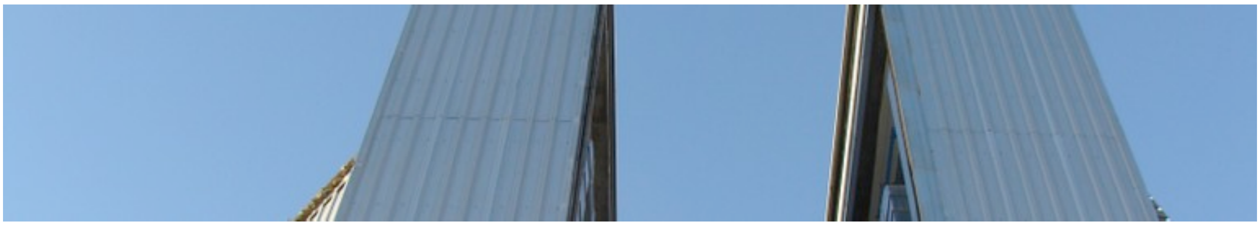

HARPS-N

DRS User Manual

Doc. Nr. OG-MAN-HAN-13-0004

Issue 1.1

August 16th, 2012

Prepared D.Sosnowska, C. Lovis …..……………..…………........………
Name Date Signature

Approved C. Lovis……..………..…..……..............……………………........…………..
Name Date Signature

Released F. Pepe…………..………..……………………….........……………...……………
Name Date Signature

Change Record

Issue/Rev.	Date	Section/Page affected	Reason/Remarks
1.0	27.07.2012	All	First Version
1.1	16.08.2012	All	Christophe's corrections

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Introduction

HARPS-N is a copy of the ultra-precise radial-velocity spectrograph HARPS developed by a consortium headed by the Geneva Observatory (hereafter **OG**) located on the 3.6-m ESO telescope at the La Silla Observatory in Chile and it is used to follow-up, by radial velocity, the "hot" candidates delivered by the Kepler satellite. Given the known period and phase, and the extreme Doppler-precision of HARPS-N, it will be possible, for the first time, to confirm and characterize Earth-mass planets. The project, as well as the present contract, is lead and conducted by OG. The HARPS-N is installed on the TNG at the Observatory on La Palma Island.

The software for HARPS North is developed as a collaboration between Geneva Observatory, ATC and TNG.

1.2 Scope of the Document

This document describes how to use properly the HARPS-N Data Reduction SW. The DRS is described in [AD-2].
The FITS keywords names have to follow the HARPS-N Dictionary [AD-1].

1.3 Documents

1.3.1 Applicable Documents

AD-1	OG-DID-HAN-13-0002	HARPS-N Dictionary	2.0	14.04.2011
AD-2	OG-TRE-HAN-51-0001	Data Reduction Software Design Report	0.1	21.02.2011

1.3.2 Applicable Drawings

AD-3				
AD-4				

1.3.3 Reference Document

RD-1				
RD-2				

1.3.4 Reference Drawings

RD-3				
RD-4				

1.4 Acronyms

AD	Applicable Document
CCD	Charge-Couple Device (detector)
HCA	Harvard Center for Astrophysics
ESA	European Space Agency
ESO	European Southern Observatory
EUR	Euro
FA	Final Acceptance
FDR	Final Design Review
FRD	Focal-Ratio Degradation
HARPS	High-Accuracy Radial-velocity Planet Searcher
HU	Harvard University
ODR	Optical Design Review
OG	Observatoire de Genève
PA	Provisional Acceptance
PDR	Preliminary Design Review
PO	Purchase Order
NEF	Northern Earth Facility
RD	Reference Document
SOW	Statement of Work
TBC	To Be Confirmed
TBD	To Be Defined
TNG	Telescopio Nazionale Galileo (La Palma, Canary Islands)
TRS	Technical Requirements Specification
WHT	William Herschel Telescope (La Palma, Canary Islands)

Chapter 2: DRS Hardware and Software Environment

2.1 Overview

The DRS runs on-line on a dedicated Linux Workstation `drs32.hn`. The DRS deals with all the aspects of the scientific reduction of the raw data, as well as the processing of these reduced data to extract the radial velocities. The DRS does not interfere with the operation of the instrument, i.e. observations are independent from the Data Reduction activity.
The DRS is designed to run automatically (no user interaction) like a batch process controlled by the Trigger right after the end of each exposure or at the completion of a sequence of exposures. It is also possible to use a DRS off-line to display and analyze reduced data.

2.2 Architecture of the data on reduction machine (drs32.hn)

The raw frames are automatically stored on the directory: `/data/raw/YYYY-MM-DD/` where `YYYY-MM-DD` is the night directory automatically created at noon. All these directories are automatically created by the DFS system at noon.
The reduced frames are automatically stored in the directory: `/data/reduced/YYYY-MM-DD/` .
The calibration frames are copied in the calibration Data Base directory: `/data/calibDB/` .
The log files of the DRS are stored in the directory: `/data/msg/` .

2.3 Execution of the On-line DRS

On-line DRS is executed by a set of system commands (recipes) sent automatically by the Trigger. These recipes need two parameters:
- night directory name [YYYY-MM-DD]
- raw frame name(s) [HARPN.YYYY-MM-DDTHH-MM-SS.SSS.fits]
(for some recipes a list of raw frame names is needed)
For example, the command:

```
cal_loc_ONE_harpn.py 2002-02-11 HARPN.2002-02-11T20-13-45.768.fits
```

reduces the raw frame `HARPN.2002-02-11T20-13-45.768.fits` with the reduction program `cal_loc_ONE_harpn`. The raw frame is read in the directory `/data/raw/2002-02-11/` and all DRS products are stored in the directory `/data/reduced/2002-02-11/`. The logs of the DRS are stored in the file `/data/msg/DRS-drs32.2002-02-11`. The list of the raw frames obtained on a night is stored in the file `/data/msg/2002-02-11.r`.
The reduction programs are executable files (actually python scripts) that contain all relevant information to carry out the reduction. No other parameters or options are needed for the reduction. Parameters specific to the exposure are in the FITS descriptors of the raw frame.
The Trigger runs automatically the appropriate reduction program of each frame or set of frames

as soon as the exposure is archived and available on the DRS machine. The reduction programs associated to each template types are listed in the Table 1.
The on-line trigger is executed on the `drs32.hn` workstation (under user harpn) with the command: `trig.csh online`

Exposure type	Templates HARPN_ech_[...]	Reuction program (Recipes)
Calibration		
CCD BIAS	cal_bias	cal_BIAS_harpn.py
CCD DARK	cal_dark	cal_DARK_harpn.py
Geometry of orders	cal_tun	cal_loc_ONE_harpn.py
Flat field sequence	cal_tunAB	cal_FF_harpn.py
ThAr-ThAr wavelength calibration	cal_thoAB	cal_TH_harpn.py
ThAr-FP wavelength calibration	cal_waveAB	cal_WAVE_harpn.py
Science Observations		
Accurate RV measurement (ThAr)	acq_thosimult	obj_TH_harpn.py
Accurate RV measurement (FP)	acq_wavesimult	obj_WAVE_harpn.py
Spectroscopy for object only	acq_objA	obj_ONE_harpn.py
Spectroscopy object and sky	acq_objAB	obj_TWO_harpn.py

Table 1 List of on-line data reduction programs

2.4 Execution of the Off-line DRS

The Off-line DRS is used to display and analyze reduced data. Off-line DRS is executed by a set of system commands (recipes) send manually through a dedicated GUI or directly from the prompter.
The reduction programs associated to reduced data are listed in the Table 2.

Description	Reduction program
Visualization	
Display one order of the E2DS spectrum	off_visu_e2ds_harpn.py
Display a domain of the S1D spectrum	off_visu_s1d_harpn.py
Display SNR per orders	off_visu_SN_harpn.py
Display the CCF and its parameters	off_visu_ccf_harpn.py
Display the RV per orders	off_visu_rvo_harpn.py
Radial velocity re-computation	
Compute the CCF	off_make_ccf_harpn.py

Table 2 List of off-line data reduction programs

2.5 Programming language

The programming language is Python, a powerful, object-oriented, interpreter programming language that is easy to extend, freely distributed, and available for most computer platforms (see `http://www.python.org/` and `http://www.vex.net/parnassus/`).

The DRS needs the following python modules:

- Mathematical and Numerical (Numeric/numpy)
- Graphical and visualization (Gnuplot)
- FITS format manipulation (pcfitsio, fitsio)
- User interface (Tkinter)
- System and files (sys, time, shutil, os)
- String manipulation (string)
- Fortran program interface (f2py)

Most of these modules are part of the python 2.4 distribution version. DRS is currently running on this version.
Some specific algorithms of the DRS are written in Fortran and C in order to increase the DRS execution.
They are included in Python library through the Fortran program interface `f2py`.

2.6 Architecture of the DRS

All the directories and files related to the Data Reduction Software of HARPS are stored in the directory: `/home/hanmgr/INTROOT/DRS_HARPN/`
From this point:

`./config`	Contains all the Instrument Configuration Data files used by the DRS.
`./fortran`	Contains all the fortran sources `name.f` and their associated python modules `namemodule.so`.
`./C`	Contains all the C sources `name.c`
`./python`	Contains the python executable *python.csh* and two initialization file *startup.py* and *startup_recipes.py*.
`./python/f2pymodule`	Contains all the modules based on fortran code and their test python scripts.
`./python/C-modules`	Contains all the modules based on C code and their test.
`./python/Recipes`	Contains all the python reduction programs.
`./python/Modules`	Contains all the python modules used by the reduction programs.

2.7 Architecture of the DRS Modules

All the functions used by the reduction programs are grouped in modules related to a specific application. Table 3 describes all the modules used by the Data Reduction programs and their field of application.

Module name	**Description**
hadmrBIAS	BIAS measurement and correction
hadmrCDB	Calibration Database access functions
hadmrDARK	DARK measurement and correction functions
hadmrEXTOR	Extraction of orders
hadmrFITS	Manipulate FITS
hadmrFLAT	FLAT FIELD measurement and correction functions
hadmrLOCOR	Localization of orders

hadmrRV	Calculation of velocity (Earth, drift, stellar)
hadmrTHORCA	Wavelength calibration
hadrgdCONFIG	Configuration Panel Function of the RGD
hadgtVISU	Graphical functions
hadgtMATH	Mathematical functions

Table 3 List of the modules used by the reduction programs

Chapter 3: On-line DRS Description

3.1 Overview

The On-line Data Reduction is automatically executed with the *Trigger*, which can be started by typing the command: `trig.csh online` on the `drs32.hn` machine (under user harpn).

3.2 Recipes

Recipes are made of specific functions available in the Modules specifically developed for the HARPS DRS (see Table 3) or part of the distribution of python modules. Dependencies between recipes and modules can be found in the DRS design document. We present in this chapter a description of the reduction task carried out by each recipe.

3.2.1 cal_BIAS_harpn.py - CCD BIAS

Inputs
Raw FITS frame obtained with the HARPN_ech_cal_bias template
Description
- Read keywords related to the CCDs parameters (readout mode, readout noise, gain)
- Correction of the bad columns of the CCDs (average of adjacent columns)
- Measurement of the mean level and the dispersion of the 4 overscan areas
- Measurement of the mean level and the dispersion of the 4 CCD ports
- Print and Display of the results
- Store results on the file `cal_BIAS_result.tbl`
Outputs
- ASCII file `cal_BIAS_result.tbl`
Quality control
- Quality control with warning message when bias level > 500 ADU or bias noise > 10 e-

3.2.2 cal_DARK_harpn.py - CCD Dark

Inputs
Raw FITS frame obtained with the HARPN_ech_cal_dark template
Description
- Read keywords related to the CCDs parameters (readout mode, readout noise, gain)
- Correction of the bad columns of the CCDs (average of adjacent columns)
- Mean level and dispersion of the Bias on the 4 overscan areas
- Mean level and dispersion of the 4 CCD ports with sigma clipping of the cosmic hits
- Calculation of the mean dark level (e-/hour)
- Calculation of the number of cosmic events (event/cm2/mn)
- Store result on the file `cal_DARK_result.tbl`
Outputs

ASCII file `cal_DARK_result.tbl`
Quality control
- Error message if the exposure time is shorter than 5 minutes, DRS stopped.
- Warning if dark level > 10 e-/hour or cosmic events > 10 event/cm2/mn

3.2.3 cal_loc_ONE_harpn.py - Geometry of orders of One fiber

Inputs
Raw FITS frame obtained from a Tungsten exposure with the HARPN_ech_cal_tunA or HARPN_ech_cal_tunB template for each fiber A and B.
Description
- Retrieve from calibDB previous last full calibration sets - Read keywords related to the CCDs parameters (readout mode, readout noise, gain)
- Read keywords related to the exposure type in order to determine the illuminated fiber
- Correction of the bad columns of the CCDs (average of adjacent columns)
- Correction of the BIAS determined in the 4 overscan areas. The bias is determined for each of the 4096 rows by average of the 50 columns. This scheme allows to conserve and correct the possible structure of the bias when the CCDs are illuminated.
- Resize of the raw frame and cut through all the spectral orders on the central row of the CCD
- Renormalization the central row in order to put all orders at the same level
- Find all orders in the normalized central row greater than 0.15. This first step allows to find the 69 orders of fiber A or B. From these starting points order position are searched and located by 20 pixel steps. At each point, profile of each order is fitted by a gaussian in order to measure its center and FWHM. On the whole frame centers and FHWM of each order are determined on 200 points.
- A 4th degree polynome is fitted for each order to constrain the center and to measure its FWHM. The typical RMS of the fit is 25 mpixels for the centering (75 mpixels for the FWHM). The FWHM ranges of orders from 3 to 4 pixels.
- The position x of the center of each orders for each rows y is stored in a FITS file with the suffix `loco`
- The FWHM of each order for each row y is stored in a FITS file with the suffix `fwhm-order`
- Parameters of 3 orders are appended in the file `cal_loc_ONE_result.tbl`
- Quality control on the number of orders identified, dispersion of the center and FWHM.
- If passed, the Quality Control updates the Calibration Data Base.

Outputs
ASCII file `cal_loc_ONE_result.tbl`
Fits files [generic_name]`_loco_A.fits` and [generic_name]`_loco_B.fits`
Fits files [generic_name]`_fwhm-order_A.fits` and [generic_name]`_fwhm-order_B.fits`
Quality control
- Error if flux level on the central row (65000 ADU > Flux > 15000 ADU), DRS stops
- Quality control on the number of orders, dispersion of the fit both on center and FWHM values of order profiles ([pix]). If Quality control fails calibDB is not updated.

3.2.4 cal_FF_harpn.py - Flat-field measurement

Inputs
Several raw FITS frames obtained from a sequence of Tungsten exposures on the two fibers

with the HARPS_ech_cal_tunAB template or HARPS_ech_cal_tunUSER (at least 5 frames in order to reach the photon noise level above the flat-field noise).

Description

- Retrieve from calibDB previous last full calibration sets - Sum of the raw frame delivered
- Correction of the bad columns of the CCDs (average of adjacent columns)
- Correction of the BIAS as described previously
- Read the last localization in the Calibration Data Base
- Horne optimum extraction of orders
- A box window is used to smooth the tungsten flux along the orders to determine a "pseudo"-blaze response.
- Divide the extracted tungsten spectrum by this blaze response to obtain the flat field spectrum.
- Computation of the SNR at the blaze center and the dispersion of the flat field
- The Blaze for each order is stored in a FITS file with the suffix `blaze`
- The Flat for each order is stored in a FITS file with the suffix `flat`
- Parameters of 3 orders are appended in the file `cal_FF_result.tbl`
- If passed Quality control update the Calibration Data Base.

Outputs

ASCII file `cal_FF_result.tbl`
Fits files [generic_name]`_blaze_A.fits` and [generic_name]`_blaze_B.fits`
Fits files [generic_name]`_flat_A.fits` and [generic_name]`_flat_B.fits`

Quality control

- Check saturation level, stop DRS if saturated
- Check FF parameters (rms [e-]; S/N), CalibDB is not updated if failed

3.2.5 cal_TH_harpn.py / cal_WAVE_harpn.py - Wavelength Calibration

Inputs

Raw FITS frame obtained with the HARPN_ech_cal_thoAB or HARPN_ech_cal_waveAB template (ThAr on both fibers or ThAr/FP)

Description

- Retrieve from calibDB previous last full calibration sets - Correction of the bad columns of the CCDs (average of adjacent columns)
- Correction of the BIAS as described previously
- Fit on raw frame a small set of thorium lines and save parameters in a tbl file with suffix `spot_thAB`
- Read the last localization and flat field in the Calibration Data Base
- Horne optimum extraction of orders without cosmic rejection
- Flat field correction
- Compute the RV drift from the last ThAr Calibration in the Calibration Data Base.
- Identify lines using reference files from the DRS configuration directory
- Fit each thorium emission line - Adjust a polynomial solution for each order with a sigma-clipping scheme
- Compute the Littrow first and second order deviation and computes the granulation of the global solution - Save E2DS Thorium spectrum in FITS file with the suffix `e2ds` with all descriptors.
- Save an image of the wavelength solution (wavelength of each orders for each pixels) in FITS file with the suffix `wave`.
- Parameters of the Thorium wavelength calibration are appended in the file

cal_TH_result.tbl.
- Listing of all Thorium lines detected is stored in an rdb file with suffix lines.
- If passed Quality control update the Calibration Data Base.

Outputs

ASCII file cal_TH_result.tbl
ASCII file [generic_name]_spot_thAB.tbl
ASCII file [generic_name]_lines_A.rdb
ASCII file [generic_name]_lines_B.rdb
FITS files [generic_name]_e2ds_A.fits and [generic_name]_e2ds_B.fits
FITS files [generic_name]_wave_A.fits and [generic_name]_wave_B.fits

Quality control

- Check if Littrow solution has rms granulation less than ~50 m/s. if found greater calibDB is not updated.
- Check chi2 value of the wavelength solution in each order.

3.2.6 obj_TH_harpn.py / obj_WAVE_harpn.py - Accurate RV measurement

Inputs

Raw FITS frame obtained with the HARPN_ech_acq_thosimult template (with ThAr on fiber B) or HARPN_ech_acq_wavesimult template (FP on fiber B).

Description

Retrieve from calibDB previous last full calibration sets - Correction of the bad columns of the CCDs (average of adjacent columns)
- Correction of the BIAS as described previously
- Read the last localization, flat field, wavelength solution and Thorium reference spectrum in the Calibration Data Base
- Horne optimum extraction of orders
- Flat field correction
- Save E2DS simultaneous reference spectrum in FITS file with the suffix e2ds with all descriptors.
- Compute the rebinned and merged spectrum S1D and save it with the suffix s1d.
- Compute the instrumental drift with the sim. reference spectrum relative to the last reference in the Calibration Data Base and save the result in the file drift_result.tbl.
- Compute the Barycentric Earth Radial Velocity.
- Compute the cross correlation function with a template mask driven by the spectral type and save the average CCF on the FITS file [generic_name]_ccf_mask_A.fits, the RV for each orders in the table [generic_name]_ccf_mask_A.tbl, and the summary of results on the table CCF_result.tbl.

Outputs

ASCII file drift_result.tbl
ASCII file CCF_result.tbl
FITS files [generic_name]_e2ds_A.fits
FITS files [generic_name]_e2ds_B.fits
FITS files [generic_name]_s1d_A.fits
FITS files [generic_name]_s1d_B.fits
FITS files [generic_name]_ccf_mask_A.fits
FITS files [generic_name]_ccf_mask_A.tbl

Quality control

- Warning if saturation level reached

3.2.7 obj_ONE_harpn.py - Spectroscopy using one fiber

Inputs
Raw FITS frame obtained with the HARPN_ech_acq_objA template
Description
Retrieve from calibDB previous last full calibration sets - Correction of the bad columns of the CCDs (average of adjacent columns)
- Correction of the BIAS as described previously
- Read the last localization, flat field, wavelength solution and Thorium reference spectrum in the Calibration Data Base
- Horne optimum extraction of orders
- Flat field correction
- Compute the rebinned and merged spectrum S1D and save it with the suffix `s1d`.
- Compute the Barycentric Earth Radial Velocity.
- Compute the cross correlation function with a template mask driven by the spectral type and save the average CCF in the FITS file [generic_name]`_ccf_mask_A.fits`, the RV for each orders in the table [generic_name]`_ccf_mask_A.tbl`, and the summary of results on the table `CCF_result.tbl`.
Outputs
ASCII file `CCF_result.tbl`
FITS files [generic_name]`_e2ds_A.fits`
FITS files [generic_name]`_s1d_A.fits`
FITS files [generic_name]`_ccf_mask_A.fits`
FITS files [generic_name]`_ccf_mask_A.tbl`
Quality control
- Warning if saturation level reached

3.2.8 obj_TWO_harpn.py - Spectroscopy using two fibers

Inputs
Raw FITS frame obtained with the HARPN_ech_acq_objAB template
Description
Retrieve from calibDB previous last full calibration sets - Correction of the bad columns of the CCDs (average of adjacent columns)
- Correction of the BIAS as described previously
- Read the last localization, flat field, wavelength solution and Thorium reference spectrum in the Calibration Data Base
- Horne optimum extraction of orders for both fibers
- Flat field correction
- Compute the rebinned and merged spectrum S1D and save it with the suffix `s1d`.
- Compute the Barycentric Earth Radial Velocity.
- Compute the cross correlation function with a template mask driven by the spectral type for both fibers and save the average CCF on the FITS file [generic_name]`_ccf_mask_A.fits`, the RV for each order in the table [generic_name]`_ccf_mask_A.tbl`, and the summary of results on the table `CCF_result.tbl`.
Outputs

ASCII file `CCF_result.tbl`
FITS files [generic_name]`_e2ds_A.fits`
FITS files [generic_name]`_e2ds_B.fits`
FITS files [generic_name]`_s1d_A.fits`
FITS files [generic_name]`_s1d_B.fits`
FITS files [generic_name]`_ccf_mask_A.fits`
FITS files [generic_name]`_ccf_mask_A.tbl`
FITS files [generic_name]`_ccf_mask_B.fits`
FITS files [generic_name]`_ccf_mask_B.tbl`
Quality control
- Warning if saturation level reached

Chapter 4: DRS data product description

4.1 Data naming rules

The raw frames are stored on FITS format by the DFS with the ESO-VLT standard naming rules: `HARPN.YYYY-MM-DDTHH-MM-SS.SSS.fits` with YYYY-MM-DD and HH-MM-SS.SSS being respectively the date and time of the beginning of observation.
Image products of the DRS are stored on FITS format with the same generic names plus an additional suffix describing its format (see next section for details) and the specific fiber name (A or B). For example: `HARPN.YYYY-MM-DDTHH-MM-SS.SSS_e2ds_A.fits` is an E2DS format image of the fiber A product by the DRS from the `HARPN.YYYY-MM-DDTHH-MM-SS.SSS.fits` raw frame.
Tables in ASCII format are also produced by the DRS. The list and the description of the content of tables can be found in the section 4.3.
The relevant logbook of the DRS is named `DRS-drs32.YYYY-MM-DD`.

4.2 Data formats

4.2.1 Raw frames

The raw frame corresponds to a 4296 x 4112 integer matrix (35 MB) written on disk in FITS format (see Fig. 4.1). This image includes a 4096x4112 sensitive zone plus 4 over and prescan zone of 50 pixels each. The following generic descriptors are used by the DRS:

DATE-OBS	Date and Time of beginning of observation [string]
RA	RA of the target [float]
DEC	DEC of the target [float]

The DRS needs as well the following **HIERARCH TNG** descriptors:

DET READ SPEED	CCD Readout mode (speed, port and gain) [string]
DET OUT2 RON	Readout noise (e-) of blue readout port [float]
DET OUT2 CONAD	Conversion from ADUs to electrons of blue readout port [float]
DET OUT4 RON	Readout noise (e-) of red readout port [float]
DET OUT4 CONAD	Conversion from ADUs to electrons of red readout port [float]
DET WIN1 DIT1	Actual sub-integration time (s) [float]
DET WIN1 DKTM	Dark current time (s) [float]
DET DPR CATG	Observation category [string]
DET DPR TYPE	Exposure type [string]
INS DET1 TMMEAN	Normalized mean exposure time [float]
INS DET1 CMMEAN	Average counts PM on fiber A [float]
INS DET2 CMMEAN	Average counts PM on fiber B [float]
OBS TARG NAME	Target name [string]

TEL TARG EQUINOX	Equinox [float]
TEL TARG PMA	Proper motion alpha (arcsec/year) [float]
TEL TARG PMD	Proper motion delta (arcsec/year) [float]
TEL TARG RADVEL	Radial velocity of target (km/s) [float]
TEL AMBI FWHM START	seeing at start [float]
TEL AMBI FWHM END	seeing at end [float]
TEL AIRM START	air mass at start [float]
TEL AIRM END	air mass at end [float]
TPL NEXP	TPL Number of exposures [integer]
TPL EXPNO	TPL Exposure number within template [integer]
TPL NAME	TPL NAME [string]

4.2.2 Localization frames _loco_ and _fwhm-order_

Centers and FWHM of orders are stored in two 4096x(n_{order}) real matrix written on disk in FITS format with the suffix `_loco_` and _fwhm-order_ added to the generic name. Each line corresponds to an order and each column to a line of the CCD raw frame. The localization frame `_loco_` contains the position of the center positions of orders. The localization frame `_fwhm-order_` contains the FWHM of orders. Pay attention to the orientation of the raw frame by comparison with the E2DS format described below. The FITS descriptor includes all descriptors of the raw frame. In addition, the order position is stored with the following **HIERARCH TNG DRS** FITS descriptor:

CAL LOC NBO	number of orders localized [int]
CAL LOC DEG	degree of the polynomial fit [int]
CAL LOC CTRi	coefficient for center order (i=NBO x DEG) [float]
CAL LOC FWHMi	coefficient for FWHM order (i=NBO x DEG) [float]

4.2.3 Flat field frames _flat_

The Flat field frame is a 4096 x (n_{order}) real matrix written on disk in FITS format with the `_flat_` suffix added to the generic name. Each line contains the normalized flat field spectrum of the orders. The row numbering of the matrix corresponds to the inverse of the column of the raw frame in order to have the wavelength increasing with pixels (see on Fig.4.1). The FITS descriptor includes all descriptors of the raw frame.

4.2.4 Blaze frames _blaze_

The Blaze frame is a 4096 x (n_{order}) real matrix written on disk in FITS format with the `_blaze_` suffix added to the generic name. Each line contains the pseudo blaze response of the orders. The row numbering of the matrix corresponds to the inverse of the column of the raw frame in order to have wavelength increasing with pixels (see on Fig. 4.1). The FITS descriptor includes all descriptors of the raw frame.

4.2.5 Wavelength frames _wave_

The wavelength frame is a 4096 x (n_{order}) real matrix written on disk in FITS format with the `_wave_` suffix added to the generic name. Each line contains the wavelength calibration of the orders. The row numbering of the matrix corresponds to the inverse of the column of the raw frame in order to have wavelength increasing with pixels. The FITS descriptor includes all descriptors of the raw frame.

4.2.6 Extracted two-dimensional spectra _e2ds_

The extracted two-dimensional spectrum (E2DS) is a 4096 x (n_{order}) real matrix written on disk in FITS format with the `_e2ds_` suffix added to the generic name. Each line contains the extracted flux of one spectral order in photo-electrons unit. The line numbering of the matrix corresponds to the inverse of the column numbering of the raw frame in order to have wavelength increasing with pixels on E2DS format. The FITS descriptor includes all descriptors of the raw frame. The following extra descriptor **HIERARCH TNG DRS** related to the localization, flat-field, Barycentric Earth Radial Velocity, Instrumental drift and wavelength calibration are included:

CAL LOC FILE	Localization file used [string]
CAL EXT OPT	Option extraction [integer]
CAL EXT SIG	Size extraction zone [float]
CAL EXT COSM	Threshold cosmic detection [float]
CAL EXT SNi	S_N order center i [float]
CAL EXT NBCOSi	NbCos detected order i [integer]
CAL FLAT WIN	Half size blaze window [integer]
CAL FLAT FILE	Flat file used [string]
CAL FLAT NBFILES	Nb of Flat files used [integer]
CAL FLAT RMSi	FF RMS order i [float]
CAL ADDED NBRFILES	Nb of raw files processed [integer]
CAL ADDED FILES	Files name of raw files used [string]
BERV	Barycentric Earth Radial Velocity [float]
BJD	Barycentric Julian Day [float]
BERVMX	Maximum BERV [float]
CAL TH FILE	Wavelength file used [string]
DRIFT REF FILE	ThAr Drift ref file used [string]
DRIFT VR	ThAr RV Drift (m/s) [float]
DRIFT NBCOS	ThAr Drift nb cosmic detected [integer]
DRIFT RFLUX	ThAr Drift Flux ratio [float]
DRIFT NBORDKILL	ThAr Drift nb orders killed [integer]
DRIFT NOISE	ThAr Drift photon noise (m/s) [float]
CAL TH ORDER NBR	Nb of orders in total [int]
CAL TH ORDER START	Numbering of the first blue order [int]
CAL TH ORDER NBLUE	Nb of blue orders [int]
CAL TH ORDER NGAP	Nb of orders in the gap [int]
CAL TH ORDER NRED	Nb of red orders [int]
CAL TH GUESS ORDER	Nb of the first guess order [int]
CAL TH GUESS LINES	File name for first guess lines [string]
CAL TH LINES	File name for tbl of cal lines [string]
CAL TH DEG LL	Degree polynomial fit ll(x, order) [int]
CAL TH DEG X	Degree polynomial fit x(ll, order) [int]
CAL TH COEFF LL	Coeff for ll(x, order) [dbl precision]
CAL TH COEFF X	Coeff for x=(ll, order) [dbl precision]

The wavelength calibration lambda(x) is related to the coefficient with the following equation:

lambda(x) = sum(i=0...d) a_i * x**i

where **d** = TNG DRS CAL TH DEG LL, a_i = TNG DRS CAL TH COEFF LLi and the internal numbering of the order (raw number in the e2ds frame).
For thorium spectrum an extension _wave_ is also produced by the DRS. It is an e2ds format image where the matrix stores the wavelength value of each pixel instead of the flux of the spectrum.

4.2.7 Extracted one-dimensional spectra _s1d_

The extracted one-dimensional spectrum (S1D) is a real vector written on disk in FITS format with the _s1d_ suffix added to the generic name. This vector contains the rebinned and merged spectral orders in relative flux corrected from the instrumental respond and stretched to the barycentric referential. The wavelength step is 0.01 Angstrom.

4.2.8 Cross-correlation function _ccf_

The cross correlation function is stored with the suffix _ccf_[template_name] with template_name the file name of the corresponding template (also called correlation mask) used to compute it. The matrix is made of n+1 CCFs corresponding to the CCFs computed for each order, plus the summed CCF over all orders. The following extra descriptor **HIERARCH TNG DRS** related to CCF is included:

CCF MASK	Template filename
CCF MAXCPP	Max count/pixel in the continuum of the CCF (e-)
CCF FWHM	FWHM of CCF (km/s) [gaussian fit]
CCF RV	Baryc Rad vel (km/s) [gaussian fit]
CCF LINES	Nb of lines used by the template
CCF CONTRAST	Contrast of CCF (%) [gaussian fit]
BERV	Barycentric Earth Radial Velocity correction
BJD	Barycentric Julian Day
BERVMX	Maximum BERV along the year
DRIFT REF FILE	ThAr Drift ref file used [string]
DRIFT VR	ThAr RV Drift (m/s) [float]
DRIFT NBCOS	ThAr Drift nb cosmic detected [integer]
DRIFT RFLUX	ThAr Drift Flux ratio [float]
DRIFT NBORDKILL	ThAr Drift nb orders killed [integer]
DRIFT NOISE	ThAr Drift photon noise (m/s) [float]

4.3 Summary tables

The DRS produces a set of summary table for performance tracking of calibration exposure. The tables have ASCII format with a TAB for separator:

cal_BIAS_result.tbl	for BIAS
cal_DARK_result.tbl	for DARK
cal_loc_ONE_result.tbl	for the order localization

cal_FF_result.tbl	for Flat-Field
cal_TH_result.tbl	for thorium calibration
drift_result.tbl	for instrumental drift
CCF_result.tbl	for CCF

4.3.1 cal_BIAS_result.tbl

Column description:

night_name	name of the night directory
file_name	name of the corresponding raw frame
ccd_mode	CCD read out mode
mean_bias1	mean bias in zone 1 (prescan Linda) (ADU)
rms_bias1	rms bias in zone 1 (prescan Linda) (ADU)
mean_bias2	mean bias in zone 2 (overscan Linda) (ADU)
rms_bias2	rms bias in zone 2 (overscan Linda) (ADU)
mean_bias3	mean bias in zone 3 (prescan Jasmin) (ADU)
rms_bias3	rms bias in zone 3 (prescan Jasmin) (ADU)
mean_bias4	mean bias in zone 4 (overscan Jasmin) (ADU)
rms_bias4	rms bias in zone 4 (overscan Jasmin) (ADU)

4.3.2 cal_DARK_result.tbl

Column description:

night_name	name of the night directory
file_name	name of the corresponding raw frame
ccd_mode	CCD read out mode
dark_time	dark exposure time (s)
mean_dark	mean dark level (e-/hour)
cosmic	number of cosmic events (event/cm2/mn)

4.3.3 cal_loc_ONE_result.tbl

Column description:

night_name	name of the night directory
file_name	name of the corresponding raw frame
fiber	fiber name (A or B)
posx_161	location of the center of the order 161 [pixel]
err_posx_161	error on order location [pixel]
fwhm_161	width of the center of the order 161 [pixel]
err_fwhm_161	error on the width of the center of the order 161 [pixel]
posx_114	same than above for order 114
err_posx_114	same than above for order 114
fwhm_114	same than above for order 114
err_fwhm_114	same than above for order 114
posx_89	same than above for order 89
err_posx_89	same than above for order 89
fwhm_89	same than above for order 89
err_fwhm_89	same than above for order 89

4.3.4 cal_FF_result.tbl

Column description:

night_name	name of the night directory
file_name	name of the corresponding raw frame
nbfiles	number of frame coadded
fiber	fiber name (A or B)
FFrms_161	rms on the flat-field at center of order 116
S_N_161	S/N ratio per extracted pixel at center of order 161
FFrms_114	same than above for order 114
S_N_114	same than above for order 114
FFrms_89	same than above for order 89
S_N_89	same than above for order 89

4.3.5 cal_TH_result.tbl

Column description:

night_name	name of the night directory
file_name	name of the corresponding raw frame
fiber	fiber name (A or B)
mean	mean value of the final solution in mpixel
rms	rms on the final solution in mpixel
N_lines	number of lines used in the final solution
err	internal error in the final solution in mpixel
rms_L0	rms on Littrow at cut #1 in mpixel
rms_L1	rms on Littrow at cut #2 in mpixel
rms_L2	rms on Littrow at cut #3 in mpixel
drift	drift in m/s compared to previous wavelength solution
Rflux	flux ration by comparison with the previous solution
Ccosmic	number of corrected cosmic
ll1ref	wavelength of reference line #1
ampl1ref	amplitude in (e-) of the reference line #1
ll2ref	wavelength of reference line #2
ampl2ref	amplitude in (e-) of the reference line #2
error_spe	estimate of the velocity photon noise error of the spectrum

4.3.6 drift_result.tbl

Column description:

night	name of the night directory
e2ds_file_name	name of the corresponding thorium e2ds spectrum
reference	name of the thorium e2ds spectrum used as reference
exp_time	exposure time (s)
VR_drift	instrumental drift (m/s) (to add to the CCF result)
Nbcosmic	nb cosmic corrected
Flu_ratio	flux ratio between thorium spectrum and reference
Nborders_killed	nb order killed by the process

4.3.7 CCF_result.tbl

Column description:

night_name	name of the night directory
file_name	name of the corresponding thorium e2ds spectrum
fiber	fiber 'A' or 'B'
mask	name of the CCF template
maxccp	maximum of count per pixel in the CCF
lines	number of spectral lines used in the CCF
contrast	contrast of the CCF (%)
RV	RV (km/s) corrected from BERV
FWHM	FWHM of the CCF (km/s)

4.4 Databases

Several databases are accessed by the DRS. The DRS uses **Calibrations Databases** to store and to retrieve calibrations. It needs a **Reduction Performance List** for the quality control. The DRS main engine uses the **Instrument Configuration Data Pool** as input reference to carry on the reduction.

4.4.1 Calibration Database

The Calibration Databases includes all relevant calibrations, which have passed properly the quality control tests. It contains BIAS frames, DARK frames, localization images, flat-fields, and thorium calibrated spectra. A **master_calib.txt** file keeps track of all the calibration frames. All calibration frames needed and accessed by the DRS during the reduction process are automatically copied on the directory where all data product of the DRS are stored. This allow the observer to have a self-consistent set of data products if he wants to reprocess his observations Calibration Database stores a full **calibration set** made of:

- order localization (A and B)
- flat-field and blaze spectrum (A and B)
- wavelength solution (A and B)
- reference thorium spectrum (B)

4.4.2 Reduction Performance list

The reduction performance list contains all the DRS parameters that are checked by the Quality Control process. This list is stored on the `/config/` directory.

4.4.3 Instrument Configuration Data Pool

All the fixed parameters needed by the DRS are in the file `hadmrICDP_HARPN.py` on the directory `/config/`. In this file, one finds the characteristics of the CCD, mapping of the FITS descriptor to DRS variable, and all the parameters of DRS algorithms optimized for HARPS.

4.4.4 Instrument Performance Database

All DRS outcome helpful to track instrument performance is stored in the following table files:

- cal_BIAS_result.tbl,
- cal_DARK_result.tbl,
- cal_loc_one_result.tbl,
- cal_FF_result.tbl,
- cal_TH_result.tbl,
- [generic name]_spot_thAB.tbl,
- [generic name]_lines_A.rdb,
- [generic name]_lines_B.rdb.

Chapter 5: Off-line DRS description

5.1 Overview

The Data Reduction Off-line corresponds to a set of tools offer to the user through a Graphical User Interface to display and analyze the reduced data provided by the On-line DRS. The GUI can be started with the offdrs.csh command. The outputs of the Off-line DRS are generated in the local directory. Note that these tools are not essential to analyze reduced data since their format is completely compatible with Midas or other tools.

5.2 Recipe

5.2.1 off_visu_e2ds_harpn.py - Display E2DS order

```
off_visu_e2ds_harpn.py [night] [e2dsfits] [order_number] [ps]
```

[night]	night directory (2012-05-11)
[e2dsfits]	E2DS fits file (HARPN.2002-02-11T20-13-45.768_e2ds_A.fits)
[order_number]	order number (0-68) *default value = 49*
[ps]	postscript file output option (0/1) *default value = 0*

5.2.2 off_visu_s1d_harpn.py - Display S1D spectrum

```
off_visu_s1d_harpn.py [night] [s1dfits] [lambda_start] [lambda_end]
[ps]
```

[night]	night directory (2012-05-11)
[s1dfits]	S1D FITS file (HARPN.2002-02-11T20-13-45.768_s1d_A.fits)
[lambda_start]	first wavelength (Angstrom) *default value 3780*
[lambda_end]	last wavelength (Angstrom) *default value 6912*
[ps]	postscript file output option (0/1) it default value = 0

5.2.3 off_visu_ccf_harpn.py - Display CCF

```
off_visu_ccf_harpn.py [night] [ccffits] [ps]
```

[night]	night directory (2012-05-11)
[ccffits]	CCF FITS file (HARPN.2002-02-11T20-13-45.768_ccf_G2_A.fits)
[ps]	postscript file output option (0/1) it default value = 0

5.2.4 off_visu_SN_harpn.py - Display S_N per orders

`off_visu_SN_harpn.py [night] [e2dsfits] [ps]`

[night] night directory (2012-05-11)
[e2dsfits] E2DS FITS file (HARPN.2002-02-11T20-13-45.768_e2ds_A.fits)
[ps] postscript file output option (0/1) it default value = 0

5.2.5 off_visu_rvo_harpn.py - Display RV per orders

`off_visu_rvo_harpn.py [night] [ccftbl] [ps]`

[night] night directory (2012-05-11)
[ccftbl] table CCF file (HARPN.2002-02-11T20-13-45.768_ccf_G2_A.tbl)
[ps] postscript file output option (0/1) it default value = 0

5.2.6 off_make_ccf_harpn.py - Re-process CCF

`off_make_ccf_harpn.py [night] [e2dsfits] [mask] [targetRV] [widthccf] [stepccf]`

[night] night directory (2012-05-11)
[e2ds] E2DS FITS file (HARPN.2002-02-11T20-13-45.768_e2ds_A.fits)
[mask] cross-correlation template (G2/K5/M2) *default is G2*
[targetRV] target Radial Velocity (km/s) *use -99999 for an automatic search*
[widthccf] half-window of the CFF (km/s) *default is 20 km/s*
[stepccf] step of the CCF (km/s) *default is 0.25 km/s*

The new CCF is saved in the file [generic name]`_ccf_mask_A.fits`