A design for the data pipeline for the prototype GBT K-Band Focal Plane Array

Introduction

See the KFPA wiki for more information on the GBT K-Band Focal Plane Array. Other links of interest:

- KFPA System Planning
  - Hardware M&C of KFPA
  - Software M&C of KFPA

Workshop

A Science and Data Pipeline Workshop was held in Green Bank at the end of November, 2007. The last 2 days of the 3 day workshop were focused on radio astronomy data pipelines, primarily processing single dish data, that are in use elsewhere. The goal of that workshop was to gain exposure to what has already been done in this field, to learn from them, and, where possible, to borrow concepts or software for use in processing KFPA data. Immediately following the workshop, a small group gathered and sketched out a design. A preliminary design following from this initial design outline was presented at the conceptual design review in February 2008. The design described here is the current working design. Some components described in the previous design are now

explicitly out of scope for the initial data pipeline (cross-correlation calibration, more complicated calibration schemes similar to "basketweaving"). Baseline fitting and removal is now shown as an explicit step in the data flow diagram. These refinements and other changes from the preliminary design followed from the KFPA data analysis meeting held in June of 2008 as well as a memo describing the planned KFPA observing modes.

Pipelines

A fair amount of time was spent at the workshop discussion what a pipeline is. In Green Bank, it has a somewhat broader definition than perhaps it has at most other similar facilities. The largest difference is that at the GBT we have historically given observers access to the rawest form of the data as it comes from the various backends. This gives observers a tremendous amount of flexibility in how they set up their observations and calibrate and reduce the resulting data. Information on switching states is included with the data. That includes whether a noise diode was injected ("on") or not ("off"); the frequency switching information; the type of scan from a list of standard scans that includes pointing, focus, two-scan position switching, frequency switching, mapping series; and recently, the ability to tell whether the subreflector was moving when a standard procedure to move the subreflector was used. Users are then expected to reduce the data using that switching information and any notes they took to average the integrations during a scan when appropriate, possibly average scans, generate maps, etc, using code that NRAO provides (e.g. GBTIDL) or that the observer writes or has received from another observer. GBTIDL procedures exists to aid in this processing (appropriately using the switching signals, averaging with appropriate weights) but the user is responsible for considerable overhead and processing (e.g. which scans to average, which scans to send to the imaging tool to produce an image, what additional spectral lines are present in the data beyond the one that is indicated in the metadata, etc.). At most other similar facilities, the initial data product has already done many of those steps to produce an initial roughly-calibrated dataset.

At other institutions a data pipeline takes the roughly-calibrated dataset, possibly does some additional editing and flagging and produces images. In Green Bank, we need a pipeline to do the initial processing as well as the subsequent flagging, editing, and imaging. In at least one case, ACSIS, there is a pipeline hidden from the user that does that initial processing as part of the black box that is ACSIS.

GBT data processing needs a pipeline apart from the needs of the KFPA. Data rates are already sufficiently high and the processing in many cases is routine enough (because of standard observing modes as well as standard configuration cases) that a useful pipeline could be developed for most GBT observations. Most observers develop their own regular processing steps. That is essentially a pipeline, even if they don't go to the bother of writing a script to automate it. Most of the design elements described in this document will be useful for other GBT spectral-line data processing and they should be designed with that wider use in mind where possible. A more general GBT data analysis pipeline will also require well-described standard observing scenarios that, when followed, the pipeline has sufficient information to usefully process and eliminate many of the steps that observers now do by hand. The individual components may also have additional features to allow explorations of the data processing outside of the default parameters that the pipeline uses. This is particularly important for advanced users or observing using "out of the box" observing strategies supported by the flexibility of the GBT in their data reduction. The individual components used in any GBT pipeline must be flexible even if the pipeline itself doesn't use all of that flexibility so that we can reuse as much code as possible.

Expected Data Rates for the KFPA

The initial backend for the proposed 7-element KFPA is the current GBT Spectrometer. There will be no more than 16384 channels per sampler when using 12.5 MHz or 50 MHz bandwidths (more channnels are possible only by sacrificing elements sampled by the Spectrometer). With 4 switching phases per integration (noise diode

plus frequency switching) there will be 65 kB per integration. The hardware-limited dump time of the spectrometer in this mode is 0.36 seconds ro 10 MB/s. More typical dump times in the current configuration are 2 seconds or 1.8 MB/s.

For a 3'x3' image (26 beams) and an integration time of 1 minute per beam, the 7-pixel KFPA will produce 406 MB of data. For a 12'x12' image (a typical IRDC) then 6.B GB of data will contribute to that image.

The above data rates and typical data volumes for are based upon the GBT Spectrometer's capabilities. The planned 61-pixel K-band array requires a new backend capable of delivering spectra across the entire 1.8 GHz instantaneous bandwidth of each feed at the same resolution as the current spectrometer. The resulting data rates and volumes will be nearly 600 times larger than the 7-pixel prototype array or about 10 GB/s. One of the goals of the data processing work for the 7-pixel prototype is to explore options for handling those large data rates without having to rewrite substantial parts of the software. In other words, the data pipeline software should scale well as the number of pixels and channels increase.

Existing GBT Data Analysis Software

The recommended path for producing images from the GBT Spectrometer is the following:

- The sdfits tool is used to produce an SDFITS file containing the raw data along with the metadata describing the observations.
- That data is processed in GBTIDL. This includes combining the data from different switching states (cal on and off, frequency switching, etc.) and calibration information (opacity, efficiency, cal temperatures) to produce calibrated data. Data smoothing, averaging, flagging, and baseline fitting and removal may also happen here.
- The data are converted to a form recognized by AIPS and AIPS is used to produce the images.

Other paths exists but are not well supported by the NRAO software division or well documented (e.g. aips++, CLASS, alternative IDL routines not based on GBTIDL).

The recommended path can be used to produce spectral line cubes from the existing 2-feed K-band receiver and GBT Spectrometer backend. The data reduction and imaging steps offer a great deal of flexibility to match the flexibility available in setting up the observations. While individual users likely have written a few scripts to take their data through those steps (i.e. very nearly a pipeline), there is no official recommended route from setup through data taking to reduced images.

The calibration step typically involves using the cal on and cal off data along with lab-measured values of Tcal to turn the raw data counts into antenna temperatures. The two largest factors that limit the calibration accuracy are that astronomical measurements of Tcal are more accurate than the lab-measured values and a scalar average Tcal value from the center 80% of the measured Tcal values across the bandpass are used to calibrate the data without regard to any structure in the Tcal spectrum. An improved calibration design to address these issues and other related calibration issues is being designed and is expected to be available to GBTIDL users next year.

Cross-correlation spectral-line data is well supported in the first step of the recommended path but is unsupported at the other 2 steps. Individual users have written their own data reduction routines for spectral-line polarimetry observations.

GBT continuum data reduction is particularly poorly supported at the moment, especially for the DCR. There are old, unsupported, aips++ DCR tools that can be used. Other continuum backends have their own data reduction

paths (e.g. OBIT for Mustang and the CCB, there is also some IDL data reduction software for Mustang that is completely independent of GBTIDL).

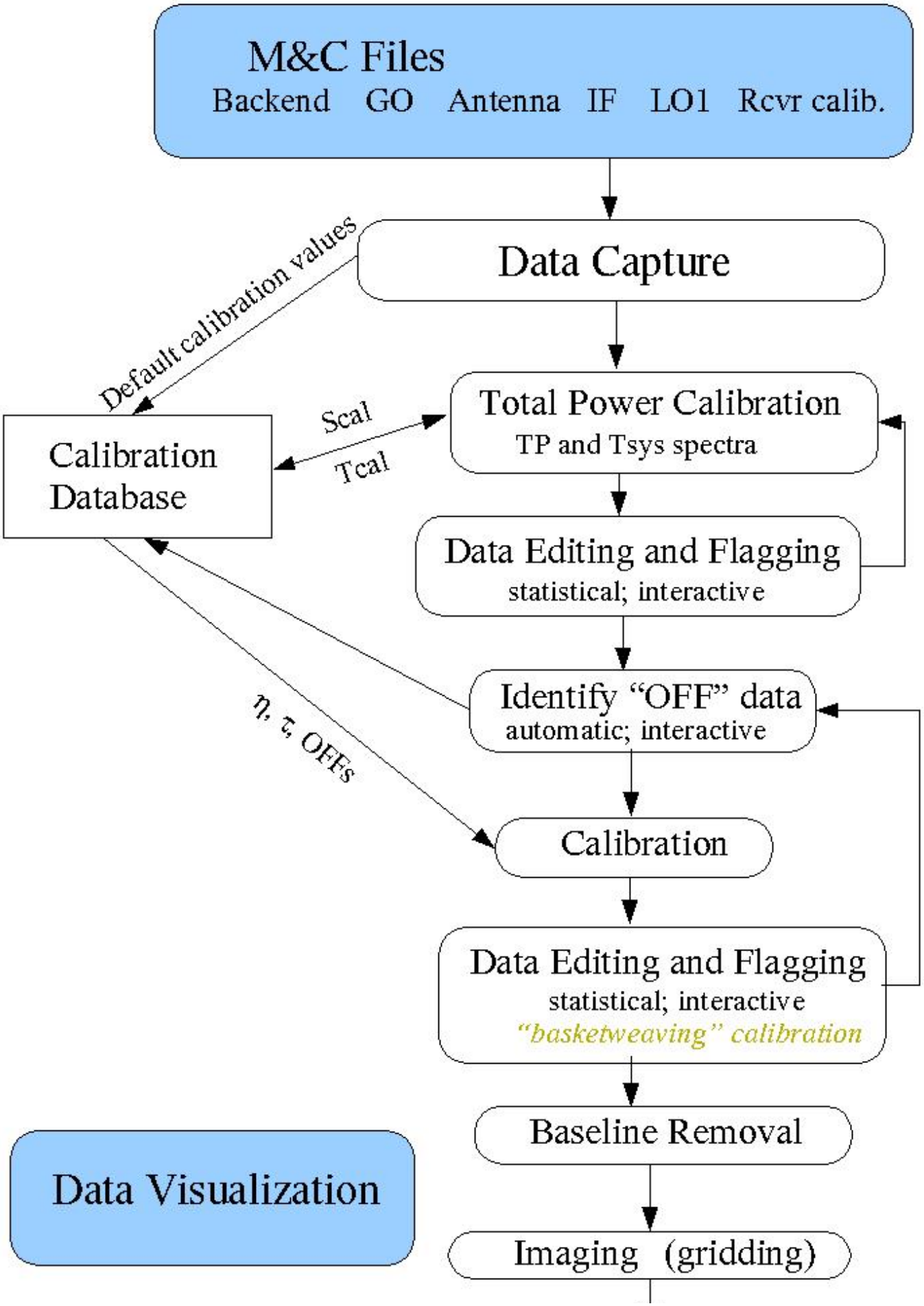
Goals of Prototype Pipeline

These goals are listed in order of priority.

- Provide sufficient data analysis support so that the hardware can be commissioned.
- Explore the analysis path, especially areas that we can not do adequately with existing tools (e.g. vector calibration, statistical data flagging and editing, large dataset visualization and editing).
- Prototype tools necessary to support the full 61-element array.
- Prototype pipeline components and mechanics to automate the data reduction as much as possible. Add the necessary metadata to the raw data to facilitate the pipeline. Define standard observing modes and data reduction paths that are processed by the pipeline.
- Based on the prototyped tools, estimate the costs associated with delivering a pipeline and necessary hardware to handle the expected data rates and volumes of the full 61-element array.
- Develop these tools, including any pipeline infrastructure, such that they can be used with data from other GBT backends. The tools should be as independent of the backend specifics, receiver specifics, and data volume as practical given the resources available for this development.

Pipeline Design

The data flow through the pipeline is described in this figure:


M&C Files
Backend GO Antenna IF LO1 Rcvr calib.
Data Capture
Default calibration values
Calibration Database
Scal
Tcal
Total Power Calibration
TP and Tsys spectra
Data Editing and Flagging
statistical; interactive
Identify "OFF" data
automatic; interactive
η, τ, OFFs
Calibration
Data Editing and Flagging
statistical; interactive
"basketweaving" calibration
Baseline Removal
Imaging (gridding)
Data Visualization

A fully functional pipeline is not necessary for the initial 7-element array using the GBT Spectrometer. Individual components are ready now or could be ready with little additional work that would provide reasonable first-pass images from this instrument. The existing sdfits tool can handle the data capture (association of meta information with telescope pointing data and raw data from the spectrometer). Some small amount of work is necessary to ensure that the feed offsets are properly translated into pointing direction on the sky in the output SDFITS file. Crude calibration can be done using existing calibration steps in GBTIDL (i.e. scalar, lab-measured Tcal values, default opacities and efficiencies). Some statistical flagging has been developed in GBTIDL using the available command-line tools. All of these GBTIDL routines are easily translatable to Python.

A crude pipeline could be assembled from the existing components to provide quick-look images useful during commissioning. As discussed previously, a pipeline for handling GBT data is useful beyond the needs of the 7-pixel KFPA. A pipeline is required by the larger bandwidths and data volumes of a new backend and 61-pixel array. Such a pipeline needs to scale well as the data volumes increase - using the available computing resources effectively to move the data through the pipeline efficiently.

All of the raw observation data starts off a collection of engineering FITS files produced by the GBT M&C system. Those files include:

- backend FITS files
- GO FITS files contain meta information describing the observations
- Antenna FITS files contains antenna pointing information and feed location information
- IF and LO1 FITS files describe the frequency setup during the observations
- receiver FITS files describe any lab-measured Tcal values

The collection is time-ordered and relationships between the scans in the data are determined by examining the metadata.

Building from the raw engineering FITS files listed above, there are then two general types of data that the KFPA pipeline will need to handle. The first is calibration scans. These will be observations of astronomical calibrators used to determine relevant calibration parameters to be applied to the rest of the data. The most common type of calibration scan is one designed to determine the Tcal spectrum and use that instead of the default lab-measured Tcal values. Everything else falls in the second type of scan. These are scans that are to be used in building up an image cube.

For example, the outline of the spectral line pipeline is (see also the attached digram showing the data flow through the major components of this pipeline):

- The individual FITS files are gathered together to produce a data set with properly labeled data and associated metadata (**data capture**).
    - This step puts the default calibration values into a calibration database (lab measured Tcal spectra, default efficiencies, weather-based opacities, etc.).
- Total-power calibration and system temperature determined (Tsys).
    - If this is a calibration scan then the result is a measurement of Scal (Tcal spectrum determined astronomically). This result is saved in the calibration database for later use.
        - User interaction with the calibration database is necessary here to watch for problems in the Scal determination. It is likely that statistical tests can be developed to reduce the need for that interaction. If the calibration database is edited interactively it will be necessary to reprocess the other data through the pipeline to refine the result.
    - For other scans, the most appropriate Scal or default Tcal spectrum is retrieved from the calibration database and used to produce the Tsys spectrum.
        - The total-power spectra and Tsys spectra are saved for later use.

- Statistical tests are done on the total power and Tsys spectra in order to auto-flag the data (e.g. RFI flagged).
    - Interactive visual flagging and editing is also necessary here for problem data. This step is obviously not part of the pipeline, but the pipeline may be restarted from this step to refine a previous result.
- OFF spectra are identified where appropriate. Standard observing modes and appropriate metadata are necessary to automate this step. OFF spectra may come from:
    - designated off pointings (the telescope is moved to and tracks a specific location that is thought to be uninteresting)
    - synthesized off spectra from the uninteresting regions in data
        - Interactive identification of appropriate regions is necessary to override the defaults captured in the metadata when unexpected astronomical signal is suspected in the synthesized off spectra
    - OFFs may be processed before being used (e.g. smoothed, replaced by an appropriate polynomial fit)
    - OFF spectra are associated with the calibration database for use as needed.
- Appropriate OFF spectra are used with other calibration information (efficiencies and opacities) to calibrate the remaining spectra.
    - The resulting collection of spectra are ready for gridding into an image.
    - Different meanings of "appropriate" need to be explored here (nearest in time, location; interpolated; other).
    - OFF in this context also covers in-band frequency-switching where the names "off" and "on" are arbitrary but the underlying math is very similar to that used in the case of an OFF sky position.
- More data editing may optionally happen here (statistical, interactive).
    - A technique of using the near crossing points in the same feed as well as different feeds to refine the calibration should be developed (related to basket-weaving). Exploration of this technique is not required for early science with the KFPA and so is out of scope for this work. It will be investigated as time permits and any pipeline implementation details or design changes must not exclude this option.
- Image the data. An image cube and associated weights (variance) cube is produced
- Iterate until happy with the data editing, flagging, appropriate offs and calibration. This last step is an interactive step and likely involves interactive editing and flagging of the data and tweaking of the default pipeline parameters. The default pipeline obviously does not include this step.
- The final image and weight cube is taken to the users favorite image analysis tool(s) for further work, including combining images.

None of the steps described above is specific to the KFPA or even to a feed array although the various statistical editing and calibration steps will clearly need to take that information into account when present. Each of these components will be useful outside of the KFPA data pipeline. Each component will be developed as independently as possible so that as new technology and tools become available they can be substituted without having significant impact on other parts of the pipeline. This should also make it more likely that the individual components can be reused outside of the KFPA pipeline or in pipelines appropriate for other GBT backends.

The same data may be processed through the pipeline multiple times depending on how much data editing, flagging, and interactive setting of OFF region and calibration values the user chooses to do. For routine observations a default single path through the pipeline can be described for standard observing modes so that quick-look images can be made available shortly after the data are taken.

Continuum data can be extracted from the spectral line data at the appropriate point in the pipeline. The

GALFACTS pipeline does that with data from ALFA at Arecibo and similar processing steps will occur in this pipeline. This step likely requires a spectrometer capable of covering a wider bandwidth than the GBT Spectrometer. This step is out of scope for the initial pipeline but will be explored as resources become available.

The calibration and processing of cross-correlation data is not explicitly described in this pipeline. Additional steps include using standard scans to calculate the Meuler matrix and usage that to calibration the other cross-correlation (polarization) scans. That work is out of scope for the intial pipeline.

Language Choice

The language used to move the data through the pipeline and interact with the individual components will be python. This was chosen because of our familiarity with it in Green Bank. The other serious contender was Perl because of its use in the ACSIS/JCMT pipeline. That was rejected because we really have no expertise in that language and we didn't want to add yet another language to the set of languages in which we have to be experts. IDL could be used (even for parallel processing, a white paper exists describing how to do that, at least for the simple cases that we would need here) but IDL is not seen as being an appropriate long-term choice for GBT data pipelines. By using Python, we also leave open the possibility that these components and methods could be used in some future shared ALMA/GBT pipeline. As we gain experience and explore the parallel processing needs we may need to revisit this decision for those parts of the pipeline that need to communicate between threads (e.g. some of the statistical flagging and editing may have complicated parallel processing needs).

Initial implementation and prototyping of the pipeline components downstream of data capture (currently sdfits) will be done in IDL. This will also facilitate reuse of that work in the GBTIDL environment to benefit current GBT observers while the pipeline is being developed. Data capture improvements and developement will continue to be done in python. Pipeline components will migrate to python as they mature or when necessary to satisfy data processing needs.

Parallelism

Most of the steps described here are embarrassingly data parallel, meaning that the processing of reasonably sized chunks of data is independent of the processing of other similarly sized chunks. For most of the steps, data from an individual sampler (feed and polarization) can be processed independently. So long as appropriate weights are carried through to the end the data can be recombined in the final step to form one regularly sampled grid.

There are two obvious exceptions to this.

- Use of cross-correlation data. For the GBT Spectrometer, the initial data capture step converts the data from the lag domain to the frequency domain. The determination of the appropriate van Vleck correction for the cross-correlation data requires that the two related auto-correlation samples be examined first. This limits the granularity of the data that can be processed in independent data paths in the data capture step. This will not be important during the initial KFPA prototype due to the limitations of the GBT Spectrometer since 14 of the 16 samplers in the Spectrometer are already tied up with the autocorrelations of the two polarizations on the seven feeds. Since the next generation spectrometer needed for the 1.8 GHz bandwidth and 61-pixel array has not been designed, its difficult to anticipate the processing needs that cross-correlation places on this step.
- Automatic data editing and flagging. Data from different pixels that sample nearly the same signal can be

used to adjust the data and possibly also flag it. Similarly, data from the same feed that cross near the same point on the sky can be used to adjust the data (this is related to basket-weaving although the pattern is more complicated). Potentially complicated combinations of data across feeds and at different times makes it difficult to know how best to parallelize this sort of task. This is likely to be a significant data processing bottleneck. This work is out of scope for the initial KFPA data pipeline.

Data Formats

The existing data capture tool produces SDFITS which can be consumed by GBTIDL and converted to a format recognized by AIPS. SDFITS will be used by this pipeline until it doesn't make sense to do so. By keeping the data in SDFITS as much as possible we can use existing tools in GBTIDL to examine the data as needed. The existing GBTIDL flag file format will also be retained for the same reason. Parallel processing needs may make it desirable to split the data up into multiple SDFITS files at an early stage of the pipeline. GBTIDL can already open a collection of SDFITS files as a single dataset. If it becomes necessary to move to a different format the portions of GBTIDL that are sdfits-specific are well isolated and could easily be augmented to handle alternative formats.

Conversion tools will be written as needed to take advantage of desired functionality (e.g. imaging) that is already present in existing astronomy software packages. In other words, we wish to focus our software development efforts and limited FTEs on writing the custom components for the GBT data pipeline and leverage mature, third-party packages for other required functionality where possible. A conversion from SDFITS to classic AIPS for imaging is already available. A conversion from SDFITS to CLASS has been requested.

Proposed Components

Data Capture

The raw backend output (FITS file) are combined with the other associated FITS files (Antenna, GO, IF, LO1, etc.) to produce a coherent data set with properly labeled data. The sdfits tool does this now. For the GBT Spectrometer, this includes doing the van Vleck correction and FFT along with flagging bad-lags. For the initial prototype, sdfits is probably adequate for this component. Most of the associated information is independent of feed. The sole exception would seem to be pointing location. The conversion from lags to feed is likely the only place that parallel processing will be necessary for this component. This operation is independent from sampler to sampler except where cross-correlations occur. In the cross-correlation case, the lag information from the related auto-correlations is used to estimate the sampling levels and that is used to do the van Vleck correction for the cross-correlation case. Cross-correlations here may be between two polarizations of the same feed or between feeds. Since 14 of the available 16 samplers in the GBT Spectrometer are used for the auto-correlation data in the 7-pixel KFPA, this mode of operation is unlikely to be used in the initial pipeline. It will be useful when a wide bandwidth spectrometer is available for extracting continuum data from the output of the KFPA and doing continuum polarimetry observations similar to the way the GALFACTS pipeline uses ALFA data at Arecibo. This component also populates the calibration database with appropriate default values.

Calibration Database

This supplies the most appropriate calibration quantities (Tcal spectrum, opacity (tau), efficiencies, etc.) as requested. "Most appropriate" here may depend on the type of observing and choices that the observer makes.

The database is populated by default values by **Data Capture** and by astronomical determination of the Tcal spectrum (Scal) from the **Total Power Calibration** component. The user may also alter the contents of this database from time to time. Some very preliminary thought has gone into a GBT calibration database design in the context of GBTIDL. That work is likely to be not be useful "as is" for this pipeline because it is being developed for GBTIDL but more importantly, it is a very temporary database since it only remembers the most recent calibration parameters plus the defaults and so it is not as versatile as needed in the long run.

Total Power Calibration

Takes the raw data produced by the data capture step and combines the cal-on data with the cal-off data to produce a time-series of total power spectra with associated system temperature (Tsys) spectra or weight spectra. The most appropriate atmosphere and efficiency corrections (tau and eta) are retrieved from the calibration database. For calibration scans where an astronomical source of known power has been observed that information is used to produce an Scal spectrum which is then saved to the calibration database. For all other scans, the most appropriate Tcal spectrum is retrieved from the calibration database. User feedback may be necessary here to help evaluate the appropriateness of any determined Scal spectra prior to their being placed in the calibration database. Alternatively, this component can put them into the database and the database would then provide tools for examining it's contents and editing them (removing or flagging bad Scal spectra, adjusting values, etc.).

This step may be combined with the data capture step but it should be available separately so that users can examine the raw data and proceed through this step without having to re-run the entire data capture step.

Flagging of Total Power Data

The total power and Tsys spectra are examined for RFI and other problems and flagged as necessary. Statistical analysis is done to automate this as much as possible. User interaction will be necessary. We currently have no tools for visually interacting with large amounts of GBT data.

Determination of Off (reference) Data

Areas free of astronomically interesting signal are identified in this step. Off data may come from specific scans designated as such (off scans, reference scans) by the observer. In that case, this step is trivial. Off data may also be synthesized from the data set being processed (e.g. regions in the scanned area that are felt to be free of signal). The determination of synthesized off data may require user interaction to refine the default areas to use in determining the off data. There will also be various options controlling how this data is synthesized and used (smoothing, polynomial fitting, etc.). Once determined, the off data is part of the calibration database so that subsequent calibration operations can use it as needed. In-band frequency-switched data is a special case that may need to be treated differently. In that case, the names "off" and "on" are arbitrary but mathematically the combination done when the calibrated data are produced is similar to how off-position data are used. It probably doesn't make sense to but the arbitrarily designated frequency-switched off data into the calibration database.

Produce calibrated data

The offs are used along with efficiencies and opacities from the calibration database to produce a time-serious calibrated data set that is ready for gridding onto an image. See the note about frequency-switched "offs" above. The output format here will match the default gridder. Initially the ACSIS format will be used so that their gridder

and related tools can be used. Translators to other data formats for other gridders are likely (e.g. AIPS, CASA, SDFITS, other).

Data Editing

Once the calibrated data are in hand some additional editing and flagging may be necessary. This includes statistical and visual flagging as well as command-line flagging for experts (similar to the flagging in GBTIDL and the statistical flagging that has been developed there and the visual flagging found in aips++ and CASA and used by some *dish* users). One specific editing technique that will be developed is using the data crossing points in the scanned region to refine the calibration. This is related to basket-weaving although the exact pattern of the feeds over the scanned region will likely be different and more complicated. This technique (related to basket-weaving) is out-of-scope for the initial pipeline.

This step is likely to be computationally intensive and it is unclear how best to parallelize this step. In some cases (e.g. the editing resulted in changes affecting the off data) it will be necessary to reprocess the data through some of the upstream components.

Baseline Fitting and Removal

This can be considered as one possible data editing option. At the June, 2008, meeting it was felt desirable to make this step an explicit option so that it is clear that users can expect to be able to do this prior to gridding should they choose to do so. This also makes it explicit that the pipeline needs to provide controls to guide the automatic baseline fitting and removal (e.g. descriptions of the signal-free regions to use in the fit, the type of function to fit).

Gridder

The gridder turns the data into a regularly sampled WCS FITS cube. The current recommended gridder for spectral line data at the GBT is classic AIPS. Initially, the pipeline will use that gridder but it will hide that detail from the user. Other gridders could be substituted here (e.g. casa, ACSIS) provided that a data translator is written to convert the relevant pipeline data into a form usable by that gridder. All of those details will be hidden from the user. Using at least one addition gridder will be useful check against the images output from classic AIPS.

Data Visualization Tools

Various capabilities for visualizing the large datasets are needed throughout the pipeline. During routine pipeline operation, visual feedback is needed to reassure the user that the data quality is good and the pipeline is operating as expected. The interactive portions of the pipeline need visualization tools to aid in data editing and flagging, including the ability to visually flag the data (e.g. draw polygons around regions to flag). Recommended image visualization tools are needed for the end-product of the pipeline. Some preliminary work on un-gridded data visualization and interactive flagging was done during the summer of 2008. That work will be adapted for use in the KFPA pipeline.

Metadata

This is not really a component but rather a discussion of some of the information likely to be necessary to

automate the data processing as much as possible. It is relatively straightforward to add new metadata to the existing GO FITS file (or other M&C FITS files if that seemed more appropriate). The sdfits tool can add any such metadata to the output SDFITS file as necessary.

Different observing strategies and science cases will likely lead to different recommended defaults for the pipeline components. Observers may wish to tweak those defaults. Some defaults may vary from field to field (e.g. the locations of appropriate signal-free regions). For each of these reasons and likely for other reasons, it is useful if the behavior of the pipeline can be described by a simple script or set of rules. The choice of rule set could be captured in the scheduling block or it could be done by the user when the pipeline is run. It is important that as much of this as possible be captured so that reasonably processed data can be recovered from the archive at any point in the future.

Development Priorities

1. Calibration
    - The GBTIDL calibration enhancements already planned will continue. There are a few competing designs on how best to take into account variations in Tcal and gain across the bandpass and these options will be explored. We will also explore how best to use the existing weather database to get default predicted zenith opacities. These enhancements have a fairly primitive database design to hold the derived calibration quantities (bandpass Tcal, user determined opacities and efficiencies). Soon after the GBTIDL enhancements are complete, work will start on a more comprehensive calibration database capable of handling the needs of the KFPA data processing described here and building on the lessons learned during the GBTIDL calibration enhancement project.
2. Data capture
    - The sdfits tool will be the data capture component at the head of the KFPA data processing path.
    - Work on improving the sdfits processing speed will continue.
    - The pointing directions recorded in the SDFITS output need to include the offsets for the appropriate feed. Currently, the output positions are simply the tracked positions with the feed offsets available in separate columns in the SDFITS output.
    - The ability to put the default values into the calibration database will be added as part of the GBTIDL calibration enhancement project.
    - With regard to the expected use cases and observing modes for the KFPA, appropriate new meta information needs to be added to the M&C FITS files and copied through the data capture process so that it can be used by any pipeline.
3. Explore parallel data processing in existing tools
    - In the current cases, the primary bottlenecks are in data capture where the raw lags are converted to spectra (the van Vleck correction and FFT) and written out as FITS files and the where the data are converted for export to classic AIPS. Given the short development time, only embarrassingly parallel options involving diving up whole tasks across different processors (e.g. each processor would fill one of the 4 spectrometer banks using sdfits) or using available parallelized libraries (e.g. FFT) to take advantage of multiple processors will be investigated. These will be investigated on existing multi-core machines. New multi-core hardware dedicated for use by the pipeline will be purchased as late in the development as possible.
4. Statistical data flagging
    - A few users have developed statistical techniques for automatically flagging data in GBTIDL. Those techniques need to be translated to Python for use in the data pipeline. Additional techniques will be developed.

5. Pipeline mechanisms
    - Concepts and code will be reused from the GALFACTS pipeline where appropriate.
6. Data Visualization
    - An NRAO summer student produced a preliminary GBT data visualization tool during the summer of 2008. It allows the user to view and visually flag SDFITS data. This work will be re-used and adapted for use by the KFPA pipeline as well as within GBTIDL.
    - Allow easy exploration of the large data set so that the observer gets a sense of the data quality and general state of the data.
    - Allow for interactive data editing (flagging).
    - Provide a default recommended tool for interacting with the images output at the end of the pipeline. This is likely to be a different tool from that used to interact with the un-gridded data.

1 Conversion tools

-
    - SDFITS to the casa MS so that the casa gridder can be used.
    - SDFITS to ACSIS data format. This will allow us to explore the ACSIS gridder and viewer (GAIA) for appropriateness for this project.

Resources

- Bob Garwood, NRAO - component development, 1 FTE
- Roberto Ricci, University of Calgary - algorithm development.
- NRAO summer student - preliminary dataset visualization tools, summer 2008

This topic: Kbandfpa > WebHome > SWPlanning > KFPADataPipelineDesign
Topic revision: r23 - 2009-01-20 - 12:53:34 - BobGarwood