

---

# nifty Documentation

*Release 1.0.0.dev1*

**Nat Comeau**

**Sep 21, 2017**



---

## Contents

---

<b>1</b>	<b>User Manual</b>	<b>3</b>
<b>2</b>	<b>Package Documentation</b>	<b>5</b>
2.1	Introduction . . . . .	5
2.2	Running from the Command Line . . . . .	5
2.3	Preparing the .cfg Input File . . . . .	11
2.4	Data Reduction Examples . . . . .	13
2.5	Tutorials . . . . .	13
2.6	Known Issues . . . . .	15
2.7	Maintaining Nifty . . . . .	16
2.8	Future Work . . . . .	17
2.9	Changelog . . . . .	17
2.10	API . . . . .	18
2.11	Example of Nifty File I/O . . . . .	18
<b>3</b>	<b>Indices and tables</b>	<b>33</b>



Nifty is a full-featured Gemini NIFS data reduction pipeline and data reduction package.



# CHAPTER 1

---

## User Manual

---

For an in-depth explanation of the underlying Python and I/O, see `Nifty user manual`.





### Introduction

### Running from the Command Line

Nifty is started with the `runNifty` command by specifying a pipeline or step with arguments and options.

The syntax used to start Nifty is:

```
runNifty <Pipeline or Step Name> <arguments>
```

To get help or list the available options, type the `runNifty` command without any arguments.

```
runNifty
```

### Examples of Running from the Command Line

Starting a Data Reduction from the Beginning

To start the NIFS pipeline, populating a configuration file interactively:

```
runNifty nifsPipeline -i
```

To start the NIFS pipeline from a pre-built configuration file:

```
runNifty nifsPipeline config.cfg
```

To do a fully automatic data reduction, downloading raw data from the Gemini Public Archive (Eg: GN-2013A-Q-62):

```
runNifty nifsPipeline -f GN-2013A-Q-62
```

To do a fully automatic data reduction, using raw data from a local directory (Eg: `/Users/ncomeau/data/TUTORIAL`):

```
runNifty nifsPipeline /Users/ncomeau/data/TUTORIAL
```

### Starting a Data Reduction from a Specified Point

You can run each step, one at a time, from the command line like so. *You need a config.cfg file in your current working directory to run individual steps. Each step requires the general config section and its unique config section to be populated.*

You can also run an individual step by turning them on or off in nifsPipeline config and running the nifsPipeline.

**nifsSort:** To only copy and sort NIFS raw data type:

```
runNifty nifsSort
```

Or use a config.cfg file like this:

```
# Nifty configuration file.
#
# Each section lists parameters required by a pipeline step.

manualMode = False
over = False
merge = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = True
calibrationReduction = False
telluricReduction = False
scienceReduction = False

[sortConfig]
rawPath = ''
program = ''
skyThreshold = 2.0
sortTellurics = True
date = ''
copy = ''

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 6
telluricSkySubtraction = True
spectemp = ''
mag = ''
hline_method = 'vega'
hlineinter = False
continuuminter = False

[scienceReductionConfig]
sciStart = 1
sciStop = 6
scienceSkySubtraction = True
```

```
telluricCorrectionMethod = ''
telinter = False
fluxCalibrationMethod = ''
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

**nifsBaselineCalibration:** To only reduce calibrations type:

```
runNifty nifsBaselineCalibration
```

Or use a config.cfg file like this:

```
# Nifty configuration file.
#
# Each section lists parameters required by a pipeline step.

manualMode = False
over = False
merge = True
scienceDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/obs28', '/
↳ Users/ncomeau/tests/newQ62/AEGISz1284/20130530/H/obs55', '/Users/ncomeau/tests/
↳ newQ62/AEGISz1284/20130530/H/obs36', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳ 20130531/H/obs36', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130621/H/obs36', '/
↳ Users/ncomeau/tests/newQ62/AEGISz1284/20130622/H/obs44', '/Users/ncomeau/tests/
↳ newQ62/AEGISz1284/20130624/H/obs75', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳ 20130626/H/obs83']
calibrationDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/
↳ Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130530/Calibrations_H',
↳ '/Users/ncomeau/tests/newQ62/AEGISz1284/20130531/Calibrations_H', '/Users/ncomeau/
↳ tests/newQ62/AEGISz1284/20130621/Calibrations_H', '/Users/ncomeau/tests/newQ62/
↳ AEGISz1284/20130622/Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳ 20130624/Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130626/
↳ Calibrations_H']
telluricDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/
↳ obs26', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/obs51', '/
↳ Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/obs30', '/Users/ncomeau/
↳ tests/newQ62/AEGISz1284/20130530/H/Tellurics/obs53', '/Users/ncomeau/tests/newQ62/
↳ AEGISz1284/20130530/H/Tellurics/obs38', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳ 20130531/H/Tellurics/obs34', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130531/H/
↳ Tellurics/obs64', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130621/H/Tellurics/obs69
↳ ', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130622/H/Tellurics/obs42', '/Users/
↳ ncomeau/tests/newQ62/AEGISz1284/20130622/H/Tellurics/obs46', '/Users/ncomeau/tests/
↳ newQ62/AEGISz1284/20130624/H/Tellurics/obs73', '/Users/ncomeau/tests/newQ62/
↳ AEGISz1284/20130624/H/Tellurics/obs77', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳ 20130626/H/Tellurics/obs81', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130626/H/
↳ Tellurics/obs85']

[nifsPipelineConfig]
sort = False
calibrationReduction = True
telluricReduction = False
scienceReduction = False
```

```
[sortConfig]
rawPath = ''
program = ''
skyThreshold = 2.0
sortTellurics = True
date = ''
copy = ''

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 6
telluricSkySubtraction = True
spectemp = ''
mag = ''
hline_method = 'vega'
hlineinter = False
continuuminter = False

[scienceReductionConfig]
sciStart = 1
sciStop = 6
scienceSkySubtraction = True
telluricCorrectionMethod = ''
telinter = False
fluxCalibrationMethod = ''
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

**nifsReduce Telluric:** To only reduce telluric data type:

```
runNifty nifsReduce Telluric
```

Or use a config.cfg file like this:

```
# Nifty configuration file.
#
# Each section lists parameters required by a pipeline step.

manualMode = False
over = False
merge = True
scienceDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/obs28', '/
↳ Users/ncomeau/tests/newQ62/AEGISz1284/20130530/H/obs55', '/Users/ncomeau/tests/
↳ newQ62/AEGISz1284/20130530/H/obs36', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳ 20130531/H/obs36', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130621/H/obs36', '/
↳ Users/ncomeau/tests/newQ62/AEGISz1284/20130622/H/obs44', '/Users/ncomeau/tests/
↳ newQ62/AEGISz1284/20130624/H/obs75', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳ 20130626/H/obs83']
```

```

calibrationDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/
↳Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130530/Calibrations_H',
↳'/Users/ncomeau/tests/newQ62/AEGISz1284/20130531/Calibrations_H', '/Users/ncomeau/
↳tests/newQ62/AEGISz1284/20130621/Calibrations_H', '/Users/ncomeau/tests/newQ62/
↳AEGISz1284/20130622/Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130624/Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130626/
↳Calibrations_H']
telluricDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/
↳obs26', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/obs51', '/
↳Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/obs30', '/Users/ncomeau/
↳tests/newQ62/AEGISz1284/20130530/H/Tellurics/obs53', '/Users/ncomeau/tests/newQ62/
↳AEGISz1284/20130530/H/Tellurics/obs38', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130531/H/Tellurics/obs34', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130531/H/
↳Tellurics/obs64', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130621/H/Tellurics/obs69
↳', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130622/H/Tellurics/obs42', '/Users/
↳ncomeau/tests/newQ62/AEGISz1284/20130622/H/Tellurics/obs46', '/Users/ncomeau/tests/
↳newQ62/AEGISz1284/20130624/H/Tellurics/obs73', '/Users/ncomeau/tests/newQ62/
↳AEGISz1284/20130624/H/Tellurics/obs77', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130626/H/Tellurics/obs81', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130626/H/
↳Tellurics/obs85']

[nifsPipelineConfig]
sort = False
calibrationReduction = False
telluricReduction = True
scienceReduction = False

[sortConfig]
rawPath = ''
program = ''
skyThreshold = 2.0
sortTellurics = True
date = ''
copy = ''

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 6
telluricSkySubtraction = True
spectemp = ''
mag = ''
hline_method = 'vega'
hlineinter = False
continuuminter = False

[scienceReductionConfig]
sciStart = 1
sciStop = 6
scienceSkySubtraction = True
telluricCorrectionMethod = ''
telinter = False
fluxCalibrationMethod = ''
use_pq_offsets = True
im3dtran = True

```

```
# Good luck with your Science!
```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

**nifsReduce Science:** To only reduce science data type:

```
runNifty nifsReduce Science
```

Or use a config.cfg file like this:

```
# Nifty configuration file.
#
# Each section lists parameters required by a pipeline step.

manualMode = False
over = False
merge = True
scienceDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/obs28', '/
↳Users/ncomeau/tests/newQ62/AEGISz1284/20130530/H/obs55', '/Users/ncomeau/tests/
↳newQ62/AEGISz1284/20130530/H/obs36', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130531/H/obs36', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130621/H/obs36', '/
↳Users/ncomeau/tests/newQ62/AEGISz1284/20130622/H/obs44', '/Users/ncomeau/tests/
↳newQ62/AEGISz1284/20130624/H/obs75', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130626/H/obs83']
calibrationDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/
↳Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130530/Calibrations_H',
↳'/Users/ncomeau/tests/newQ62/AEGISz1284/20130531/Calibrations_H', '/Users/ncomeau/
↳tests/newQ62/AEGISz1284/20130621/Calibrations_H', '/Users/ncomeau/tests/newQ62/
↳AEGISz1284/20130622/Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130624/Calibrations_H', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130626/
↳Calibrations_H']
telluricDirectoryList = ['/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/
↳obs26', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/obs51', '/
↳Users/ncomeau/tests/newQ62/AEGISz1284/20130527/H/Tellurics/obs30', '/Users/ncomeau/
↳tests/newQ62/AEGISz1284/20130530/H/Tellurics/obs53', '/Users/ncomeau/tests/newQ62/
↳AEGISz1284/20130530/H/Tellurics/obs38', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130531/H/Tellurics/obs34', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130531/H/
↳Tellurics/obs64', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130621/H/Tellurics/obs69
↳', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130622/H/Tellurics/obs42', '/Users/
↳ncomeau/tests/newQ62/AEGISz1284/20130622/H/Tellurics/obs46', '/Users/ncomeau/tests/
↳newQ62/AEGISz1284/20130624/H/Tellurics/obs73', '/Users/ncomeau/tests/newQ62/
↳AEGISz1284/20130624/H/Tellurics/obs77', '/Users/ncomeau/tests/newQ62/AEGISz1284/
↳20130626/H/Tellurics/obs81', '/Users/ncomeau/tests/newQ62/AEGISz1284/20130626/H/
↳Tellurics/obs85']

[nifsPipelineConfig]
sort = False
calibrationReduction = False
telluricReduction = False
scienceReduction = True

[sortConfig]
rawPath = ''
program = ''
skyThreshold = 2.0
```

```

sortTellurics = True
date = ''
copy = ''

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 6
telluricSkySubtraction = True
spectemp = ''
mag = ''
hline_method = 'vega'
hlineinter = False
continuuminter = False

[scienceReductionConfig]
sciStart = 1
sciStop = 6
scienceSkySubtraction = True
telluricCorrectionMethod = ''
telinter = False
fluxCalibrationMethod = ''
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!

```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

## Preparing the .cfg Input File

Nifty reads data reduction parameters with a config parser developed by Michael Foord. See <http://www.voidspace.org.uk/python/configobj.html> for full documentation on the parser.

## Interactive Input Preparation

The best way to learn about what each config file parameter does is to populate an input file interactively by typing:

```
runNifty nifsPipeline -i
```

This will, for each parameter, print an explanation and supply a default parameter that you can accept by pressing enter. The output file is named “config.cfg”.

## An Example Input File

Nifty includes a default configuration file in the runtimeData/ directory. As of v1.0b2, It looks like this: .. TODO(nat): This is out of data! Update this!

```
# Nifty configuration file.
#
# Each section lists parameters required by a pipeline step.

manualMode = False
over = False
merge = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = True
calibrationReduction = True
telluricReduction = True
scienceReduction = True

[sortConfig]
rawPath = ''
program = ''
skyThreshold = 2.0
sortTellurics = True
date = ''
copy = ''

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 6
telluricSkySubtraction = True
spectemp = ''
mag = ''
hline_method = 'vega'
hlineinter = False
continuuminter = False

[scienceReductionConfig]
sciStart = 1
sciStop = 6
scienceSkySubtraction = True
telluricCorrectionMethod = ''
telinter = False
fluxCalibrationMethod = ''
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```



## Data Reduction Examples

### Observations of Titan

Recipe used: defaultConfig.cfg

### Observations of a Moderate Redshift Galaxy

Recipe used: defaultConfig.cfg

Let's reduce NIFS data of a moderate redshift galaxy, located at  $z \sim 1.284$ . This is a faint target, so after making individual cubes we use the reported telescope P and Q offsets to blindly merge our final cubes.

As this program is out of its proprietary period and available on the Gemini Public Archive, we can use the default-Config.cfg configuration file and specify its program ID to reduce it.

```
runNifty -f GN-2013A-Q-62
```

We could also launch the reduction from a provided configuration file.

Contents of the configuration file:

```
TODO(nat): When finalized fill this out!
```

To launch the reduction:

```
runNifty <configurationFile>
```

## Tutorials

### H Line Removal

The H-line removal can be done non-interactively, but it is advised that this be performed interactively and using the “vega\_tweak” method in order to accurately scale the vega spectrum. In the interactive mode for the initial scaling and call to “telluric” these are the cursor keys and colon commands (from <http://iraf.net/irafhelp.php?val=telluric&help=Help+Page>):

- ? - print help
- a - automatic RMS minimization within sample regions
- c - toggle calibration spectrum display
- d - toggle data spectrum display
- e - expand (double) the step for the current selection
- q - quit
- r - redraw the graphs
- s - add or reset sample regions
- w - window commands (see :/help for additional information)
- x - graph and select from corrected shifted candidates
- y - graph and select from corrected scaled candidates

- `:help` - print help
- `:shift [value]` - print or reset the current shift
- `:scale [value]` - print or reset the current scale
- `:dshift [value]` - print or reset the current shift step
- `:dscale [value]` - print or reset the current scale step
- `:offset [value]` - print or reset the current offset between spectra
- `:sample [value]` - print or reset the sample regions
- `:smooth [value]` - print or reset the smoothing box size

To decrease the scale or shift value, the cursor must be under the spectrum and to increase these values the cursor must be above the spectrum. Occasionally, this will not work in which case the value can be designated with a colon command.

If using the `vega_tweak` or other interactive line removal method, the lines can be removed in a `splot` environment (commands found here: <http://stsdas.stsci.edu/cgi-bin/gethelp.cgi?splot.hlp>). The most useful commands for this are:

- `k + (g, l or v)`

Mark two continuum points and fit a single line profile. The second key selects the type of profile: `g` for gaussian, `l` for lorentzian, and `v` for voigt. Any other second key defaults to gaussian. The center, continuum at the center, core intensity, integrated flux, equivalent width, and FWHMs are printed and saved in the log file. See `d` for fitting multiple profiles and `-` to subtract the fit.

- `w`

Window the graph. For further help type `?` to the “window:” prompt or see help under `gtools`. To cancel the windowing use `a`.

It is necessary to press ‘`i`’ before ‘`q`’ once the h-lines have been removed in order to save the changes.

## Custom Telluric Corrections

You can supply your own continuum-normalized telluric correction spectrum by placing a “`telluricCorrection.fits`” one D spectra and a “`fit.fits`” that you used to normalize that spectra in the science observation directory. Both spectra must be 2040 pixels long and have the data extension located in header unit zero. Note that overwrite must be turned off for this to work.

## Merging Data Cubes

Nifty offers a few ways to merge data cubes. These are all contained in the `nifsMerge.py` script.

**Note:** If cubes are not transposed each call to `iraf.imcombine()` can take 25 minutes or more. The current implementation is much slower at combining when cubes have a y-offset.

Cubes Were Shifted by Hand with `QFitsView` or Similar

A user can shift cubes by hand and add the prefix “`shift`” to the cube name. The pipeline will automatically find these cubes and combine them with `gemcube`.

If no “`shift`”-prefixed cubes exist the user has two more choices to make. The first is to generate offsets automatically or to provide an offsets file by hand.

Generating an `offsets.txt` File

If `use_pq_offsets` is `True` in the `config.cfg` file, *Nifty* will determine offsets automatically from the `POFFSET` and `QOFFSET` entry of each cubes `.fits` header. Otherwise, *Nifty* will pause and wait for you to provide a suitably formatted `offsets.txt` file in the `scienceObjectName/Merged/date_obsid/` directory.

Using `iraf.im3dtran()`

Cubes have been found to combine ~50 times faster when the `y` and `lambda` axis are swapped with `iraf.im3dtran()`. In our tests we found it took ~25 minutes to merge cubes without transposition and ~0.5 minutes to merge cubes in a transposed state.

We have found the cubes produced with and without transposition to be identical. We have made the default not to use transposition but we urge further testing.

## Known Issues

### nifsPipeline.py

### nifsSort.py

Object and Sky frame differentiation

If the sorting script does not create a skylist in the object or telluric observation directories this means that the offsets between sky frames and object frames were smaller than expected. A skyframelist can be manually created and saved in the appropriate directory, or the limit placed on the offset can be changed. To lower the limit manually do a search in `nifsSort.py` for “`skyframelist.append(entry)`” and change the `<= <value_in_arcseconds>` to something new.

### nifsBaselineCalibration.py

- **Long data file path names are not fed to IRAF tasks.** It seems IRAF task parameters

must be 99 characters or less. Nifty’s data files are stored in the Astroconda environment packages directory; for example, on my system it is “`/Users/nat/miniconda2/envs/niftypip/lib/python2.7/site-packages/nifty/pipeline/`”. If you have a long username, for example, this can cause the path name to be too long to be parsed by `iraf`. *Temporary fix:* I have made the names of all the data files short enough for it to work okay on my system. **Please let me know if this seems to be causing you issues and I can come up with a better fix.**

### nifsReduce.py

- `z-band` data is not capable of a flux calibration (yet!).
- Seems to be missing the first peak of the ronchi slit when `iraf.nfsdist()` is run interactively. This does not seem to be a problem.
- We noticed that `iraf.nfsdist()` produced different results when run interactively and non-interactively. To test check the log for `iraf.nfsdist()` output and make sure it is identifying 8/8 or 9/9 peaks every time.
- `iraf.nftelluric()` was not built to be run automatically. A lightly tested modified version that allows an automatic telluric correction is included in the `extras` directory but more testing is needed. For now `applyTelluricPython()` is recommended.

## nifsMerge.py

## nifsUtils.py

## General Issues

- A longstanding bug (see [astropy](#) ) in astropy has made it difficult to build Nifty as a binary executable.
- The conversion of print statements to logging.info() statements was messy. Some of these may still not be properly converted and will throw nasty tracebacks. However these seem to have no effect on the functioning of the code.
- Logging is still not perfect. One or two iraf tasks are sending their log files to “nifs.log” instead of “Nifty.log”.

## Maintaining Nifty

### Documentation

Right now there exists five forms of documentation.

Paper

README.rst

Manual

Nifty’s manual gives a broad overview of how the code functions. It is included as a pdf.

.rst Files in the docs/ directory

This file, others like it in the docs/ directory and the README are written in reStructuredText. This markup language integrates well with Python’s automatic documentation builder (we used Sphinx) and Github as well as being human readable. You can read more about reStructuredText [here](#).

Comments and DocStrings in Source Code

Nifty uses the Google docstring style. Examples of docstrings can be found [here](#).

Other Python comments use the following convention:

- A # is followed by a space and a capital letter.
- All comments end in a period where possible.

### Pipeline Structure

See the Nifty paper for a high level overview. Nifty general runs with the following procedure:

- A script in the scripts/ directory is called from the command line.
- This script imports the relevant pipeline and steps from the nifty/pipeline/ and nifty/pipeline/steps/directories.
- The script then launches the appropriate pipeline,
- This pipeline launches the appropriate steps,
- These steps launch the appropriate routines, and
- These routines launch the appropriate sub-routines.

Nifty is built at the lowest level from Python and IRAF subroutines. It is built so that it is relatively easy to change the implementation of the underlying task.

## Updates

### Version Numbers

Nifty uses semantic versioning(see <http://semver.org/>). This means version numbers come in

`MAJOR.MINOR.PATCH`

In brief, when releasing a version of Nifty that is not backward-compatible with old test recipes, or changes break the public API, it is time to increment the MAJOR version number.

..TODO(nat): maybe make this a little clearer.

## Code Conventions

Nifty was partly written using [atom](#). Error messages, warnings and updates were partly written using templates in the included snippets.cson file.

Where possible, nat used 2D (and higher) dimensional lists to implement error checking flags. These are particularly prominent in sort.

Variables and functions were named using conventions in the [Python Style Guide](#). Specifically a mix of camelCase and lower\_case\_with\_underscores was used.

## Future Work

Throughout the code, ncomeau has placed many TODO notes. These are good to check out and could be a good starting point.

Some other nice to haves:

- Add Python 3 compatability(if possible)
- Ability to restart individual steps easily
- Add some sort of status indicators... Like progress
- Paralellisation?
- Analysis tools: automatic velocity field? Dispersion?

## Changelog

All notable changes to this project will be documented in this file.

The format is based on [Keep a Changelog](#) and this project adheres to [Semantic Versioning](#).

## Unreleased

All in-development changes will be tracked here.

- Adding unit tests for each step and integration test for pipeline.
- Finished integrating multiple types of cube merging.

## 1.0b4 - 2017-09-12

Much refined and patched Beta release. Still not finished but much more robust.

- Verified overwrite; it seems to be safe to use now.
- Fixing telluric correction and absolute flux calibration.
- Added 1D extraction routine.
- Preliminary addition of three types of cubes and cube merging.

## 1.0b1 - 2017-09-08

Preliminary Beta release.

- Syntax errors mean this version will not compile.
- Fixing merge flip due to differences between NIFS + ALTAIR and NIFS w/o ALTAIR on the bottom port.

## 1.0a1 - 2017-08-31

Preliminary Alpha release.

- .whl uploaded to PIP, docs uploaded to

ReadTheDocs and preliminary DOI assigned.

## API

## Example of Nifty File I/O

This is an example of how the Nifty directory tree appears after each step of the data reduction. These directory trees were created using a custom **nifftree** bash command:

```
find . -name .git -prune -o -print | sed -e 's;[^/]*;/;|____;g;s;____|;|;g'
```

Add the following line to your ~/.bash\_profile to create the **nifftree** alias:

```
alias nifftree="find . -name .git -prune -o -print | sed -e 's;[^/]*;/;|____;g;s;____|;|;g' "
```

## nifsPipeline Data Reduction

Config file used (slightly out of date but still a useful example):

```
# Nifty configuration file.
#
# Each section lists parameters required by a pipeline step.

manualMode = True
over = False
merge = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = True
calibrationReduction = True
telluricReduction = True
scienceReduction = True

[sortConfig]
rawPath = '/Users/ncomeau/data/TUTORIAL_HD141004'
program = ''
skyThreshold = 2.0
sortTellurics = True
date = ''
copy = ''

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 6
telluricSkySubtraction = True
spectemp = ''
mag = ''
hline_method = 'vega'
hlineinter = False
continuuminter = False

[scienceReductionConfig]
sciStart = 1
sciStop = 6
scienceSkySubtraction = True
telluricCorrectionMethod = 'gnirs'
telinter = False
fluxCalibrationMethod = 'gnirs'
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

Starting directory structure:

```
.
|___config.cfg
```

Command used to launch Nifty:

```
runNifty nifsPipeline config.cfg
```

Directory structure after sorting:

```
.
|___config.cfg
|___HD141004/                # Object name, from science header
| |___20100401/              # UT date, from science header
| | |___Calibrations_K/      # Calibrations for a given science observation
| | | |___arcdarklist        # Textfile list of lamps-off arc frames
| | | |___arclist            # Textfile list of lamps-on arc frames
| | | |___flatdarklist       # Textfile list of lamps-off flats; same_
↪length as flatlist
| | | |___flatlist           # Textfile list of lamps-on flats; same length_
↪as flatdarklist
| | | |___N201004*.fits       # Raw Calibration Frames
| | | |___original_flatdarklist # Unmodified textfile list of lamps-off flats
| | | |___original_flatlist   # Unmodified textfile list of lamps-on flats
| | | |___ronchilist          # Textfile list of lamps-on ronchi flats
| | | |___K/                  # Grating of science and telluric frames
| | | |___obs107/             # Science observation, from science headers
| | | | |___N201004*.fits     # Raw science frames
| | | | |___scienceFrameList  # Textfile list of science frames
| | | | |___skyFrameList      # Textfile list of science sky frames
| | | | |___Tellurics/
| | | | |___obs109/           # A single standard star observation directory
| | | | |___N201004*.fits     # Raw standard star frames
| | | | |___scienceMatchedTellsList # Textfile matching telluric observations with_
↪science frames
| | | | |___skyFrameList      # Textfile list of standard star sky frames
| | | | |___telist            # Textfile list of standard star frames
|___Nifty.log                 # Master log file
```

Now in nifsBaselineCalibration:

After Step 1: Get Shift, two new files appear.

```
.
|___config.cfg
|___HD141004/
| |___20100401/
| | |___Calibrations_K/
| | | |___arcdarklist
| | | |___arclist
| | | |___flatdarklist
| | | |___flatlist
| | | |___N201004*.fits
| | | |___original_flatdarklist
| | | |___original_flatlist
| | | |___ronchilist
| | | |___shiftfile          # Textfile storing name of the reference shift file
| | | |___sN20100410S0362.fits # Reference shift file; a single lamps-on flat run_
↪through nfprepare
```



```
|____Nifty.log
```

After Step 2: Make Flat and bad pixel mask, several new files and intermediate results appear.

```
.
|____config.cfg
|____HD141004/
| |____20100401/
| | |____Calibrations_K/
| | | |____arcdarklist
| | | |____arclist
| | | |____flatdarklist
| | | |____flatfile           # Textfile storing name of final flat
| | | |____flatlist
| | | |____gnN20100410S0362.fits      # Median-combined with gemcombine() and
↳prepared lamps-on flat
| | | |____gnN20100410S0368.fits      # Median-combined with gemcombine() and
↳prepared lamps-off flat
| | | |____N201004*.fits
| | | |____nN201004*.fits           # Result of running raw frames through
↳nfprepare()
| | | |____original_flatdarklist
| | | |____original_flatlist
| | | |____rgnN20100410S0362.fits      # Result of running gemcombine() lamps-on
↳flats through nsreduce()
| | | |____rgnN20100410S0362_flat.fits  # Final rectified flat; result of
↳nsslitfunction()
| | | |____rgnN20100410S0362_sflat.fits # Preliminary flat; result of nsflat()
| | | |____rgnN20100410S0362_sflat_bpm.pl # Final bad pixel mask; later used in
↳nffixbad()
| | | |____rgnN20100410S0368.fits      # Result of running gemcombine() lamps-
↳off flats through nsreduce()
| | | |____rgnN20100410S0368_dark.fits  # Final flat dark frame
| | | |____ronchilist
| | | |____sflat_bpmfile           # Textfile storing name of final bad
↳pixel mask
| | | |____sflatfile
| | | |____shiftfile
| | | |____sN20100410S0362.fits
|____Nifty.log
```

After Step 3: Wavelength Solution, similar files are created as well as a database/  
↳directory containing wavelength solutions for each slice.

```
.. code-block:: text
```

```
.
|____config.cfg
|____HD141004/
| |____20100401/
| | |____Calibrations_K/
| | | |____arcdarkfile
| | | |____arcdarklist
| | | |____arclist
| | | |____database/           # Contains textfile results from
↳nswavelength(), nfsdist(), nffitcoords(), nifcube()
| | | |____idwrgnN20100401S0137_SCI_*_ # Textfiles containing wavelength
↳solutions for a particular slice
```

```
| | | |___flatdarklist
| | | |___flatfile
| | | |___flatlist
| | | |___gnN20100401S0137.fits          # Median-combined with gemcombine() arc_
↪dark frame
| | | |___gnN20100410S0362.fits
| | | |___gnN20100410S0368.fits
| | | |___gnN20100410S0373.fits          # Median-combined with gemcombine() arc_
↪frame
| | | |___N201004*.fits
| | | |___nN201004*.fits                # Results of running raw frames through_
↪nfprepare()
| | | |___original_flatdarklist
| | | |___original_flatlist
| | | |___rgnN20100401S0137.fits        # Results from nsreduce() of combined arc_
↪dark frame
| | | |___rgnN20100410S0362.fits
| | | |___rgnN20100410S0362_flat.fits
| | | |___rgnN20100410S0362_sflat.fits
| | | |___rgnN20100410S0362_sflat_bpm.pl
| | | |___rgnN20100410S0368.fits
| | | |___rgnN20100410S0368_dark.fits
| | | |___ronchilist
| | | |___sflat_bpmfile
| | | |___sflatfile
| | | |___shiftfile
| | | |___sN20100410S0362.fits
| | | |___wrgnN20100401S0137.fits      # Final wavelength calibration frame
|___Nifty.log
```

After Step 4: Spatial Distortion, the last step of the calibration reduction, more files are added to the database directory.

```
.
|___config.cfg
|___HD141004/
| |___20100401/
| | |___Calibrations_K/
| | | |___arcdarkfile
| | | |___arcdarklist
| | | |___arclist
| | | |___database/
| | | | |___idrN20100410S0375_SCI_*_      # Textfiles containing spatial solutions_
↪for particular slices
| | | | |___idwrgnN20100401S0137_SCI_*_
| | | |___flatdarklist
| | | |___flatfile
| | | |___flatlist
| | | |___gnN20100401S0137.fits
| | | |___gnN20100410S0362.fits
| | | |___gnN20100410S0368.fits
| | | |___gnN20100410S0373.fits
| | | |___gnN20100410S0375.fits          # Median combined with gemcombine()_
↪lamps-on ronchi frame
| | | |___N201004*.fits
| | | |___nN20100401S0137.fits          # Results of running raw lamps-on ronchi_
↪frames through nfprepare()
| | | |___original_flatdarklist
| | | |___original_flatlist
```

```

| | | |____rgnN20100401S0137.fits
| | | |____rgnN20100410S0362.fits
| | | |____rgnN20100410S0362_flat.fits
| | | |____rgnN20100410S0362_sflat.fits
| | | |____rgnN20100410S0362_sflat_bpm.pl
| | | |____rgnN20100410S0368.fits
| | | |____rgnN20100410S0368_dark.fits
| | | |____rgnN20100410S0375.fits          # Results of running combined lamps-on_
↪ronchi frame through nsreduce() AND nfsdist()
| | | |____ronchifile                    # Text file storing name of final ronchi_
↪frame
| | | |____ronchilist
| | | |____sflat_bpmfile
| | | |____sflatfile
| | | |____shiftfile
| | | |____sN20100410S0362.fits
| | | |____wrgnN20100401S0137.fits
|____Nifty.log

```

The final directory structure after `nifsBaselineCalibration`, should look something like. The products used by appropriate standard star and science observation directories are the “rgn” prefixed final ronchi file, the “wrgn” prefixed final wavelength solution file, the “database/” directory, the “s” prefixed shiftfile, the “rgn” prefixed and “\_flat.fits” suffixed final flat field correction frame, the “rgn” prefixed and “\_sflat\_bpm.pl” suffixed final bad pixel mask.

```

.
|____config.cfg
|____HD141004/                                # OT object name; from science frame .
↪fits headers
| |____20100401/                                # Date; from science frame .fits headers
| | |____Calibrations_K/                        # Calibrations directory; All the work_
↪in this step happens in one of these
| | | |____arcdarkfile                        # Text file storing name of final_
↪reduced arc dark
| | | |____arcdarklist                       # Text file storing name of arc dark_
↪frames
| | | |____arclist                           # Text file storing name of arc frames
| | | |____database/                          # Directory with text file results of_
↪nswavelength() and nfsdist()
| | | | |____idrgnN20100410S0375_SCI*_         # Textfiles containing spatial solutions_
↪for particular slices
| | | | |____idwrgnN20100401S0137_SCI*_        # Textfiles containing wavelength_
↪solutions for particular slices
| | | |____flatdarklist                      # Text file storing names of lamps-off_
↪flats; pipeline uses this, not original_flatlist
| | | |____flatfile                          # Text file storing name of final flat_
↪field correction frame, corrected for slice to slice variation
| | | |____flatlist                          # Text file storing names of lamps-on_
↪flats; pipeline uses this, not original_flatlist
| | | |____gnN20100401S0137.fits               # Median combined and prepared arc frame
| | | |____gnN20100410S0362.fits               # Median combined and prepared lamps-on_
↪flat
| | | |____gnN20100410S0368.fits               # Median combined and prepared lamps-off_
↪flat
| | | |____gnN20100410S0373.fits               # Median combined and prepared arc dark_
↪frame
| | | |____gnN20100410S0375.fits               # Median combined and prepared lamps-on_
↪ronchi frame
| | | |____N201004*.fits                      # Raw calibration frames

```

```

| | | |____nN20100401S0137.fits          # Results of running raw lamps-on ronchi_
↳frames through nfprepare()
| | | |____original_flatdarklist          # Text file list of lamps-off flats, NOT_
↳taking P and Q offset zero-points into account
| | | |____original_flatlist              # Text file list of lamps-on flats, NOT_
↳taking P and Q offset zero-points into account
| | | |____rgnN20100401S0137.fits          # Final reduced, combined and prepared_
↳arc frame
| | | |____rgnN20100410S0362.fits          # Final reduced, combined and prepared_
↳lamps-on flat
| | | |____rgnN20100410S0362_flat.fits      # Final flat field correction frame,_
↳corrected for slice to slice variations with nsslitfunction()
| | | |____rgnN20100410S0362_sflat.fits     # Preliminary flat field correction_
↳frame. Result of nsflat()
| | | |____rgnN20100410S0362_sflat_bpm.pl  # Final bad pixel mask. Result of_
↳nsflat()
| | | |____rgnN20100410S0368.fits          # Final reduced, combined and prepared_
↳lamps-off flat frame
| | | |____rgnN20100410S0368_dark.fits      # Final flat field correction dark frame;
↳ result of nsflat()
| | | |____rgnN20100410S0375.fits          # Results of running combined lamps-on_
↳ronchi frame through nsreduce() AND nfsdist()
| | | |____ronchifile                     # Text file storing name of final ronchi_
↳frame
| | | |____ronchilist                     # Text file list of lamps-on ronchi flat_
↳frames
| | | |____sflat_bpmfile                   # Text file storing name of final bad_
↳pixel mask frame
| | | |____sflatfile                       # Text file storing name of preliminary_
↳flat field correction frame
| | | |____shiftfile                       # Text file storing name of shift file;_
↳used to get consistent shift to the MDF
| | | |____sN20100410S0362.fits            # Shift file; used to get consistent_
↳shift to MDF. Result of nfprepare()
| | | |____wrgnN20100401S0137.fits          # Final wavelength solution frame._
↳Result of nswavelength()
|____Nifty.log                            # Logfile; all log files should go here.

```

## nifsReduce of Tellurics

After Step 1: Locate the Spectrum, calibrations are copied over from the appropriate calibrations directory and each raw frame is run through nfprepare().

```

.
|____config.cfg
|____HD141004/
| |____20100401/
| | |____K/
| | | |____Tellurics/
| | | | |____obs109/
| | | | |____database/          # Database from appropriate_
↳calibrations directory
| | | | | |____idrgnN20100410S0375_SCI_*_  # Spatial distortion database text files
| | | | | |____idwrgnN20100401S0137_SCI_*_ # Wavelength solution database text_
↳files
| | | | | |____N201004*.fits

```

```

| | | | | ____nN201004*.fits # Results of running each raw frame_
↳through nfprepare()
| | | | | ____rgnN20100410S0375.fits # Final reduced ronchi flat frame from_
↳appropriate calibrations directory
| | | | | ____scienceMatchedTellsList
| | | | | ____skyFrameList
| | | | | ____telist
| | | | | ____wrgnN20100401S0137.fits # Final reduced arc frame from_
↳appropriate calibrations directory
| ____Nifty.log

```

After Step 2: Sky Subtraction, the only files that are written are in standard star observation directories. Each prepared standard star frame is sky subtracted with `gemarith()`, and then the sky-subtracted prepared frames are median combined into one frame.

```

obs109/
| ____database/
| | ____idrgnN20100410S0375_SCI_*_
| | ____idwrgnN20100401S0137_SCI_*_
| ____gnN20100401S0139.fits # Single median-combined standard star frame
| ____N201004*.fits
| ____nN201004*.fits
| ____rgnN20100410S0375.fits
| ____scienceMatchedTellsList
| ____skyFrameList
| ____snN201004*.fits # Sky subtracted, prepared standard star frames
| ____telist
| ____wrgnN20100401S0137.fits

```

After Step 3: Flat fielding and Bad Pixels Correction:

```

obs109/
| ____brsnN20100401S0138.fits # Flat fielded and bad pixels corrected standard_
↳frames; results of nffixbad()
| ____database/
| | ____idrgnN20100410S0375_SCI_*_
| | ____idwrgnN20100401S0137_SCI_*_
| ____gnN20100401S0139.fits
| ____N201004*.fits
| ____nN201004*.fits
| ____rgnN20100410S0375.fits
| ____rsnN201004*.fits # Flat fielded standard frames; results of_
↳nsreduce()
| ____scienceMatchedTellsList
| ____skyFrameList
| ____snN201004*.fits
| ____telist
| ____wrgnN20100401S0137.fits

```

After Step 4: 2D to 3D transformation and Wavelength Calibration:

```

obs109/
| ____brsnN201004*.fits
| ____database/
| | ____fcfbrsnN20100401S0138_SCI_*_lamp # Textfile result of nffitcoords()
| | ____fcfbrsnN20100401S0138_SCI_*_sdist # Textfile result of nffitcoords()
| | ____fcfbrsnN20100401S0140_SCI_*_lamp
| | ____fcfbrsnN20100401S0140_SCI_*_sdist

```

```

| |____fcfbrsnN20100401S0142_SCI_*_lamp
| |____fcfbrsnN20100401S0142_SCI_*_sdist
| |____fcfbrsnN20100401S0144_SCI_*_lamp
| |____fcfbrsnN20100401S0144_SCI_*_sdist
| |____fcfbrsnN20100401S0146_SCI_*_lamp
| |____fcfbrsnN20100401S0146_SCI_*_sdist
| |____idrgnN20100410S0375_SCI_*_
| |____idwrgnN20100401S0137_SCI_*_
|____fbrsnN201004*.fits          # Results of nffitcoords()
|____gnN20100401S0139.fits
|____N201004*.fits
|____nN201004*.fits
|____rgnN20100410S0375.fits
|____rsnN201004*.fits
|____scienceMatchedTellsList
|____skyFrameList
|____snN201004*.fits
|____telist
|____tfbrsnN20100401S0138.fits  # Results of nftransform()
|____wrgnN20100401S0137.fits

```

After Step 5: Extract 1D Spectra and Make Combined Telluric:

```

obs109/
|____brsnN201004*.fits
|____database/
| |____fcfbrsnN201004*_SCI_*_lamp
| |____fcfbrsnN201004*_SCI_*_sdist
| |____idrgnN20100410S0375_SCI_*_
| |____idwrgnN20100401S0137_SCI_*_
|____fbrsnN201004*.fits
|____gnN20100401S0139.fits
|____gxtfbrsnN20100401S0138.fits  # Median-combined extracted standard star spectra;
↪ result of gemcombine()
|____N201004*.fits
|____nN201004*.fits
|____rgnN20100410S0375.fits
|____rsnN201004*.fits
|____scienceMatchedTellsList
|____skyFrameList
|____snN201004*.fits
|____telist
|____telluricfile                # Text file storing name of median-combined_
↪ extracted standard star spectrum.
|____tfbrsnN201004*.fits
|____wrgnN20100401S0137.fits
|____xtfbrsnN201004*.fits        # Extracted 1D standard star spectra; result of_
↪ nfextract()

```

After Step 6: Create Telluric Correction Spectrum, the telluric standard data reduction is complete. The final products of the reduction are telluricCorrection.fits, the final continuum-normalized telluric correction spectrum, and fit.fits, the continuum used to normalize the final telluric correction spectrum. These two products are copied to an appropriate science observation directory and used by the ‘gnirs’ telluric correction method.

```

obs109/
|____brsnN201004*.fits
|____database/
| |____fcfbrsnN201004*_SCI_*_lamp

```

```
| |____fcfbrsnN201004*_SCI*_sdist
| |____idrgnN201004*_SCI*_
| |____idwrgnN201004*_SCI*_
|____fbrsnN201004*.fits
|____final_tel_no_hlines_no_norm.fits # Final telluric correction spectrum NOT_
↳continuum normalized
|____fit.fits # Continuum used to normalize the final_
↳telluric correction spectrum
|____gnN20100401S0139.fits
|____gxtfbrsnN20100401S0138.fits
|____N201004*.fits
|____nN201004*.fits
|____rgnN20100410S0375.fits
|____rsnN201004*.fits
|____scienceMatchedTellsList
|____skyFrameList
|____snN201004*.fits
|____std_star.txt # Text file storing temperature and magnitude_
↳of standard star
|____tell_nolines.fits # H-line corrected standard star spectrum
|____telist
|____telluric_hlines.txt # Text file storing what linefitAuto() and_
↳linefitManual did. Empty file for now
|____telluricCorrection.fits # Final continuum-normalized telluric_
↳correction spectrum
|____telluricfile
|____tfbrsnN201004*.fits
|____wrgnN20100401S0137.fits
|____xtfbrsnN201004*.fits
PRODUCTS/
```

The final telluric observation directory structure after nifsReduce Tellurics:

```
obs109/ # Base standard star observation directory;_
↳from .fits headers
|____brsnN201004*.fits # Results of nffixbad()
|____database/ # Database directory containing text file_
↳results of nswavelength(), nfsdist(), nffitcoords()
| |____fcfbrsnN201004*_SCI*_lamp # Text file result of nffitcoords()
| |____fcfbrsnN201004*_SCI*_sdist # Text file result of nffitcoords()
| |____idrgnN201004*_SCI*_ # Text file result of nfsdist()
| |____idwrgnN201004*_SCI*_ # Text file result of nswavelength()
|____fbrsnN201004*.fits # Results of nffitcoords()
|____final_tel_no_hlines_no_norm.fits # Final telluric correction spectrum NOT_
↳continuum normalized
|____fit.fits # Continuum used to normalize the final_
↳telluric correction spectrum
|____gnN20100401S0139.fits # Median combined and prepared sky frame
|____gxtfbrsnN20100401S0138.fits # Final median-combined and extracted one D_
↳standard star spectrum; result of gemcombine()
|____N201004*.fits # Raw standard star and standard star sky_
↳frames
|____nN201004*.fits # Prepared standard star and standard star sky_
↳frames; results of nfprepare()
|____rgnN20100410S0375.fits # Final ronchi flat frame; copied from_
↳appropriate calibration directory. Result of nfsdist()
|____rsnN201004*.fits # Flat fielded, cut, sky subtracted, and_
↳prepared standard star frames. Results of nsreduce()
```

```
|____scienceMatchedTellsList      # Textfile used to match this standard star_
↳observation directory with certain science frames
|____skyFrameList                 # Textfile list of standard star sky frames
|____snN201004*.fits              # Sky subtracted and prepared standard star_
↳frames. Results of gemarith()
|____std_star.txt                 # Text file storing temperature and magnitude_
↳of standard star
|____tell_nolines.fits            # H-line corrected standard star spectrum
|____tellist                      # Text file list of standard star frames
|____telluric_hlines.txt          # Text file storing what linefitAuto() and_
↳linefitManual did. Empty file for now
|____telluricCorrection.fits       # Final continuum-normalized telluric_
↳correction spectrum
|____telluricfile                 # Text file storing name of final median-
↳combined and extracted one D standard star spectrum
|____tfbrsnN201004*.fits          # Results of nftransform()
|____wrgnN20100401S0137.fits      # Final reduced arc frame; copied from_
↳appropriate calibrations directory
|____xtfbrsnN201004*.fits         # One D extracted standard star spectra;_
↳results of nfextract()
PRODUCTS/                        # Products directory; currently not used for_
↳anything
```

## nifsReduce Science

After Step 1: locate the spectrum,

Our perspective is inside the science observation directory as all changes, until step 5, happen there.

```
obs107/
|____database/                   # Database directory and associated text files_
↳copied from the appropriate calibrations directory
| |____idrgnN20100410S0375_SCI*_
| |____idwrgnN20100401S0137_SCI*_
|____N201004*.fits              # Raw science and science sky frames
|____nN201004*.fits             # Prepared science and sky frames. Results of_
↳nfprepare()
|____original_skyFrameList       # Sky frame list without taking P and Q zero-
↳point offsets into account
|____rgnN20100410S0375.fits      # Final reduced ronchi flat; copied from_
↳appropriate calibrations directory
|____scienceFrameList           # Text file list of science frames
|____skyFrameList               # Text file list of science sky frames. If an_
↳original_skyFrameList exists, this is the result of taking P and Q zero-point_
↳offsets into account
|____wrgnN20100401S0137.fits     # Final reduce arc frame; copied from_
↳appropriate calibrations directory
```

After Step 2: Sky Subtraction. This is a bit different than the telluric sky subtraction as we do not subtract a median-combined sky frame from each science frame; we subtract the sky frame of (hopefully) same exposure time closest in time to the science frame from each science frame.

```
obs107
|____database/
| |____idrgnN20100410S0375_SCI*_
| |____idwrgnN20100401S0137_SCI*_
```



```

|___N201004*.fits
|___nN201004*.fits
|___original_skyFrameList
|___rgnN20100410S0375.fits
|___scienceFrameList
|___skyFrameList
|___snN201004*.fits           # Sky-subtracted and prepared science frames.
↪Results of gemarith()
|___wrgnN20100401S0137.fits

```

After Step 3: Flat Fielding and Bad Pixels Correction:

```

obs107/
|___brsnN201004*.fits           # Bad pixel corrected and flat fielded science frames.
↪ Results of nffixbad()
|___database/
| |___idrgnN201004*_SCI_*_
| |___idwrgnN201004*_SCI_*_
|___N201004*.fits
|___nN201004*.fits
|___original_skyFrameList
|___rgnN20100410S0375.fits
|___rsnN201004*.fits           # Flat fielded science frames. Results of nsreduce()
|___scienceFrameList
|___skyFrameList
|___snN201004*.fits
|___wrgnN20100401S0137.fits

```

After Step 4: 2D to 3D transformation and Wavelength Calibration

```

obs107/
|___brsnN201004*.fits
|___database/
| |___fcfbrsnN201004*_SCI*_lamp      # Text file result of nffitcoords()
| |___fcfbrsnN201004*_SCI*_sdist     # Text file result of nffitcoords()
| |___idrgnN20100410S0375_SCI_*_
| |___idwrgnN20100401S0137_SCI_*_
|___fbrsnN20100401S0182.fits        # Results of nffitcoords()
|___N201004*.fits
|___nN201004*.fits
|___original_skyFrameList
|___rgnN20100410S0375.fits
|___rsnN201004*.fits
|___scienceFrameList
|___skyFrameList
|___snN201004*.fits
|___tfbrsnN201004*.fits           # Results of nftransform()
|___wrgnN20100401S0137.fits

```

After Step 5: Make Uncorrected, Telluric Corrected and Flux Calibrated Data Cubes and Extracted One D Spectra:

Changes take place in both science observation directories AND objectName/ExtractedOneD/ directories.

In a science observation directory:

```

obs107/
|___actfbrsnN201004*.fits           # Final telluric corrected data cubes
|___bbodyN201004*.fits             # Unshifted or scaled blackbody used to flux
↪calibrate cubes

```

```

|___brsnN201004*.fits
|___combinedOneD # Textfile storing name of combined
↳extracted one D standard star spectra
|___ctfbrsnN201004*.fits # Final uncorrected data cubes
|___cubesliceN201004*.fits # One D extracted spectrum of cube used to
↳get telluric correction shift and scale
|___database/
| |___fcfbrsnN201004*_SCI*_lamp
| |___fcfbrsnN201004*_SCI*_sdist
| |___idrgnN20100410S0375_SCI*_
| |___idwrgnN20100401S0137_SCI*_
|___factfbrsnN201004*.fits # Final flux calibrated AND telluric
↳corrected data cubes
|___fbrsnN201004*.fits
|___finaltelCorN201004*.fits # Final shifted and scaled fit to telluric
↳correction
|___gxtfbrsnN20100401S0182.fits # One D extracted and combined standard star
↳used to derive the telluric correction used on these cubes
|___N201004*.fits
|___nN201004*.fits
|___oneDcorrectedN201004*.fits # One D telluric corrected slice of cube;
↳this was used to get the shift and scale of the final correction
|___original_skyFrameList
|___rgnN20100410S0375.fits
|___rsnN201004*.fits
|___scaledBlackBodyN201004*.fits # Blackbody scaled by flambda and ratio of
↳experiment times; telluric corrected cube multiplied by this
# to get flux calibrated AND telluric
↳corrected cube.
|___scienceFrameList
|___skyFrameList
|___snN201004*.fits
|___telCorN201004*.fits # UNSHIFTED AND SCALED telluric correction
↳for each science cube
|___telFitN201004*.fits # UNSHIFTED AND SCALED fit to telluric
↳correction for each science cube
|___tfbrsnN201004*.fits
|___wrgnN20100401S0137.fits
|___xtfbrsnN201004*.fits

```

In the scienceObjectName/ExtractedOneD/ directory:

```

ExtractedOneD/
|___20100401_obs107/ # Science data and observation, from .fits
↳headers of science frames
| |___xtfbrsnN201004*.fits # Extracted one D spectra from UNCORRECTED cubes.
↳Results of nfextract()
|___combined20100401_obs107.fits # Median-combined, extracted one D spectra.
↳Result of gemcombine()

```

The final science observation directory and scienceObservationName/ExtractedOneD/ directory should look something like this:

In each science directory:

```

obs107/
|___actfbrsnN201004*.fits # Final telluric corrected data cubes
|___bbodyN201004*.fits # Unshifted or scaled blackbody used to flux
↳calibrate cubes

```

```

|____brsnN201004*.fits           # Bad pixel corrected, reduced, sky
↳subtracted and prepared science frames
|____combinedOneD                # Textfile storing name of combined
↳extracted one D standard star spectra
|____ctfbrsnN201004*.fits        # Final uncorrected data cubes
|____cubesliceN201004*.fits      # One D extracted spectrum of cube used to
↳get telluric correction shift and scale
|____database/
| |____fcfbrsnN201004*_SCI*_lamp  # Text file results of nffitcoords()
| |____fcfbrsnN201004*_SCI*_sdist # Text file results of nffitcoords()
| |____idrgrN20100410S0375_SCI*_  # Text file results of nfsdlist()
| |____idwrgnN20100401S0137_SCI*_ # Text file results of nswavelength()
|____factfbrsnN201004*.fits      # Final flux calibrated AND telluric
↳corrected data cubes
|____fbrsnN201004*.fits          # Results of nffitcoords()
|____finaltelCorN201004*.fits    # Final shifted and scaled fit to telluric
↳correction
|____gxtfbrsnN20100401S0182.fits # Median-combined and extracted one D
↳spectra from UNCORRECTED cubes. Results of gemcombine()
|____N201004*.fits              # Raw science and science sky frames
|____nN201004*.fits             # Prepared raw science frames. Results of
↳nfprepare()
|____oneDcorrectedN201004*.fits   # One D telluric corrected slice of cube;
↳this was used to get the shift and scale of the final correction
|____original_skyFrameList       # Text file storing names of science sky
↳frames, not taking P and Q offset zero points into account
|____rgnN20100410S0375.fits      # Final reduced, combined and prepared
↳ronchi flat frame. Result of nfsdlist()
|____rsnN201004*.fits            # Flat fielded, sky subtracted and prepared
↳science frames. Result of nsreduce()
|____scaledBlackBodyN201004*.fits # Blackbody scaled by flambda and ratio of
↳experiment times; telluric corrected cube multiplied by this
# to get flux calibrated AND telluric
↳corrected cube.
|____scienceFrameList            # Text file storing names of science frames
|____skyFrameList                # Text file storing names of science sky
↳frames; pipeline uses this and not original_skyFrameList
|____snN201004*.fits            # Sky subtracted, prepared raw science
↳frames. Results of gemarith()
|____telCorN201004*.fits         # UNSHIFTED AND SCALED telluric correction
↳for each science cube
|____telFitN201004*.fits         # UNSHIFTED AND SCALED fit to telluric
↳correction for each science cube
|____tfbrsnN201004*.fits        # Results of nftransform()
|____wrgnN20100401S0137.fits     # Final reduced wavelength solution frame.
↳Result of nswavelength()
|____xtfbrsnN201004*.fits        # Extracted one D spectra from each
↳UNCORRECTED science cube. Result of nfextract()

```

In the scienceObjectName/ExtractedOneD/ directory:

```

ExtractedOneD/
|____20100401_obs107/           # Science data and observation, from .fits
↳headers of science frames
| |____xtfbrsnN201004*.fits      # Extracted one D spectra from UNCORRECTED cubes.
↳Results of nfextract()
|____combined20100401_obs107.fits # Median-combined, extracted one D spectra.
↳Result of gemcombine()

```

## nifsMerge

nifsMerge.py is called as the last step of nifsReduce Science to merge data cubes. It produces three cube merging directories: an UNCORRECTED, a telluric corrected, and a telluric corrected AND flux calibrated directory. Here are two examples of the structure:

First, from the test data we have been using (HD141004) the final merged directory structure should look something like:

```
.
|____config.cfg
|____HD141004/
|  |____20100401/
|  |  |____Calibrations_K/
|  |  |____K/
|  |  |____obs107/
|  |____ExtractedOneD/

|  |____Merged_telCorAndFluxCalibrated/    # Merging directory for final telluric_
↪corrected AND flux calibrated data cubes
|  |  |____20100401_obs107/
|  |  |  |____cube_merged.fits
|  |  |  |____factfbrsnN201004*.fits      # Unmodified, final telluric corrected AND_
↪flux calibrated data cubes. Copied from appropriate science observation directory
|  |  |  |____offsets.txt                 # Offsets provided to imcombine(); see_
↪manual for details
|  |  |  |____out.fits
|  |  |  |____transcube*.fits             # Transposed data cubes. Results of_
↪im3dtran()
|  |  |____20100401_obs107_merged.fits    # Final merged cube for obs107

|  |____Merged_telluricCorrected/          # Merging directory for telluric corrected_
↪data cubes
|  |  |____20100401_obs107/
|  |  |  |____actfbrsnN201004*.fits      # Unmodified, final telluric corrected data_
↪cubes. Copied from appropriate science observation directory
|  |  |  |____cube_merged.fits
|  |  |  |____offsets.txt
|  |  |  |____out.fits                   # Offsets provided to imcombine(); see_
↪manual for details
|  |  |  |____transcube*.fits             # Transposed data cubes. Results of_
↪im3dtran()
|  |  |____20100401_obs107_merged.fits    # Final merged cube for obs107

|  |____Merged_uncorrected/               # Merging directory for UNCORRECTED data_
↪cubes
|  |  |____20100401_obs107/
|  |  |  |____ctfbrsnN201004*.fits      # Unmodified, final UNCORRECTED data cubes._
↪Copied from appropriate science observation directory
|  |  |  |____cube_merged.fits
|  |  |  |____offsets.txt                 # Offsets provided to imcombine(); see_
↪manual for details
|  |  |  |____out.fits
|  |  |  |____transcube*.fits             # Transposed data cubes. Results of_
↪im3dtran()
|  |  |____20100401_obs107_merged.fits    # Final merged cube for obs107

|____Nifty.log
```

## CHAPTER 3

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`