MPCS 53014 - Big Data Application Architecture
Final Project Write-Up

Shane McVeigh smcveigh

1 Summary

My goal for the final project was to build a bird dataset based on the Great Backyard Bird Count, an Audobon Society initiative that occurs every February and allows people to submit bird sightings from their backyard. I used a dataset from GBIF that contains over 7 million occurrences of bird sightings from the years 1998 to 2009. The dataset was 3.3 GB in size. I then matched this with a biome shapefile using a python script that gave a bird table based on what biome they were sighted in, based on the latitude and longitude from the GBIF dataset. Ultimately, I built my web application to have three primary endpoints: a sightings by state query page, a sightings by biome query page, and a "submit new sighting" page that wrote to a speed layer.

2 Data Preparation

As stated above, I used a GBIF dataset to use for my bird sightings data lake ground truth. This came in the form of a CSV file around 3.3 GB in size. There was minimal data cleansing required for this dataset.

The other primary source file I used for my application was a biome shapefile from the US EPA, which filtered down biomes to level 3 specificity. This shapefile couldn't directly match with the latitude longitude without some data preparation, as it would have needed to query the shapefile each time otherwise.

My next step was to ingest the bird data from GBIF into a HQL table on the cluster. I created the `ingest_birds.hql` script which first mapped each column of the CSV data to Hive, using a tab-delimited row format (no SerDes were required since the data was relatively clean). After creating the initial Hive table, I copied that to an ORC table for more efficient data lookup. This table was `smcveigh_birds`.

After ingesting the GBIF bird data, I needed to map each bird sighting to a biome using the latitude and longitude columns from the GBIF table. The first thing I needed to do in order to accomplish this was to create two new columns for my eventual next Hive table, `smcveigh_birds_and_biomes`, which were rounded latitude and rounded longitude. Creating these and rounding the raw latitude and longitude allowed me some level of precision for the eventual biome mapping.

I used the biome shapefile mentioned above and developed a python script (`process_shp.py`) that used the `geopandas` library to map every possible 0.1-rounded latitude and longitude coordinate pair within the bounds of the North American continent to map all of those pairs to a biome based on the shapefile. This took about an hour to run, but resulted in a 1.5 GB csv file that mapped every lat-long pair to a biome. Once this was available, I sent this to the cluster as well, and created the ORC table `smcveigh_biomes` with `ingest_biomes.hql`, to be joined to the birds table.

Once I had the two primary source ORC tables for birds and biomes, I joined them to create the primary table I would use going forward, the `smcveigh_birds_and_biomes` table in `join_birds_to_biomes.hql`. The process for this was to use the rounded latitude and longitude columns I created earlier to map to the biomes table, which resulted in every bird sighting having an associated biome in my new table.

3 Batch Views

Once I had the birds and biomes table available, my next step was to create several batch views that would ultimately be surfaced in the web application. I created three primary batch views in `create_hbase_views.hql`:

- Bird Sightings by State
- Bird Sightings by Biome
- Bird Sightings by State and Year

I first created these views as Hive tables, then created an associated HBase table and copied the hive table to HBase. I did specify some SerDes to assist with the process of creating the HBase tables that can be viewed in the hql script referenced. These were all run on the cluster. Once I had the HBase tables, I was ready to serve my batch views.

4 Speed Layer

The speed layer uses a kafka topic `smcveigh_bird_observations` that then writes to an HBase table designed for the speed layer `smcveigh_birds_by_state_speed`, which was created and designed in `create_kafka_table.hql`. I then used the speed-layer-weather archetype .scala script as a template for creating my `StreamBirdObservations.scala` (within the `smcveigh-speed-layer-birds` folder in github repo). This .scala script ultimately writes back to the speed table with an increment for each time a bird sighting is entered in the web application. This updates the HBase table in real-time so that an end user on the web application can see the influence of their bird sightings on the data.

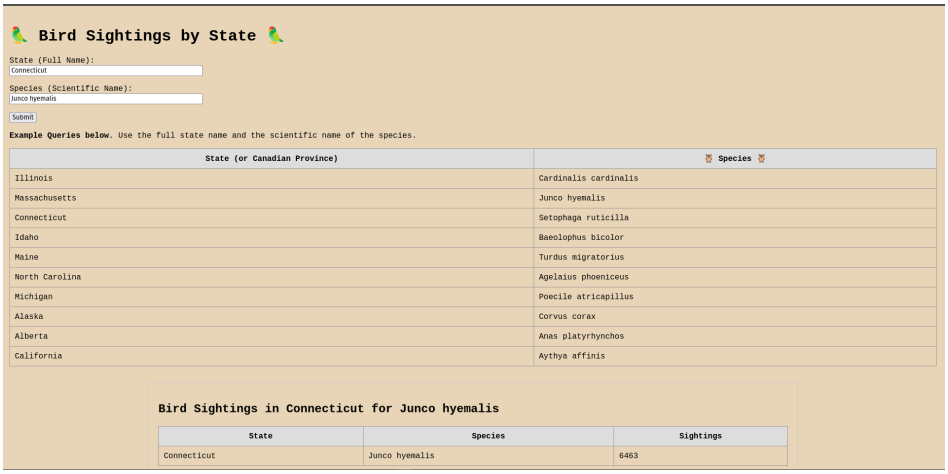
5 Web Application

My goal for the web application was to have three primary endpoints:

- Bird Sightings by State (`index.html`)
- Bird Sightings by Biome (`biome.html`)
- Submit Bird Sighting (`submit-birds.html`)

The Bird Sightings by State and Bird Sightings by Biome are both serving layer views that query the HBase speed layer table mentioned in the previous section. I did originally have these pages pulling from a more static view. However, I elected to have them serve the speed table so that users could see their updates in real-time. I did ensure that from a design perspective, I still had the previous static tables `smcveigh_birds_by_biome_hbase` and `smcveigh_birds_by_state_hbase` that could be re-surfaced due to any issues with the speed layer. The speed layer table does not currently write back to a "ground truth" table, so as to keep the integrity of the ground truth. Instead, long-term, I think it would make sense to re-write the speed layer table with batch data from the two tables I mentioned above. That way, users could see their real-time updates, but it could be validated later.

The Web Application uses the concepts from the example flights and weather application, including the html pages, the mustacche template, and the use of node to run the `app.js` file. I made some design decisions with the html for a more streamlined look, and kept the results iframe for the query. Ultimately, the web application does include all three of those endpoints, with screenshots below. The web application also includes some references to make the data querying easier. State names should be in full, and species names are the scientific names of the species. I included some sample queries on each page in order to help the user. A full list of biome names is linked in the biome page for users who are curious about particular biomes.



Bird Sightings by State

State (Full Name):
Connecticut

Species (Scientific Name):
Junco hyemalis

Submit

Example Queries below. Use the full state name and the scientific name of the species.

State (or Canadian Province)	Species
Illinois	Cardinalis cardinalis
Massachusetts	Junco hyemalis
Connecticut	Setophaga ruticilla
Idaho	Baeolophus bicolor
Maine	Turdus migratorius
North Carolina	Agelaius phoeniceus
Michigan	Poecile atricapillus
Alaska	Corvus corax
Alberta	Anas platyrhynchos
California	Aythya affinis

Bird Sightings in Connecticut for Junco hyemalis

State	Species	Sightings
Connecticut	Junco hyemalis	6463

Figure 1: Index Page

Bird Sightings by Biome

Biome (Full Name):
Central Corn Belt Plains

Species (Scientific Name):
Junco hyemalis

Submit

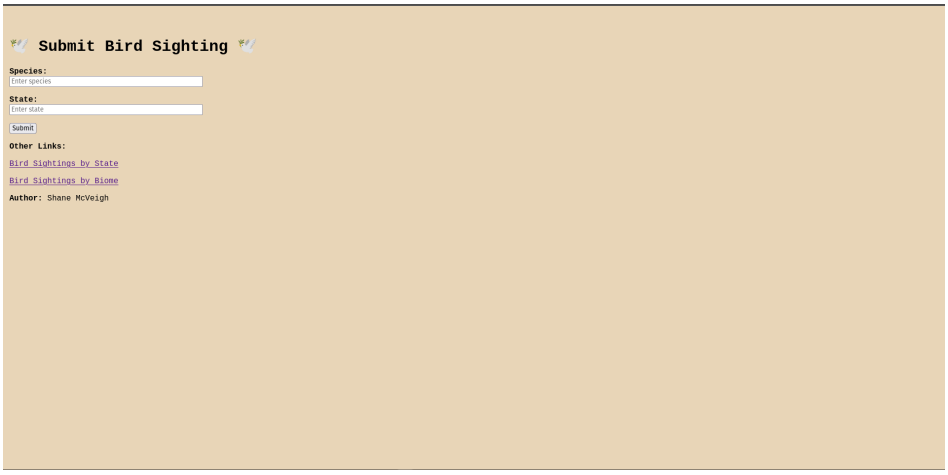
Example Queries below. A full list of the biome names and map are included in this pdf document: North American Level III Ecoregions (PDF)

Biome	Species
Central Corn Belt Plains	Cardinalis cardinalis
Central Corn Belt Plains	Junco hyemalis
Ozark Highlands	Setophaga ruticilla
Ozark Highlands	Baeolophus bicolor
Atlantic Coastal Pine Barrens	Turdus migratorius
Atlantic Coastal Pine Barrens	Agelaius phoeniceus
California Coastal Sage, Chaparral, and Oak Woodlands	Poecile atricapillus
California Coastal Sage, Chaparral, and Oak Woodlands	Corvus corax
South Central Plains	Anas platyrhynchos
South Central Plains	Aythya affinis

Bird Sightings in Central Corn Belt Plains for Junco hyemalis

Biome	Species	Sightings
Central Corn Belt Plains	Junco hyemalis	7811

Figure 2: Biome Page



Submit Bird Sighting

Species:
Enter species

State:
Enter state

Submit

Other Links:

Bird Sightings by State

Bird Sightings by Biome

Author: Shane McVeigh

Figure 3: Submit Page