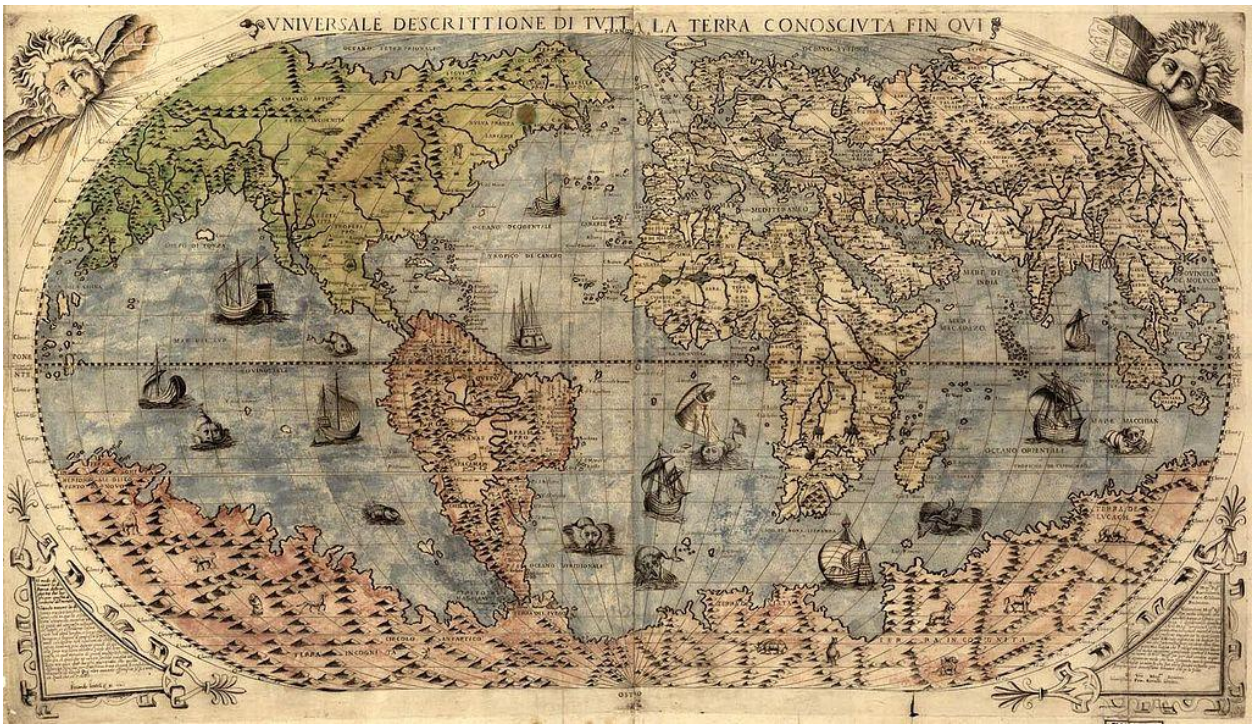
Influences of Renowned Artists

Process Book

Nick Lord-Ender-Laing, Nishita Kharche, & Eisen Montalvo

Insert a representative image or even style the page in a good way



We could kick around an aesthetic like this, similar to the 2012 example process book, to get across our emphasis on both history and world-wide influences

Table of Contents

Overview & Motivation
- Inciting Story and High-Level Motivations
- Concrete Technical Deliverables
- If We Have the Time

Related Work
- Initial Inspirations
- Inspirations While Revising

Questions
- Core Questions That We Are Trying to Answer
- Development of Existing and Creation of New Questions

Data
- From Book to Table
- Punching Up the Data
- Outside of the Tables

Exploratory Data Analysis
- The Map and the Old Timeline
- The Birth of the Network View
- The Ribbons Meet the Artists
- Briefly Exploring the Adjacency Matrix

Design Evolution
- Early Stages
- Post-Proposal: Reworking What's There
- Post-Proposal: Brand New Parts

Implementation (images should reflect final versions when talking about current state, so some placeholders for now)
- Network and Map Implementation
- Chord Visualization Implementation
- Adjacency Matrix Implementation

Evaluation

(Add image if appropriate)

Overview & Motivation

Inciting Story and High-Level Motivations

When brainstorming an idea from scratch for this project, we contemplated three very different ones. The first two were fine, but we spent only a few minutes speculating on fun and interesting things we could do with each. The third idea, by comparison, totally and effortlessly blindsided us. We spent easily over ten minutes just unloading exciting idea after exciting idea. By the end of the session, it went without saying that this third idea was the one we were going with, and that is the idea that we report on now.

This concept being: a data visualization that conveys relationships between each person in a set of historic artists. More specifically, we want to convey *influential* relationships. In the realm of art, it is not uncommon for an artist to have inspired the work of other artists. What makes this link so interesting to us is that they can extend from the influencer to another person hundreds of years later and on an entirely different continent (see fig. _).

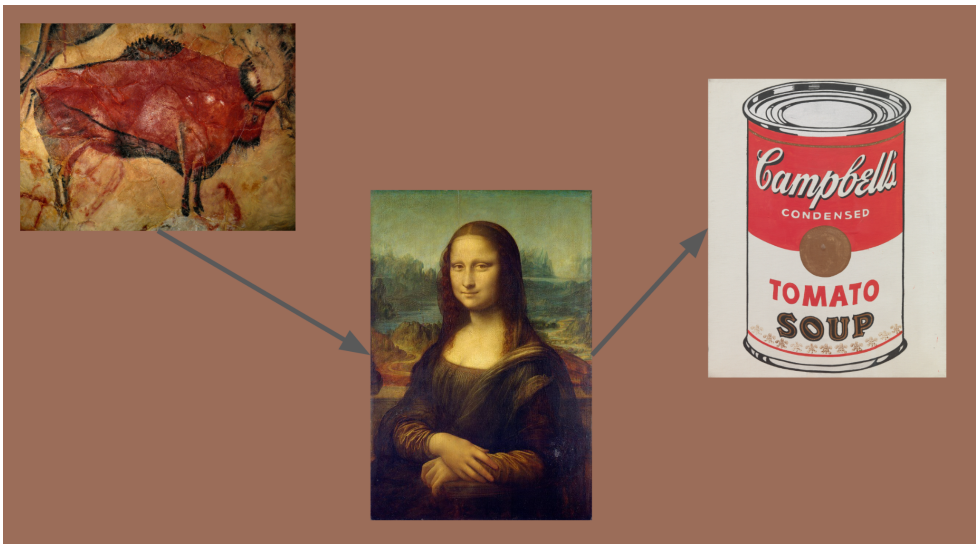


Figure _. *An extreme influential link between the work of a cave painter and da Vinci, and da Vinci and Andy Warhol (this is not an actual link; it is sheerly for the sake of a high-level example)*

We not only want to share these fascinating invisible connections, but we could not help but think that they are begging to be made visible. Further, no one has taken it upon themselves to make this type of

rich information accessible and centralized. This is plainly evident by one of our more concrete inspirations: art books. A member of the team had a particular interest in books that touch on these inspirational throughlines, and the best way to centralize and compare those connections was to flip through the pages and hope you remembered the last link you read about. A digestible data visualization that extracts the inspirations from those books seemed like a great remedy for both these wants of ours to convey this interesting type of data, and the gaps that we could not help but notice after conceiving the idea.

In a way, our overarching motivation is education. While one can say that about any visualization, we want to fully embrace a more traditional educational tone. The subject of historic artists lends itself naturally to that, and we want a visualization that would not look out-of-place as a museum display. The level of polish and presentation that we would need in order to match that is also a big goal for us. We really want to make it interactive too. We do not want something that is comparable to an unimpressive static map that you would glance at and then walk away from. We want to effortlessly convey to a user that there are more interesting things about our data than the default view that they can find out about by digging around. Our want for this virtual digging is important, and thus it informed many, if not all of our design decisions.

Concrete Technical Deliverables

With our broader motivations laying the abstract groundwork for our project, we next decided on some solid more technical goals that we want to accomplish by the end. These goals changed over time (see *Design Evolution* for further details), and so these represent the final version of our deliverables. Without these four views implemented and with interaction between them, we would consider our project a failure. The first view that we knew we must have is a world map. We repeatedly asked ourselves “do we really need a map?” and while its exact purpose changed, its presence as a goal never went away. Plotting these artists on that map is crucially important because it takes our data and

grounds it in the real world. Another way of grounding this data set visually is to include a view that shows a picture of each artist. We really want to convey that these are people who had their own lives (not just numbers), and a portrait is a simple and effective way of doing that. More specifically, we want a view with a standard network visualization showing images of the artists on each node, and edges showing their influences. We would then filter the map to *only* plot those artists selected from the network view. This way, we can have *ideal* spacing with all of our data there in our network, and *accurate* spacing without worrying about clutter on the map. Avoiding clutter by dividing this into two views goes a long way in presenting something that looks good, to put it plainly.

On the note of a pleasing presentation, we would consider our project a failure if we do not also create another more artistic view. Our data set is related to master artists after all, and so we want a colorful, and visually striking creative view that does this type of data justice. We thought of a D3 chord visualization (see later sections for specifics) to show all of our data in a digestible way that also meets the aforementioned visual criteria (see fig. _).

Figure _. *An example of how similar a D3 chord chart is to a painter's palette. This shows how naturally this chart lends itself to this kind of data in one way.*

On the other hand, a more blunt view is an equally important goal for us. We want to cater to those who like a less glossy type of visualization, and given our relationship-driven data, an adjacency matrix was the perfect choice.

If We Have the Time

Lastly, we can briefly touch on our optional goals. These are things that would be nice to have, but we would not consider the project a failure if they are not present in the final product. We want our visualization web page to be perfectly legible on a browser regardless of the window size. While there can be some awkward ratios for any given window, not to mention vertically-oriented displays like phones, we want to accommodate for as many possibilities as we can. At the same time, we know that most of our users will be viewing this on a computer with at least HD resolution and horizontal orientation. So while we developed with this visual scalability in mind, we would not consider our project a failure if we do not optimize for the extreme minority of users that use vertically-oriented computer monitors, or other unusual means of viewing like that. Another stretch goal is incorporating some text-based information on the artists. If we have the time, we would like to somehow incorporate links to the Wikipedia articles for each of our artists, and a user could go to the page if they want to learn more.

Related Work

Initial Inspirations

During our initial planning phase, our foremost inspiration for this project as a whole is our general interest in the history of artists and how they relate to each other via inspirations. We discussed how this would lend itself naturally to a type of network visualization (in relation to our latter interest) that could rely on a world map in order to ground it. More specifically, the books, *Art: The Definitive Visual Guide* and *Artists: Their Lives And Works* were our main source of inspiration. The stories of the people detailed in those books were the most fascinating parts, and we want to convey aspects of those stories in a visual way. Our focus on the stories of these people led us naturally to a technological source of inspiration, the screen-based displays you often find in museums (see fig. _).

Figure _. *(Figure numbers likely to change as we make edits) - https://prodisplay.com/market-sectors/public/museums-galleries/*

We want our data visualization to convey a fairly neutral tone that encourages exploration and learning. We want to make something similar to this, but achievable on a web browser.

Inspirations While Revising

After we discussed our proposal with our assigned TA, we had a few takeaways concerning the direction of our project. One of them was that we should be visualizing more interesting data using more traditional methods. We then brainstormed some interesting traits about the artists in our data (namely, art medium, and means of death), and thought about representing those using a parallel coordinate visualization (see fig. _).

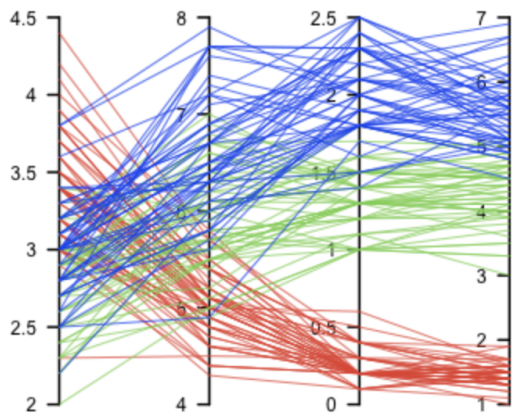


Figure _. (wikipedia article on parallel coords)

However, we decided that this kind of visualization would not be very interesting because it is unlikely that art medium has any correlation with means of death (for example). We then remembered a somewhat similar visualization technique that would be both more aesthetically pleasing and appropriate for our type of data: chord visualization.

In particular, we really like the famous *The Lord of the Rings* chord visualization (see fig. _). We were inspired to use this visualization's look and functionality to convey those tangential traits of our artists. In combination with dropdowns that can change both the grouping of our artists and the data attribute they map correspond to (medium, means of death, etc.), we can have a dynamic version of this visualization (for more details, see *Design Evolution* for further details).

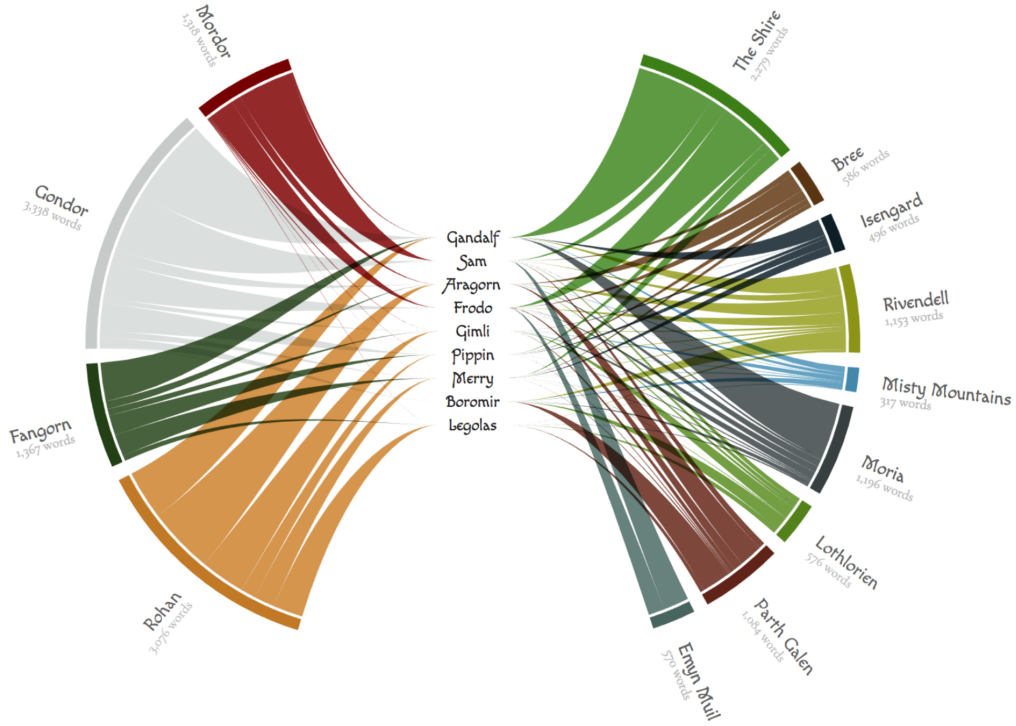
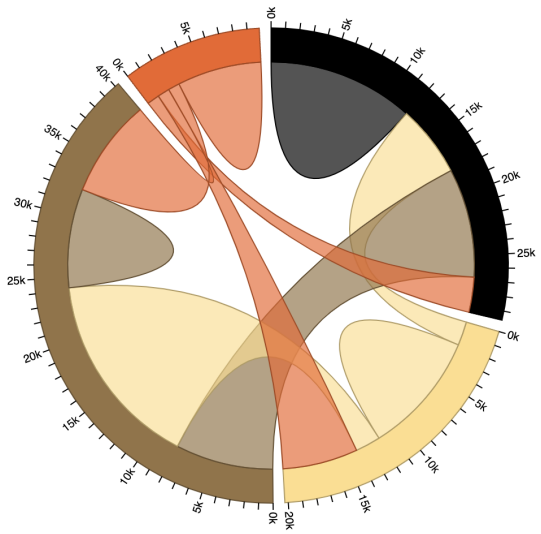


Figure _. (https://www.datasketch.es/project/lotr)

In addition to this example, we also took large inspiration from the in-class discussion on chord visualizations, and the accompanying code (see fig. _). However, we took even more direct inspiration from the freely available chord and ribbon code on the official Github repository for D3. We followed the footsteps of the creator of figure _, and decided to take the general methodology in the D3 code, and tweak it to look closer to what we want (see *Implementation* for further details).




```
<!DOCTYPE html>
<meta charset="utf-8">
<style>
    body {
        font: 10px sans-serif;
    }

    .group-tick line {
        stroke: #000;
    }

    .ribbons {
        fill-opacity: 0.67;
    }
</style>
<svg width="500" height="500"></svg>
<script src="https://d3js.org/d3.v7.min.js"></script>
<script>
    {
        // Our input data
        let matrix = [
            [11975, 5871, 8916, 2868],
            [1951, 10048, 2060, 6171],
            [8010, 16145, 8090, 8045],
            [1013, 990, 940, 6907]
        ];

        let svg = d3.select("svg"),
            width = svg.attr("width"),
            height = svg.attr("height"),
            outerRadius = Math.min(width, height) * 0.5 - 40,
            innerRadius = outerRadius - 30;
```

Figure _

Another part of the course that inspired us was the lecture on world map projections. Originally, we used an equirectangular projection to get some code off the ground quickly, but after learning about other projections from that lecture, we changed it to the one that we learned is the most well-rounded: Winkel tripel (*Citation to lecture slides perhaps*). You can see this difference in our early code in figure _.

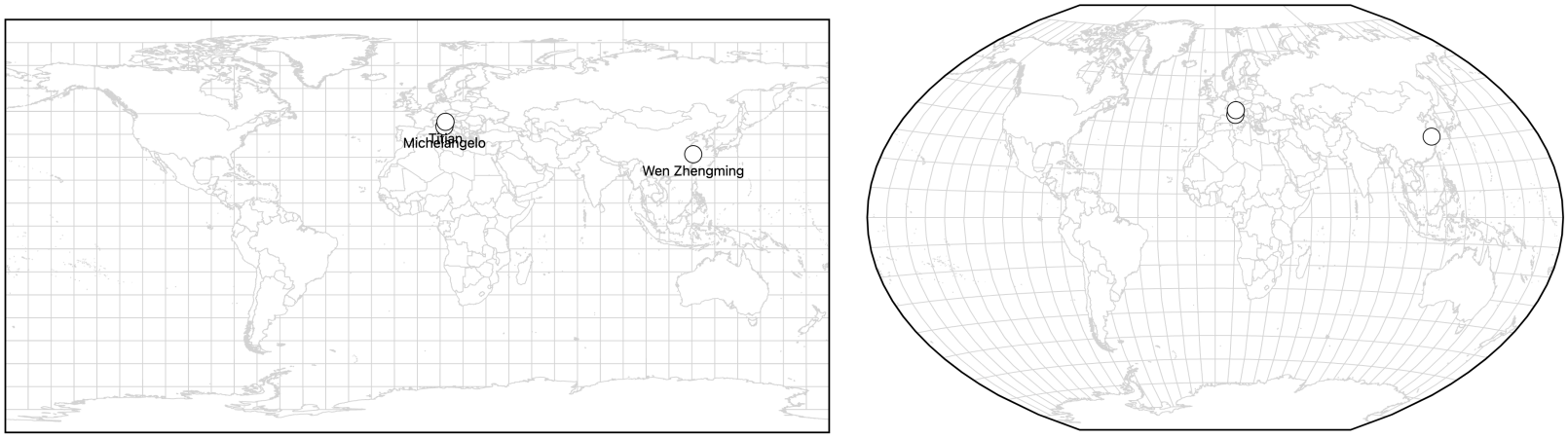


Figure _. …

A short while after that lecture, we took a lot of notes on week 12's lecture, *Visualizing Networks and Trees*, and week 13's lecture, *Visualizing Multivariate Networks*. These two, and especially the latter one, was just what we needed to think of better network visualizations that we could add to our work. The type of data in the class examples seemed very similar to our influence data, and we arrived on an adjacency matrix as a good simple view to add. In particular, we took inspiration from Micha Stubb's observable implementation of one in D3 (see fig. _).

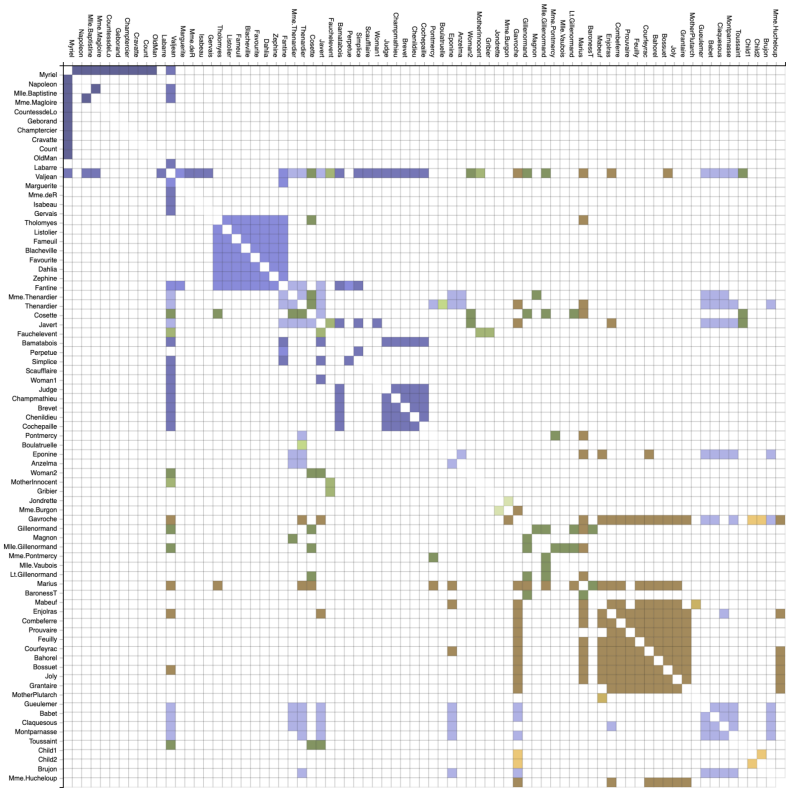


Figure _. Emphasize visual trends

We also took direct inspiration from our hands-on experiences with the homework assignments for this class. In particular, homework 6's usage of a D3 physics simulation gave us the idea to put one in our project's code. In particular, we used one for the rendering of our artist nodes in our traditional node and edge network view (see fig. _).

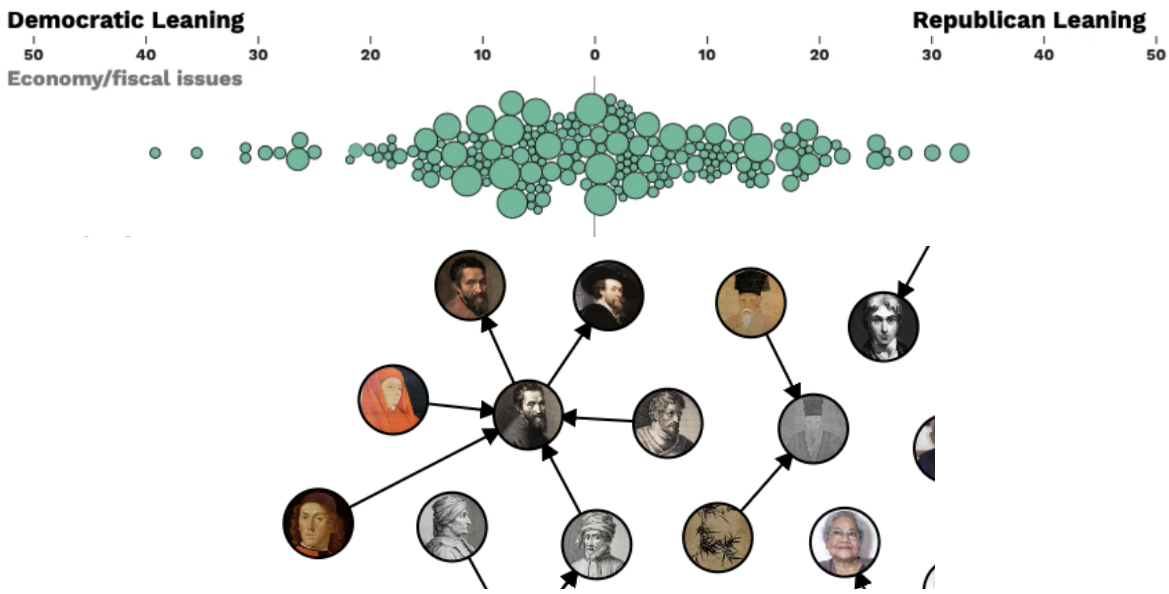


Figure _.

Questions

Core Questions That We Are Trying to Answer

The most important questions that we want to answer visually are those in relation to the nature of inspirations between artists across all of history. A good way to contextualize this is to look at an example artist like Diego Velázquez. The first couple of questions we have about Velázquez are,

> ***"Which notable artists were major inspirations for Velázquez?" and "which notable artists did Velázquez inspire similarly?"***

Within these major questions, we then have more detailed questions concerning the nature of the inspiration both ways. For example,

> ***"Did Velázquez inspire people while he was alive, or did his work become more inspirational long after his death? If so, how long after?"***

are examples of the nuance of those deeper questions we are aiming to answer.

Meanwhile, there are more cut and dry questions that we want to answer. Still using Velázquez as an example, we would like to know,

> ***"When did Velázquez die? When was he born? Where was he born? When did he die? Where did he die? Who were his contemporaries?"***

While these are all simple facts that one can find easily on Wikipedia, one of our goals is to centralize the blunt answers to these questions visually. All together, the answers to these questions build a more comprehensive picture of the people in our data.

Development of Existing and Creation of New Questions

Of our initial questions, the ones that changed the most clearly are those related to influences between artists. Initially, we simply wanted to answer the question,

"Who influenced whom?"

That question became more layered, and after we finished our project proposal, we wanted to answer the more nuanced questions that fall under that main question more overtly. For example,

"What is the range of this particular influence relationship temporally?"

is a question we now want to clearly answer visually rather than simply providing the dates through text and letting the audience figure it out.

Lastly, we developed one of our components to answer broad new questions that apply to groups of data and attributes of focus. So with any user-selected attribute and grouping, we could answer any question that fits the mold of,

"How many artists per selected grouping have a certain attribute?"

For example, if a user wants to group our artists by century, and selects artistic medium as the attribute of focus, we would be able to answer:

"How many artists in a given century used a particular medium?"

This easily extends to grouping by continent, selecting means of death as the attribute, etc. Answering these questions will both establish context, and convey interesting trends (see *Implementation* & *Design Evolution* for more details). Additionally, the different combinations of groupings and attributes thus leads to a large variety of questions answered. These revisions are key, and we believe that they will make our audience more keen to explore, which is one of our main intentions.

Data

From Book to Table

While our code that directly renders our visualization took a while to get right, a process that took a comparable amount of time was collecting our data. While there is plenty of data on historically significant artists, we found no data set that contains influences amongst them. So there was not a data set out there that we could use directly or even derive data from. We created our own data set by curating information from many sources *manually*. Our invaluable sources throughout this process were primarily books, and then we used Wikipedia to fill in any missing simplistic information like locations of birth and death. The books we referred to are: *Art: The Definitive Visual Guide* and *Artists: Their Lives And Works*, (see fig. _ for covers) and the most important data we derived from there is the aforementioned influential links.

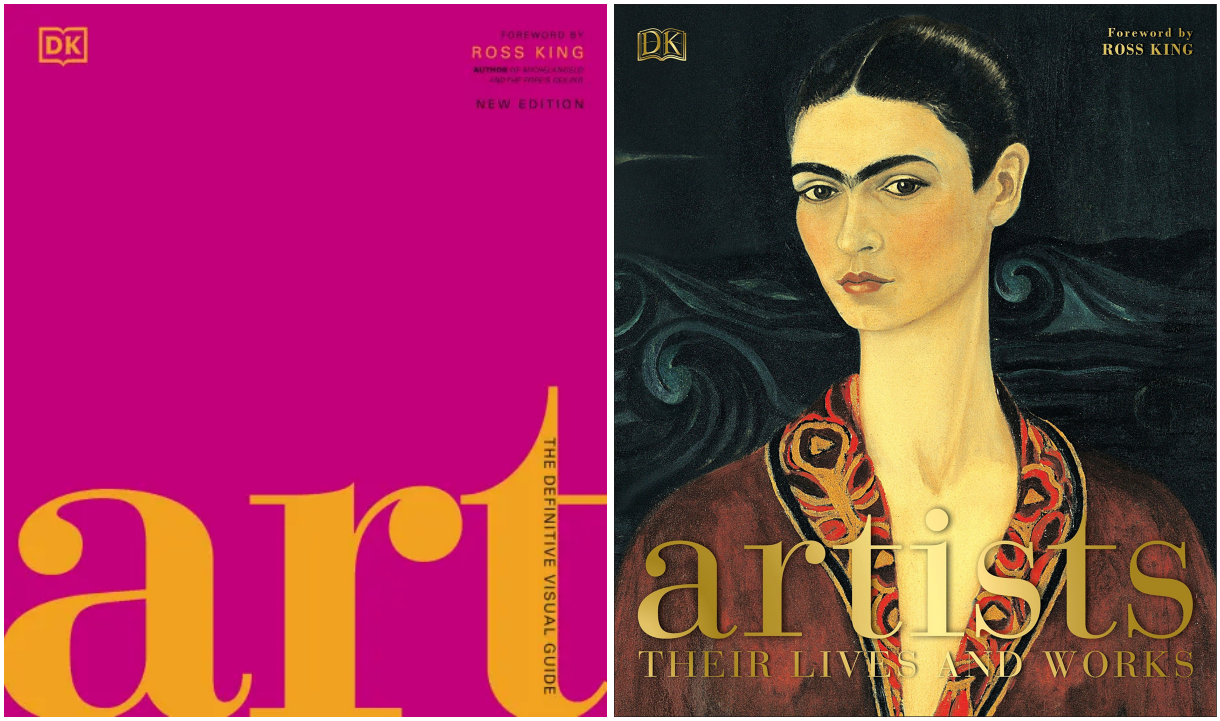
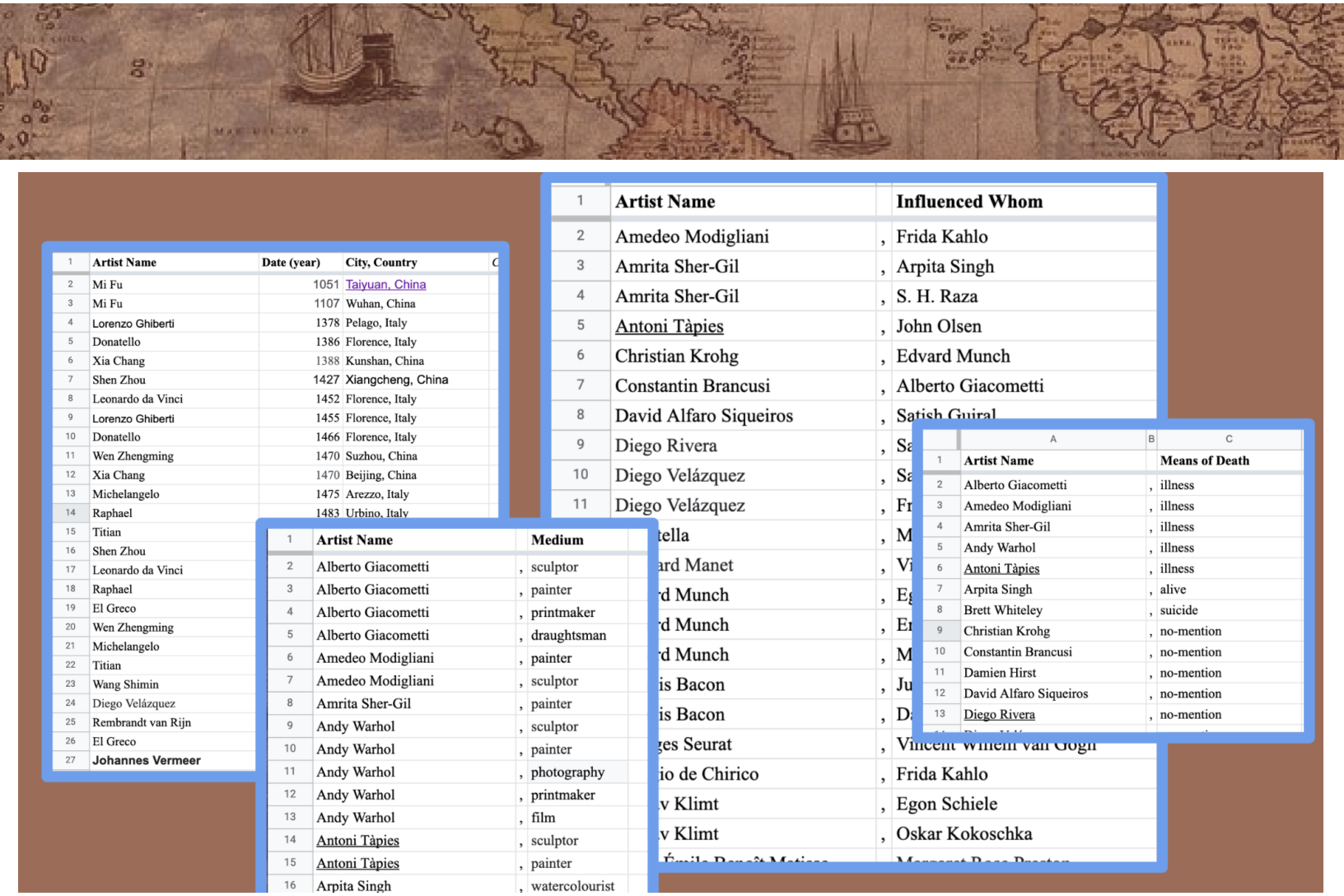


Figure _.

Our data set is fairly small, around 100 artists, but the information on each artist is detailed, and spread across multiple tables. We created those tables on Google Sheets (see fig. _).



1	Artist Name	Date (year)	City, Country
2	Mi Fu	1051	Taiyuan, China
3	Mi Fu	1107	Wuhan, China
4	Lorenzo Ghiberti	1378	Pelago, Italy
5	Donatello	1386	Florence, Italy
6	Xia Chang	1388	Kunshan, China
7	Shen Zhou	1427	Xiangcheng, China
8	Leonardo da Vinci	1452	Florence, Italy
9	Lorenzo Ghiberti	1455	Florence, Italy
10	Donatello	1466	Florence, Italy
11	Wen Zhengming	1470	Suzhou, China
12	Xia Chang	1470	Beijing, China
13	Michelangelo	1475	Arezzo, Italy
14	Raphael	1483	Urbino, Italy
15	Titian		
16	Shen Zhou		
17	Leonardo da Vinci		
18	Raphael		
19	El Greco		
20	Wen Zhengming		
21	Michelangelo		
22	Titian		
23	Wang Shimin		
24	Diego Velázquez		
25	Rembrandt van Rijn		
26	El Greco		
27	Johannes Vermeer		

1	Artist Name	Medium
2	Alberto Giacometti	sculptor
3	Alberto Giacometti	painter
4	Alberto Giacometti	printmaker
5	Alberto Giacometti	draughtsman
6	Amedeo Modigliani	painter
7	Amedeo Modigliani	sculptor
8	Amrita Sher-Gil	painter
9	Andy Warhol	sculptor
10	Andy Warhol	painter
11	Andy Warhol	photography
12	Andy Warhol	printmaker
13	Andy Warhol	film
14	Antoni Tàpies	sculptor
15	Antoni Tàpies	painter
16	Arpita Singh	watercolourist

1	Artist Name	Influenced Whom
2	Amedeo Modigliani	Frida Kahlo
3	Amrita Sher-Gil	Arpita Singh
4	Amrita Sher-Gil	S. H. Raza
5	Antoni Tàpies	John Olsen
6	Christian Krohg	Edvard Munch
7	Constantin Brancusi	Alberto Giacometti
8	David Alfaro Siqueiros	Satish Guiral
9	Diego Rivera	Sa
10	Diego Velázquez	Sa
11	Diego Velázquez	Fr
	tella	M
	ard Manet	Vi
	rd Munch	Eg
	rd Munch	Er
	rd Munch	M
	is Bacon	Ju
	is Bacon	Da
	ges Seurat	Vincent Willem van Gogh
	io de Chirico	Frida Kahlo
	v Klimt	Egon Schiele
	v Klimt	Oskar Kokoschka

	A	B	C
1	Artist Name		Means of Death
2	Alberto Giacometti		illness
3	Amedeo Modigliani		illness
4	Amrita Sher-Gil		illness
5	Andy Warhol		illness
6	Antoni Tàpies		illness
7	Arpita Singh		alive
8	Brett Whiteley		suicide
9	Christian Krohg		no-mention
10	Constantin Brancusi		no-mention
11	Damien Hirst		no-mention
12	David Alfaro Siqueiros		no-mention
13	Diego Rivera		no-mention

Figure _.

We manually filled a table that contains the artists' mediums of art, a table that has their means of death, another with the influence-based connections, and one with locations and dates associated with their births and deaths. Then, the key for accessing an artist's data across different tables is simply their name.

This method had the great advantage of keeping our data just as we wanted it to be. It would be pointless in a visualization like ours to include artists with no influence link in the set, and thus we kept a standard that all of our artists would have to have at least one link. This also allowed us to get very familiar with the data, as we would even personally discuss particular characters among the set.

Punching Up the Data

This method was difficult in some areas, and a big example involves a library we use for converting a city and country to latitude and longitude. If an artist's location did not fit that format, we

would have issues. For example, Marcel Duchamp's birthplace is Blainville-Crevon, France. This is a commune, and so we checked Google Maps to see which city it roughly maps to. If the closest city was not clear on Google Maps, we then had to reference a more detailed map of France (see fig. _).

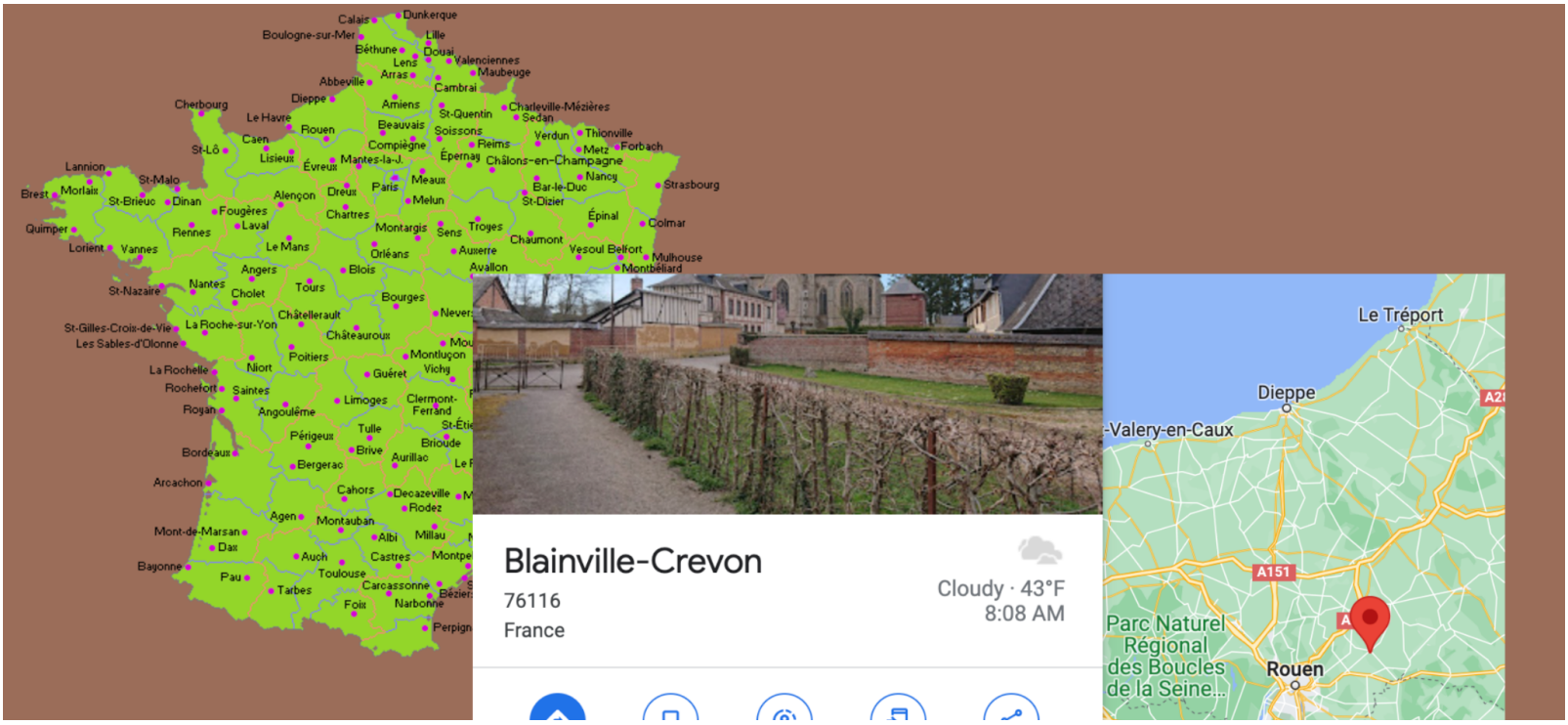


Figure _. …

While examples like these did not pop up often, they appeared enough to be noteworthy, and we had to use similar processes if something like that happened in a different country. We also cleaned up our data in some minor ways to make it more code-friendly. An example was getting rid of any and all accent marks on our artists' names.

Also, there were a small handful of times when we wanted to augment our data to have at least one artist from each continent or at least one from each century. In these few times, we would simply take to various articles online on those very topics.

Outside of the Tables

After gathering all of our data into tables, there were some elements and processes that did not fit cleanly into our Google Sheets workflow. Firstly, it is worth noting that we considered adopting a database approach for this project, but because of the size of our data, we decided to use local .json

files. This also removes the perhaps needless task of maintaining a database properly. We used scripts to convert our .csv files into JSON (see fig. _), and then stored them in a simple folder. On the subject of data locality, a chunk of our data that we also keep local is the portrait images for each artist in our set. Rather than scraping images from wikipedia of various dimensions, we manually gathered portrait images for all roughly 100 of our artists. We then used a clever Adobe Photoshop workflow to convert them into circular thumbnail images (see fig. _). These images are all of the same size, and that size is small enough so that they do not take up too much space. Lastly, we augmented our JSON-ified table containing means of death for each artist to include another attribute. That being the local file path to that artist's thumbnail image.

Overall, the data collection and adjustment processes were extremely hands-on, but we find this appropriate given the nature of our data.

Exploratory Data Analysis

The Map and the Old Timeline

The first way in which we visually explored our data was plotting the appropriate part of it onto an early version of our map component. After seeing the data this way (see fig. _), a particularly important improvement became obvious.

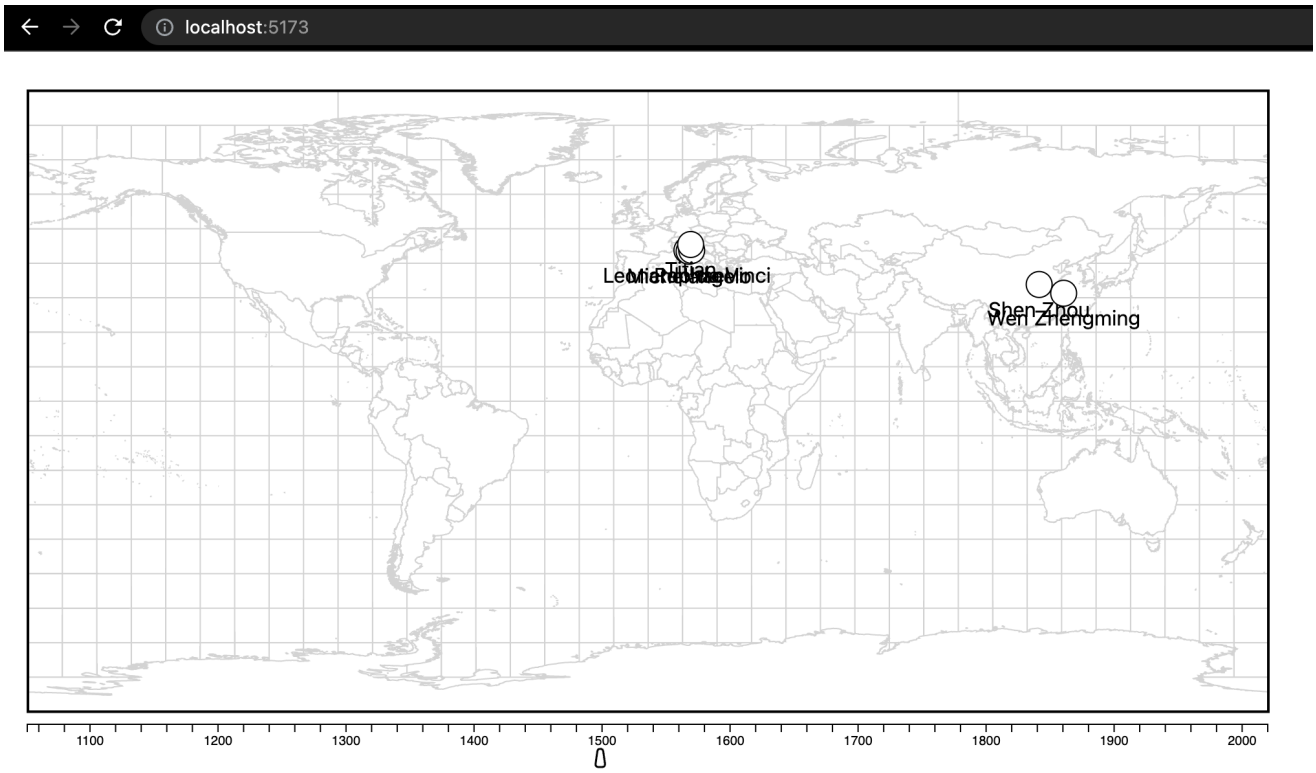


Figure _ …

While we are no longer filtering the map view based on a selected date on a timeline, the general problem of clustering nodes led us to looking into D3 force physics simulations to space these markers well. The fact that the artists' nodes cluster in particular regions around particular timelines also led us to the greater decision of abandoning the timeline component as a whole, and filtering the map view in a different way (see *Implementation*).

Something else that became apparent through our exploration in relation to figure _ is that there are geographic gaps in our data. We had little to no artists from Africa, South America, Canada, and parts of east Asia. To make the most of our screen real estate we both augmented the data to fill in

these gaps, and made the map only one of four views instead of our main component (that idea being a remnant from some of our earliest plans).

Speaking of gaps, we also have some temporal gaps in our data. This problem exists mainly between the oldest artist in our set, who lived around the 11th century, and the late 14th century. As a minimum remedy, we wanted to have at least one artist per century. At the same time, the realization that it would take a lot more data to pad out every slight temporal gap in our data led us even further away from our originally planned timeline component.

The Birth of the Network View

Something else that our map-related analysis led us to is our current network view. We looked at these apparent clusters in our data, and we could not really get rid of them due to how interesting the links are. We thought, *if only we could space our nodes optimally without worrying about location-related accuracy*. To attain just that, we decided after this analysis to create a more traditional node network view (see fig. _).

Figure _.

We developed a way of using this new view to filter the map view in a way that prevents geographic or temporal uncomfortable clustering (see later sections for more details). Additionally, something that figure _'s presentation showed us is that there are some obvious islands in our network. These island groups could take priority over other nodes that are not as significant. For example, if we need to get rid

of some nodes for whatever reason, or replace some with others for the sake of geographic or temporal diversity, we would acknowledge that these islands are probably not good candidates for deletion.

The Ribbons Meet the Artists

Our chord and chart underwent some experiments with fake data while getting it off the ground in its very early stages. The crux of this component are the groupings and subgroupings of all of the data (the arcs), and the mapping of those subgroups onto their attributes (the ribbons show this). When working with just the arcs, we replicated a realistic number of groups, 10, and experimented with an odd number of them as well (see fig _).

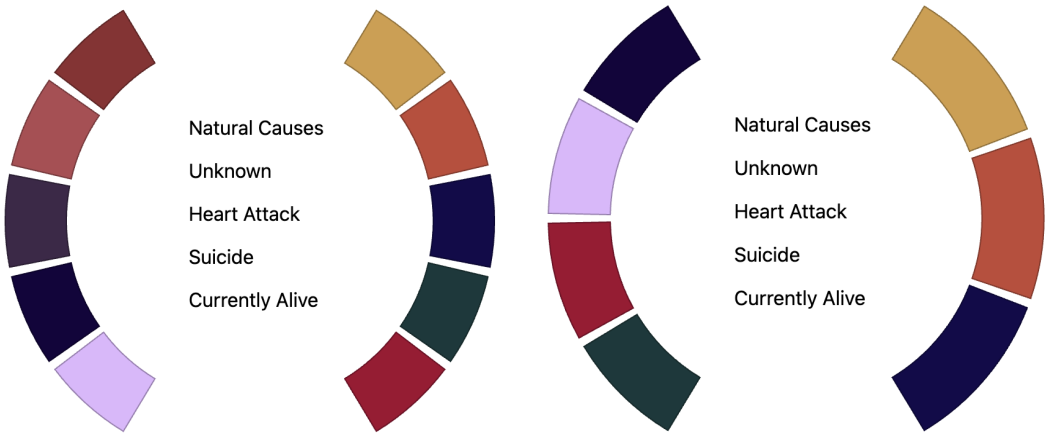


Figure _. *Two examples of our early chord visualization on n=10 (left) and n=7 (right) data groupings.*

Through this, we saw that we likely would not have to worry about running out of space for this unorthodox component, but wiring real data to it made us make a decision for the better.

When wiring real data initially, we realized that a uniform sizing for each arc would be a potential missed opportunity in regards to encoding. We decided that we should encode the number of artists in a grouping by the size of the arc, even if this meant complicating the program. After accounting for both this and wiring real data, one can see how it looked in figure _ when combined with our ribbon generating method.

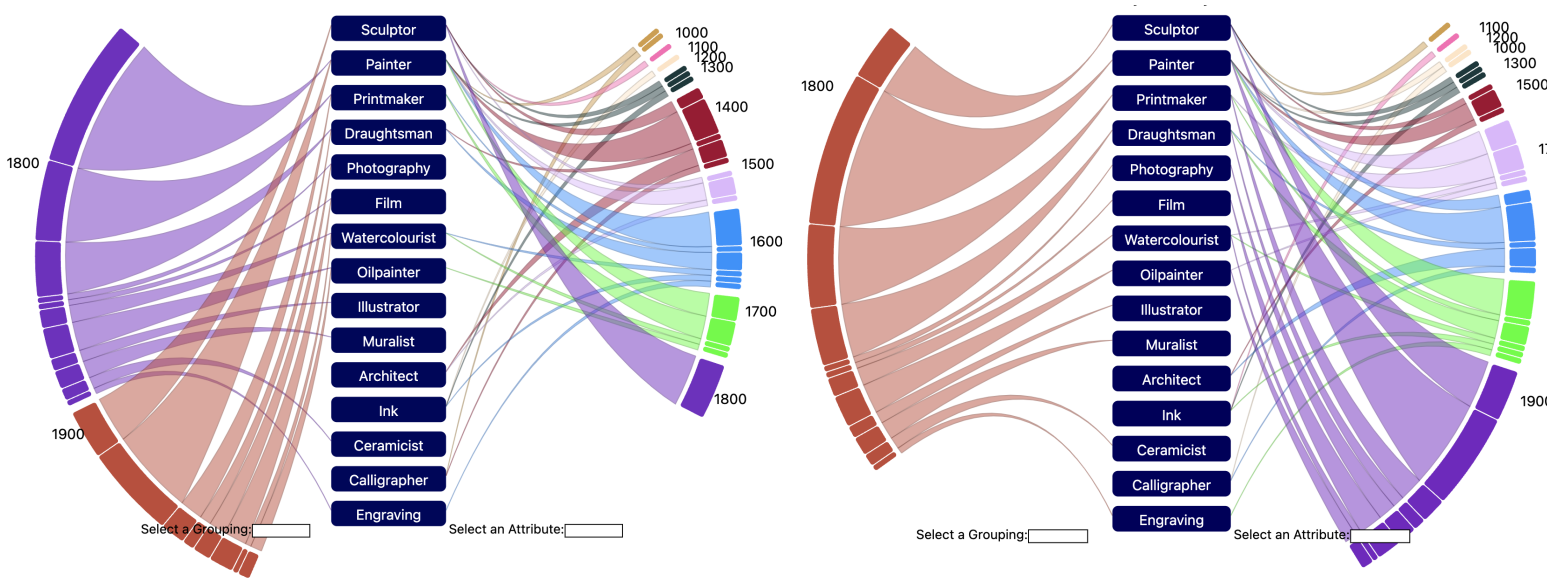


Figure _. Sorted vs unsorted

After analyzing figure _'s left image, we realized that to make the chart look better, we needed to sort the ribbons by the size of their groups. This led us to figure _'s right image. When tweaking the code for that version, we did not want to separate a group's ribbons and chords from each other. However, we decided in the end that maintaining symmetry was more important than keeping our groups close together at all costs. This is because we use both color and text as an encoding for that grouping anyway, and this happy medium is still sensical (see fig. _).

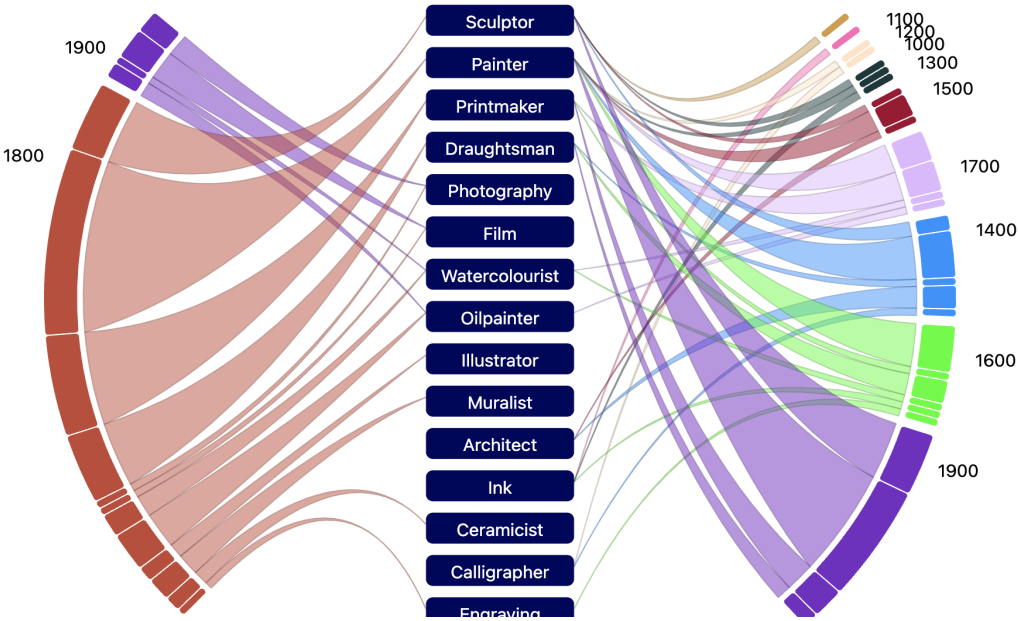


Figure _. …

Briefly Exploring the Adjacency Matrix

Compared to our other final components, our adjacency matrix had the least amount of growth from exploratory data analysis. That is because compared to all of the other components, the adjacency matrix is by far the least complicated, and that is by design. In our initial renderings of the data this way (see fig _), one main thing became obvious: we needed sorting.

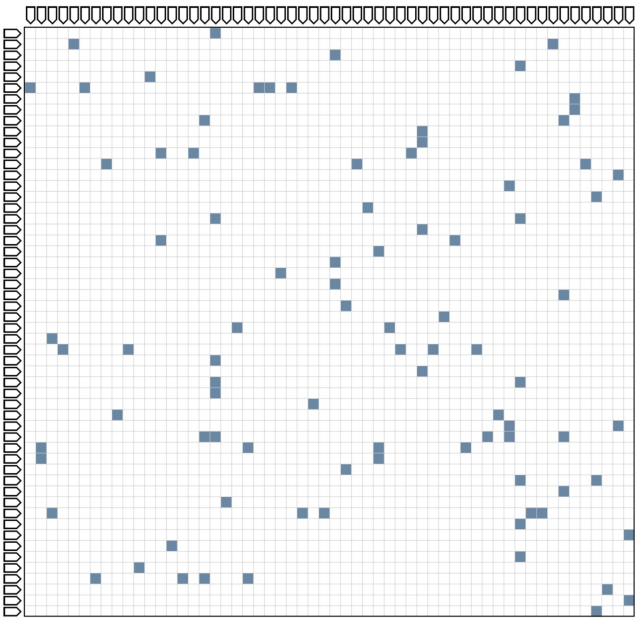


Figure _.

As we learned in class, sorting is a great addition for most adjacency matrices because it makes otherwise unseen trends become obvious. More specifically, we wanted to implement sorting based on the number of influences. That is what we show in this matrix, and thus sorting on any other data attribute would probably not be very helpful. If we sort by number of influences on the other hand, regions of the matrix become more interesting because that is what we are encoding by color.

Design Evolution

Early Stages

Our elements that remained constant include a world map, timeline, arrows showing influences, and text-based artist identifiers, but the rest of the design has certainly changed. The first major development occurred before we wrote much code at all. We synthesized our first three sketches into a single agreed-upon design. Our first prototype emphasized the timeline a bit too much, and lacked filtering to avoid influence arrow clutter (see fig. _). The second sketch carved out some good ideas for avoiding that clutter, but it did not utilize geospatial encoding well (see fig. _). Then our third design emphasized a niche part of our data that we may not even have in the end (that being 'lost' artworks) (see fig. _)

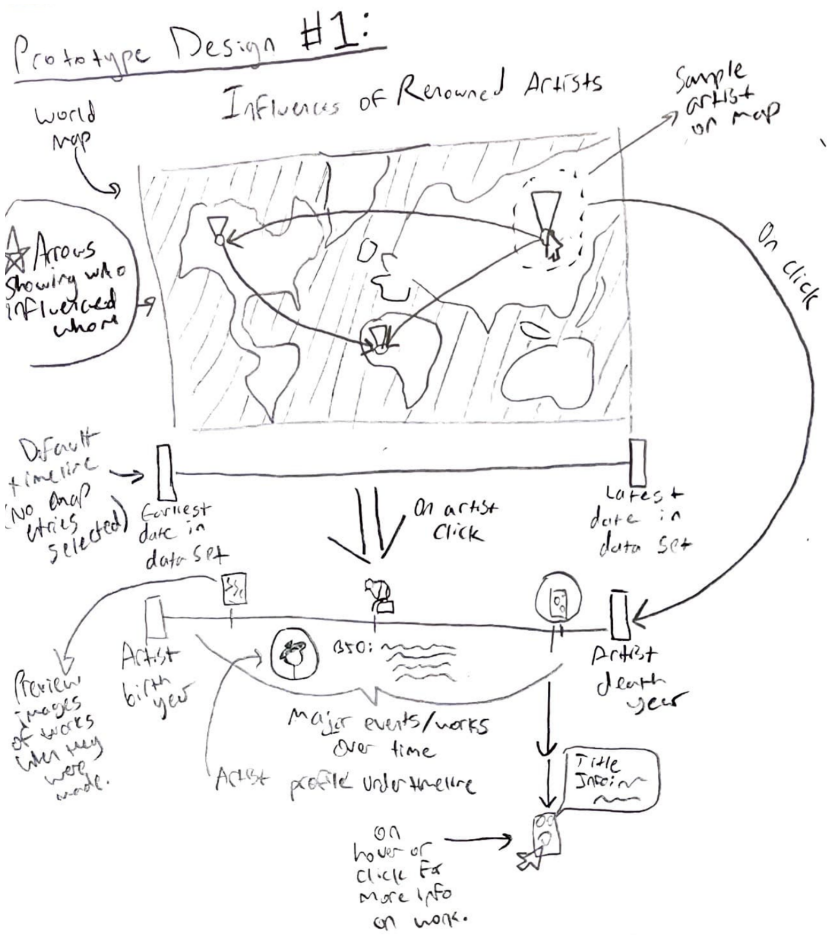


Figure _. *Mockup design sketch number one.*

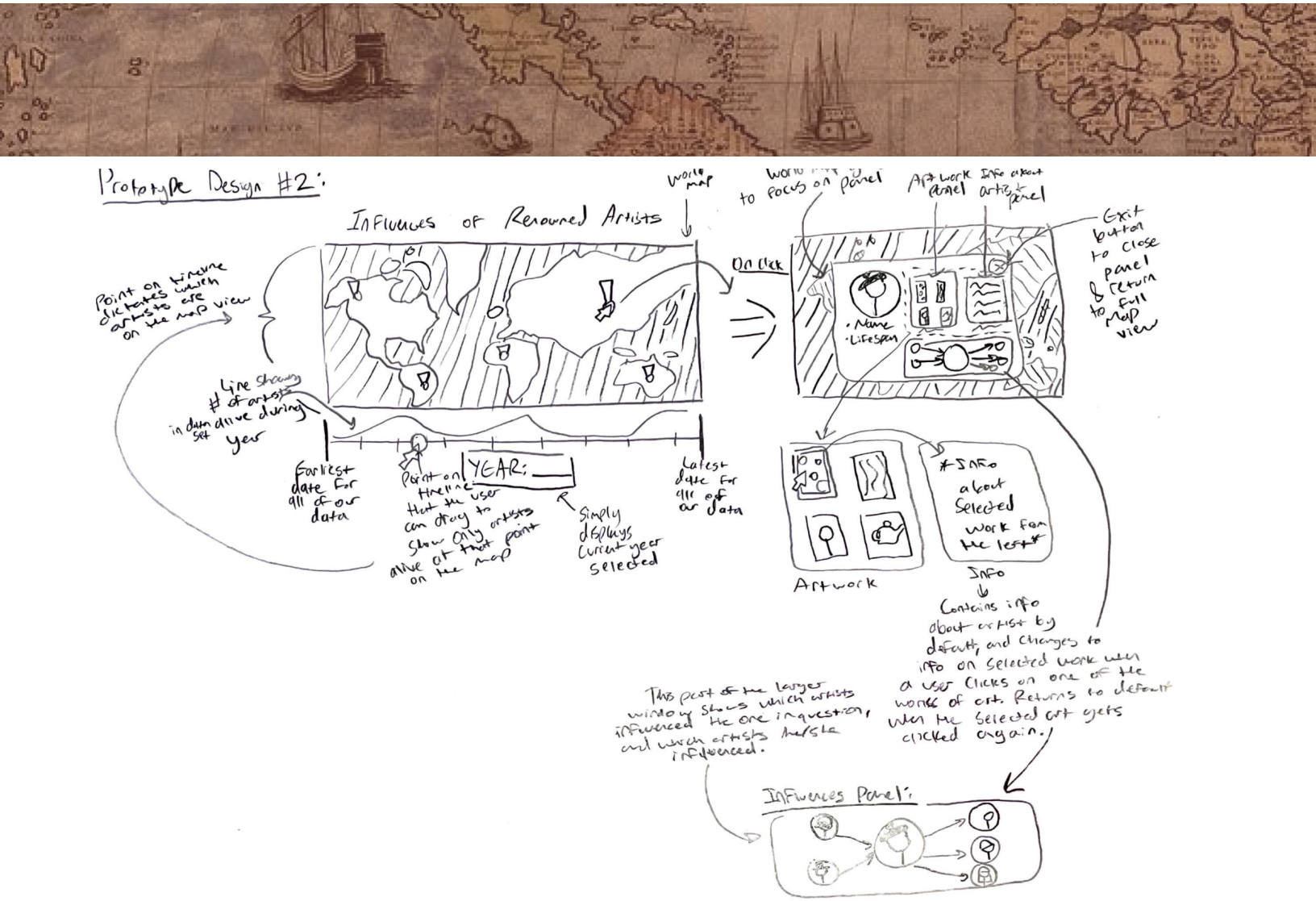


Figure _. *Mockup design sketch number two.*

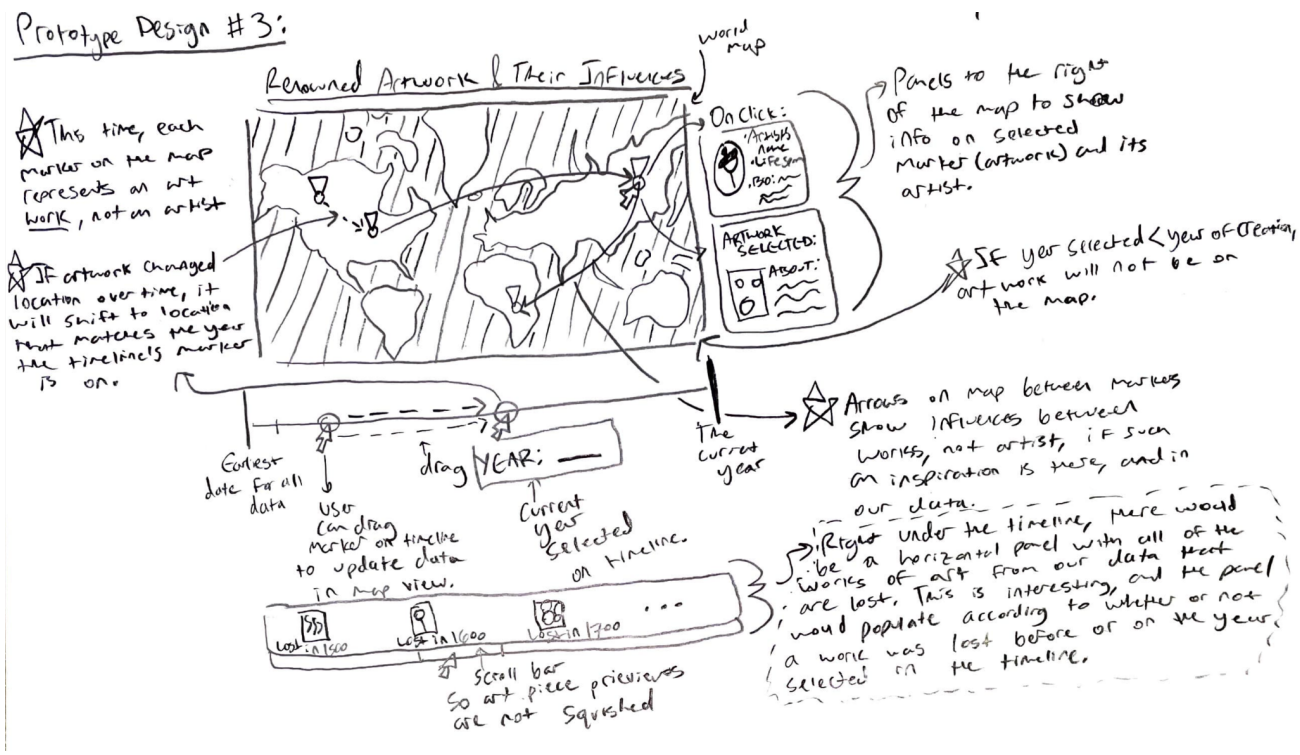


Figure _. *Mockup design sketch number three*

The last major sketch we drew before starting to code leverages the best encodings from the 3 prior designs. Our world map view uses the fairly strong 2D positioning as a visual encoding for artist location. Plus, the fact that it is a world map effortlessly establishes context that everyone can infer. Additionally, the use of a world map conveys the actual distance that the influence between two artists spans. The orientation, or shape, of our influence arrows encode the kind of influential relationship between artists. The timeline uses 1D length (which is even more easily readable than 2D) to encode time well.

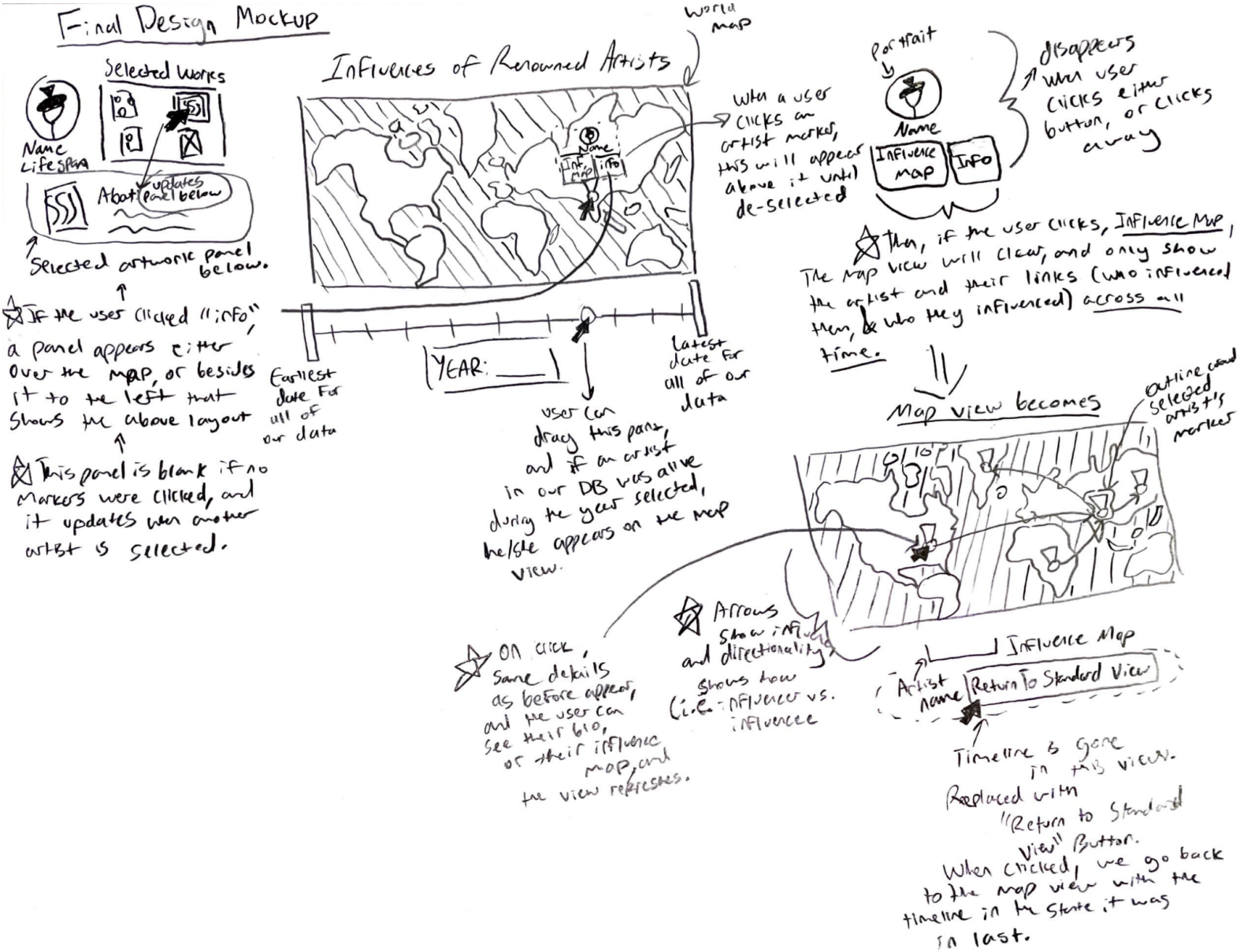


Figure _. *…..*

Post-Proposal: Reworking What's There

After getting a little bit of code off the ground and scrutinizing our proposal's ideas with our TA, we massively retooled our design. Starting small, we quickly sketched the idea of a D3 force physics simulation on our map view's nodes to avoid clutter (see fig. _ and *The Map and the Old Timeline* for more details).

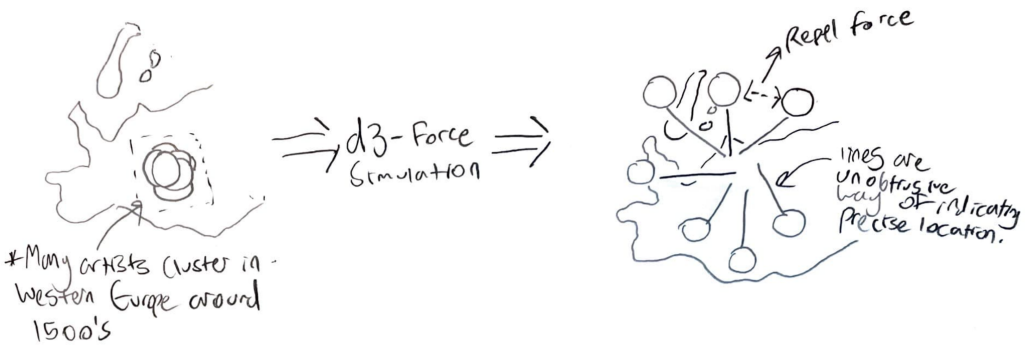


Figure _. …

Another simple alteration that would enhance our design relates to the influence arrows between artists. In addition to using directionality to encode the type of influence relationship, we want to use the width of the arrows to encode the time an influence has traveled (see fig. _).

Figure _. …

This encoding would add depth to the influence map and even contribute identifiable trends. For example, if the arrows are wide, that conveys that the artist's popularity saw a resurgence long after death. This also eliminates the need for users to calculate differences in years themselves.

With those minor final design aspects addressed, we can now elaborate on our major design overhaul between our proposal and now. The updated overall design (see fig. _) now has more views and some minor removals.

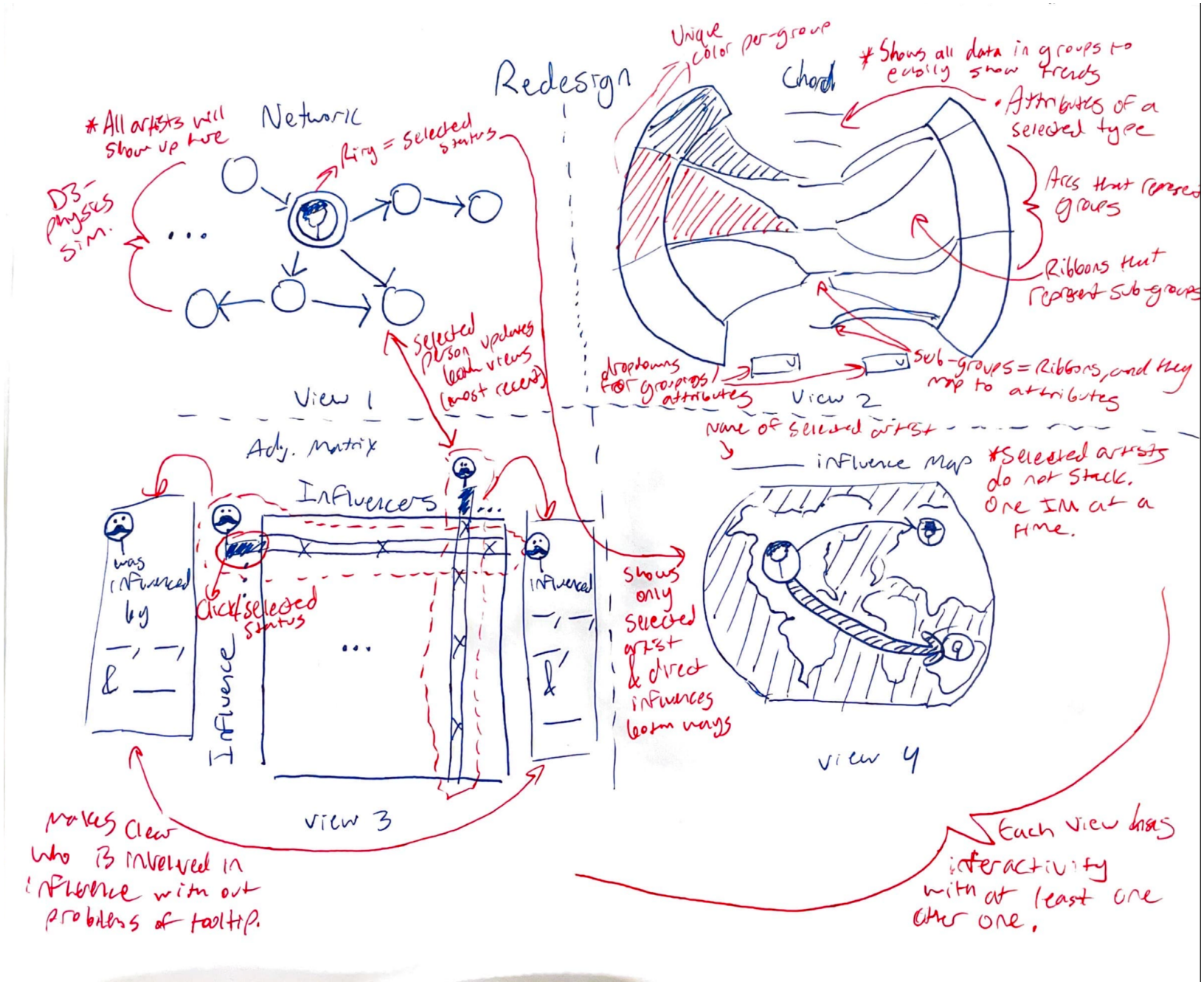
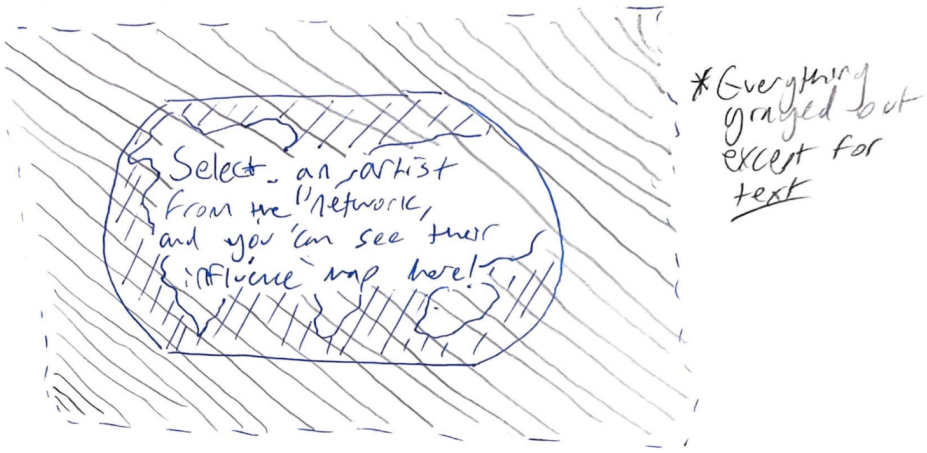


Figure _. …

This redesign drops the aforementioned timeline component entirely, and opts for a more open-ended approach with greater variety compared to our proposal. Meanwhile, the previous map view is now split in two: the network view and the influence map view specifically. This way we do not have to

worry nearly as much about node spacing (see *The Map and the Old Timeline*), and we have essentially modularized the complex map into a simple high level and a low level view. The former shows all of the data with ideal spacing, and the latter zooms in on the nature of a particular person's influences temporally and geographically. Also, if the user has not yet selected a node on the network that the map can focus on, we designed a simple visual state for that map (see fig. _)



Additionally, the chord component provides both more color and another kind of high-level overview. Meanwhile, the adjacency matrix provides both a high-level view by showing all of the data, and an individual view through its row and column highlighting. This overall lends itself much better to user exploration, and the level of interactivity we were aiming for here makes all of the components more connected in a way that complements them.

While figure _ provides a good blueprint for what we committed to make in the end, it lacks nuance in conveying the details of our brand new components: the chord visualization and the adjacency matrix.

Post-Proposal: Brand New Parts

Back when we finished our proposal's design, we were fully prepared to devote a lot of time to conveying information about both the artists' artworks and their biographies. In light of new design ideas, we have redirected the time that we would have spent on that to exciting brand new

visualizations. The first of these brand new views is our chord visualization. Our intention with this one is to provide a complete overview of our data in an aesthetically pleasing and unique way. One can see how we intended on accomplishing both of those things by looking at our fleshed out design for it (see fig. _).

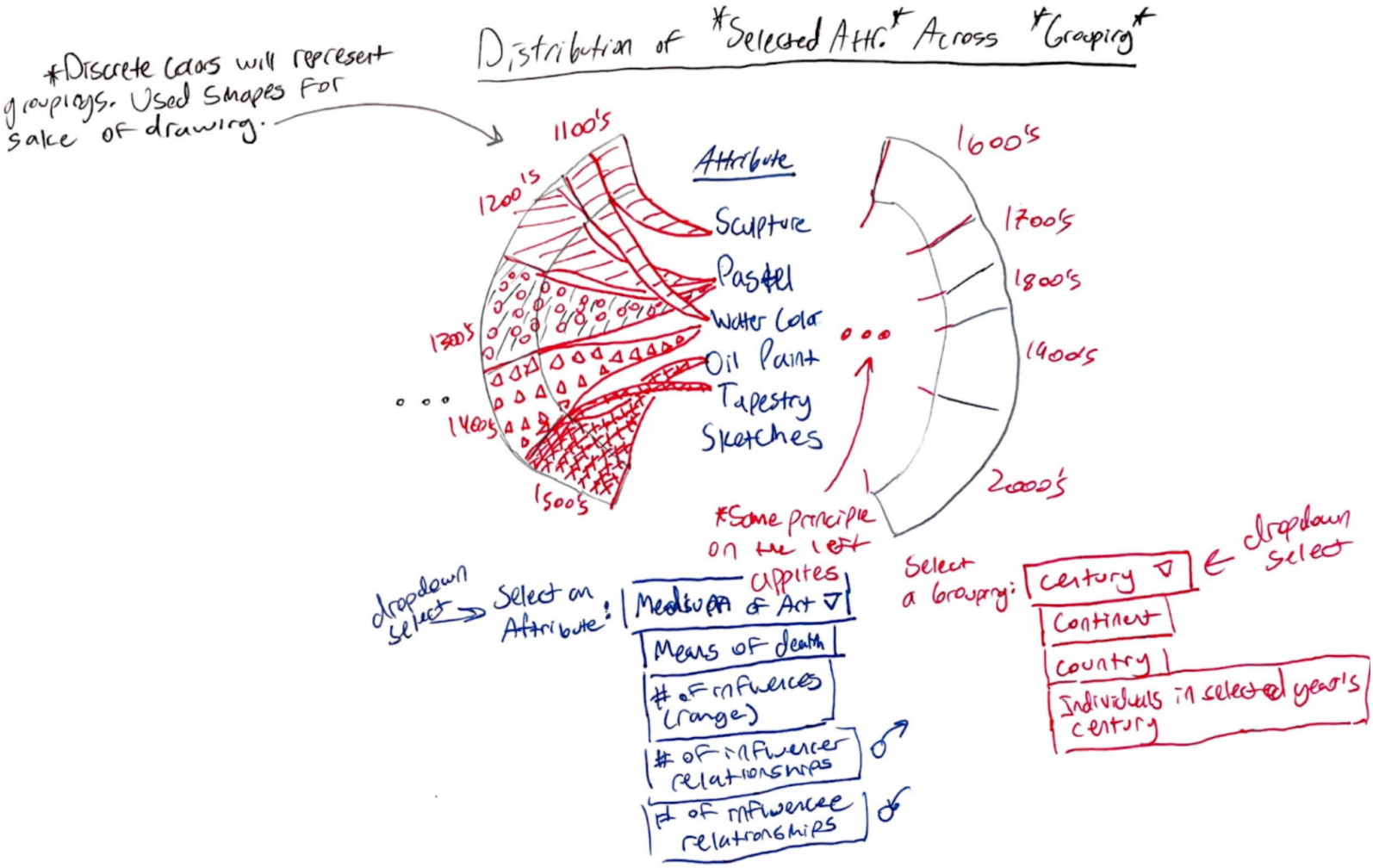


Figure _. …

A large part of the chord component's design is color. Color serves to encode each discrete grouping, and we want to choose colors that would make something nice to look at. Meanwhile, the ribbons in the visualization, that connect the arcs to their respective attributes, were a specific choice because of how stylish and effective they are. In relation to their effectiveness, they utilize 2D size and positioning, which

are fairly strong encodings Additionally, shape is a key encoding here, and curvature more specifically easily conveys trends among groups.

Something else unique about the chord visualization is the kind of data that it shows. While our main data relates to influences directly, here we make use of some data that we have which no other component conveys. That being artists' medium and means of death. Additionally, our possible combinations of groupings and attributes leads to more possibilities for a curious user and more data derivation (deriving the continent from our location data, tallying number of influences per person, etc.).

A way in which the design of the chord component developed further was us debating whether or not we want to show different kinds of groupings on either side of the attributes. So we contemplated having all of the data grouped by century on one side, and all of the data grouped by continent on the other for example. We liked the idea of showing more like that, but we did not really see what valuable takeaways a user could gain from a direct comparison like that specifically. Also, perhaps more importantly, we tested it out, and we simply did not have enough space to make it look good (see fig. _), and so the one grouping design remained.

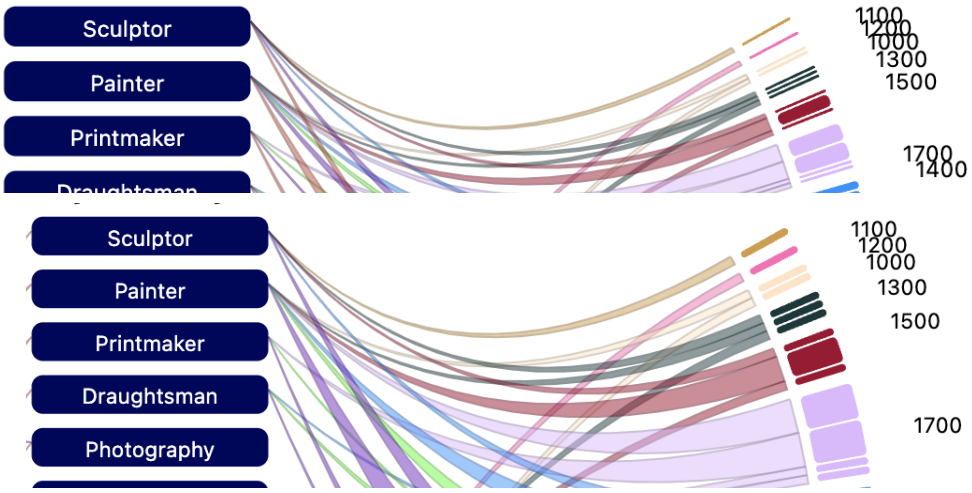


Figure _.

Our second and last brand new designed component is our adjacency matrix, and while it is more simple in some ways compared to the other components, we actually managed to get creative in a few ways during the sketch process (see fig. _).

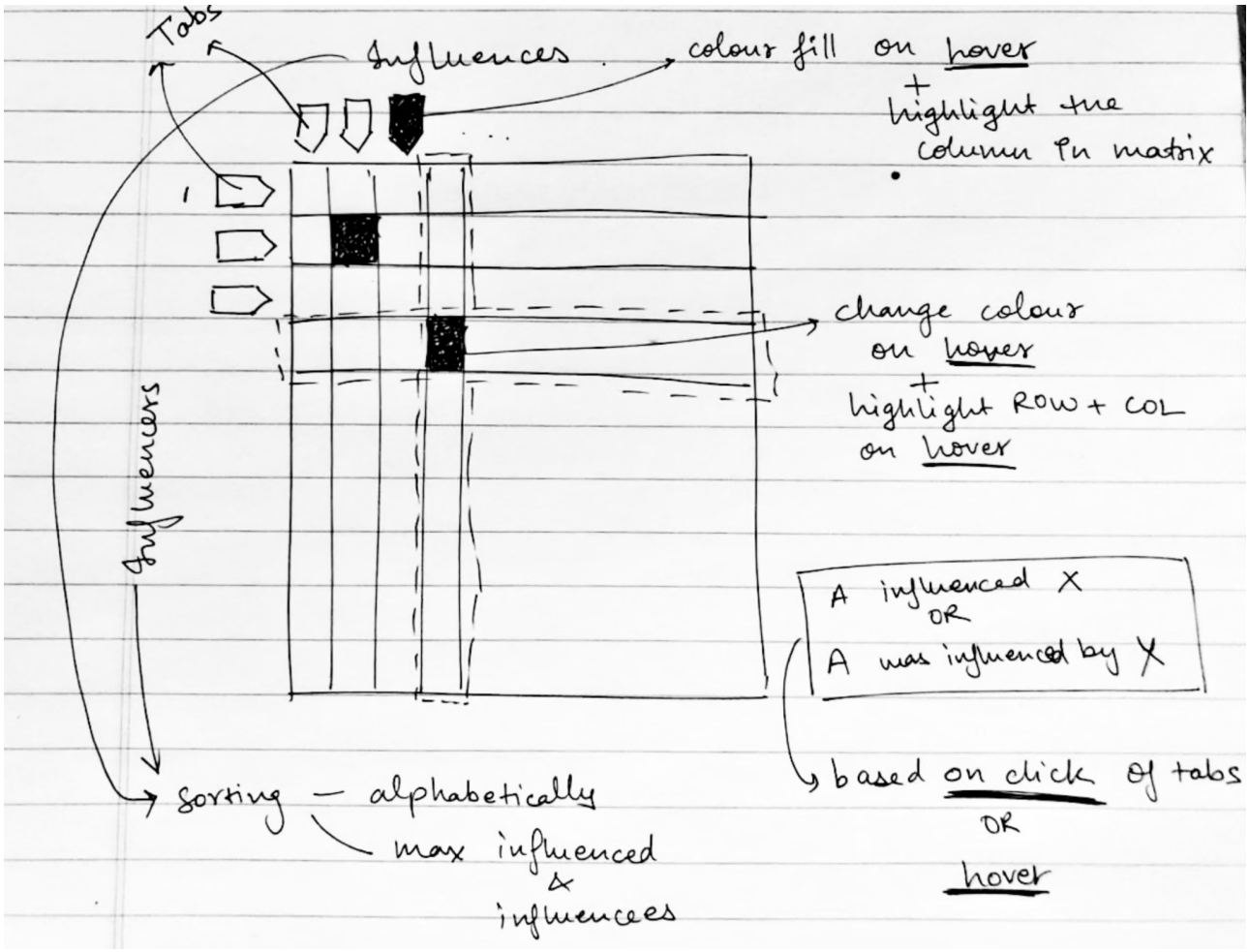


Figure _. …

Our design has many of the standard features of an adjacency matrix, and those fit our influence data well. One can see both the people who influenced somebody and those whom they influenced by looking at the person's row and column respectively. We designed this identification to be easy with color as an encoding for a selected cell or an entirely selected row or column.

However, the place where we get really creative is with the tooltip, or lack thereof. When contemplating how to show the information behind a cell explicitly, our minds immediately went to tooltips, but we thought of a less fleeting way to do the same thing. We designed this view to have one or two boxes off to the side that show the information behind a cell or a row and column combination when clicked. This allows for more calm analysis on the end of the users, and giving them the ability to sort the matrix can allow for easy insights.

Implementation (images should reflect final versions when talking about current state, so some placeholders for now)

Network and Map Implementation

While the network and map visualizations are different views within our grander visualization, they are far and away the most closely linked of all of our components. This should be obvious after we go through the various states that both can be in. Upon loading the entire page, the network visualization will show all of our artists as nodes with their portraits on them (after settling down from a brief physics simulation), and arrows between them showing the kind of influence between pairs of nodes. At the same time, the map component should appear grayed out with a message encouraging a user to select a node from the network (see fig. _) ***Insert final version of default state upon loading, and highlight the network and map components***. Looking at these two views independently of all of the others, nothing will happen to them until the user interacts with the network. For starters, if a user hovers over a node in the network, it will get bigger and the name of the artist will appear (see fig. _).

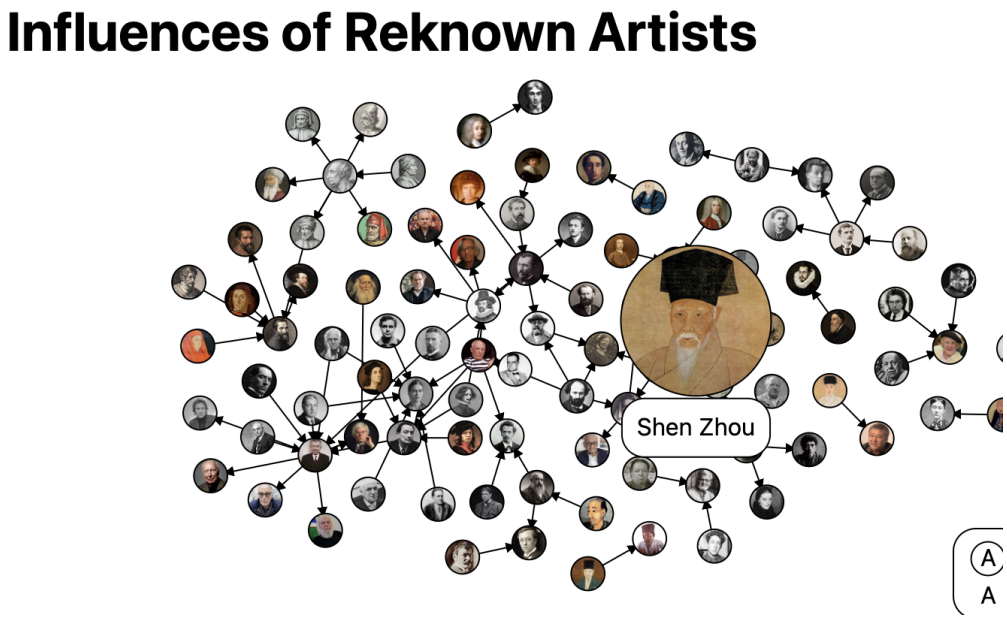


Figure _.

Then, upon clicking on any of those nodes, the selected one will have a unique border to indicate its selected status, and the map view will update to show that artist's influence map (see fig. _). ***Insert final***

version of influence map. It should be clear now that the state of the map is *entirely* dependent on user interaction with the network.

In terms of code, our implementation makes use of the D3-geo-projection package to create our Winkel tripel projection. This is the world map path for the map view. Our final version of these two visualizations takes the code that we created for our older map component, and splits it in two for reasons that we mentioned prior. That includes the physics force simulation on the network's nodes, which we made using D3.

Chord Visualization Implementation

Despite the chord view being more open-ended than the network or map views in terms of use, we can go through its functions state-by-state as we did before. Upon loading, the chord visualization will have a grouping and an attribute already assigned to it. This will most likely be century and artistic medium respectively ***(Minor thing, but maybe change. Random combinations on loading?)***. From there, a user can change the state of the chord visualization by first selecting different values in the two dropdowns for grouping and attribute (see fig. _) ***Insert picture of final controls for this view***. These can be any combination, the chart will update accordingly (including its title), and there will never be a case where none are selected for dropdown.

Another change a user can make by interacting with the chord visualization relates to color and focus. When a user hovers over a ribbon on the chart, its opacity will increase alone, making it stand out more (see fig. _) (***insert picture of this)***. Similarly, if a user hovers over the arc that contains all of the ribbons for a group, all of the ribbons within that arc increase in opacity (see fig. _) (***Insert final picture***). The same thing will happen if a user clicks on an arc instead, but the opacity change will remain until the user clicks that same arc *again* (the opacity increase does not somehow stack with this plus on hover events). The user can do this with all of the arcs if they want to hypothetically, but the purpose is to assist in focus.

The hover and click functionality for arcs also applies for attributes in a very similar way. If one hovers over an attribute, the ribbons that are connected to it increase in opacity for focus The ribbons remain that way if one *clicks* the attribute instead, and they stay this way until they click it again (see fig. _). (***Insert picture of this in its final state***). *Another function related to the mouse is a tooltip that will show the exact number of artists that a particular ribbon encapsulates (**Not 100% sure if we want this detail actually**)*

In relation to the other views, there is a common functionality across all of them in relation to the chord component. Whether it is in the network, the map, or the adjacency matrix, when a user clicks on an artist, the ribbon that contains that person will increase in opacity until the user changes focus to another one with the mouse or changes the whole thing with the dropdowns. There will never be a case where this does not happen because this component shows all of the data no matter the grouping, but there very well could be cases where multiple ribbons highlight in response to these other views. This is because for some attributes, it is entirely possible that they have more than one. For example, da Vinci was a painter, but he was also a sculptor, an architect, and a draftsman. Thus, all four of those corresponding ribbons in his grouping would highlight (see fig. _) (***Insert example of this***).

Lastly, the chord component's technical implementation was a decent challenge for us conceptually. When initially looking at D3 chord visualizations for inspiration (see *Inspirations While Revising*), we soon realized that what we wanted was different from what D3's chord and ribbon code provided. Thus, we went into D3's code for chords and ribbons, learned what they were doing, and developed something similar. Our method starts by simply calculating start and end angles that each ribbon will need. We also used these angles to make the arcs on the perimeter of the chart, and they are each padded to separate groups of ribbons spatially. The most interesting part was then generating the ribbons. This process relied on D3's support for quadratic Bezier curves, and its arc() function. For a somewhat detailed explanation, see figure _ (do keep in mind that this only details the general process, not the many minor adjustments that we needed for fine tuning).

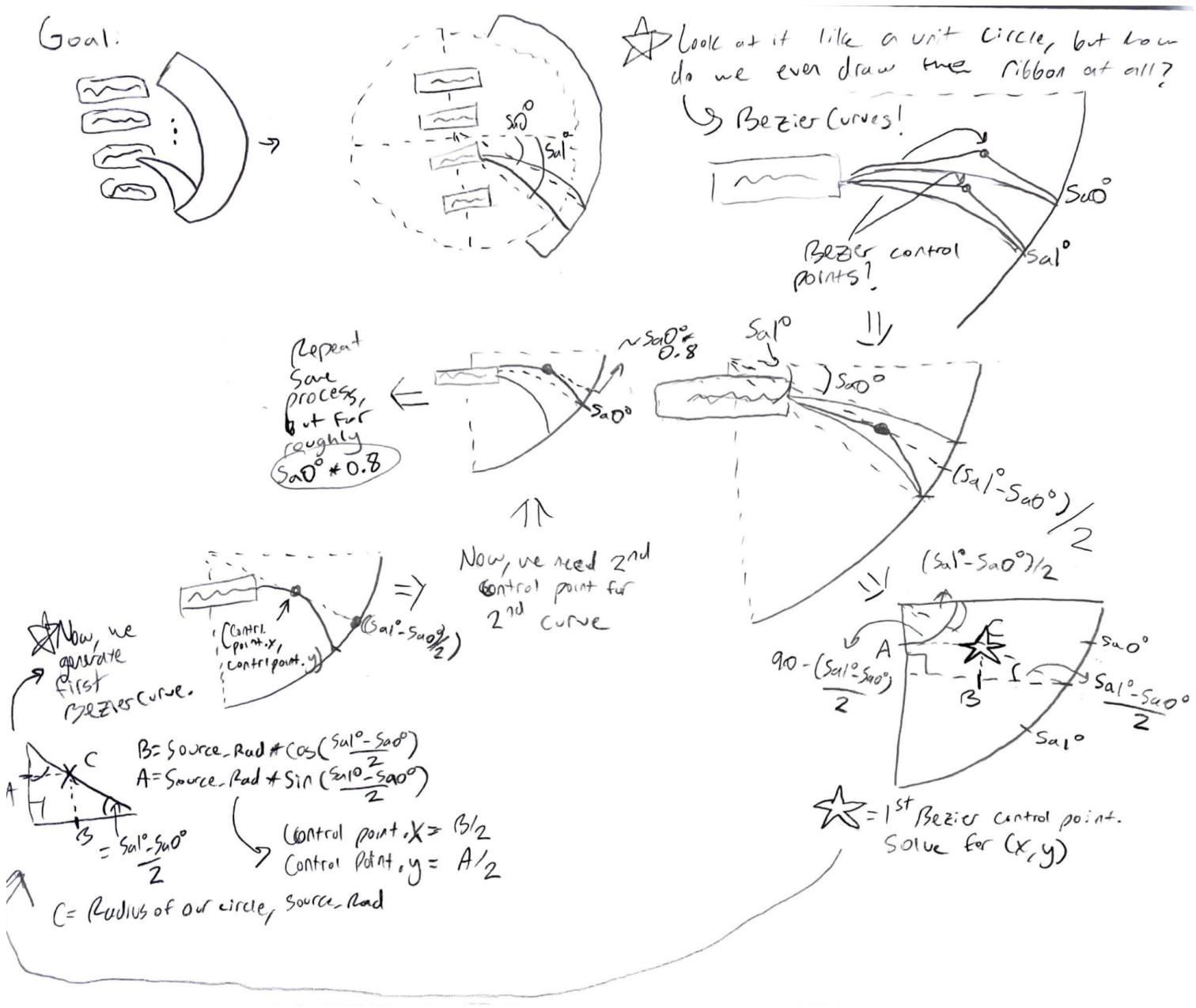


Figure _. …

Adjacency Matrix Implementation

Last but certainly not least, our adjacency matrix view has some functionality that is worth noting. When looking at the view alone, we can start by discussing its mouse-based functions. When the user hovers over a cell in the adjacency matrix quite a bit happens compared to our other components' hover events. Not only will the cell change color to indicate which one the user is on exactly, but the column

and row that the cell lies in will also gain focus through a change in color. Then most importantly, a text box near the matrix will say what the relationship is that the cell represents (see fig. _).

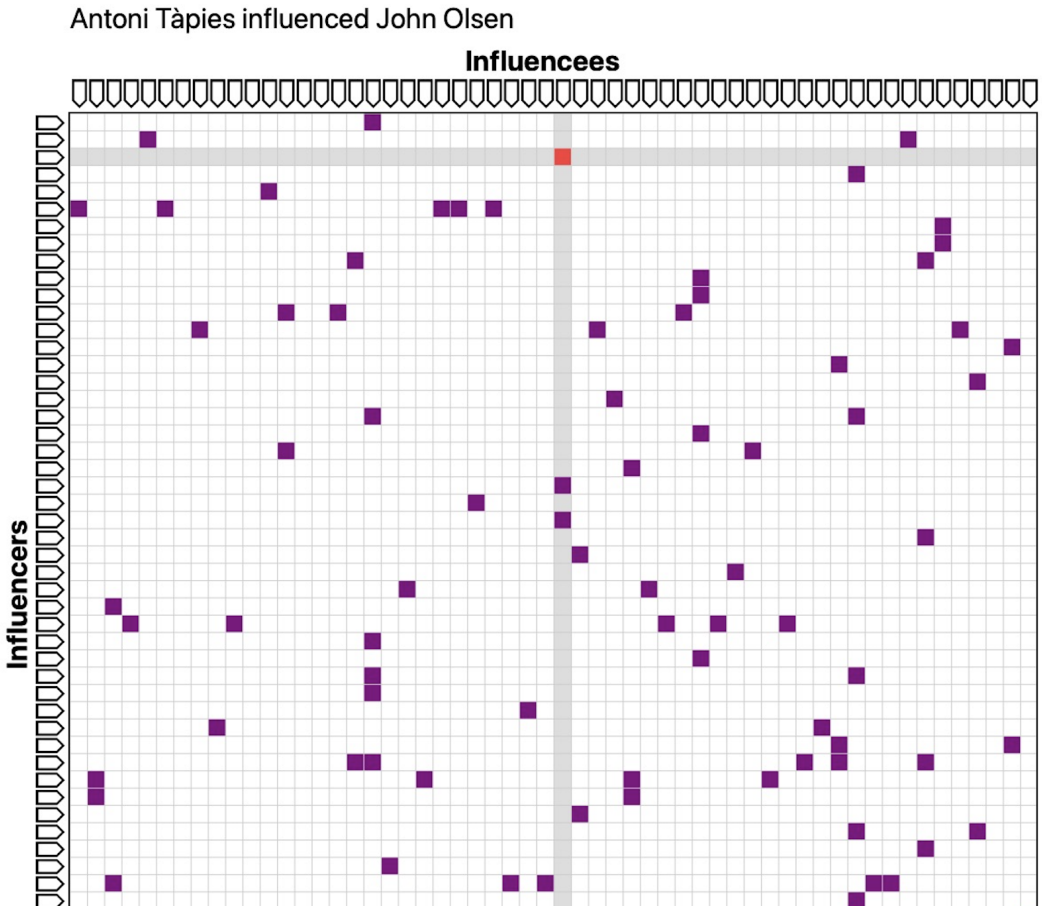


Figure _. ..

A user can also click on the tabs over the columns and rows depicted in figure _ in order to select the artist that it corresponds to. The reason for this is to display *all* of the direct influences either to or from that artist above and below the matrix respectively (see fig. _) (***Insert final fig of this***). This event in particular has noticeable impacts on other views. Similarly to clicking a node in the network view, clicking an artist tab here populates the map view. Additionally, the chord visualization will highlight any ribbons that artist falls under, and that person will have the selected visual status in the network view.

Finally, there will also be a visible dropdown that will allow the user to sort the matrix according to the number of total influences per person, number of people who inspired them specifically, number of people that they inspired explicitly, etc. (see fig. _). (***Add final image when this is in***).

State diagram showing overall interactivity is a work in progress and might change even more/we might not even need it

Evaluation

Yet to be written