MTS Cover Figure & Building a Sankey in Tableau

Steps for the Cover Figure (“Figure 0”)

# Documents & Directories

The **MTS\_Sankey** folder comes with several folders/files. The names of the folders and the organizational structure is baked into the scripts to take the raw MTS data and transform it into the various datasets to create the figures in Tableau. Be aware that renaming folders or rearranging this directory structure will require updating the code.

# Steps

Please note that multiple paths or solutions exist, both in the Python scripts and Tableau step-by-step instructions. These instructions aim to be detailed, but do not purport to be the only way to achieve the desired end result.

### Prep the data

To go from the downloaded MTS data to “figure datasets” that are Tableau-ready, we will need to run two scripts to wrangle the data into the format we want. The Sankey cover figure requires us to first run the script used for the other figures (*MTS\_Create\_Viz\_Datasets\_for\_Tableau.py*) and then run the Sankey-specific script (*sankey\_merging\_data\_with\_model.py*)

### Running the Python script: Sankey\_merging\_data\_with\_model.py

This script can be found in the scripts folder. Several items will need to be edited once to get started and others, every time a new MTS is being read in. To help facilitate the process of finding these spots in the code, there will be areas marked with *“””CHANGE ME!”””* so that you can search through the script.

1. To start, run the entire script *MTS\_Create\_Viz\_Datasets\_for\_Tableau.py*. See the other user guide for instructions.
2. Next, with the Sankey script, find the “CHANGE ME” sections and…
   1. Change the directories to match your local machine
   2. Change the current month, current fiscal year, previous month, and previous fiscal year variables
   3. Change the path variable to match the filename,
      1. e.g ., ” fig\_cover\_0617\_made\_170921”.
      2. (Note that the first script we ran placed its output here:

**MTS > MTS\_Figures > data > output > figure\_datasets**.)

1. Run the entire script. This will likely take at least 15 minutes. Two files will be written: one for receipts, and one for outlays.
   1. (Note where the output files will end up, e.g.:
   2. **MTS > MTS\_Sankey > data >** “”fig0\_cover\_0517\_modified\_for\_sankey\_outlays\_only\_170922”
2. Read the end of the output after running the script. It will tell you if there was a deficit or surplus, so you know which Tableau workbook file to use.
3. It will also have the values for total receipts, total outlays, surplus, and deficit, so that you can just copy-paste that over to Tableau.

## Tableau – Replacement of Old Sankey Visualization

When a new MTS comes out and everything goes as planned in Tableau ☺

|  |  |  |
| --- | --- | --- |
| **STEP** | **INSTRUCTIONS** | **SCREENSHOT** |
| 1. Select which Tableau workbook to use (deficit or surplus) | Select which Tableau workbook to use based on whether there was a deficit or surplus this month. At the end of running the Sankey script, the output will print a message telling you which to use. |  |
| 1. Add the two data sources to the workbook and make necessary copies of the dataset within Tableau. | Add the two data sources –receipts, outlays– by going to the Data menu and selecting “New Data Source.” These files are in the **Sankey > Data** folder.  For both datasets, right click and create two duplicates. Right click each dataset and rename to a shorter name. In the end, you should have outlays v1, v2, and v3; and a receipts v1, v2, v3. These are duplicates, but we will need separate datasets later to have different colors applied to the same variables on the dashboard. |  |
| 1. Replace the old data with the new | On a worksheet, go to the Data menu and select “Replace Data Source…”. Replace each old dataset with the new dataset. You will need to replace six datasets in all, three for receipts, and three for outlays. If you can’t see the dataset you’re trying to replace, go to the worksheet that uses that dataset.  The data source each worksheet should be using is in the worksheet’s name, e.g. “Pillar 1 – Sources **(rec v1)**” |  |
| 1. Close the old data sources | In the top left corner of a worksheet, in the Data tab, close the old data sources.  If those data sources are in use, you will see a pop up window with a warning. |  |
| 1. Show all y-axis labels | On each sheet, right click the pill in Rows. Select “Show Header.” ‘  Right click the y-axis, select “Edit Axis…”  Select automatic instead of fixed. This will help the picture re-adjust to that month’s numbers before we fix all the figures to the same axis values later on. |  |
| 1. Ensure all pillars’ items are ordered correctly in order to line up with the Sankey. | In the pillars, the position of the item in the legend affects where objects are displayed in the bar chart. (Note: this is not true for the sankey curves).  Pillar1 – right click in the color legend, and select “Sort.” Select descending, sort by field, choose Stage1 Source Amount, and sum as the aggregation method. Then drag the spacers in between each real source. If the pillar’s bars are horizontal lines, ensure the Marks card dropdown is set to “bar” instead of “automatic.”  Pillar 2 – drag items in the color legend so that the order from top to bottom is: Receipt, Deficit, Dummy.  Pillar 3 – Drag the dummy function to the bottom. All others’ orders do not matter because they are the same color.  Pillar 4 – Similar to Pillar 1, right click in the color legend and sort the values in descending order by their outlay amount. Then, place spacers in between each real function. Drag surplus to the top, if applicable. |  |
| 1. Adjust colors | Ensure colors are as expected, as some may change when you bring in the new data source.  Pillar 1 – All should be white.  Sankey 1-2: Green  Pillar 2 – Receipts are dark green, deficit is gold (if applicable), and dummy is white. See hex codes document for the colors used.  Sankey 3-4: Teal  Pillar 3 – All functions are dark teal, surplus is gold (if applicable), and dummy is white.  Pillar 4 – All should be white. |  |
| 1. Set all axes on the dashboard view to the same fixed value to ensure the worksheets all align. | On the dashboard view, all worksheets have the y-axis header showing. Whichever y-axis has the highest value, right click that axis, and select “Edit axis…”  Select the “Fixed” radio button, copy the max value under “Fixed End,” and hit OK.  Use that value to set every other y-axis’ max value under “Fixed end.” This will ensure all worksheets on the dashboard are lining up because they have the same y-axis values. |  |
| 1. Remove the y-axis labels from each worksheet | Right click on the axis, de-select “Show header” |  |
| 1. Ensure all source/function values are always displayed, and spacer labels are always hidden | Right click a box in a pillar, hover over “Mark Label”, and then select “Always Show” or “Never Show” for functions and spacers, respectively.  Note: you can ctrl+click to highlight all the real data at once and select “Always show” and then all the spacers and select “Never show” |  |

## Tableau - Building Sankey from Scratch

Likely useful for if/when replacement doesn’t go quite right.

|  |  |  |
| --- | --- | --- |
| **STEP** | **INSTRUCTIONS** | **SCREENSHOT** |
| 1. Add T | Place ‘T’ in Columns. Change to dimension.  This field is essentially establishing an x-axis. |  |
| 1. Create a ‘Curve Plotted’ calculated field. | From the Analysis menu, Create a calculated field, ‘Curve Plotted’.  (I use underscores to start calculated fields so that they are at the top of the list, but this is not necessary.)  This variable tells Tableau to plot the top line of the polygon when the variable MinMax is “Max” and to plot the bottom line of the polygon with it’s “Min,” giving us the top and bottom of the shape of the Sankey curve.  Drag ‘Curve Plotted’ to Rows. Change to dimension. |  |
| 1. Change the colors | Drag ‘Stage1 Source’ to color for the Receipts side, or ‘Stage4 Function’ to color. |  |
| 1. Change the mark to Polygon. | From the dropdown menu below Marks, select Polygon.  Until we give Tableau a path for how to connect the dots, it doesn’t know in which order to connect our dots, or how to create the polygon we want. The viz may look weird at this stage. |  |
| 1. Add a path for Tableau to connect the dots to form a polygon. | Drag the ‘Path’ pill to the Path card. Change to dimension.  Note: Path won’t be a card option until polygon is selected.  Note: if Path is not changed to a dimension, you can add MinMax to the Detail card. Both these options should yield the same result, and your viz should finally look like a Sankey at this point. |  |
| 1. Edit the x-axis to run from [-6, 6]. | Right click the x-axis (T) and select “Edit axis…”  Select fixed and change the range from -6 to 6.  Right click the x-axis and de-select “Show Header” |  |
| 1. Adjust location of text and graphs on dashboard | This is monitor dependent. Ensure you are adjusting using the monitor from which you’ll take a screenshot.  In the top left corner of the dashboard view, select the Layout tab. Here you can adjust the (x,y) position manually of each sheet or floating object, as well as the (width, height) of the boxes themselves. If there are spaces between the Sankey curves and the bars, ensure that Pillar 2 is “behind” the 1-2 sankey curve, and that Pillar 3 is “behind” the 3-4 sankey curve (similar to sending objects forward/backward in PowerPoint). |  |

# Miscellaneous Design Notes

* Fonts are in Arial and font style guidance can be found in files 5a or 5b in the **Documentation** folder, “5a Treas\_BFS\_\_MTS\_Cover\_092517.” (Note: we originally recommended Sans Source Pro, but realized not all fiscal machines will allow this font.)
* Colors used in Tableau were not from the built-in color palettes, and so custom color palettes will need to be created. To import custom color palettes into Tableau, see the “Importing Custom Color Palettes into Tableau” document. There is also a separate document noting the hex codes used for the colors.
* For screenshot purposes, use the dashboard view, and view on full-screen mode.
* An issue brought up during our conversations was “masking” information about receipts/outlays by combining too many things into an “other” category. In the Sankey, I had to group functions into “Other” because there are negative values. I grouped just enough small functions until I had a positive “Other” category in the code. However, other categories could be chosen to go into the “Other” category, and this can be modified in the Sankey script. Lastly, “Other” is now only ~%2B for May 2017, whereas before, “Other” contained a much larger portion of the outlays.

# Summary

First run the “MTS\_Create\_Viz\_Datasets\_for\_Tableau.py” script. Then run the “sankey\_merging\_data\_with\_model.py” script. Be sure to change the handful of things in each script before running, such as the filenames for the new month’s MTS. Add both data sources to Tableau and make 2 copies of each so that you have 3 receipt datasets and 3 outlay dataset. A unique dataset will be used for each sheet to allow different colors on the same dashboard. Replace last month’s with this month’s datasets. Ensure the pillars are in the correct order by sorting the legend in descending order and inserting the Spacers are in between the real sources/functions. Remember that the order of the items in the legend will affect the pillars (bar charts) but not the Sankey. Ensure the colors are correct, as some may change when you load in new data.

# Contact Info

Please do not hesitate to reach out. Our team is happy to assist and ensure all the documents and files in this package are accessible and useful.

Janelle Becker | [becker\_janelle@bah.com](mailto:becker_janelle@bah.com)

Justin Marsico | [justin.marsico@fiscal.treasury.gov](mailto:justin.marsico@fiscal.treasury.gov)