Interactive Dashboard of Green Bank Telescope Antenna Data

E. Lowe[1] and T. Chamberlin[2]

***Abstract*— With the vast amount of collected Antenna Data from the Green Bank Telescope (GBT), the Interactive Dashboard of GBT Antenna Data provides an interface to visualize sky positions. The dashboard allows users to manipulate the visualizations based on various factors such as date of observation or receiver and backend type. This dashboard is another tool that users can use to look at archival observations or observations based on certain criteria.**

I. INTRODUCTION

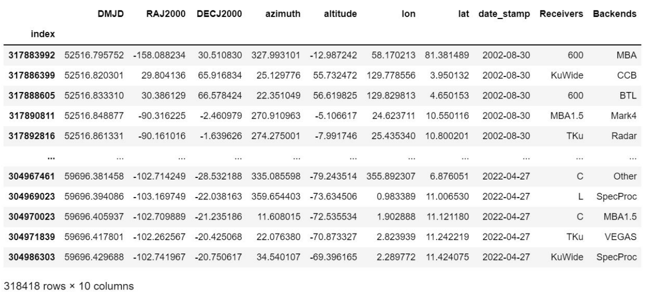
Data collected from the GBT is stored in the GBT Archive. While there are various methods for querying this Archive, there is currently no way of exploring its data visually. Here, we present an interactive dashboard that links archival GBT Antenna positions to GBT Archive data. This dashboard allows users to search based on sky positions using an interactive map, as opposed to manually typing in sky coordinates or object names.

II. GETTING THE DATA

A. Initial Data Sampling

The periods in which the GBT is collecting data are referred to as "scans". During each scan, the Antenna Manager writes the antenna position on the sky at 10Hz to an "Antenna FITS File". The Antenna Manager writes one file per scan. The FITS files from around the last ∼20 years were gathered, sampled at 1Hz, and stacked. This means one antenna position was taken from the file at every one second for the past ∼20 years. The resultant data consisted of ∼300M antenna positions. For development, test datasets were created by sampling down to 30M, 300k, 30K, and 3K positions. The 300K and 3K subsample were primarily used throughout the process of creating the dashboard.

B. Viewing the Data Information



index	DMJD	RAJ2000	DECJ2000	azimuth	altitude	lon	lat	date_stamp	Receivers	Backends
317883992	52516.795752	-158.088234	30.510830	327.993101	-12.987242	58.170213	81.381489	2002-08-30	600	MBA
317886399	52516.820301	29.804136	65.916834	25.129776	55.732472	129.778556	3.950132	2002-08-30	KuWide	CCB
317888605	52516.833310	30.386129	66.578424	22.351049	56.619825	129.829813	4.650153	2002-08-30	600	BTL
317890811	52516.848877	-90.316225	-2.460979	270.910963	-5.106617	24.623711	10.550116	2002-08-30	MBA1.5	Mark4
317892816	52516.861331	-90.161016	-1.639626	274.275001	-7.991746	25.435340	10.800201	2002-08-30	TKu	Radar
...	...	...	...	...	...	...	...	...	...	...
304967461	59696.381458	-102.714249	-28.532188	335.085598	-79.243514	355.892307	6.876051	2022-04-27	C	Other
304969023	59696.394086	-103.169749	-22.038163	359.654403	-73.634506	0.983389	11.006530	2022-04-27	L	SpecProc
304970023	59696.405937	-102.709889	-21.235186	11.608015	-72.535534	1.902888	11.121180	2022-04-27	C	MBA1.5
304971839	59696.417801	-102.262567	-20.425068	22.076380	-70.873327	2.823939	11.242219	2022-04-27	TKu	VEGAS
304986303	59696.429688	-102.741967	-20.750617	34.540107	-69.396165	2.289772	11.424075	2022-04-27	KuWide	SpecProc

318418 rows × 10 columns

Fig. 1. Table View of 300K Dataset

The initial data that was given for each antenna position was the Decimal Modified Julian Date (DMJD), right ascension (RAJ2000), declination (DECJ2000), azimuth, and altitude. Using the RAJ2000 and DECJ2000, the longitude (lon) and latitude (lat) of the positions were calculated. In addition, the DMJD was converted into a date stamp (date_stamp). For exercise and implementation, information from the receivers and backends were randomly assigned to each antenna positions since they were not originally provided.

III. PLOTTING THE DATA

The antenna position data can be viewed as three different sets of data. The first set is the RAJ2000 and DECJ2000 positions (J2000 Coordinate System), the second set is the longitude and latitude positions (Galactic Coordinate System), and lastly, the azimuth and altitude positions. Using the Python library GeoViews, all sets were individually plotted to create three projections [1]. The J2000 and Galactic coordinate systems data were projected on a Mollweide map. For the azimuth/altitude data, they were projected on an Azimuthal Equidistant map.

A. Problem of Overplotting

With the immense amount of data, overplotting becomes an inevitable obstacle. Figure 2 shows the overplotted projections. Overplotting causes the points not to be distinct anymore because there are numerous points clumped together or even plotted on top of each other. The distribution of the points cannot be clearly determined when overplotting occurs. Therefore, limited information can be derived from these plots.

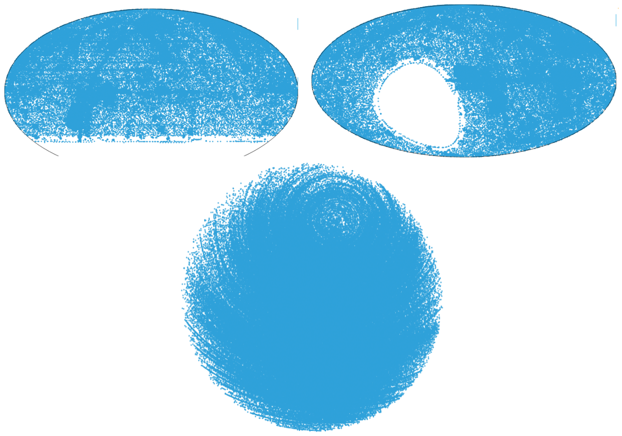


Fig. 2. Initial Projections: J2000 (Top Left), Galactic (Top Right), Azimuth/Altitude (Bottom)

[1] West Virginia University Institute of Technology
[2] Green Bank Observatory

B. *Using Datashader as a Solution*

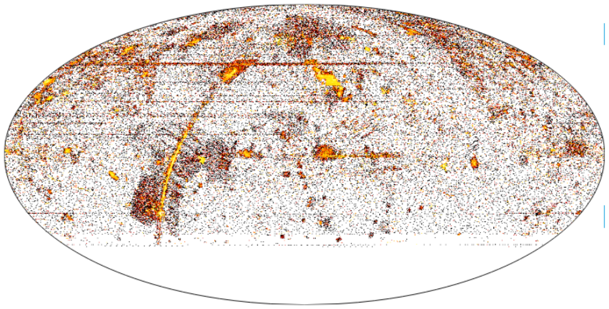
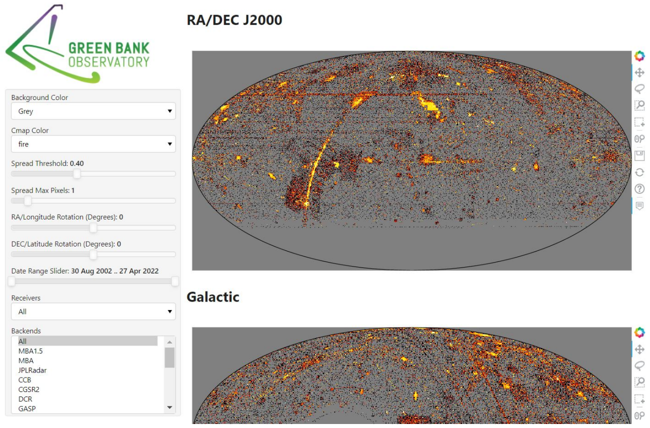


Fig. 3. J2000 Projection after using Datashader

The python library, Datashader [1], can be used to solve the problem of overplotting. Datashader shades the data based on the density of the points. Areas with a higher number of points are shaded bright yellow and those with lower numbers are shaded a red. Utilizing Datashader, the dispersion of data is better seen in the projections. Examining Figure 3, we can see that most observations in this dataset were taken along the galactic equator. This information could not be concluded with the previous projections shown in Figure 2.

IV. CREATING THE DASHBOARD



GREEN BANK OBSERVATORY

Background Color: Grey

Cmap Color: fire

Spread Threshold: 0.40

Spread Max Pixels: 1

RA/Longitude Rotation (Degrees): 0

DEC/Latitude Rotation (Degrees): 0

Date Range Slider: 30 Aug 2002 .. 27 Apr 2022

Receivers: All

Backends: All

- All
- MBA1.5
- MBA
- JPLRadar
- CCB
- CGSR2
- DCR
- GASP

Fig. 4. Initial View of Dashboard

A. *Projections*

Previous projections were grouped together in a column as shown in Figure 5. This column was placed to the right of the dashboard. This allows users to see all three projections at the same time.

B. *Widgets*

The dashboard has 9 widgets that the user can use to filter the projections. Each widget was created using the python library Panel [2]. Figure 6 shows the widgets' name, type, and choices.

In the first two widgets, users are able to interact with the background color and color map (cmap) widgets. Allowing the user to change the colors of both these properties help them view data in a way that is preferred and most accessible to them. For example, if users want to view the data in a different color scheme they are able to change the cmap or if they want a better contrast between the points and background, the background color can be changed.

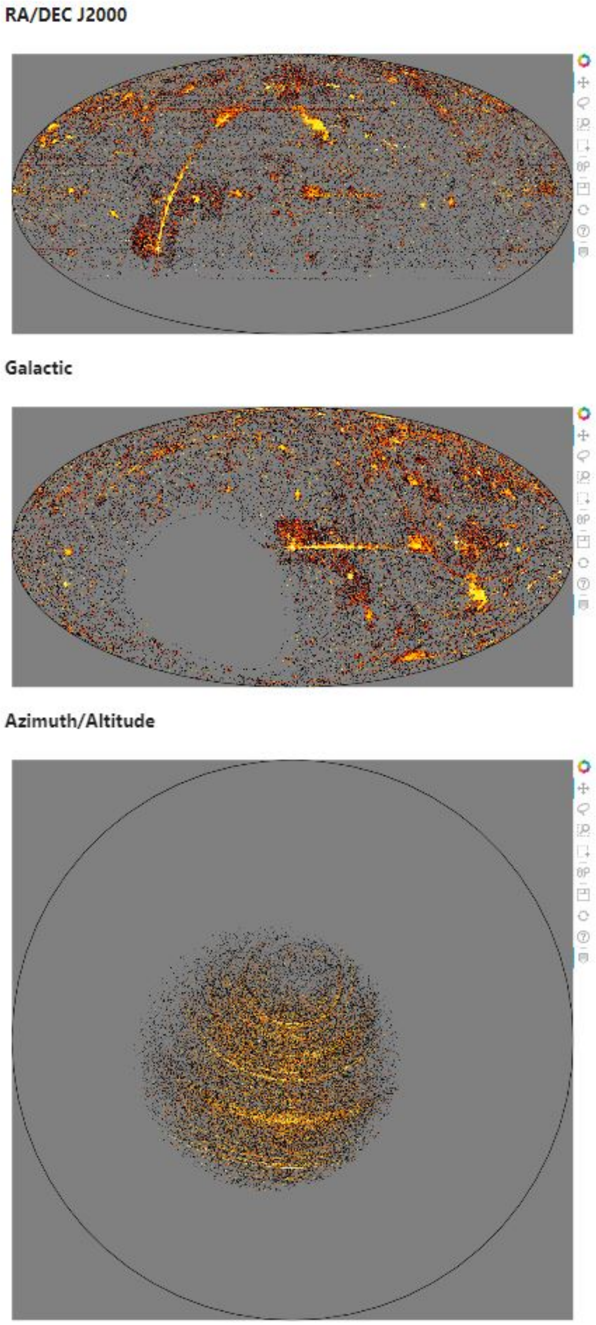
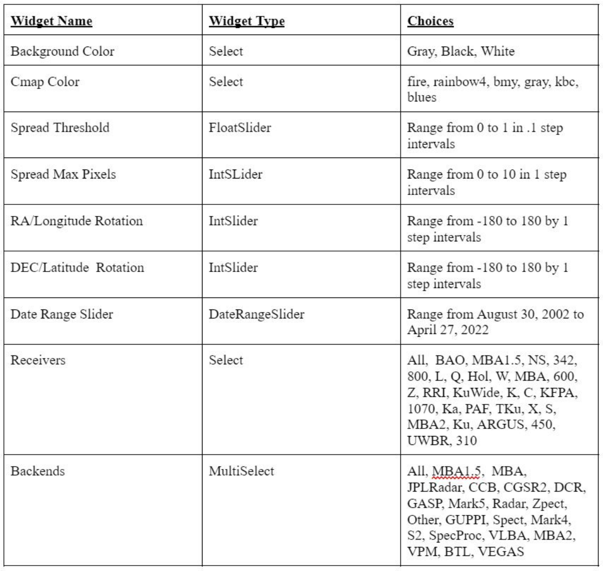


Fig. 5. Column of all Projections

Users can also change the spread threshold and spread max pixels. The spread threshold determines how far to spread the points from each other and the spread max pixels is the number of maximum pixels to spread on all sides. Next, the user can rotate the RA/lon and the DEC/lat of the projections. Rotating projections provides the ability for



Widget Name	Widget Type	Choices
Background Color	Select	Gray, Black, White
Cmap Color	Select	fire, rainbow4, bmy, gray, kbc, blues
Spread Threshold	FloatSlider	Range from 0 to 1 in .1 step intervals
Spread Max Pixels	IntSLider	Range from 0 to 10 in 1 step intervals
RA/Longitude Rotation	IntSlider	Range from -180 to 180 by 1 step intervals
DEC/Latitude Rotation	IntSlider	Range from -180 to 180 by 1 step intervals
Date Range Slider	DateRangeSlider	Range from August 30, 2002 to April 27, 2022
Receivers	Select	All, BAO, MBA1.5, NS, 342, 800, L, Q, Hol, W, MBA, 600, Z, RRI, KuWide, K, C, KFPA, 1070, Ka, PAF, TKu, X, S, MBA2, Ku, ARGUS, 450, UWBR, 310
Backends	MultiSelect	All, MBA1.5, MBA, JPLRadar, CCB, CGSR2, DCR, GASP, Mark5, Radar, Zpect, Other, GUPPI, Spect, Mark4, S2, SpecProc, VLBA, MBA2, VPM, BTL, VEGAS

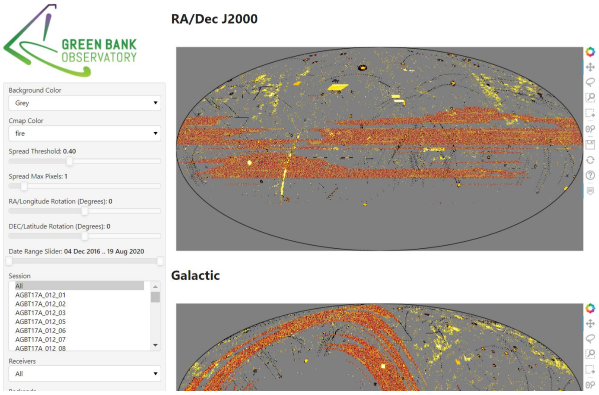
Fig. 6. Information of the Widgets

the user to view the points in different angles than how the points are normally projected on the map. Users can change what points are projected based on date with the Date Range Slider. Lastly, users can filter the projections based on receiver and backends used for the observation.

The type for each widget was selected based on how efficient accessing the data would be. For example, the backend receiver is a MultiSelect widget to allow the users to select multiple backends that go to one receiver. They now have the option to see one receiver and multiple backends or one receiver and one backend. All the widgets provide a seamless way to interact with the dashboard.

C. Session Widget

Using another subset of data, ~2M antenna positions from projects AGBT17A and AGBT17B, a session widget is able to be created. Since the session information was given in this dataset, it can now be a filter for the projections. Users can view the observations based on the session they were taken in. Figure 7 shows the session widget with in the dashboard.

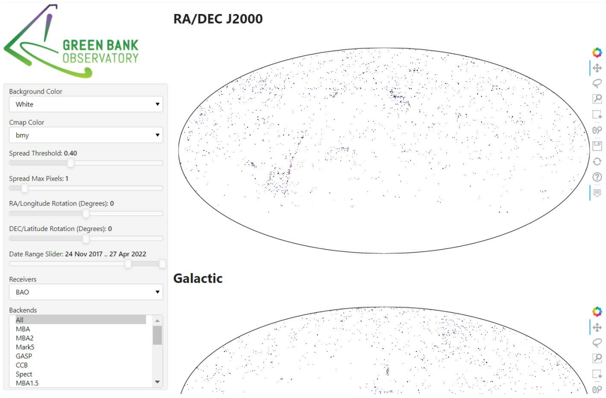


Background Color: Grey
Cmap Color: fire
Spread Threshold: 0.40
Spread Max Pixels: 1
RA/Longitude Rotation (Degrees): 0
DEC/Latitude Rotation (Degrees): 0
Date Range Slider: 04 Dec 2016 .. 19 Aug 2020
Session: All, AGBT17A_012_01, AGBT17A_012_02, AGBT17A_012_03, AGBT17A_012_05, AGBT17A_012_06, AGBT17A_012_07, AGBT17A_012_08
Receivers: All

Fig. 7. Session Widget with 2M Dataset

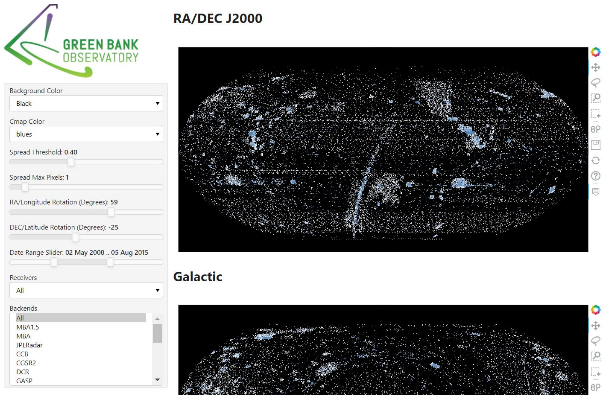
V. INTERACTING WITH THE DASHBOARD

The first way that a user can interact with the dashboard is using the widgets. When a widget's value is changed, the projections are changed to show only the points that meet the current criteria selected. This allows the user to only display the data that they want to view. Figure 8 and Figure 9 are examples of how the projections can be filtered.



Background Color: White
Cmap Color: bmy
Spread Threshold: 0.40
Spread Max Pixels: 1
RA/Longitude Rotation (Degrees): 0
DEC/Latitude Rotation (Degrees): 0
Date Range Slider: 24 Nov 2017 .. 27 Apr 2022
Receivers: BAO
Backends: All, MBA, MBA2, Mark5, GASP, CCB, Spect, MBA1.5

Fig. 8. Projections filtered with Background Color: White, Cmap Color: bmy, Date Range: November 24, 2017-April 27, 2022, and Receiver: BAO



Background Color: Black
Cmap Color: blues
Spread Threshold: 0.40
Spread Max Pixels: 1
RA/Longitude Rotation (Degrees): 59
DEC/Latitude Rotation (Degrees): -25
Date Range Slider: 02 May 2008 .. 05 Aug 2015
Receivers: All
Backends: All, MBA1.5, MBA, JPLRadar, CCB, CGSR2, DCR, GASP

Fig. 9. Projections filtered with Background Color: Black, Cmap Color: Blues, RA/Longitude Rotated 59 degrees, DEC/Latitude Rotated -25 degrees, and Date Rane: May 2, 2008 - August 5, 2017

The side bar of each projection gives the users the ability to pan, box zoom, wheel zoom, and save a photo of the projections. Users are also able to use a hover tool over the projections. Currently, the hover tool shows the projected coordinate system numbers for the values of the positions of the points and not the correct coordinate system. The issues with this tool will be addressed in section VIII.

VI. OTHER FEATURES

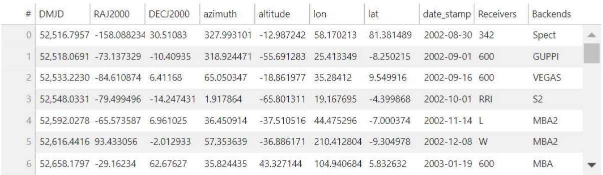
Another issue that occurs when using Datashader is that it returns a rasterized image of the data, and therefore the metadata of the points are lost (e.g. coordinates of the antenna positions in their original coordinate system). With this metadata lost, certain features are not possible on the shaded data because the actual points are not being interacted with anymore. When using the "raw" points shown in Figure

2, there are other features that are able to be used to interact with these unshaded points.

For performance purposes, the projections in subsections A and B are using a 3K subset of data to demonstrate the features.

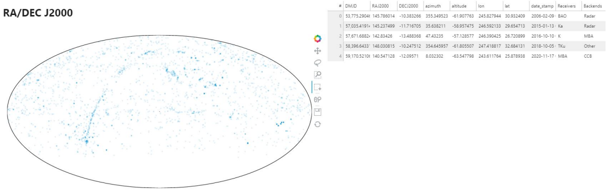
A. *Link Selection with Table*

The Link Selection features links the points on the projection to a table. The Selection table shown in Figure 10 is initially populated with all the data shown in Figure 1. The table can only be linked to one projection at a time– namely, the J2000 projection. When a user selects the box tool to highlight an area of points on the projection, these highlighted points' data will be displayed on the table. Figure 11 shows 4 highlighted points' information in the table. This tool enables users to extract information from the exact points desired.



#	DMJD	RAJ2000	DECJ2000	azimuth	altitude	lon	lat	date_stamp	Receivers	Backends
0	52,516.7957	-158.088234	30.51083	327.993101	-12.987242	58.170213	81.381489	2002-08-30	342	Spect
1	52,518.0691	-73.137329	-10.40935	318.924471	-55.691283	25.413349	-8.250215	2002-09-01	600	GUPPI
2	52,533.2230	-84.610874	6.41168	65.050347	-18.861977	35.28412	9.549916	2002-09-16	600	VEGAS
3	52,548.0331	-79.499496	-14.247431	1.917864	-65.801311	19.167695	-4.399868	2002-10-01	RRI	S2
4	52,592.0278	-65.573587	6.961025	36.450914	-37.510516	44.475296	-7.000374	2002-11-14	L	MBA2
5	52,616.4416	93.433056	-2.012933	57.353639	-36.886171	210.412804	-9.304978	2002-12-08	W	MBA2
6	52,658.1797	-29.16234	62.67627	35.824435	43.327144	104.940684	5.832632	2003-01-19	600	MBA

Fig. 10. Selection Table



#	DMJD	RAJ2000	DECJ2000	azimuth	altitude	lon	lat	date_stamp	Receivers	Backends
0	53,775.2904	145.786014	-10.383266	355.349523	-61.907763	245.827944	30.932409	2006-02-09	BAO	Radar
1	57,035.4191	145.237499	-11.716705	35.638211	-58.957475	246.592133	29.654713	2015-01-13	Ka	Radar
2	57,671.6882	142.83426	-13.488368	47.43235	-57.128577	246.390425	26.720899	2016-10-10	K	MBA
3	58,396.6433	148.030815	-10.247512	354.645957	-61.805507	247.418817	32.684131	2018-10-05	TKu	Other
4	59,170.5210	140.547128	-12.09571	8.032302	-63.547798	243.611764	25.878938	2020-11-17	MBA	CCB

Fig. 11. J2000 Projection linked to Selection Table

B. *Datalink among all Projections*

Another feature that is able to be used with the "raw" points is linking all three projections together via a Datalink. With this Datalink, the user is able to use the lasso tool to highlight points on one projection and the corresponding points are highlighted on the other projections as shown in Figure 12. The Datalink helps visualize points located across all three projections.

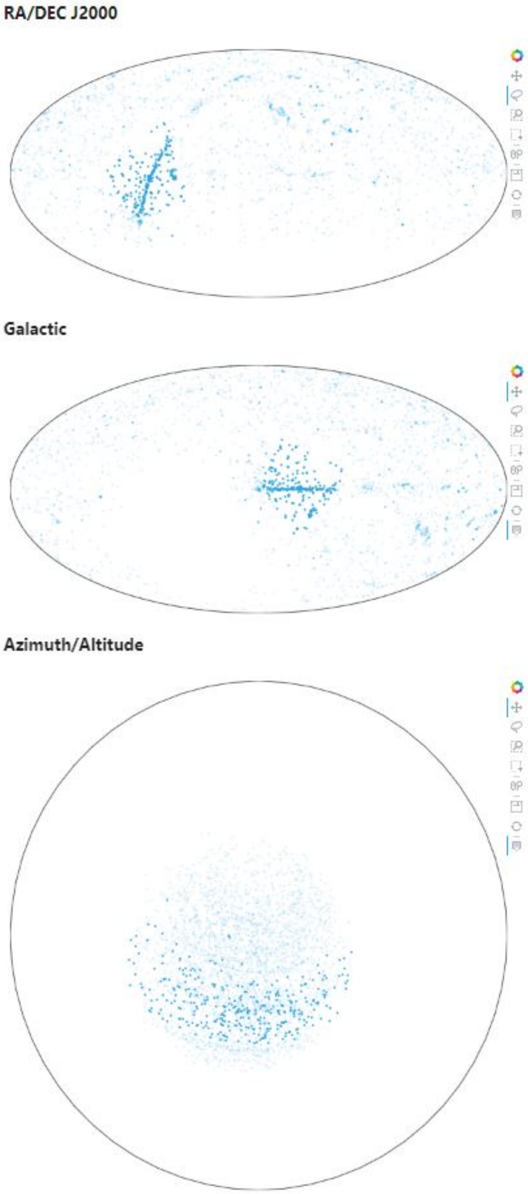


Fig. 12. Datalink across all Projections

C. *Quadmesh Hover Tool*

One feature that is able to be used with both shaded and unshaded projection is the Quadmesh Hover Tool. When using this tool a quadmesh, a 2D rectangular grid, is overlaid on the projections. Each square is called a bin of the quadmesh, the count of points in the bin is calculated. So, when using the Hover Tool, it will highlight the bin and show the amount of points that are currently in the bin. Sometimes it is hard to see how many points are currently being displayed and so the Quadmesh Hover tools gives users the exact number of points that they are hovering over. In addition, when the user zooms into the projections, the quadmesh is resized and will show the accurate bin sizes for the zoomed-in projection.

D. *Combining Points with Datashaded Imagery*

As a final feature, we demonstrate proof-of-concept combining both Points and Datashaded imagery. To maintain performance, a maximum of 1,000 points are randomly sampled from those within the current view. Looking at the initial projection, all the points will not appear. As the user zooms in, the points that were disregarded in the down sampling will eventually be shown once the total number of points in the view drops below 1,000.

The purpose of this feature was to overlay invisible decimated points over the shaded points and have the user interact with the invisible points only. The decimated points still holds their metadata and the features of the Linkselection

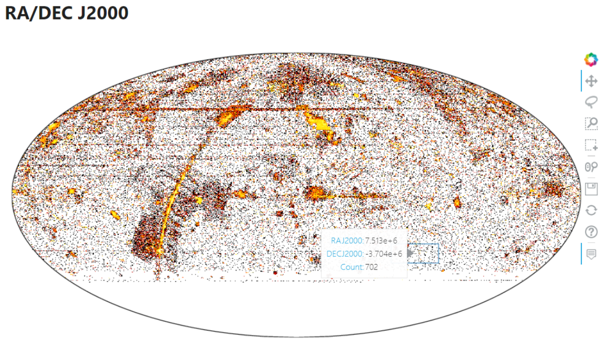


Fig. 13. Quadmesh Hover Tool

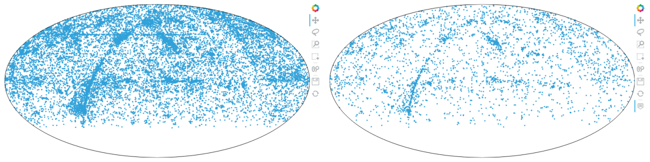


Fig. 14. Normal 30K Projection (Left) versus Decimated 30K Projection (Right)

and Datalink therefore works with these points. Overlaying the points would allow the user to see the shaded data but interact with the decimated points only. These decimated points could potentially be linked to an external source, as well. If a user zooms in and finds the observation they are looking for, they could click on the point on the map and be brought to the GBT Archive information about that observation. This feature was not able to get started in this project's time frame.

VII. Future Work

There are various features of the dashboard that work separately, but ideally need to work together. The features of the Link Selection and Datalink could potentially work if the decimated points feature was fully executed. This means figuring out how to let the user only interact with the decimated points so the features that were only capable with the "raw" points are able to appear with the shaded data. If this works, this would allow us a way to link points to the external data source of the GBT Archive.

Another feature that will be worked on is getting more information in the Selection Table. It would be helpful if data such as the Script Name, Observer, Operator, etc is shown in the Selection table as well. This could provide users more context on what they are looking at.

In addition, getting the hover tool to show the correct location of points versus the projected coordinate system numbers is a quirk that needs to be fixed. The projected coordinate system doesn't provide any valuable information. Based on the projection that is being used, the location of the points are lost. Only PlateCarree or GOOGLE_MERCATOR projections show the correct locations of the points. Seeing why certain projections per serve the points location is the next step of fixing the hover tool.

Lastly, getting everything to work together with the ~300M antenna positions. As of right now, using that many points breaks the application. The application is not able to render that many points at once. During this project, the subset of ~2M was the largest dataset that worked with the dashboard.

VIII. Python Libraries Used

The five main python libraries used for this project were Holoviews [4], Geoviews, Bokeh [5], Panel, and Datashader.

Holoviews helps visualize data in Jupyter Notebook. Geoviews is an extension of Holoviews and it was used to create all the projections. Bokeh is a backend for Holoviews and it does the plotting in the web browser. Panel is what allowed the creation of the interactive dashboard and widgets. Datashader took the discrete points and made rasterize images for the projections.

IX. Conclusions

The overall goal of this project was to give users a tool that can help visualize sky positions data taken by the GBT and to link these positions to more information about the observations. The dashboard allows users to filter the antenna data by date range, receiver, backend, or even by session. These filters help users fine-tune the observations that they want to look at.

The linking of antenna positions to more information in the dashboard was not completed. The project almost got to the point of almost being ready to link the points to an external source.

Even though the project did not complete the end goal, it still serves a valuable purpose of helping users find the observations in an interactive dashboard. Eventually, people will be able to navigate through the GBT Archive via this dashboard.

Acknowledgment

E. Lowe thanks T. Chamberlin for his guidance and support throughout this project. In addition, E. Lowe thanks W. Armentrout and B. Gregory for their encouragement during the 2022 NRAO/GBO REU program. The Green Bank Observatory is a facility of the National Science Foundation operated under cooperative agreement by Associated Universities, Inc.

References

[1] Geoviews: Python library for visualizing geographical, meteorological, and oceanographic datasets URL https://geoviews.org/.

[2] James A. Bednar, Jim Crist, Joseph Cottam, and Peter Wang (2016). "Datashader: Revealing the Structure of Genuinely Big Data", 15th Python in Science Conference (SciPy 2016).

[3] Panel: Python library that lets you create custom interactive web apps and dashboards by connecting user-defined widgets to plots, images, tables, or text URL https://panel.holoviz.org/.

[4] HoloViews: Python library for data analysis and visualization URL https://holoviews.org/.

[5] Bokeh Development Team (2018). Bokeh: Python library for interactive visualization URL http://www.bokeh.pydata.org.