Southern African Large Telescope

Title: SALT Long-slit Data Reduction

Author(s): Alexei Kniazev

Doc. number: 2350AC0100

Version: 1.5

Date: *May 19, 2008*

Keywords: Long-slit, Data reduction, Pipeline recipe

Approved: David Buckley (Ast Ops Manager)

Signature:________________ Date:____________

ABSTRACT

In this document I am discussing long-slit data reduction in terms of SALT RSS spectral data. Main parameters of tasks of IRAF `LONGSLIT`*,* `APEXTRACT` *and* `CCDRED` *packages are summarized that are crucial for reduction of this type of SALT data. I am also present steps of long-slit reduction in terms of recipe for SALT pipeline of long-slit reduction.*

South African Astronomical Observatory, Cape Town 7925, South Africa

Contents

1 Glossary **3**

2 Introduction **4**

3 Long-slit Reduction Recipe **4**

3.1 Preliminary steps 5

3.2 Reference Spectra reduction 6

3.3 Standard star reduction 7

3.4 Science data reduction 8

4 RSS long-slit Data Reduction System in terms of Pipeline Recipe **11**

4.1 General Pipeline 11

4.2 Basic ideas for RSS long-slit recipe 12

4.3 Data types 12

4.4 Names convention 12

4.5 Pipeline Reduction Recipe 13

5 Data Quality Control **17**

1 Glossary

Data Flow System (DFS): a system that can handle the flow of data and informaion within SALT. It includes subsystems for proposals handling, science data archiving, science data pipeline processing and handling and data quality control.

Data Organizer (DO): software used in the on-line envoronment to support the automatic pipeline data reduction of incoming data.

Data Products (DP): data files delivered to PIs as result of the execution SALT observing programs. DP include observation frames, observatory calibrations, meteorological and seeing measurements relevant to primary observations, etc.

Header Keywords: header keywords can be standard FITS or SAAO extension.

Raw Frame (RF): Result of execution of an exposure by the ?OCS?. The execution of an Observation Block may generate one or more Raw Frames. Raw Frames are stored as FITS files. Their headers contain all the information relevant for pipeline data reduction, quality control and archiving.

Reduction Block (RB): A Reduction Block contains all the information needed to reduce one or more RFs: it contains the name of the pipeline recipe to be applied, the list of input frames, the list of associated master calibration files, and the name of the output pipeline products to be generated.

Reduction Block Scheduler (RBS): software, part of DFS, which operates on Reduction Blocks, obtained from the DO: it analyzes the RB, checks the presence of input files, and uses the corresponding data reduction recipe.

Reduction Recipe: standard procedure for reducing observational data in a standard way. Recipies are implemented for each of the instrument standard templates. Those scripts take as input RFs and execute them in a particular Data Reduction System (DRS).

2 Introduction

The Prime Focus Imaging Spectrograph that was renamed to the Robert Stobie Spectrograph (RSS) is a multi-mode instrument and can be used for the narrow-band imaging, long-slit spectroscopy, multi-object spectroscopy, Fabri-Perot imaging spectroscopy, imaging polarimetric and spectropolarimetric observations. From the general point of view the long-slit mode of observations has to be moslty used with RSS.

The goal of this document to show how to reduce RSS long-slit data after these data were primary reduced and such reduction steps were done previously: overscan , bias correction, cosmic rays, gain, crosstalk and merging of the amplifiers.

3 Long-slit Reduction Recipe

RSS long-slit data are very standard long-slit data and they can be reduced using common used tasks from IRAF [1] packages `LONGSLIT` and `APEXTRACT`.

As my experience, the first important step has to be done is to destroy Multiple Extensions FITS (MEF) header we have after current version of RSS primary reduction. After that, flatfielding and possibly vignetting correction have to be done as well as trimming of spectral data.

Long-slit reduction itself can be started after that. From my point of view it is easier to devide sequence of steps for the whole long-slit reduction procedure into amount of steps for three main types of the long-slit data:

1. Reference spectrum – it is used to build 2D wavelength transformation;
2. Spectrophotometric standard star – it is used to build sensitivity curves;
3. Science data itself.

I will descibe below each of these steps in more details implementing them for RSS data.

[1] IRAF: the Image Reduction and Analysis Facility is distributed by the National Optical Astronomy Observatory, which is operated by the Association of Universities for Research in Astronomy, In. (AURA) under cooperative agreement with the National Science Foundation (NSF).

3.1 Preliminary steps

1. In my current understanding Multiple Extensions FITS (MEF) headers we have now after SALT primary data reduction is only way to make simple things in a very complex way. Personally, I do not see currently ANY reasons why we are doing it except to make me crazy in time I am trying to reduce any SALT data. For this reason, first I am doing with spectral SALT data – I am destroing FITS MEF headers. Potentially, I hope, the way exist to use IRAF `fitsutil` package for that. Unfortunately, I did not find how to do with any tasks from this package and use for that MIDAS, where any extension can be extracted with coping of zero-level header to that.

2. Spectral data have to be devided to illumination corrected flat field. For that I usually either used task `mkillumflat` from `CCDRED` package. The illumination pattern is determined by heavily smoothing the image using a moving "boxcar" average. The output image is the ratio of the input image to the illumination pattern. The illumination pattern is normalized by its mean to preserve the mean level of the input image:

   ```
   mkillumflat input=Flat.fits output=Flat_n.fits
   xboxmin=3 yboxmin=3 xboxmax=5 yboxmax=5

   ccdproc images=@SpecList.lst flatcor+ zerocor- flat=Flat_n.fits
   ```

   Sometimes I do not like the result that `mkillumflat` creates. In this case I use fast median smoothing algorithm with circular window size, that was implemented by ourself (Special Astrophysical Observatory) into MIDAS.

3. In case of very extended objects correction for vignetting also have to be done. Usually for that long-slit spectra of twilight-sky are used. Task `illumination` from `LONGSLIT` package I usually use for that. The illumination calibration is determined by fitting functions across the slit (the slit profiles) at a number of points along the dispersion, normalizing each fitted function to unity at the center of the slit, and interpolating the illumination between the dispersion points. The fitted data is formed by dividing the dispersion points into a set of bins and averaging the slit profiles within each bin. The interpolation type is a user parameter. For example:

   ```
   flatcombine input=@Sky_Spec.lst output=Sky_Spec.fits scale=mode combine=median

   illumination images=Sky_Spec.fits illumina=Sky_Spec_n.fits nbins=40
   function=chebyshev order=15

   ccdproc images=@SpecList.lst illumcor+ zerocor- illum=Sky_Spec_n.fits
   ```

4. Trimming has to be done to exclude non-useful areas: either on the base of information for certain spectral setups or for each object.

3.2 Reference Spectra reduction

Reduction of reference spectra consist of three main steps that are shown on the flow shart in Figure 1.

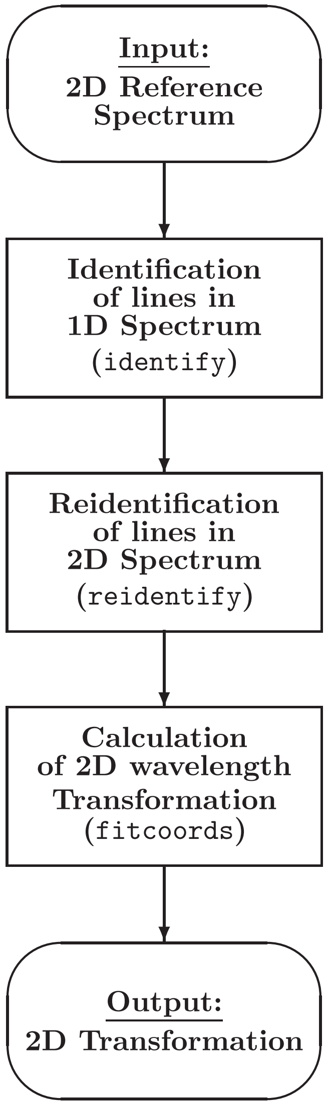
Input:
2D Reference Spectrum
Identification of lines in 1D Spectrum (identify)
Reidentification of lines in 2D Spectrum (reidentify)
Calculation of 2D wavelength Transformation (fitcoords)
Output:
2D Transformation

Figure 1: Flow chart of the reference spectrum reduction. All input and output data are shown with ovals.

1. First step, that is identification of lines of reference spectrum in one particular row of 2D spectrum could be done usind task `identify`. For example, for reduction of RSS grating GR900 I usually used:

   ```
   identify images=refspec.fits coordlist=cuar.dat function=chebyshev
   order=5 fwidth=6. cradius=6.
   ```

where cuar.dat is file with identificated lines of CuAr lamp of RSS.

2. Next step is reidentification of identifited lines in all others rows of 2D spectrum could be done usind task `reidentify`. Below is an example of parameters I usually used for reduction of RSS data:

   ```
   reidentify reference=refspec.fits images=refspec.fits interactive=no
   newaps=yes override=no refit=yes nlost=20 coordlist=cuar.dat verbose=yes
   ```

3. The last step is calculation of 2D transformation using task `fitcoords`. For example, for reduction of RSS data I usually used:

   ```
   fitcoord images=refspec interactive=yes combine=no
   functio=legendre xorder=5 yorder=3
   ```

3.3 Standard star reduction

Reduction of standard star consist of five steps that are shown on the flow shart in Figure 2:

1. First step is making of 2D wavelength transformation using task `transform`. This IRAF program uses transformation previously calculated with `fitcoords`. For example:

   ```
   transform input=stdstar.fits output=stdstarw.fits fitnames=refspec
   interptype=linear flux=yes blank=INDEF x1=INDEF x2=INDEF dx=INDEF
   y1=INDEF y2=INDEF dy=INDEF
   ```

   I would like to note here that parameter “interptype” is preferable to be of type “linear” here.

2. Background subtraction has to be done after that using task `background`. In case of standard star this task easily can be done in the non-interactive mode:

   ```
   background input=stdstarw.fits output=stdstarws.fits axis=2 interactive=no
   naverage=1 function=chebyshev order=5 low_rej=2 high_rej=1.5 niterate=5 grow=0.
   ```

3. One-dimensional spectrum has to be extracted as a next step with task `apall` from `APEXTRACT` package:

   ```
   apall input=stdstarws.fits format=onedspec interactive=no
   nfind=1 llimit=-40 ulimit=40 t_order=10 ylevel=INDEF
   ```

   In case of the standard star this task easily can be done in the non-interactive mode.

4. Standard star distribution has to be copied from the database using task `standard`:

```
standard input=stdstarws.fits output=sens.dat caldir=../caldir
interact=no star_name=feige110 extinct=../caldir/suzextinct2.dat
```

5. The last step is to create sensitivity curve using task `sensfunc`. Generally, this step can not be done in the non-interactive mode since includes averaging of all standard stars that were observed during the night:

   ```
   sensfunc standard=sens.dat sensitiv=sens.fits interactive=yes
   ```

3.4 Science data reduction

Reduction of any science data consist of three steps that are shown on the flow shart in Figure 3:

1. First step is making of 2D wavelength transformation using task `transform`. This IRAF program uses transformation previously calculated with `fitcoords`. For example:

   ```
   transform input=object.fits output=objectw.fits fitnames=refspec
   interptype=linear flux=yes blank=INDEF x1=INDEF x2=INDEF dx=INDEF
   y1=INDEF y2=INDEF dy=INDEF
   ```

   I would like to note here that parameter "interptype" has to be of type "linear", otherwise one will have problems with flux transformation for emission lines.

2. Background subtraction has to be done after that using task `background`. Depending on the type of studied object this task can be done in the interactive or in the non-interactive mode:

   ```
   background input=objectw.fits output=objectws.fits axis=2 interactive=yes
   naverage=1 function=chebyshev order=2 low_rej=2 high_rej=1.5 niterate=5 grow=0.
   ```

3. The last step is to correct 2D spectrum for sensitivity curve and extinction could be done using task `calibrate`:

   ```
   calibrate input=objectws.fits output=objectwsf.fits
   sensitivity=sens.fits extinction=../caldir/suzextinct2.dat
   ```

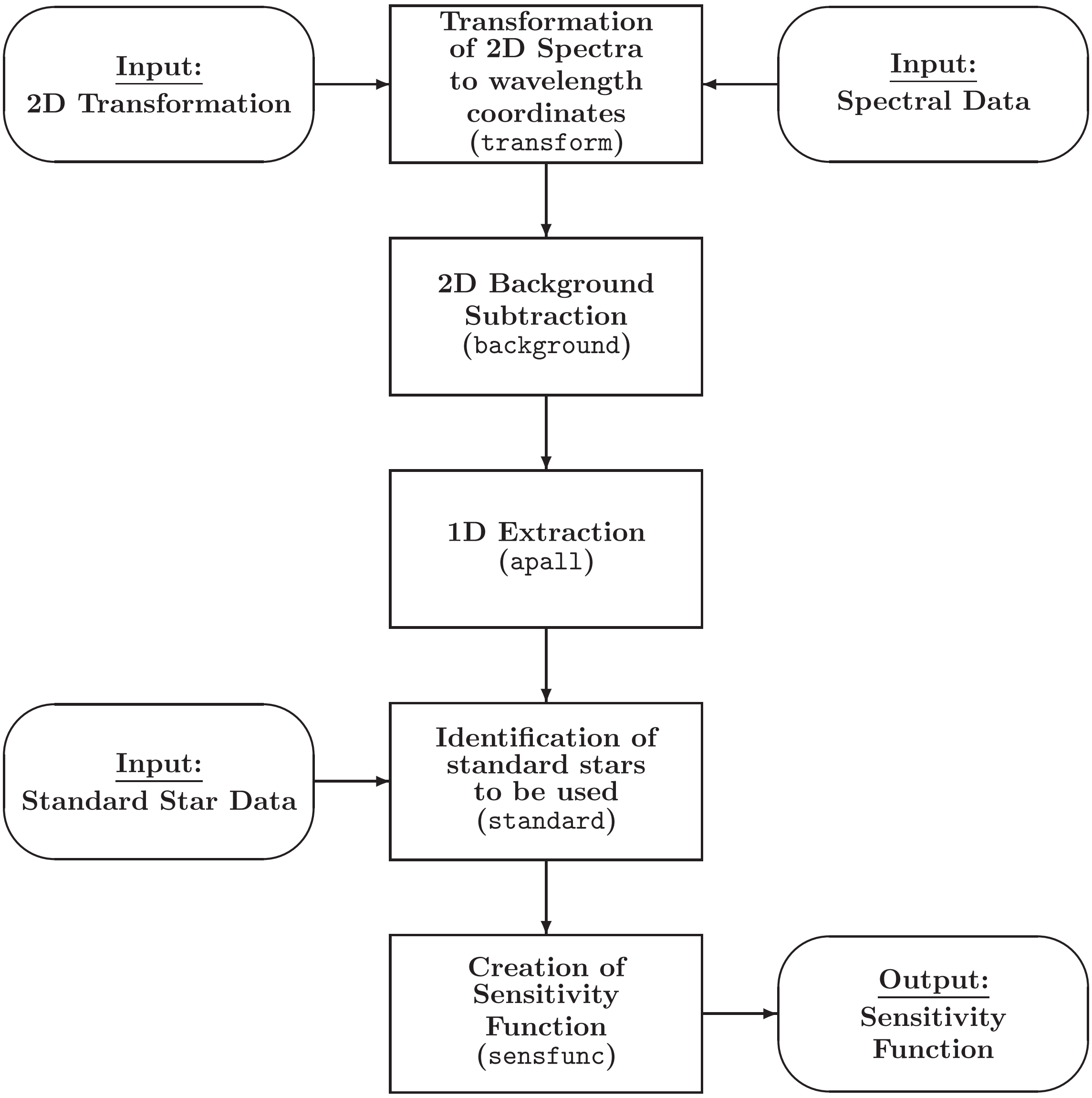
Input:
2D Transformation
Transformation of 2D Spectra to wavelength coordinates (transform)
Input:
Spectral Data
2D Background Subtraction (background)
1D Extraction (apall)
Input:
Standard Star Data
Identification of standard stars to be used (standard)
Creation of Sensitivity Function (sensfunc)
Output:
Sensitivity Function

Figure 2: Flow chart for the Standard star reduction

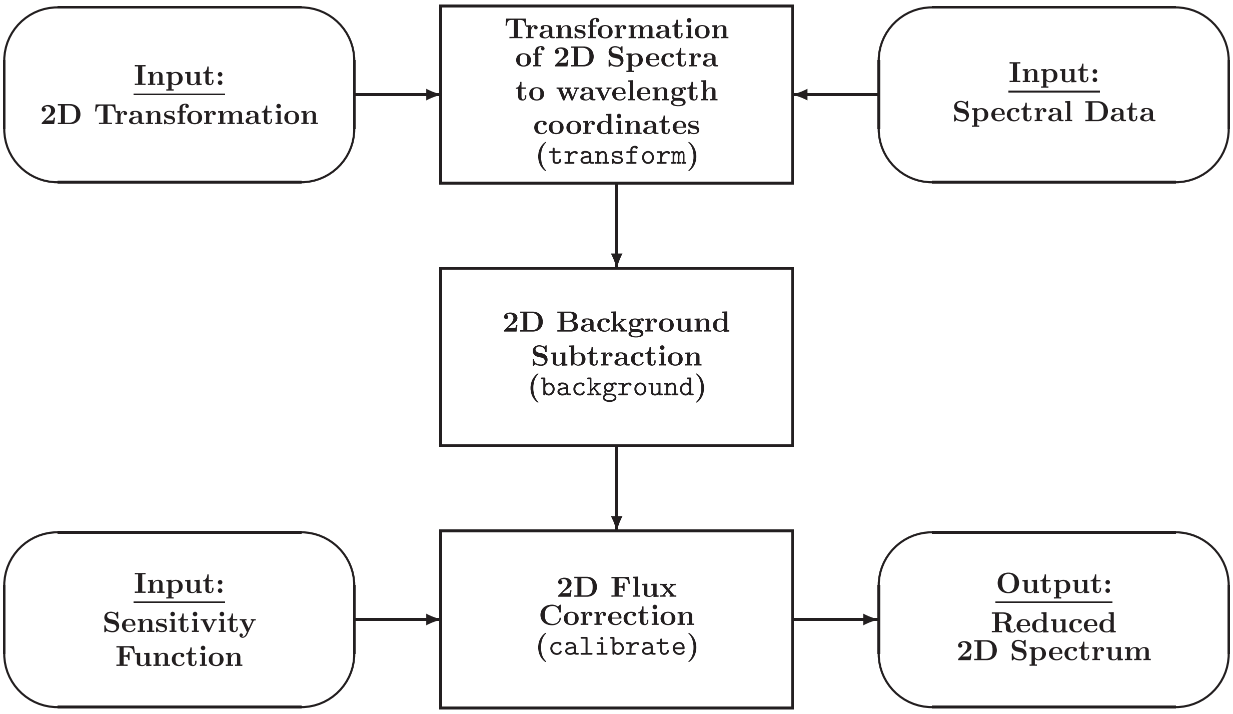

Figure 3: Flow chart for the object reduction

4 RSS long-slit Data Reduction System in terms of Pipeline Recipe

4.1 General Pipeline

Following to my understanding the general flow chart for the General Pipeline is shown in Figure 4.

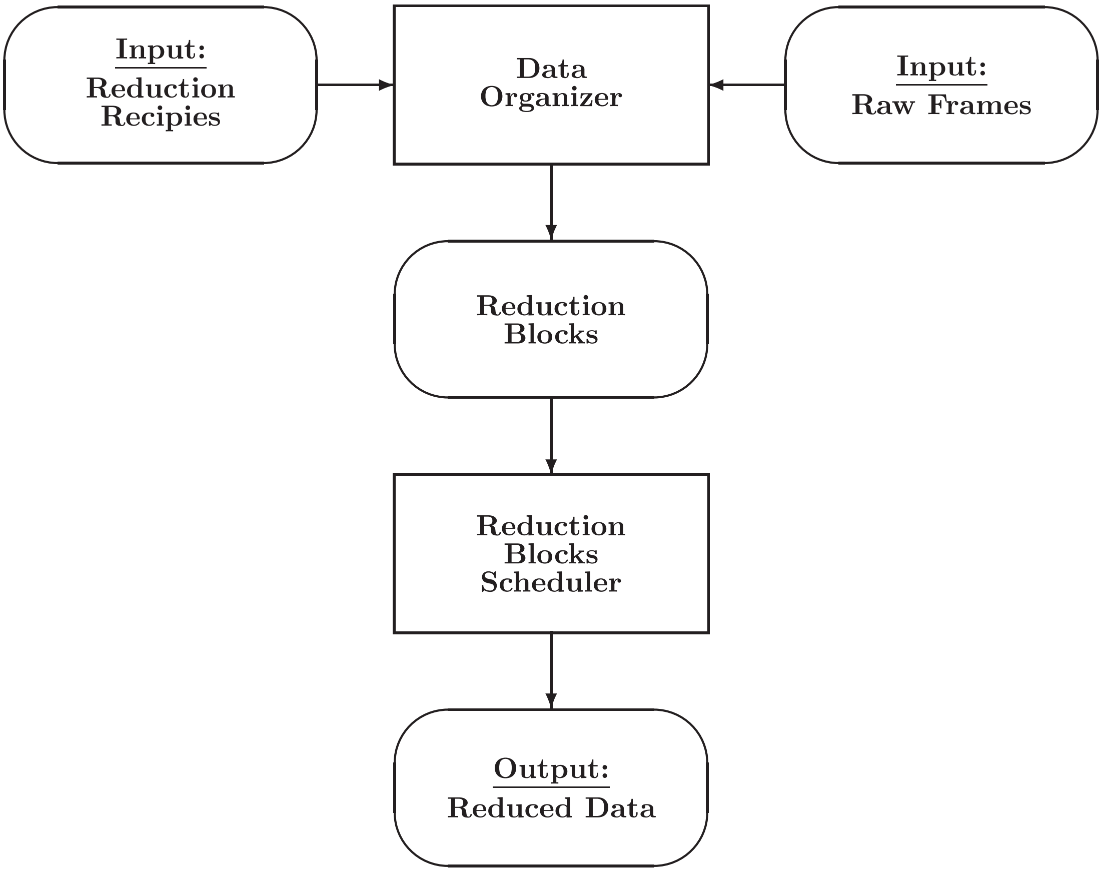

Figure 4: Flow chart for the General Pipeline

I do not want to discuss here any DO software that is extremely important but very large project anyway. As an examples of currently used DOs please see description of MIDAS Data Organizer context in the Volume B of ESO-MIDAS User's Guide or Gasgano's Users Manual (VLT-PRO-ESO-19000-1932).

Below I suggest that primary reduced data are organized into Reduction Blocks and type of pipeline is selected (simple long-slit).

4.2 Basic ideas for RSS long-slit recipe

RSS design assumed big, but finitive amount of spectral configurations. From my point of view, as a first step, it is relatively easy to create amount of programs for RSS long-slit data reduction that will be built only with use of basic IRAF tasks. It is useful, from my point of view, design the future pipeline in the way that it will automatically save all parameters for the reduction of different setups into the Spectral Setups Database (SSDB) to use it next time in the automatic mode. SSDB can be either real database or just a specific directory, where all files for different setups and dates are collected. So, the long-slit recipe, in my opinion, has to support, as minimum, next ideas:

1. To support the uniform system of file naming, where after each step of the reduction file name has to be changed in the standard way;
2. To save and to use different spectral setups as well for the different observational dates;
3. To save and to use sensitivity curves that were taken for the different spectral setups for the different dates;
4. To save and restore specific parameters for the different IRAF tasks to reduce data of the different spectral configurations.

4.3 Data types

One of the options just to use the same file types as IRAF uses in its packages `LONGSLIT` and `APEXTRACT` that means standard FITS and ASCII files. For this reason, as a result of our primary reduction of long-slit data we have to have zero-level FITS headers only and have to exclude multi-extension FITS format (MEF), from my point of view.

4.4 Names convention

It is possibly to imaging different names convention rules. These rules depend on many factors and ideas.

First, file names after each step of reduction. Historically, as an evolution from GEMINI package, SALT primary reduction adds prefix after each step of reduction. In this way it is easy to see bulk of images that passed certain step of reduction, but much harder to see the status of reduction of each observed file, from my point of view. For this reason, personally, I prefer not prefixes, but suffixes:

File+c.fits – 2D spectrum after cosmic filtration;

Filec+e.fits – 2D spectrum after working area extraction;

Filece+w.fits – 2D spectrum after wavelength transformation;

Filecew+s.fits – 2D spectrum after background subtraction;

Filecews+f.fits – 2D spectrum after flux calibration;

Example:	**mrpP200610260003cew.fits**	long-slit spectrum after wavelength calibration
	mrpP200610260003cewsf.fits	totally reduced long-slit spectrum

Below I show suggested names convention assuming that SSDB is just a specific directory and names of files itself have to contain all different parameters that make them unique:

1. For reference spectra:

   LAMPID_GRATING_ARANGLE_GRANGLE_BIN, where:

   LAMPID – reference spectrum identification (content of FITS descriptor **LAMPID**);

   GRATING – grating identification (content of FITS descriptor **GRATING**);

   ARANGLE – articulation angle identification (content of FITS descriptor **CAMANG** in the format of F5.2).
   WARNING: This is NOT the REAL value of articulation angle, but an ideal value that was requested.

   GRANGLE – grating angle identification (content of FITS descriptor **GRTILT** in the format of F5.2).
   WARNING: This is NOT the REAL value of grating angle, but an ideal value that was requested.

   BIN – binning identification (content of FITS descriptor **CCDSUM**).

Example:	**ThAr_PG0900_27.25_13.63_2x2.fits**	ref. spectrum for ThAr
	CuAr_PG0900_27.25_13.63_2x2.fits	the same for CuAr

2. For sensitivity curves:

   Std_GRATING_ARANGLE_GRANGLE_BIN_DATE, where:

   GRATING – grating identification (content of FITS descriptor **GRATING**);

   ARANGLE – articulation angle identification (content of FITS descriptor **CAMANG**);

   GRANGLE – grating angle identification (content of FITS descriptor **GRTILT**);

   BIN – binning identification (content of FITS descriptor **CCDSUM**);

   DATE – date of observation in the format **YYYYMMDD**, where:

   YYYY – year;

   MM – month;

   DD – day.

4.5 Pipeline Reduction Recipe

Flow charts are shown in Figures 1–3 has to be changed propelry following the pipeline ideas. Such new flow charts are shown in Figures 5–7.

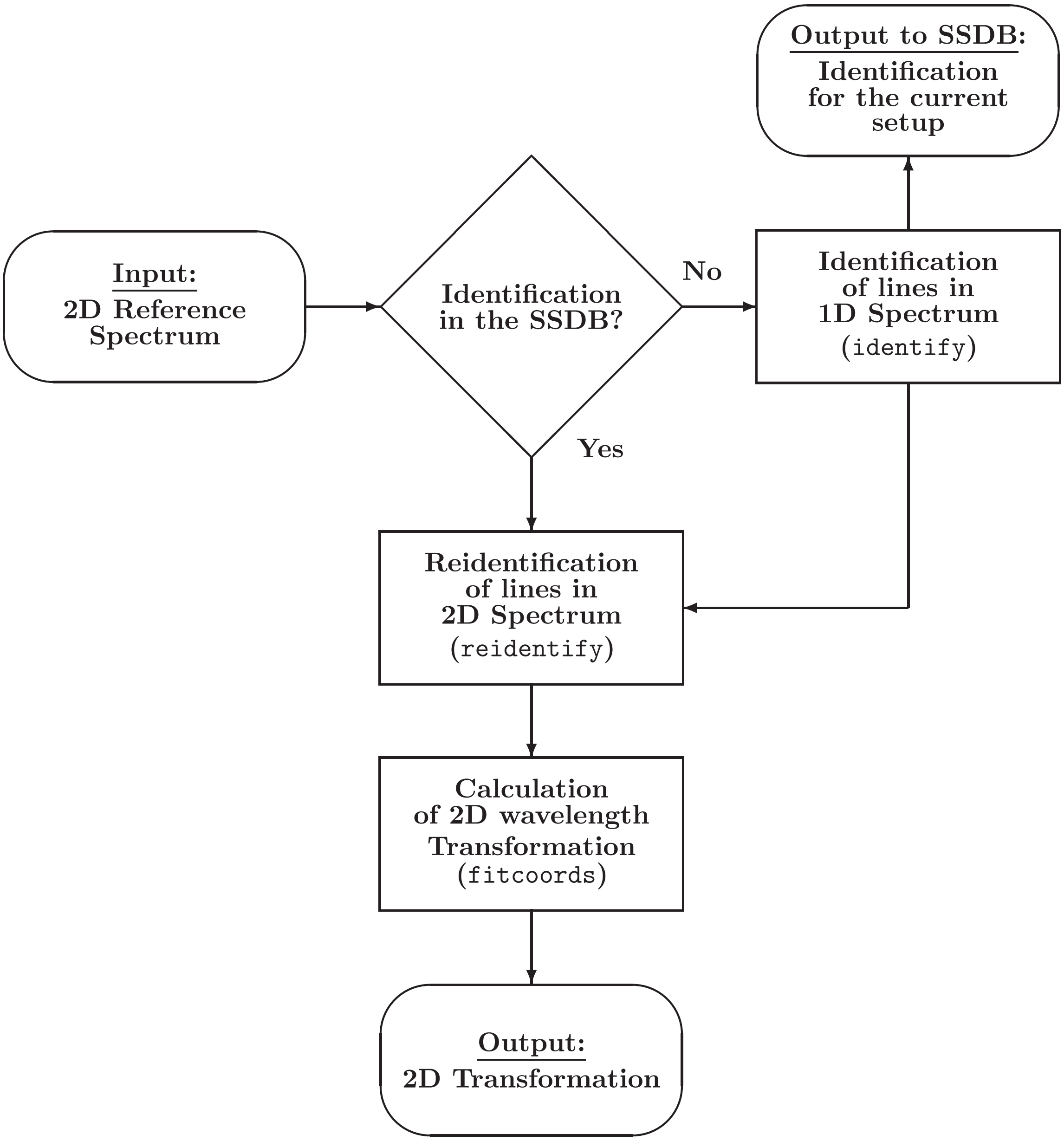
Output to SSDB:
Identification for the current setup
Input:
2D Reference Spectrum
Identification in the SSDB?
No
Yes
Identification of lines in 1D Spectrum (identify)
Reidentification of lines in 2D Spectrum (reidentify)
Calculation of 2D wavelength Transformation (fitcoords)
Output:
2D Transformation

Figure 5: Flow chart of the reference spectrum reduction using pipeline

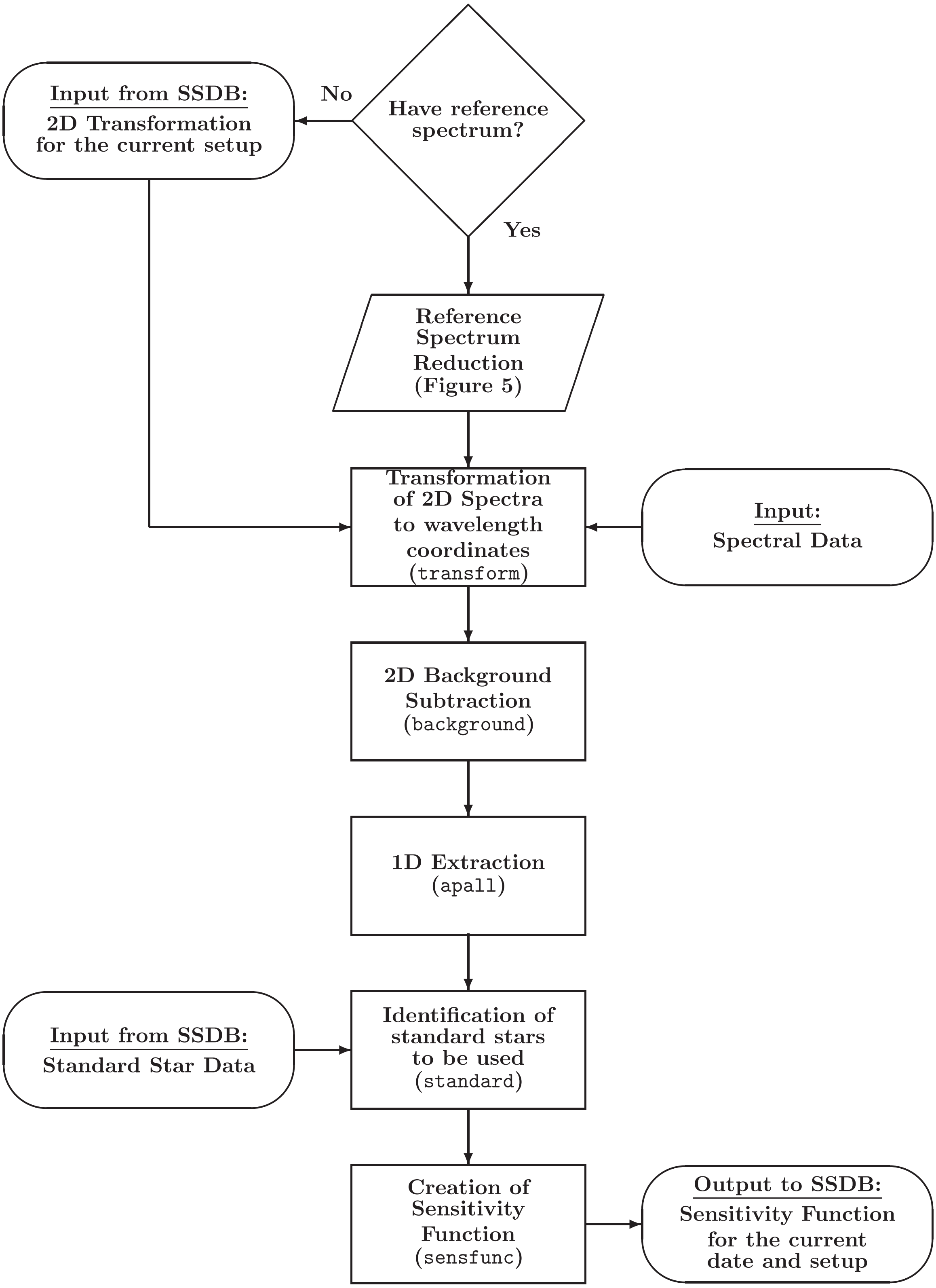
Input from SSDB:
2D Transformation for the current setup
No
Have reference spectrum?
Yes
Reference Spectrum Reduction (Figure 5)
Transformation of 2D Spectra to wavelength coordinates (transform)
Input:
Spectral Data
2D Background Subtraction (background)
1D Extraction (apall)
Input from SSDB:
Standard Star Data
Identification of standard stars to be used (standard)
Creation of Sensitivity Function (sensfunc)
Output to SSDB:
Sensitivity Function for the current date and setup

Figure 6: Flow chart for the Standard star reduction using pipeline

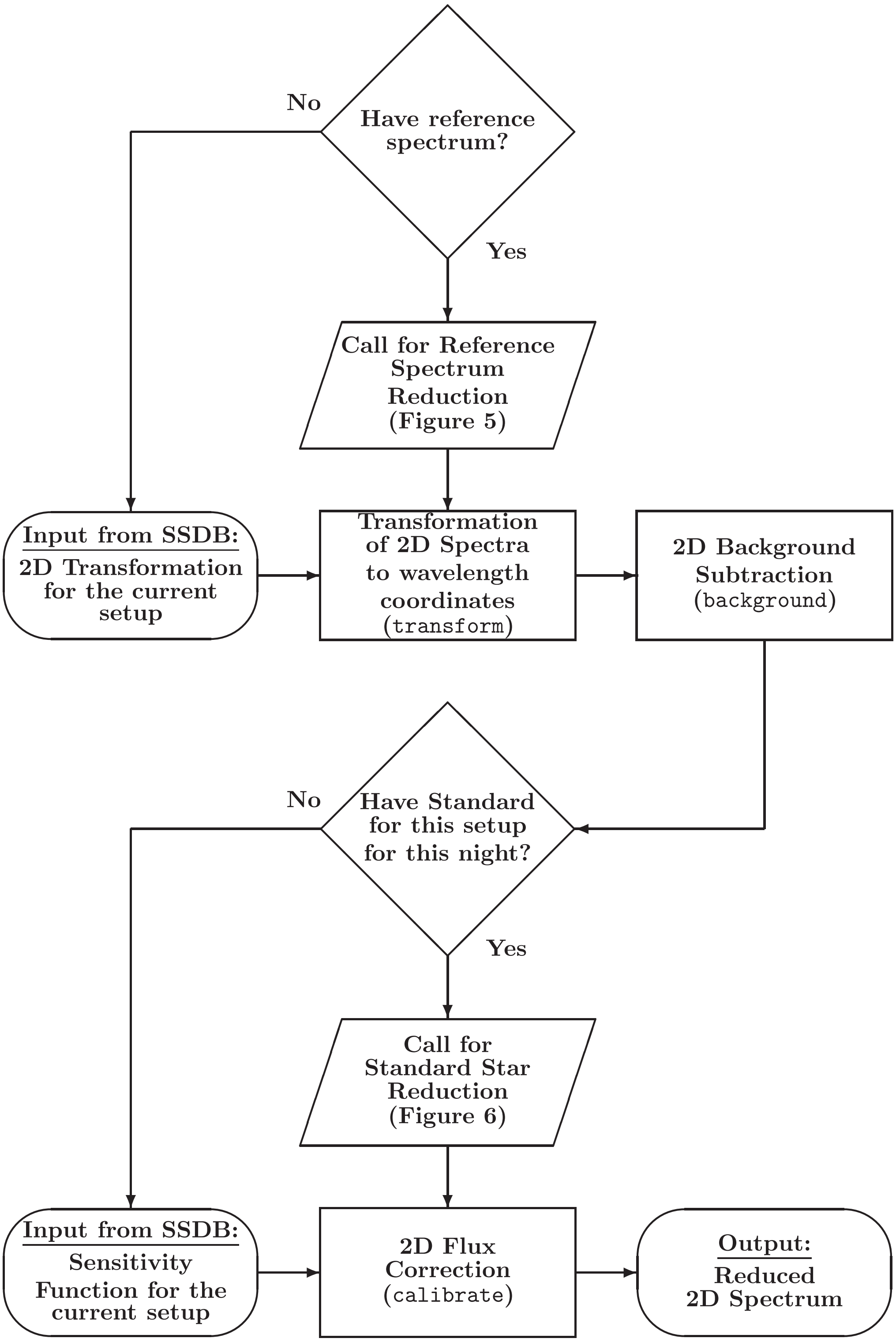
Have reference spectrum?
No
Yes
Call for Reference Spectrum Reduction (Figure 5)
Input from SSDB:
2D Transformation for the current setup
Transformation of 2D Spectra to wavelength coordinates (transform)
2D Background Subtraction (background)
Have Standard for this setup for this night?
No
Yes
Call for Standard Star Reduction (Figure 6)
Input from SSDB:
Sensitivity Function for the current setup
2D Flux Correction (calibrate)
Output:
Reduced 2D Spectrum

Figure 7: Flow chart for the object reduction using pipeline

5 Data Quality Control

As a check on the performance of the data reduction, the following statistics have to be recorded for each scientific spectrum:

- the formal accuracy of 2D polinomial fit for the wavelength transformation;
- the wavelength for begining, middle and last pixel for the current spectral setup?
- the wavelength for the standard night sky lines: 5577, 6300, 6363?