Chapter 1. PACS Spectroscopy
Launch Pad I

1.1. Introduction

Welcome to the PACS data reduction guide (PDRG). We hope you have gotten some good data from PACS and want to get stuck in to working with them. The PDRG explains the running of the data reduction pipelines for PACS spectroscopy in HIPE, and how to interact with the data. This first chapter—the PACS Spectroscopy Launch Pad I—is a “ReadMeFirst” to working with PACS spectroscopy. The questions we will answer are:

1. How do I get and save a PACS spectroscopy observation?
2. How do I understand what type of observation I have? What is the quality report?
3. What are the different types of cubes in my observation?
4. How can I quickly plot/display my spectra?
5. What scripts or cookbooks can help me work with PACS spectroscopy?

The second chapter—the PACS Spectroscopy Launch Pad II—is about the pipeline, and should be read by everyone working with PACS spectroscopy data, whether you want to re-reduce the data or not. Chapter 2 answers the following questions:

1. What are the pipelines available for reducing PACS data (either by me or by the HSC)
2. Do I need to re-run the pipeline (and if so, which one?)
3. Is it anyway useful to re-run the pipeline to improve some of the results?
4. Which are the crucial pipeline tasks?
5. I have a point source: what do I do next?
6. I have an extended source: what do I do next?
7. Where do I go to learn about the errors in my spectra?

The rest of the PDRG is:

- **Chapter 3**: gives more detail on the calibration files used in the pipeline processing, on getting and saving data, on where the pipeline scripts are, and an explanation of the differences between these interactive pipeline scripts. An observation gotten from the HSA will only have been processed through one type of pipeline: this chapter is also useful for understanding whether you should consider running one of the other pipelines for your observation.

- **Chapter 4**: concerns the beginning (Level 0-0.5) and the end (post-processing) parts of the reduction of PACS data, which are almost the same for all the pipeline scripts. Chapter 5: explains the data reduction from Level 0.5 to 2 for chop-nod mode observations. Chapter 6: explains the data reduction from Level 0.5 to 2/2.5 for unchopped mode observations.

- **Chapter 7**: explains in more detail some of the more crucial pipeline tasks: wavelength regridding; flatfielding; using the different cube mosaicking tasks (drizzle, interpolate, project); transient correction for the unchopped mode; background subtraction for the unchopped mode; point-source corrections in the point-source pipeline. The errors in the final PACS spectroscopy products are
also explained here. Whether you run the pipeline or not, read this chapter to understand the effect these crucial tasks can have on the appearance of the final spectra the pipeline produces.

- **Chapter 8**: explains the post-pipeline tasks that are provided for point sources and semi-extended sources, and which can be run after any pipeline script. These tasks must be run for these types of sources, so we also explain how to run them from an observation just gotten from the HSA but not reprocessed through the pipeline.

- **Chapter 9**: explains the post-processing tasks to deal with mapping observations, creating mosaic cubes of various type, the differences between the various types of cubes, and what each one is best used for. Whether you run the pipeline or not, read this chapter to understand why the final cubes of your observation looks the way they do.

- **Chapter 10**: includes scripts and general information about plotting PACS data for diagnostic reasons during the pipeline processing, describes the viewers for inspecting PACS data (spectrally and spatially); and explains how to work with masks.

Additional reading can be found on the HIPE help page, which you can access from the HIPE Help#Help Contents menu. This covers the topics of: HIPE itself, I/O, scripting in HIPE, and using the various data inspection and analysis tools provided in HIPE. We will link you to the necessary bits of this documentation—we do not repeat the information given there: only material that is PACS-specific is in the PDRG. You can also consult the PACS calibration pages on the Herschel Science Centre site, where the Observer's Manual and calibration documentation and information are provided (herschel.esac.esa.int/twiki/bin/view/Public/PacsCalibrationWeb). This is also linked from the PACS section of the HIPE help page. Information on the calibration of PACS data is not covered in the PDRG.

Text written like this refers to the class of a product (or to any product of that class). Different classes have different (java) methods that can be applied to them and different tasks will run (or not) on them. See the [Scripting Guide](#) to learn more about classes. Text written like this refers to the parameters of a task.

### 1.1.1. Terminology

The following definitions will be useful to know:

- **HIPE** Herschel Interactive Processing Environment
- **DAG**: the HIPE Data Analysis Guide (this explains the general HIPE data analysis tools)
- **SG**: the Scripting Guide (a guide to scripting in HIPE)
- **PACS URM** the User's Reference Manual, describes the PACS tasks, their parameters and their function
- **PPE** in *PACS Products Explained* the PACS Products Explained, which is about the products you get from the HSA or which you produce while pipeline processing PACS data
- **HSA, HSC** Herschel Science Archive and Herschel Science Centre
- **AOT** different PACS observing modes were programed with different Astronomical Observing Templates by the proposer
- **Level 0** products are raw and come straight from the satellite
- **Level 0.5** products have been partially reduced, and corrected for instrument effects by tasks for which no interaction is required by the user
- **Level 1** products have been more fully reduced, some pipeline tasks requiring inspection and maybe interaction on the part of the user
• **Level 2** products are fully reduced, including tasks that require the highest level of inspection and interaction on the part of the user. Post-pipeline processing is done on these data.

• **Level 2.5** products can be found for unchopped range observations—which have separate on-source and off-source observations—in the on-source observation. At this level the off-source has been subtracted from the on-source, and so post-pipeline processing can be done on these data.

• **SPG**: standard product generation: the data reduced with a standard pipeline by the HSC. For each major version of HIPE a new SPG is run (this taking a few months of time). The SPG version of any observation is shown in the HSA results tab, in the "Summary" tab of the Observation Viewer, and also in the Meta datum "creator" of the ObservationContext. "SPG 13" is the SPG using the pipeline scripts of HIPE 13.

• **spaxel**: a spatial pixel, the spatial unit of an Integral Field Unit (IFU). Each spaxel contains the spectrum from one unique "pixel" on the sky. The native spaxel size of PACS are 9.4x9.4 arcsec; when you mosaic together cubes the resulting cube’s spaxels are smaller, but they are still called "spaxels".

### 1.2. Getting and saving PACS observations

Herschel data are stored in the **HSA**.

• They are identified with a unique number known as the Observation ID (obsid). You can find the obsid via the HSA.

• They can be downloaded directly into HIPE, or one at a time to disk, or many as a tarball.

• The data you get from the HSA is an **Observation Context**, which is a container for all the science data and all the auxiliary and calibration data that are associated with an observation, and includes the **SPG** products. The entire observations is stored on disk as individual FITS files with an organisation and naming that is not straightforward. The **ObservationContext** you load into HIPE contains links to all these files, and GUIs are provided to navigate through the layers.

There are several ways to **get and save observations from the HSA or disk** via HIPE.

• **Get the data directly from the HSA into HIPE on the command line, and then save to disk**:

```python
obsid = 134........ # enter your own obsid
# To load into HIPE:
myobs = getObservation(obsid, useHsa=True)

# To load into HIPE and at the same time to save to disk
# A: to save to the "MyHsa" directory (HOME/.hcss/MyHsa)
myobs = getObservation(obsid, useHsa=True, save=True)

# B: to save to your "local store" (usually HOME/.hcss/lstore)
myobs = getObservation(obsid, useHsa=True)
saveObservation(myobs)

# C: to save to another disk location entirely, use:
pool1 = "\Volumes/BigDisk/
pooln = "NGC3333"
myobs = getObservation(obsid, useHsa=True)
saveObservation(myobs, poolLocation=pool1, poolName=pooln)
```

See the **DAG sec. 1.4.5** for more information on getObservation (for example, how to log on to the HSA before you can get the data, and more about "MyHSA"), and **Section 3.2**. For full parameters of getObservation, see its **URM** entry.

• **To get the data back from disk into HIPE**:

  * **A and B**: If you saved the data to disk with the default name and location (either [HOME]/.hcss/MyHSA or [HOME]/.hcss/lstore) then you need only specify the obsid:

```python
obsid = 134........ # enter your obsid here
```

---

**PACS Spectroscopy Launch Pad I**

Build 14.0.1408
myobs=getObservation(obsid)

C: If you used saveObservation with a poolName and/or poolLocation specified:

obsid = 134...... # enter your obsid here
pooln = "NGC3333"
myobs=getObservation(obsid, poolLocation=pooln, poolName=pooln)

In Chapter 3 you can find a longer summary of getting and saving. The PACS pipelines use these command-line methods. To learn about the GUI methods, see chap. 1 of the DAG.

1.3. What type of observation do I have?

PACS spectrometer observations were executed following a set of observing templates: the AOTs. Before re-reducing you data you need to know the observing mode. It is also helpful to know the observing mode because known issues and calibration uncertainties can vary between AOTs.

The PACS spectroscopy observing modes included these variation:

- **Chop-nod, unchopped, or wavelength switching**: the difference between these lies in the observing technique used to allow the telescope+astronomical background to be subtracted. The first mode was used for most observations, chopping and nodding with a high frequency between the source and the background; the unchopped mode was used for sources too far from a clear sky background, so the background was observed separately to the on-source location; the third was also for sources far from a clear sky background, and the background was instead sampled from wavelengths offset from the wavelength of the observation.

- **Line or Range spectroscopy**: with a wavelength stretch that encompasses one unresolved line only (Line); or an observer-defined wavelength range (Range) or the full spectral energy distribution range of PACS (SED, which is considered part of Range).

- **Single pointed, undersampled mapping (tiling), Nyquist mapping, or oversampled mapping**: refers to the pointing mode: a single pointing; a large-step raster to cover a large field-of-view (tiling); a smaller-step raster to achieve a Nyquist spatial sampling of the beam; or very a fine-sampling raster to oversample the beam.

For the mapping modes, whether the beam is undersampled or not is important when combining the pointings of the raster into a single, mosaic cube. For undersampled modes, a mapping algorithm is not possible, instead we interpolate between overlapping/nearby spaxels in the raster to create a single cube. For the Nyquist or oversampled modes, we offer two tasks that remap and combine the data from the spatial grid of the raster, to a new grid that covers the entire field-of-view with smaller spaxels.

A more detailed summary for each mode is provided in Section 3.3. Note that all observations always contain data from the blue and the red camera: the observer-specified camera plus a "free" camera. The only cause of data missing in one camera is if the entire spectral range there falls outside of the filter function, or if there was an instrument anomaly during the observing run: in these cases there will be a "qualitySummary" in the ObservationContext.

In the HSA (search results table) it is indicated if the observation is line spectroscopy or range spectroscopy. Once you have downloaded the observation into HIPE you can find all relevant information in this way:

obsSummary(myobs)
In the text printed to the Console of HIPE by this command, look for the section "AOT and instrument configuration", e.g.

<table>
<thead>
<tr>
<th>AOT and instrument configuration:</th>
</tr>
</thead>
<tbody>
<tr>
<td>AOT: PacsLineSpec</td>
</tr>
<tr>
<td>Mode:  Mapping, Chop/Nod</td>
</tr>
<tr>
<td>Bands:  B3A R1 (prime diffraction orders selected)</td>
</tr>
<tr>
<td>Is bright: YES (shortened range mode)</td>
</tr>
<tr>
<td>Raster lines: 5</td>
</tr>
<tr>
<td>Raster columns: 5</td>
</tr>
<tr>
<td>Raster line step: 14.5 (arcseconds)</td>
</tr>
<tr>
<td>Raster point step: 16.0 (arcseconds)</td>
</tr>
<tr>
<td>Chopper:  large throw</td>
</tr>
<tr>
<td>Nod cycles: 1</td>
</tr>
</tbody>
</table>

and there you will find:

- **AOT** (line or range)
- **Mode** (mapping or pointed; and unchopped, wavelength switching, or chop-nod)

For Mapping modes, note down the size of the steps (Raster line or point step) and number of steps (Raster lines or columns); this information will be useful later.

## 1.3.1. What is the quality report?

One of the products held in an ObservationContext, and which you will see when you view that observation with the Observation viewer, is a quality report. This comes in the form of a "quality" or "qualitySummary". They both contain the same information, but the qualitySummary (if present) contains a report created after a manual check of the observation has been done at the HSC. This process takes a long time and so not all observations will have a summary, or some will have ones produced several years ago, and which may no longer be relevant (e.g. the pipelines have improved over the years and some problems of the past are now dealt with).

Click on the +quality/+qualtySummary from within the Data tab of the Observation viewer and the report viewer will open to the right of that tab. The most important things to check are the Quality flags and the comments. The flags and the more technical comments are decoded in a document on the the "known issues" page: herschel.esac.esa.int/twiki/bin/view/Public/DpKnownIssues.

## 1.4. What are the different types of cubes in my observation?

### 1.4.1. The science-quality cubes

The cubes produced at the end of the SPG pipeline are found in Level 2 for most observation, Level 2.5 for the on-source observation of an unchopped range pair. Since there can be, within any observation, several pointings and multiple wavelength ranges, all the related cubes are gathered together in lists, also know as contexts: one context for the red camera and one context for the blue camera. In addition, PACS produces different types of cube depending on the observing mode of the observation, and you will find a separate set of red and blue context for each type of cube. Finally, note that the range of cubes provided has recently changed: anything reduced by SPG 12.x.x will have fewer cubes in the Level 2 that observations reduced by SPG 13.0.

To find these contexts:

- Double-click on your observation in the Variables panel (or right-click and select the Observation Viewer). The viewer will open in the Editor pane of HIPE.

- In the directory-like listing on the left of the Observation viewer (under "Data"), click on the + next to the "level2" (or "level 2.5" if there is one)
The names of the contexts start with **HPS3D** (Herschel-PACS 3-dimension) and then contain the letters indicating the specific type of cube held in therein:

- **HPS3DP[R|B]** are the red and blue projected cubes,
- **PS3DR[R|B]** are the red and blue rebinned cubes,
- **HPS3DD[R|B]** are the red and blue drizzled cubes,
- **HPS3DI[R|B]** are the red and blue interpolated cubes.

(You will also see **HPS3D[R|B]**, but these are not one of the final cubes of the pipeline and you can ignore them for now.) Click on the + next to the **HPS3DXX** to see their individual, numbered, cubes:

![Figure 1.1. Some of the cubes of Level 2: the R rebinned (HPS3DR[R|B]), with one cube per wavelength range (the outside 0 and 1) and per pointing (the inside 0 to 13); and the P rojected (HPS3DP[R|B]), with one cube per wavelength range (0 and 1: the pointings have been combined into a single cube)](image)

A summary of the Level 2/2.5 cubes is given here. See **Section 10.5** to learn more about the native footprint of the PACS integral field unit, and **Section 9.2** to learn for what purpose and for which observing mode each cube has been provided ("native" cubes and mosaic cubes). (To find the observing mode of your observation, see **Section 1.3**.)

- **HPS3DR[R|B]**. The rebinned cubes in this context are of class *PacsRebinnedCube*. There is one cube per wavelength range (2 in the screenshot above) and per pointing (14 for each pointing in the screenshot) as specified in the observing proposal. These cubes have an irregular spatial grid and a non-equidistant wavelength grid (i.e. the bin sizes scale with resolution, which scales with wavelength). The spaxels are 9.4". These cubes are provided in SPG 12 and 13. Every observing mode has these cubes.

For observations of point or slightly extended sources, the correctly-calibrated spectrum must be extracted from these rebinned cubes.

- **HPS3DP[R|B]**. The projected cubes in this context are of class *SpectralSimpleCube* and are result of the pipeline task specProject run on the rebinned cubes. There is one cube per wavelength range specified in the observing proposal (0 and 1 in the screenshot above). In the screenshot above you will see that there is only one cube per wavelength, because the all pointings have been combined. They have a regular spatial grid and also a non-equidistant wavelength grid.
These cubes are the spatially regridded, mosaic versions of the rebinned cubes. They are provided in your HSA ObservationContext for "Mapping" mode observations, with the number of steps and step sizes (see the previous section) that allow for at least a Nyquist number of pointings over the FWHM of the beam, for the red and blue camera independently. The numbers you need to know are:

- **Nyquist mapping**: in the blue, step sizes of up to 16" with a 3x3 raster; in the red step sizes of up to 24" with a 2x2 raster

- **Oversampled mapping**: in the blue, step sizes of up to 3.0" with a 3x3 raster; in the red 4.5" step sizes with a 2x2 raster

These cubes are provided in SPG 12 and 13. The spaxels are 0.5" for pointed observations, and up to 3" for mapping observations (the exact value depending on the mapping and spectral coverage mode).

- **HPS3DD[R|B]**. The **drizzled cubes** in this context are of class SpectralSimpleCube and are the result of the pipeline task drizzle running on the cubes that are the immediate precursor to the rebinned cubes. There is one cube per wavelength range specified in the observing proposal.

These cubes are the spatially regridded, mosaic versions of lower-level cubes (those preceding the rebinned cubes). They have a regular spatial grid and also a non-equidistant wavelength grid. They are provided in your HSA ObservationContext for observations of "Line spec" AOT with a "Mapping" mode, with the number of steps and step sizes (see the previous section) that allow for at least a Nyquist number of pointings over the FWHM of the beam, for the red and blue camera independently. The numbers you need to know are the same as given above.

These cubes are provided in SPG 13 only, and have a spaxel size that depends on the spatial sampling of the raster. You can run drizzle yourself also on "Range spec" AOTs if the range is less than a few microns (Section 7.7), the necessary cubes for this can also be found in Level 2.

- **HPS3DI[R|B]**. The **interpolated cubes** in this context are of class SpectralSimpleCube and are the result of the pipeline task specInterpolate on the rebinned cubes. There is one cube per wavelength range specified in the observing proposal.

These cubes are the spatially regridded, mosaic versions of the rebinned cubes. These are new to SPG 13, and provided in your HSA ObservationContext for spatially undersampled mapping (tiling) and single pointing observations (line- and range-scan), and always have a spaxel size of 4.7". They have a regular spatial grid and also a non-equidistant wavelength grid. Undersampled mapping is any mapping mode for which the steps sizes are larger than the Nyquist values given above, or the number of steps lesser, for each camera independently.

For pointed or undersampled mapping observations of extended sources, PACS recommends that the science measurements are made from the rebinned cubes. This is because it is never possible to accurately recover the source morphology from these observations, since the PACS beam has not been fully sampled: the gaps in the spatial coverage correspond to gaps in the collected signal. The mapping tasks, drizzle and specProject, cannot work correctly on these types of observations. However, it is possible to create a (good) approximation of the morphology of the source over the spectral range using interpolation between the original spatial grid and a new, regular spatial grid, at each wavelength in the cube. **New in HIPE 13 is a task that does this—specInterpolate.** The interpolated cubes are a good approximation of the observed patch of sky for tiling observations and (although somewhat less so) for single pointings, but the best spectra to measure for science will still be those of the rebinned cubes.

Go to the + next to the HPS3DXX. The list of numbers (+0, +1, +2...) are the individual cubes. The tooltip that appears when you hover over a cube in the listing will help you work out what the contents of that cube is (wavelength range, position in raster, type of cube). Click on the numbers to see the associated cube open, either within the Observation Viewer, or to see it in a new window, right-click on the number and chose the Spectrum Explorer (SE: see the DAG chap. 6). You can also drag-and-drop the cubes (to the Variables panel) to take them out of the ObservationContext.
Using the SE you can look at the spectra of your spaxels, perform mathematical operations, extract spectra or sub-cubes, make velocity and flux maps and fit your spectra.

Once you have located the cubes you want you can export them from HIPE by right-clicking on the cube in the Observation viewer or the Variables, and "Send" it to be saved as a FITS file. Note that this will only work for individual cubes, not for the contexts they are contained within.

Note
For some unchopped range observations in the HSA, there is only one pointing for the off-source observation, but a raster for the on-source observation. For such cases, the type of cubes provided by the SPG for the on-source and off-source will be the same (HPS3D[R|B], HPS3DP[R|B] and HPS3DI[R|B]), but the spaxel sizes will be different for the projected (HPS3DP[R|B]) cubes: 0.5'' for the off-source and 1.5'' or 3'' for the on-source.

1.4.2. The standalone browse products

The standalone browse products are created from the Level 2/2.5 cubes produced by SPG 13. Their purpose is primarily to provide a quick download of PACS cubes directly from the HSA. The standalone products are: the interpolated, drizzled, and/or projected cubes (depending on the AOT) but with an equidistant wavelength grid (i.e. each bin in the spectral grid is the same size); and the data of the rebinned cubes as a table. All are provided as FITS files.

These reason for providing cubes with an equidistant wavelength grid is that these cubes then have a full WCS, with equidistant grids in the two spatial and the spectral direction. Cube viewers (e.g. ds9) can deal nicely with these cubes. A table of the data of the rebinned cubes is also provided: these cubes can never have a regular spatial grid, but by providing the spectral data as a table, the user can read them into most other software and deal with them perhaps more easily than in cube format.

These standalone products are also in the ObservationContext you download from the HSA, in Level 2/2.5, and so you will see them with the cubes mentioned in the previous section. They have the same name with an "EQ" added.

• HPS3DEQP[R|B]. The projected cubes in with an equidistant wavelength grid.
• HPS3DEQD[R|B]. The drizzled cubes with an equidistant wavelength grid.
• HPS3DEQI[R|B]. The interpolated cubes with an equidistant wavelength grid.
• HPSTBR[R|B]. The rebinned cubes but in a tabular format rather than a 3d product.

Figure 1.2. Some of the standalone browse products: the equidistant cubes, together with the standard pipeline cubes

Read the PPE in PACS Products Explained for more information on these products.

As with the science-grade products, to see the cubes in the Observation Viewer (from the Data panel) click on the Level 2 (or 2.5), and then on the + next to the particular HPS3D you want to look at.
The cubes themselves are listed there, numbered, and you can open them from there (double-click), drag-and-drop to the Variables panel, and save as a FITS file either from the Variables or from the Observation viewer directly (right-click and chose to "Send" it to be saved as a FITS file); note that saving as FITS will only work for individual cubes, not for the contexts they are contained within.

1.5. I just want to look at my cubes!

OK, so you only want to look at your Level 2 (or 2.5) cubes to see what they look like. How can you do this?

First, which Level 2 cubes to look at? As explained previously (Section 1.4) there are several types of cubes: rebinned, projected, drizzled, interpolated, and for each you get blue and red band spectral ranges, and for some you also get an equidistant (standalone browse product) version. Not all observations have all these cubes, it depends on the AOT, but you should always have a choice of the re-binned, two of the others, and one with an equidistant wavelength grid.

If you have a Level 2.5 in your ObservationContext—which you will get with SPG data processed with HIPE 11 and higher—then you are looking at the on-source observation of an unchopped range scan observation-set. This Level contains cubes for which the background has been subtracted.

To get the cube you want, first identify which "HPS3DX[R|B]" (Herschel PACS Spectroscopy cube [type] red|blue context) you want from the Level 2/2.5, and then extract the cube therefrom. This can be done from within the Observation viewer, Data tab, by clicking on the + next to the "HPS3DXX"; looking at the tooltips as you hover over the cubes contained therein (these cubes are listed as +0, +1..., these being "slice" numbers) to find out their wavelength coverage; and then dragging and dropping one of those cubes to the Variables panel of HIPE. You can do the final drag-and-drop part on the command line, e.g.

```python
# To get the first level 2 blue drizzled cube
cubes = obs.refs["level2"].product.refs["HPS3DDB"].product.refs[0].product
# To get the second level 2.5 red projected cube
cubes = obs.refs["level2_5"].product.refs["HPS3DPR"].product.refs[1].product
```

and so-forth. To know which "HPS3DXX" to get:

- for pointed observations, open the rebinned and/or interpolated cubes: HPS3DR[R|B], HPS3DI[R|B]
- for fine mapping observations of line scans, open the drizzled cubes: HPS3DD[R|B]
- for fine mapping observations of range scans, open the projected cubes: HPS3DP[R|B]
- for coarse mapping (tiling) observations, look at the interpolated cubes: HPS3DI[R|B]

But, to be honest, if you just want a quick look at the spectra to see what you have and what the quality of the data are, you can look at any cube. There is no difference in the calibration or spectral shape, very little difference in the SNR, and differences in flux levels in each spaxel are mostly due to the different sizes of the spaxels.

1.5.1. Quick-look cube tools

There are a number of GUIs that can be used to inspect PACS Level 2/2.5 cubes. These are explained in the DAG chaps 6 and 7. When you have a cube highlighted in the Variables pane of HIPE (or in the directory listing in the Data panel of the Observation viewer) you can call up these tasks via the right-click menu. Note that all of these GUIs work on the individual cubes, not on the context they are contained within (see Section 1.4), so you need to go down past the HPS3DXX level in the Level 2 layer of the ObservationContext, to the +0, +1...

- To scroll through 2D wavelength slices of your cubes you can use the Standard Cube Viewer.
• ★ The SpectrumExplorer (see the DAG chap. 6). This is a spectral/spatial visualisation tool for spectra and cubes. It allows for an inspection and comparison of spectra from individual spaxels or from separate cubes, and it gives access to various mathematical tasks via its Spectral Toolbox menu (see chap. 6.4).

• ★ The Cube ToolBox, which you also access via the SpectrumExplorer: see the DAG (chap 6.4, chap 6.5). Together with the Spectrum Toolbox, this allows you to inspect a cube spatially and spectrally at the same time. It also has analyses tasks#you can make line flux maps, velocity maps, and extract out spectral and spatial regions.

• ★ The Spectrum Fitter GUI, which you also access via the SpectrumExplorer: see the DAG chap. 7. This GUI allows you to fit the spectra of your cubes with a variety of models. (For cubes it should be accessed via the Toolbox menu of the Spectrum Explorer, not directly from the HIPE Tasks panel.)

1.5.2. Create quick-look images or spectra from the cubes

A few quick inspection tasks so you can get a feel for your cube data:

• Extract the spectrum of a single spaxel from a pointed observation/rebinned cube with the task extractSpaxelSpectrum, which will only work a rebinned cube (HSP3D[R|B]).

```python
slicedRebinnedCube = obs.refs["level25"].product.refs["HPS3DRR"].product
spaxelX, spaxelY = 2, 2
slice = 0
spectrum = extractSpaxelSpectrum(slicedRebinnedCube, slice=slice, 
   spaxelX=spaxelX, spaxelY=spaxelY)
```

See above to know how to identify the cube "slice" you want to extract from the "HPS3DXXX". To know which spaxel you want to extract, open the cube with the Standard Cube Viewer, and the spaxelX and Y coordinates of the spaxel under the mouse can be found at the bottom-left of the viewer.

• Extract the spectrum of a single spaxel from a mapping observation/any cube with the cube GUIs provided in HIPE. See above to know how to identify the cube "slice" you want from the "HPS3DXX". Extract the cube, e.g. for the second cube

```python
cube = obs.refs["level2"].product.refs["HPS3DPR"].product.refs[1].product
```

Open the cube in the Spectrum Explorer (right-click menu on "cube" in the Variables panel of HIPE), and from there select the Cube Toolbox (the icon at the top of the "SE"), which will open in the top part of the SE.

The Cube Toolbox tasks are located in the drop-down menu to the right of the plot panel: to extract a single spectrum use "extractRegionSpectrum", where you can select out a single spaxel with a click on the cube image. See the DAG (chap 6.7.3) to learn more about using the SE and the Cube Toolbox.

• Extract the summed or average spectrum of a region: using the Cube Toolbox mentioned above, you can also select a rectangular or circular region using the task extractRegionSpectrum: see the DAG (chap 6.7.5). This task will work on most of the Level 2/2.5 cubes: HSP3D[R|D|I|P][R|B]

• Extract an image of a single wavelength point is done the most rapidly with the Standard Cube Viewer, on any cube, right-click menu "Extract current layer".

• Extract a wavelength-integrated image: following the steps for "Extract the spectrum of a single spaxel" above to select the correct cube and open the Cube Toolbox, from there select the task "IntegrateSpectralMap". See DAG (chap 6.7.10.1) to learn how this works.
• **Plot the spectrum of a spaxel together with the RMS estimate**: using the pipeline helper task "plotCubesStddev". See Section 7.6.1 for a longer explanation of this task. The product this task works on is HPS3DR[R|B], i.e. the set of rebinned cubes.

## 1.6. Useful scripts and cookbooks for working with PACS data

There are useful scripts provided for working with PACS spectroscopy that can be obtained via the HIPE Scripts menu.

• **Fitting PACS cubes**: three scripts are provided for fitting a spectral line in PACS cubes and making images from the fitting results, e.g. integrated flux and velocity. These scripts are:

1. **Spectroscopy: Fitting mapping observations (mosaic cubes)**. For the mosaic cubes of mapping observations, and starting from the interpolated, drizzled, or projected cubes (HPS3D[I,D,P][R|B] in the ObservationContext Level 2/2.5).

2. **Spectroscopy: Fitting mapping observations (pre-mosaic cubes)**. Also for mapping observations but this time starting from the rebinned cubes (HPS3DR[R|B] in an ObservationContext). The difference with the previous script is that the fitting is done on these individual cubes of the raster and then the mosaicking is done on the fitting result images, i.e. creating 2d mosaics (images) rather than 3d mosaics (cubes).

3. **Spectroscopy: Fitting single pointing cubes**. For pointed observations, creating fitting images is more qualitative than quantitative, but nonetheless is useful for visualising the results for extended sources observed as a single pointing. The script starts with the interpolated cubes (HPS3DI[R|B] in an ObservationContext).

• **Point sources**: for point sources it is necessary to use the tasks provided to produce a correctly-calibrated spectrum of the point source from the rebinned cubes (HPS3DR[R|B] in the ObservationContext). The two scripts provided are for point sources that are located in the central spaxel of the cube, and for those located in a different spaxel: **Spectroscopy: Point source loss correction (central spaxel)** and **Spectroscopy: Point source loss correction (any spaxel)**.

• **Unchopped range observations**: for this mode the on-source and off-source observation are two separate observations, which need to be reduced separately and then subtracted. The Pipeline menu of HIPE includes a script to do this in a largely automatic manner: the script **Spectroscopy: Off-subtraction and post-processing in unchopped range spectroscopy** is provided for those who wish to do this in a more manual, controlled way.

• **Spectroscopy: Combine PACS and SPIRE spectra**: is a script that is aimed at observations of point sources, where you wish to combine the spectrum of these two instruments into a single spectrum. Note: this is not a mathematical combination, the spectra are simply stored in a single product, for ease of viewing and transporting.

• **Spectroscopy: Re-create the standalone browse products**: this script shows you how to re-create the standalone browse products, which are located in the ObservationContexts of SPG 13 and can be obtained from the HSA directly. These products are easier to read into other software, but they may not be suitable for doing science on: see the PPE in PACS Products Explained for more information.

• **Spectroscopy: Post-processing for extended sources**: using the tasks provided in the pipeline scripts, you are shown how to create the various types of mosaic cubes but starting from an ObservationContext gotten from the HSA.

• **Spectroscopy: Convolution for spectral images**: this script shows you how to take two spectral images (e.g. as created in one of the three useful fitting scripts) and convolve the shorter wavelength image to the beam of the longer wavelength image: the images can then be directly compared to each other.
We are in the process of writing Cookbooks to explain some of the more difficult aspects of working with PACS spectroscopy, and to take the reader through some of the more common work-flows, are in the process of being written. These can be found on the PACS "documentation" page on the HSC web-site: herschel.esac.esa.int/twiki/bin/view/Public/PacsCalibrationWeb.
Chapter 2. PACS Spectroscopy
Launch Pad II

2.1. Introduction

This chapter should be read by those wondering whether it is necessary to run the pipeline on their data themselves. For line scan observations of reasonably bright sources, it is unlikely that you will need to re-run the pipeline, but for all other modes it may improve your results if you do. This chapter should answer the following questions:

1. Where and what are the PACS spectroscopy pipelines?
2. For what observations do I need to re-pipeline the data?
3. For what observations will it be useful to re-process the data, to try to achieve a better result?
4. Which are the crucial pipeline tasks that have the greatest effect on the resulting spectra, and for what observations should I consider checking the results of these tasks?
5. I have a point source: what do I do next?
6. I have an extended source: what do I do next?
7. Where do I go to learn about the errors in my spectra?

2.2. The PACS pipelines

2.2.1. The SPG scripts: the automatic pipeline run by the HSC

The SPG scripts are those run at the HSC in automatic mode. There is one type of pipeline run for unchopped line, range, for chopped line, range, and for wavelength switching. The SPG for a HIPE user release is run after the user release has been completed, but takes a few months to run and hence there is a lag between a release of HIPE and the availability of those SPG products in the HSA: you could be reading this PDRG in a HIPE version 13, but looking at an observation gotten from the HSA processed with HIPE Track 12.1 ("SPG v12.1"). The pipeline scripts that the SPG uses are provided in the HIPE pipelines menu (see the next section), in the submenu "SPG" scripts, but it is not recommended that you run these yourself (use the interactive ones instead).

To know which SPG your downloaded observation was processed with, look at the Summry tab for the observation in the Observation viewer, or, for an observation called "obs",

```
print obs.meta."creator"
```

2.2.2. The interactive pipeline scripts

The pipeline menu in HIPE is split into five: chop-nod line, chop-nod range, unchopped line, un-chopped range, wavelength switching (an old mode, which uses the unchopped line pipeline script). Inside each of these are a choice of pipeline scripts.

For the **chop-nod modes** there are three pipeline flavours:

1. Using the calibration block to flux calibrate the data.
2. Using the telescope background spectrum to flux calibrate the data.
3. And using the telescope background to flux calibrate, and using knowledge of the beam to correct the pointing jitter, for bright point sources only.

In addition, we also offer one or two "helper" pipeline scripts, which do work extra to the pipeline.

The differences between the "calibration block" and the "telescope background" pipelines are (i) historical (the calblock method is the oldest) and (ii) approach in flux calibrating the data and doing the background subtraction. For most observations, both methods are valid, they are simply different approaches to the same problem. The first script is generally better for brighter sources and is necessary if you want to calibrate spectral lines longwards of 190µm, the second script is generally better for fainter sources and observations of long duration, but otherwise both methods are applicable to all sources.

For the unchopped modes in HIPE 13 we now provide two pipeline scripts:

1. The original script, which uses the "calibration block" method, and includes a transient correction for the line scan AOTs only.

2. A new script, which also uses the "calibration block" method, and includes an improved transient correction for line and range scan AOTs; this script has not yet been tested on a large number of PACS observations, so you should pay close attention when running it.

3. The final script in the menu is to reduce both the on-source and off-source observations by calling on the desired pipeline script (1 or 2 above) and then subtracting the background (the off-source spectra). There is also a PACS useful script offered via the HIPE Scripts menu do to this subtraction (Scripts#PACS Useful scripts#Spectroscopy: Off-subtraction and post-processing in unchopped range spectroscopy).

How many wavelength ranges are included in your observation, and whether you have a single pointing or a mapping observation does not matter: all pipelines handle all these cases.

To learn about these pipeline scripts, and more about the "helper scripts": see Section 3.4.2.

To access the scripts: go to the HIPE menu Pipelines#PACS#Spectrometer. The scripts assume:

- The data are already on disk or you can get them from the HSA using getObservation (so you must know the "obsid").

- You have the calibration files on disk; normally you will use the latest update (updates are searched for automatically when you start HIPE), but you can run with any calibration tree version: see Section 3.6.3 to know how to change the version of the calibration tree you are using.

- You do the red and the blue camera separately.

- For unchopped range scans the off-source observations are a separate obsid to the on-source observations, and so the on-source and off-source must be reduced separately, and then subtracted from each other.

- Different wavelength ranges and different pointings within a single observation are smoothly dealt with within the pipelines.

To run the scripts,

- Read the instructions at the top, and at least skim-read the entire script before running it.

- It is highly recommended you run line by line (at least the first time).

- To edit and save the script, you need to save the script to a new, personalised location: otherwise you are changing the script that comes with your HIPE installation. However, as these scripts evolve with time, do not blindly continue to use that pipeline script for future processing: always check against the latest release of HIPE.

As you run the scripts.
• Plotting and printing tasks are included, with which you can inspect the data layout or the spectra themselves.

• You will be offered various ways to save the intermediate data products to a pool on disk. Saving cubes or spectra as FITS files is only possible towards the end of the pipeline, when single FITS-able products are created.

### 2.3. For what observations must I re-pipeline the data?

There are some types of observations for which it is necessary or strongly recommended to re-pipeline the data.

1. **Spectral flatfielding for range scan AOTs**: this is not run in SPG 13 and it is recommended that you re-pipeline the data to include this task. Doing flatfielding for the range scan observations will result in a better SNR. For the longer ranges, especially, it will remove the "fringing"-like pattern that the SPG cubes often have. It will also improve the appearance of sharp rises or drops in the data.

   The flatfielding task runs on data from Level 0.5 in all the chop-nod pipeline scripts and in the original unchopped pipeline scripts (called "LineScan" or "Single obs"). For the new unchopped pipeline scripts (called "...with transients correction") the flatfielding is part of the transients correction and you can also start with Level 0.5 (if working from an SPG 12 or 13 observation).

2. **Transients correction for unchopped range scan AOTs**: this is not run in SPG 13 and it is recommended that you re-pipeline the data to include this task. Transients (e.g. the tails following cosmic rays) affect the response of the instrument, and so affect the flux levels of the spectra over short and long time-scales; bad transients will have a negative influence on the final spectra. A new pipeline script has been provided to do this, and you need to begin from Level 0.5 (if working from an SPG 12 or 13 observation).

3. **If you are interested in spectral lines redder than 190µm**, it will be necessary to run a different pipeline script to that used by SPG 13. The script you need is the one that uses the calibration block to flux calibrate the data, and is called "Single obs" for range scan AOTs and "LineScan" for line scan AOTs; within this you need to use a particular version of the "RSRF" calibration file. See Section 5.3 (chop-nod) or Section 6.4 (unchopped) for more explanation. You will need to re-reduce your data from Level 0.5 (if working from an SPG 12 or 13 observation).

4. **Updates to the calibrations.** Any changes to the calibration between the time the data you have were processed and the current status will require running some or all of the pipeline. You can consult the "What's New" pages to find out what is new in each track (e.g. herschel.esac.esa.int/twiki/bin/view/Public/HipeWhatsNew13x for Track 13), and to learn how to install calibration updates go to Section 3.6. *We note here that there have been no significant spectroscopy calibration updates since SPG 12* and so if you have an observation gotten recently from the HSA you do not need to re-pipeline your data for calibration updates.

### 2.4. For what observations is it useful to re-process the data with a different pipeline?

For certain types of observations it is useful to try out a different pipeline script to that run by the SPG: to see if you can get a better result, to check for contamination; or to try to improve the results of some of the crucial pipeline tasks.

1. **Bright point sources**: with continuum flux levels exceeding a few 10s Jy, or those with very bright spectral lines.

   A special pipeline script for bright point sources—but only where the source is located in the central spaxel—is provided for chop-nod mode observations. This end result of this script is a calibrated
spectrum of the point source, and this spectrum should be cleaner than the standard, point-source calibrated spectrum that you can obtain yourself from the cubes created by the SPG.

This pipeline script is provided in the "Chopped line scan & short range scan" and the "Chopped large range scan SEDs" menus as "Point Source Background Normalization". You can start the pipeline from Level 0.5 if working from an SPG 12 (or 13) observation. The pipeline is explained in Section 5.2.

2. **Check for contamination in the off-source pointings** for chop-nod mode observations using the script "Split on-off" provided in their Line and Range scan menus. If the continuum level in the off-cubes is higher than in the on cubes, or spectral lines are visible, then you probably have contamination. Note, however, that small levels of contamination cannot be detected in this way.

Another way to identify possible off-source contamination for chop-nod mode AOTs is to run the two main pipeline scripts and compare the results. One pipeline method uses the calibration block to flux calibrate and the other uses the telescope background, and they also treat the background subtraction differently. The "calibration block" pipeline is the one "lineScan" for line scans and "Single obs" for range scans, the "telescope background" pipeline is called "Background normalization" (this is run by the SPG). After running the various scripts, compare the rebinned cube spectra (e.g. using the Spectrum Explorer): if the integrated line fluxes are different in the two results, you should check carefully for contamination, and you should probably favour the result from the calibration block pipeline (while noting that this will not "get rid" of the contamination, but it will at least not exaggerate its effect).

For unchopped range scan observations, checking the off-source data is straightforward: compare (e.g. using the Spectrum Explorer) the Level 2 rebinned cubes (e.g. HPS3DR[R|B]) from the off-source observation to the same cubes for the on-source observation. For unchopped line scans, doing this comparison is also simple but requires a few additional lines of code as the appropriate products are not immediately available in the ObservationContext, but need to created. This is explained in Section 10.8.

3. **Transients correction for unchopped mode data**. A new set of transient correction tasks have been incorporated into a new script for the range and for the line scan AOTs: presented as "...with transient correction" in the Line and Range pipeline menus. These pipelines should produce cleaner spectra than those produced by the SPG, and hence are worth testing out. However, note that these scripts are still in a Beta version, they have not yet been tested on a large number of observations.

### 2.5. Which are the crucial pipeline tasks that I should consider checking and re-running?

1. **Spectral flatfielding**. The flatfielding can be a crucial task for improving the SNR of the final spectra. The flatfielding operates in the spectral domain. For each spaxel of a cube there are several discrete spectra that need to be averaged to create the final single spectrum. If some of the discrete spectra are discrepant in signal level (e.g. due to a transient), the flatfielding will correct this (moving those which are too high down, and those which are too low up, while maintaining the average), and so improving the SNR of the subsequently-averaged spectrum.

   *Flatfielding for line scans usually works well, however for very faint (continuum of only a few Jy) and very bright (many 10s to 100 Jy) sources it is worth checking the results by performing the flatfielding yourself, with the "verbose" pipeline script parameter to set True (to produce intermediate plots). The flatfielding has a multiplicative effect and so for bright spectra, the effect is relatively higher than for intermediate flux spectra. For faint spectra, the correction is very difficult to compute when the continuum is near 0, and hence it is also worth checking that the results are reasonable—if the flatfielded spectra have the same SNR as those before, it is even not worth doing a flatfielding at all. To do the flatfielding, you need to run the pipeline script from Level 0.5 and if you are happy with the results using the default parameters, there is no need to continue after the flatfielding task.*
Flatfielding for range scans is not applied in the pipeline scripts (as mentioned previously) and so you do need to re-run the pipeline from Level 0.5. Again, pay closer attention to the very faint and the very bright sources.

Flatfielding for range scans but with a range of less than a few microns and with a fairly simply continuum shape: we recommend you consider using the flatfielding task from the line scan pipeline script, as it can produce better results: for range scans a polynomial fitting is done but for line scans only a median correction is computed, and for short ranges the fitting order used is probably far too high. Again, pay closer attention to the very faint and the very bright sources.

For sources with broad spectral lines or absorption lines, you should also check the flatfielding. The task needs to identify the continuum to work, and it does this by identifying lines and excluding them. However, broad lines and absorption lines are not well detected by the task.

The flatfielding steps in the pipeline scripts (including how to use the line scan task for range scans) are explained in Section 5.2.6 and Section 5.2.7 (chop-nod line, range), Section 6.2.7 and Section 6.2.8 (unchopped line, range). How to compare different flatfielding attempts, and some more things to pay attention to is explained in Section 7.3.

2. Wavelength grid. The wavelength grid used in the SPG pipeline (which is also the default in the pipeline scripts) has been chosen to give the best-looking spectra for most observations. The wavelength grid of the final cubes is created from the individual grids that are present in each of the 16 pixels that feed each spaxel, each of which is slightly offset from the others. The pipeline regularises this collection of grids into a single grid, common to all spaxels and all cubes of the same wavelength setting. This regularised grid is created by the task "wavelengthGrid" with the aid of two parameters—oversample and upsample—which determine the spectral sampling (“what fraction of the resolution should the bin sizes be?”) and which data-points are sampled for each bin (“how many bins do you shift forward when calculating the new grid?”). It is important to note that if upsample is #1, the signal in the bins is correlated, and this means the noise is also correlated and measurements of the noise will give values too low. If this is a problem you will need to re-run the pipeline from Level 0.5 until the end, and chose the parameters of the wavelength grid differently. The interplay and effect of choosing different values of oversample and upsample is explained in more detail in Section 7.2.

Note that the wavelength grid created by the task wavelengthGrid is regular, but it is not equally dispersed: the dispersion is a function of resolution, which is a function of the wavelength: in this way the spectra are at least Nyquist sampling the spectral resolution at every wavelength in the spectrum.

3. For the unchopped modes (line and range), for those with more than one off-source pointing, the background subtraction is crucial. If the background level changes—a not uncommon occurrence because of the effect of transients—then the background subtraction will suffer. The task that does this, "specSubtractOffPosition", offers three algorithms for computing the background to subtract, and it is recommended you experiment with these. For this you need to re-run the pipeline from Level 0.5 until the end, and chose the parameters of the wavelength grid differently. The background subtraction is explained in Chapter 6, and some examples of background subtraction are included in Section 7.4.

2.6. What to do with point sources (and semi-point sources)?

If your scientific goal is to obtain one spectrum of a point source in the observation, you stop the pipeline at the creation of the final rebinned cubes (called "slicedFinalCubes" or "slicedDiffCubes", or HPS3DR[R|B] in Level 2/2.5 of the ObservationContext). Then you have to extract your point source spectrum using one of the tasks provided to do this, since you have to apply the point source flux correction. These tasks are explained in Secs Section 8.3 and Section 8.4 and their use in the pipeline can be found in Section 4.3.
The post-processing tasks for point sources are part of the pipeline scripts, and hence are documented using the terminology and product names of the pipeline (Section 4.3). However, if you want to try some of these tasks on your SPG products, as just gotten from the HSA, we have created two useful scripts that show how to run these tasks: on a pipeline product or on any observation gotten from the HSA: Scripts#PACS useful scripts#Spectroscopy: Point source loss correction (any spaxel) and Spectroscopy: Point source loss correction (central spaxel).

A few tips:

1. For bright point sources with continuum levels exceeding 10 Jy, we also offer a 'Point Source Background Normalisation' interactive pipeline script for chop-nod observations (explained together with all chop-nod pipelines in Section 5.2) which is not part of the SPG. The resulting spectrum should be smoother than that gotten from the SPG.

2. For very faint point sources, where there is more noise than signal in the spaxels surrounding the central one, use the output "c1" from the task extractCentralSpectrum (Section 8.4). This task as used in the pipeline scripts is explained in Section 4.3.

3. For point sources with an uncertain continuum level: any unchopped mode dataset or observations calibrated especially with the RSRF for lines longwards of 190µm: use the result "c1" (fainter sources) or "c9" (brighter sources) produced by extractCentralSpectrum, never use "c129" (see Section 8.4). This task as used in the pipeline scripts is explained in Section 4.3.

4. For point sources processed with the point source background normalisation pipeline: do not use the result "c1" produced by extractCentralSpectrum (see Section 4.3).

5. For point sources where you have SPIRE data, or several PACS bands, and you want to push the extracted point source spectra together into a single product, you can use a useful script Scripts#PACS useful scripts#Spectroscopy: combine PACS and SPIRE spectra. This script is explained in Section 8.2.

We also offer a task to extract the spectrum of slightly extended sources: see Section 8.5 for more information.

For semi-extended sources, i.e. those that are still entirely contained within the field-of-view of a single pointing (about 47"x47") and for which you have a good idea of the source morphology, and which is centred on the central spaxel, there is a task to extract and calibrate its spectrum. See Section 8.5.

2.7. What to do with extended sources

For extended sources observed with a single pointing, a tiling, or a full mapping raster, there is a choice of several different types of cubes that can be created. This depends on the AOT of the observation, on what type of mapping and whether it is a line scan or a range scan: see Section 1.3 for advice on how to know this, and for see Section 3.3 for more information.

If you want to create cubes that are not provided by the ObservationContext you got from the HSA, but do not wish to run a pipeline script, you can follow the instructions in two new PACS Useful scripts: Scripts#PACS useful scripts#Spectroscopy: Post-processing for extended sources, and Spectroscopy: Re-create the standalone browse products. It is also a good idea to try the tasks in this script even if you are happy with the type of cube provided, because the ideal spaxel size to chose for mapping/tiling observations when creating mosaic cubes depends to some degree on the source structure and the spatial sampling. The values used by the SPG are good, but may not be the best for your observation. The useful script can be used to try different spaxel sizes and compare results.

For pointed observations, the final recommended cube is the "rebinned cube" (HPS3DR[R|B]), but interpolated cubes can be useful for a qualitative view on your source.

A few tips:
1. Drizzle will only produce sensible results for spectral regions of 1 or 2 microns, and it also takes a long time to run.

2. specInterpolate is not recommended for Nyquist or oversampled maps. The interpolation done by this task is between 3 points around the requested spatial grid, and for specInterpolate it is not recommended to choose spaxels too small (3" should be a lower limit). In highly-sampled maps the offsets between separate pointings is such that the ideal spaxel size

3. For specInterpolate the spaxels chosen should not be too small, as the interpolation can create very artificial maps. 3" should be small enough.

4. For single pointings, bear in mind always that the beam is very much undersampled: the spaxels are not small enough to fit at least 2 or 3 of them within the FWHM of the beam. This means that there are gaps in the spatial coverage, which will result in a loss of information—flux—such that the true source morphology can never be fully reconstructed. and there will always be some flux loss between spaxels. How important this is depends on the size and morphology of your source. It is recommended that you compare fluxes to those obtained from photometry.

Once you know what cube you want to work with, and where to get it from—Level 2.5 for unchopped range observations, Level 2 for all others—you can then work with your cubes with various tools in HIPE. Some useful scripts and cookbooks have been written to introduce you to the tools that HIPE provides for inspecting and fitting cubes: see Section 1.6.

2.8. Where can I learn about the errors in my spectra?

A detailed explanation of how error estimates are provided for PACS spectroscopy is given in Section 7.6. You are encouraged to read this section carefully.

Of the Level 2/2.5 cubes produced by the pipeline, the following have an error array:

- Rebinned cubes (HPS3DR|B]): an stddev array, which is a calculation of the scatter of the data-points that went in to creating each data-point (of each wavelength, of each spaxel) of these cubes.

- Drizzled and projected cubes (HPS3D|D,PIR|B]: an error array, which is essentially a propagation of the "stddev" array of the rebinned cubes.

For point sources extracted from rebinned cubes using the point source tasks, you will find the following:

- extractCentralSpectrum: the outputs all have a weights array, which is the propagated stddev of the rebinned cube but inverted: 1/stddev**2.

- extractSpaxeSpectrum, pointSourceLossCorrection: the output of these two tasks has no weights or errors

We stress that all of these "errors" are based on the scatter in the data-points of the Level 1 PacsCubes: these data-points are averaged together (per spaxel) along a wavelength grid to create the rebinned cubes. The standard deviation in these data-points, i.e. the data scatter, is the basis for all subsequent errors. That means that the spectral rebinning that is done along this wavelength grid is very important to the values of the "errors" that all subsequent products contain: you should therefore also read Section 7.2 to learn more about how this regridding affects the errors.

Calibration uncertainties are not included in these error values.