

# Adding Relevant Annotations to the Green Bank Telescope's Radio Frequency Interference Graphics User Interface

## Document Management Information

- Author: Alex Daniel Seidman
- Revision Number: 1.0
- Revision Release Date: 2022-08-29
- Purpose of Revision: Final Draft.
- Scope of Revision: All sections of document.

## Purpose and Scope

This document details the recent development, addition, and implementation, of clickable annotations to the Green Bank Telescope's Radio Frequency Interference Graphics User Interface (gbt-rfi-gui, or simply the GUI going forward) that displays key information for printed the RFI.

## Architecture Background

### Problem Background

Radio Frequency Interference (RFI) is the bane of radio astronomy. Like how a photobomber can interrupt a perfectly good picture, signals and transmissions from countless different sources can interfere with a radio astronomer's observations, causing data to be inaccurate, or often near unreadable. This can include signals from satellites, radio transmissions, TV broadcasts, or even phone connections. All of which can be disastrous to astronomers and their work.

To help prevent this, here at Green Bank we have a Graphics User Interface (GUI) that lets you catalog previous instances of RFI detected by our main telescope. Users can specify which receiver band they would like to focus on, a start and end radio frequency, and a specific time range—to which the GUI will print out a graph of all the RFI detected based on the set parameters. However, it does not display any *contextual* information—that is, you cannot tell just from the graph what the cause of the RFI is. This is a problem for users of varying

backgrounds, as it makes using the GUI for actual data analysis unnecessarily harder, when it really doesn't need to be. Enter, this project: adding information-rich annotations into the GUI itself.

## **Goals and Context**

Our main goal with this project is to make information for identifying RFI sources simple as possible and done within the GUI itself. Many people can use and will want to use this GUI in the future, but not all the users will be able to interpret the graphs or have easy access to information that will allow them to interpret the graphs. The aim of this project is to, at the bare minimum, allow for anyone to use the GUI and get information as to the source of the RFI being shown.

## **Solution Background**

### **Approaches**

Throughout working on this project two different systems were considered and attempted: a "hover method" and a "clicking method". The only difference on the user's end would be how they interact with the graph, either by hovering over the graph to display the annotations, or by clicking at the area of interest. In the end, we determined that clicking would be the better of the two options, as it would lead to the information only being seen when the user wants it to be seen, reducing any possible visual distractions.

### **Coverage**

There are two groups that were considered while adding in the annotation feature—general users, and general programmers. The purpose of the annotations is to help both veterans and newcomers to astronomy alike, giving a way for anyone to instantly get information on the RFI that the GBT records. However, consideration for future developers of this program was also taken into consideration. Since more additions will be made without my supervision, making sure that the code is condense, understandable, and easy to add on to was a large goal as well, so that future programmers can improve the usability of the GUI.

Code View

User End

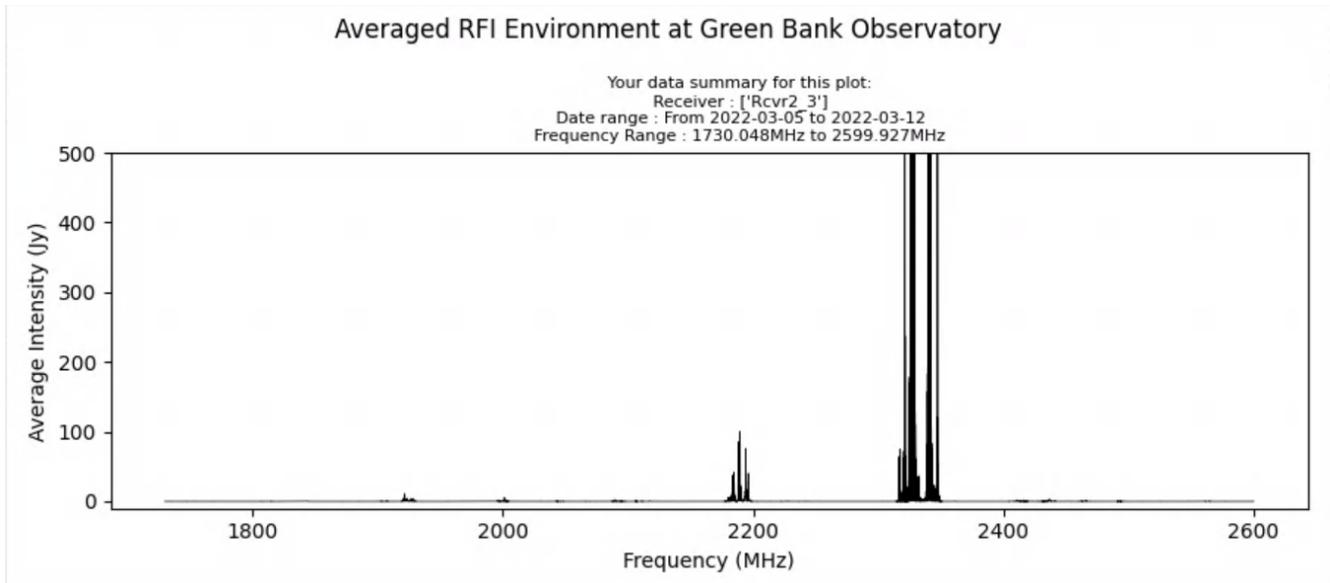


Figure (A)

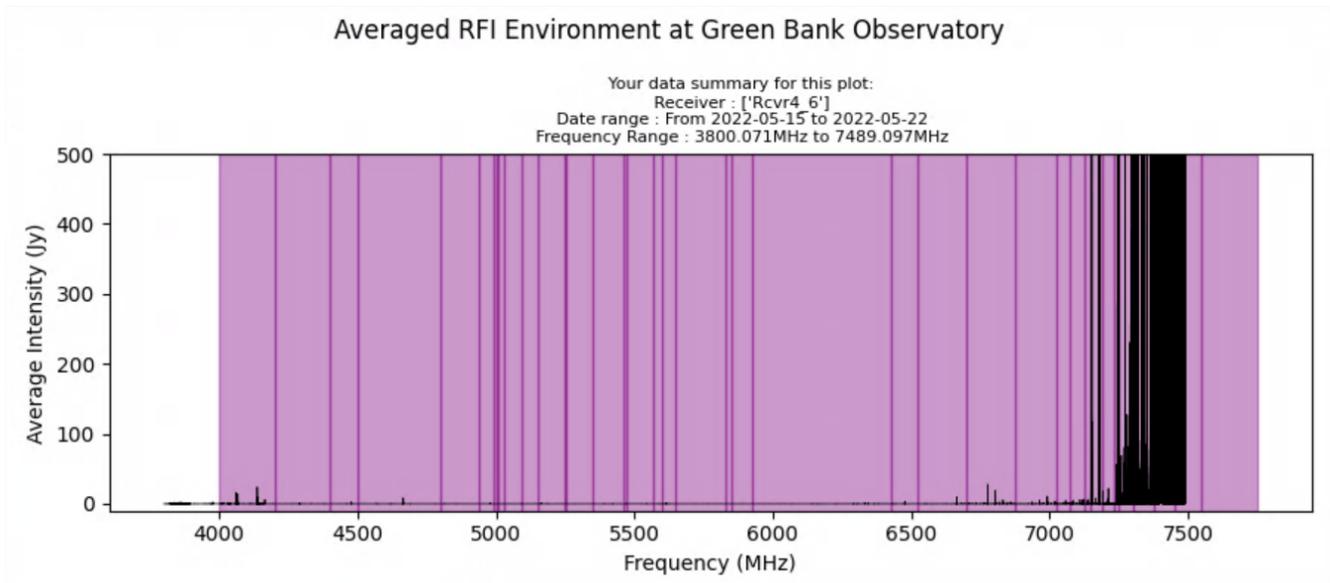


Figure (B)

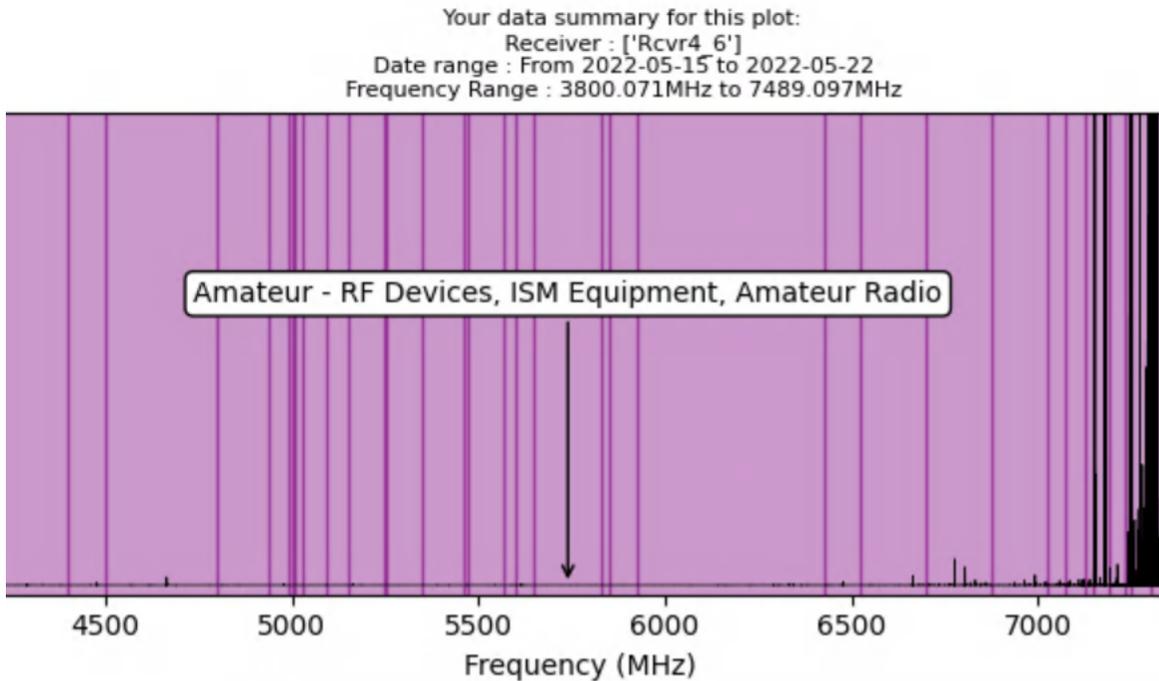


Figure (C)

### Element Catalog

- Fig (A) – Example of a graph before the annotations were implemented. The spikes on the graph are where there are sources of RFI.
- Fig (B) – Example of a graph after the annotations were implemented. Each line is the start and end of a new frequency range of RFI data.
- Fig (C) – Example of an annotation. The annotation appears in the middle of the data range for visual clarity.

### Context and Architecture

The original graph that was printed by the GUI (Figure (A)) was suitable for most people using the GUI. You plugged in the dates and frequency, and got data—simple. But, not as information-rich as it could be. With this update to the GUI, the graph is now populated blocks of information (Figure (B)). Each block designates a range of RFI, with a start and end frequency, that you can click on. Once clicked, the graph displays an annotation (Figure (C)) that contains information related to the RFI range selected. Currently, the information displayed is information gathered from the FCC’s Radio Allocation Database, which is also what the annotations’ ranges are based on.

## Programmer View

```
#click to get rfi info
self.getrfi = self.getrfi_func(start_frequency, end_frequency)
def onclick(event):
    print('%s click: button=%d, x=%d, y=%d, xdata=%f, ydata=%f' %
          ('double' if event.dblclick else 'single', event.button,
            event.x, event.y, event.xdata, event.ydata))
    click_rfi = self.getrfi.copy()
    click_rfi.drop(click_rfi[click_rfi['start'] > event.xdata].index, inplace = True)
    click_rfi.drop(click_rfi[click_rfi['end'] < event.xdata].index, inplace = True)
    click_rfi = click_rfi.reset_index()
    for row in range(click_rfi.shape[0]):
        mid = (click_rfi["end"][row] - click_rfi["start"][row])/2
        annot = ax.annotate(text=click_rfi["Comments"][row], xy=(click_rfi["start"][row]+mid,0), xytext=(click_rfi["start"][row]+mid,300), ha="center", textcoords="data",
                           bbox=dict(boxstyle="round", fc="w"),
                           arrowprops=dict(arrowstyle=">"), wrap=True)
        fig.canvas.draw()
        Artist.remove(annot)

#plot rfi info
if len(self.getrfi)>0:
    for row in range(self.getrfi.shape[0]):
        krfi = ax.axvspan(self.getrfi["start"][row], self.getrfi["end"][row], color="purple", alpha=.4)

fig.canvas.mpl_connect('button_press_event', onclick)
```

Figure (D)

```
# get known rfi dat
def getrfi_func(self, start_frequency, end_frequency):
    getrfi = pd.read_csv(r'/home/sandboxes/aseidman/gbt-rfi-gui/gbt_rfi_gui/fccsheet.csv', usecols=[
        'start', 'end', 'comments'])
    # drop the rfi that's outside of entered range
    getrfi.drop(getrfi[getrfi['start'] < start_frequency].index, inplace = True)
    getrfi.drop(getrfi[getrfi['end'] > end_frequency].index, inplace = True)
    #reset
    getrfi = getrfi.reset_index()
    print(getrfi)
    return getrfi
```

Figure (E)

```
freqdict = {
    ['Prime Focus 1']: [290, 920],
    ['Rcvr1_2']: [1150,1730], #'Rcvr1_2'
    ['Rcvr2_3']: [1730,2600], #'Rcvr2_3'
    ['Rcvr4_6']: [3800,7800], #'Rcvr4_6'
    ['Rcvr8_10']: [7800,11600], #'Rcvr8_10'
    ['Rcvr12_18']: [12000,15400], #'Rcvr12_18'
    ['RcvrArray18_26']: [18000,27500], #'Rcvr18_26'
    ['Rcvr26_40']: [26000,39500], #'Rcvr26_40'
    ['Rcvr40_52']: [39200,50500] #'Rcvr40_52'
}

if start_frequency == None:
    lower = freqdict[str(receivers)][0]
    start_frequency = lower

if end_frequency == None:
    upper = freqdict[str(receivers)][1]
    end_frequency = upper
```

Figure (F)

## Element Catalog

- Figure (D) – The main section of code that creates the annotations. This handles the culling of information besides the data clicked on by the user and printing all the visuals onto the chart.

- Figure (E) – The code that handles obtaining the data to use for the annotations. This includes the starting and end values, as well as all the written information about each range of RFI.
- Figure (F) – Code that automatically sets the frequency ranges when the user does not specify one.

## **Architecture Background**

There are three blocks of code that were added to the GUI. When the GUI is first booted, the code in Figure (E) is run, which obtains all the data that will be printed onto the graph. The code in Figure (D) has two main parts. The first part is the click function, which culls the information to just the range the user clicked on, and then prints the annotation with the respective data in the middle of the frequency range. The second part simply displays the overlay of frequency ranges specified by the data obtained in Figure (E). The code in Figure (F) is simply for when the user does not specify a frequency range, as the code would default to a very small and very large value for the start and end frequencies, respectively.

## **Reference Materials**

What to put here

- Relevant Links from the Wiki
- RFI Definition Link
- Something GBO Related

## **Glossary and Acronym List**

The following specialized terms are used:

- GBO: Green Bank Observatory
- RFI: Radio Frequency Interference
- GUI: Graphics User Interface