What's New in User Release 2.0

Herschel Data Processing

1. Summary

- **Documentation**

  - New manual: *Read Me First*.
  - New manual: *Quick Start Guide*.
  - New manual: *HIPE Owner's Guide*.
  - The *HowTo Documents* have been renamed *Data Analysis Guide* and now include material previously found in the *Advanced User's Manual*.
  - The *Advanced User's Manual* has been renamed *Scripting and Data Mining*, and its contents have been reorganised.
  - New documentation categories.
  - URM categorised view now includes descriptions.
  - Developer documentation for modules now reachable from Help system TOC.
  - Updates to instrument guides.

  See Section 2 for more information.

- **HIPE**

  Herschel Interactive Processing Environment (HIPE) new functionality added for user release 2.0 with respect to user release 1.2:

  - Tip of the day
  - Enhancements in Jython and text editors
  - Preferences links
  - Opened editors are re-opened the next session
  - HIPE title shows version and selected editor
  - Perspectives can be reset to their default layout
  - Source code of Java tasks
  - Multiple selection in the navigator view
  - Non-blocking opening of FITS files
  - Displaying meta data with different formats
  - Refresh of viewers when data changes
  - More log messages
  - Run and stop buttons always present

  - Quick clean of variables
  - Bugfixes

  See Section 3 for more information.

- **TablePlotter and OverPlotter**

  In this version, we rename the OverPlotter plot as "OPL" and the TablePlotter plot as "TPL(<dataset name>)". When switching to any column, the coordinate values can be seen by moving the cross icon on the plot. The axis is adjusted properly when in overlay mode in TablePlotter. The "Preference" button is changed to "Layer Props" where the layer properties can be set. We also move the zoom/pan factors to HIPE->Edit->Preferences where the factors can be set and saved. The layer "Appearances" is under construction. We also introduced a new feature which is to zoom in/out the plot using the mouse middle button. The zoom is centered at the spot where the mouse points.

  - HCSS-6235 was implemented
  - HCSS-6236 was implemented
  - HCSS-6238 was implemented
  - HCSS-6795 was implemented
  - HCSS-7593 was implemented
  - HCSS-8430 was implemented
  - HCSS-8356 was implemented
  - HCSS-8637 was implemented
  - HCSS-6866 was implemented
  - HCSS-8585 was implemented

  Section 4 for more information.

- **Utilities toolbox**

  - SimpleAsciiTableReader task
  - OpenTask task
  - FitsReader task update

  See Section 5 for more information.

- **Image toolbox**

  ImportImage now also imports the coverage.

  See Section 6 for more information.

- **Image dataset**

  Fixed some Wcs problems. Fixed a problem with coverage in SimpleCube.

  See Section 7 for more information.

- **Image display**

  - It is possible to set the ticks for the axis manually. Automatically setting the ticks is more intelligent now! More preferences to set the axes automatically.
  - Using the middle-button moouse wheel, you can zoom in and out on the image.
  - The status bar of the Display is now split in two lines. Extra information is the wave value in case Display shows a cube. You can change from the image view to the error/coverage/exposure/flag view.

  See Section 8 for more information.

- **Spectrum fitter**

  A whole new implementation of the SpectrumFitter has been made. This is called 'ExplrFitter' in the HIPE applications list. This new implementation uses the SpectrumExplorer for its visualisation. It supersedes SFTool and SFTask. In fact, those two are no longer available.

- **Source extractor**

  The most significant changes are listed here.

  - New tasks: sourceExtractorSussextractor and sourceExtractorDaophot
  - Jy/beam maps are now accepted as inputs, with the beamArea specified
  - DAOPHOT (as well as SUSSEXtractor) provides source positions to sub-pixel accuracy, with uncertainties
  - DAOPHOT (as well as SUSSEXtractor) provides uncertainties in the flux measurements
  - Input source lists (positions of known sources) are no longer filtered during extraction
  - Priors on flux and background are now accepted as optional input parameters for SUSSEXtractor

  See Section 9 for more information.

- **Plot**

  New features and other improvments.

  See Section 10 for more information.

- **Data input/output**

  - New FITS keywords added.
  - FitsArchive provides 'save' methods for working with File and OutputStream.

  See Section 11 for more information.

- **Datasets**

  - Dataset implements java.util.Map interface.

  See Section 12 for more information.

- **Array toolbox**

  - New REPEAT functionality added.

  See Section 13 for more information.

- **Product Access Layer**

  PAL query interruptible, local store storage support gzip and thread safety improvement

  - PAL query interruptible
  - Local store support storing gzipped products
  - Thread safety improvement

  See Section 14 for more information.

- **Calibration Sources Database**

  - Switch from custom protocol to HTTP
  - Improvements to documentation
  - GUI improvements

  See Section 15 for more information.

- Virtual Observatory

  - VO table import/export
  - Support for all applications on the External Tools page

  See Section 16 for more information.

- **QualityContext and QCR Browser**

  New improvements on the QCR Browser including a better graphical user interface and providing more info

  See Section 17 for more information.

- **Product Generation**

  RA & DEC, protection againts hang-ups, new OC metadata and better user functionality

  See Section 18 for more information.

- **Bulk reprocessing tool**

  Major upgrade of the bulk reprocessing utility tool

  See Section 19 for more information.

- **Task framework**

  - Tasks now provide a log message with the full invocation

  See Section 20 for more information.

- **Auxiliary Products**

  - A new function to de-correct aberration from product data has been implemented

  See Section 21 for more information.

- **User Installer**

  More comprehensive and intuitive error handing mechanism

See Section 22 for more information.

- **HIFI OTF mapping**

  First version of a customized, easier GUI for the DoGridding task. GriddingTask can work with one or a number of SpectrumContainers i.e. with Spectrum1d, Spectrum2d, HifiSpectrumDatasets and SpectralSimpleCubes. Gridding tasks avoid extrapolating data outside the borders of the map. Coordinates of reference pixel can be specified, full control of the location of the area to be regridded. An input WCS can be provided to perform the projection or rotate the result map. Output cube of weights together with the usual cube of flux intensities. Cubes can have ImageIndex for input spectra with irregular frequency spacing. Enhanced output table detailing the convolution process, to keep track of the spectra contributing to each pixel. Cubes get an ImageIndex array of wave values when the input datasets are not regularly spaced. Gridding tasks inform of their process and they can be interrupted. Better handling of exceptions. Fixed many minor issues, see Section 23 for more information.

  - DoGridding basic and expert GUI including combos for filter type and weight mode
  - More flexible selection of input datasets
  - Choose the location of the regular grid (reference pixel and its coordinates).
  - Optional WCS input to perform the projection and for the result cubes
  - Output cube weights
  - Avoid extrapolating data outside the borders of the map.
  - Convolution table: keep track of the contributions to each pixel, provide more details about the regular grid, about the filter function and about the shape of the regular grid.
  - Cubes get an ImageIndex array of wave values when the input datasets are not regularly spaced along the frequency axis.
  - Gridding tasks inform of their process and they can be interrupted.
  - Better handling of exceptions.

  See Section 23 for more information.

- **SPIRE**

  Updates in the calibration framework and in the spectrometer pipeline.

  See Section 24 for more information.

2. Documentation

The documentation has been reorganised to make it better suited to the current status of the Herschel Data Processing software. During the past few years the software has grown a lot, many new features have been added and GUIs are now available where previously the command line was the only option. Our two main manuals, the *User's Manual* and the *HowTos*, grew by incremental updates and developed duplication and inconsistencies, while some sections (sometimes entire chapters) went slowly stale.

The following updates try to address these deficiencies. This is the first release with the new structure, so expect a few hiccups.

- **New manual: Read Me First.**

This manual introduces you to HIPE documentation and describes what other guides are available.

- **New manual: Quick Start Guide.**

  This manual gets you started using HIPE with the minimum of fuss, confusion, time wasted, and helpdesk tickets raised.

- **New manual: HIPE Owner's Guide.**

  This guide introduces you to the general features of HIPE, such as perspectives and views. It is mainly drawn from chapter 2 and 17 of the old *HowTo Documents*, *Introduction to HIPE* and *How to Save/Play Back Scripts in HIPE*.

- **The HowTo Documents are now called "Data Analysis Guide".**

  The *Data Analysis Guide* includes much of the old *HowTo Documents*, plus corresponding parts of the old *Advanced User's Manual*. For instance, plotting is now described in a single chapter, instead of being split between two manuals. Each chapter has *How To* and *In Depth* sections which roughly map the old subdivision between the two manuals.

- **The Advanced User's Manual is now called "Scripting and Data Mining".**

  The User's Manual has been for a long time *the* manual on Data Processing software. Its role has progressively changed towards describing more advanced features, with an emphasis on low-level data structures and command-line interaction. The new title and content organisation reflect this new role.

- **New documentation categories.**

  The table of contents of the HIPE Help System now organises manuals into more meaningful categories.

- **URM categorised view now includes descriptions.**

  The *Categorized view of Commands* sections in each User Reference Manual now show descriptions for each task and command listed.

- **Developer documentation for modules now reachable from Help system TOC.**

  Package-level developer documentation, which was previously reachable following a tortuous path via the Javadoc, is now directly accessible from the table of contents of the HIPE Help System.

- **Updates to instrument guides.**

  Instrument teams have been working hard to update their documentation to the latest features of their software.

3. HIPE

The following list informs about changes in the HIPE framework. Therefore, it doesn't include improvements in particular views like the Data Access perspective, image viewers, ICC-particular tools, etc. For more information about them, please consult their own specific packages.

New functionality available:

- Tip of the day

  HIPE opens a typical Tip of the day dialog with useful advices and tricks. Whether it is opened or not at start up is easily configured by just clicking on a check box. This window is also available through the Help menu.

- Enhancements in Jython and text editors

Script and text editors provide new useful functionality:

- Highlighting (colors, whether to activate it) is now configurable through preferences.
- Syntax highlighting recognizes multiline strings (triple-quotes).
- Size of tab (number of characters) can be customized.
- Text editor more robust against Java Swing bugs.
- General improvements in find-replace dialog.

- Preferences links

  Some preferences may be related with others. Now preference panels may point to related categories. For example, Jython editor preferences include links to fonts and plain text editor preferences.

- Opened editors are re-opened the next session

  Editors are preserved between sessions, with the nuance that, if variables are not saved, editors and viewers associated to them are not reopened. Saving variables on exit or not can be configured through preferences.

- HIPE title shows version and selected editor

  Title of the HIPE window reflects the current version of the software. It also contains the name of the current script or variable viewer. In addition, the splash window has now the application title, which is shown in the iconified window of the desktop task bar when starting HIPE.

- Perspectives can be reset to their default layout

  Release 1.2 introduced persistence of perspective layouts, so user customizations are preserved when switching between perspectives, and among different HIPE sessions. Now rolling back to the default layouts is also possible, by right-clicking on the button of the current perspective, and pressing "Reset".

- Source code of Java tasks

  It is possible to open the source code of tasks provided with the installation that are written in Java. In order to do this, right-click on a task in the Tasks view, and select "View source".

  Note that this only works if you installed the source code. To do so when using the HIPE installer, select the checkbox next to the question *Would you like to have the source code installed?* This is only available if you choose the *Advanced* installation. If you are installing a developer build via the Continuous Installation System, use the `--src=yes --unpack=yes` command-line options.

- Multiple selection in the navigator view

  When you want to remove more than one file from HIPE, it is not needed to do it one by one. Now you can select many files and delete them at once.

- Non-blocking opening of FITS files

  FITS files are opened in the background, so it is possible to work with HIPE meanwhile, instead of waiting for the load to finish.

- Displaying meta data with different formats

  In the standard viewers, meta data is shown in a table. Parameters of type date can be shown in UTC (default) or TAI. Integers can be displayed in decimal (default) or hexadecimal. The format can be chosen through a right click on the corresponding cell.

- Refresh of viewers when data changes

  In the standard viewers for products, datasets and meta data, changes in the associated variable (e.g. with a Jython command) are reflected with an asterisk in the viewer's tab. Pressing F5 or "File > Refresh" shows the updated contents in the viewer.

- More log messages

  The Log view shows messages (that pass the existing filter) logged from all views through the standard Java logging mechanism. Moreover, the selected log filter is persisted among sessions.

- Run and stop buttons always present

  Stop button is not only shown when a Jython script is selected. Any perspective that contains the Console view would display this button in the main toolbar. Save and Print buttons are also kept in the main toolbar, when the editor area is shown.

- Quick clean of variables

  Getting rid of all existing variables is now easier. The new "Clear" button in the Variables view's toolbar allows to do it without needing to select the variables.

- Bugfixes

  This version contains fixes for bugs found in the previous release.

4. TablePlotter and OverPlotter

In this version, we rename the OverPlotter plot as "OPL" and the TablePlotter plot as "TPL(<dataset name>)". When switching to any column, the coordinate values can be seen by moving the cross icon on the plot. The axis is adjusted properly when in overlay mode in TablePlotter. The "Preference" button is changed to "Layer Props" where the layer properties can be set. We also move the zoom/pan factors to HIPE->Edit->Preferences where the factors can be set and saved. The layer "Appearances" is under construction. We also introduced a new feature which is to zoom in/out the plot using the mouse middle button. The zoom is centered at the spot where the mouse points.

New functionalities available:D_IA_GUI_EXPLORER_0_51

- HCSS-6235 was implemented.

  When the overlay feature is used in TablePlotter, the y-axis is adjusted according to the data in each layer. The layer who has the biggest ydata is referred.

- HCSS-6236 was implemented.

  The icon which shows the current corrdinates now is available in each plot throughout the colum (y-axis)/row(x-axis) selectors. It provides a handy tool to see the current coordinate values.

- HCSS-6238 was implemented

  The OverPlotter has a new tab name "OPL" and the TablePlotter has a new name "TPL(datasetname)".

- HCSS-6795 was implemented

  All layer free scale which bounded to the old buggy PlotXY library works fine with the current PlotXY pacakge.

- HCSS-7593

  The usage on Configuration class is completely removed from ia_gui_explorer. It no longer causes any changes in the configuration.

- HCSS-8430

  A new hipe prefrence is introduced in this release. Users can open Edit/Preference in hipe to change, reset the zoom and pan factors.

- HCSS-8356

  ia_gui_explorer now uses the new axis locking class, AxisGroup to lock and unlock single or group of axes. The deprecated locking methods from PlotXY are removed.

- HCSS-8637

  The Preference button in the previous version is renamed as "Layer Props". It is used to set preferences for layer only. The dialog to set pan/zoom factors is deleted.

- HCSS-6866

  ia_gui_explorer no longer sets the major and minor tickmarks. Instead, the tickmarks will be determined by PlotXY according to data to be plotted.

- HCSS-8585

  The middle button will zoom in and out the plot while the button is rolled. The zoom is centered on the position where the mouse pointer is.

5. Utilities toolbox

New features available:

- SimpleAsciiTableReader task

  Provides a simplified front-end for reading tabular data, but it is not as flexible as AsciiTableReader.

- OpenTask task

  Open via code the GUI of a registered task.

- FitsReader task update

  Now if FitsReader cannot generate a product of the expected type, it will load it as a plain product.

6. Image toolbox

New features/changes:

- Updated ImportImage to use the coverage.
- Corrected deprecated code
- HCSS-8704: Remove Popups from ellipseHistogram
- HCSS-6340: Image viewer not offered with SPIRE level2 image datasets
- HCSS-7955: Broken link in package.html
- HCSS-8320: ia_toolbox_image: NullPointerException when double-clicking on clamp task
- HCSS-8322: ia_toolbox_image: NullPointerException when double-clicking on ellipseHistogram task

- Made TaskUtil public.
- HCSS-8234: RotateTask unable to rotate images that contain error and coverage
- HCSS-8147: SimpleImage.setImage should fail with an exception rather than a warning in the log
- HCSS-8147: SimpleImage.setImage should fail with an exception rather than a warning in the log

7. Image dataset

New features/changes:

- Corrected problem with coverage in SimpleCube
- Corrected deprecated code
- Fixed problem when crpix3 is a long and not a double.
- HCSS-8371: Broken doc link in both JAVA Doc and Software Documentation
- HCSS-7936: SimpleImage removes coverage and exposure maps
- HCSS-8147: SimpleImage.setImage should fail with an exception rather than a warning in the log
- HCSS-8100: WCS isValid() fail for WCS with CDELT2 equal zero

8. Image display

New features/changes:

- Corrected deprecated code
- HCSS-6340: Image viewer not offered with SPIRE level2 image datasets
- HCSS-8745: Scan Map Level 2 products image, error, coverage datasets cannot be viewed as an image
- HCSS-8344: add wave value to the coordinates in the status bar of the Display showing a Cube
- HCSS-2093: Image axis tick selection inappropriate - adding missing ticks - Part 3
- Fixed problem with autoscaling of layers in HIPE
- HCSS-8694: Popup shown when executing normal valid code
- HCSS-2093: Image axis tick selection inappropriate - adding missing ticks
- HCSS-8466: size of flux cell in Display is too small
- HCSS-6340: Image viewer not offered with SPIRE level2 image datasets - Part 1
- HCSS-8462: Display.zoomWorldCoordinates(double, double, float) seems to work wrongly (upside down)
- HCSS-8584: DPAT2: include a middle-button roll zoom in ImageViewer
- HCSS-5909: DPAT1: ImageDisplay should allow to copy coordinate of current cursor position - part 1
- HCSS-1308: The tick spacing and tick labeling of the axis should be manually adjustable

- HCSS-2093: Image axis tick selection inappropriate
- PACS-1646: PACS EDP #6: Image viewer should auto-scale to tab size
- HCSS-8362: Display unable to open RGB images
- HCSS-8339: ia_gui_image: relocate "graphic" and "plot" preferences
- HCSS-8234: RotateTask unable to rotate images that contain error and coverage
- HCSS-8192: Display send exceptions on simplecube and images with ERROR and FLag
- Fix for HCSS-8094: Images are displayed upside down!

9. Source extractor

New features available:

- New tasks: sourceExtractorSussextractor and sourceExtractorDaophot

  Each algorithm now has its own task, to simplify the interface. The "sourceExtractor" task still functions as before, for backwards compatibility

- Jy/beam maps are now accepted as inputs, with the beamArea specified

  The "beamArea" task parameter allows the extent of the beam to be provided (in square arcsec). It is recommended to use a Gaussian approximation for this, to accompany the default Gaussian approximation for the point response function (FWHM provided by the "fwhm" task parameter)

- DAOPHOT (as well as SUSSEXtractor) provides source positions to sub-pixel accuracy, with uncertainties

  Uncertainties are estimated from the signal-to-noise ratio

- DAOPHOT (as well as SUSSEXtractor) provides uncertainties in the flux measurements

  Uncertainties are estimated from the signal-to-noise ratio, which is derived from the significance of the detection in the DAOPHOT H-filtered image

- Input source lists (positions of known sources) are no longer filtered during extraction

  Previously SUSSEXtractor was altering the positions and DAOPHOT was applying roundness and sharpness filters to remove certain sources. Additionally, the input source list is not now re-sorted during extraction

- Priors on flux and background are now accepted as optional input parameters for SUSSEXtractor

  Default values are provided for the flux priors. If no background priors are provided, these are estimated from the image itself

- Better detection of local maxima

  Spurious detections near the edges of the images are now greatly reduced. Default local maximum search radius ("pixelRegion") is now set to 1.5 pixels. I.e., a pixel is considered to be a local maximum if it is greater in value that all of its 8 immediate neighbours (ignoring bad pixels)

- Improved default size of PRF

  Based on input FWHM. See log message when task is executed

- Logging messages

These have been tidied up

- Weight map

  A bug has been corrected so that SUSSEXtractor now handles the error map correctly

- Thresholding in SUSSEXtractor

  The implicit log(evidence) threshold has been removed for when SUSSEXtractor is used in signal-to-noise mode

10. Plot

- New Axis types

  Two new axis types are added: RIGHT_ASCENSION and DECLINATION

- Mouse behaviours improvements.

  Mouse behaviour binding is configurable in HIPE preference

  Rotating mouse wheel can zoom in/out the plot

  Dragging with mouse middle button can pan plot

  A tiny mouse drag can be treated as a single click

- New style of properties window

11. Data input/output

New features available:

- New FITS keywords added
  - velocityDefinition (HCSS) = VELDEF (FITS)
  - radialVelocity (HCSS) = VFRAME (FITS)
- FitsArchive provides 'save' methods for working with File and OutputStream.

  Two new 'save' methods have been created. A java.io.File or java.io.OutputStream can be used as argument.

12. Datasets

New features available:

- Dataset implements java.util.Map interface.

  Any class that extends from CompositeDataset, Product class included, implements java.util.Map interface for accessing its datasets.

13. Array toolbox

New features available:

- New REPEAT functionality added.

It repeates an array, one rank higher than the input, with the new dimension containing n repeated copies of the input array.

14. Product Access Layer

New features include:

- PAL query interruptible

  PAL query can be interrupted now from Hipe or Jide. See HCSS-7315 for more information.

- Local store support storing gzipped products

  See HCSS-7704 for more information.

- Thread safety improvement, deprecate SynchronizedPool and Lockable interface on ProductPool

  SynchronizedPool(in herschel.ia.pal.util package) and Lockable interface on ProductPool were deprecated now, since all pools implementation were supposed to be thread safe. See HCSS-8040, HCSS-7811 for more information.

15. Calibration Sources Database

New features available:

- Switch from custom protocol to HTTP.

  This will avoid problems with firewalls. Also, the HTTP daemon has better performance and reliability, when multiple users access the DB simultaneously.

- The user manual has been improved. Example scripts have been updated.

- GUI panels showing tables now allow to save these tables to FITS files.

16. Virtual Observatory

New features available:

- Herschel Products can now be exported as VO tables, such that they can be read by VO applications supporting this protocol.

  When a Product is exported, for example by sending it to Topcat, it is sent using a particular ID. The datasets will appear in Topcat as tables. When one of the tables is sent back to HIPE, the ID that was used to send the product is used again, and HIPE will update the existing product in HIPE with the data from Topcat. A copy of the original Product is maintained.

- All applications on the External Tools page are now supported, both for sending data to them, or receiving from them.

17. QCR Browser

New improvements on the QCR Browser:

- Reports the pipeline final state

- Shows when a summary report already exist for a given list of observations

- Removable User comments (even when the QC was already saved)
- SPA can create custom quality flags

18. Product Generation

New improvements are:

- New averaged RA and DEC algorithm
- Better protection againts locking problems
- New compulsory metadata
- Manual start-up of the automatic processing
- Import of observations form a file
- Properties

19. Bulk reprocessing tool

Major upgrade of the bulk reprocessing utility tool:

- Bulk reprocessing based on a list of observations
  - Enable/disable OD processing for the ods containing the previous list of observations
- Filtering capabilities: Retrieve a list of obsid for bulk reprocessing from a query to the HSA (the syntax must be the PAL one e.g p.meta['odNumber']==121)
- Filtering capabilities: Retrieve a list of observations based on an external file with observation ids
- Suspend/Resume/Stop the jobs (no need to wait for the completion of the OD).
- Information about planned jobs and active jobs
- The GUI reports:
  - Job id for every pipeline running
  - Processing time for each pipeline

20. Task framework

New features available:

- Tasks now provide a log message with the full invocation

  Tasks now print a message with the full call, showing all parameters passed, including optionals with defaults. Example: INFO: [Full Call] restore(file="/home/jadiaz/.hipe/session/variables.ser" [mandatory])

21. Auxiliary Products

- A new function to de-correct aberration from product data has been implemented. The new function's signature is `PointingArray`

`PointingUtils#correctAberration(PointingArray pt, Long[] fineTimes, OrbitEphemerisProduct e)` This function avoids using Ephemeris.

22. User Installer

A more comprehensive and intuitive error handing mechanism has been added to the User Installer. The User Installer, in case of an error in the underlying developer intaller, receives a report from it, writes in the log, informs the user and aborts the installation. This will help us give more comprehensive support to our user community

23. HIFI OTF mapping

List of new features available and problems fixed in this release.

- Customized, easier GUI for DoGridding task

  The DoGridding task offers a GUI where one can use combos to select the type of filter and the mode to assign weights. In addition, the GUI offers a "basic/expert" button to toggle between a list of parameters used more often and another one of those used less frequently.

- GriddingTask to make cubes from SpectrumContainer, SpectrumContainerBox or collection of SpectrumContainers

  The DoGridding task is available to work with products of type HifiTimelineProduct, the main product of the HIFI pipeline. But, in addition, a task is offered which accepts more generic inputs. This is called GriddingTask. It can accept any container able to behave as an SpectrumContainer. This means that the GriddingTask can accept Spectrum1d datasets, Spectrum2d, HifiSpectrumDatasets and even SpectralSimpleCubes. This task can accept one individual container or a collection of them e.g. an ArrayList of HifiSpectrumDatasets. It can work also with any product which implements the SpectrumContainerBox interface.

- Convolution table: details about the convolution process

  The DoGridding task offers as an option (via the "detail" boolean input) the possibility to create a table providing full details of which spectra have contributed to each pixel of the regular grid, which were their input fluxes and weights, the weights computed by the filter function, the distance of each contributor spectrum to the pixel, etc. It also includes a set of parameters in the metadata of this output table describing the size of the beam, the pixel size, the type of filter and its parameters, the dimensions of the regular grid, etc. This output table can be used to created fancy Displays or plots of the construction of the cube, overlaying the spectra coordinates or the filter length on a display of the output cube(s).

- Avoid extrapolating data outside the borders of the map.

  This is a fix for SPR HIFI-2916: OTF mapping tasks (DoGridding, GriddingTask) must not extrapolate data.

  The algorithm used by the gridding tasks is a based in a convolution with a filter function, within a kernel centered at each pixel of the regular grid. This method has an intrinsec problem at the outer sides and in the edges of the area observed: it 'extrapolates' or directly copies the intensity from the nearest neighbouring spectra. This is something common to all the image reconstruction techniques based on a local interpolation with a convolution filter. For maps not rotated with respect to the north axis, the task computes an area that spreads half pixel outwards the sides of the area actually observed. So this problem is not noticeable.

  However, for images made for maps observed with certain rotation angle (non zero PATT), the result grid contains 'corner' areas which are several pixels away from the area actually observed. In some of those pixels, outside the area observed, one or two pixels away, the contributions are very weak, due to the closest spectra, but once the interpolated spectra are renormalized by dividing by the accumuled weight, the intensity of those spectra nearby the pixel outside the map becomes 'restored' to a value very close or idential to that of the neighbour spectrum. To avoid this, the pixels outside the map are now detected by checking the total weight accumulated for each pixel, and blanking those pixels where that accumulated weight is below a threshold value which clearly identifies those pixels which lie outside the area observed. A new property ("hcss.dp.hifi.otf.filter.threshold") has been added to control that. Also the task has a new optional input to disable this blanking process ("extrapolate" boolean input, false by default).

The detection of the pixels outside the observed area is based on checking the global weight computed for each pixel by the convolution. These pixels get a very low accumulated weight, compared to pixels inside the map. Hence once the convolution process is performed, a post-processing is done to review the accumulative (total) weight. The pixels with very low total weight i.e. outside the map are blanked, at present by setting their flux intensity to NaN.

The property hcss.hifi.dp.otf.filter.threshold can be used to specify which should be the minimum accumulated weight. Every pixel whose result weight is below the value assigned to the hcss.hifi.dp.otf.filter.threshold will be "blanked" by setting its flux to the Double.NaN value. If this property is not set in the (HIPE) environment, the gridding task will discard pixels whose total weight is less than a 1% (0.01) of the maximum accumulated weight.

- Output cube of weights

  In addition to the result cube of flux intensities, each output cube product includes and additional three dimensional matrix of weight values. These are the result weights computed for each pixel of the cube, for every channel, as a result of combining the weights assigned to each spectrum (e.g. read from the weights column) and the weights computed by the filter function. The more contributions there are for a pixel (from more and closer spectra), the bigger this weight becomes. These weights matrix can be used to double check the validty of certain pixel.

- Choose the location of the regular grid (fix HIFI-2950)

  The user can specify the coordinates of a reference pixel within the regular grid. Thus the user can control the location of the regular grid. This solves SPR HIFI-2950: cube parameters CRVAL1, CRVAL2 do not change regardless the refPixelOffset input. The solution is the following: the user can specify both which is the pixel coordinates of the reference point, within the regular grid, and the coordinates in the sky of such pixel. Based on these inputs, a proper grid will be computed and the headers of the result cube (CRPIXi, CRVALi, etc.) are properly computed.

- More flexible selection of input datasets

  This has been solved by two means:

  - More ways to sepcify which datasets should read from the input HifiTimelineProduct: the user can specify certain types of dataset to be read and whether to ignore those flagged with "isLine" in the summary table or not. The user can specify more than one type of datasets (e.g. observations containing "other" as type of dataset. The user can also specify some of the datasets, by providing their indices within the HifiTimelineProduct. By default only the 'science' datasets where 'isLine' is true are read to compute each cube.

  - GriddingTask can be invoked with the result of previously using the SelectSpectrum task. This means that the user can make sophisticated selections before eventually making a cube based on that selection, instead of simply picking a given dataset or a number of them.

- Possibility to provide/specify a WCS for the cubes and to perform the projection.

  By default the OTF mapping performs a GLS projection of the spectra coordinates into a flat map. However, the user may provide an input WCS to perform such projection. In such case the pixel coordinates of each input spectra are computed with the given WCS. In addition, the spatial parameters of the input WCS is copied to the output cubes (additional parameters are added to describe the wave (frequency) axis of the cube. This allows also to rotate the result cube by using an input WCS with certain rotation angle.

- Flux unit and description is set in the result cubes (fix SPR HIFI-2924)

  Now the result cubes include units and description of the flux data

- HCSS-compliant header in cubes metadata.

  Now the metadata of the cubes created with the gridding cubes is complete according to the specification for Herschel products. No standard parameter is missing

- Other (minor) problems fixed.

  Namely, the following SPR's have been fixed

  - HIFI-2098 gridding task should not use dummy pointing products if they are not supplied
  - HIFI-2101 the available weighting options should be in a drop-down menu rather that hand entered.
  - HIFI-2098 gridding task should not use dummy pointing products if they are not supplied.
  - HIFI-2949 remove old inputs from DoCube and DoGridding task. Now positions have to be previously computed with DoPointingTask.No pointing, siam, auxiliary products are accepted any more.
  - HIFI-2924 flux unit and description set in the result cubes
  - HIFI-2906 hifiGriddingTask does not need 'science' type of datasets as user input
  - HIFI-2845 spectra do not have correct sky frequency in LSB of cubes

- Cubes get an ImageIndex array of wave values when the input datasets are not regularly spaced

  When the input spectra have not been regridded along the frequency axis, i.e. the channel separation is not constant, an ImageIndex is created to specify the frequency of each layer (image) of the cube.

24. SPIRE

- Calibration Framework
  - The calibration products have been removed from the build.
  - The CalImport class has a partially revised API, containing some new methods and the ability to select URLs as well as file names.
  - For backward compatibility, the cal_import command with no arguments will continue to work as long as the user is on-line (it now downloads the data over the internet rather than reading it from the build).
- Spectrometer Pipeline
  - Spectrometer pipelines are now able to merge like building blocks for long observations.
  - Browse product image that is created by the spectrometer pipelines shows the spectra for the central detectors of each array.
  - Pointing information is now computed using the Beam Steering Mirror Timeline.
  - A function that corrects detector samples that have been clipped has been added to the spectrometer pipelines.

- The default algorithm for the Baseline Correction function has been changed from a polynomial fit to a low frequency Fourier fit.