Arctic Explorer Process Book

Ethan Ransom & Dylan Wootton

Table of Contents

Overview and Motivation

The concentration of sea ice plays an important role in creating global climate models due to its ability to influence solar radiation absorption. As such, climate researchers are interested in understanding patterns of change in ice concentration.

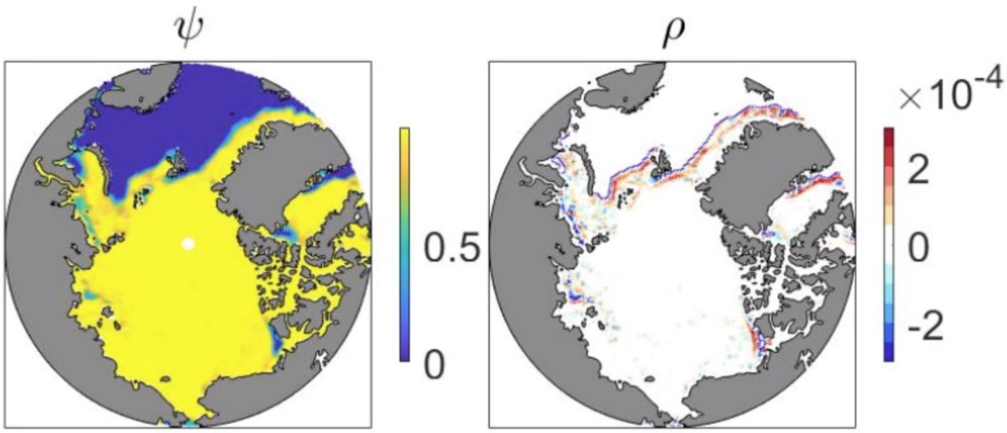
Our tool, Arctic Explorer, displays data relevant to the concentration of sea ice. This tool helps researchers understand both the spatial and temporal nature of sea ice change. Ice data has traditionally been viewed using geographic projections. While visually engaging, these maps often make it difficult to explore temporal trends in the data.

By providing both a spatial view and a temporal view of data, Arctic Explorer allows researchers to understand how sea ice changes over time. Our collaborators in the Golden Lab, have expressed difficulty understanding temporal patterns in the data. This tool was built to address this aim.

Using these patterns, our collaborators plan to adjust models to account for how the Arctic impacts the global climate. Such improvements would provide a better prediction of climate change and inform strategies to prepare for such changes.

Related Work

We were introduced to the dataset and problem space by the Golden lab. During a scoping meeting with a researcher in the Golden Lab, Delaney Mosier, we were introduced to the data in the form of a video recording of concentration values over the course of a year. Below is a screenshot from this video.



Questions

Our project currently aims to answer the following questions:

- How does global sea ice concentration change over months in a year? In years in a decade?
- Do temporal patterns exist in total concentration? Do temporal patterns exist at specific latitudes and longitudes?
- How do more recent melting trends compare to older trends? Is sea ice melting faster now? Is it growing slower? If so, by how much?

The following questions were our past question list:

- How does global sea ice concentration change over months in a year? Over years in a decade?
- On average how does sea ice concentration change throughout a year? Throughout a decade?
- Are there specific locations that are changing more rapidly than others? Do they exhibit any unique temporal patterns?
- At a given latitude and longitude, what months of a year is that location free of ice? Is that pattern consistent across years?

The original questions were created from an initial interview with Delaney. From this meeting, we realized that the lab currently lacked the functionality to analyze the temporal aspects of the sea ice data. However, during our exploratory data analysis, we realized that it was necessary to be more specific with the questions we were asking. This included specifying which aspect of the data we should analyze (ie total concentration, average concentration, total coverage, *etc*...) and how we would analyze it (by comparing it to older time points, by looking at the values isolated, *etc*...).

Data

Data was obtained from the NOAA Sea Ice Extent Dataset. The data was obtained from the FTP server via Batch Link Downloader. A MatLab script, SateliteData.m was obtained from the researchers and modified. This involved parsing the .nc file and transforming it into JSON encodings.

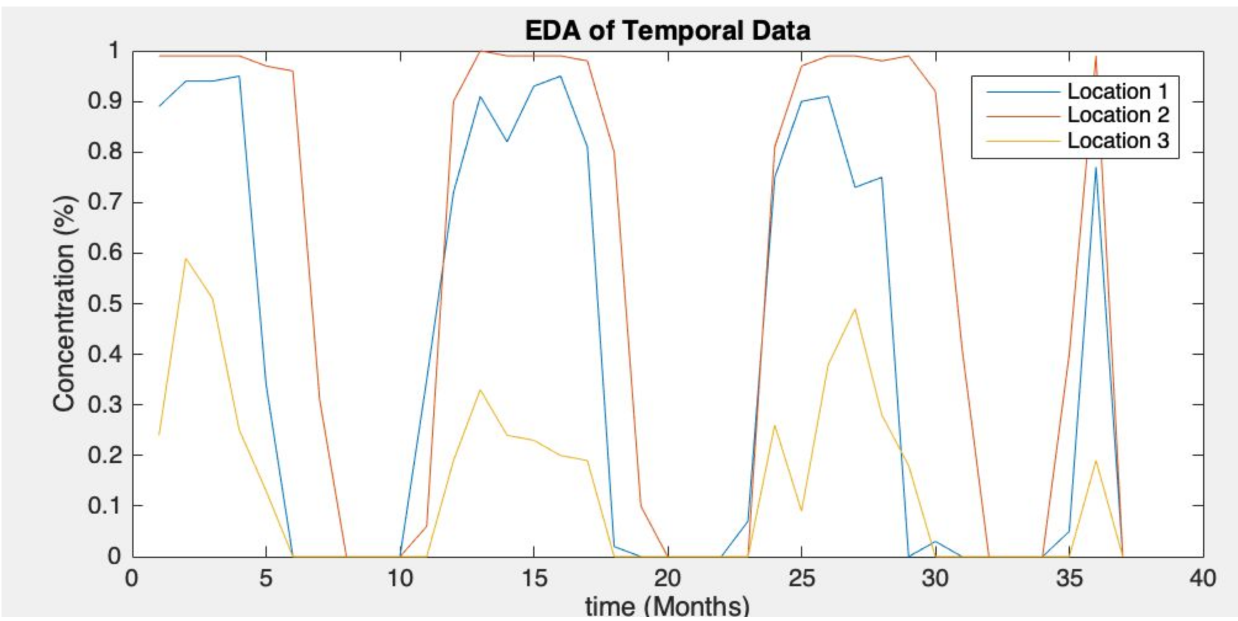
Two data structures were built using these maps. The first one is a three dimensional JSON Matrix. Each 'slice' of the matrix represents the 2D map of concentration values at given latitudes and longitudes. This representation is ideal for displaying the concentration on a map as it is easy to access all of the concentration values at a given time.

The second representation was to build an object that contained one field per “gridsquare”, with each field representing one latitude and longitude pair. These fields store the time series values for that location. This representation is ideal for displaying how concentration changes over time. Additionally, this representation requires less space as we were able to compress many of the arrays due to repeated values. For example, the values that were far away from the arctic had 0% concentration year round.

Exploratory Data Analysis

We used Delaney's ice concentration analysis an an initial visualization. This animation confirmed the quality of the dataset. It was a continuous, gap-free sampling of ice concentrations with very high temporal and spatial resolution. As the animation plays through the course of a year, we could clearly see the retreat and advance of the ice through summer and winter, respectively.

Additionally, exploratory analysis was conducted on the temporal aspect of the data. Using Matlab, a simple script was used to visualize one latitude and longitudes data over a given time period. A compilation of 3 location's time series data is displayed:

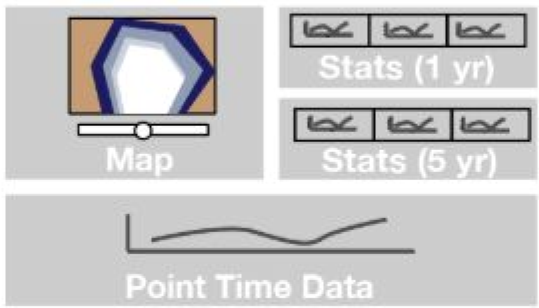


From both of these explorations, we could see that there were strong seasonal differences between the summer and winter months. Given the periodicity of this, we decided to implement a feature to let our user select which allows for them to select the datapoints they want to view (such as only view data from January, only view data from the winter months, or only view data from a given year).

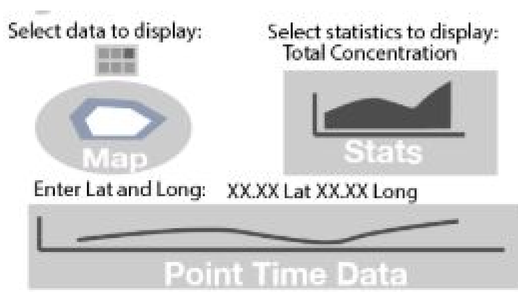
Design Evolution

Throughout the design process, we've iterated through a variety of designs.

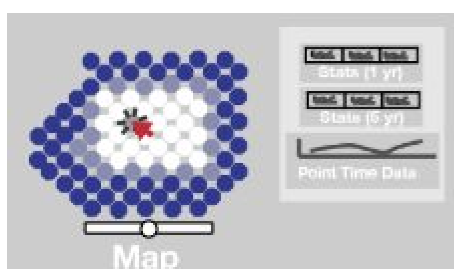
Project Proposal Designs:



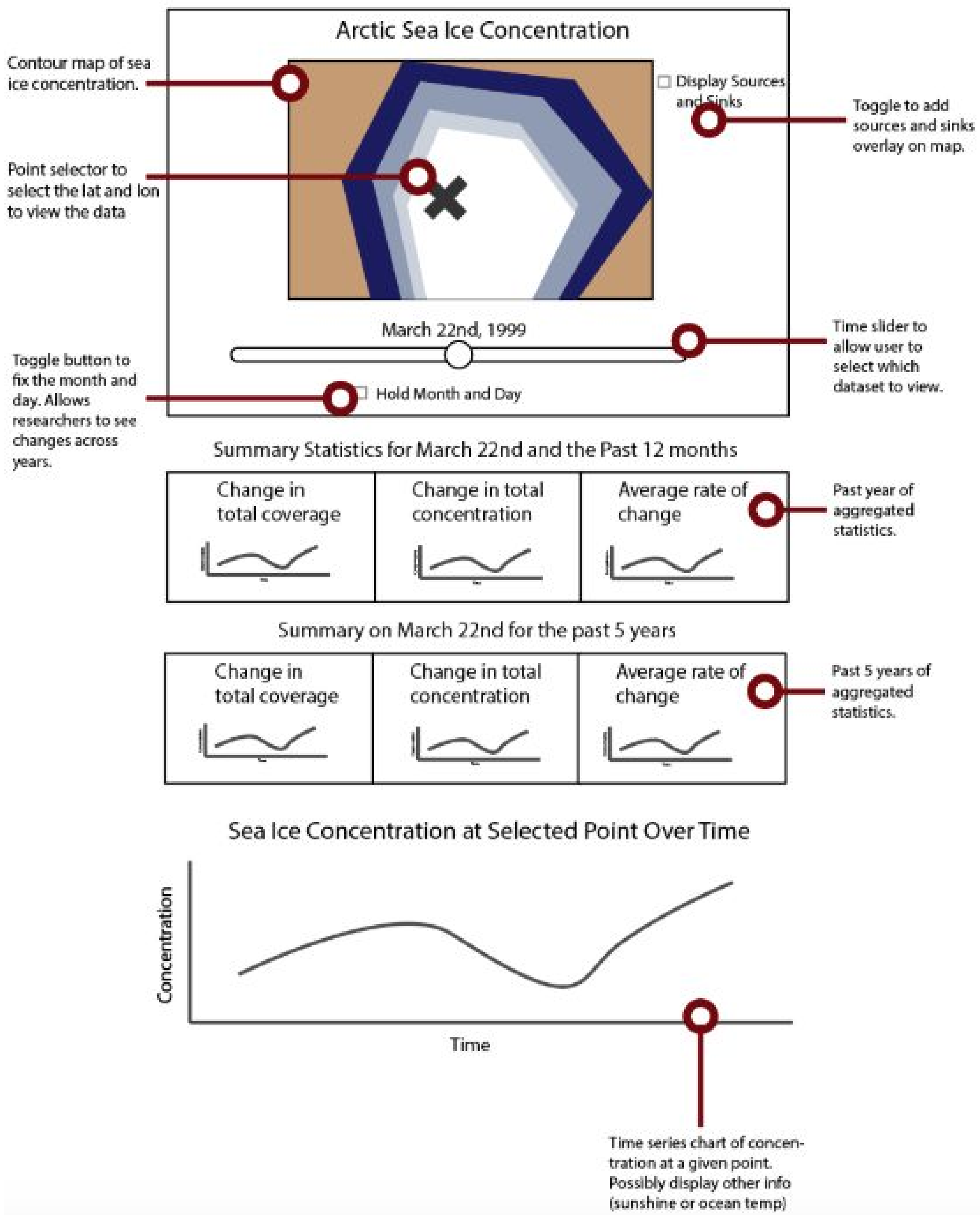
This design was our original design after meeting with Delaney. It captured the fundamental features that are necessary to convey relevant information to the user.



This design was another iteration that we believed would offer more control to the user. Rather than have the slider view (which could make it difficult to select a given data point), we provide the user with the ability to click a calendar and select the specific time point they're interested in. While we didn't utilize this approach in our final proposal design, it was the starting point for our 'heat map' data selection tool that is in our current design.

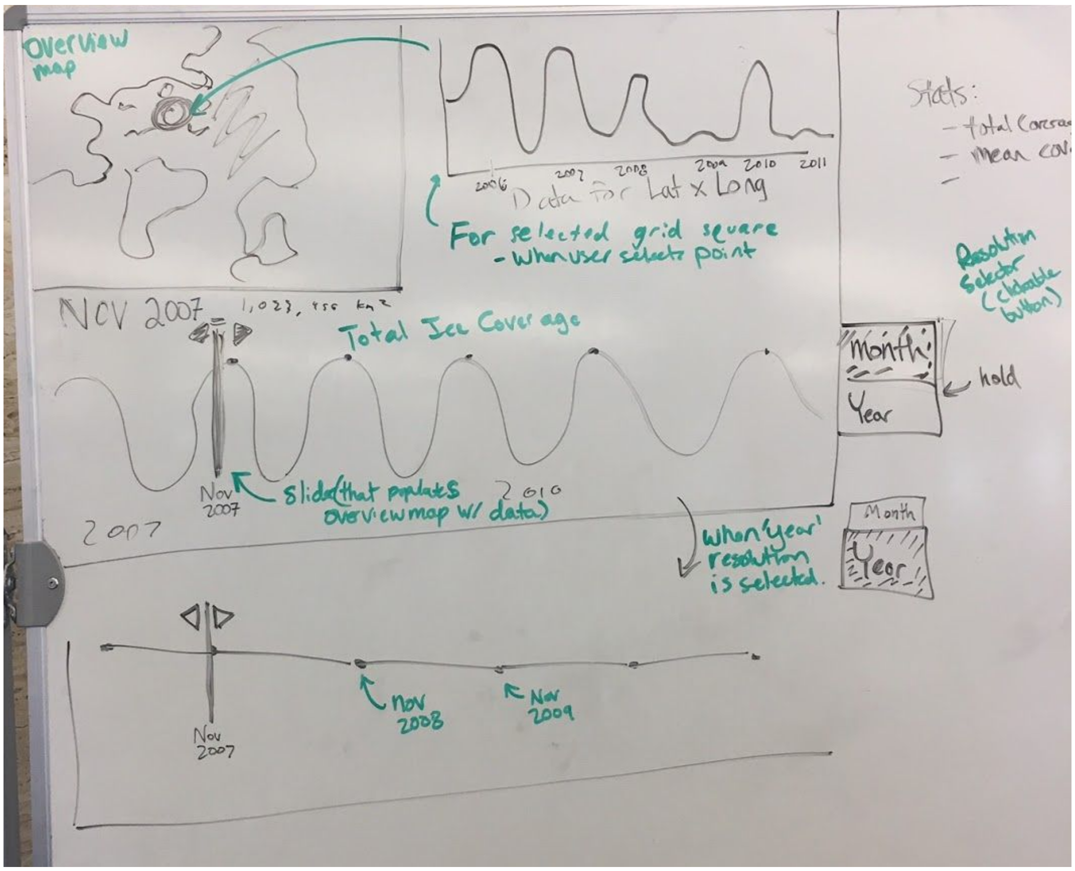


Draft Design:



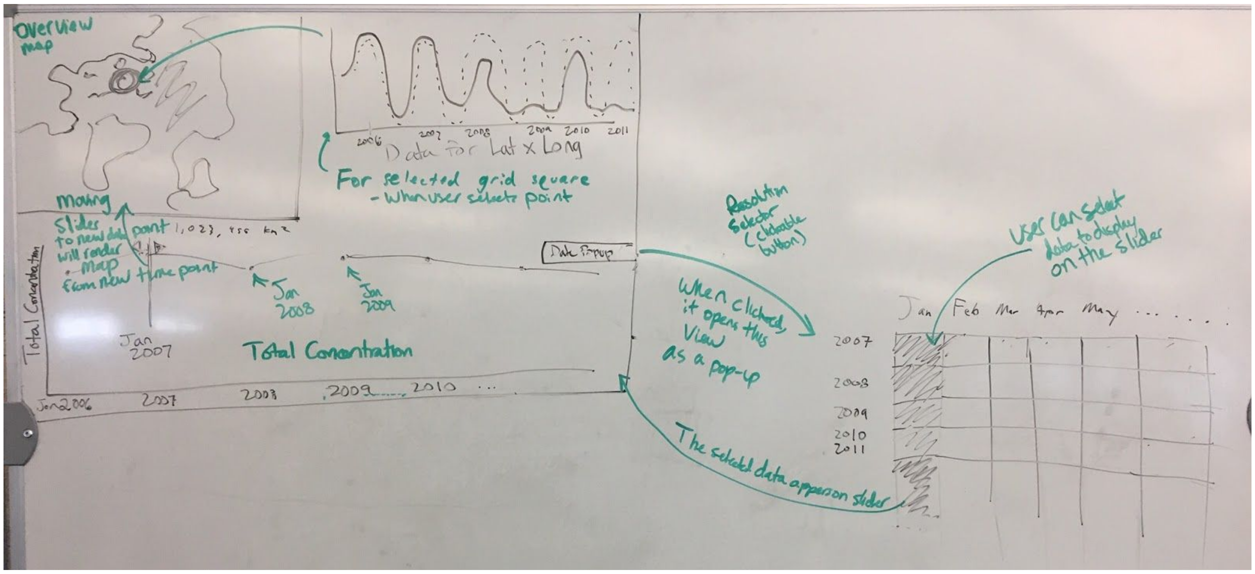
Additional Designs:

Design 2a:

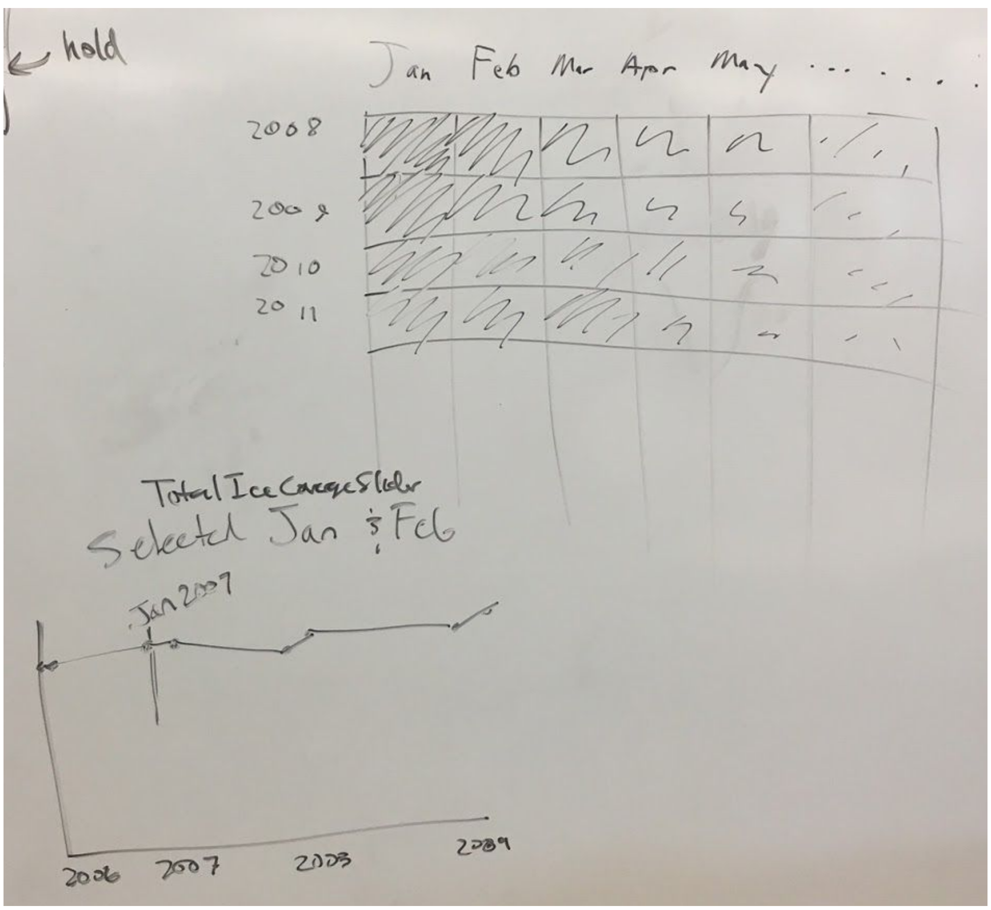


This design incorporates a slider that slides across a chart of total concentration. The user can move this slider back and fourth, and when they move it to a new month, it populates the overview map with the corresponding data.

Design 2b: Heatmap Incorporation

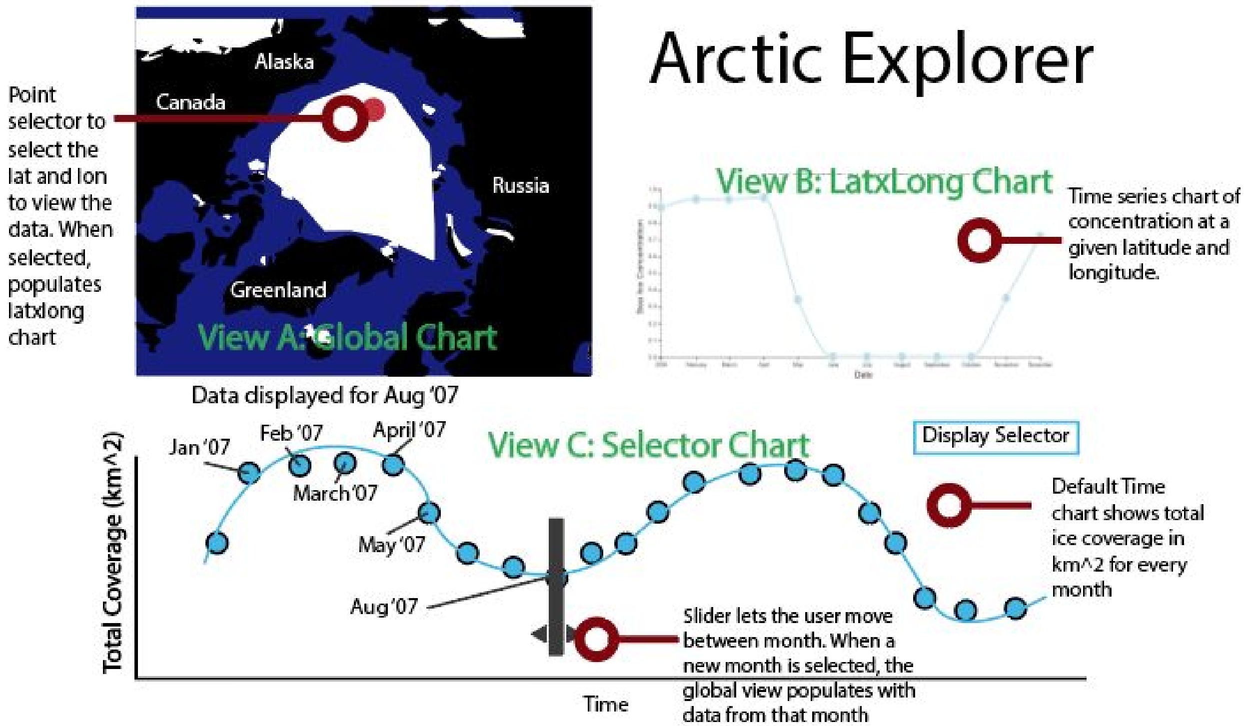


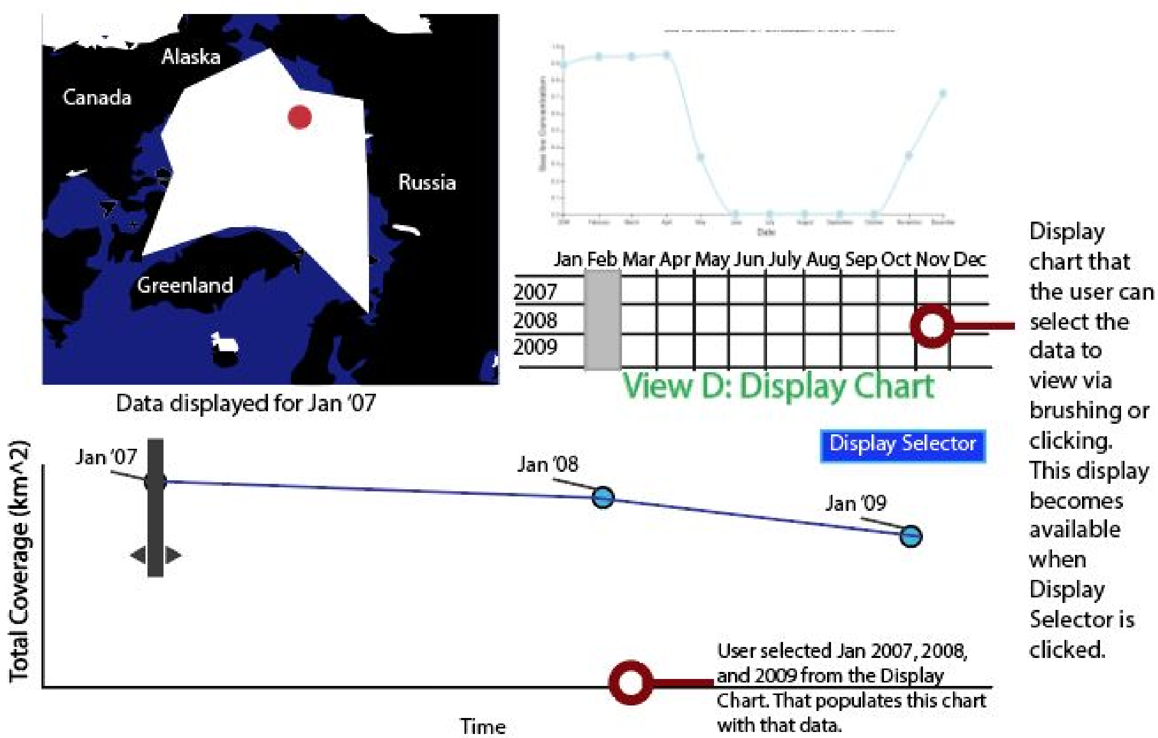
This view allows the user to manually select or to brush and select the data to display. The user can click on the 'heatmap chart' (displayed to the right) to select the months/years they want to view.



Display chart in more detail.

Design 2b: Digitized





This design is our current design that we are building. This design allows for the user to have as much control over the data as possible, and it allows for them to do custom queries which were not possible in past designs.
This design also leverages position to encode data about the total coverage. This will allow for our users to quickly understand patterns inside of the data. The query selector helps to bolster this ability as it allows the user to 'ignore' other data that may make it difficult to see relationships over time (such as the decline in total coverage as is displayed in the past design.

Implementation

The figure below shows the current state of the implementation:

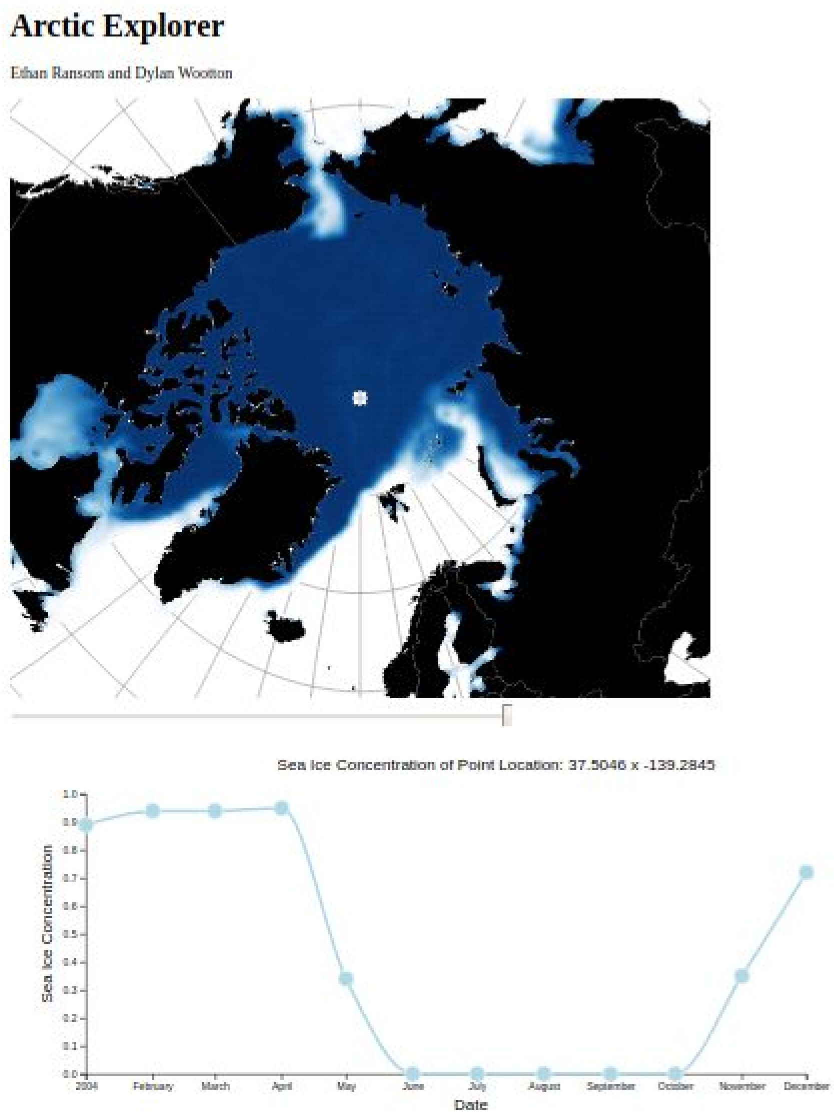


Figure: The state of our visualization at the milestone.

The top view is a map of a polar projection of the north pole. Counterintuitively, the deeper the blue, the higher the sea ice concentration. The dataset is the concentration at each month in the year of 2004. (12 datapoints.) The slider under the map can be used to select the month to be displayed on the map. The initial render of the landform outlines takes much less than a second, however the first render of the sea ice takes around 3 seconds. We are looking into optimizing the rendering, as well as techniques that could let us coalesce or downsample the data. Fortunately, consecutive renders take less than a second, we believe because d3 is reusing existing circle elements. The bottommost view is a line chart. It displays the concentration of ice of the grid square at the given latitude by month. In the final version of our visualization the user will be able to select which grid square is displayed here.

Please see Appendix A: Progress Journal at the end of this document if you are interested in a full description of the progress of the implementation up to this point.

Evaluation

We're waiting to hear back from the Golden lab for a time next week that works for them to do a design review. We hope to get their feedback on if the designs we are pursuing will be useful in answering the questions that they are trying to answer in their research.

Additionally, as the design becomes more fleshed out in weeks 4 and 5 we will be performing evaluation testing with the Golden lab to determine if our visualization is useful in their research.

Appendix A: Progress Journal

After a cursory investigation, we were unable to find any information on the ".nc" format that the raw NOAA data is provided in. As a workaround, we modified Delaney's MATLAB script (which can read the .nc files) to output three .csv files for a single timepoint, one for lat data, one for lon data, one for psi data. The row and column indicies could be used to tie the files together and get a {lat, lon, psi} tuple for each gridpoint. We built a simple script that could "zip these csvs together into one csv, with each row representing a gridsquare.

By taking these csv files, as well as the world.json file from Homework 4 and using the Gringorten Quincuncial projection we were able to render the first iteration of our map view:

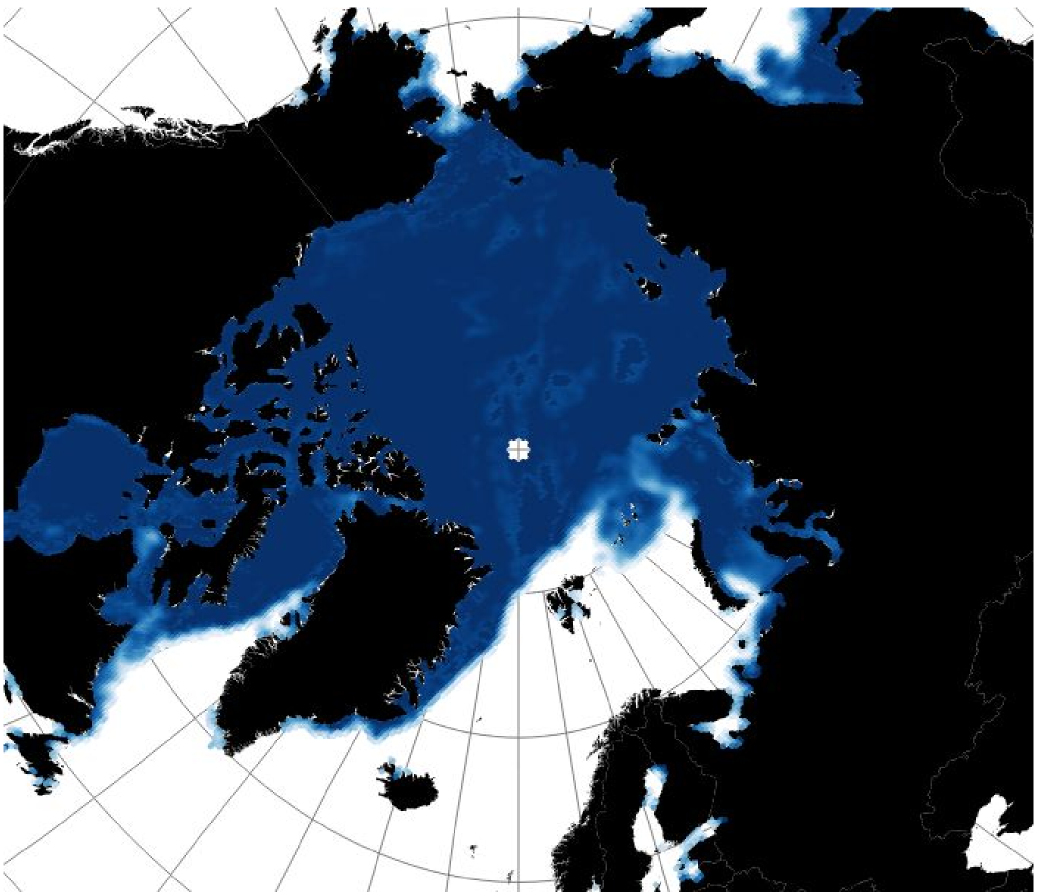


Figure: first iteration of the map view. Counterintuitively, blue is ice and white is ice-free ocean.

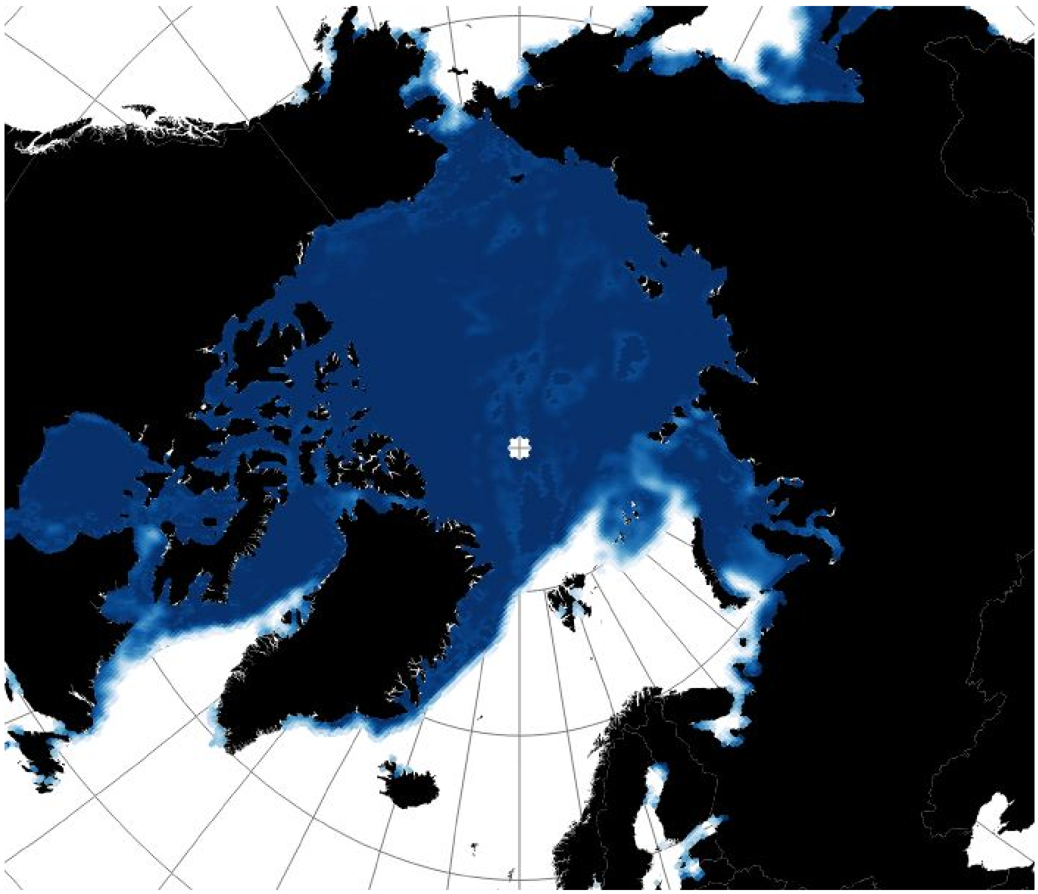
It was at this point that we began to run into issues with both the space and time runtime efficiency of our visualization. The map would take several seconds to render, and the ice grid squares would take upwards of 10 seconds. To solve the map issues we used the Mapshaper tool (https://mapshaper.org/) to clip the map features to north of 30 degrees latitude. This helped only marginally. We also used Mapshaper's simplify feature to downsample the detail of the landmasses. This lead to a major improvement in map rendering times. To fix the rendering of the gridsquares, we took another look at the data. The data contained “NaN” values for gridsquares who fell onto a landmass. Additionally, for many datapoints, especially those in summer months, a large number of gridsquares contained zeroes, which for rendering purposes could be ignored. This cut down the number of gridsquares to render from ~130,000 to only ~20,000 for our initial timepoint.

Next, we aimed to render multiple months’ worth of data at a time. Our first iteration of a new MATLAB script outputted an array of 2-dimensional matrices, which the rows and columns of the matrix corresponding to the rows and columns of the gridsquares. (Because the gridsquares are stationary, the latitude and longitude of the gridsquares can be stored in a separate file.) This format proved to be somewhat space-inefficient. Our estimates showed that a file storing 30 years could be as large as 1 Gigabyte. We’ve moved to what we believe to be our final format, which means storing gridsquares first, with each gridsquare storing an array containing its value for each timepoint. In this format, gridsquares which fall onto land, and thus whose only value is “NaN”, can be eliminated.

To complete our visualization up to the point of the milestone we implemented a slider to allow for year selection, and a line chart showing the sea ice coverage of a specific gridsquare. (In the final visualization the gridsquare will be chosen by the user.)

Project History:

1. We modified Delaney's matlab script to output three .csv files for a single timepoint, one for lat data, one for lon data, one for psi data. The row and column idicies could be used to tie the files together and get a {lat, lon, psi} tuple for each gridpoint
2. We build a simple script that could "zip these csvs together into one csv, with each row representing a gridsquare
3. We built a proof of concept d3 script that could render the world.json file, as well as the single timestamp:



[Pair programming Notes from Nov 4th]

4. We continued to modifiy the matlab script to output multple months. We changed up the format a little bit. Now it's a json array with an object for each month, and a "psijson" property that has an array of datapoints for each gridsquare. We can crossreference with a zipped lat.csv+lon.csv.
5. We discussed the space/time efficienty
6. We clipped the world.json to only that above 30 degrees N latitude, and simplified it using Mapshaper. This had a noticeable improvement on the render time.
7. We modified the d3 script to render a chosen month. (in progress)
8. To accomplish by Friday November 9th
    a. Map rendered with multiple views
        i. All JSON files created

b. Clipped World.json smoothed (to remove white)
c. Conversion to Lat Lon arrays
d. Cleaned up repo
e. Process book updated
f. Meeting with TA
g. Schedule meeting with Delaney and professor golding.
h. White Boarding Demo: