHT

Table 2: Listing of calibration options. First options giving the ATBD implementation, followed by options calling development/improved calibration as developed at SRON.

OPTIONS		SHORT DESCRIPTION (see also appendix C)
none		do not apply any correction (same effect as option **-lv1c**)
atbd		apply full calibration according to ATBD (=0,1,2,3,4,5,6,7,9,E)
0		Memory effect correction (*defaults to "0ml"*)
	m	Reticon memory correction (channel 1-5)
	l	Epitaxx non-linearity correction (channel 6-8), using no external source
1		Dark Current correction (*defaults to "1acv"*)
	a	Analog Offset correction
	c	(constant) Dark Current correction
	v	(variable) Dark Current correction
	s	Solar straylight correction
	L	*Limb MDS are corrected using the Limb-dark measurement*
	D	dark correction values extracted "Dark_Average" ADS
2		Pixel-to-Pixel Gain correction
3		Etalon correction
4		Spectral Straylight correction
5		Wavelength calibration
6		Polarization sensitivity correction
7		Radiance sensitivity correction
	k	radiance correction values calculated from keydata "key_radsens.h5"
8		Calculate reflectances: divide Earth spectrum with Solar spectrum (doppler shift corrected)
9		Apply Bad/Dead Pixel Mask
M		Apply m-factors from directory "m-factor_06.00" with auxiliary files
E		Calculate/estimate the total relative accuracy on the measured signal.
sron		apply full development/improved calibration (=0+,1+,2,3,4+,5+,6+,7+,9+,E)
0+		Memory effect/non-linearity correction from an auxiliary files with the name *MEMcorr.h5* and *NLcorr.h5* (*defaults to "0ml+"*)
1+		Dark Current correction derived by the SRON Sciamachy Monitoring Facility (*defaults to "1acv+"*)
2+		PPG correction derived by the SRON Sciamachy Monitoring Facility
4+		Improved straylight correction for channel 2 (*S. Slijkhuis et al*)
5+		Improved wavelength calibration (ch7/8), from J. Schrijver (SRON)
6+		Polarization calibration improvements, as suggested by J. Krijger (SRON). Polarization sensitivity correction values are read from an auxiliary file with the name *psp_patch.h5*
8+		Calculate reflectances use external database with Sciamachy Solar spectra *Not implemented*
9+		Apply Bad/Dead Pixel Mask derived by the SRON Sciamachy Monitoring Facility
M+		apply m-factors from file "m-factor_06.00.h5"
p		**[L1c]** multiply radiances in L1C product. The 8192 multiplication factors are read from auxiliary (ASCII) file. The path to this file must be passed using the environment variable SCIA_CORR_L1C.