Information Visualization: Milestone 2

Project: *Console Wars*

Team: *Marks & Channels*

March 25, 2020

README.md

1. *Rationale for design choices*

   Data Abstraction

   The data that we worked with includes genre (level: 11), year (range: 1980-2020), and platform_comany (level: 5). User can select a level or a subset of year range to hone in on subset of data.

   Task Abstraction

   Game Controller View: The view allows for exploration of the dataset divided into few major categories based on an attribute. The exploration is facilitated by the hovering and clicking actions, which allow the user to view additional information.

   Main View: Compare attributes among the different categories of data. This is facilitated by the use of widgets (filters) that allow the user to reduce the number of items they view on the screen.

2. *Description of the visual encoding choices*
   - Main View

     2D Size: encoding Global Sales of a game

     Spatial Region: encoding console company (Sony, Microsoft, Nintendo, PC, Others)

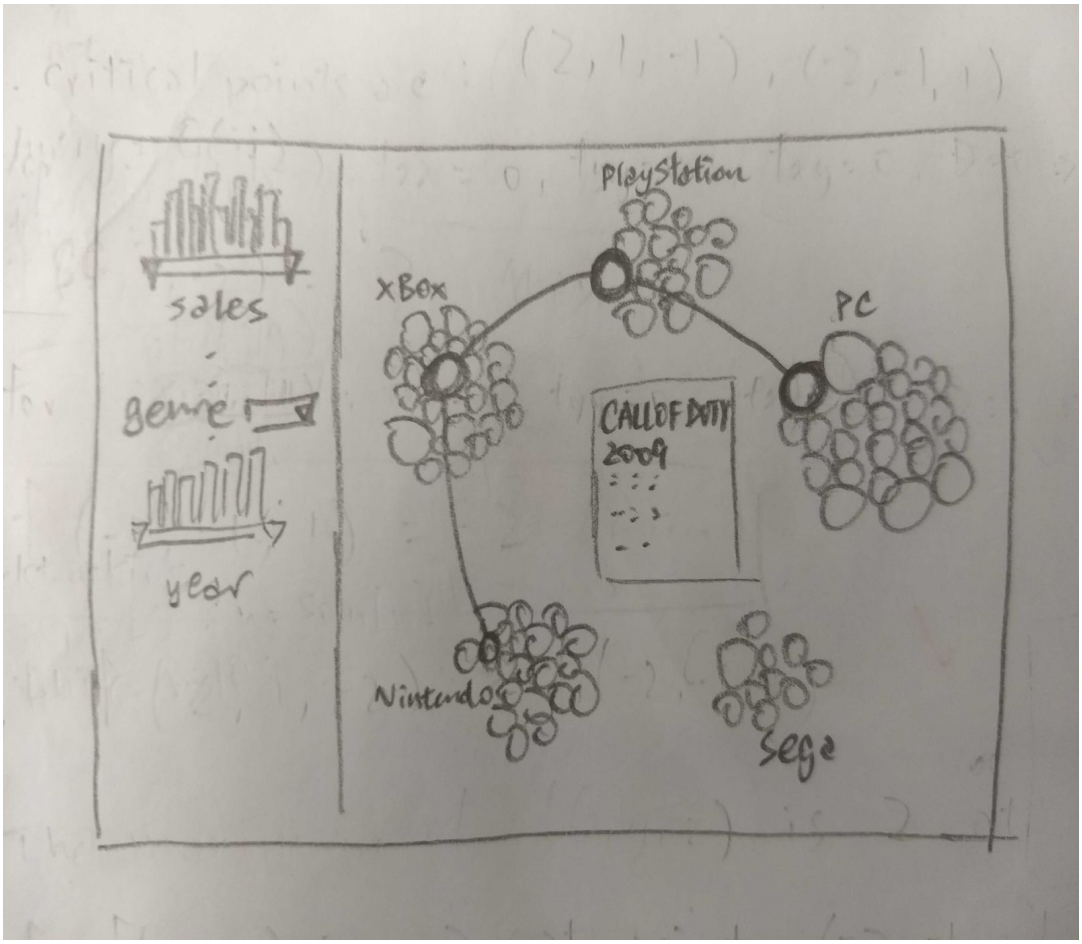
     Color Hue: encoding which game has been selected (and its releases on other consoles)
   - Filter View

     Widget: dropdown to filter games by genre

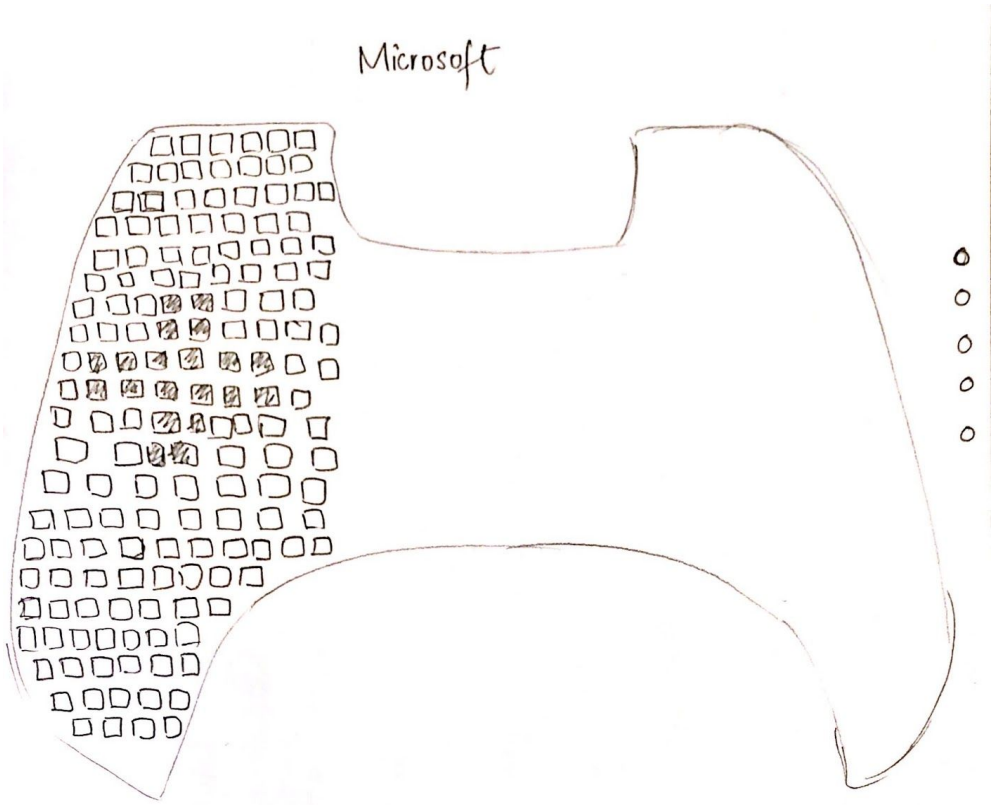
     Widget: d3-brush to filter games by year of release, horizontal position encodes the year
   - Game Controller View

     Point Mark: Each game is a point on the controller

3. *Change in vision since proposal*



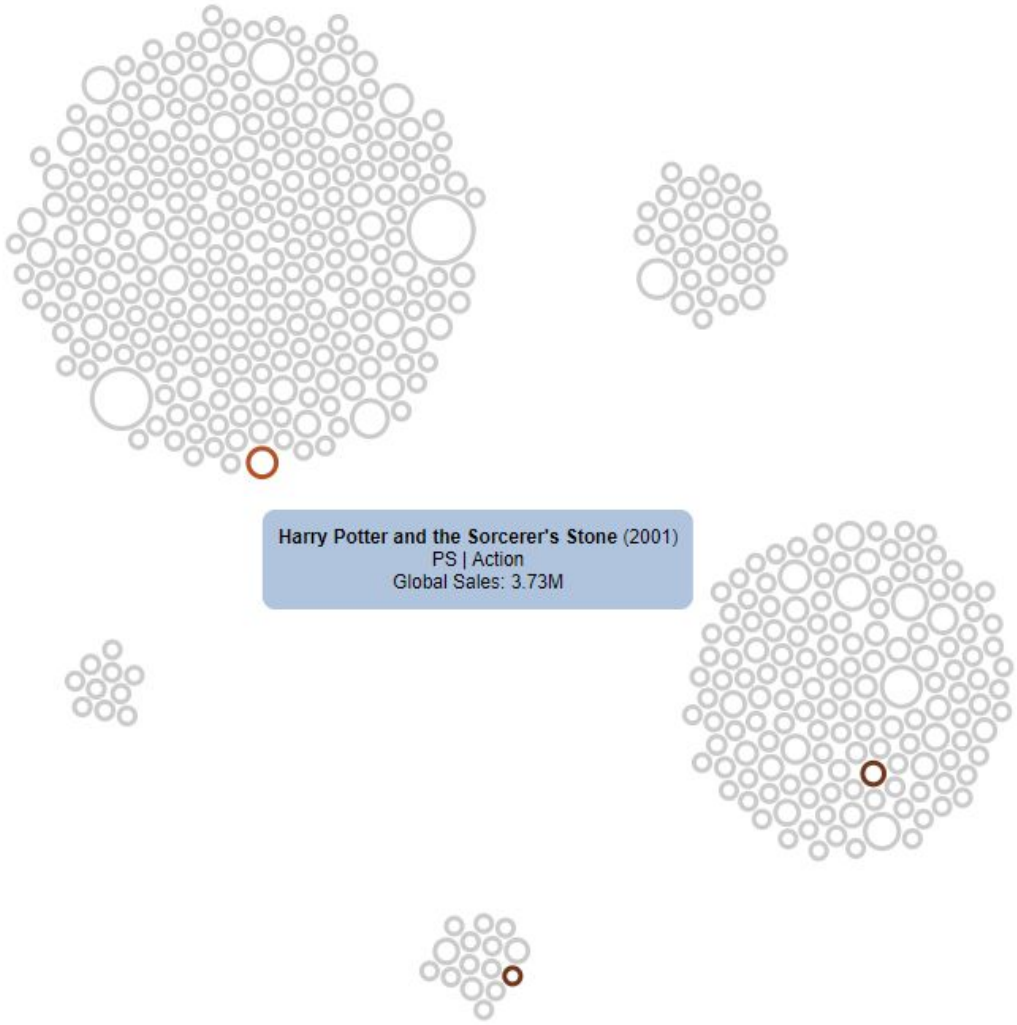
- The Main view (right side of the image above) displays each game as a circle node, and they are grouped in clusters of their console_company. When a game is selected, the same game released on a different console will also be highlighted and planned to be linked by a line (not implemented yet). More information about the selected game will be displayed in the center (for now, only text, but we imagined it can show the regional sales of that game in a world map.
- The **Filter view** currently includes **two filter widgets** — a dropdown for selecting the **genre of games to display**, and a brush slider to choose a range of years of game releases. Any change in the widgets will subsequently reflect in the Main view by displaying only the matching rows.
- In light of TA's comments about fitting 15k points on the screen, we divided the points into the platform company type. We are now displaying points for each of the **platform companies** in a separate controller. The user will be able to **scroll** to see the multiple controllers. On hover, the user will be able to see a tooltip for more information about the game. The user is able to **switch between the main view and the controller view**.



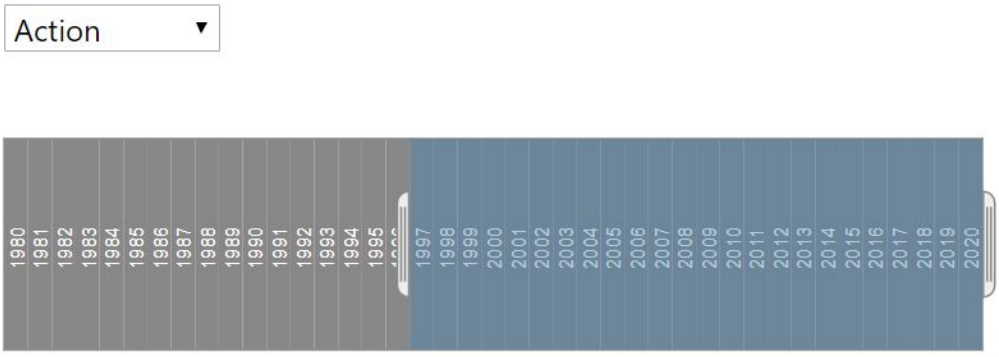
- *Does our visualization enable the tasks we set out to facilitate or successfully communicate the story we want to tell?*
    - For game enthusiasts, the layout allows them to **explore** the **history** of console wars.
    - For game marketing personales, the layout allows them to **filter** by desired attributes like year, or genre to **identify trends** and **patterns** (e.g. in recent years, which console to release an action game yields higher sales?)

4. *At least one screenshot that illustrates our current prototype*

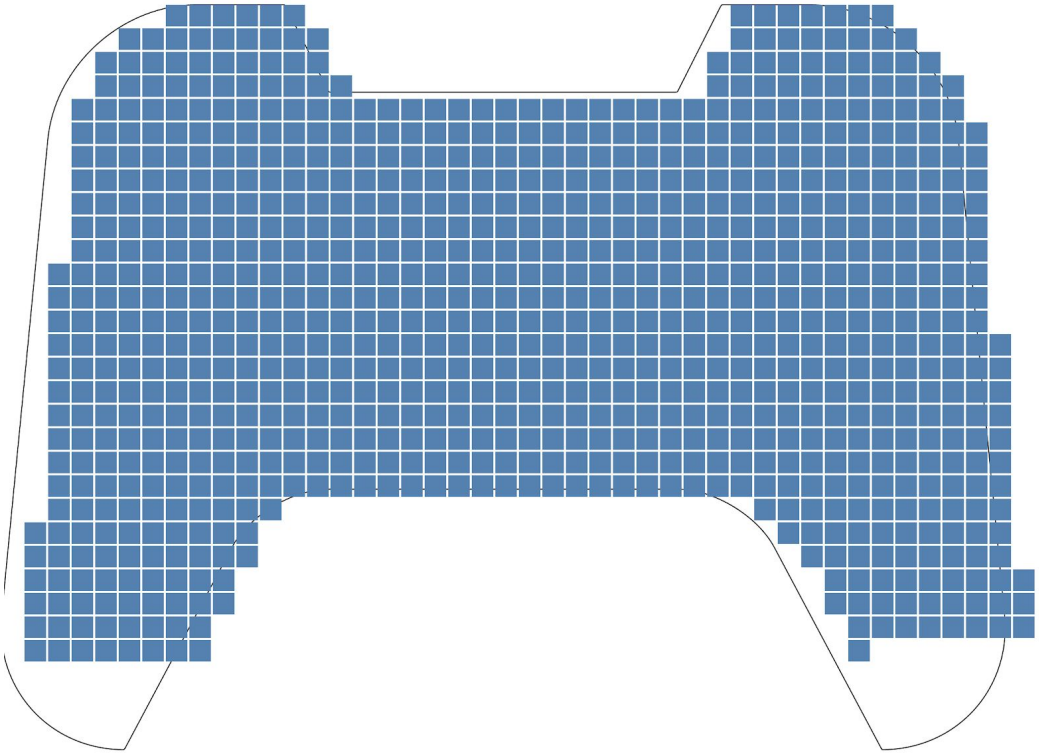
Main View



Filter View



Controller View



5. *Link to the original data source*
    - https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings

6. *Brief description of current data preprocessing pipeline, if there is one.*
    - We modified the original dataset first by removing items that did not have a game name (removed 2 items). We also noted some items have “N/A” for some of its attributes like publisher, genre or year, but since we are focusing on creating static views for now, we can deal with these items with “N/A” attributes later when we implement filtering.
    - In terms of runtime processing, we are parsing the dataset in **preprocessing.js**. In this file, we have different functions to parse the dataset into different data structures/formats needed for different views. The corresponding formated data will be set in their corresponding view Class. (i.e. mainView.data = formatDataForMainView();)

Project Management

1. *Status update*
    - *Update work breakdown and schedule with current status. How long did it take for each chunk of work, in comparison to original estimate? Are there any new chuck of work we needed to do that wasn’t originally documented?*
        - **Main View** - took around **30 hours**, around the original estimate. Most of the time spent on figuring out D3.force(). The tooltip in the middle is also implemented, which was not originally documented.
            - Fix issue breaking the clusters and force when merging with the filter view - 4/4
            - Optimization to reduce potential lag - 1/4
            - Fix cluster separation problems - 4/4
            - Revise design to label clusters with respect to cluster size - 8/4

        - **Filter View** - took around **25 hours**, which was a lot more hours than expected. They were mostly focused on project restructuring, which paved the way to figuring out how data communication between the Filter view and the Main view would be done; the rest was spent on actual filtering of data rows, implementing the brush slider widget, researching into optimizing and reducing lagging, trying external libraries in the process, and so on.

- **Game Controller** - The Game Controller till date has taken around **20 hours**. Most of the time has been spent figuring out the math behind fitting points inside a path. We are still trying to figure out how to show all the points for the controller while not leaving any excess white space. The future work will include adding tooltips for each point inside a controller. Once the points are in place, we will fix the css to make the controllers more colorful (resembling actual controllers)

2. *Contributions breakdown*
    - *Which team member worked on which tasks and their responsibilities? Did everyone contribute equally?*
        - We all worked on a view; Megha worked on the game controller view, Nicole worked on the filter view and Wesley worked on the main view. We contributed equally.

3. Team process

	Weak	**Satisfactory**	**Good**	**Excellent**	**What are specific actions you want to take to address issues, if there are any?**
Team has a clear vision of the problem(s)				✓	
Team is properly organized to complete task and cooperates well				✓	
Team managed time wisely				✓	
Team acquired needed knowledge base				✓	
Efforts communicated well within group				✓	