nifty Documentation

Release 1.0.0.dev1

Nat Comeau

Jan 26, 2021

Contents

Nifty4Gemini is a full-featured Gemini data reduction framework for Gemini instruments and here we provide the documentation for its associated NIFS pipeline.

CHAPTER 1

Package Documentation

1.1 Introduction

Nifty4Gemini, for now, uses Python 2.7. Please keep this in mind.

1.2 Running from the Command Line

Nifty4Gemini 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
```

1.3 Running from Python

Python Programmers: even though Nifty's API has not been defined yet, you can still run Nifty Pipelines, Steps, Routines and Tasks from a Python interpeter by importing them.

For example, to run a final cube merge from the Python interpreter:

```
# Import relevant modules; you can find those at top of nifsMerge.py
from pyraf import iraf, iraffunctions
# Import relevant local script
import nifty.pipeline.steps.nifsMerge

# Set up iraf
```

(continues on next page)

(continued from previous page)

```
iraf.gemini()
iraf.gemtools()
iraf.gnirs()
iraf.nifs()

# Define routine parameters
scienceDirectoryList = []
cubeType = "uncorrected"
mergeType = "sum"
use_pq_offsets = True
im3dtran = True
over = False

# Start the cube merging
nifty.pipeline.steps.nifsMerge.mergeCubes(scienceDirectoryList, cubeType, mergeType,␣
↪use_pq_offsets, im3dtran, over="")
```

You might be able to figure out what imports you need by checking the tops of the relevant scripts.

1.3.1 Examples of Running from the Command Line

Starting a Data Reduction from the Beginning

Supply the -i flag to start the NIFS pipeline, populating a configuration file interactively:

```
runNifty nifsPipeline -i
```

Supply a config.cfg file to start the NIFS pipeline from a pre-built configuration file:

```
runNifty nifsPipeline config.cfg
```

Supply the -f flat 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
```

Supply the -f flag to do a fully automatic data reduction, using raw data from a local directory (Eg: /Users/ncomeau/data/TUTORIAL):

```
runNifty nifsPipeline -f /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 use a config.cfg file like this:

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

manualMode = False
```

(continues on next page)

(continued from previous page)

```
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = True
calibrationReduction = False
telluricReduction = False
scienceReduction = False
telluricCorrection = False
fluxCalibration = False
merge = False
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = '/Users/nat/data/TUTORIAL'
program = ''
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
```

(continues on next page)

(continued from previous page)

```
mergeStart = 1
mergeStop = 3
mergeType = 'median'
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 use a config.cfg file like this:

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

manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = False
calibrationReduction = True
telluricReduction = False
scienceReduction = False
telluricCorrection = False
fluxCalibration = False
merge = False
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = '/Users/nat/data/TUTORIAL'
program = ''
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True
```

(continues on next page)

(continued from previous page)

```
[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
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 use a config.cfg file like this: *Make sure to populate scienceDirectoryList, telluricDirectoryList and calibrationDirectoryList before running!*

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

manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = False
calibrationReduction = False
telluricReduction = True
scienceReduction = False
telluricCorrection = False
fluxCalibration = False
merge = False
```

(continues on next page)

(continued from previous page)

```
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = '/Users/nat/data/TUTORIAL'
program = ''
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
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 use a config.cfg file like this: *Make sure to populate scienceDirectoryList, telluricDirectoryList and calibrationDirectoryList before running!*

```
# Nifty configuration file.
```

(continues on next page)

(continued from previous page)

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

manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = False
calibrationReduction = False
telluricReduction = False
scienceReduction = True
telluricCorrection = False
fluxCalibration = False
merge = False
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = '/Users/nat/data/TUTORIAL'
program = ''
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
```

(continues on next page)

(continued from previous page)

```
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

nifsTelluric Correction: To only derive and apply a telluric correction use a config.cfg file like this: *Make sure to populate scienceDirectoryList, telluricDirectoryList and calibrationDirectoryList before running!*

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

manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = False
calibrationReduction = False
telluricReduction = False
scienceReduction = False
telluricCorrection = True
fluxCalibration = False
merge = False
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = '/Users/nat/data/TUTORIAL'
program = ''
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4
```

(continues on next page)

(continued from previous page)

```
[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

nifsFluxCalibration: To only do a flux calibration use a config.cfg file like this: *Make sure to populate scienceDirectoryList, telluricDirectoryList and calibrationDirectoryList before running!*

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

manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = False
```

(continues on next page)

(continued from previous page)

```
calibrationReduction = False
telluricReduction = False
scienceReduction = False
telluricCorrection = False
fluxCalibration = True
merge = False
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = '/Users/nat/data/TUTORIAL'
program = ''
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

nifsMerge Cube Merging: To only merge final data cubes use a config.cfg file like this: *Make sure to populate scienceDirectoryList, telluricDirectoryList and calibrationDirectoryList before running!*

Note: You can start and stop at step 1,3, or 5, create your own waveoffsetsGRATING.txt file in the merged_MERGETYPE directory, and then start and stop from step 2, 4, or 6 to specify your own offsets for the final cube merging step. This is useful for non-sidereal targets.

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

manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = False
calibrationReduction = False
telluricReduction = False
scienceReduction = False
telluricCorrection = True
fluxCalibration = False
merge = True
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = '/Users/nat/data/TUTORIAL'
program = ''
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
```

(continues on next page)

(continued from previous page)

```
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

And run the nifsPipeline with:

```
runNifty nifsPipeline config.cfg
```

1.4 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.

1.4.1 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".

1.4.2 An Example Input File

Nifty includes a default configuration file in the runtimeData/ directory. As of v1.0b2, It looks like this:

```
TODO(nat): Updated example
```

1.5 Data Reduction Examples

1.5.1 Observations of Titan; GN-2014A-Q-85

Note: v1.0.0 has some problems with the final cube merging. I believe this is because the last step of cube merging does not take into account different RAs and Decs. I am hoping to implement a fix in a coming update.

Once you raise the telluricTimeThreshold to 7200 seconds and turn off the telluric sky subtraction, this data reduction works very well in full-automatic mode.

Configuration file used:

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

niftyVersion = '1.0.0'
manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = []
telluricDirectoryList = []
calibrationDirectoryList = []

[nifsPipelineConfig]
sort = True
calibrationReduction = True
telluricReduction = True
scienceReduction = True
telluricCorrection = True
fluxCalibration = True
merge = True
telluricCorrectionMethod = 'gnirs'
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''

[sortConfig]
rawPath = ''
program = 'GN-2014A-Q-85'         # This was added
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 7200      # This had to be tweaked

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = False   # This had to be turned off

[scienceReductionConfig]
```

(continues on next page)

(continued from previous page)

```
sciStart = 1
sciStop = 5
scienceSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6

[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

1.5.2 The Massive Black Hole in M87

This program is problematic for nifsSort.py as some individual observations span multiple days, but calibrations are associate by individual dates (for now). To run this reduction, after sorting, you will have to go through by hand and make sure each calibrations directory contains the required calibrations and text files.

Config file used:

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

manualMode = False
over = False
extractionXC = 15.0
extractionYC = 33.0
extractionRadius = 2.5
scienceOneDExtraction = True
scienceDirectoryList = ['/Users/nat/tests/blackHole/NGC4486/20080416/K/obs29', '/
↪Users/nat/tests/blackHole/NGC4486/20080417/K/obs11', '/Users/nat/tests/blackHole/
↪NGC4486/20080421/K/obs11', '/Users/nat/tests/blackHole/NGC4486/20080422/K/obs19', '/
↪Users/nat/tests/blackHole/NGC4486/20080422/K/obs21', '/Users/nat/tests/blackHole/
↪NGC4486/20080423/K/obs21', '/Users/nat/tests/blackHole/NGC4486/20080523/K/obs29']
telluricDirectoryList = ['/Users/nat/tests/blackHole/NGC4486/20080416/K/Tellurics/
↪obs27', '/Users/nat/tests/blackHole/NGC4486/20080416/K/Tellurics/obs13', '/Users/
↪nat/tests/blackHole/NGC4486/20080417/K/Tellurics/obs32', '/Users/nat/tests/
↪blackHole/NGC4486/20080421/K/Tellurics/obs35', '/Users/nat/tests/blackHole/NGC4486/
↪20080422/K/Tellurics/obs17', '/Users/nat/tests/blackHole/NGC4486/20080423/K/
↪Tellurics/obs23', '/Users/nat/tests/blackHole/NGC4486/20080423/K/Tellurics/obs26',
↪'/Users/nat/tests/blackHole/NGC4486/20080523/K/Tellurics/obs30']
```

(continues on next page)

(continued from previous page)

```
calibrationDirectoryList = ['/Users/nat/tests/blackHole/NGC4486/20080416/Calibrations_
↪K', '/Users/nat/tests/blackHole/NGC4486/20080417/Calibrations_K', '/Users/nat/tests/
↪blackHole/NGC4486/20080421/Calibrations_K', '/Users/nat/tests/blackHole/NGC4486/
↪20080422/Calibrations_K', '/Users/nat/tests/blackHole/NGC4486/20080423/Calibrations_
↪K']

[nifsPipelineConfig]
sort = False
calibrationReduction = False
telluricReduction = False
scienceReduction = True
telluricCorrection = False
fluxCalibration = False
fluxCalibrationMethod = 'gnirs'
mergeMethod = ''
merge = True

[sortConfig]
rawPath = ''
program = 'GN-2008A-Q-12'
proprietaryCookie = ''
skyThreshold = 2.0
sortTellurics = True
telluricTimeThreshold = 5400

[calibrationReductionConfig]
baselineCalibrationStart = 1
baselineCalibrationStop = 4

[telluricReductionConfig]
telStart = 1
telStop = 5
telluricSkySubtraction = True

[scienceReductionConfig]
sciStart = 1
sciStop = 5
scienceSkySubtraction = True

[telluricCorrectionConfig]
telluricCorrectionStart = 1
telluricCorrectionStop = 9
hLineMethod = 'vega'
hLineInter = False
continuumInter = False
telluricInter = False
tempInter = False
standardStarSpecTemperature = ''
standardStarMagnitude = ''
standardStarRA = ''
standardStarDec = ''
standardStarBand = ''

[fluxCalbrationConfig]
fluxCalibrationStart = 1
fluxCalibrationStop = 6
```

(continues on next page)

(continued from previous page)

```
[mergeConfig]
mergeStart = 1
mergeStop = 3
mergeType = 'median'
use_pq_offsets = True
im3dtran = True

# Good luck with your Science!
```

1.5.3 Observations of a Moderate Redshift Galaxy

Recipe used: defaultConfig.cfg

Let's reduce NIFS data of a moderate redshift galaxy, located at z ~ 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 defaultConfig.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>
```

1.6 Tutorials

1.6.1 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.

1.6.2 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.

1.6.3 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 tranposed state.

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

1.7 Known Issues

1.7.1 nifsPipeline.py

1.7.2 nifsSort.py

Two things can routinely cause nifsSort to fail.

Object and Sky frame differentiation

Relevant warning:

..code-block:: text

> **WARNING in sort: science " <scienceObservationName>** in " <currentWorkingDirectory> does not have a scienceFrameList.") I am trying to rewrite it with zero point offsets.

Solution:

- Change the skyThreshold parameter (given in arc seconds) and re-run sort.

If the sorting script does not create a skyFrameList in the object or telluric observation directories this means that the offsets between sky frames and object frames were different than expected. A skyFrameList can be manually created and saved in the appropriate directory, or the skyThreshold parameter modified.

Telluric and Science Frame Matching

Relevant warning:

..code-block:: text

> **WARNING in sort: no tellurics data found for science " <scienceImageName>** in " <currentWorkingdirectory>

Solution:

- Raise telluricTimeThreshold to something above 5400 seconds.

By default standard star observations are matched with each science frame if they are within 1.5 hours (5400) seconds in UT start time. Sometimes just a few science frames will be outside that threshold.

If you see the relevant warning after running sort, you will have to raise the telluricTimeThreshold in the config.cfg file if you want to do telluric corrections.

1.7.3 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.**

1.7.4 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.

1.7.5 nifsMerge.py

- When overwrite is turned on, merging the final cubes from multiple directories redundantly repeats.

This is a good quick fix for someone to implement. - *Note: v1.0.0 has some problems with the final cube merging.* I believe this is because the last step of cube merging does not take into account different RAs and Decs. I am hoping to implement a fix in a coming update.

1.7.6 nifsTelluric.py

nifsTelluric.py and nifsFluxCalibrate.py rely on several IRAF tasks finding information in the correct image header. If this implicit header data matching changes in future IRAF implementations these tasks will break. A fix could be to explicitly specify the data header for each task. TODO(nat): Try to do this before you leave.

- iraf.imarith() is having some strange errors. I'm using astropy and numpy instead to do the 2 imarith steps.
- See nifsFluxCalibrate for the iraf.gemini() and iraf.imarith() bug.

1.7.7 nifsFluxCalibrate.py

- **If iraf.gemini() is called in nifsTelluric.py and the two steps are run back to back,**

iraf.imarith() does not work in nifsFluxCalibrate.py. iraf.imarith can't seem to open the result of step 3; it throws this error:

..code-block:: text

    3_BBodyN20100401S0182 is not an image or a number

When I ran nifsTelluric.py and nifsFluxCalibrated.py back to back, using nifsPipeline.py as a wrapper, iraf.imarith() *worked when iraf.gemini() was commented out in nifsTelluric.py*. It crashed with the above error when iraf.gemini() was uncommented.

I've implemented a just-as-good solution using astropy and numpy.

1.7.8 nifsUtils.py

1.7.9 General Issues

- A longstanding bug (see astropy) in astropy has made it difficult to build Nifty4Gemini 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 functionioning 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".

1.8 Maintaining Nifty4Gemini

1.8.1 Documentation

Right now there exists four forms of documentation.

Paper

README.rst

.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

1.8.2 Tests

This is a todo. Currently we do not have automated tests.

1.8.3 Pipeline Structure

See the Nifty4Gemini paper for a high level overview. Nifty4Gemini 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.

Nifty4Gemini 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 tasks.

1.8.4 Updates

To update Nifty4Gemini, do five things:

- *Try to do your development in a new branch or fork, not the master branch of the repository*
- Before uploading, do a few test data reductions.
- Pick an appropriate version number based on semantic versioning; update the setup.py
- Commit all changes to GitHub
- Create a new Github Release
- **'Upload the latest version to PyPi.org < https://packaging.python.org/tutorials/distributing-packages/>'_**

..code-block:: text

    rm -r dist build # Clean up old files python setup.py bdist_wheel # Build python wheels twine upload dist/* # Upload to PyPi.org

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.

1.8.5 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.

Code style was influenced by the Google Python Style Guide.

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.

1.9 Future Work

Throughout the code, ncomeau has placed many TODO notes. These are things that should be reviewed at some point.

Future work:

- Implement more telluric and flux calibration methods.
- Implement instrument signature removal routine.
- Implement cosmic ray removal routine.
- Implement differential atmospheric refraction correction routine.
- Implement full automatic Gemini (on server) reduction routine.
- Implement a "don't save intermediate products" switch
- Object Oriented rewrite; see the JWST calibration pipeline. Pipelines, Steps, Routines and Tasks may implemented better as software objects.
- XDGNIRS integration and NDMAPPER (James EH Turner) integration
- Analysis tools: automatic velocity field? Dispersion?
- Python 3 compatability(if possible)
- Compiling as a self-contained executable
- Full AstroConda integration

1.10 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.

1.10.1 Unreleased

All in-development changes will be tracked here.

- Adding unit tests for each step and integration test for pipeline.

1.10.2 1.0.1 - 2017-10-01

Minor patches and small feature adding release.

- Fixed bug in nifsFluxCalibration.py
- Added ability to use your own waveoffsets+'GRATING'.txt file in nifsMerge.py
- Updated documentation

1.10.3 1.0.0 - 2017-09-12

First production release. Works fairly smoothly once raw data is located and sorted properly.

- Rigorously tested first type of cube merging; not final wave shifting of cubes; with dummy data.
- Finished integrating multiple types of cube merging.

1.10.4 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.

1.10.5 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.

1.10.6 1.0a1 - 2017-08-31

Preliminary Alpha release.

- .whl uploaded to PIP, docs uploaded to

ReadTheDocs and preliminary DOI assigned.

1.11 API

Note: I didn't have time to implement this using Sphinx automodule. Nifty has fairly good docstrings and you can use individual steps, routines and tasks by importing them. This is a todo.

1.12 Example of Nifty File I/O

Note: this is out of data. v1.0b12 compartmentalized the data of each step, so you can safely delete the intermediate products of each step. This is moderately correct up the extraction of 1D spectra.

This is an example of how the Nifty directory tree appears after each step of the date reduction. These directory trees were created using a custom **niftree** 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 **niftree** alias:

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

1.12.1 Example Data Reduction Products:

```
|____.DS_Store
|____config.cfg
|____ExtractedOneD
| |____20130527_obs28
| | |____xtfbrsnN20130527S0264.fits
| | |____xtfbrsnN20130527S0266.fits
| | |____xtfbrsnN20130527S0267.fits
| | |____xtfbrsnN20130527S0269.fits
| | |____xtfbrsnN20130527S0270.fits
| | |____xtfbrsnN20130527S0272.fits
| |____20130530_obs36
| | |____xtfbrsnN20130530S0261.fits
| | |____xtfbrsnN20130530S0263.fits
| |____20130530_obs55
| | |____xtfbrsnN20130530S0254.fits
| | |____xtfbrsnN20130530S0256.fits
| |____20130531_obs36
| | |____xtfbrsnN20130531S0162.fits
| | |____xtfbrsnN20130531S0164.fits
| |____20130621_obs36
| | |____xtfbrsnN20130621S0248.fits
| | |____xtfbrsnN20130621S0250.fits
| | |____xtfbrsnN20130621S0251.fits
| | |____xtfbrsnN20130621S0253.fits
| |____20130622_obs44
| | |____xtfbrsnN20130622S0327.fits
| | |____xtfbrsnN20130622S0329.fits
| | |____xtfbrsnN20130622S0330.fits
| | |____xtfbrsnN20130622S0332.fits
| |____20130624_obs75
| | |____xtfbrsnN20130624S0078.fits
| | |____xtfbrsnN20130624S0080.fits
| | |____xtfbrsnN20130624S0081.fits
| | |____xtfbrsnN20130624S0083.fits
| | |____xtfbrsnN20130624S0084.fits
| | |____xtfbrsnN20130624S0086.fits
| |____20130626_obs83
| | |____xtfbrsnN20130626S0108.fits
| | |____xtfbrsnN20130626S0110.fits
| | |____xtfbrsnN20130626S0111.fits
| | |____xtfbrsnN20130626S0113.fits
| |____combined20130527_obs28.fits
| |____combined20130530_obs36.fits
| |____combined20130530_obs55.fits
| |____combined20130531_obs36.fits
| |____combined20130621_obs36.fits
| |____combined20130622_obs44.fits
| |____combined20130624_obs75.fits
| |____combined20130626_obs83.fits
|____Merged_telCorAndFluxCalibrated
```

(continues on next page)

(continued from previous page)

```
| |____.DS_Store
| |____20130527_obs28_merged.fits
| |____20130530_obs36_merged.fits
| |____20130530_obs55_merged.fits
| |____20130531_obs36_merged.fits
| |____20130621_obs36_merged.fits
| |____20130622_obs44_merged.fits
| |____20130624_obs75_merged.fits
| |____20130626_obs83_merged.fits
| |____temp_mergedH.fits
| |____TOTAL_mergedH.fits
| |____waveoffsetsH.txt
|____Merged_telluricCorrected
| |____20130527_obs28_merged.fits
| |____20130530_obs36_merged.fits
| |____20130530_obs55_merged.fits
| |____20130531_obs36_merged.fits
| |____20130621_obs36_merged.fits
| |____20130622_obs44_merged.fits
| |____20130624_obs75_merged.fits
| |____20130626_obs83_merged.fits
| |____temp_mergedH.fits
| |____TOTAL_mergedH.fits
| |____waveoffsetsH.txt
|____Merged_uncorrected
| |____20130527_obs28_merged.fits
| |____20130530_obs36_merged.fits
| |____20130530_obs55_merged.fits
| |____20130531_obs36_merged.fits
| |____20130621_obs36_merged.fits
| |____20130622_obs44_merged.fits
| |____20130624_obs75_merged.fits
| |____20130626_obs83_merged.fits
| |____temp_mergedH.fits
| |____TOTAL_mergedH.fits
| |____waveoffsetsH.txt
|____Nifty.log

.
|____config.cfg
|____ExtractedOneD/                    # Extracted One D Science Spectra
| |____20130527_obs28/
| |____20130530_obs36/
| |____20130530_obs55/
| |____20130531_obs36/
| |____20130621_obs36/
| |____20130622_obs44/
| |____20130624_obs75/
| |____20130626_obs83/
| |____combined20130527_obs28.fits
| |____combined20130530_obs36.fits
| |____combined20130530_obs55.fits
| |____combined20130531_obs36.fits
| |____combined20130621_obs36.fits
| |____combined20130622_obs44.fits
| |____combined20130624_obs75.fits
| |____combined20130626_obs83.fits
```

(continues on next page)

(continued from previous page)

```
|____Merged_telCorAndFluxCalibrated/    # Telluric corrected AND flux calibrated
| |____TOTAL_mergedH.fits
|____Merged_telluricCorrected/          # Telluric corrected cubes
| |____TOTAL_mergedH.fits
|____Merged_uncorrected                 # Uncorrected cubes
| |____TOTAL_mergedH.fits
|____Nifty.log
```

1.12.2 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
```

(continues on next page)

(continued from previous page)

```
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
| | | | | |____tellist                  # 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/
```

(continues on next page)

(continued from previous page)

```
| | | |____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
```

(continues on next page)

(continued from previous page)

```
.
|____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/
| | | | |____idrgnN20100410S0375_SCI_*_      # Textfiles containing spatial solutions␣
↪for particular slices
| | | | |____idwrgnN20100401S0137_SCI_*_
```

(continues on next page)

(continued from previous page)

```
| | | |____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
```

(continues on next page)

(continued from previous page)

```
| | | |____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.
```

1.12.3 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
| | | | | |____tellist
| | | | | |____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
|____tellist
|____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
```

(continues on next page)

(continued from previous page)

```
|____nN201004*.fits
|____rgnN20100410S0375.fits
|____rsnN201004*.fits               # Flat fielded standard frames; results of␣
↪nsreduce()
|____scienceMatchedTellsList
|____skyFrameList
____snN201004*.fits
|____tellist
|____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
|____tellist
|____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
```

(continues on next page)

(continued from previous page)

```
|____scienceMatchedTellsList
|____skyFrameList
|____snN201004*.fits
|____tellist
|____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
|____tellist
|____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()
```

(continues on next page)

(continued from previous page)

```
|____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
```

1.12.4 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_*_
```

(continues on next page)

(continued from previous page)

```
| |____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/
```

(continues on next page)

(continued from previous page)

```
| |____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
```

(continues on next page)

(continued from previous page)

```
|____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()
| |____idrgnN20100410S0375_SCI_*_       # Text file results of nfsdist()
| |____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 nfsdist()
|____rsnN201004*.fits                   # Flat fielded, sky subtracted and prepared␣
↪science frames. Result of nsreduce()
```

(continues on next page)

(continued from previous page)

```
|____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()
```

1.12.5 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
```

(continues on next page)

(continued from previous page)

```
| | | |____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 2

Indices and tables

- genindex
- modindex
- search