USCCR Update Process

# Prerequisites

Ensure the following tools are installed on the machine being used to process USCCR data:

1. Microsoft Excel 2010 or later
2. PostgreSQL 9.0 or later - <https://www.postgresql.org/>
3. PostGIS 2.0 or later (available on Windows through PostgreSQL StackBuilder)
4. GDAL/OGR 2.1.3 Windows Binaries - <http://www.gisinternals.com/query.html?content=filelist&file=release-1600-x64-gdal-2-1-3-mapserver-7-0-4.zip>
5. Git for Windows - <https://github.com/git-for-windows/git/releases/download/v2.11.1.windows.1/Git-2.11.1-64-bit.exe>
6. Any text editor (Notepad++ recommended)
7. PgAdmin3 (version 1.16 or later) - <https://www.postgresql.org/ftp/pgadmin3/release/v1.22.2/win32/>

In addition to the above tools, a GitHub account with membership in the Zekiah organization is required.

# Initial Setup

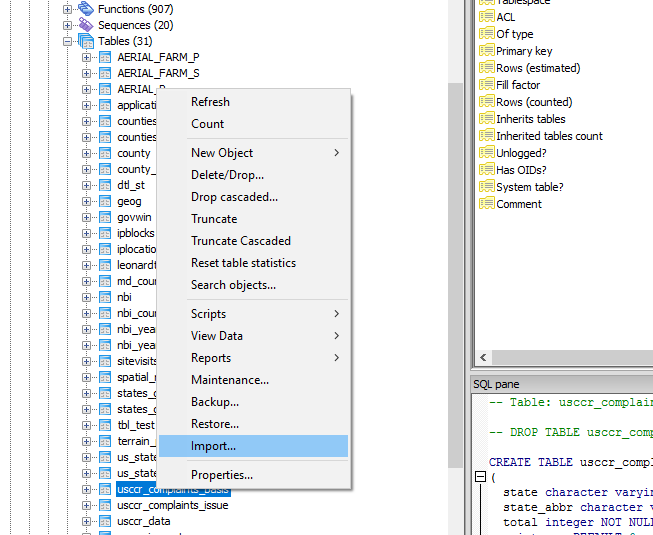
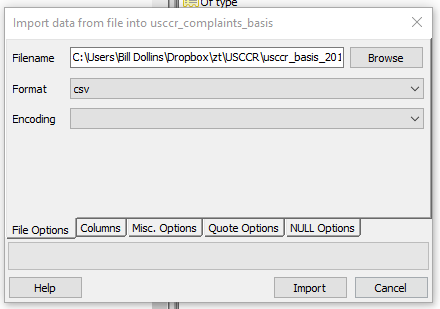
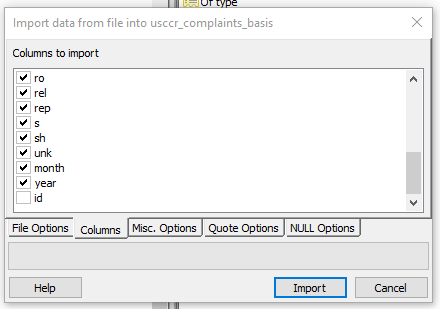
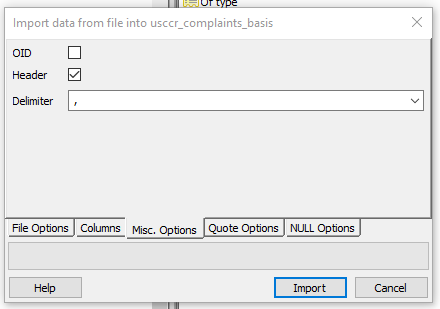
The following one-time-only steps will need to be performed before processing data for the first time

1. Create PostgreSQL database in which to perform processing
2. Import state boundary data
3. Import historical USCCR data
4. Edit batch files with correct database and login information
5. Using SQL scripts, create the necessary database objects
   1. List objects/scripts here
6. Clone USCCR site from GitHub to local disk using the following command: “git clone <https://github.com/Zekiah/Zekiah.github.io.git>”
7. Clone usccr\_tools repo from GitHub to local disk using the following command: “git clone <https://github.com/Zekiah/usccr_tools.git>”

# Processing Data

When an update is received, that data will be processed into a formatted spreadsheet with two tabs: “Basis” and “Issue.” Using Microsoft Excel, verify that the data in each tab has been formatted correctly. It is important to ensure the “month” column contains a value that is one higher than the previous update. The “month” column is simply a sequential number that should be reset for each year. For example, the first update of 2017 should have a “month” value of 1. The second update should have a value of 2. The term “month” is a legacy from the start of the project; this data no longer actually represents the month. This data is necessary in order to correctly calculate the current period and year-to-date totals as require by USCCR.

Once the data format has been verified, complete the following steps:

1. Export each tab to a CSV using the “Save As” function in Excel. You should have two files for the current period. The file names should use the following convention: usccr\_basis\_{year}\_{month}.csv and usccr\_issue\_{year}\_{month}.csv where “{year}” and “{month}” are replaced with the year and month values of the current update. For example, files for the first update of 2017 should be usccr\_basis\_2017\_01.csv and usccr\_issue\_2017\_01.csv
2. Import CSVs into PostgreSQL  
   Importing the CSVs can be accomplished completely within PgAdmin using the following steps
   1. Right-click on the target table (usccr\_complaints\_basis or usccr\_complaints\_issue) and choose “Import…”  
      
   2. In the resulting dialog, navigate to the CSV file, which will appear in the “Filename” box. Then, choose “csv” in the “Format” dropdown. When these steps are completed, select the “Columns” tab.  
      
   3. In the “Columns” tab, scroll down and uncheck the “id” column. This column is calculated in the database and should not be imported. When this is complete, choose the “Misc. Options” tab.  
      
   4. In the “Misc. Options” tab, make sure the “Header” checkbox is checked. In the “Delimiter” dropdown, choose the comma as the delimiter.  
      
   5. After the above steps have been completed, click the “Import” button. Repeat these steps for both the basis and issues CSV files. The columns in step “C” will be different between the tables, but the “id” column must be unchecked for both files.
3. Export the GeoJSON files used in the map. To perform this task, simply use the ogr\_cmd.bat batch file to export the GeoJSON files from the database. This batch file expects the month and year as parameters. For the first update of 2017, the command would be:  
     
   ogr\_cmd.bat 01 2017  
     
   The batch file will produce two GeoJSON files. Using the example the above, the files will be named usccr\_basis\_2017\_01.geojson and usccr\_issue\_2017\_01.geojson
4. Export the tabular totals. This data is used to produce the site’s bar charts. It is produced by running the usccr\_totals\_csv.bat batch file. Again, it expects the month and year as parameters. For the first update of 2017, the command would be:  
     
   usccr\_totals\_csv.bat 01 2017  
     
   This command will yield a file named usccr\_table\_source\_2017\_01.csv
5. Copy new data files to the site’s data folder. Using the first update of 2017 as an example, the data processing steps above will have produced the following files:  
     
   usccr\_basis\_2017\_01.csv  
   usccr\_basis\_2017\_01.geojson  
   usccr\_issue\_2017\_01.csv  
   usccr\_issue\_2017\_01.geojson  
   usccr\_table\_source\_2017\_01.csv  
     
   Copy these files into the ‘data’ folder of the site, which is located where you cloned the site from GitHub above.

# Updating the Site

With the data processed and staged, it is now time to update the web site and commit the changes. All updates occur in two JavaScript files; no HTML editing is required.

## Updating config.js

1. Navigate to the folder where you cloned the Zekiah.github.io site and go into the usccr\js folder, which contains the sites JavaScript files.
2. Open config.js in a text editor

For the following edits, we will use the first update of 2017, covering a date range of 12/3 to 1/7 as an example. With that example, edit the first eight (8) lines of config.js to read:  
  
var global\_month = "12/3 – 1/7";

var global\_month\_abbr = "12/3 – 1/7";

var global\_year = "2017";

var global\_geodata\_issue = "usccr\_issue\_2017\_01.geojson";

var global\_tabledata\_issue = "usccr\_issue\_2017\_01.csv";

var global\_geodata\_basis = "usccr\_basis\_2017\_01.geojson";

var global\_tabledata\_basis = "usccr\_basis\_2017\_01.csv";

var global\_tablesource = "usccr\_table\_source\_2017\_01.csv";

1. Save config.js and close the file

## Updating history.js

1. Navigate to the folder where you cloned the Zekiah.github.io site and go into the usccr\js folder, which contains the sites JavaScript files.
2. Open history.js in a text editor  
     
   For the following edits, we will use the first update of 2017, covering a date range of 12/3 to 1/7 as an example. With that example, we will add an entry into the global\_history array. The edits are highlighted below:  
     
   var global\_history = {"201501": {"path": "./?month=01&year=2015&abbrev=1%2F1%20-%201%2F25", "label": "1/1 - 1/25/2015"},

"201502": {"path": "./?month=02&year=2015&abbrev=1%2F26%20-%202%2F20", "label": "1/26 - 2/20/2015"},

"201503": {"path": "./?month=03&year=2015&abbrev=2%2F23%20-%203%2F20", "label": "2/23 - 3/20/2015"},

"201504": {"path": "./?month=04&year=2015&abbrev=3%2F23%20-%204%2F24", "label": "3/23 - 4/24/2015"},

"201505": {"path": "./?month=05&year=2015&abbrev=4%2F27%20-%205%2F22", "label": "4/27 - 5/22/2015"},

"201506": {"path": "./?month=06&year=2015&abbrev=5%2F25%20-%206%2F19", "label": "5/25 - 6/19/2015"},

"201507": {"path": "./?month=07&year=2015&abbrev=6%2F22%20-%207%2F14", "label": "6/22 - 7/14/2015"},

"201508": {"path": "./?month=08&year=2015&abbrev=7%2F20%20-%208%2F14", "label": "7/20 - 8/14/2015"},

"201509": {"path": "./?month=09&year=2015&abbrev=8%2F17%20-%209%2F11", "label": "8/17 - 9/11/2015"},

"201510": {"path": "./?month=10&year=2015&abbrev=9%2F14%20-%2010%2F16", "label": "9/14 - 10/16/2015"},

"201511": {"path": "./?month=11&year=2015&abbrev=10%2F19%20-%2011%2F20", "label": "10/19 - 11/20/2015"},

"201512": {"path": "./?month=12&year=2015&abbrev=11%2F21%20-%2012%2F27", "label": "11/21 - 12/27/2015"},

"201601": {"path": "./?month=01&year=2016&abbrev=12%2F28%20-%202%2F2", "label": "12/28/2015 - 2/2/2016"},

"201602": {"path": "./?month=02&year=2016&abbrev=2%2F3%20-%202%2F21", "label": "2/3/2016 - 2/21/2016"},

"201603": {"path": "./?month=03&year=2016&abbrev=2%2F22%20-%203%2F18", "label": "2/22/2016 - 3/18/2016"},

"201604": {"path": "./?month=04&year=2016&abbrev=4%2F25%20-%205%2F20", "label": "4/25/2016 - 5/20/2016"},

"201605": {"path": "./?month=05&year=2016&abbrev=" + encodeURIComponent("5/23/2016 - 6/17/2016"), "label": "5/23/2016 - 6/17/2016"},

"201701": {"path": "./?month=01&year=2017&abbrev=" + encodeURIComponent("12/3/2016 - 1/7/2017"), "label": "12/3/2016 - 1/7/2017"}

};  
  
The edits are summarized as follows:  
a. Copy the entire line beginning with “201605”

b. Add a comma to the end of the line

c. Press “enter” to create a new, empty, line below it

d. Paste the copied text onto the new line (highlighted line above)

e. Make the edits highlighted in cyan

1. Save history.js and close the file

# 5.0 Publishing Changes

The USCCR is hosted using GitHub pages, so publishing is simply committing changes back to GitHub. Use the following steps to publish the changes:

1. In a command window, navigate to the folder where you cloned the Zekiah.github.io site
2. Type the command following command (the trailing dot is part of the command): git add .
3. Type the command: git commit –m “some commit message is required”
4. Type the command: git push origin master
   1. You will be prompted for your GitHub user name and password
5. Site publication is complete