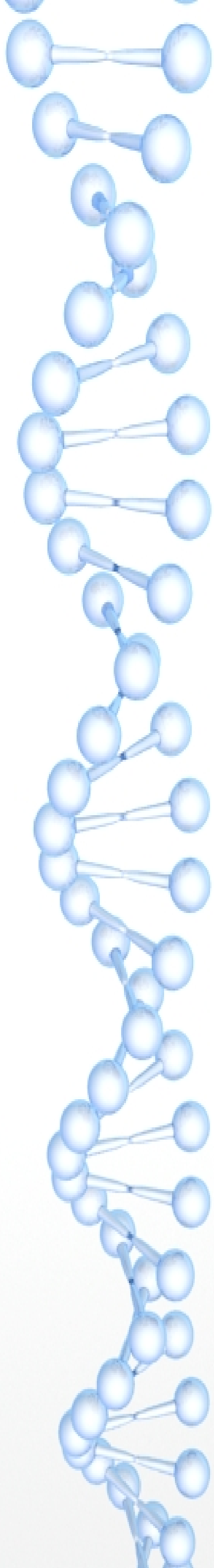
# Long-slit data reduction: some advices

From: Alexei Kniazev



# The plan of this tedious talk

1. **Very short introduction**

2. **Some advices**

3. **One example of the system of shell scripts for the long-slit reduction**

# Short Introduction

This presentation – is JUST my opinion. It is up to you to take it into account seriously on NOT

# IT Advices

- Hate Windows – it is dead way! (Example)

- Study UNIX (Linux/Mac OS)

- Study UNIX shell and scripting in it. Continuously during your work you operate with different files. Shell makes this job efficient

- Be ready to study as many as possible different programming languages – you never know what you will need in the future.

# Data Reduction Advices (1)

- **Study ANY (MANY?) standard systems for the reduction of astronomical data (SSRAD): IRAF (PyRAF), MIDAS, IDL …**

- **Read as many tutorials as possible. (Example)**

- **Try to NEVER use the high level programming language (C, C++, FORTRAN ...) for your data reduction. Always try to use one/two/many SSRADs: as minimum 90% of your needs are there.**

- **If you did something 2 (TWO) times, ALWAYS try to make some script to do the same thing for the third and above times**

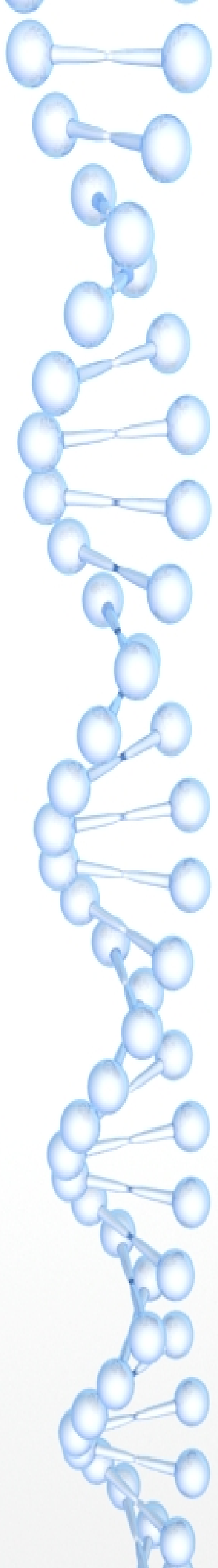
# Data Reduction Advices (2)

(1) If you know how to reduce ONE type of the data and you obtained another one, which you do not know how to reduce – first, what you need is to think: Does any SSRAD can reduce it?

(2) IF the answer is NO – then: Does anyone did reduction of such data before? What did this person use?

(3) IF the answer is NO – then: How to transfer the second type of the data to the first one (Example with long-slit and MOS)?

(4) And only here you need to think about to do something by yourself for these new type of observational data



# Data Reduction Advices (3)

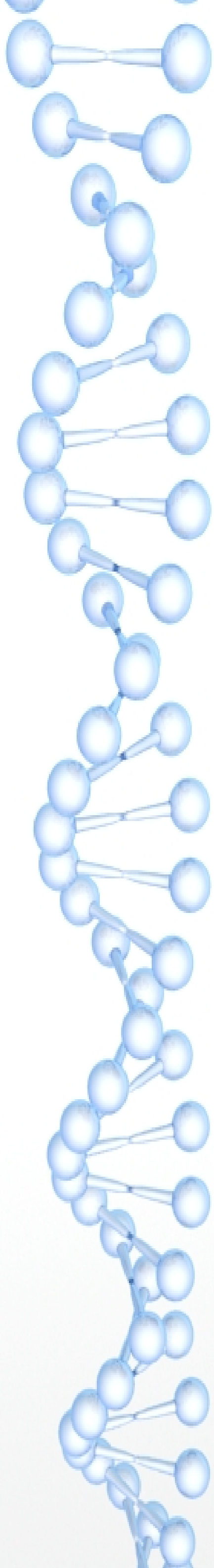
**Before to reduce ANY observational data think carefully about:**

**(1) Uniformity of directories structure (names for directories with primary reduced data, finally reduced data, dependence of these names from setup and night names, … etc)**

**(2) Uniformity of names convention (how will file names changed after EACH step of your reduction)**

**(3) Data types for your input/output data: Input as FITS, output as FITS? BINARY Table? FITS with extensions?**

# Data Reduction Advice:
# Long-Slit Reduction



PySALT or IRAF or MIDAS
or something else?

You need to select it by yourself!

# Data Reduction Advice:
# Long-Slit Reduction

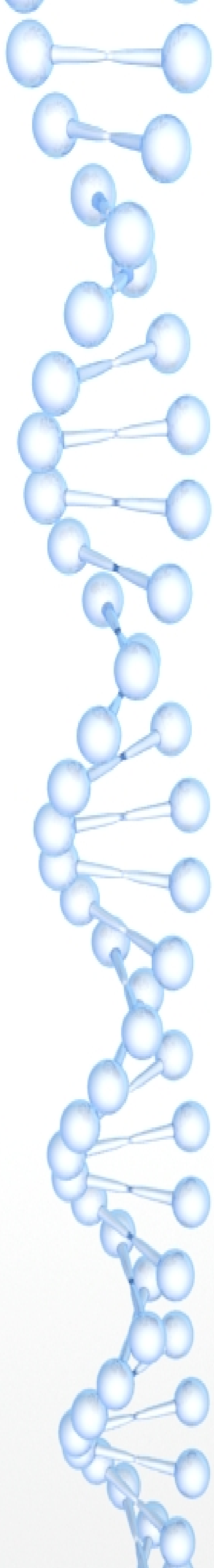
## My choice was:

UNIX SHell +IRAF = SHyRAF

**+ sometimes: MIDAS, IDL, PyRAF, ftool, perl ...**

# Long-Slit Reduction with IRAF

**The 2D long-slit reduction is a standard and very well known procedure for people using IRAF. The longslit and apextract packages have all the possibilities to produce this reduction in the correct way.**



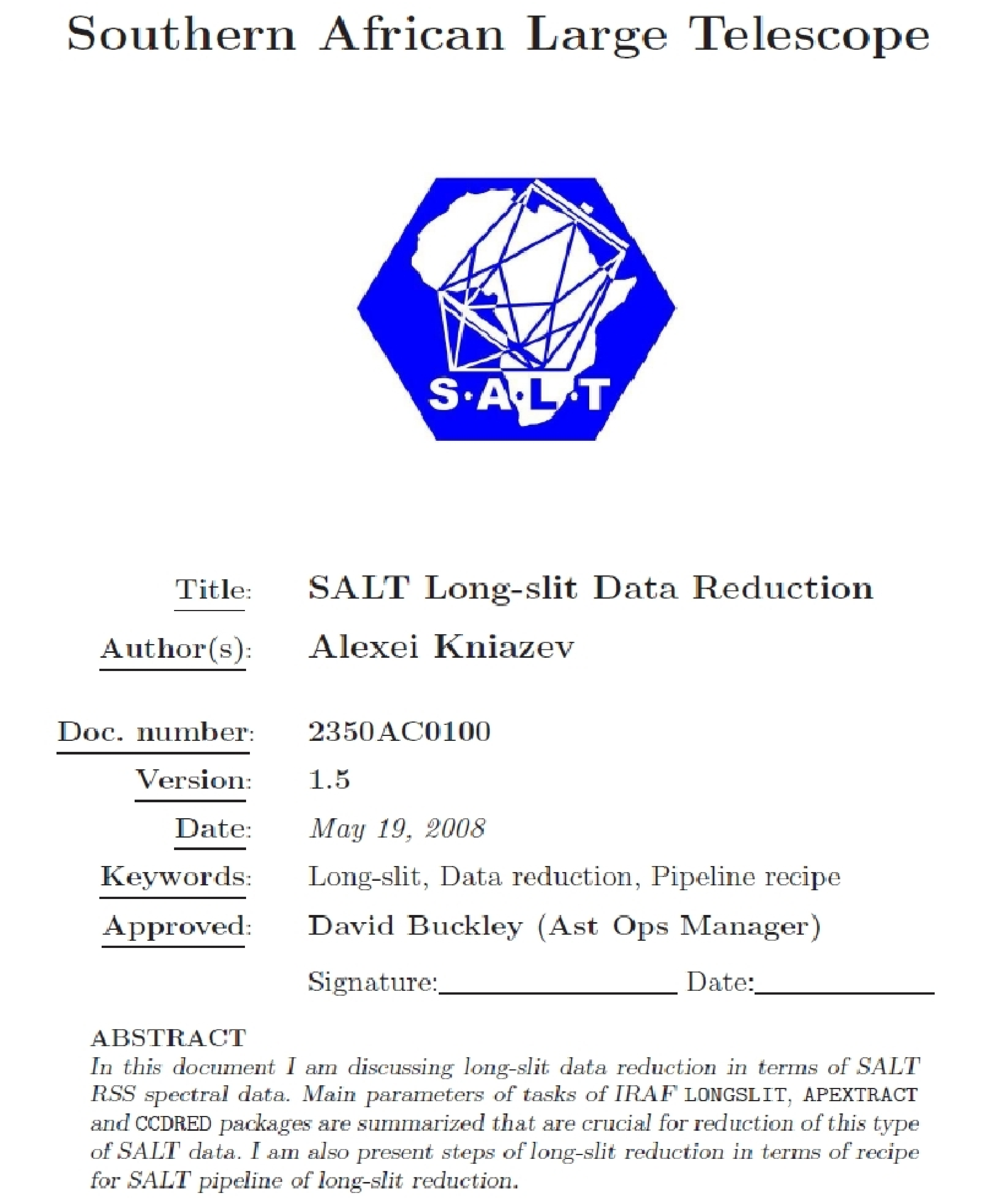
# Long-Slit Reduction with IRAF

**IRAF commands and flow-charts of long-slit reduction for ANY long-slit data from ANY telescope are shown:**

https://sciencewiki.salt.ac.za/index.php/Long_Slit_Reduction_Recipe

**Or**

**http://www.saao.ac.za/~akniazev/pub/SALT_Long_slit.pdf**



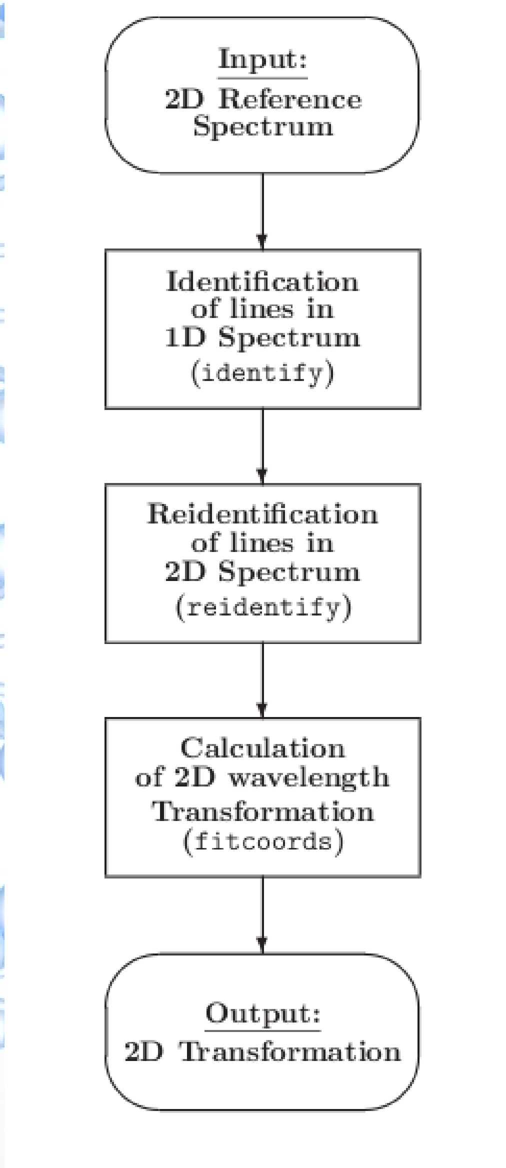
# Southern African Large Telescope

| | |
|---|---|
| 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.*



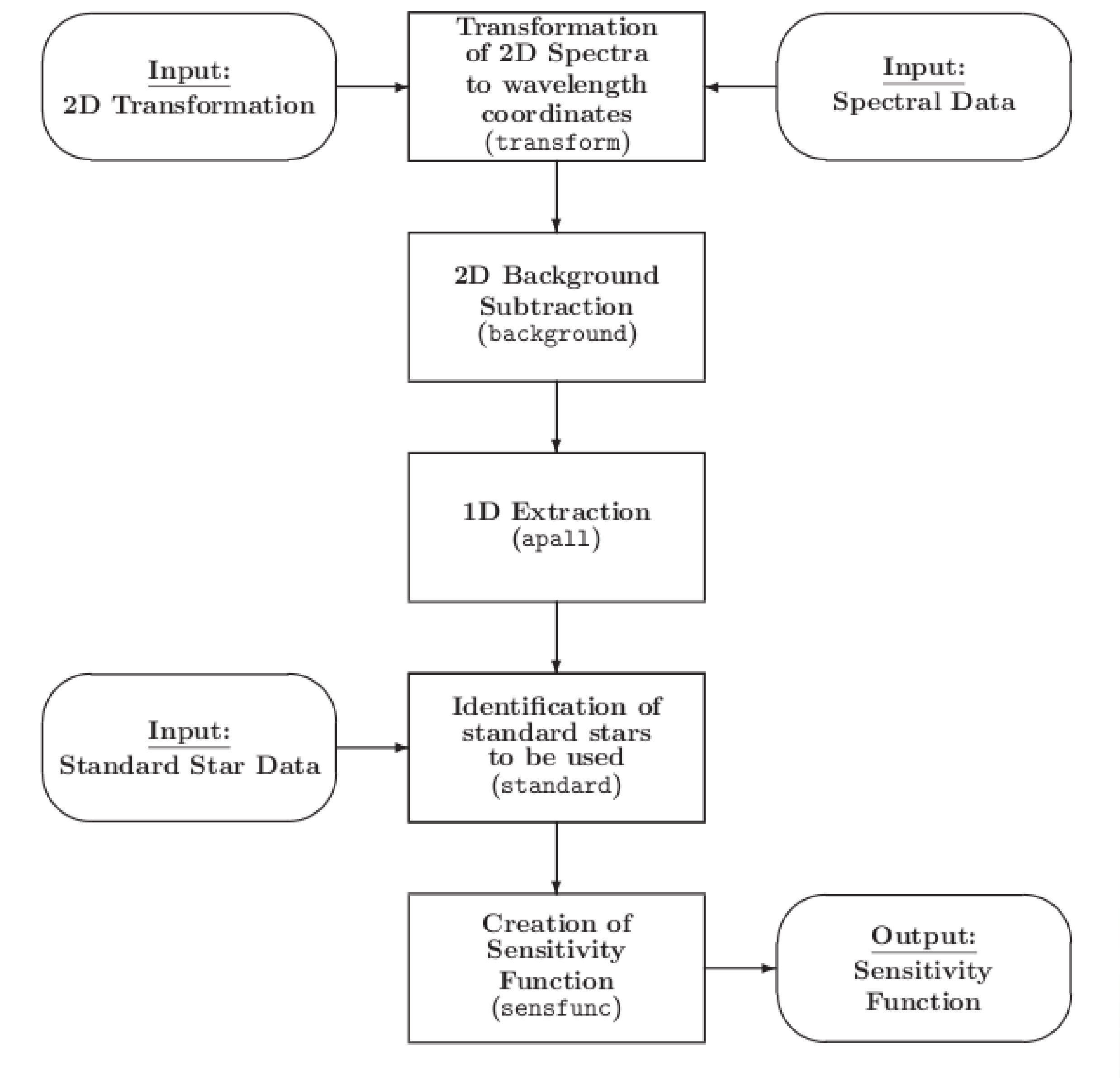
Input:
2D Reference Spectrum

Identification of lines in 1D Spectrum (identify)

Reidentification of lines in 2D Spectrum (reidentify)

Calculation of 2D wavelength Transformation (fitcoords)

Output:
2D Transformation



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

# Long-Slit Reduction with IRAF

As a VERY LAZY PERSON who reduce A LOT of non-uniform data from different telescopes with different spectral setups I can say:

- “I AM BORED” to follow a uniform system of file naming, where after each step of the standard reduction, file name has to be changed in the “standard” way
- “I AM BORED” to follow a uniform way to save and restore different spectral configurations for different dates
- “I AM BORED” to follow a uniform system to save and restore spectrophotometric standards/sensitivity curves for the different spectral configurations for different dates
- “IT IS practically IMPOSSIBLE” to save and restore specific parameters for the different IRAF tasks to reduce data of different spectral configurations and/or different dates and/or different hardware (telescope+instrument)

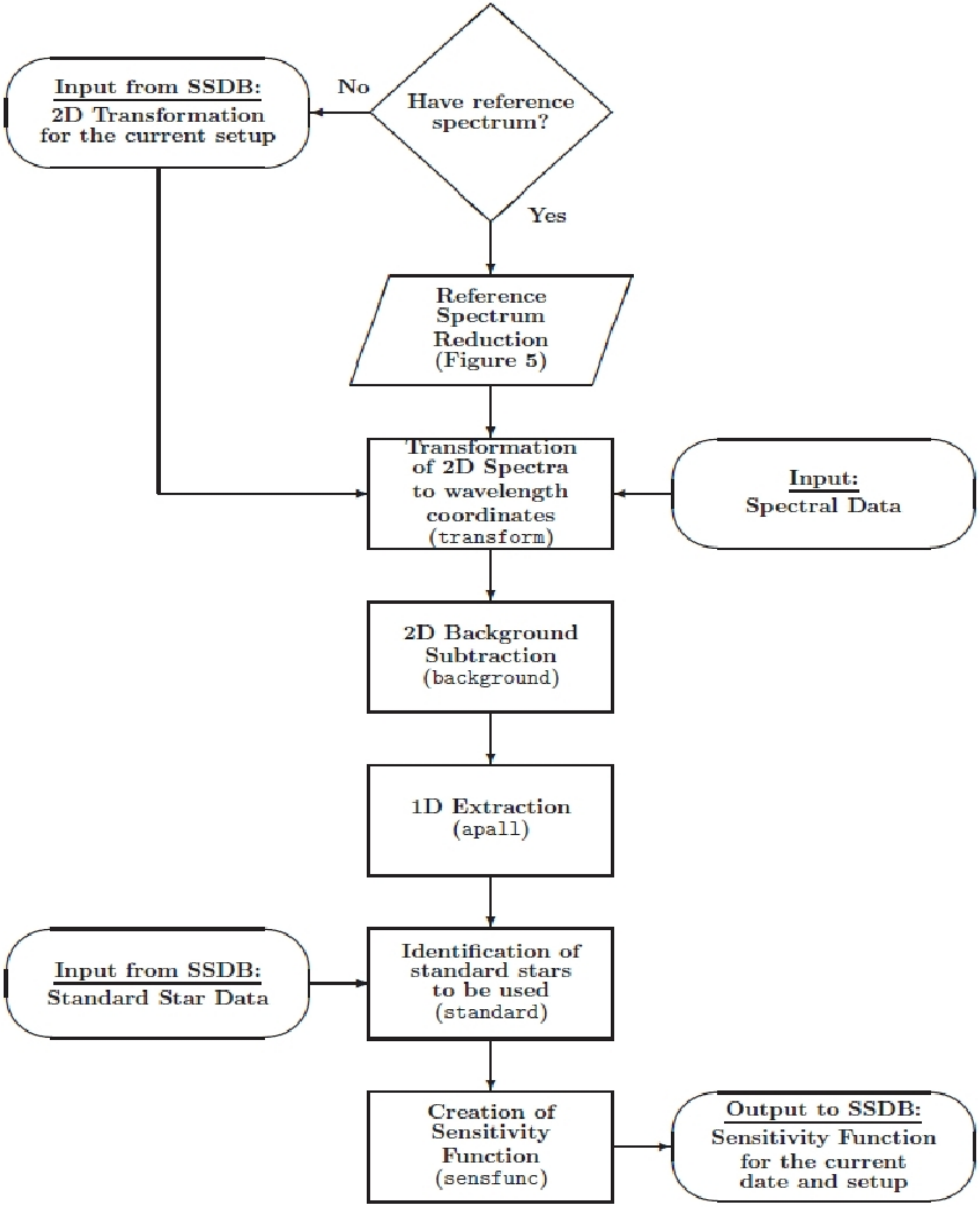

Input from SSDB:
2D Transformation for the current setup
No
Have reference spectrum?
Yes
Reference Spectrum Reduction (Figure 5)
Transformation of 2D Spectra to wavelength coordinates (transform)
Input:
Spectral Data
2D Background Subtraction (background)
1D Extraction (apall)
Input from SSDB:
Standard 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

# Southern African Large Telescope

S·A·L·T

Title: **Long-slit reduction with SHyRAF package RELSA**

Author(s): Alexei Y. Kniazev

Doc. number: 2350AC0099

Version: 1.0

Date: *December 15, 2006*

Keywords: Long-slit, Data reduction, Pipeline recipe

Approved: David Buckley (Ast Ops Manager)

Signature:________________ Date:____________

# Long-slit Reduction: RAIL

**The easiest and efficient way to solve most of above problems was to write my own scripts:**

- **Shell calls for any specific tool/programs you need to make a reduction of data: IRAF, IDL, MIDAS, PyRAF, ftool, perl etc**

- **Shell makes temporary IRAF program “IRAF_file_name.cl” that finally is executed in the standard way cl<IRAF_file_name.cl**

- **Since Shell was designed for the file-system operations, it is Shell responsibility to check that all directories, files and configuration files exist and/or properly copied**

- **It is also Shell responsibility to form correct file names, lists and react to any errors**

# RAIL features

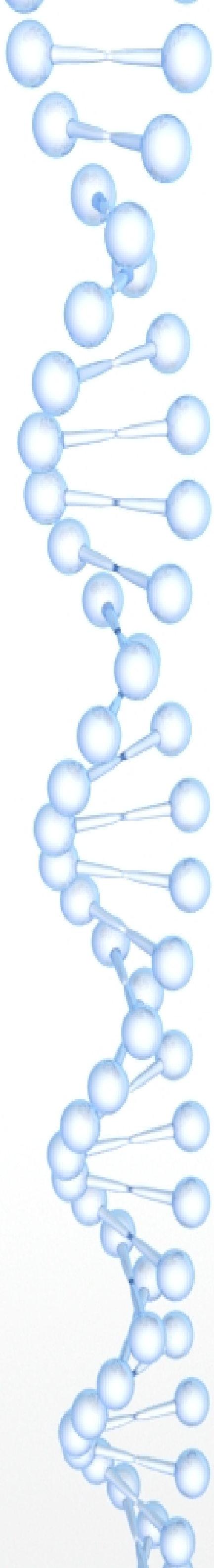
- Suggested that user HAS experience with long-slit reduction in IRAF

- Suggested that input FITS-files were preliminary reduced: (1) bias, (2) gain, (3) cross-talk (4) mosaic was done, (5) cosmic was cleaned and (6) spectral flat was done

- Suggested that FITS-files DO NOT HAVE extensions and FITS-header is CORRECT

- Needs that IRAF, PyRAF (basic) were installed on your computer

- Could be installed into any directory (GLOBAL), but this directory has to exist in your ${PATH} environment



# RAIL features

- Has one configuration file “LONG_epar.sh”
- Any main task or subroutine, when it is executed, looks for this configuration file in the local directory first and in the global directory second. All above means that user can have different local conf. files in each data reduction directory
- Has uniform names convention, but user can change it
- Each program call has format

```
LONG_<program> param1=value param2=value … paramN=value
```

# RAIL features

Currently RAIL can reduce data from below instruments:

- RSS - long-slit mode of Robert Stobie Spectrograph of SALT
- SCORPIO - long-slit mode of SCORPIO multi-mode instrument installed in the prime focus of the SAO RAS 6m telescope
- GSPEC - Grating Spectrograph with SITE CCD at SAAO 1.9m telescope (removed from 1.9m)
- TWIN – long-slit spectrograph at 3.5m telescope at Calar Alto (Spain)

# RAIL example 1

1. I have my RSS data observed on 20061026 and primary reduced. Files are:
mrpP20061026003ee.fits – Science exposure
mrpP20061026004ee.fits – Another Science exposure
mrpP20061026005ee.fits – ThAr spectrum for this setup
mrpP20061026017ee.fits – Spectrophotometric standard

2. I never reduced such setup yet.

3.
```
ls mrpP20061026003ee.fits >obj.lst
ls mrpP20061026005ee.fits >arc.lst
```

4. `Long_red.sh obj=obj.lst arc=arc.lst` – first play with object and arc to be sure that wavelength calibration
is fine and parameters of background procedure produce correct result.

5.
```
ls mrpP2006102600[34]ee.fits >obj.lst
ls mrpP20061026005ee.fits >arc.lst
ls mrpP20061026017ee.fits >std.lst
```

6. `Long_red.sh obj=obj.lst arc=arc.lst std=std.lst` – since solution for this setup was saved, all reduction will be done in the automatic/interactive way (user's choice in `LONG_epar.sh`) till the end, including spectrophotometric standard and correction for it.

# RAIL example 2

1. I have my RSS data observed on 20161026 and primary reduced. Files are:

```
mrpP20161026003ee.fits – Science exposure (setup 1)
mrpP20161026004ee.fits – Another Science exposure (setup 1)
mrpP20161026005ee.fits – ThAr spectrum for this setup 1
mrpP20161026006ee.fits – Science exposure (setup 2)
mrpP20161026007ee.fits – Another Science exposure (setup 2)
mrpP20161026008ee.fits – Ar spectrum for this setup 2
mrpP20161026017ee.fits – Spectrophotometric standard for setup 1
mrpP20161026018ee.fits – Spectrophotometric standard for setup 2
```

2. I reduced both these setups before (previous nights of observations)

3\.

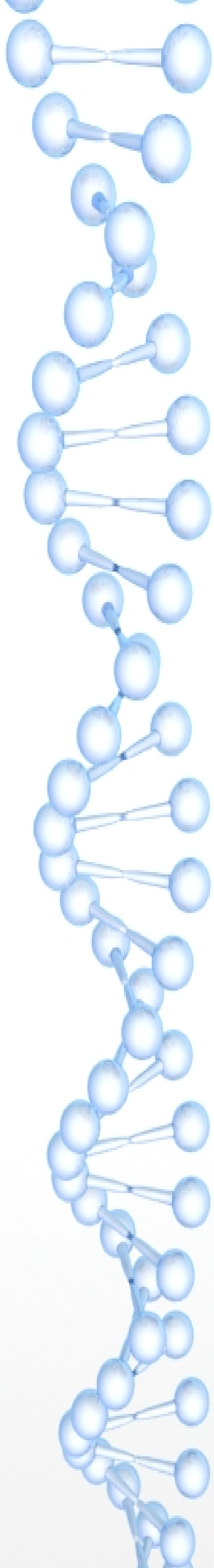
```
ls mrpP2006102600[34]ee.fits >obj1.lst
ls mrpP20061026005ee.fits >arc1.lst
ls mrpP20061026017ee.fits >std1.lst
ls mrpP2006102600[67]ee.fits >obj2.lst
ls mrpP20061026008ee.fits >arc2.lst
ls mrpP20061026018ee.fits >std1.lst
```

4\.

```
Long_red.sh obj=obj1.lst arc=arc1.lst std=std1.lst
Long_red.sh obj=obj2.lst arc=arc2.lst std=std2.lst
```

5. OR:

```
for I in 1 2;do Long_red.sh obj=obj$i.lst arc=arc$i.lst std=std$i.lst;done
```

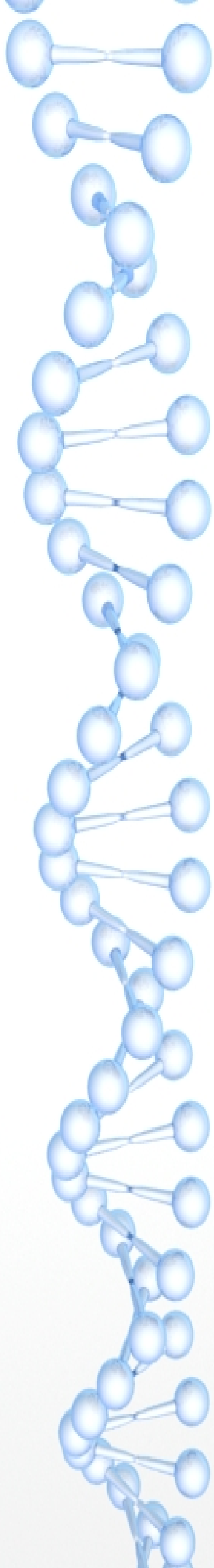


# RAIL residential address

**The current version of manual is located at:**
**http://www.saao.ac.za/~akniazev/pub/RAIL_README.pdf**

**The current version of the package is located at:**
**http://www.saao.ac.za/~akniazev/pub/RAIL.tar**

**Somebody: Does RAIL has errors?**
**Alexei: Yes, surely. I tested it only by myself.**



# The End