

Dale Jackan  
Geog 385  
Jennings  
11/22/2019

## **Project Summary**

### **Purpose**

The purpose of my project is to provide a work flow for collecting, viewing, editing, and summarizing or reporting data about the salamanders of California. In order to accomplish this I created a collector application, a web map application, and a dashboard.

### **Description**

The data collection and development had been accomplished in previous classes where I had developed a database to store the data for a project such as this. A lot of what I had to do to create the structure for this project was based off the layers, domains, design, etc of that previous database. The salamander range layer and habitat types domain info were acquired from the CDFW, and the city and county layers were from Caltrans.

The design was influenced by a discussion I had with the instructor where the work flow has divided into three pieces each with it's own task: collector = data collection; web map application = data collection, viewing, editing, quality assurance; dashboard = summarizing data. The goal for developing a collector application was to keep it as simple and informative as possible without a heavy processing load with a focus on phones, though it could be used on a tablet they were not the main target of that process. When developing the web map application the focus was on viewing, filtering, and editing data with an eye toward use on tablets, laptops, and desktops.

The most important part was trying to make sure that my initial feature layer I created was setup correctly using ArcPro prior to uploading to ArcGIS Online (AGOL). This did not always go smoothly and I had to come up with a procedure to download the layer from AGOL edit it in ArcPro and upload back to AGOL. I used the following procedure:

1. Create a project in ArcPro and then add the data from AGOL to my map in ArcPro.
2. Import the layers into the project gdb using the catalogue pane.
3. Delete the layers from AGOL and make sure the layers imported into the project gdb are now added to the map, and if not add them.
4. Do whatever processing task, editing, etc you needed to do.
5. SAVE the project
6. Publish back to AGOL.

The major reason why I used this procedure is because I had an extensive list of habitat types from the CDFW and it was much easier to use the "Table to Domain" tool in ArcPro to create the domain from an imported excel file than it was to enter everything by hand. I also had an issue with fields/columns being in an undesirable order for data entry in collector. There were two methods for correcting this. The first was to bring it back into ArcPro and correct, or open

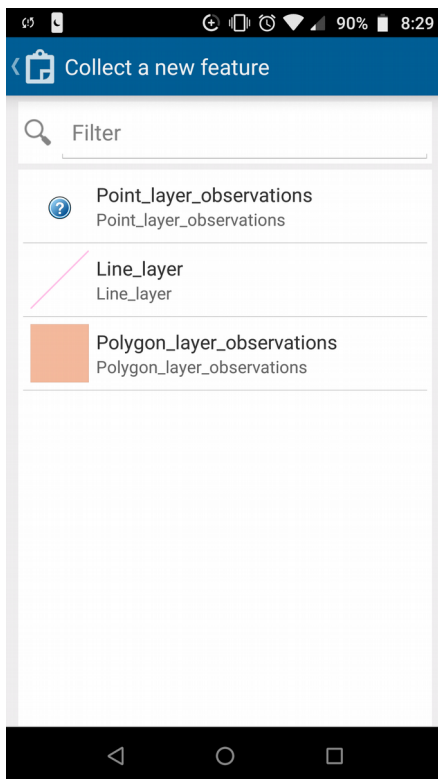


Figure 1: The feature type data collection options.

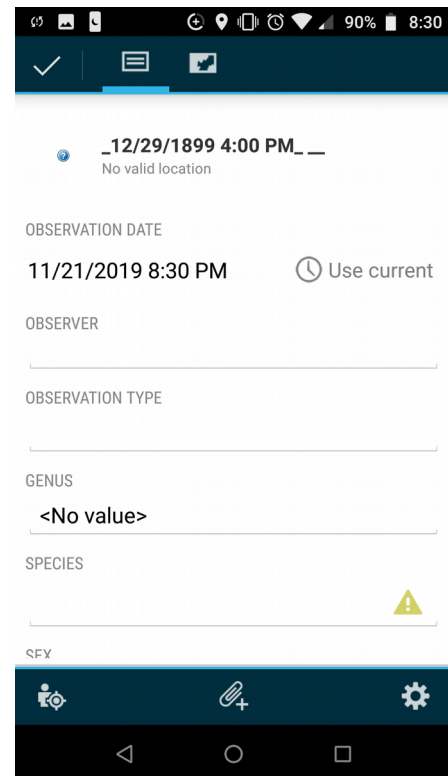


Figure 2: The initial form fields as you collect data.

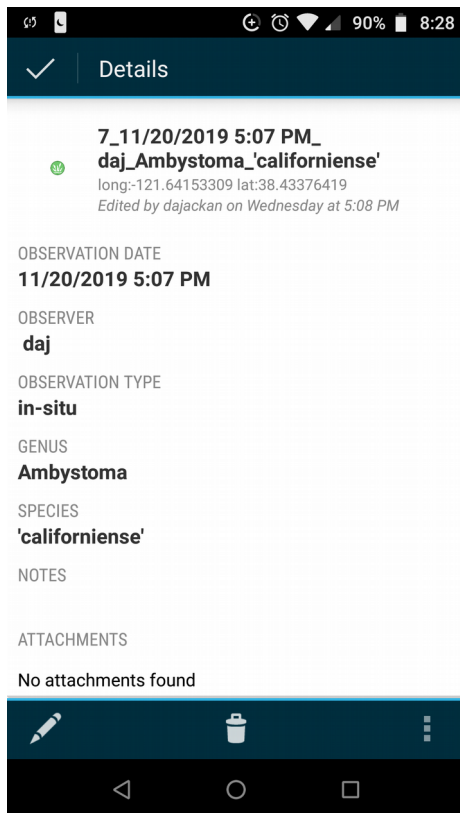


Figure 3: Some of the fields filled out when collecting data.

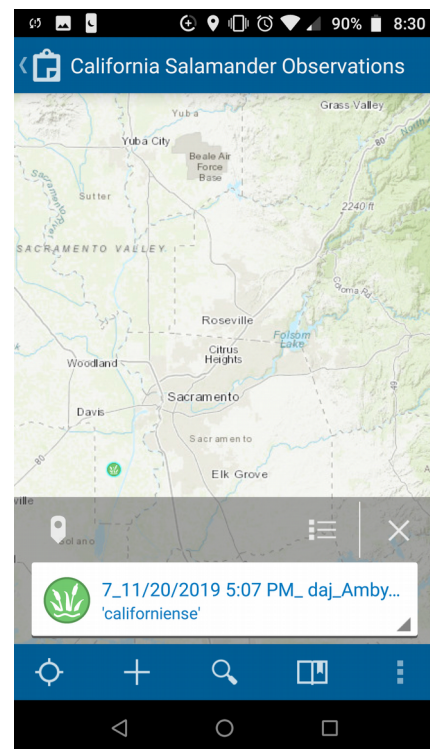


Figure 4: The screen just after you complete data collection.

the layer in a Map Viewer and use “Configure Pop-up->Configure Attributes” to reorder the fields/columns. I also created an editable layer for data collection and based on my experience it needed to be able to collect points, lines, and polygons. At the “Content” page you click the “Create” button and choose “Feature Layer->Build A Layer” and then in this case I chose “Points, lines, and polygons”. After the new feature layer is created go to “View Details->Settings” and turn on editing. At this point I had two feature layers. One feature layer had the salamander ranges, cities, and counties layers. The other feature layer was editable for data collection and could collect points, lines, and polygons. Those two feature layers were then used as the base for my web maps. I made two web maps one web map was for data collection and use with collector, and the other for a web map for the web map application. The collector app can be seen in Figures 1-4. The web map for collector had only the editable feature layer, while the other web map for viewing and editing etc used the editable feature layer and the feature layer with range maps, cities, and counties. By design this meant the collector app was lightweight, whereas the web map for the web map application provided the range maps, counties, and cities for context and analyses. The symbolization was the same between the two maps where I used the fields observation\_type (points), feature\_type (lines), and feature\_type (polygons) for symbolization on unique values. I customized what was viewed in the pop-ups, but for the web map used with the web map application I added a custom expression to the pop-up on the point layer:

```
var intersectLayer =Intersects(FeatureSetByName($map,"Map - range_maps2"), $feature)
```

```
var list = ""
var n=1
for (var f in intersectLayer){
  list += n + '. ' + f.CName + TextFormatting.NewLine
  n=n+1
}
return list
```

The result of this expression is that the pop-up reports the common names of the salamander ranges the point is within (Figure 5.). If you do not see the species you just found within the list then you may have just found a range extension. I found the expression was best applied to the point layer as opposed to the county polygon layer. Now if you just wanted to apply it to the county layer then you could find out what salamander species are found in the county you selected. You do not want to have the expression active on the point and polygon layer at the same time because the way intersect works it could yield misleading results showing multiple counties for the point depending upon zoom level. For the most part the range maps, cities, and county layers are not displayed and are there to act as information sources for queries etc.

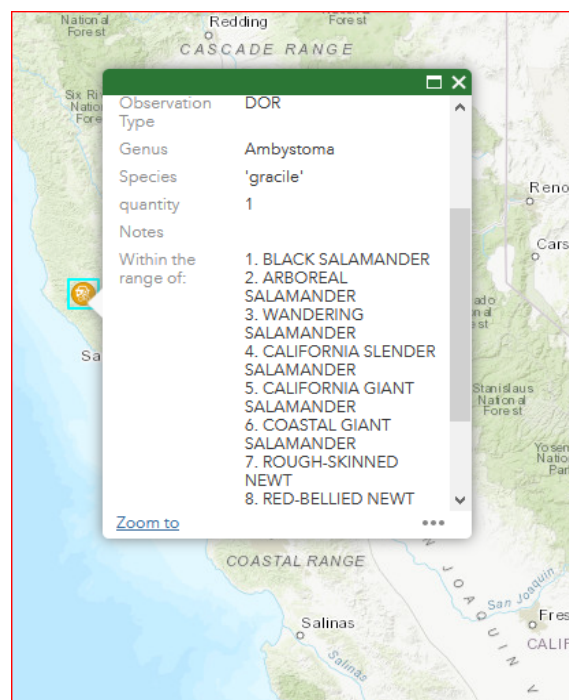


Figure 5: The result of the custom arcade expression in the pop-up.

I added several widgets to the web application (Figure 6.), but the main one was the filter widget. I found out that there was a known [bug](#) where the icon designated for the filter would not replace the standard icon when the filter was applied. I experimented with the capabilities of the filter and I applied a filter set that basically used the two fields “genus” and “species”. What this meant is even though AGOL cannot use contingent values from ArcPro I could still set the genus and species to particular value and where both values are met resulting in a filter for a specific species of salamander. Contingent values would still be helpful for data entry, but this works well for viewing the data.

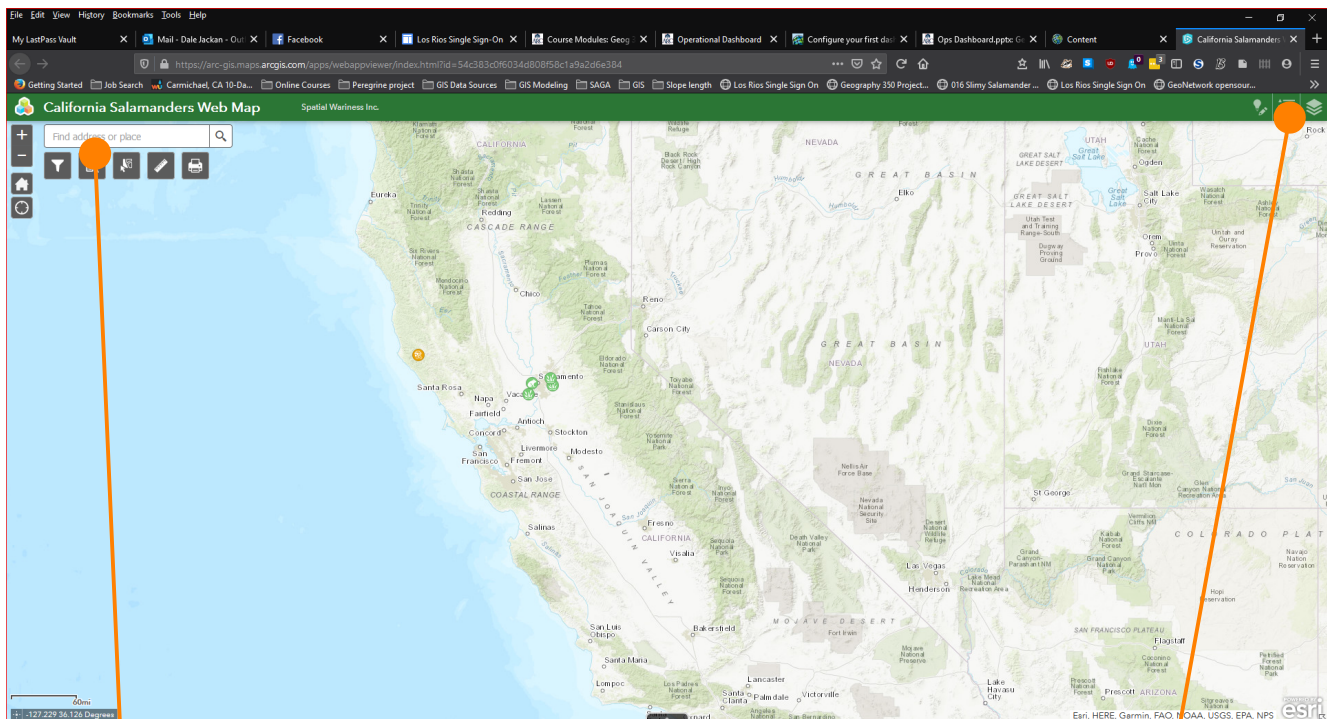
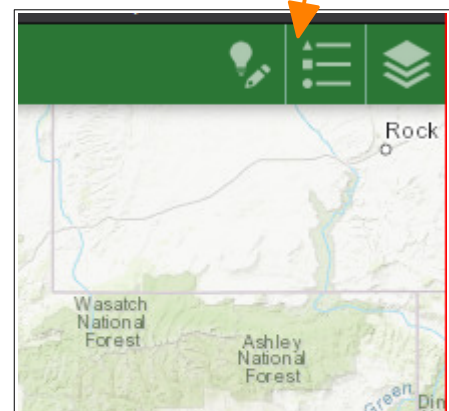
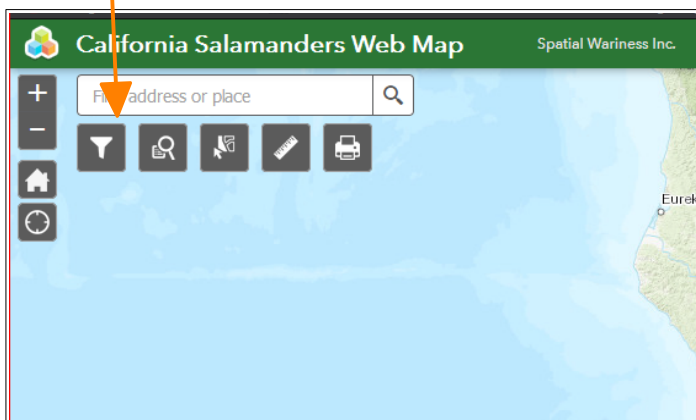


Figure 6 An overview of what the web map application looks like.





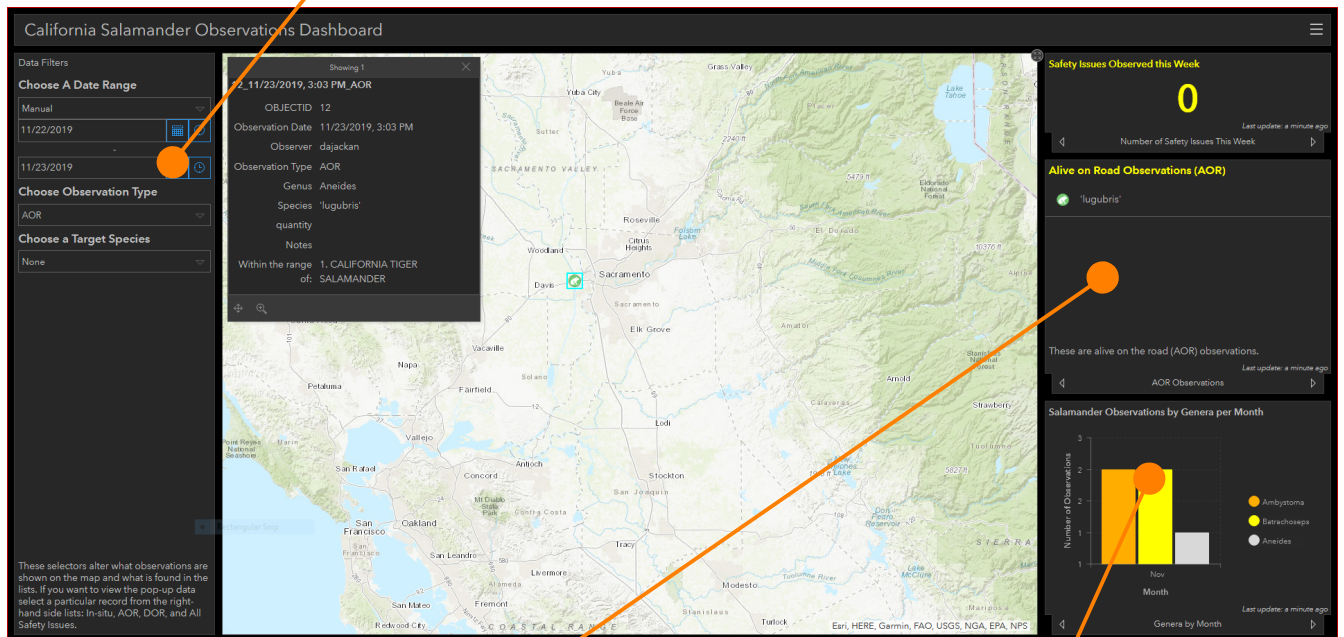
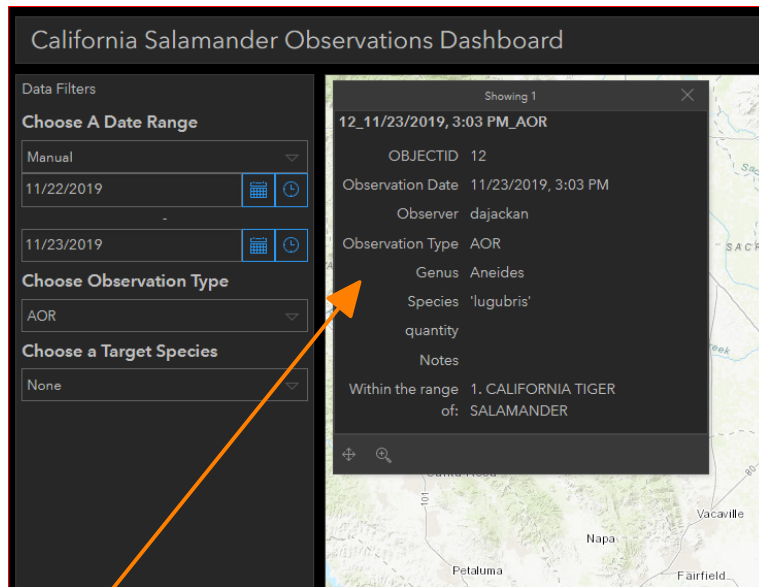
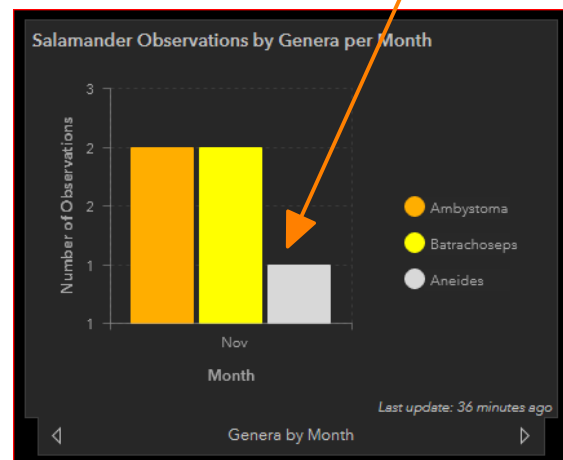
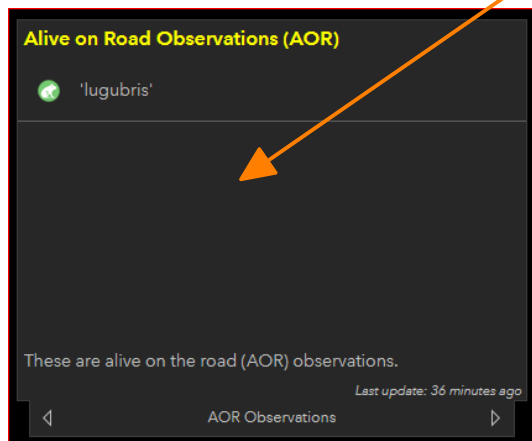


Figure 7: Overview of dashboard with selector, list, and a bar chart active shown by arrows.



For the dashboard (Figure 7.) I created an indicator, lists, serial charts, a pie chart, and a selector. The indicator tells you how many safety issue observations were found “this week” and there is an associated list showing the observations for “this week”. I created two bar graphs. One bar graph shows how many observations per day for “this week”, and the other shows how many of each observed salamander genus for the month. The pie chart shows the overall proportion of each salamander genus in total from the records. I created a selector that allows the user to narrow or filter the observations by date, observation type, and target species. The selectors also target the map and then lists I made for each observation type: In-situ, AOR, DOR, and Safety Issues. By having the selectors target the lists and map they act as a filter on what the lists and map shows.

## Discussion

I had fun and learned a lot during this project. I would say I completed a reasonable data collection, editing, and viewing work flow utilizing the collector app, web map app, and dashboard. I mainly wanted to learn how to create and setup the web applications and I feel I successfully accomplished this. Some things that could be improved on are I wish the collector app (Android) had the ability to copy a record and then the observer could go in and adjust only the fields that need to be changed. As in I see the same species etc, but there are two males and one female, make the record for the male copy the record and alter the field for sex and quantity and your done. I also learned the collector app is different between iOS and Android, but the Android version is still beta. I also would like for there to be a way to use the dashboard to tally how many observations for the week etc were outside the known range for a species, but I have been unable to find a method in the dashboard for this. Having “unique” as a filter choice for the dashboard gauge or indicator would be very useful because it would allow me to keep track of how many unique species have been observed out of the total known species for the state. The custom arcade expression for the pop-ups also didn’t carry over to every element of the dashboard. If I click on the icon in the map it will work, but if I click on the observation in the list dashboard widget etc the arcade expression results will not work appear.

I need to be more careful with my domains and should have sorted them alphabetically in the tables prior to turning into domains. I might need to rework my defaults and editable and non-editable fields. I may need to rethink how important the GPS metadata from the phone is to the records. The GPS metadata is entered automatically, but does consist of a large number of fields. I had some issues using the Android version of collector and finding the data I had collected. I believe the manner in which you find you data is supposed to be through the search, but the default is essentially useless. In order to compensate for this issue you have to go into “View item details” → “Settings” for the webmap used by collector and scroll down to “Application Settings” and check “By Layer” and create expressions for fields the search in collector can look for (Figure 8.). Somehow some of the fields were doubled and I need to clean that up and understand how that happened. I suspect it may have been from uploading and overwriting the layer using ArcPro. The values for wetland in the polygon layer etc need to be fleshed out more, what was used was for demonstration purposes and not a proper extensive list of types. I will say some of the types might be available in the habitat types domain which was sourced from, as of this time, the current official CDFW listing. The other feature types and or classifications in the polygon layer could probably use closer examination and elaboration. There is also potential for the cities layer to be used for nearby functionality or proximity analysis. In all I learned a lot and found this experience very beneficial.

Application Settings

Select the tools and capabilities to enable in applications that access this web map.

☒ Find Locations [-]

Hint text

Place or Address

☒ By Layer

Observations\_g - Point\_

Observer

Equals

×

Observations\_g - Line\_

Creator

Equals

×

Observations\_g - Polyg

Creator

Equals

×

Add Layer

Window Snip

☐ By Address

☒ Use in Collector for ArcGIS

Figure 8: This is where you add fields for collector to search to find your data.