The GBT Archive Process

Melinda Mello, Dana Balser, Brian Kent
December 19, 2013

Table of Contents

1. Preface

Much of the GBT archiving system documentation is in wiki form. The major contributers of this effort are Dana Balser, Brian Kent, Melinda Mello and Gareth Hunt. This document contains much of the original wiki and document text verbatim and only serves to transfer the original documentation into word format. The link to the wiki documentation is https://staff.nrao.edu/wiki/bin/view/OSO/OSOGBTArchiveDocs

2. Background

The GBT has been in operation for over a decade. During most of this time a real archive did not exist but data were backed up on tape or disk in Green Bank (GB). Here we define "archive" as data, meta-data that describe these data, and any ancillary data (e.g., weather, logs, etc.). To be useful a user must be able to search and retrieve data from the archive. In 2011 a project was started to create a GBT archive with copies in GB and CV, connected to the NRAO-wide archive access tool (AAT) in Socorro (SO). The main project goals were completed and the GBT archive was released to the public on 2012 October 1.

Currently not all data are archived. Back-ends developed externally are not included and data are taken off site by the P.I. All VLBI data are stored on disk and correlated elsewhere. Only pulsar folded data (e.g., pulsar timing data) exist in the GBT archive, even for NRAO back-ends due to the large data rates and volumes. This does not include meta-data which does exists in most cases, so the user can at least determine what has been observed. Pulsar search data from the Green Bank North Celestial Cap (GBNCC) survey is stored in CV but not accessible to users via the AAT. Unlike the VLA and ALMA, the GBT archive is not the only way that users gain access to their data. GBT users are responsible for performing the observations in most cases and thus they require real-time access to GBT data. Users have local GB computer accounts and process data on local disks. GBT data are currently archived on longer times scales (24 − 48 hr) compared to other NRAO telescopes. Since the GBT archive release about 40 GB/month of data have been downloaded. Data consist of either the raw FITS files or a "filled" single-dish FITS file (SDFITS).

3. Data Rates and Volumes and Storage Limitations

Current GBT data rates and volumes are modest, excluding pulsar data. The entire GBT archive is only 11 TB, producing an average data rate of 3 GB/day or 0.3 Mbs. This is expected to change as VEGAS becomes the primary back-end. Based on current proposals the expected data rate will be 1.4 PB/year or a constant rate of 360 Mbs.

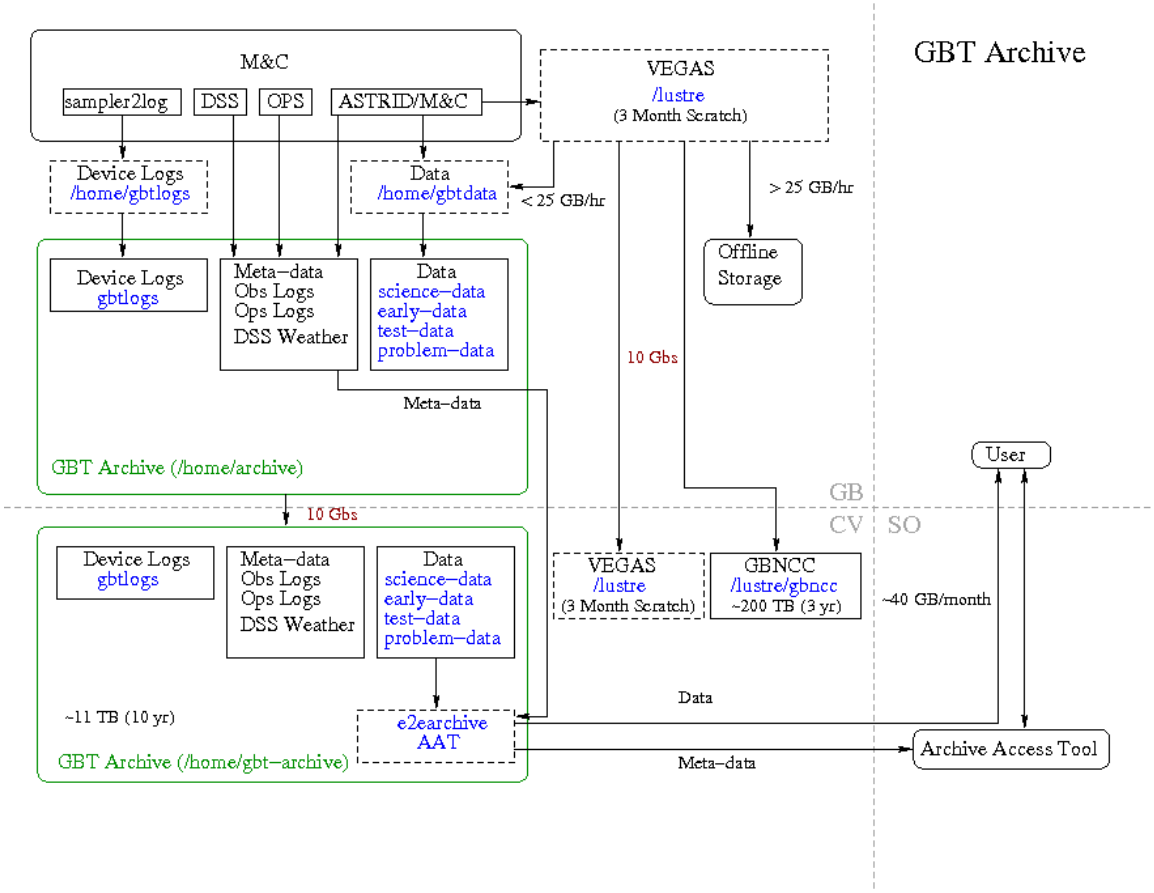
There are three constraints to the size of the GBT archive: power used for cooling, speed of the data link, and cost of spinning disk. Currently the limiting factor is the data link

with a rate of 45 Mbs. Therefore, we can only allow data to be archived with data rates < 25 GB/day. A constant data rate of 25 GB/day corresponds to ~9 TB per year or ~18 TB for two copies (GB and CV). When the new 10 Gbs link becomes operational around 2014 the limiting factor will be the cost of spinning disk. Based on our expected budget we estimate a data rate threshold of 25 GB/hr. Therefore, projects with data rates below this threshold will be automatically copied to the GBT archive. A constant rate of 25 GB/hr corresponds to ~ 220 TB per year or ~ 440 TB for two copies. Of course a constant rate of 25 GB/hr is unlikely as many projects will use much lower rates. We may refine this threshold rate in the future. Projects with data rates > 25 GB/hr will be temporarily stored on spinning disk for only 3 months (scratch) and then copied to non-spinning disk (offline) in GB. Only one copy of the offline data will exist. If funds are available we plan to copy the 3 month scratch data to CV as a backup copy.

Pulsar data have historically been treated differently than other data because of the large data sets. But VEGAS will change this paradigm. The plan is to eventually treat all data in the same way. This will be simplified if VEGAS becomes the standard pulsar back-end. This is assumed in Figure 1, where all large data sets will be produced by VEGAS and be stored on lustre. Currently, however, pulsar data are generated by GUPPI and stored on Beef. Folded pulsar data, much smaller than search pulsar data, are copied over to the luster file system to be archived. The pulsar GBNCC survey data are copied to CV for permanent storage but are not part of the archive.

4. GBT Archiving Processes

GBT archiving is achieved through cooperation between the computing and software divisions of the GBT. The computing division handles data storage for each observation and is responsible for the transfer of raw data files to the NRAO archive spinning disk area hosted in Charlottesville. For pulsar data observations in which the use of spinning disk is not feasible, the computing group is responsible for shipment of the disks to Charlottesville. The software group is responsible for mining raw data, generating metadata, storing metadata locally in Green Bank, and providing AAT access to the metadata. Table 1 lists processes comprising the overall GBT archiving system. Figure 1 depicts the GBT data products and the data flow between NRAO sites.



.

Figure 1: Diagram of the GBT Archive.

The light gray dashed lines indicate the location. The GBT archive is shown as the green colored box and resides in both GB and CV. It consists of data, meta-data, and ancillary data (e.g., weather data). Dashed boxes indicate data storage that is temporary. Data can be searched and accessed via the NRAO-wide archive access tool (AAT) located in SO.

GBT ARCHIVE PROCESS DOCUMENT

Process	Data Flow	Primary	Process	Verification	Time Scale
Data into GB Archive	GB: /home/gbtdata --> GB: /home/archive/science-data, /home/archive/test-data, /home/archive/problem-data	ChrisClark	Cron Job (except Problem Data	Email	Daily
Logs into GB Archive	GB: /home/gbtlogs --> GB: /home/archive/gbtlogs	ChrisClark	Manually	None (visual inspection)	3-6 months
Pulsar folded GUPPI data into GB Archive I	GB: /data1/beef --> GB: /lustre/pulsar/scratch/GUPPI2	PaulDemorest	Cron Job	Email	Daily
Pulsar folded GUPPI data into GB Archive II	GB: /lustre/pulsar/scratch/GUPPI2/science-data --> GB: /home/archive/science-data	ChrisClark	Cron Job	Email	Daily
VEGAS data into GB Archive	GB: /lustre/gbtdata --> GB: /home/archive/science-data, /home/archive/test-data, /home/archive/problem-data	ChrisClark	Cron Job	Email	Daily
Data from GB to CV	GB: /home/archive/science-data, /home/archive/test-data, /home/archive/problem-data, /home/archive/early-data --> CV: /home/gbt-	ChrisClark	Cron Job	Email	Daily

Process	Data Flow	Primary	Process	Verification	Time Scale
	archive/science-data, /home/gbt-archive/test-data, /home/archive/gbt-problem-data, /home/archive/gbt-early-data				
Databases from GB to CV	GB: GBT meta-data (mysql), Observing (myslq), Operator (dbase), DSS (postgres) --> CV: /home/gbt-archive/databases/gbt_metaArchive, /home/gbt-archive/databases/turtle, /home/gbt-archive/databases/Dbase, /home/gbt-archive/databases/Dss, /home/gbt-archive/databases/weather	ChrisClark	Cron Job	Email	Daily
Meta-data from GB to CV (for AAT)	GB: meta-data database (mysql) --> CV: /home/gbt-archive/AAT	MelindaMello	Cron Job	Email	Daily
Meta-data from CV to SO (for AAT)	CV: /home/gbt-archive/AAT --> SO: AAT meta-data database	JohnBenson	Data Analyst	Email	Daily

Table 1: GBT Archiving System Processes

5. General Metadata Archiving Process

Before 2011 the GBT did not archive any metadata and there was no interface between the AAT and the GBT. An offline archiver was developed which mined FITS files to obtain metadata for the 10+ years of observations. This offline archiver code is currently used as a daily cron job to archive metadata for new observations as well. This mining is done through FITS files. An online archiver has been developed which archives the metadata directly from ASTRID (observers interface) in real time. It does not access the FITS files at all, but obtains GBT metadata directly from the observing system. The online archiving mitigates some issues with the offline archiving such as correct Receiver determination and access to the lustre file system. It has been running in tandem with the offline archiver for many months and will be released soon. At that point the offline archiver will be relegated to regeneration of metadata for existing observations.

The metadata archive process is divided into several steps:

- OFFLine: GBT FITS files are mined for archive metadata. Metadata are stored in the gbt_metadata database.
- ONLine: archive data are mined from observations as scheduling blocks are executed within the GBT observing system and then stored in the gbt_metaArchive database.
- Conversion of metadata in the GBT database to AAT csv format. AAT csv files are writen to disk.
- Copy AAT csv files from GB to CV if Session FITS files are available in CV.
- The AAT runs a script daily that ingests the csv files into the AAT after which the data becomes available throught the AAT UI.

6. Detailed Metadata Archiving Process

The archive process generates metadata and populates the gbt_metadata database with all new GBT scans. The GBT scan metadata are harvested from the Scanlog, GO, IF and Backend FITS files in the offline mode. Nominally "new" scans are those in /home/gbtdata that were observed in the previous 24 hours. For the online version the same metadata are archived in real time. These metadata are defined in this document in the subsequent Data section.

The GBT database contains metadata for all scans run with the GBT but only science observations are archived to the AAT. Typically all science sessions run within the GBT are archived to the AAT, but not all, as some data are considered proprietary. Proprietary scans are those that contain data generated with user backends. These metadata are excluded from metadata csv files that are sent to the AAT.

Six csv files, A.K.A AAT tables, are produced and then ingested into the AAT. Documentation for each table can be found at http://www.aoc.nrao.edu/~jbenson/archivedocs/E2EArchiveDBTables. These csv files are placed on a disk in Charlottesville in /home /gbt-archive/AAT/. The format of these csv files were requested by the AAT developers and are most likely optimized for the

existing database scheme but it is not expected that changes to the database schema will result in changes to the format of these files.

There may be a delay of up to 48 hours for raw data to be rsynced from the Green Bank archive to the Charlottesville archive. Metadata in the GBT archive database are written to the AAT tables within 24 hours of the projects' raw data becoming available in the Charlottesville disk area but not before the raw data is available.

6.1. The GBT Metadata Database

Metadata for GBT scans are stored in the gbt_metaArchive database for offline archiving and the archive_metaArchive_online in the online processing. The structure of these databases is nearly identical. There is an ongoing effort to regenerate metadata for the GBT. When complete the gbt_metadata database will become the released metadata database and also the online archive GBT database.

There are 10 tables in the GBT metaArchive database. The archive is scan based. The main table is the scan table. Almost all other tables have relationships to it. Each relationship in the scan table to entries of other tables is through a foreign key relationship. A valid scan should have a reference to 1 entry in each other table.

A brief description of its table and its relationships are given below:
The project table contains data describing the project associated with a scan. A project record can be referenced by many scans and many sessions.

- ***The scan table*** contains metadata associated with a single GBT scan. A GBT scan is the smallest atomic unit in a GBT observation. Each scan should have 1 association with every other table in the database.
- ***The session table*** contains data describing the session and references a project table. A session may be referenced by many scans.
- ***The obsProcedure table*** contains data describing the observing intent of a scan. An obsProcedure may be referenced by 1 or more scans in 1 or more projects.
- ***The obsParameter table*** contains data describing the spectral windows contained within the scan. An obsParameter contains the GBT specific information for each window including the hardware and the relevant hardware configuration. An obsParameter record may be referenced by 1 or more scans.
- ***The coordinates table*** contains data describing the position observed in the scan. A coordinate may be referenced by 1 scan.
- ***The error table*** is a lookup table and is NOT dynamically updated. It is referenced by 1 or more scans in 1 or more projects.
- ***The history table*** contains data related to the GB to Socorro AAT table transmission. It records the file and date that scans from the GB metadata was archived to Socorro. A history may be referenced by 1 or more scans in 1 or more projects.

- ***The file table*** contains the file information related to 1 scan and is referenced by a single scan.

6.2. The Metadata

The table below describes each piece of metadata in the database and specifies the origination of the data product. Its intended audience is GBT developers and GBT scientists familiar with the GBT systems and data products. Most data in both the offline and online archiver versions originates from the same data source within the ASTRID or M&C system. Whenever this is the case, the *Online Origination* column in the table below contains the value "same as offline".

The original requirements document that defines the metadata are GBT_Memo_278 and can be viewed at the GBT archive wiki documentation page at
https://safe.nrao.edu/wiki/pub/GB/Knowledge/GBTMemos/GBT_Memo_278.pdf

Table Name	Field Name	Description	Offline Origination	Online Origination
Project	projectId	primary key	Auto Increment: Referenced by the scan and session tables	Same as offline
Project	name	Name of project	In most cases the project name is derived from the directory name where the metadata was originally archived from e.g. /home/gbtdata/PROJECT_NAME_session. There are several cases where the project name does not conform to any standard. For these older datasets the corrected project name is specified in a lookup table.	From the scan object

GBT ARCHIVE PROCESS DOCUMENT

Project	Propname	proposal name string	In most cases the proposal name is derived from the project name. For typical GBT projects the proposal name is equivalent to the project name with the "A" in AGBT" removed and a "/" added after the prefix GBT, and the "-" characters substituted for a "_". For VLBA and other projects the following regular expression is used : proj = re.findall(r'\b(\w+?)(\d+)(\w*)', fullname). There are several cases where the project name does not conform to any standard. For these older datasets the corrected project name is specified in a lookup table.	Derived from the project name within the GBT observing tool “ASTRID”
Session	sessionId	primary key	Auto Increment: Referenced by the scan table	Same as offline
Session	projectId	foreign key	Associates a project with the session	Same as offline
Session	name	the session name	In most cases the session name is derived from the directory name where the metadata was originally archived from e.g. /home/gbtdata/project_SESSION_NAME. There are several cases where	Derived from the session name within ASTRID

			the project name and session name does not conform to any standard. For these older datasets the corrected session name is specified in a lookup table.	
Session	Go FITS Version	String representing the era of the metadata as represented in the GO FITS files	Currently there are 3 version of FITS files represented, "1.0", "2.0" and "3.0". "3.0" is the most recent and most complete. Version "1.0" contains metadata that is least standard and more suspect. Meta data derived from GO FITS files taken before ASTRID are labeled with Version 1.O. Meta data derived from GO FITS files taken with ASTRID are labeled with "2.0". All other meta data are labeled "3.0".	FITSVER within Astrid
Session	MsgFlag	string describing a session level problem with the data	This string is intended to be used to indicate a serious problem with all the data within a session. An example of when a message might be added is when a system level problem affecting the pointing of the telescope is discovered AFTER several observations were run. This message string will be added post archival and will need to be communicated to the	Same as offline

			Socorro archiving staff through email or phone communication as there is no infrastructure to support this and it is expected to be rarely used.	
Session	MsgLevel	severity level of problem with session data	Integer value describing the severity of the problem with all the associated session data. This level needs to be a non null and non zero value if a MsgFlag is added to a session (see above entry). Higher values indicate more severe problems	Same as offline
Observer	observerId	primary key	Auto Increment: Referenced by scan table	Same as offline
Observer	name	observers name	Name of observer who ran the scan as read from GO FITS file	Observer name from within Astrid
File	fileId	primary key	Referenced by scan table	Same as offline
File	name	file path name	Directory path of the location of the file in Charlottesvile archive. Prior to the associated scan being archived, this string will indicate the directory path in GreenBank where the meta data was derived	Expected file path and name of GO FITS file.

			from.	
File	date	date of file creation	Read from the ScanLog.FITS	From the Scan object, it is the ScanCoordinators' parameters startTime
File	size	size of the directory path of the project	This value is derived from the size of the project dir when the metadata was generated. Metadata generated from new scans is generated from a GreenBank directory /home/gbtdata/PROJECT_SESSION. "Legacy" data, i.e. pre-gb archive processing was derived from /home/archive/science-data/PROJECT_SESSION	Generated when the metadata csv for NRAO archive files are generated. The online version will have the file size associated with the CV lustre file system where the raw data is archived
coordinates	coordinateId	primary key	reference by scan	Same as offline
coordinates	RA	not used	Originally added for online archiving purposes, but design changed. Should add RA from GO FITS file here, not required but nice	
coordinates	DEC	not used	Originally added for online archiving purposes, but design changed. Should add RA from GO FITS file	

			here, not required but nice	
coordinates	equinox	not used	Originally added for online archiving purposes, but design changed. CAN BE REMOVED	
coordinates	radesys	not used	Originally added for online archiving purposes, but design changed. CAN BE REMOVED	
coordinates	planetary	not used	Originally added for online archiving purposes, but design changed. CAN BE REMOVED	
coordinates	obsaz	az position in degrees	Computed using RAJ2000, DEC2000, date observed, and position of GBT	Same as offline
cordinates	obsel	el position in degrees	Computed using RAJ2000, DEC2000, date observed, and position of GBT	Same as offline
coordinates	RAJ2000	ra pos	Derived from the RA in the GO FITS file and is dependent on the coord system in use at the time of observation	Derived from the RA and DEC within Astrid and is dependent on the coord system in use at the time of observation

coordinates	DECJ2000	dec pos	Derived from the DEC in the GO FITS file and is dependent on the coord system in use at the time of observation	Derived from the RA and DEC within Astrid and is dependent on the coord system in use at the time of observation
obsProcedure	obsProcedureId	primary key	Auto Increment: Referenced by scan	Same as offline
obsProcedure	name	Name of observing procedure	Read from the GO FITS KEYWORD PROCNAME	Same os offline. Set by the Procedure object within Astrid
obsProcedure	type	Type of observing procedure	Read from the GO FITS keyword “PROCTYPE”. Observing "PROC" data are related to specific antenna movements	Same os offline Set by the ASTRID Procedure object within Astrid
obsProcedure	procscan	Additional scan intent information	Read from the Go FITS file Keyword “PROCSCAN”.	Same os offline Set by the ASTRID Procedure object
obsProcedure	obsType	Type of observation	Read from the GO FITS KEYWORD OBSTYPE. Values of "Spectroscopy" in the GO FITS file are changed to LINE in the meta data. Obs procedure relate to observationation goals, i.e line or continuum	Same os offline. Set within Astrid to "CONTINUUM", LINE" or "UNKNOWN" and is based on the value from the config tool
obsParameter	obsParamet	primary key	Referenced by scan	Same as offline

	erID			
obsParameter	backend	Backends in use	Read from the ScanLog.FITS file As such, support for backends that do not write FITS files is limited. However backends such as mustang and VLBA_DAR are derived from the configuration values within the GO header keywords	Derived from Scancoordinators device list at the start of the scan
obsParameter	receiver	The name of the receiver	The receiver values are derived from the receiver to backend connections in the IF FITS files. When more than 1 receiver to backend connection exists, the existence of a receiver FITS file with the appropriate time/date is used as the determining factor. If the appropriate FITS file does not exist then the GO history FITS keyword “HISTORY” config values are used to choose the receiver value. NoiseSource is not used unless it is the ONLY connected receiver	Obtained from Scan object
obsParameter	nchan	The number of channels	Associated with backend in use and may contain a list. The list is of length equal to the number of spectral	Derived from the config tools spectral window information which is used to generate

			windows observed. The value is derived by the backend and its associated FITS file when applicable For the DCR, Zpectrometer, VLBA_DAR and CCB26_40 backends, the values are always 1. For MUSTANG data mchan =64. For VEGAS data the NCHAN is the value in the VEGAS header NCHAN keyword . For Spectrometer it is derived from the KEYWORD value for NLAGS. For SpectralProcessor data the value is derived from the TDIM5 KEYWORD in the DATA table. For GUPPI the value is derived from the GO KEYWORD header config data. For SPIGOT (misnomer, since I believe spigot will be listed as spectrometer in metadata., the value will be based on the bandwidth,20 MHZ< 2048 or 1024 for higher bandwidths.	the ASTRID GO IFWindow Binary FITS table
obsParameter	bandwidth	The bandwidth associated with the backend in	Current limitation is that it does not support multiple bandwidths for multiple backends. It may contain a list. The list is of len equal to the	Derived from the config tools spectral window information which is used to generate the ASTRID GO

		the scan	number of spectral windows observed. The value is derived by the backend and its associated FITS file when applicable For the DCR the value is derived from the IF FITS file. For Mustang data the value is 18GHz. For VEGAS data the bandwidth is derived from the value of the BASE_BW VEGAS header KEYWORD. For Spectrometer it is derived from the value in the BANDWIDTH PORTtable. For SpectralProcessor data the value is derived from the BANDWIDTH value in the RECEIVER table For GUPPI and SPIGOT observations the value is derived from the GO KEYWORD header config data. For CCB_46 the value is set to 2.5 GHz	IFWindow Binary FITS table
obsParameter	velocity	the source velocity	Retrieved from the GO FITS file VELOCTY keyword. The data originates from the user configuration keyword values	Obtained from the Scan object
obsParameter	velociityDef	the velocity reference	Retrieved from the GO FITS file VELDEF KEWORD, The data originates from the user	Same as offline. Obtained from the

		frame	configuration keyword values	Scan object.
obsParameter	restfreq	the rest frequencies observed	For GO FITS files prior to version 2.8, the metadata are retrieved from the GO FITS files RESTFRQX, where X is a character between 0 and zz. A list of rest frequencies as specified by the user in the configuration keywords. This list may be augmented with derived values dependent on the number of spectral windows and the age of the FITS files. For scans using the spectrometer the rest frequency list is derived as follows: The initial value for the number of windows is retrieved from the number of samplers in the spectrometer NUMSAMP keyword in the SAMPLER FITS table. This value is must be divided by the number of feeds/beams in use and is itself derived from the IF FITS file. Since each beam supports 2 polarities, the derived value for number of spectral windows must be divided by 2. If cross products were in use then this value is divided again by 2. The number of windows is then used	**Derived from the config tools spectral window information which is used to generate the ASTRID GO IFWindow Binary FITS table

		to retrieve the appropriate RESTFRQX from the GO FITS header; Where X is the window designator. Currently only windows 1-8 are supported. If the number of spectral windows in the scan is greater than the number of RESTFRQX keywords in the GO FITS files, the GO HEADER configuration keywords are used to determine the remaining rest frequencies. If this information is not available in the GO FITS file, the rest frequencies are set to unknown for the remaining spectral windows. For scans using the spectral processor the rest frequency list is derived as follows: The number of sampler is derived from the len of the BANDWD column in the RECEIVER FITS table. This value is must be divided by the number of feeds/beams in use and is itself derived from the IF FITS file. Since each beam supports 2 polarities, the derived value for number of spectral windows must be divided by 2. If cross products were in use then this value is divided again by 2. This is the number of spectral

		windows. The number of windows is then used to retrieve the appropriate RESTFRQX from the GO FITS header; Where X is the window designator. Currently only windows 1-8 are supported. If the number of spectral windows in the scan is greater than the number of RESTFRQX keywords in the GO FITS files, the GO HEADER configuration keywords are used to determine the remaining rest frequencies. If this information is not available in the GO FITS file, the rest frequencies are set to unknown for the remaining spectral windows For scans using the DCR the rest frequency list is derived as follows: The number of samplers is retrieved from the derived NRCVRS primary FITS keyword co. This value is must be divided by the number of feeds/beams in use and is itself derived from the IF FITS file. Since each beam supports 2 polarities, the derived value for number of spectral windows must be divided by 2. If cross products were in use then this value is divided again by 2. This is the

		number of spectral windows. The number of windows is then used to retrieve the appropriate RESTFRQX from the GO FITS header; Where X is the window designator. Currently only windows 1-8 are supported. If the number of spectral windows in the scan is greater than the number of RESTFRQX keywords in the GO FITS files, the GO HEADER configuration keywords are used to determine the remaining rest frequencies. If this information is not available in the GO FITS file, the rest frequencies are set to unknown for the remaining spectral windows For scans using GUPPI, SPIGOT and VLBA_DAR backends the number of spectral windows is retrieved from the GO FITS header configuration value for nwin. For scans using CCB26_40 and VLBA project codes where the backend could not be determined: the number of spectral windows is always set to 1 For old spectral line data for which the GO FITS file does not contain a RESTFRQ keyword, the value of the SKYFREQ

			keyword in the GO FITS file is used. For GO FITS files generated with GO version>2.7 the restfreqs are taken from the spectral window information as provided by the config tool.	
obsParameter	poln	the polarization observed	Derived from the receiver and backend paths in the IF FITS file and the cross product mode of the observation. The internal representation of this is the traditional GBT strings representing pols e.g. XX, LL, YY, RR, YR	**Derived from the config tools spectral window information
obsParameter	poln_num	the polarization observed	Derived from the value of poln from a lookup table. This internal representation is required by the Socorro Archive	Same as offline
obsParameter	mode	The mode string representing the mode of the backend	Backends supporting mode are GUPPI, VEGAS, Spectrometer and SpectralProcessor. Modes are obtained through the backend FITS files.	NA. currently not supported in online version

obsParameter	rec_band	receiver band	Derived from the IF FITS file in conjunction with the backend in use, the Receiver FITS files if applicable and the receiver specified in the GO header configuration values. If only 1 receiver is listed in the IF FITS file then that receiver is used. However in many cases, multiple receivers are listed within the FITS file. In these cases, a receiver FITS file is searched for whose name (date time tag) matching the scans date and time. If one is found this receiver was the receiver in use. Most receivers at the GBT do not create scan by scan FITS files, so this check rarely solves the dilemma. The receiver in the configuration is used as the last resort in the receiver determination	From Astrid derived from the Receiver as reported by the scan object within Astrid
Error	errorId	primary key	referenced by scan	
Error	errorMsg	Description of the problem	A static string	Same as offline
Error	severity	Severity level	Static. Currently all scan level errors are considered low priority errors and their value is equal to 1	Same as offline

History	historyId	primary key	Auto Increment: referenced by scan	Same as offline
History	archivalDate	The date the AAT csv file generated	internally generated by the archive process	Same as offline
History	aatFilename	The directory path and name of the AAT csv file	internally generated by the archive process	Same as offline
History	version	The GBT archiving software version number that the AAT file was generated with	internally generated by the archive process	Same as offline
Scan	scanId	primary key		
Scan	number	the M&C scan number	The value is retrieved from the GO FITS file and contains the scan designation number as set in the M&C system. Duplicate entries are possible but are flagged with this error condition. Duplicate entries are only allowed for scans that have different observing dates and times, i.e scan number has been reused but contains different data	Same as offline. From the Scan object

			than the previous archived scan	
Scan	object	the name of the object observed	The value is retrieved from the GO FITS	Same os offline file. Retrieved from Scan object within Astrid
Scan	obsIdentifier	description of observation	The value is retrieved from the GO FITS file and contains the value of the object as described by M&C ScanCoordinators' obsid parameter (as of 10/25/2012. An MR is in process which will make the value and use of OBSID to be deprecated. (see MR 8Q312 for additional information)	Same os offline. Retrieved from Scan object within Astrid
Scan	projectID	fkey for the project table	project associated with this Scan	Same as offline
Scan	observerID	fkey for the observer table	observer associated with this Scan	Same as offline
Scan	obsProcedureId	fkey for the obsProcedure	obsProcedure perfumed associated with this Scan	Same as offline
Scan	obsParameterId	fkey for the obsParameter table	obsParameter associated with this Scan	Same as offline

Scan	coordinateId	fkey for the coordinates table	coordinate associated with this Scan	Same as offline
Scan	errorId	fkey for the error table	Error, if applicable, associated with this Scan. a 0 indicates no error	Same as offline
Scan	historyId	fkey for the history	archive history associated with this Scan	Same as offline
Scan	fileId	fkey for the file table	file associated with this Scan	Same as offline
Scan	sessionId	fkey for the session table	session associated with this Scan	Same as offline
Scan	dateObserved	the date of the observation	Retrieved from the GO FITS file and contains the date and start time of the scan	Same as offline. Retrieved from the Scan object
Scan	integrationTime	the integration time	The value is derived by the backend and its associated FITS file when applicable For VEGAS it is read in from the value of the DURATION keyword header in the DATA table For SpectralProcessor data it is read in from the value of the INTTIME keyword header in the DATA	Derived from the config tools spectral window information

			table For DCR data it is read in from the value of the DURATION keyword header in the DATA table For CCB data it is read in from the value of the INTEGRAT keyword in the primary header For GUPPI and VLBA_DAR and Zpectrometer data it is read in from the value of the GO FITS History config keyword tint If no backend is found but it is a VLBA project code it is set to 0. For GO FITS file version >= 2.8 the value is retrieved from the config tools spectral window table.	
Scan	scanlength	the length of the scan in seconds	Derived from the start and stop times in the Scanlog.FITS	Same as offline. Derived from Scan object
Scan	archived	Integer describing whether the scan is archive able and whether it already has been archived	Internally generated by archive process. A value of -1 indicates that this scan should not be archived to the AAT because the data was taken with a user backend. A value of -2 indicates that this scan should not be archived to the AAT because it is test data. A value of 1 indicates that the scan	Same as offline

			has been archived. A value of 0 means it is achievable but is not presently archived. This typically occurs when the FITS data has not been transferred to the CV archive from GB	
Scan	Notes	string that contains dynamic error messages for the online archiver	Not used by offline archiver	This field is populated whenever the online archiver determines that the telescope as described by the config tool differs from the telescopes actual configuration. This might result in inaccurate metadata and is flagged with the ERRORId=8. The differences between the actual and expected telescope configuration is listed in this field

7. Desired Improvements

The following 3 items are suggested improvements. These items should require a relatively small amount of effort and should result in improvements in data quality, process improvement and expanded functionality.

- As described in a previous section, metadata for the VEGAS data may be available but the raw FITS files may be too large for mirroring in CV. In these cases the metadata should be augmented with this information and a description to the user should be provided within the AAT UI that

describes the process or contact person from which the user may obtain the raw FITS data products.

- The intermediate step of generating the AAT csv files should be removed. The transfer of data should be from the GBT database to the Socorro database.
- The addition of Level 1 (as defined by the AAT-PPI Functional Requirements Document) data products are not supported in the current infrastructure within the GBT database or in the GBT-AAT interface. There are already level 1 GBT data products available on disk in GB and there is a desire to make these available to the user community through the AAT. It is expected that the number of GBT Level 1 data products will increase as the GBT pipeline matures.

8. Document References

The original GBT Archiving documentaiton from which the contents of this document is wholely derived are at: https://staff.nrao.edu/wiki/bin/view/OSO/OSOGBTArchiveDocs