cs171 - Final Project

Ceruti, Mychal

Process Book

2014.04.08

Table of Contents

CS171 | Final Project

CS171 | Final Project

Hollywood Actor's Bankability Over Time (1993-2013)

This visualization provides you with an interactive and engaging visualization showing the rise, fall, and overall performance of the most popular Hollywood actors and actresses over the last 20 years.

Proposal | Inspiration

Related Work

The Hot List

Most everyone has heard of the A,B,C, and D-list. The terms were created by veteran journalist James Ulmer to gauge the bankability of actors and directors...

Proposal | Questions

Proposal | Problems

Early Issues

Difficult to Quantify

Although there exists a wealth of data available online on movies, including cast lists, rating, budgets et cetera, finding consistent, historical aggregated data that quantify an actor's talent, popularity, and other attributes that are essential to determining their bankability (defined by The Hot List's page as: *" the degree to which an actor's name alone can trigger full financing for a movie.".)* Finding this data has proven to be a hurdle that may be impossible to overcome with the time given financial and time constraints. As a result, I will still endeavor to create a visual tool that can effectively compare celebrities' careers over time.

Data is Everywhere (and they want you to pay for it)

To determine an actor's ability to "put butts in seats", there are several factors that have to be taken into consideration. The Ulmer scale, for example takes into account talent, ...

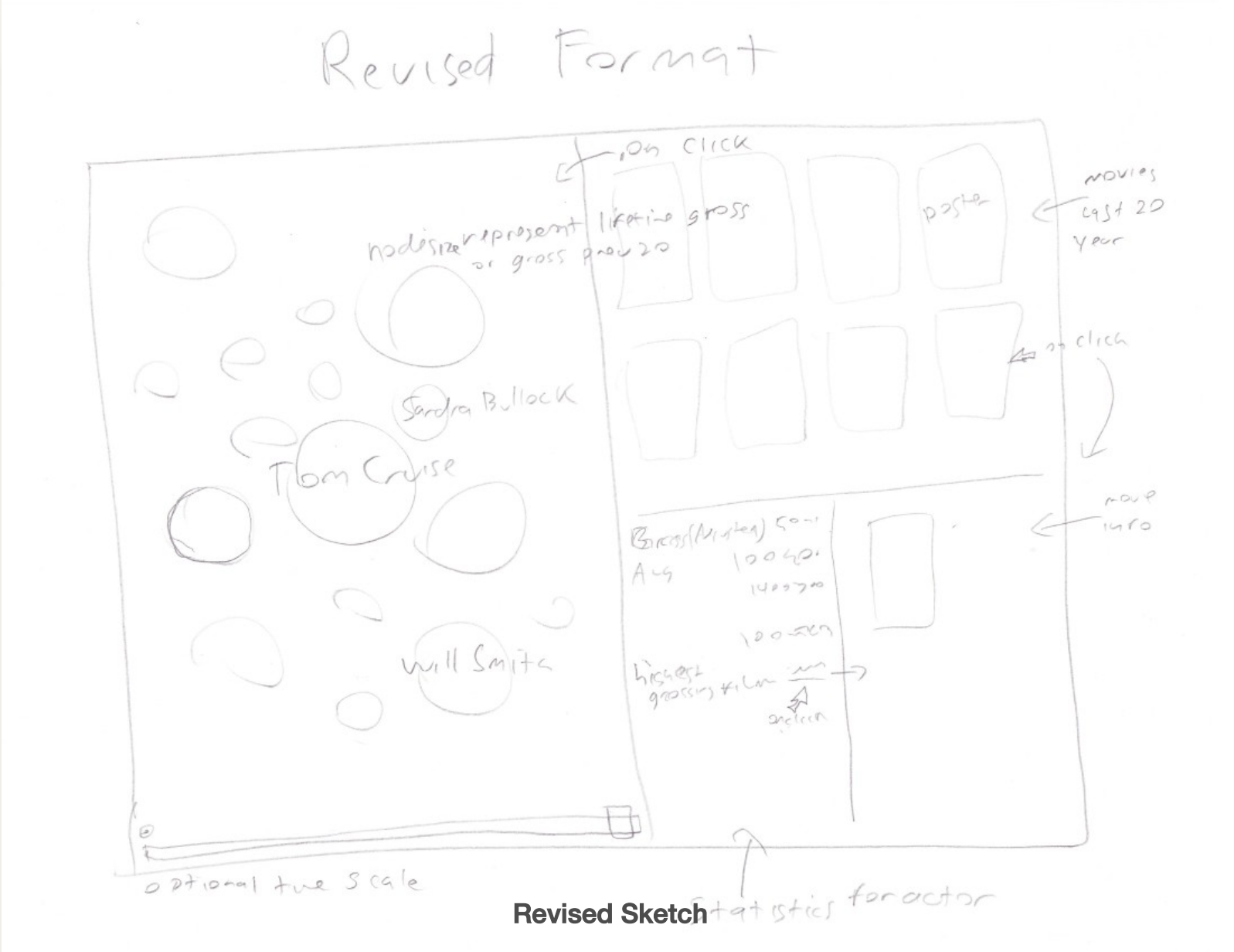
Change of Direction

For Design Studio 3 we were tasked with providing and receiving feedback from other groups as to the design and scope of our project. During my interview with Yash, he pointed out that while interesting, my goals may have been a bit ambitious for what would be expected from a one-man-team. After a short brainstorming session, we had come up with a solution that, while moving away from my original vision, would still achieve what I had set out to accomplish, namely a fun, engaging tool that would show the development of an actor or actors over a span of time in their careers. The solution allowed me to use much of the code I had written for my original designs, and I was introduced to another source of movie data that was robust enough and accessible enough to grab all the data I needed for my visualization.

I will be leaving in my previous methods and the type of data that I had grabbed so far for the purposes of showing my process, but as of 2014.04.10, I've decided to make a fairly major change as far as the scope of my project. Despite the new direction however, the visualization will still accomplish what was the primary objective of my project

Proposal | Change of Direction

.



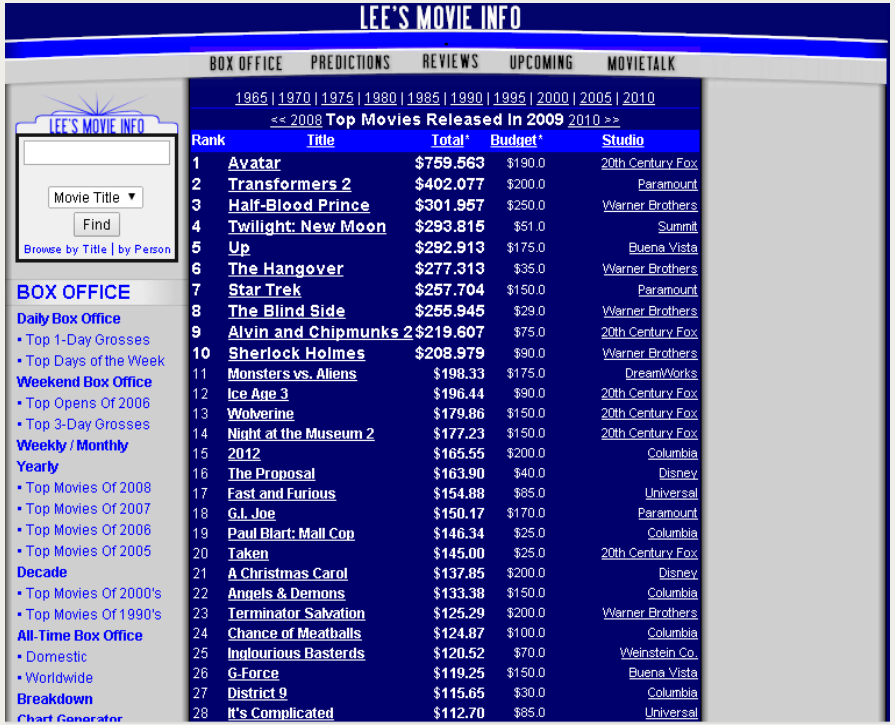
Revised Sketch

Process | Data

Data Collection

After I had a rough idea for what I wanted to accomplish, I began my search for sites where I could gather the data that I needed. IMDB and RottenTomatoes.com both provided APIs that allowed access to much of their data, but scraping data some of the other websites proved to be a little more difficult.

Lee's Movie Info: I needed to find an easy way to gather lists of the top movies from last 20 years. My first choice was Wikipedia or IMDB, but as I was still wrapping my head around the use of the IMDB API and writing code that could gather that information from Wikipedia would have proven too time consuming I settled on manually wrangling the data from Lee's Movie Info. Lee's Movie Info has its top 50 list of movies sorted in tables that were easy to copy and paste into a spreadsheet which I then converted to a CSV file. Though the site had a lot of other valuable information, I couldn't find an API that granted me easy access to it so I decided to look elsewhere.

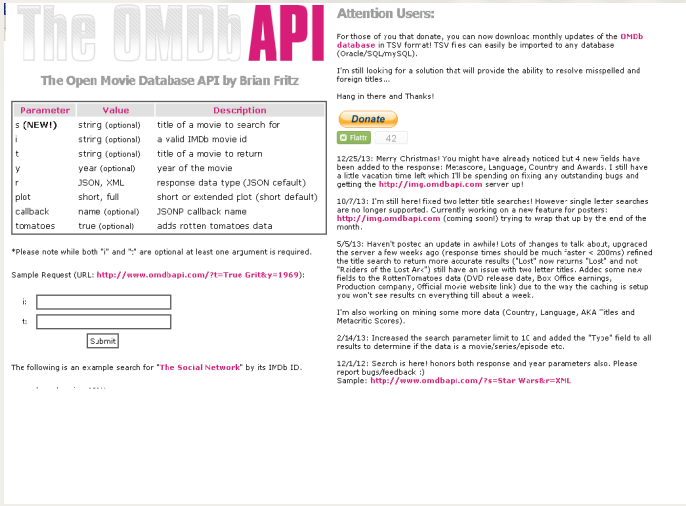


Lee's Movie Info: Top Movies of 2009

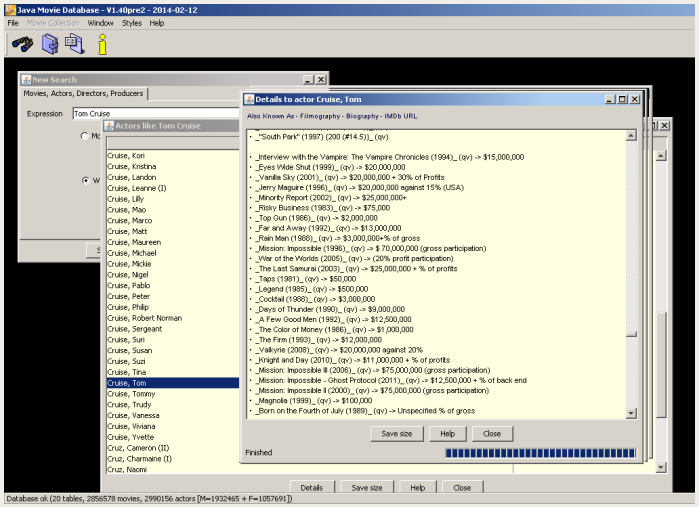
Process | Data

- **OMDB API:** As IMDB is one of the largest, most comprehensive, consistent, and free databases of film data, it was an obvious choice. While it does have has a publicly accessible API, it doesn't have any official support or documentation. Thankfully, there are many websites that provide their own methods for accessing IMDB to collect data. I used OMDB API(http://www.omdbapi.com/).

Java Movie Database: IMDB does not allow screen scraping, and all of the public APIs I found that accessed IMDB's didn't provide methods for scraping the salary data, an essential component of my project. IMDB provides plain-text versions of most of their data, but the files are large and difficult to parse through. Because of this I searched for a tool to make the job easier and came across the Java Movie Database (JMDB). JMDB loaded all of the plaintext data into a database on my local server and provided a GUI that allowed me to search and export all of the information I needed. Though this was ultimately more tedious than gathering information programatically (which would have been possible, but I felt I didn't have sufficient time to learn and write code that searched through the databases), it got the job done.



OMDB API Homepage



Java Movie Database

Process | Data

Cleanup and Structure

The revised visualization required much more compact data, and was more easily accessible and can allow for a more dynamic graph. The original data structure used 2 sets of data, one for movies and one for actors. …

The most recent structure has...

Old Structure

Dataset 1

movie:{

Dataset 2

New Structure

Dataset 1

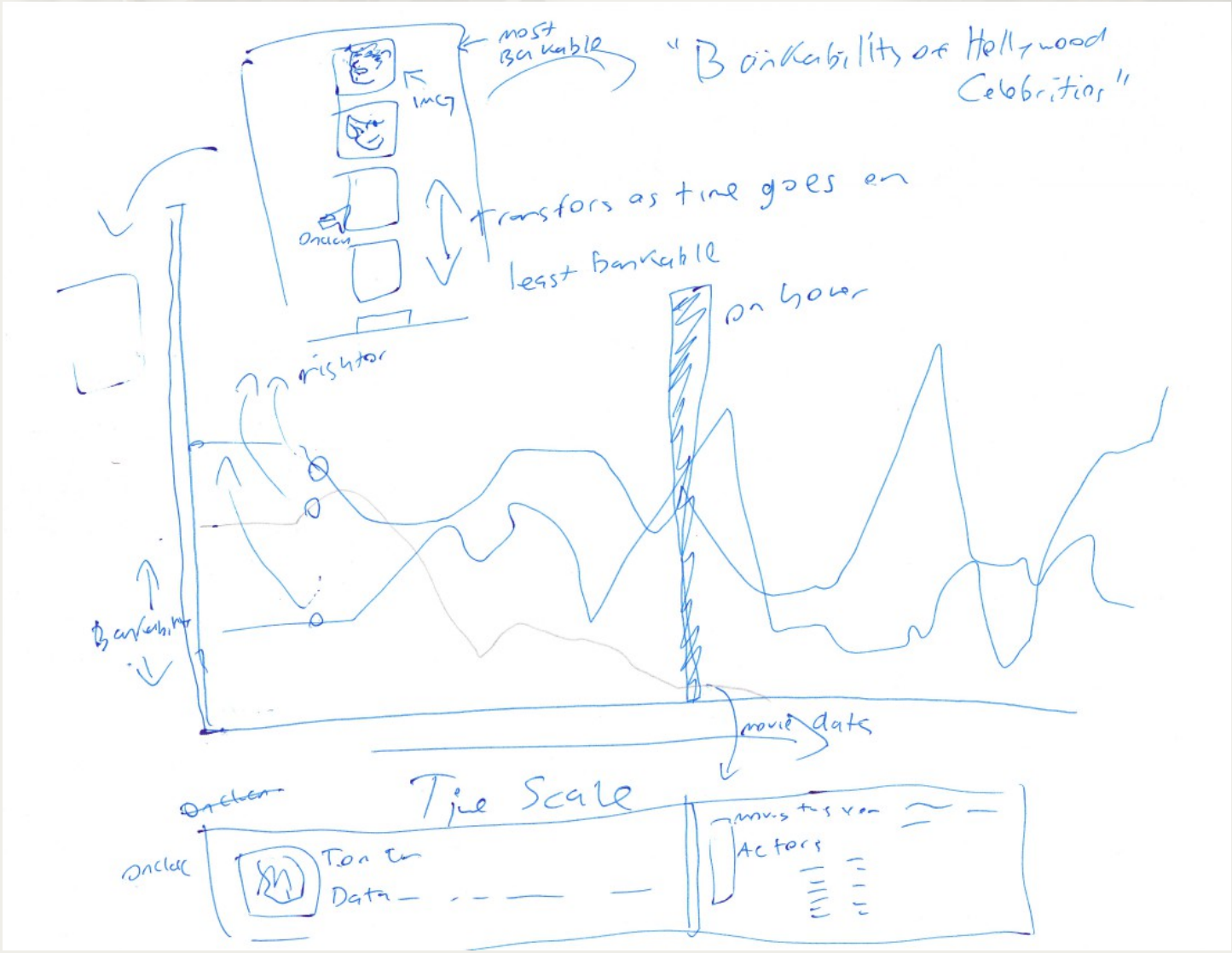
[movie:{grossDomestic:112, grossGlobal:238.2, sum:350.2, actors:[array](list of actors),releaseDate:{dateobject}, poster:"http://www.bar.com/poster.jpg"}, movie2:{gross...}]

Dataset 2

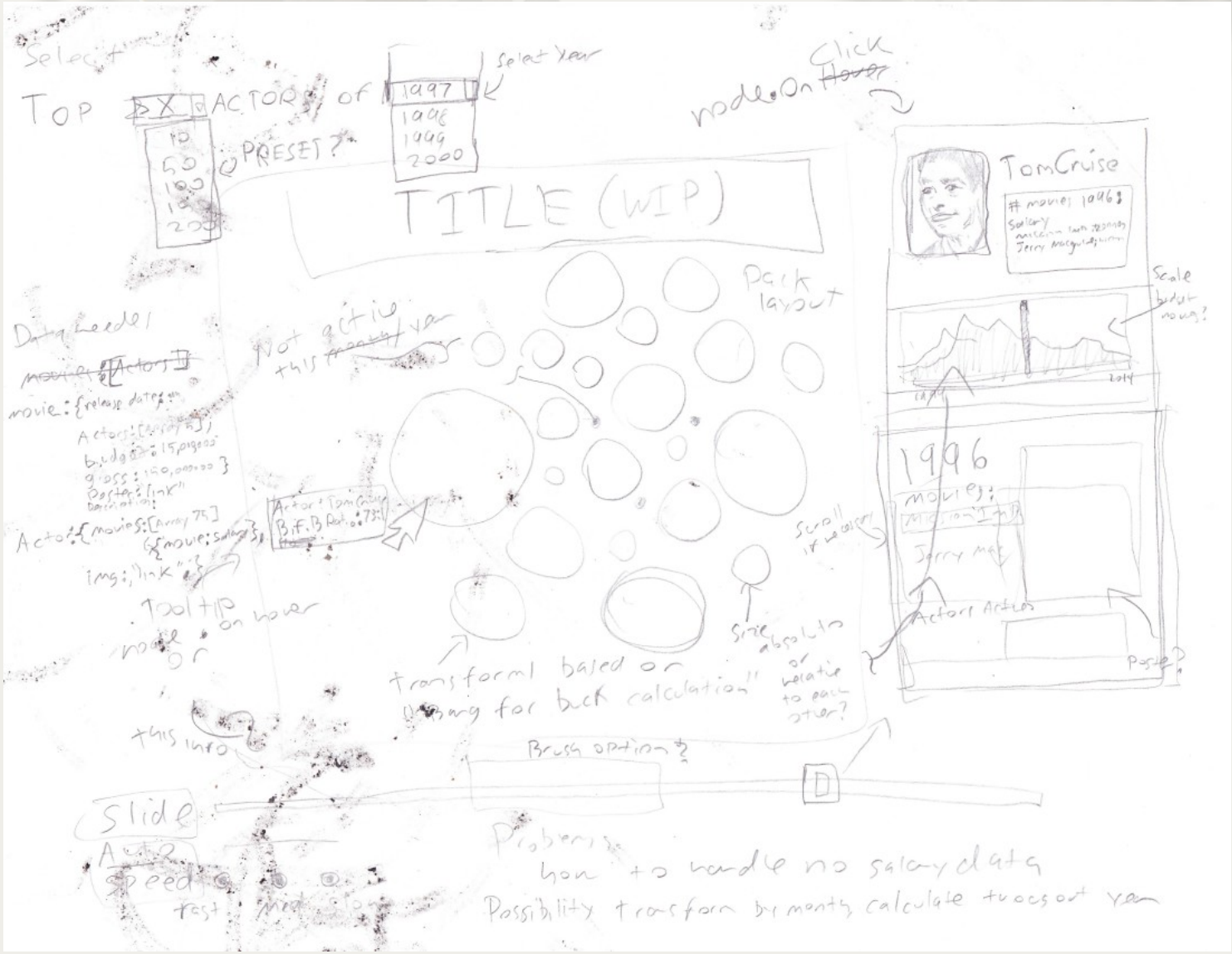
[actor:{careerGross:409.3, image:" http://www.foo.com/actor.jpg", movies:[array](list of movies)}, actor2:{careerGross... }]

Process | Analysis

Visualization | Design



Visualization | Design




Visualization | Design

Visualization | Features


VISUALIZATION | ANALYSIS

Conclusions

Bibliography