

Data Visualisation COM-480 Milestone II

Xinran Tao, Ahmed Ezzo, Guanqun Liu

May 6, 2022

1 Tools & Association with Lectures

Tool	Usage	Lecture
React.js	JS library for main user interface development	Viz, Text Design
Three.js	JS library for creating animated 3D computer graphics in web	Creative Coding, Data Art
D3.js	JS library for creating dynamic and interactive data visualization	Interactive D3.js
Leaflet	JS library for dynamic and static graphs display	Data Abstraction, Graph Viz
Pandas	Python library for data processing and analysis	Data Acquisition & Processing

2 Goals & Sketches

Our project goal is to provide potential game developers with the latest trend in popular games and hot topics in game development. We aim to provide an intuitive reference to developers about rising stars in the game industry and market change over the period.

2.1 Overview

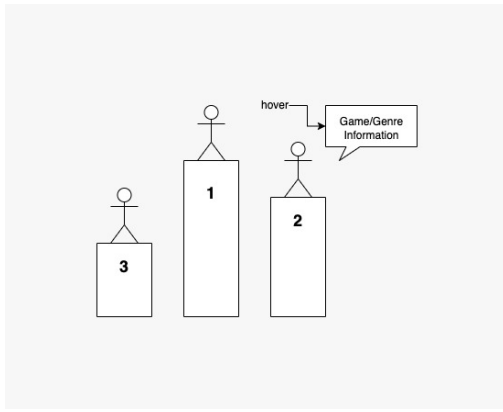


Figure 1: Sketch of top game ranking

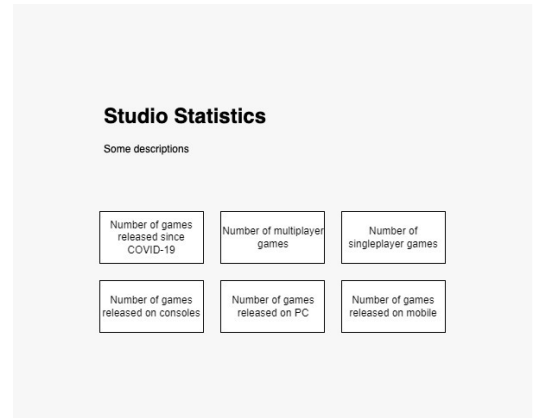


Figure 2: Sketch of studio-based statistics

Our first focus will be to interactively display the annual top 3 (may expand to top 5 later) games from 2019 to 2021 (Fig.1). When the audience hovers around the ranking bar, the figure will show the details of this game, such as game genre, game console, interactive type, and content brief. Then we will visualize the statistics of representative game studios (e.g. EA, Ubisoft, Nintendo) to dynamically show their overall game development statistics (Fig.2). We plan to implement them by Leaflet and D3 respectively.

2.2 Specific Visualizations

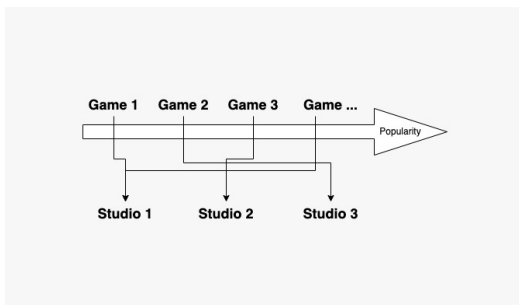


Figure 3: Sketch of popular games & studios

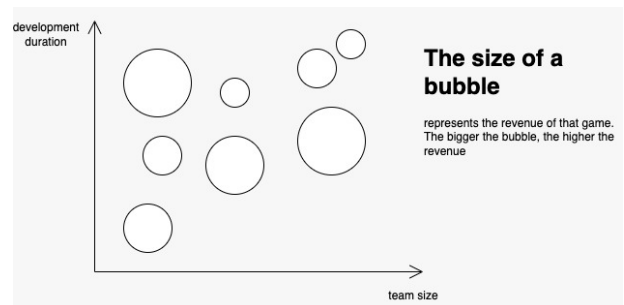


Figure 4: Sketch of multidimensional bubble graph

Grounded on games' popularity (daily online players as standard), we start from a studio perspective to arrange the top ranking games for their development teams (Fig.3). Audiences can also see the development timeline from texts when they click on the game and see all games belong to this studio when hovering around the studios. Then we will show an interactive word cloud (Fig.5) including keywords currently striking the game industry to let technical developers be aware of what contents are dominating players' interests. For developers who are aware of the development team size, we offer them

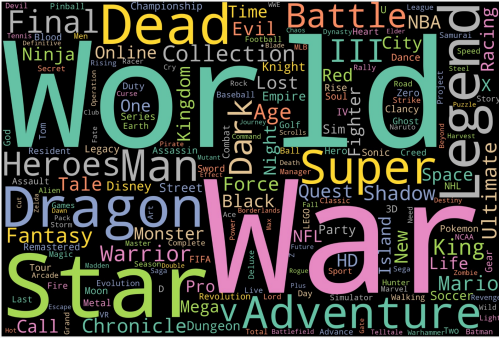


Figure 5: Sketch of keywords in game industry

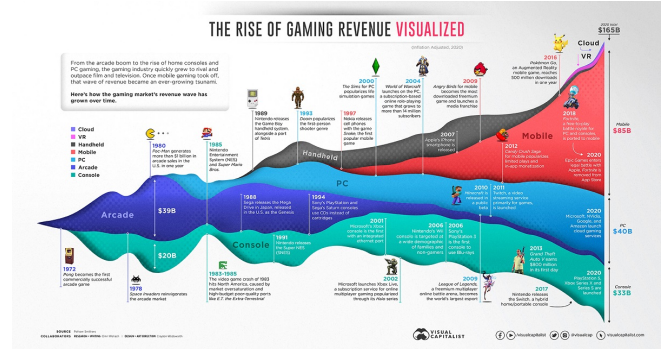


Figure 6: Sketch of game revenue change

a bubble plot (Fig.4) showing the market performance of popular games regarding their development duration and team size and representing their revenue by the area of bubbles. Both Fig. 3 and Fig. 4 are designed to be high-dimensional, which means we will fit conditions such as genres, game platforms, and whether belongs to an indie game to our visualization. To help audiences better understand the market change, we will later implement a timeline and event-based revenue map similar to Fig. 6 showing the fluctuation of revenue over time on different gaming devices. The events will be displayed once the audience moves the mouse to the icons. As developers care more about the local game market, we add an interactive worldwide map (Fig.7) to help audiences easily find out what games are the most popular over the period and what runner-ups consist of its biggest competitors.



Figure 7: Sketch of map for local popular game

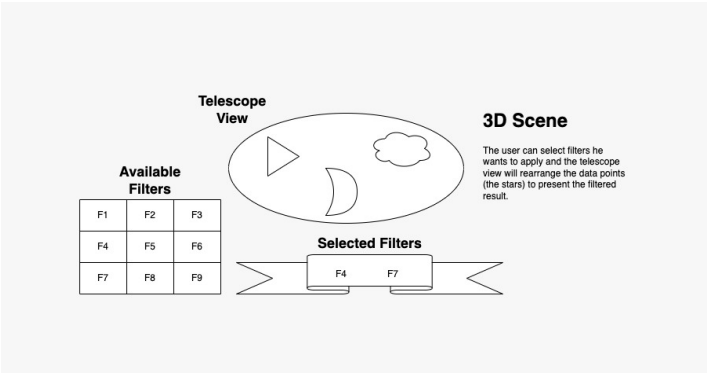


Figure 8: Sketch of 3D scene of game universe

3 Extra Contents

3.1 Filtering

Besides showing the data we initially provide in the graphs, audiences may intend to further explore the data from different scales. Such as in Fig. 1, 2, and 3, we will add filters of game genre, game console, studio, and timeline (or period) to expand users' choices. In Fig. 8, we will combine these filters into a composite palette for the user to explore.

3.2 3D Animated Scene

In Fig. 8, we will use Three.js to design a 3D scene of observing the planets (popular games) from different telescopes (filters of game genre, game console, market revenue, etc.) and users can freely combine these filters to observe the game universe through the telescope by mouse movements or keyboard instructions, which works as a game-form conclusion to our visualization project. In this scene, animations can be added in the process of changing filters (telescope in the scene) and in displaying the games (stars in the scene) to deliver our data more entertainingly and interactively. Scripted texts will also be displayed once the filters change, to briefly tell the game features given the new condition.