NEONARVAL DATA FORMAT

A. LÓPEZ ARISTE

1. The names of files in NeoNarval

NeoNarval will use a standardized naming convention based upon the time of file creation, plus suffixes for identification. An example is the best way to illustrate this. Let us take the file
`Narval_20180314_003846_st0.fts`
The first section `Narval` identifies the instrument, which in this case corresponds to old data from Narval The next 2 parts give the date and time of observation. `20180314_003846` corresponds to an observation made by Narval on March, 14th 2018 at UT time 00:38:46. Actual times of exposure start and end are part of the metadata, so this date and time are just the ones of creation of the file and serve the purpose of creating an ordered sequence of files coming from the instrument.

Finally, `st0` identifies the mode of observation and the data level. `st` stands for a scientific *star* observation, and 0 means that this is level 0 raw data. Other observation modes are flat field (ff), bias `(bi)`, calibration with a Th-Ar lamp `(th)` or with a Fabry-Perot `(fp)`. These labels are kept at all data levels.

A given observation can be traced at all data levels and through reduction process because it keeps is radical `Narval_20180314_003846` at all times. The only exception to this rule begins in level 2, when 4 level-1 files are combined for polarimetric demodulation. The resulting level-2 file keeps the name of the first file in the sequence of 4, and the 3 other disappear, though their names are kept in the metadata of the level-2 file.

2. Definition of data levels for NeoNarval

- **Level 0**: Raw data. Filenames carry a 0 in the suffix before .fts. These are the files provided by the acquisition, before the DRS starts working on them.
- **Level 1**: Uncalibrated data. Filenames carry a 1 in the suffix before .fts. These files contain extracted spectra after application of the DRS onto the level 0 raw data. Wavelength is just an ordinal index, there is no demodulation for polarization applied. There are two intensity spectra, one per beam, for all orders. There is no normalization of the continuum. Only data containing scientific observations (`st1.fts`), or calibration series from the Th-Ar lamp (`th1.fts`) or Fabry-Perot (`fp1.fts`) reaches this level. Non-distributed
- **Level 2**: Calibrated data. Filenames carry a 2 in the suffix before .fts. Wavelength calibration has been applied and continuum has been normalized. Polarimetric demodulation is performed, so that the spectra in the main data extension are intensity, polarization, null profiles and errors, as well as the continuum that served to normalize the spectra. This data may be distributed.

- **Level 3**: Filenames carry a 3 in the suffix before .fts. LSD routines have been applied to the level 2 data and the result added to the level 2 file in a new data extension. This data is distributed.

3. Structure of FITS files

3.1. **General structure.** The format of FITS files is structured in extensions. At least one extension (called primary) is required. Each extension is made of a header and a data section. The header contains the metadata. This metadata goes from basic information about the number of extensions or the size of the data in the data section, to information about the observation: object, observer's name, time of the observation, instrument status, telescope pointing, etc. The most basic parts of these metadata were present also in the raw FITS files of Narval, but were lost, or scattered through accompanying log files, in the reduced Narval files.

The primary extension is the one with the largest amount of metadata. It contains all the information related to the conditions of the observation. All secondary extensions carry only information concerning the actual data contained in its data extension.

3.2. **Level 0 files.** There is only one (primary) extension. The header contains the main basic information from the telescope, instrument, and object observed. The data section contains a binary integer array of size 4096×4096 with the image from the CCD.

A secondary extension is foreseen containing as data 2 columns with time and the reading of the photometer during the acquisition.

3.3. **Level 1 files.** The primary extension contains just the main header. No data section. The following header keys are added: `FLATNAME`, `BIASNAME`, `PSFNAME` , `PSFWIDTH`, `SUPERRES`, and `LEVEL1` The 2nd extension contains as data 4 columns labelled `Wavelength1`, `Beam1`, `Wavelength2`, and `Beam2`. Wavelength may just be and index array. While `Beam1` and `Beam2` columuns contain floats representing the extracted intensity in the 2 beams for all orders. The header of this 2nd extension containes, other than the compulsory formatting keys, the following keys:

The foreseen 3rd extension carries out the photometer data from Level 0

3.4. **Level 2 files.** The primary extension contains just the main header. No data section. The following header keys are added to those of the primary section in Level 1: `FILE1`, `FILE2`, `FILE3`, `FILE4`, and `LEVEL2`

The 2nd extension contains as data 6 floating point columns labelled: `Wavelength1`, `Intensity`, `Polarization`, `Null1`, `Null2`, `Error` and `Continuum`. Wavelength is in Angstroms. Intensity, polarization and the null profiles are obtained after demodulation. Error is computed as the standard deviation of the null profiles in a small range around each wavelength bin. The intensity is normalized, and the 7th column, `Continuum`, carries the continuum factor that has been applied to every wavelength bin to the extracted spectra to normalize them. The following keys are added to the header of this 2nd extension:

The foreseen 3rd extension carries out the photometer data from Level 0

3.5. **Level 3 files.** The first 3 extensions are identical to Level 2 files. The following keys have been added to the primary header: LEVEL3

The 4th extension contains the result of performing LSD on the extracted spectra. The data is made of 6 floating point columns: `Velocity`, `Intensity`, `Weight1`, `Weight1`, `Weight1`, and `Weight1`. The `Velocity` column contains the velocity bins of the LSD profile in km/s. The `Intensity` column the LSD intensity, computed using as weight the line depression as retrieved from the mask file. The other columns contain the LSD result with different weighing schemes: 1,2,3, and 4. respectively. The name of the mask is given in the header of this extension as key `MASK`. Other added keys in this header are: `NBLINES` , the number of lines from the mask used for the LSD, and `WEIGHTS` which contains the description of the weights used in the LSD. The 5th extension contains velocimetry results (to be described).

4. Reading examples in Python

As a help for adapting codes to the new file formats we give here a few basic examples on how to read NeoNarval files. We have selected Python as the language for these examples for 2 reasons: it illustrates the power and simplicity of the FITS format when handled with widely available reading libraries, and second, the DRS is written in Python 2.7 and the sources are freely available, including the FITS writing sections.

The standard FITS library is freely available and included in the astropy package

```
> import astropy.io.fits as pyfits
```

The alias `pyfits` is just for memory of the old name of the package before it got included in `astropy`. Let us suppose we want to read a NeoNarval FITS file `Narval_20180314_003846_st0.fts`.

```
> a=pyfits.open('Narval_20180314_003846_st0.fts')
```

The object `a` has the shape of a list of structures, one per extension. Hence `a[0]` refers to the primary extension. We can now read the metadata and data in this extension

```
> header=a[0].header
> img=a[0].data
```

The header is loaded in the list `header`. A given key, for example `OBJECT`, can be readily accessed as `header['OBJECT']` which will be a string with the name of the observed star. `header[25]` will access the key in position 25, whatever it is. Since the opened file is a level 0 file, `img` will contain an integer array of size 4096×4096 that we can show on the screen or use as a mathematical matrix for data reduction. After reading, the file object is closes with `a.close()`

At level 1, the structure of the FITS file changes, but we open identically

```
> a=pyfits.open('Narval_20180314_003846_st1.fts')
```

The header in the primary extension `a[0].header` behaves as before, in level 0, and so do extensions 2 and 3 (`a[1].header` and `a[2].header`). However, there is no data in the primary section:

```
> img=a[0].data
> print(img.shape)
[]
```

Data in level 1 (and 2 and 3) is contained in extensions 2 and beyond in table form. An easy way to read them is to keep in memory the name of the columns and do for example:

```
> wvl=a[1].data['Wavelength1']
> sp= a[1].data['Beam1']
```

This will convert `wvl` and `sp` into 2 arrays of the same shape containing respectively wavelength (which at level 1 is just an index) and the extracted intensity spectra from beam 1. All formatting is taken care by the FITS library, and we do not even care about how many columns there are in the data section, as long as we remember the labels, names, of each column.

Thus, at level 2, we can recover the wavelength in Angstroms and the Stokes parameter as

```
wvl=a[1].data['Wavelength1']
sp= a[1].data['Polarization']
```

and do this independently of the fact whether a column labelled `Continuum`, for instance.

Once `wvl` and `sp` have been read in that way, we can plot them using, for example, pyplot:

```
plt.plot(wvl,sp)
plt.show()
```

In level 3, the LSD data is found in extension 3, so to read it one can do:

```
> a=pyfits.open('Narval_20180314_003846_st3.fts')
vel=a[1].data['Velocity']
V= a[1].data['Stokes']
```

and then plot vel and V as for the example above.

If one happens to forget the labels that identify each column, whether in level 2 or 3, whether in extensions 2, 3, 4 etc, one can recover the info automatically with

`liste=a[2].data liste.columns.info()`

4.1. **For Python newbies.** Python is already installed or easily installable in Linux and Mac OS X systems. In Windows systems, its installation may be easier by installing a development environment like Spyder or Anaconda.

Python has many ways of being used. For users evolving from IDL, the easier way is just to call it in the command line:

```
#> python
```

This leads us into a command line shell where python orders can be just typed:

```
>print('Hola')
Hola
> 3+2
5
```

If you have been provided with a python code, you just call it as an argument. For example, to run the DRS do

```
#> python DRS_step0.py
```

Coming back to the python command shell, you may want to plot data. Python, like C, is provided with nothing for plotting or even doing math beyond the basic operations.

As in C, you must start by loading a few useful libraries:
`> import numpy as np`
This allows us to import the numpy library, we tell python that we shall refer to it by the alias `np`. We can now do some more interesting math:
`> np.sqrt(2.)`
`1.4142135623730951`
We also want to plot spectra. There are many libraries available for that, but one that works in an intuitive manner for users coming from Matplot or IDL is matplotlib, in particular its package pyplot:
`> import matplotlib.pyplot as plt`
You are now ready to load the library to read FITS files, read NeoNarval files, plot them and do math, as in the examples shown above in this section.

The web is full of examples, help and python libraries to do almost anything you want to do. You can google for example `python plot 1D` in your browser to get some nicely written tutorials, or information about option, or examples on how to do anything to your plots.

5. FITS Keywords

Extension 1, all levels

- SIMPLE = T / conforms to FITS standard
- BITPIX = 8 / array data type
- NAXIS = 0 / number of array dimensions
- EXTEND = T
- COMMENT FITS (Flexible Image Transport System) format defined in Astronomy and
- COMMENT Astrophysics Supplement Series v44/p363, v44/p371, v73/p359, v73/p365.
- COMMENT Contact the NASA Science Office of Standards and Technology for the
- COMMENT FITS Definition document #100 and other FITS information.
- LONGITUD= 0.1333 / [Deg E] Longitude
- LATITUDE= 42.9333 / [Deg N] Latitude
- OBSELEV = 2869.4 / [m] Altitude
- OBJECT = 'PI1UMA ' / Observed Object Name
- OBSTYPE = 'STELLAR ' / [CIEL,FOCUS,CALIB,TEST] Observation Type
- SPECTYPE= 'G1.5VbC2001MNRAS.328...45M' / Spectral type of the object
- MAGB = 6.26 / B Magnitude of object
- MAGV = 5.64 / V Magnitude of object
- CMTOBS1 = 'Observateur' / [ ]
- CMTOBS2 = '———————' / [ ]
- CMMTOBS = ' ' / Commentaire Observateur
- CMTOBS = ' ' / Commentaire Observateur
- ORIGINE = 'Bernard Lyot (2m)' / France-TBL Telescope
- RUN_IDEN= 'L162N08 ' / Identifiant pre-traitement
- PI_NAME = 'PETIT ' / Responsable Mission

- OBSERVER= 'qos ' / Intervenant
- CMTIMA1 = 'Image Data'
- CMTIMA2 = '——————————'
- INSTRUM = 'NARVAL ' / Instrument Name
- INSTMODE= 'Polarimetry, R=65,000' / [POL3/SPEC3/SPEC6] Instrument Mode + Slices
- EREADSPD= 'Normal: 5.0e noise, 1.35e/ADU, 40s' / [NORMAL/FAST/SLOW/XSLOW] CCD Re
- FILENAME= '230616o.fits' / Base Filename Acquisition
- PATHNAME= 'L162N08/02dec16/POL3/Normal/V' / Original Directory name Acquiistion
- TIMESYS = 'UTC '
- TIME = 'hrs:mn:sec'
- DATE-OBS= '2016-12-03'
- DATE = '2016-12-03T05:28:54.977' / Data Observation
- TIME-OBS= '05:28:54.977' / Begin Exposure
- TIMEEND = '05:33:56.731' / End Exposure
- HI_CUT = -999.9 / Hight Cut visualisation
- LO_CUT = -999.9 / Low cut visualisation
- EXPTIME = 300. / [sec] Integration time
- CMTTPL1 = 'Template Information'
- CMTTPL2 = '——————————'
- CMMTSEQ = 'V exposure 1, sequence 1 of 1' / Verbal Sommary Template
- A_IDENT = 'STELLAR_55' / Automatic serie of sequence Identification
- A_IDSEQ = 'V ' / [Q,V,W,I,FLAT,BIAS....] Name of running sequenc
- A_NUMSEQ= 1 / Number of the sequence in the serie
- A_NBSEQ = 1 / Total amount of same sequences in the serie
- A_IDFILE= 1 / Number of the file in the sequence
- CMTCCD1 = 'Detecteur Information'
- CMTCCD2 = '——————————'
- PIXSIZE1= 0.000135 / [um] Size pixel along axis X
- PIXSIZE2= 0.000135 / [um] Size pixel along axis Y
- CTYPE1 = 'PIXEL ' / Pixel coordinate system
- CTYPE2 = 'PIXEL ' / Pixel coordinate system
- CRVAL1 = 1049. / [pixel] Centre rotation axis X
- CRVAL2 = 2306. / [pixel] Centre rotation axis Y
- CDELT1 = 1. / Binning factor along x
- CDELT2 = 1. / Binning factor along y
- CRPIX1 = 1049. / [pixel] Value of X ref. pixel
- CRPIX2 = 2306. / [pixel] Value of X ref. pixel
- D_TEMP1 = -110. / [Deg] Ccd temperature
- D_GAIN1 = 1.35 / [e/adu] Ccd readout gain
- D_SPEED1= 40. / [seconds] Ccd readout time
- D_BRUIT1= 5. / [ev] Ccd readout noise
- CMTTEL1 = 'Telescope + Bonnette Information'

- CMTTEL2 = '—————————'
- DATE_JUL= 2457726.
- LST-OBS = '10:19:21.044'
- POSTO_RA= 129.799
- POSTO_DE= 65.021
- POSTO_EQ= 2000.
- ENCOD_RA= 24.457
- ENCOD_DE= 65.161
- RA_DEG = 130.173
- DEC_DEG = 64.956
- COORD_EQ= 2016.924
- RA = '8:40:41.55'
- DEC = '64:57:20.4'
- AIRMASS = 1.112
- FOCUSPOS= 3181. / [Volts] Focalisation Z cassegrain
- BONANGLE= 0.09
- ROTANGLE= 34.28
- XPROBE = -999.
- YPROBE = 38.38
- ZPROBE = 1.
- T_HYD1_B= 0. / [%] Pre-Pilote Hydrometrie Exterieur
- T_HYD1_E= 0. / [%] Post-Pilote Hydrometrie Exterieur
- T_HYD2_B= 0. / [%] Pre-Pilote Hydrometrie interieur
- T_HYD2_E= 0. / [%] Post-Pilote Hydrometrie interieur
- T_TEM1_B= 0. / [DegC] Pre-Pilote Temperature miroir
- T_TEM1_E= 0. / [DegC] Post-Pilote Temperature miroir
- T_TEM2_B= 0. / [DegC] Pre-Pilote Temperature exterieur
- T_TEM2_E= 0. / [DegC] Post-Pilote Temperature exterieur
- CMTGUI1 = 'Guidage Information'
- CMTGUI2 = '—————————'
- GUINAME = 'L041T05gui.fits'
- GUIMAGN = -999.
- GUIFWHMX= -999.
- GUIFWHMY= -999.
- GUIFLUX = -999.
- CMTPOL1 = ' Polarimetrie Information'
- CMTPOL2 = '—————————'
- ADCSTATE= 'IN ' / [IN/OUT/FAULT] Polarimetre Axe-I Tiroir
- ADC1BEG = 214.0439 / [x] Position ADC1 Pre-Acquisition
- ADC2BEG = -57.71906 / [x] Position ADC2 Pre-Acquisition
- ADC1END = 211.627 / [x] Position ADC1 Post-Acquisition
- ADC2END = -55.76578 / [x] Position ADC2 Post-Acquisition
- P_NAME1 = 'TROU ' / [x] Axe-C Roue de Calibration
- P_NAME2 = 'P1 ' / [x] Axe-D Fabry-Perot
- P_NAME3 = 'P5 TP: 35020' / [x] Axe-E Rhomboedre 1

- P_NAME4 = 'P4 TP: 15080' / [x] Axe-F Rhomboedre 2
- P_NAME5 = 'Wollaston' / [IN/OUT/FAULT] Axe-G Wollastone/Wed
- P_NAME6 = '345 ' / [Magnitude] Axe-H Roue a Densite
- P_TEMP1 = -2.1192 / [DegC] Sonde Temperature Tiroir Rhom2
- CMTCAL1 = ' Calibration Information'
- CMTCAL2 = '————————'
- CALSTATE= 'OFF ' / [ON/OFF] Power Boite de Calibration
- C_LAMP1 = 'OFF Flux_0' / [ON/OFF - 1,12] Halogen Lamp Red 12v/50w
- C_LAMP2 = 'OFF Flux_0' / [ON/OFF - 1,12] Halogen Lamp Blue 12v/50w
- C_LAMP3 = 'OFF ' / [ON/OFF] Hallow Cathode Lampe 1
- C_LAMP4 = 'OFF ' / [ON/OFF] Hallow Cathode Lampe 2
- C_TEMP1 = -3.8725 / [DegC] Temperature Boite Calibration
- CMTSPC1 = 'Spectrometrie Information'
- CMTSPC2 = '————————'
- EEMSHUT = 'OFF ' / [ON/OFF] Shutter APD-Exposemetre
- EEMSTATE= 'OFF ' / [ON/OFF] Power APD-Exposemetre
- EMSMAGN = -999. / Magnitude Exposemetre
- EEMCNTS = '-999.0 ' / RMS Exposemetre
- AGITSTAT= 'OFF ' / [ON/OFF] Power Agitateur
- S_LAMP1 = 'OFF ' / [ON/OFF] Lampe de Calibration
- S_NAME1 = '-4.838 ' / [x] Axe-A Focus Camera
- S_NAME2 = 'OPEN ' / [x] Axe-B Hartman
- S_NAME3 = '3SLICES ' / Axe-C Slicer-Bench Verin
- S_NAME4 = 'HOME ' / Axe-D Tiroir Lampe de calibration
- S_NAME5 = '3SLICES ' / Axe-E Slicer
- S_NAME6 = '3SLICE_POL' / Axe-F Decker
- ERHSPBEG= 37.2132 / [%] Pre-Acq Hygrometrie Spectro
- ERHSPEND= 36.5642 / [%] Post-Acq Hygrometrie Spectro
- EPRSPBEG= 8.0354 / [mbar] Pre-Acq Pression Diff Spectro
- EPRSPEND= 8.0293 / [mbar] Post-Acq Pression Diff Spectro
- ETSP1BEG= 12.0107 / [DegC] Pre-Acq Temp. Down Mirroir Spectro
- ETSP1END= 12.0758 / [DegC] Post-Acq Temp.Down Mirroir Spectro
- ETSP2BEG= 11.7016 / [DegC] Pre-Acq Temp. Camera Spectro
- ETSP2END= 11.6978 / [DegC] Post-Acq Temp.Camera Spectro
- ETSP3BEG= 11.9714 / [DegC] Pre-Acq Temp. Up Mirroir Spectro
- ETSP3END= 11.9906 / [DegC] Post-Acq Temp. Up Mirroir Spectro
- ETSP4BEG= -999. / [DegC] Pre-Acq Temp. Cryostat Spectro
- ETSP4END= -999. / [DegC] Post-Acq Temp.Cryostat Spectro

Extension 1,keys added in Levels 1,2 and 3:

- FLATNAME= 'Narval_20161202_152101_fl1.fts' Name of the file used for the flat
- BIASNAME= 'Narval_20161202_151920_b10.fts' Name of the file used for the bias
- PSFNAME = 'Narval_20161202_153059_th0.fts' Name of the file used to extract the PSF
- PSFWIDTH= 1 Number of pixels at the sides of the central column in the PSF

- SUPERRES= 2.0 Over-binning of the detector pixels
- LEVEL1 = 'Narval_20161203_052854_st1.fts' Name of the level 1 file

Extension 1, keys added in Levels 2 and 3:

- FILE1 = 'Narval_20161203_052854_st1.fts' Name of the 1st level-1 file in the sequence
- FILE2 = 'Narval_20161203_053454_st1.fts' Name of the 2nd level-1 file in the sequence
- FILE3 = 'Narval_20161203_054102_st1.fts' Name of the 3rd level-1 file in the sequence
- FILE4 = 'Narval_20161203_054704_st1.fts' Name of the 4th level-1 file in the sequence
- LEVEL2 = 'Narval_20161203_052854_st2.fts' Name of the level 2 file

Extension 1, key added at Level 3

- LEVEL3 = 'Narval_20161203_052854_st3.fts' Name of the level 3 file

Extension 2, Level 1

Extension 2, Levels 2 and 3

- XTENSION= 'BINTABLE' / binary table extension
- BITPIX = 8 / array data type
- NAXIS = 2 / number of array dimensions
- NAXIS1 = 48 / length of dimension 1
- NAXIS2 = 337127 / length of dimension 2
- PCOUNT = 0 / number of group parameters
- GCOUNT = 1 / number of groups
- TFIELDS = 6 / number of table fields
- TTYPE1 = 'Wavelength1'
- TFORM1 = 'D '
- TUNIT1 = 'pixels '
- TTYPE2 = 'Intensity'
- TFORM2 = 'D '
- TUNIT2 = 'ADU '
- TTYPE3 = 'Polarization'
- TFORM3 = 'D '
- TUNIT3 = 'Ic '
- TTYPE4 = 'Error '
- TFORM4 = 'D '
- TUNIT4 = 'Ic '
- TTYPE5 = 'Null '
- TFORM5 = 'D '
- TUNIT5 = 'Ic '
- TTYPE6 = 'Continuum'
- TFORM6 = 'D '
- TUNIT6 = 'ADU

Extension 3, Level 3

- XTENSION= 'BINTABLE' / binary table extension

- BITPIX = 8 / array data type
- NAXIS = 2 / number of array dimensions
- NAXIS1 = 32 / length of dimension 1
- NAXIS2 = 201 / length of dimension 2
- PCOUNT = 0 / number of group parameters
- GCOUNT = 1 / number of groups
- TFIELDS = 4 / number of table fields
- TTYPE1 = 'Velocity'
- TFORM1 = 'D '
- TUNIT1 = 'km/s '
- TTYPE2 = 'Intensity'
- TFORM2 = 'D '
- TUNIT2 = 'normalized to continuum'
- TTYPE3 = 'Stokes '
- TFORM3 = 'D '
- TUNIT3 = 'normalized to continuum'
- TTYPE4 = 'Null '
- TFORM4 = 'D '
- TUNIT4 = 'normalized to continuum'
- MASK = 'mask.6000.45.p00.correct' Name of the mask used for LSD
- NBLINES = 7658 Number of spectral lines in the mask added during the LSD
- WEIGHTS = 'g ' Description of the weighting scheme used for LSD