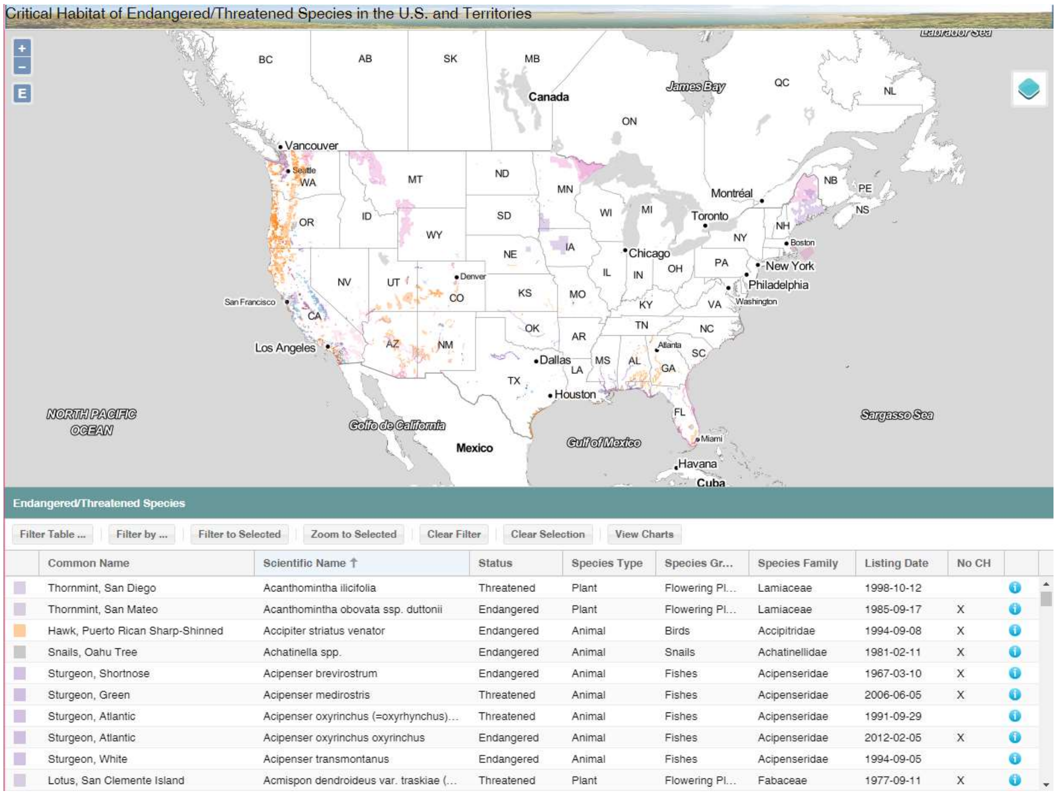
An Open-Source Approach to Creating a National Critical Habitat Web Map



Critical Habitat of Endangered/Threatened Species in the U.S. and Territories

Endangered/Threatened Species

	Common Name	Scientific Name ↑	Status	Species Type	Species Gr...	Species Family	Listing Date	No CH	
🟪	Thornmint, San Diego	Acanthomintha ilicifolia	Threatened	Plant	Flowering Pl...	Lamiaceae	1998-10-12		ⓘ
🟪	Thornmint, San Mateo	Acanthomintha obovata ssp. duttonii	Endangered	Plant	Flowering Pl...	Lamiaceae	1985-09-17	X	ⓘ
🟧	Hawk, Puerto Rican Sharp-Shinned	Accipiter striatus venator	Endangered	Animal	Birds	Accipitridae	1994-09-08	X	ⓘ
⬜	Snails, Oahu Tree	Achatinella spp.	Endangered	Animal	Snails	Achatinellidae	1981-02-11	X	ⓘ
🟪	Sturgeon, Shortnose	Acipenser brevirostrum	Endangered	Animal	Fishes	Acipenseridae	1967-03-10	X	ⓘ
🟪	Sturgeon, Green	Acipenser medirostris	Threatened	Animal	Fishes	Acipenseridae	2006-06-05	X	ⓘ
🟪	Sturgeon, Atlantic	Acipenser oxyrinchus (=oxyrhynchus)...	Threatened	Animal	Fishes	Acipenseridae	1991-09-29		ⓘ
🟪	Sturgeon, Atlantic	Acipenser oxyrinchus oxyrinchus	Endangered	Animal	Fishes	Acipenseridae	2012-02-05	X	ⓘ
🟪	Sturgeon, White	Acipenser transmontanus	Endangered	Animal	Fishes	Acipenseridae	1994-09-05		ⓘ
🟪	Lotus, San Clemente Island	Acmispon dendroideus var. traskiae (...	Threatened	Plant	Flowering Pl...	Fabaceae	1977-09-11	X	ⓘ

Deanna Sarro

GEOG 596B: Capstone Project

Penn State University

Master of Geographic Information Systems Program

Advisor: Jim Detwiler

Introduction

The U.S. Army Corps of Engineers (USACE) has a requirement to track impacts to species and their critical habitat as a result of work performed at sites where the public has obtained permits to alter waters of the United States[1] so as not to adversely alter navigation. The objective of the project was to create a web mapping application that can be used by the USACE, as well as other government agencies and the public, to identify areas where critical habitats exist and to provide information about all endangered and threatened species located within the United States and its territories. The project was also intended to analyze and put into practice the use of open-source technology for data storage, retrieval, and display.

Background

The Endangered Species Act of 1973 (ESA)[2] provides for the conservation of endangered and threatened species throughout the world. The species identified on the ESA list are managed by the U.S. Fish and Wildlife Service (USFWS) and the National Marine Fisheries Service (NMFS), which is part of the National Oceanic and Atmospheric Administration (NOAA). There are approximately 2,270 species worldwide listed as endangered or threatened under the ESA, of which 650 are only found in foreign countries. The remaining ~1,620 species are found within (but not limited to) the United States and it territories.

One method of protecting threatened and endangered species is through the creation of critical habitat areas. Critical habitats are “the specific areas within the geographic area, occupied by the species at the time it was listed, that contain the physical or biological features that are essential to the conservation of endangered and threatened species and that may need special management or protection.”[3] Of the ~1,620 U.S. species, critical habitat has been designated for 704, slightly less than half. A study conducted by the Center for Biological Diversity found that the endangered and threatened species with a defined critical habitat are “more than twice as likely” to be increasing in population.[4] However, the USFWS has a limited budget and believes that resources are better spent getting new species added to the ESA list and under federal protection rather than defining critical habitat for species on the existing list.

Objective

Based on the ESA, only Federal agencies are required to consult with the USFWS in regards to any destruction or adverse modification to critical habitats; private land owners are not required to consult with the USFWS provided any modifications to the area are not federally funded or authorized.[5]

A mapping application for the public depicting the location of critical habitats will promote knowledge and understanding of possible impacts to species listed under the ESA when changes to the landscape do not require federal oversight. Federal agencies will benefit from a comprehensive web map to identify critical habitats overlapping with their projects and for analysis with other datasets.

Critical habitat web maps that exist today are limited in the display of information pertaining to threatened and endangered species and their associated critical habitats. There are very few web mapping applications showing all critical habitat data. Most critical habitat maps are either limited to a particular species or group of species or contain static images.

The most comprehensive online web application that currently exists is the “Critical Habitat for Threatened & Endangered Species [USFWS]” [6] which uses an Esri web server to create a map showing the location of all critical habitats. This map suffices as a basic display of the core critical habit information but does not provide the level of information necessary for the USACE.

Expanding on the functionality and datasets within a new viewer will allow for more complex analysis of the critical habitats and all listed species. A few enhancements in the new web map to provide the needed tools include:

- List all species in the table, not just those with a designated critical habitat, and define a method to display this information on the map
- Use different colors and fill techniques to make overlapping critical habitats more apparent
- Expand on the table filtering options to include the species group and the state(s) in which the species is found
- Create an interactive relationship between features on the map and records in the table
- Select all overlapping critical habitats located at a specific point
- Provide additional information obtained through spatial analysis for interaction with other datasets
- Link to the species information on the USFWS Environmental Conservation Online System (ECOS) and/or other web sites with information about the species.

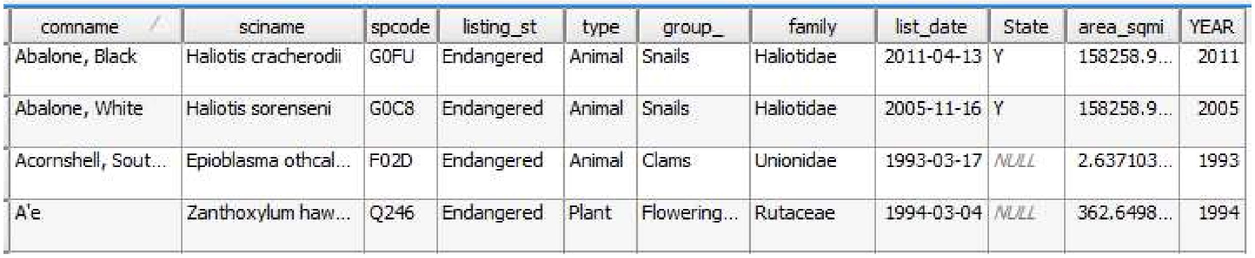
Data Acquisition

Critical habitat data is available for download from the USFWS web site in shapefile format and contains the spatial information defining the critical habitats as well as some attribute information. The data is also available as a WMS but this format does not provide the ability to perform spatial analysis nor does it contain the attributes needed for the web map. There are both polygon and line shapefiles for critical habitat data. The lines were buffered at 5 meters to store as areas with the original polygon data.

A complicating factor is that not all endangered and threatened species have a defined critical habitat. As noted earlier, the actual number of species with designated critical habitat is less than half the total number of species on the ESA list found within the U.S. and its territories. A listing of all species, regardless of critical habitat designation, is available from the USFWS on several different web sites, each containing different attribute information.

Three different USFWS sources were necessary to obtain all of the attributes required for the web application. These include:

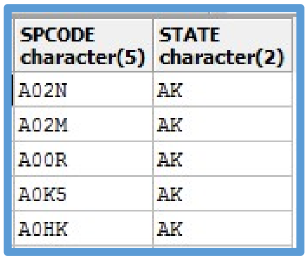
- **U.S. FWS Threatened & Endangered Species Active Critical Habitat Report** – polygon and line critical habitat shapefiles including common name, scientific name, species code, and listing status (only for species with designated critical habitat)
- **Listed Animals/Listed Plants** – web-based listing of all species covered by the ESA including scientific name, common name, and species group
- **TESS Query web service** – XML query tool for USFWS database to obtain species code, scientific name, status, species family, and listing data.



comname	sciname	spcode	listing_st	type	group_	family	list_date	State	area_sqmi	YEAR
Abalone, Black	Haliotis cracherodii	G0FU	Endangered	Animal	Snails	Haliotidae	2011-04-13	Y	158258.9...	2011
Abalone, White	Haliotis sorenseni	G0C8	Endangered	Animal	Snails	Haliotidae	2005-11-16	Y	158258.9...	2005
Acornshell, Sout...	Epioblasma othcal...	F02D	Endangered	Animal	Clams	Unionidae	1993-03-17	NULL	2.637103...	1993
A'e	Zanthoxylum haw...	Q246	Endangered	Plant	Flowering...	Rutaceae	1994-03-04	NULL	362.6498...	1994

Figure 1: Attribute Information. *Graphic showing a subset of the attribute information that was compiled from the three different USFWS data sources.*

One attribute not included in the shapefiles or the other two USFWS sources is the state in which each species is found. Individual state lists were downloaded and compiled into a national list for querying and associating to existing critical habitat data to identify those species that do not have critical habitat and need to be shown uniquely. The state lists were extracted from the USFWS for each state ("Listed species believed to or known to occur in <state>") and then merged into a national file for use in the application. The linkage between the spatial data and the state lookup table is the species code (SPCODE) field.



SPCODE character(5)	STATE character(2)
A02N	AK
A02M	AK
A00R	AK
A0K5	AK
A0HK	AK

Figure 2: Species to State Relationship. *Graphic showing a subset of the table containing the species code in relation to each state in which the species is found.*

Data Storage

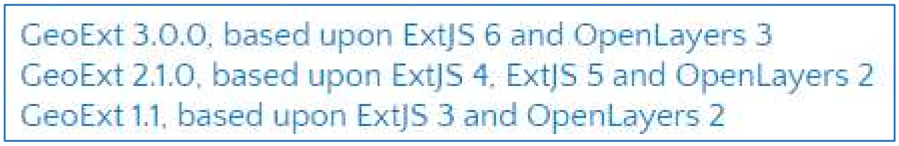
PostgreSQL is an open-source, object-relational database providing secure access and storage of data while providing robust functionality. The PostGIS extension to PostgreSQL provides support for spatial information. Information stored in the PostGIS database is made accessible to the mapping application through GeoServer. GeoServer is an open-source server for publishing data. The advantage of using GeoServer as a throughput for providing the data to the application is to provide a level of security by adding a layer between the database and the application. It ensures access to the data by following the same-origin policy to protect against malicious code injection.

Application Programming Interface

Using Free and Open Source Software (FOSS) allows for leveraging existing code at no cost and provides greater flexibility in adapting and editing source code to meet the needs of the application. Two primary FOSS application programming interfaces (APIs) are available: OpenLayers and Leaflet. Both APIs are open-source javascript libraries that allow for loading, displaying, and rendering maps. OpenLayers offers a more detailed set of base code and plug-in libraries due to its maturity and level of functionality included in the standard library base. Leaflet is much smaller in size which results in quicker loading time of the API but offers less built-in functionality. Considerations regarding the complexity of the application and the set of tools already available from other developers in order to alleviate the need to create functionality from scratch led to the decision to use OpenLayers.

The ExtJS application framework extends the capabilities of the application interface by providing ready-made tools for creating panel sections within a web site for displaying information such as a map or table grid as well as the ability to create new windows and toolbars. GeoExt is “a JavaScript framework that combines the GIS functionality of OpenLayers with the user interface of the ExtJS library provided by Sencha.”[7] Combined with the GeoExt extension, data stored for the table grid is related to the map features providing for selections within one panel to interact with information in another panel.

The latest version of OpenLayers, version 3, was selected for use to take advantage of newer technology, including the ability to use vector tiles (see below for more on this topic). OpenLayers 3 is a complete re-write of the base code library which made any portability from OpenLayers 2 fairly nonexistent. Because of the re-write of OpenLayers 3, there were also new versions of ExtJS and GeoExt released to correspond with the changes to the base OpenLayers code. Since the latest release of OpenLayers is relatively new and ExtJS was required to follow, the code for GeoExt was still under development when the application was created.

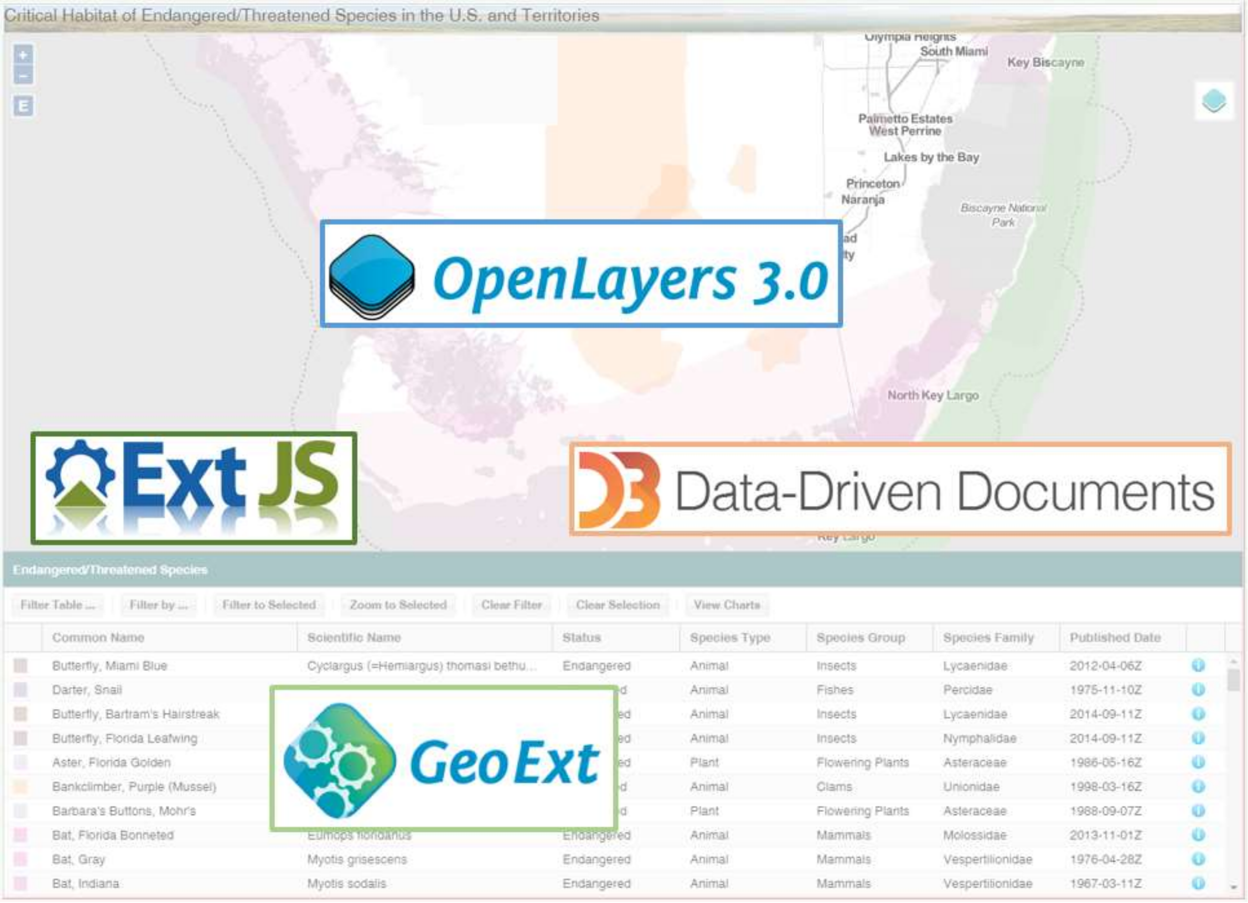


GeoExt 3.0.0, based upon ExtJS 6 and OpenLayers 3
GeoExt 2.1.0, based upon ExtJS 4, ExtJS 5 and OpenLayers 2
GeoExt 1.1, based upon ExtJS 3 and OpenLayers 2

Figure 3: OpenLayers, ExtJS, and GeoExt Version Compatibility. *The ExtJS and GeoExt libraries are built upon the base code of OpenLayers. Both must use the appropriate version to work in sync with the selected version of OpenLayers.*

The final JavaScript library used for the application was the open-source D3.js framework for data visualization via graphs and charts. D3 consumes data in JSON and CSV formats and uses HTML, SVG, and CSS to create data-driven visualization components. The limitation on file formats that D3 can read requires that data be queried from the PostGIS database in the correct row/column arrangement and saved in the JSON or CSV format before use by the D3 code.

The ability to create graphs based on spatial and attribute information provides tools that make the application more beneficial to users. Graphs showing statistics regarding species or critical habitat data alone or in combination with other datasets such as states and watersheds gives users greater power in making informed decisions. The same can be said for geospatial analysis tools. Charts make possible the visualization of information that is often necessary to complement information gleaned from a map.



Common Name	Scientific Name	Status	Species Type	Species Group	Species Family	Published Date
Butterfly, Miami Blue	Cyclargus (=Hemiargus) thomasi bethu...	Endangered	Animal	Insects	Lycaenidae	2012-04-06Z
Darter, Snail		d	Animal	Fishes	Percidae	1975-11-10Z
Butterfly, Bartram's Hairstreak		ed	Animal	Insects	Lycaenidae	2014-09-11Z
Butterfly, Florida Leafwing		ed	Animal	Insects	Nymphalidae	2014-09-11Z
Aster, Florida Golden		ed	Plant	Flowering Plants	Asteraceae	1986-05-16Z
Bankclimber, Purple (Mussel)		d	Animal	Clams	Unionidae	1998-03-16Z
Barbara's Buttons, Mohr's		d	Plant	Flowering Plants	Asteraceae	1988-09-07Z
Bat, Florida Bonneted	Eumops floridanus	Endangered	Animal	Mammals	Molossidae	2013-11-01Z
Bat, Gray	Myotis grisescens	Endangered	Animal	Mammals	Vespertilionidae	1976-04-28Z
Bat, Indiana	Myotis sodalis	Endangered	Animal	Mammals	Vespertilionidae	1967-03-11Z

Figure 4: Application Layout with Software Components. *The map area is created using OpenLayers 3, the table grid is created using GeoExt, both the map and table grid areas are ExtJS panels, and D3 is used for showing charts of the data.*

Data Formats

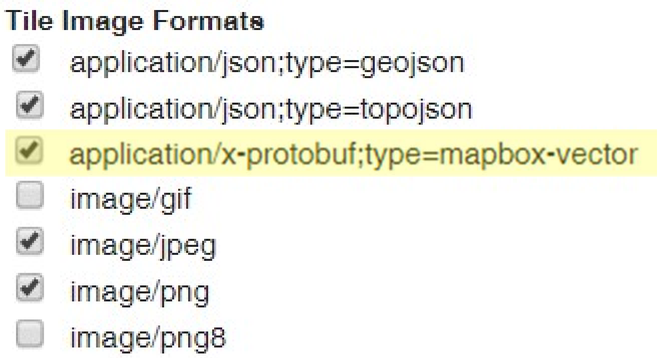
Data formats that were considered include:

- File-based content
  - KML
  - GeoJSON
  - Shapefile
- Web services
  - Web Mapping Service (WMS)
  - Web Feature Service (WFS)
- Pre-generated raster tiles
- Vector tiles

File-based content is easy to use but does not provide the necessary security measures afforded by using a mapping service. Creating raster tiles allows for no-cost data creation using FOSS tools such as TileMill to create the images. These tiles provide 1) quicker loading of the data as the images are already created and 2) the opportunity to incorporate better cartographic techniques. The downside to using preset tiles is there is less ability to perform spatial analysis with only basic queries being possible. Raster tiles require a large amount of storage space when the data covers an expansive area and when a lot of zoom levels are created.

Another concept that is increasing in availability and flexibility is vector tiles. Vector tiles are images created "on-the-fly" on the client side instead of preset images and offer more flexibility in content and analysis while still taking advantage of the tile concept for quicker display. According to the analysis results from GIS Cloud, vector tiling makes it "possible to produce more tiles per second, the size of those tiles is significantly lower and therefore less bandwidth is needed" [8] in comparison to that of raster tile images. Vector tiles allow for easily changing the style of the data since the tiles are not pre-generated. They also do not require the intense storage space that is necessary for pre-generated raster tiles.

The original intent of the application was to implement a process for using vector tiles. It is an easy process to make vector tiles available through GeoServer. From the Tile Caching – Tile Layers page, select the layer for which you want to make vector tiles available and check the option for "application/x-protobuf;type=mapbox-vector" under the Tile Image Formats section. One note is that vector tiles are only an option in the Enterprise edition of GeoServer. This edition is made available for free to students but not for commercial or government use.



Tile Image Formats

☑ application/json;type=geojson

☑ application/json;type=topojson

☑ application/x-protobuf;type=mapbox-vector

□ image/gif

☑ image/jpeg

☑ image/png

□ image/png8

Figure 5: GeoServer Tile Image Formats. *Tile Image Formats section from the Tile Caching – Tile Layers page in GeoServer showing selection of option for providing data in Mapbox vector tile format.*

In order for the highly detailed critical habitat data to draw faster, it was generalized using two different methods: 1) the bend simplify method to smooth out sharp edges and 2) the point remove method to remove extraneous points along the outline. While the application itself adhered strictly to open-source technology, the generalization was performed using Esri's ArcGIS software. Open-source options, such as PostGIS and QGIS, only provide the point remove method of generalization which did not yield the necessary smoothing required for the application data. Due to the complexity of the data and the amount of time to generalize the complete dataset, only one level of generalization was created for the application.

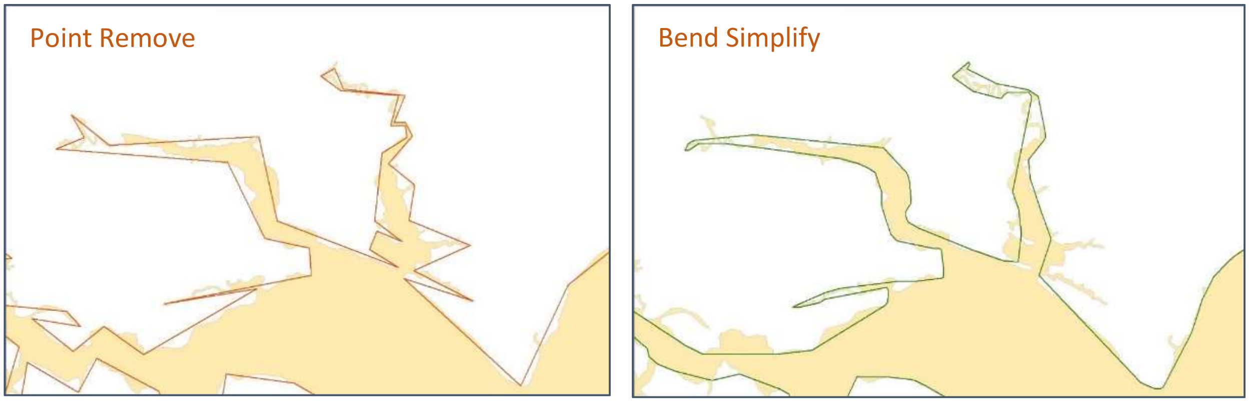


Figure 6: Point Remove and Bend Simplify Generalization Results. *The point remove method removes points at a specified distance. The bend simplify method smooths lines by removing extraneous bends.*

The other factor leading to the use of the Esri generalization toolset was the ability to check for and repair boundary changes that led to topological errors, such as self-intersecting lines. Including the option to check topology led to a much longer processing time but ensured that the generalized data was usable.

Because of the complexity of the original data, it was not possible to generalize all of the data as one file. The data was broken out into sections of the U.S. in order for the process to complete. Also, critical habitat polygons that shared boundaries were generalized together to ensure that the resulting outlines corresponded to each other.

While the intended goal was to use the vector tile format, several factors made this option not usable. First, the load time was fairly high and loading subsequent zoom levels even more time consuming. Second, GeoExt does not currently read feature and attribute information delivered in the vector tile format. Due to these factors, the use of vector tiles was not the best choice with the chosen API.

Other options regarding the format used to publish the data include using a Web Feature Service (WFS) to obtain the data in GeoJSON format, using a raw GeoJSON file stored and accessed directly from a file directory, and using pre-generated raster tiles created with the TileMill software. Accessing the data from a GeoJSON file has a really slow load time. Also, this method of accessing the data creates a cross domain error for Internet browsers that implement security measures against pulling information from different domains. Pre-generated raster tiles have the fastest load times but this format does not provide attribute information, only an image of the spatial data. The format chosen for publishing the data was the WFS/GeoJSON approach. The load time is still not acceptable, especially for public use, but does provide quick display of data after the initial load and works well with GeoExt.

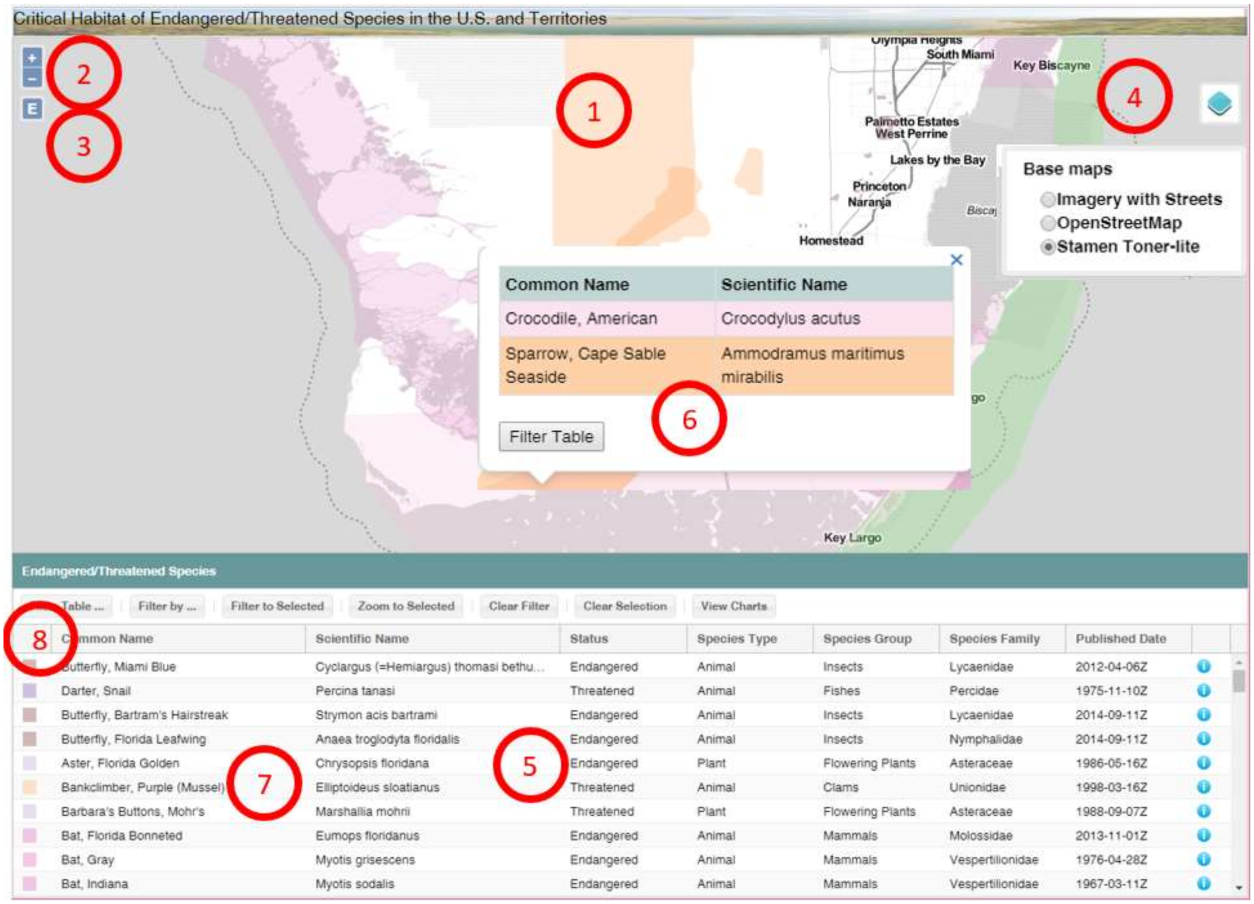
	Load	Zoom	Select
Vector Tiles	26 sec	40 sec	~~GeoExt~~
WFS/GeoJSON	**45 sec**	**<1 sec**	**<1 sec**
GeoJSON (external)	120+ sec	<1 sec	<1 sec
Raster Tiles (TileMill)	3 sec	<1 sec	No attributes

Figure 7: File Format Load Time Comparison. *Time comparison for each tested file format showing seconds for loading, zooming, and selecting features.*

Future development will include the creation of individual data layers generalized for each, or at least several, available zoom level. Based on research regarding data loading, having layers generalized for each zoom level is one of the primary methods used to improve display speed. An example of how the use of generalization improves load times can be seen by comparing the load time of the original detailed file to that of the generalized file. The generalized dataset is half the size and displays 4x faster than the original detailed file.

Basic Functionality

Based on needs analysis for the USACE and components that would make an overall useful application, the following functionality were identified. The application has a simple, user-friendly interface with more advanced functionality available for those who need it.



Base maps

○ Imagery with Streets
○ OpenStreetMap
● Stamen Toner-lite

Common Name	Scientific Name
Crocodile, American	Crocodylus acutus
Sparrow, Cape Sable Seaside	Ammodramus maritimus mirabilis

Filter Table

Endangered/Threatened Species

Table ... | Filter by ... | Filter to Selected | Zoom to Selected | Clear Filter | Clear Selection | View Charts

	Common Name	Scientific Name	Status	Species Type	Species Group	Species Family	Published Date	
■	Butterfly, Miami Blue	Cyclargus (=Hemiargus) thomasi bethu...	Endangered	Animal	Insects	Lycaenidae	2012-04-06Z	ⓘ
■	Darter, Snail	Percina tanasi	Threatened	Animal	Fishes	Percidae	1975-11-10Z	ⓘ
■	Butterfly, Bartram's Hairstreak	Strymon acis bartrami	Endangered	Animal	Insects	Lycaenidae	2014-09-11Z	ⓘ
■	Butterfly, Florida Leafwing	Anaea troglodyta floridalis	Endangered	Animal	Insects	Nymphalidae	2014-09-11Z	ⓘ
■	Aster, Florida Golden	Chrysopsis floridana	Endangered	Plant	Flowering Plants	Asteraceae	1986-05-16Z	ⓘ
■	Bankclimber, Purple (Mussel)	Elliptoideus sloatianus	Threatened	Animal	Clams	Unionidae	1998-03-16Z	ⓘ
■	Barbara's Buttons, Mohr's	Marshallia mohrii	Threatened	Plant	Flowering Plants	Asteraceae	1988-09-07Z	ⓘ
■	Bat, Florida Bonneted	Eumops floridanus	Endangered	Animal	Mammals	Molossidae	2013-11-01Z	ⓘ
■	Bat, Gray	Myotis grisescens	Endangered	Animal	Mammals	Vespertilionidae	1976-04-28Z	ⓘ
■	Bat, Indiana	Myotis sodalis	Endangered	Animal	Mammals	Vespertilionidae	1967-03-11Z	ⓘ

Figure 8: Basic Functionality. *Graphic showing where each of the basic functions are located/accessed.*

1. Display critical habitat overlaid with basemap – The map section shows the critical habitat spatial data using unique colors for each species group and fill transparency to show the basemap under the data.

2. Provide pan/zoom tools – Basic pan and zoom functionality is included. Panning is the default cursor tool and works when the user clicks and drags the map. Zooming in/out is accomplished through either the mouse wheel or the plus/minus buttons in the upper left. Users can also draw a box to zoom in to by holding the Shift key while dragging a box around the area of interest.

3. Include button to zoom to full/original extent – Clicking the “E” (extent) button zooms the map to the original extent.

4. Provide multiple basemaps and tool to change basemap selection – Hovering over the basemap tool shows three options for imagery, OpenStreetMap, and the light toner basemap from Stamen.

5. Display table of all species – The table contains a list of all species within the U.S. that are classified as endangered or threatened, not just those species with a designated critical habitat.

6. Select critical habitat feature from map and highlight associated record in table – Clicking on the map over critical habitat features brings up a popup window identifying the features found at that location. Holding the Ctrl button while dragging a box will select all critical habitat features that are found within the area of interest as well as include species found within the state that do not have a defined critical habitat. Clicking the "Filter Table" button on the popup will filter the table rows to those species included in the popup. Filtering the table will also filter the features displayed on the map.

7. Select species from table, highlight critical habitat on map, flash upon initial selection, and highlight associated record in table – Users can select one or more rows (holding down the Shift key to select more than one) from the table. Selected features will flash when each row is selected and will be highlighted on the map.

8. Provide legend for map layer color scheme – The colors used for the critical habitat data are based on the species group value. The color is coordinated between the features on the map, the icon on each row, and in the popup window showing features that were selected.

Advanced Functionality

Using the built-in capabilities of GeoExt, a toolbar with buttons was added to the panel containing the grid table. Functions were written for each of the buttons to perform advanced filtering and data display.

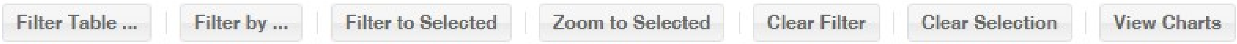
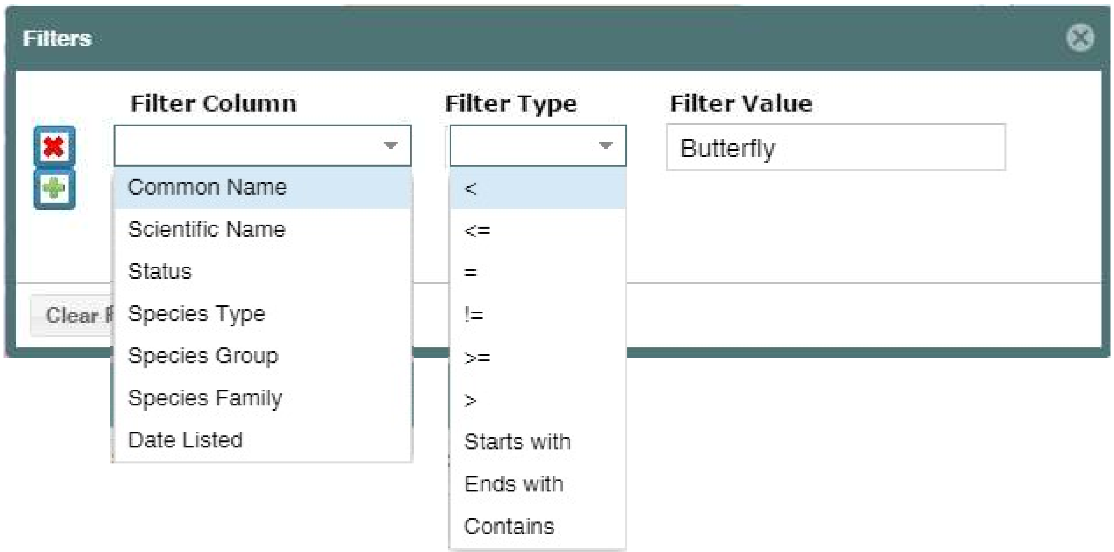


Figure 9: Advanced Functionality. *Toolbar within table grid containing advanced functionality.*

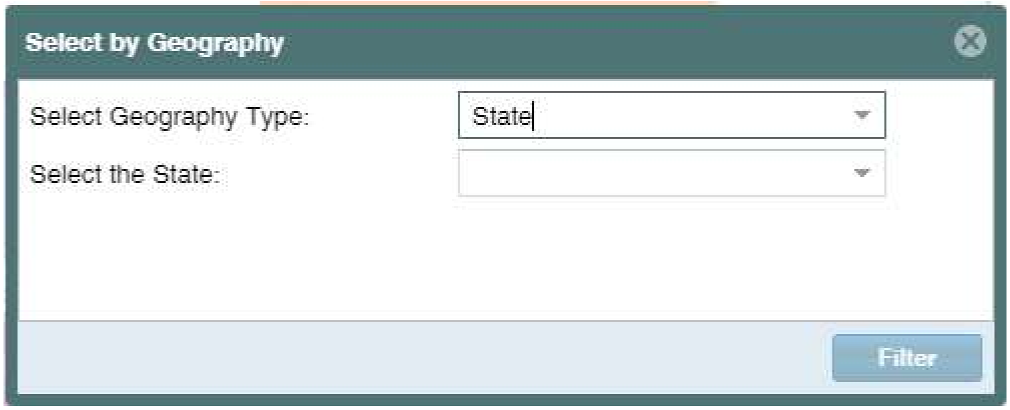
1. Filter Table … – This button opens a new window providing filtering options. Users can filter the rows displayed in the table by choosing columns, operators, and entering values. Features on the map are filtered to only display for those rows matching the table filter.



Filter Value: Butterfly

***Figure 10: Filter Table.** The Filter Table window provides options for how the table is filtered.*

2. Filter by … - This button opens a new window for filtering the table rows based on the geography in which they interact. Options include choosing species found within a state or within a 10-digit HUC watershed (HUC10). Information for the state and HUC10 in which a species is found is included in the database as separate lookup tables that are related to the species table.



Select Geography Type: State

Select the State:

***Figure 11: Filter Table.** The Filter Table window provides options for how the table is filtered.*

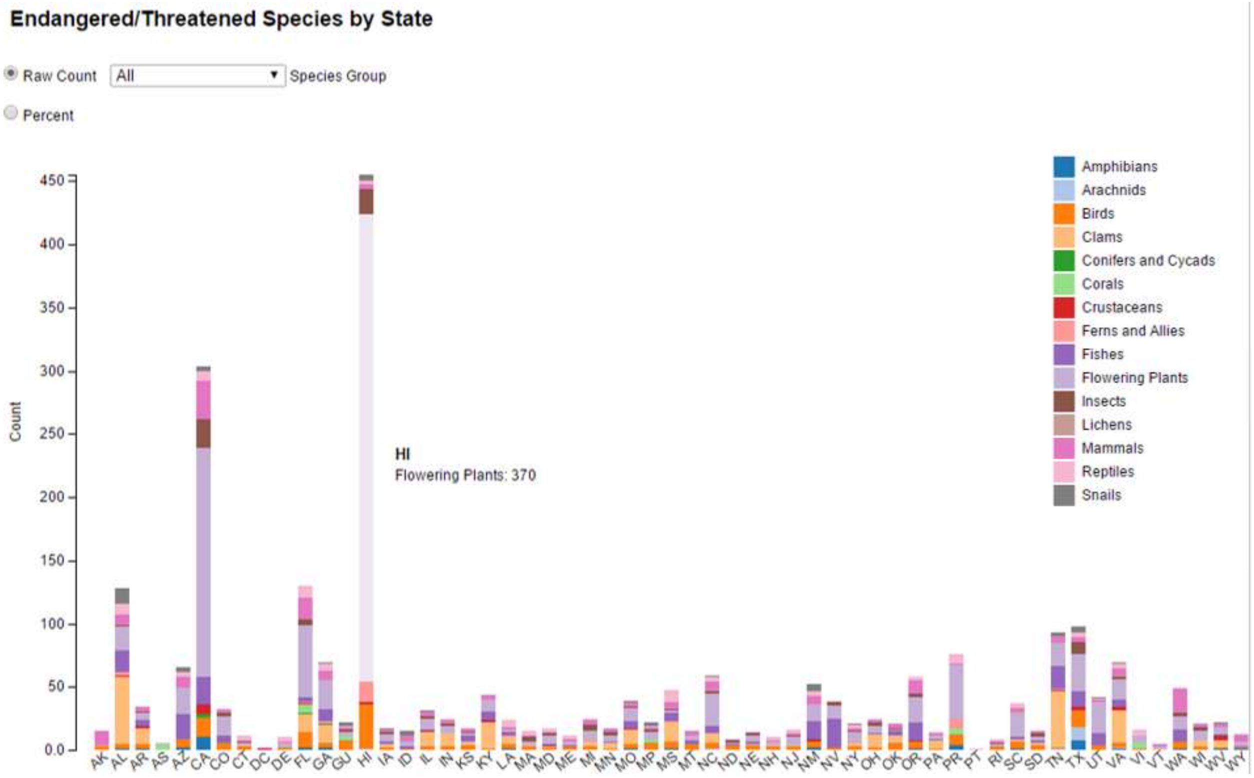
3. Filter to Selected – When one or more rows in the table are selected, clicking this button will filter the table to just those selected rows. Features on the map are filtered accordingly.

4. Zoom to Selected – Clicking this button will zoom the map into the area where the selected features are located.

5. Clear Filter – This button will clear any filtered rows and return the table to the original content.

6. Clear Selection – This button will clear all rows from the selection set.

7. View Charts – This button opens a new browser tab displaying the graphics created using the D3 JavaScript library. Users can view the raw count of each species group by state, the percent of each species group by state, or the raw count of a selected species group by state. Hovering over any stack in the bar will display the count for that block. The color scheme used for the chart matches the colors used to display the features on the map.



Endangered/Threatened Species by State

● Raw Count All ▾ Species Group

○ Percent

__Figure 12: Results of View Chart.__ The graph displayed after clicking the View Charts button initially loads showing the raw count of each species group by state. Hovering over a block will display the state, species group, and count as shown here for Hawaii containing 370 species of flowering plants on the ESA list.

Data Display

The complexity of the critical habitat spatial data makes showing each individual polygon difficult. There can be 20+ polygons for different species sharing the same or overlapping locations. This overlap is reflected on the map by showing the boundaries for each critical habitat area in a darker color than the fill as to make it apparent that there is a delineation. However, species sharing the exact same area of critical habitat also share the same boundary which makes distinguishing each area problematic.

Several methods were examined for the best approach to display the data. First, a basic outline could be used but this approach does not make clear which side of the boundary is the actual interior of the polygon, especially with irregular shapes. The second option was to use a hatch pattern. This takes care of making apparent the interior of the polygon but hatches on top mask any underlying hatches for overlapping polygons. A third method is using an opaque fill so that overlapping polygons are identifiable and the underlying basemap is still visible. The last approach requires creation of an internal buffer of the polygon boundary which is then used in conjunction with the original polygon boundary. This method works fairly well at displaying the outline while identifying the interior side of the polygon. The caveat with this approach is that the buffered polygon is only the width of the buffer created, not the entirety of the polygon, which does not allow for selecting the polygon other than within the buffer. Based on these tests, the method chosen for displaying the critical habitat polygons was using an opaque fill.

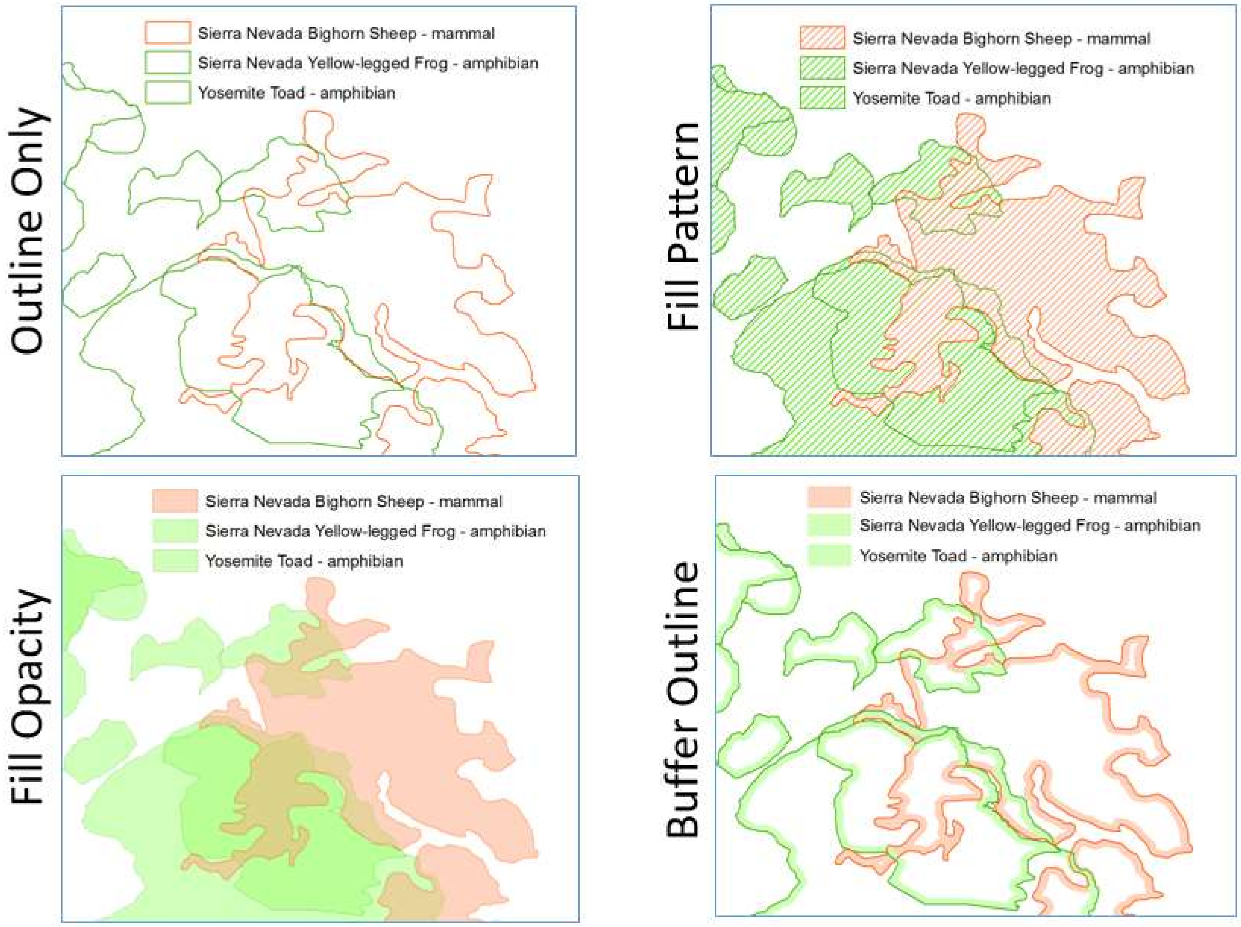
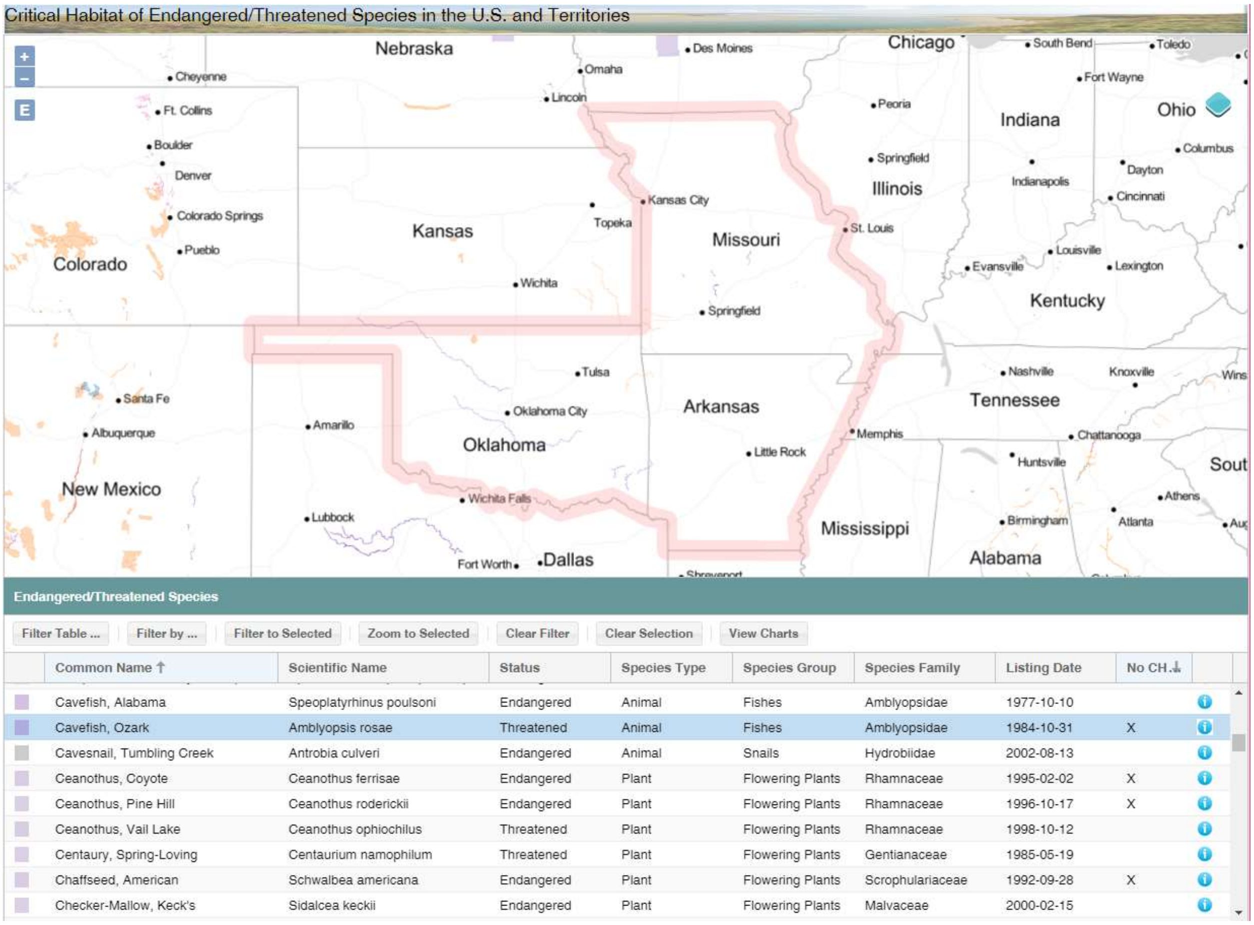

Outline Only
Sierra Nevada Bighorn Sheep - mammal
Sierra Nevada Yellow-legged Frog - amphibian
Yosemite Toad - amphibian
Fill Pattern
Sierra Nevada Bighorn Sheep - mammal
Sierra Nevada Yellow-legged Frog - amphibian
Yosemite Toad - amphibian
Fill Opacity
Sierra Nevada Bighorn Sheep - mammal
Sierra Nevada Yellow-legged Frog - amphibian
Yosemite Toad - amphibian
Buffer Outline
Sierra Nevada Bighorn Sheep - mammal
Sierra Nevada Yellow-legged Frog - amphibian
Yosemite Toad - amphibian

Figure 13: Display Options. *Each of the four options shown above were tested to determine the best approach for displaying the critical habitat areas. The two species shown in green are the same color since they are the same species group.*

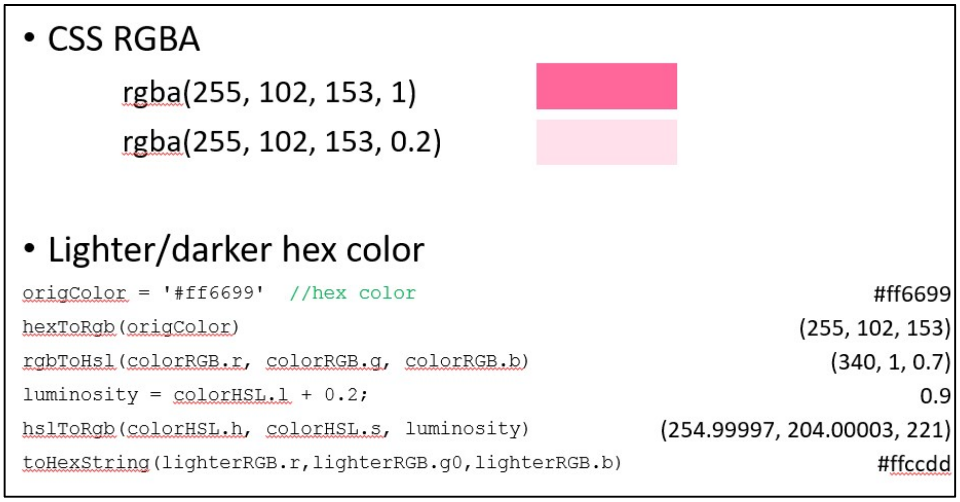
The next hurdle was to define a process to best show where species are located when they do not have a designated critical habitat. The only spatial information known about these species is the state in which they can be found. The boundaries for each of the states in which a species was found were merged to create a polygon for that species. This multi-state boundary was then used to display spatial information about the species as a highlighted boundary showing the state(s) in which the species can be found when the species is selected from the table.



	Common Name	Scientific Name	Status	Species Type	Species Group	Species Family	Listing Date	No CH.	
	Cavefish, Alabama	Speoplatyrhinus poulsoni	Endangered	Animal	Fishes	Amblyopsidae	1977-10-10		ⓘ
	Cavefish, Ozark	Amblyopsis rosae	Threatened	Animal	Fishes	Amblyopsidae	1984-10-31	X	ⓘ
	Cavesnail, Tumbling Creek	Antrobia culveri	Endangered	Animal	Snails	Hydrobiidae	2002-08-13		ⓘ
	Ceanothus, Coyote	Ceanothus ferrisae	Endangered	Plant	Flowering Plants	Rhamnaceae	1995-02-02	X	ⓘ
	Ceanothus, Pine Hill	Ceanothus roderickii	Endangered	Plant	Flowering Plants	Rhamnaceae	1996-10-17	X	ⓘ
	Ceanothus, Vail Lake	Ceanothus ophiochilus	Threatened	Plant	Flowering Plants	Rhamnaceae	1998-10-12		ⓘ
	Centaury, Spring-Loving	Centaurium namophilum	Threatened	Plant	Flowering Plants	Gentianaceae	1985-05-19		ⓘ
	Chaffseed, American	Schwalbea americana	Endangered	Plant	Flowering Plants	Scrophulariaceae	1992-09-28	X	ⓘ
	Checker-Mallow, Keck's	Sidalcea keckii	Endangered	Plant	Flowering Plants	Malvaceae	2000-02-15		ⓘ

Figure 14: Selected Species with no Critical Habitat. *This map shows the technique used to show the states in which a species is found when the species does not have a designated critical habitat. As shown above, the Ozark Cavefish does not have a critical habitat area defined but can be found in the states of Missouri, Oklahoma, and Arkansas.*

After determining the method for displaying the data, unique colors for each species group were defined and stored in a lookup table relating each species group to a hex color. This information was then related to each species row in the database table for use in OpenLayers. Opacity levels in OpenLayers 3 for polygon style settings is applied to both the fill and outline. For the purposes of this map, using an opacity on the border made it hard to distinguish from other borders. The alternative was to use the CSS capability of assigning an RGBA color value. This is the standard RGB (red, green, blue) color definition with the addition of A (alpha) to set a transparency value. The RGBA value was then set for the fill while leaving the boundary at 100% of its color. While using the RGBA route worked for the styling, it did not when applied to the popup table. For the coloring used to show the information in the popup with the same level of opacity, the original hex color was converted to RGB then to HSL (hue, saturation, luminosity), the L value was increased and then converted back to RGB, and finally converted to the hex value that equates to the level of transparency applied to the polygons using the RGBA setting.



- CSS RGBA

```
rgba(255, 102, 153, 1)
rgba(255, 102, 153, 0.2)
```

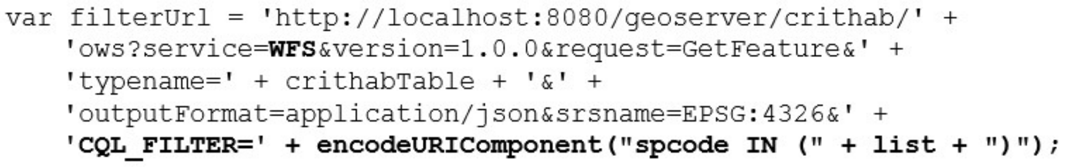
- Lighter/darker hex color

```
origColor = '#ff6699'  //hex color                                    #ff6699
hexToRgb(origColor)                                           (255, 102, 153)
rgbToHsl(colorRGB.r, colorRGB.g, colorRGB.b)                    (340, 1, 0.7)
luminosity = colorHSL.l + 0.2;                                            0.9
hslToRgb(colorHSL.h, colorHSL.s, luminosity)       (254.99997, 204.00003, 221)
toHexString(lighterRGB.r,lighterRGB.g0,lighterRGB.b)                  #ffccdd
```

Figure 15: Defining Transparency. *Two methods used to define transparency and lighter hex color.*

Map and Table Filters

Filtering of the map and table used different processes. Map filtering worked by using a CQL filter on the URL calling the WFS. CQL stands for Common Query Language and is "a query language created by the OGC [Open Geospatial Consortium] for the Catalogue Web Services specification."[9] It is intended to be human readable and is similar in structure to SQL (Structured Query Language). For filtering the map features, a list of all species codes that were filtered in the table was sent to a function that updated the layer source using this list within the CQL_FILTER option of the WFS URL. This method can also be applied to a WMS. The ability to update the layer source on the fly in OpenLayers made it possible to refresh the information that was displayed on the map.



```
var filterUrl = 'http://localhost:8080/geoserver/crithab/' +
    'ows?service=WFS&version=1.0.0&request=GetFeature&' +
    'typename=' + crithabTable + '&' +
    'outputFormat=application/json&srsname=EPSG:4326&' +
    'CQL_FILTER=' + encodeURIComponent("spcode IN (" + list + ")");
```

Figure 16: WFS URL with CQL Filter. *The code shows the URL used to call the WFS from GeoServer with the CQL filter option.*

Filtering the information in the table took advantage of the built-in GeoExt *filterBy* method. This method accepts an array of values and uses these values to capture the rows matching the filter criteria. The "Filter Table" button, the "Filter by" button, the "Filter Table" button on the popup, and the "Filter to Selected" button each stores the appropriate species code values in an array which then gets sent to a function using the *filterBy* method to limit the rows displayed in the table grid to the values in the array.

Future/Where to Next

While the basic functionality and core analysis is incorporated into the web mapping application, there are still a few additional components that will improve the usability and information obtained. These include:

- Multiple levels of generalization – Create multiple levels of generalized critical habitat data to display at different zoom levels to improve the initial and subsequent load times.
- Additional analysis layers – The current application uses spatial analysis to determine the overlap of critical habitat with states and HUC10s. This will be expanded to include land cover and impervious surface cover from the National Land Cover Database (NLCD) and population data from the U.S. Census Bureau.
- Search by address/coordinates – New ways of searching critical habitat and moving the focus of the map will be included by allowing users to enter an address or coordinates, zooming the map to that location, and selecting the critical habitat areas that coincide with the address location or coordinates.
- Expand charts – The current charts rely solely on counts and percentages of species groups by states. Other charts to be created are:
    - Species added by year using the listing date of the species to the ESA list
    - Species by size of critical habitat
    - Critical habitat to land cover type
    - Critical habitat to percent of impervious surface cover
    - Acres of critical habitat by state
- Alaska/Hawaii insets – The possibility of adding inset areas within the map panel to show Alaska and Hawaii in separate windows will be examined.
- Resize map/table – The application has static percentages of the web page area covered by the map and table panels. This will be updated to allow the user to increase the height of one while decreasing the size of the other while ensuring they both continue to fit within the web page dimensions.
- Move application to USACE server – The current application resides on a personal computer. The data and application code will need to be ported over to an internal USACE server to make the application available to USACE personnel.
- Obtain permission to release to public – The viewer is not only helpful to the USACE. Other government agencies and the public are interested in accessing this information. Approval of making the application publicly available will be determined.
- Maintenance – The critical habitat data does not change very frequently. However, changes to the full list of species under the ESA does change. These additions or deletions will need to be tracked and a periodic update to the data will need to be made. Maintenance also includes keeping the code running. No version updates are necessary, an advantage of using FOSS. The code will include comments so that it can be transferred and edited by other programmers in the future.

References

1. 40 CFR 230.3, *Clean Water Rule: Definition of "Waters of the United States" Part 230.*
2. 16 U.S. Code § 1531-1544, *Endangered Species Act of 1973.*
3. 16 U.S. Code § 1532(5)(A)(i), *Endangered Species – Definitions***.**
4. Martin F.J. Taylor et al., *The Effectiveness of the Endangered Species Act: A Quantitative Analysis*, 55 BioScience 360, 360-67 (2005).
5. U.S. Fish & Wildlife Service. "Critical Habitat – What is it?" https://www.fws.gov/midwest/endangered/saving/CriticalHabitatFactSheet.html. Accessed September 2, 2016.
6. *Critical Habitat for Threatened & Endangered Species [USFWS]* web map application http://fws.maps.arcgis.com/home/webmap/viewer.html?webmap=9d8de5e265ad4fe09893cf75b8dbfb77
7. GeoExt. *"GeoExt – JavaScript Toolkit for Rich Web Mapping Applications."* http://www.geoext.org. Accessed October 25, 2016.
8. GIS Cloud. "*Realtime Map Tile Rendering Benchmark: Rasters vs. Vectors*." <http://www.giscloud.com/blog/realtime-map-tile-rendering-benchmark-rasters-vs-vectors/>. Accessed June 13, 2016.
9. Open Source Geospatial Foundation. "*GeoServer 2.10x User Manual – CQL and ECQL*." < http://docs.geoserver.org/stable/en/user/tutorials/cql/cql_tutorial.html>. Accessed October 26, 2016.