Data Driven Dota, Process Book for CS171, Spring 2014

Angela Fan, Robbie Gibson, Benjy Levin

April 2014

1 Overview and Motivation

Dota 2 is a multiplayer online battle arena game developed by Valve Corporation. Players play individual, independent matches that each take about an hour. Matches involve two teams of five players, one side called "Radiant" and the other side called "Dire." Players each control a character called a "hero," which starts off weak, and grows in strength through leveling up and gaining gold. Levels allow players to unlock or upgrade unique abilities, while gold allows players to buy items that improve their hero. Players play on a square map (please see figure), with the lower left hand corner and upper right hand corner occupying a stronghold for the Radiant and Dire, respectively. The game ends when the opposing team's stronghold, called an "Ancient," is destroyed. These Ancients are protected by defensive structures called "Towers," that the opposing team can destroy for a gold bonus. Please see Figure 1 for an image of the Dota2 map.

Figure 1: Dota Map. At each end of the map are the Radiant and Dire strongholds, called Ancients (marked by a rectangle). Defending the Ancients are a series of Towers, drawn as circles.

Dota2 players, however, do not have many forms of secondary analysis with which to critique their play, and help them improve it. The primary source of Dota2 statistics is the website Dotabuff, which provides some summary statistics for each player, information related to individual matches, and trend information for heroes and items over the entire playing population. However, we wanted to implement a site whose central theme is time—we wanted to allow users to explore their entire match history, glean information about their play style and habits, and see trends in their gameplay.

2 Related Work

As mentioned above the current Dota 2 statistics websites are aimed at a different niche. Dotabuff does an excellent job of letting you see any one match from your entire history and it even has a few summary statistics. It also examines

trends over all public games ever played. However, it doesn't have very many individual statistics or a way to examine your play on a higher level than just looking at all your matches.

The other major Dota 2 stats website is Datdota. However, this aggregates statistics from professional games only, and is thus useless to a player wanting to see where they personally could improve.

3 Main Questions

As a user...

1. How can I compare how I played in different games?
2. How can I see if I improved over time?
3. How can I see if my game strategy was effective?
4. How can I tell what heroes I play best?

4 Data

4.1 Gathering Data

The important data we are visualizing is the history of a given user's matches. To get this data, we used the Valve API to query for a user's data. Unfortunately, the API had some major problems, although we was able to work around most of them. The biggest problem for us, not using a real web server, was that the API didn't support cross-orign resource sharing. In a perfect world, we'd figure out how to run a web server and do the querying from a script somewhere, rather than client-side Javascript. Furthermore, as Valve requested we only make approximately one API call per second, and players can have in the thousands of matches, this further limited our ability to dynamically gather data for any user. Again, in a perfect world, we'd store our information in a database, only querying the API to update things, but that's also difficult, and not the best use of our time on this project. In the end, we used Python to statically gather the matches of 6 users (also, we ended up only being able to gather the most recent 500 matches for any user) and then save that for use.

We also had to gather a bunch of supplementary data for display purposes. The Valve API returns things like heroes played and items bought in the form of unique numerical ids, so we needed a way to convert those ids into actual names and information. For this, we built a JSON object that mapped ids to all the useful data about heroes, items, and abilities. This information was acquired using information already assembled by other users and a series of undocumented further Dota 2 API calls.

We also had to gather all the images required for the project. Each hero, item, and ability has a recognizable picture in game, and displaying those in our visualization would be a major benefit. We decoded the format Valve used to store the images on their servers, and, used a Python script to pull the images and store them in our own files.

4.2 Interpreting Valve API Data

We felt that we should create our own library since we will be making many function calls that require the Valve API data across all of our mini-visualizations. This was particularly pertinent for the game meta data such as the names of heroes and their main attributes, item information and all game related images. We store all of this information in corresponding JSON files, and we only require to load this data into our visualization once from our library. Any subsequent requests of this information is made through public methods that are available via our library.

We named our library D2 - as a fun spinoff of Dota 2 and D3.

5 Design Studio Feedback

We received feedback from **Matthew Warshauer**.

He made a number of good points, which we enumerate below. His comments are written in boldface, our responses are below each comment.

1. **Is it possible to learn about aggregate behavior of other Dota players as a way to evaluate your own play?** First, the API calls are difficult to make, because Valve limits you to 1 call per second, and we would need a larger set of users, since you can only get the latest five hundred games per user. Assuming we got

this data, evaluating other players in a naive aggregate way is not ideal. This is because when you play a Dota game, you only play against people of similar skill level (based on Valve matchmaking), so a pure aggregation would not be successful in allowing you to learn how to get better. A very good player probably doesn't want to see aggregates based on a pool of players that may be primarily worse than s/he is. Additionally, the Valve ranking is private and cannot be retrieved from the API data, so we have no way of filtering which users are better players than others. We agree this is a good idea, but does not seem possible at this point.

2. **Is it possible to do dynamic API calls?** Yes, but it would take a long time. As we addressed above, making one API call takes one second, and each user's games must be queried individually. Ideally we would build a database and call all users matches, and then load data from the database, but that is a little beyond the scope of the project.

3. **Hero chord diagram is not good. Should replace with matrix diagram.** We believe that this is an excellent suggestion, and will look at the Les Mis co-occurence matrix for ideas. This is definitely a good way to visualize this kind of data, but it needs to be filterable. An issue though is that this type of visualization would have a low data to ink ratio, because much of the matrix may be empty (i.e. if we have ten heroes, they may play often with some heroes, but much of the 10x10 matrix may be blank). Additionally, such a matrix has the same problems as a genetic heatmap- often not very easy to read and hard to tell what to sort by.

   A possible other avenue of exploration, inspired by Matthew, is adding more functionality to our chord graph. We should have a description of the chord graph and what it is showing, tooltip over each chord so you can see the number of games those two heroes have played, and be able to read off what elements those two chords link.

   That being said, we will certainly pursue both ideas and describe which one we ultimately go with as a design decision.

4. **Color coding of bars by cost isn't very effective, since you already have sorting by cost.** We agree. We think we should color-code by win/loss rate with that item.

5. **Perhaps bring up a list of games in which you bought this item. Can you see if a certain item is correlated to winrate?** That seems possible, and would be very interesting on items that are not commonly purchased- for example, if I as a player run the risk of buying this rare costly item, do I win with it? Is there a very expensive item that even if I purchase, acts like wasted gold?

   However, this is difficult for very commonly purchased items. We will start implementing this idea and see if there is a number of games played cutoff for displaying the games. Additionally, we are not yet sure where to put this games list, but will work on that when we decide layout.

6. **Graph descriptions are necessary.** Yes. We have not yet made these, but the final product will have these.

7. **Everything that is filterable should be un-filterable. User should know where s/he can filter.** We agree. We will have a description in each graph and at the top of the page about the filters. We will try and make sure the filters go clearly with the correct graph, and that it is clear to the user what filters what. We may have a "help" button to help users navigate this system.

8. **Green means two things- win, and agility. That is confusing. I recommend red/black for loss/win, respectively.** We agree, to an extent. However, the shades of green are clearly different- agility is closer to a highlighter shade, while the winrate green is a dark green. Red/black for loss/win we do not think is as intuitive as red/green. We will look into arranging graphs with similar colors next to each other, so color is more of a visual link rather than a differentiator between graphs.

6 Exploratory Data Analysis

We performed a number of exploratory data analyses with our dataset. Instead of starting development and switching around datasets, we thought it would be easier if we stuck with a consistent dataset. We chose to use Robbie's. This is because Robbie has played a significant number of games with the hero Windranger, and has a consistent item build. This means that we can glance at some of the graphs and clearly see if we are on the right track. If we see Windranger leading with way more games than every other hero, and see his common build of Phase Boots + Force Staff + Mekansm, we are likely on the correct track.

First, we plotted a histogram of all heroes and the number of games in which they have been played. From this histogram, we decided that a bar chart would not be a good way to show heroes played. So many heroes were 0 or 1

that it was difficult to see them on the bar chart. Thus, we explored alternative designs. Eventually, we came up with the hero sunburst design, which was useful because users could zoom in, so small values are less of a concern.

We also plotted a histogram of all items purchased. This eventually became the items bar chart. We decided to keep it a bar chart, instead of transforming it into a sunburst plot, because unlike the heroes, there are no real parent-child connections in the item graph. Multiple heroes can purchase the same item, whereas it is clearly defined which heroes have the primary attribute "agility."

Lastly, we plotted a scatterplot of hero GPM and hero XPM. However, the scatterplot didn't really mean anything- it looked like a Uniform distribution. It was impossible to see any trends or compare performance from game to game. Coloring by hero didn't make much sense, as there are 108 different heroes. Eventually, we kept the scatterplot design, but this initial exploratory analysis turned into a scatterplot of this game's gpm/xpm vs. average gpm/xpm on a hero. This allows the user to compare their play to their average play, which is far more useful and is capable of actual comparison.

After these three beginning visualizations, we started making our graphs, and went through many design iterations (please see Design Evolution section). We chose to take more time to go through design iterations rather than make exploratory plots, because our project is exploratory in nature. We wanted it to be an organic exploration of our Dota2 data, as currently there is no way to visualize or analyze play.

7 Design Evolution

Please see figures.

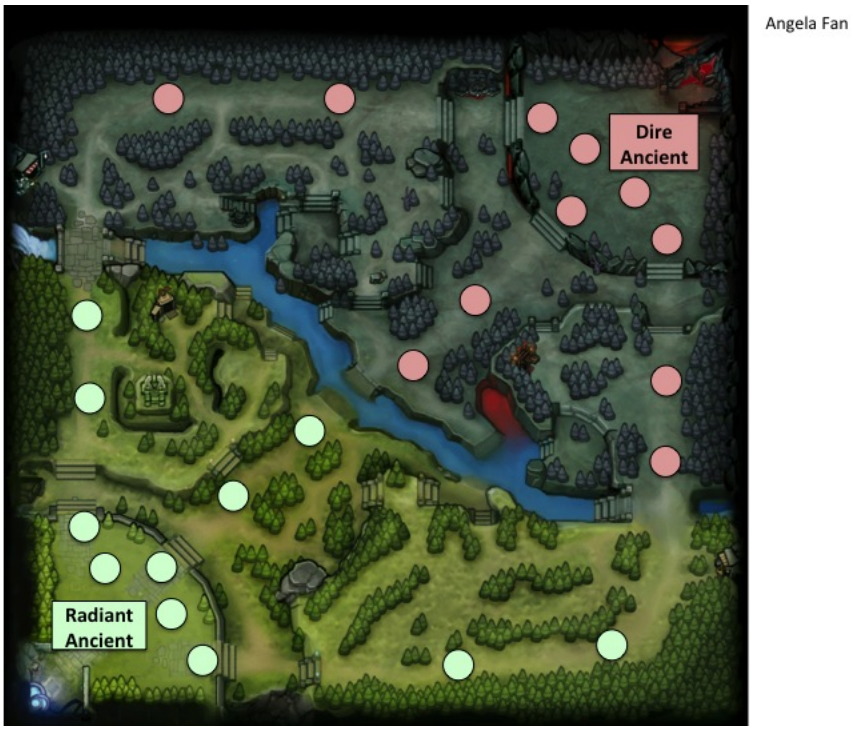


Figure 2: Design Process

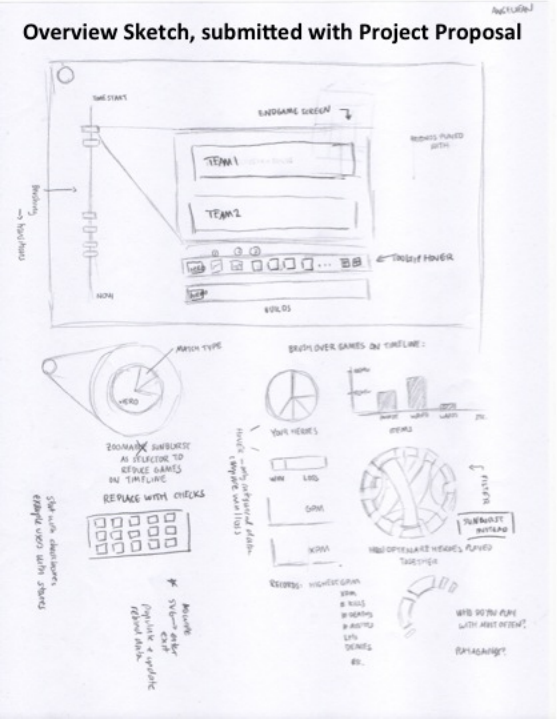


Figure 3: Design Process

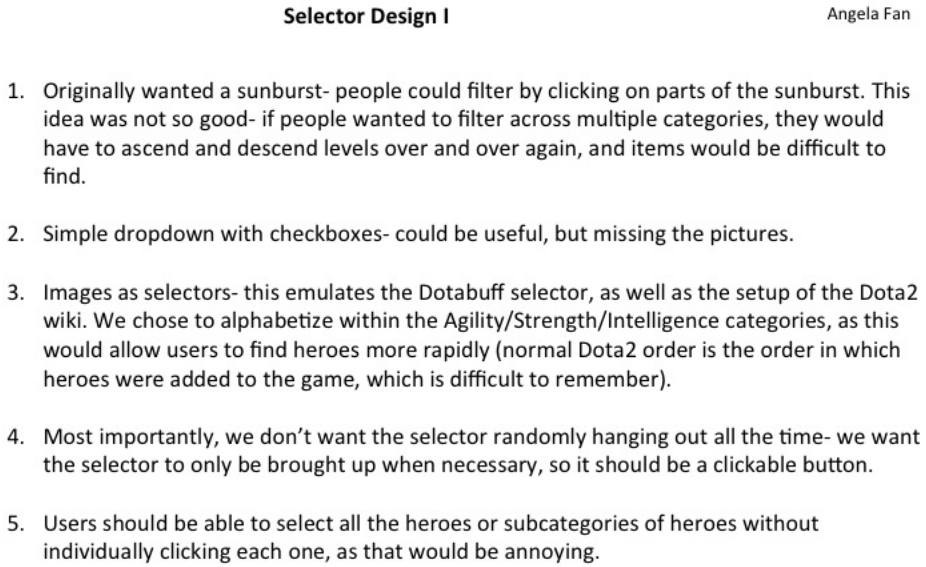


Figure 4: Design Process

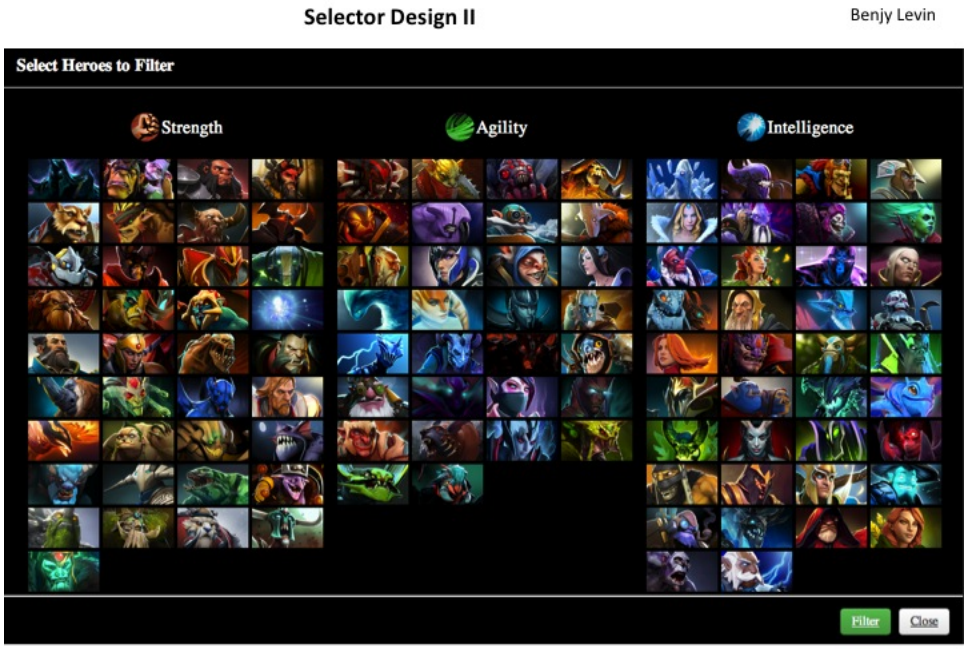


Figure 5: Design Process

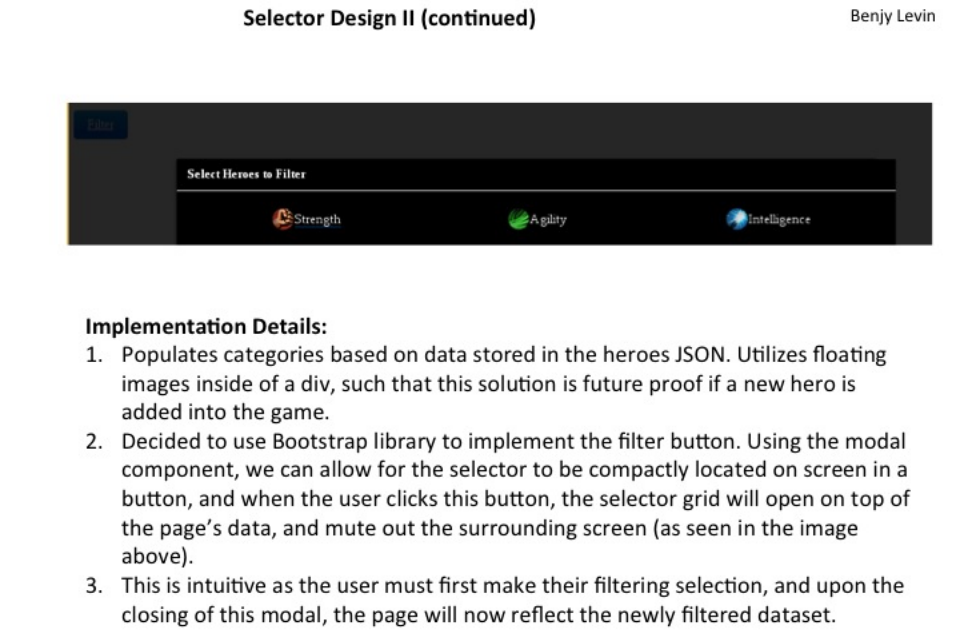


Figure 6: Design Process

Selector Work III Robbie Gibson

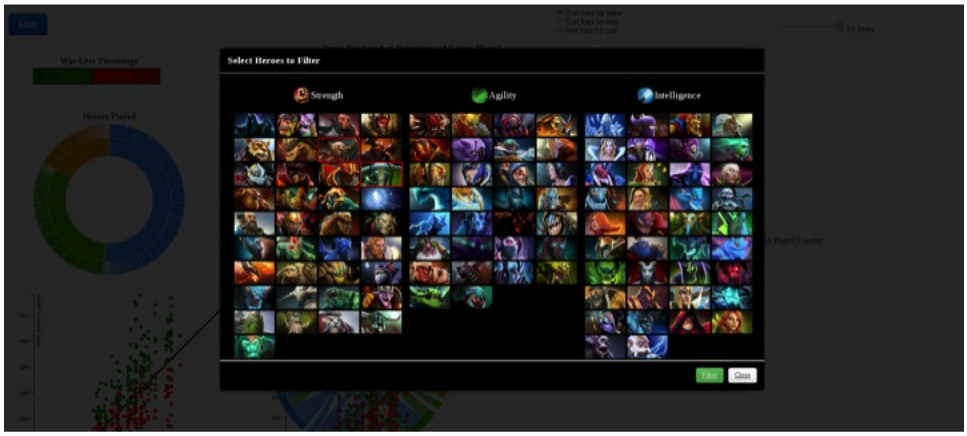


Figure 7: Design Process

Selector Work IV Benjy Levin

For ease of selecting each category, we made it such that if the user clicks on the heading: Intelligence, Agility or Strength, it will either select all the heroes in that category, or deselect all if they are all already selected.

We also solved the bug such that if the user makes a selection and then clicks CLOSE (and not filter), when the user reopened the modal, it displayed the unfiltered selection. Now, every time the modal is opened, the user is shown the heroes selected from the most recent filtering of the data set by heroes. This was a consistency error.

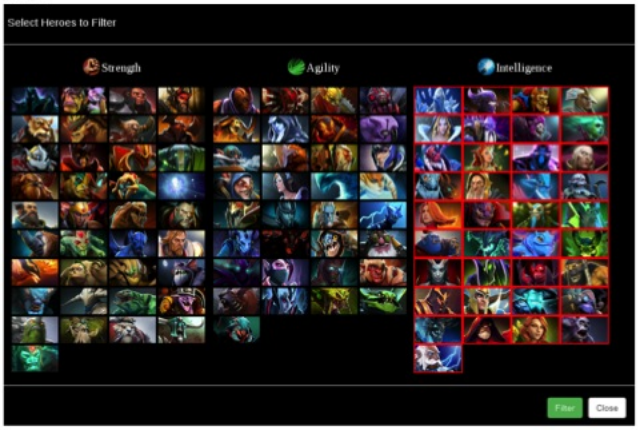


Figure 8: Design Process

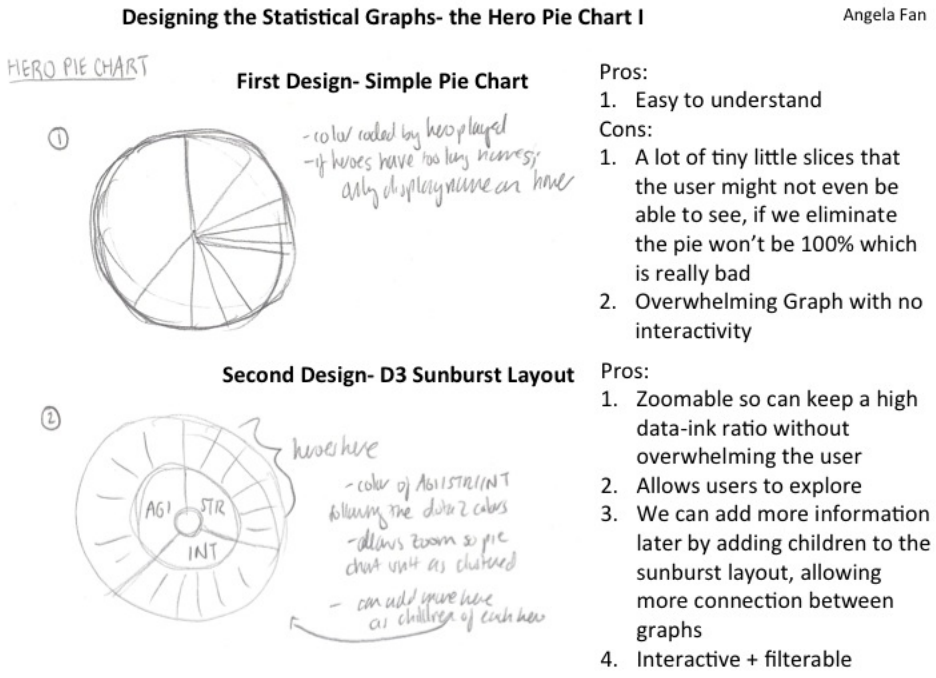


Figure 9: Design Process

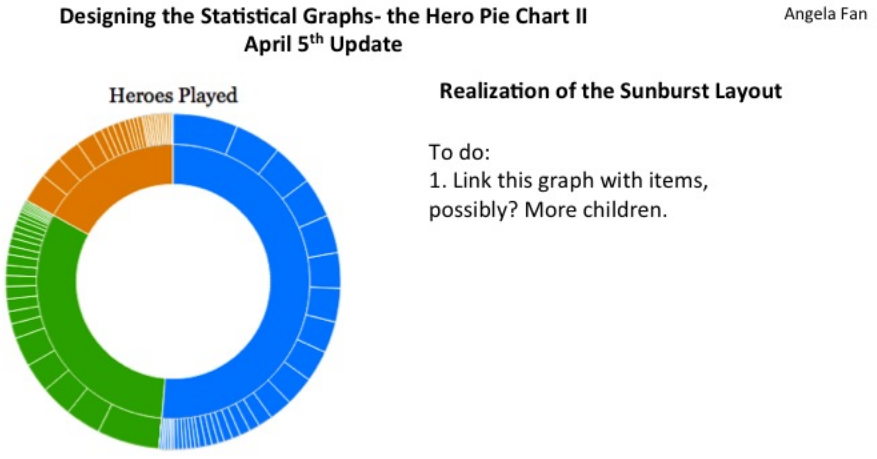


Figure 10: Design Process

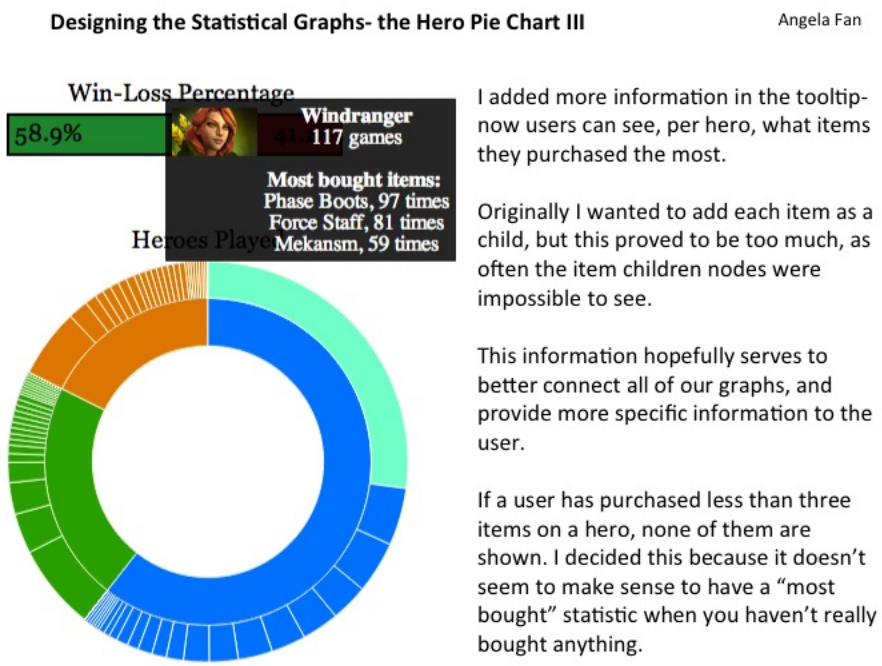


Figure 11: Design Process

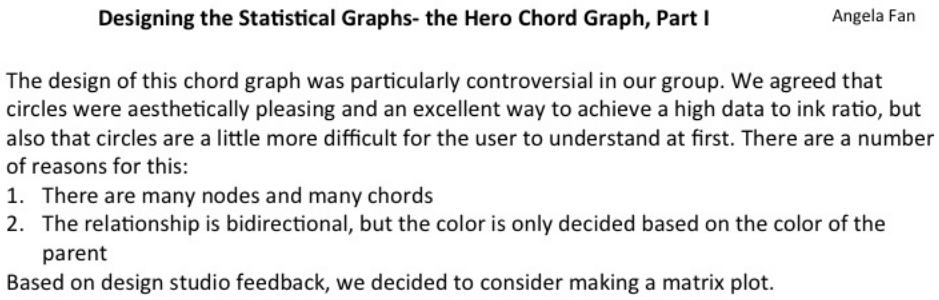


Figure 12: Design Process

Designing the Statistical Graphs- the Hero Chord Graph, Part I Angela Fan

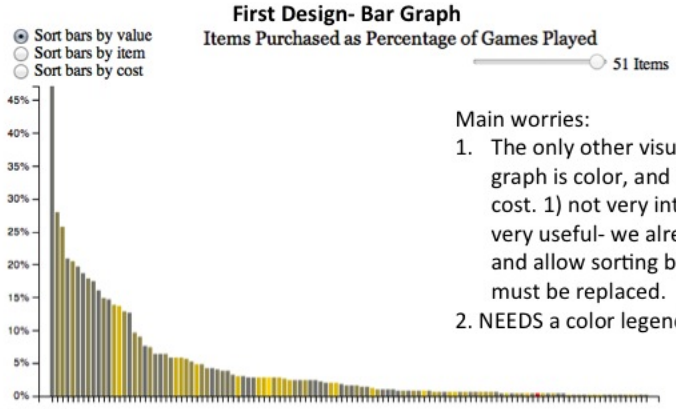
In the end, I chose to stay with the chord diagram. I decided this because I think that visualization is partially about pushing the boundaries. People didn't know they hated floppy discs until Steve Jobs took them away. Sometimes, users just need to learn to understand types of visualizations that are coming in the future.

However, in order to remedy some of the drawbacks of the chord diagram, I implemented a tooltip that allows users to tooltip over each chord. This lists which heroes the chords are going to, so users can just read off the hero names. Additionally, it lists the value that the chord represents, giving users a numerical understanding of the area the chord represents. I think this is better than a matrix color scale, since people inherently have better area perception than color scale perception.

Figure 13: Design Process

Designing the Statistical Graphs- the Items Bar Graph, Part I Angela Fan

First Design- Bar Graph



Main worries:

1. The only other visual element of this graph is color, and here it is colored by cost. 1) not very intuitive and 2) not very useful- we already report the cost and allow sorting by cost. This coloring must be replaced.
2. NEEDS a color legend.

Features:

1. Slider bar filter so users can find top items
2. Sorting bars by value: this sorts by number of times the item was purchased- basically the main point of the graph
3. Sorting bars by item: this sorts alphabetically, so users can find the item they want- so if a user is more interested in one item, they can find that item
4. Sorting bars by cost: users may find this relevant, especially if they want to see how often they purchase a big expensive item
5. Tooltip for item, item picture, cost, and number of times purchased

Figure 14: Design Process

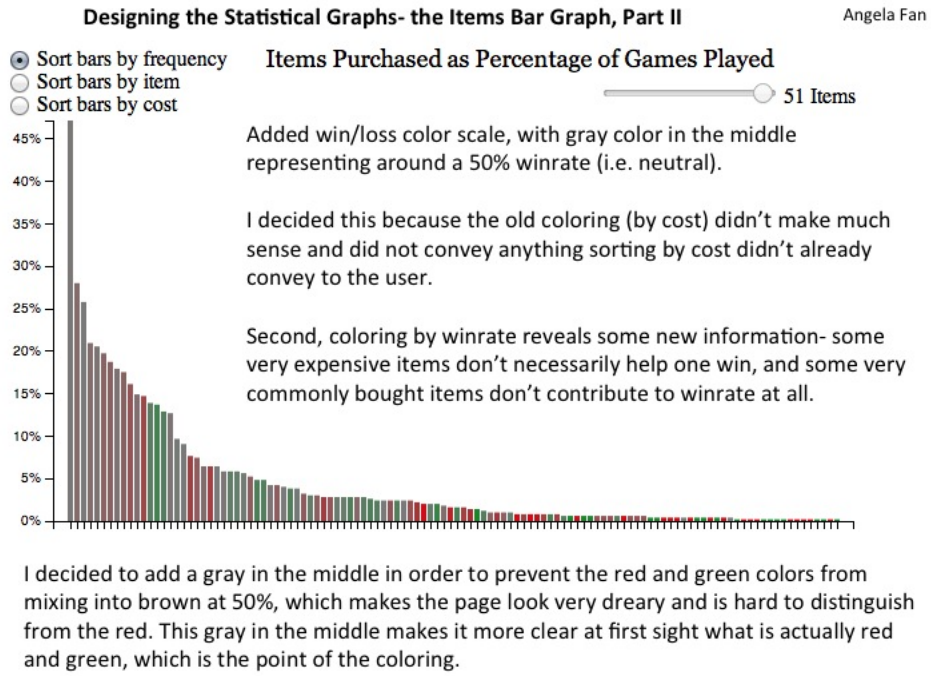


Figure 15: Design Process

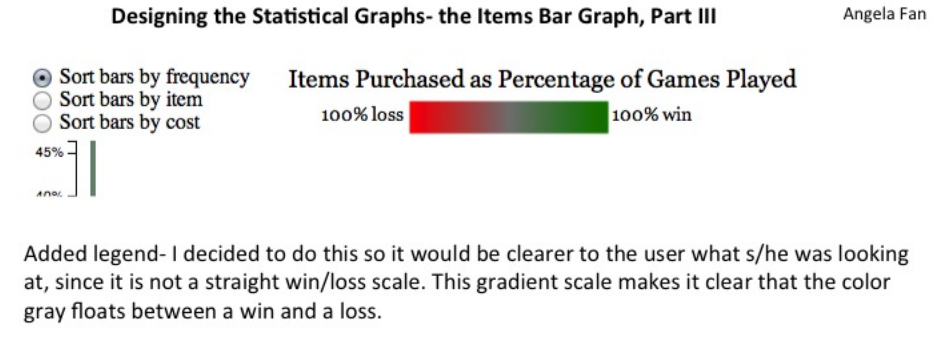
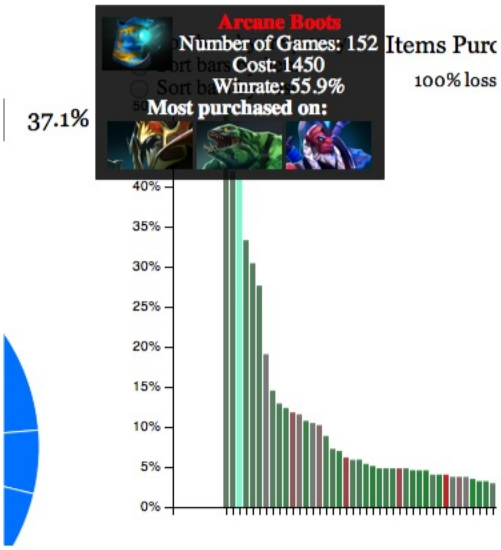


Figure 16: Design Process

Designing the Statistical Graphs- the Items Bar Graph, Part IV Angela Fan



I added in the tooltip, the images of the heroes you have most purchased a given item on.

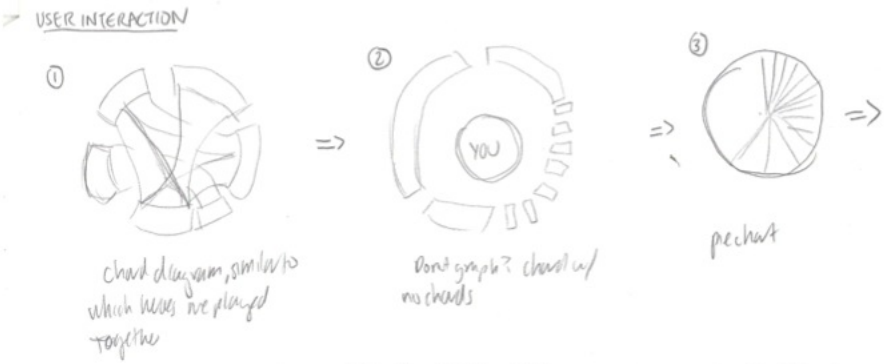
This was to fit in with our theme of connected graphs.

Additionally, this information cannot really be found elsewhere, but is useful and interesting for the user to know.

I chose to add this information in the tooltip, because the graph already shows a lot of items. I didn't want to bring up a table of users or anything like that, because the page is quite crowded and I am not sure where to put it so it fits nicely, given all of the sorting mechanisms that move the bar height around.

Figure 17: Design Process

Designing the Statistical Graphs- the User Interaction Graph, Part I Angela Fan



Chord Diagram

Pros:

1. Looks pretty cool, makes sense intuitively

Cons:

1. We don't have the data for that many users, so we can't represent interactions between other users. There aren't that many chords!

No chords Chord Diagram

Pros:

1. Makes sense. Chord diagram without chords.

Cons:

1. Chord diagram without chords == bar chart?

Pie Chart

Pros:

1. Most sense! Intuitive!

Cons:

1. Users who I've only played one game with aren't very interesting, but if we filter, the pie chart no longer represents 100% anymore

Figure 18: Design Process

Designing the Statistical Graphs- the User Interaction Graph, Part II Angela Fan

Bubble Chart

Pros:

1. More interesting visually than pie chart
2. Can filter it so it doesn't necessarily need to represent 100%, but can display only users that you've played more than one or two games with

Cons:

1. Not as intuitively understandable as a pie chart
2. The central bubble that represents "you" is very very large compared to the other bubbles

Figure 19: Design Process

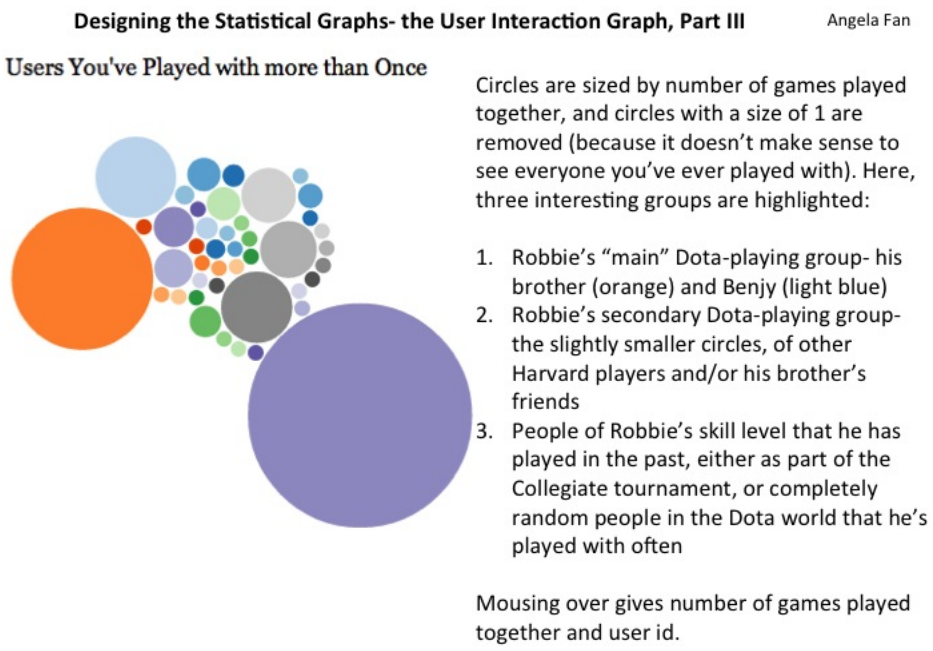


Figure 20: Design Process

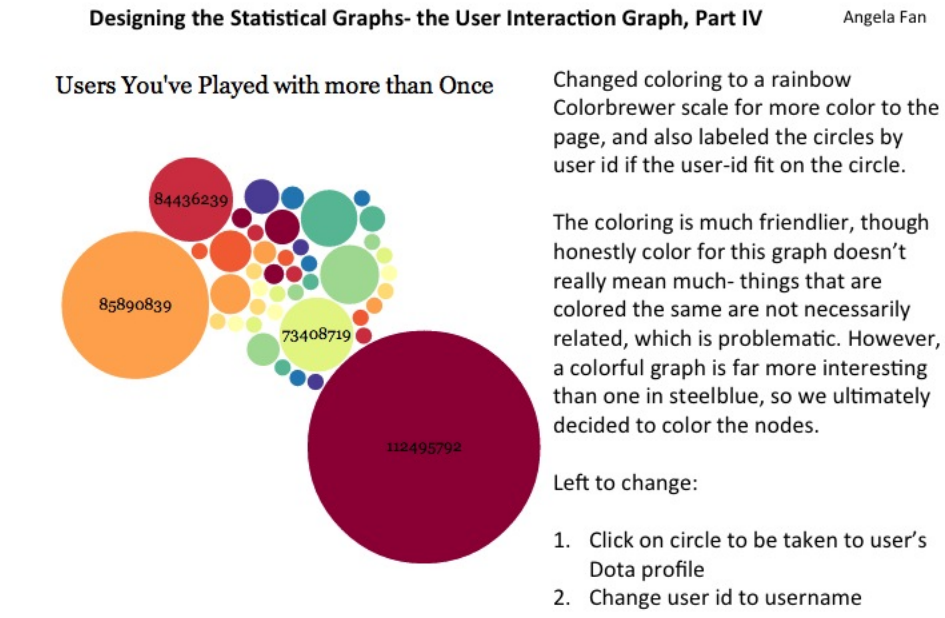


Figure 21: Design Process

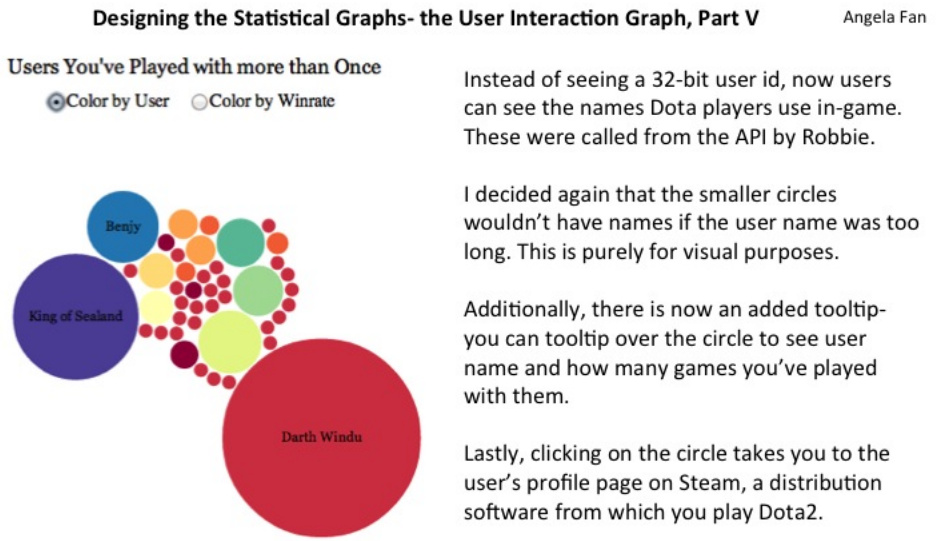


Figure 22: Design Process

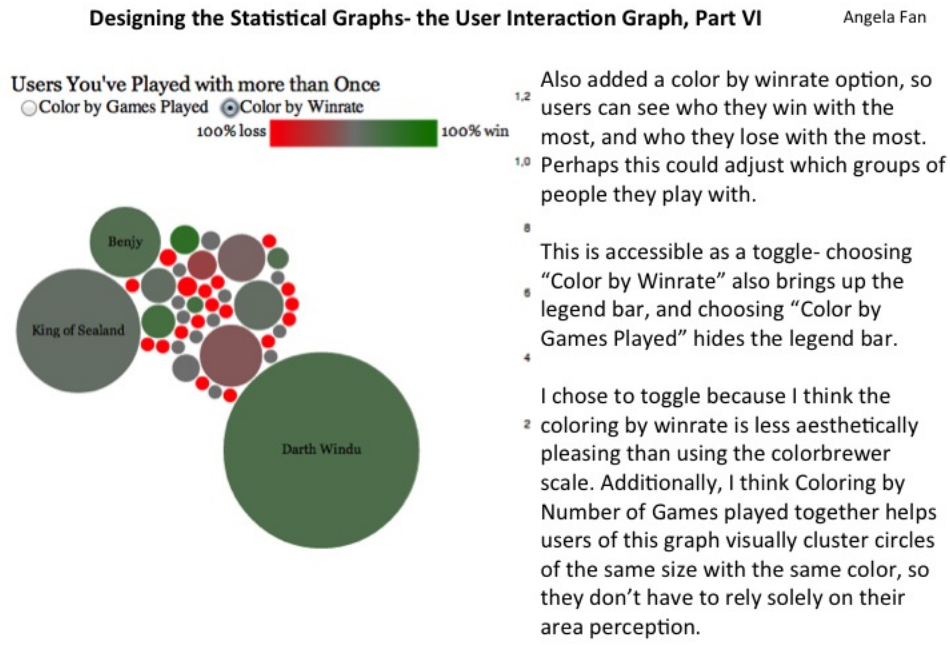


Figure 23: Design Process

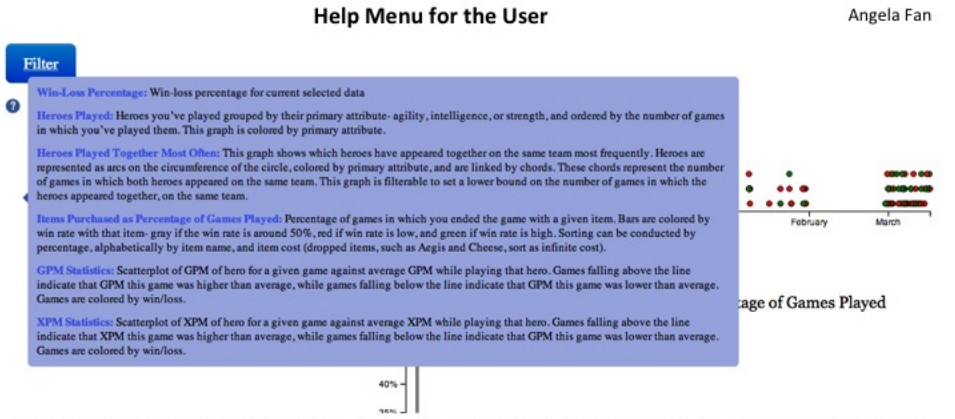


Figure 24: Design Process

Designing the Statistical Graphs- the GPM/XPM Graphs I Angela Fan

Line Graph

Time on X-axis, GPM on Y-axis
Pros:
1. Basic

Cons:
1. Not good measure- no normalization
2. No good comparative ability
3. Terrible way to detect trends

Line Graph II

Time on X-axis, GPM on Y-axis, but only the user's average is plotted. There is a dropdown menu at the top of the graph, so user can pick to change which hero is plotted

Pros:
1. Can now try and detect hero-based trends

Cons:
1. Not good measure- still no normalization
2. No good comparative ability between heroes
3. Introduces yet another selector that overcomplicates the whole graph system

Figure 25: Design Process

Designing the Statistical Graphs- the GPM/XPM Graphs II Angela Fan

Scatterplot I

Average GPM for that hero on the x-axis, GPM for that game on that hero on the x-axis

Pros:
1. Comparative! Not between heroes but at least between games

Cons:
1. Not intuitive, since below the line is a "good game"
2. Still no normalization, and not comparative between heroes

Scatterplot II

Average GPM for that hero on the y-axis, GPM for that game on that hero on the x-axis

Pros:
1. Comparative! Not between heroes but at least between games
2. Intuitive, since being above the line is great game and being below the line is a bad game

Cons:
1. Still no normalization, and not comparative between heroes

Figure 26: Design Process

Designing the Statistical Graphs- the GPM/XPM Graphs III, April 5 Update

Angela Fan

Normalization Needs

1. Over game length- in a longer game, heroes on average get much stronger than in a shorter game
2. Data bias- sometimes the game ends accidentally before it actually begins- these GPM/XPM values are skewing the mean
3. Based on hero role- if you play different roles, sometimes you have different GPM/XPM. If a hero can fit multiple roles (carry/support, like Alchemist), then the mean isn't a good representation of an average game of either role. For example, if you play carry Alch 50% of the time and support Alch 50% of the time, the mean GPM floats between the two, but that doesn't mean that your support Alch games are "worse" games on average
4. This graph doesn't help evaluate how heroes do with farm, or how fast a hero can farm, which are both interesting questions.

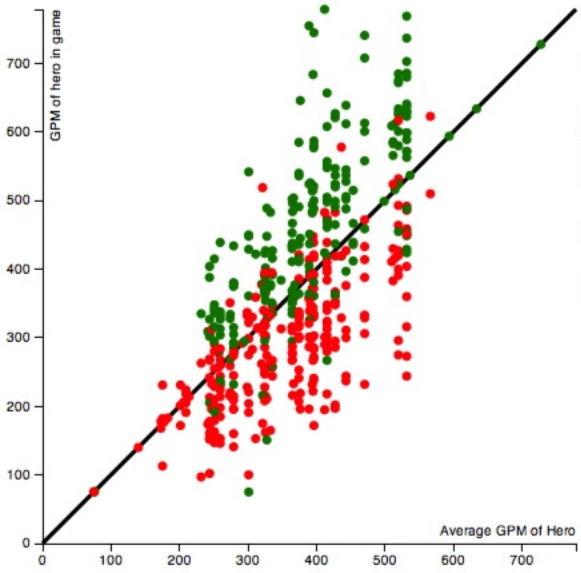
Future Development

1. A good question this graph could answer is "what happened in that game???" if the game is "worse" than average. We need to have an on click event, that brings up the end-game screen of the respective game.

Figure 27: Design Process

Designing the Statistical Graphs- the GPM/XPM Graphs IV, April 5 Update

Angela Fan



Looks pretty good.

Heroes are lined up in lines based on the x-axis, which are heroes that are played often.

Has a tooltip mouseover.

Need to have on-click event.

Figure 28: Design Process

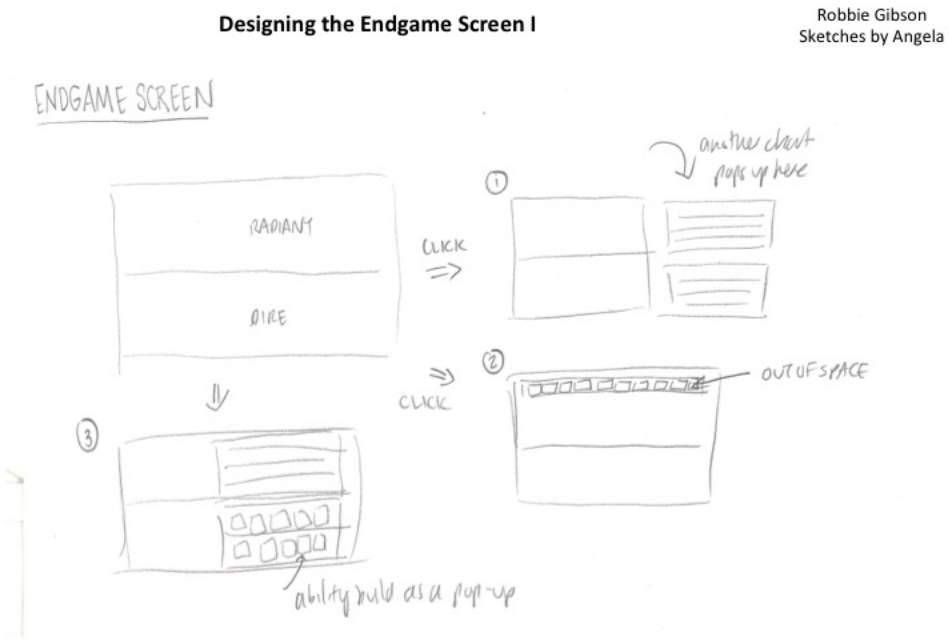


Figure 29: Design Process

Figure 30: Design Process

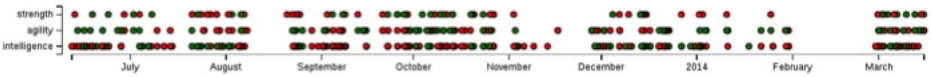
Designing the Timeline Benjy Levin

Design Process and Evolution:

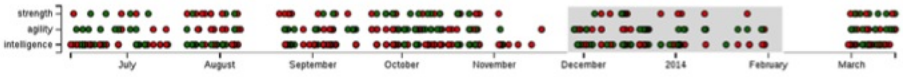
1. We originally planned to have a timeline down the left side of the page, displaying each user's games. Has a brush feature so users can zoom in to a certain time period.
2. Decided to implement this using the image of the hero that the player played in a given game. However, upon implementing this solution, we realized that a given user could typically have 500+ games and thus 500+ icons, which were far to many to scroll through in a div with scrollbars.
3. In addition to this issue, we realized that time is more intuitively perceived linearly from left to right, not up to down or down to up, and when a brushable timeline is implemented, it is far better design practice to have the entire time period in view, rather than having to scroll.
4. Decided to pursue a different implementation.

Figure 31: Design Process

Designing the Timeline Benjy Levin



1. Overview of all games played over time, divided into three rows based on the main attribute (strength, agility, intelligence) of the hero you played in that game. Colour encoded as red for win or green for loss for that game.



2. Timeline is brushable (see highlighted section above), and upon brush, the graph will zoom into that area and redraw the x axis with more detail based on this new selection – see

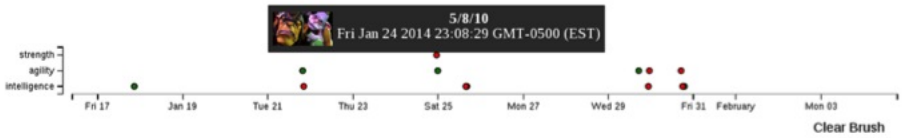
3. Upon zooming in, a "Clear Brush" button will appear in the bottom right, that will logically clear the zoomed in brush when clicked.

Figure 32: Design Process

Designing the Timeline (continued)

Benjy Levin

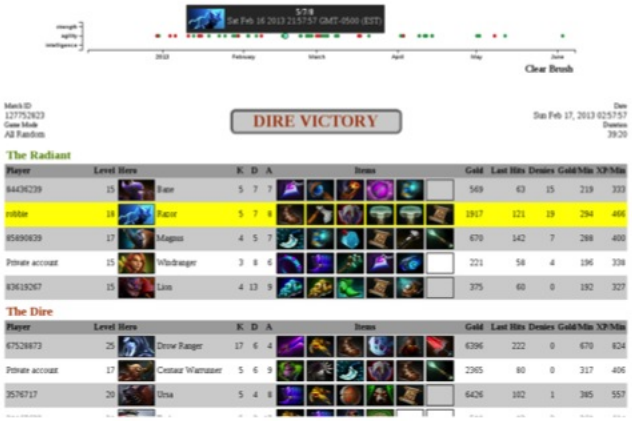
4. Upon mouseover, a tooltip is displayed with some further details about that game, such as the image of the hero you played, your number of kills/deaths/assists in the game, and an exact game time.



Future Development

On click of each node, display the summary statistics for that particular game throughout our visualization.

Figure 33: Design Process



We decided that it would be nice to highlight the match dots when you hover over them. This also highlights the match dot in the GPM/XPM graphs, and stays when you click on one to show the match screen.

Figure 34: Design Process

Dynamically Changing CSS – colorblind.css

We felt that since we are using 2 very color-blind unfriendly colors (red and green) which traditionally represent “win” and “loss” in Dota 2 and universally, it would be appropriate to give the user the option of dynamically changing these colors to something more color-blind friendly if required.

By dynamically adding (or removing) colorblind.css to the DOM, which then over-shadows our main.css file, and then redrawing the graphs, we were able to achieve the desired toggle on/toggle off effect.

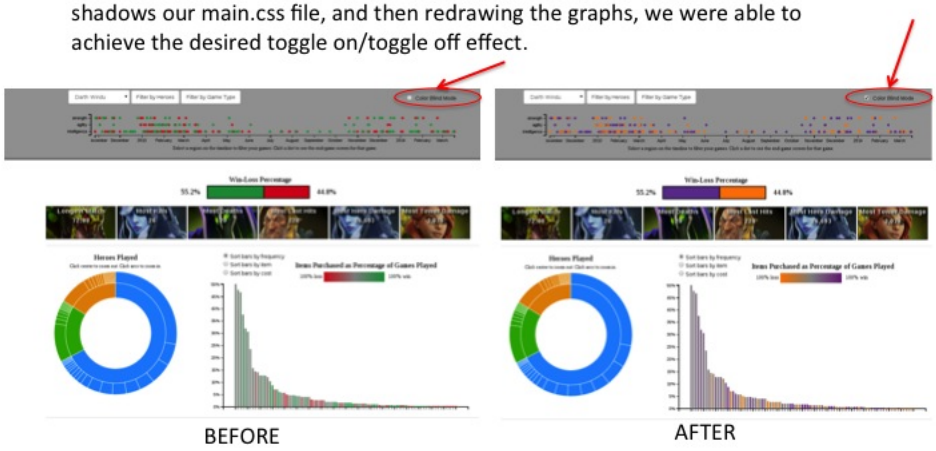
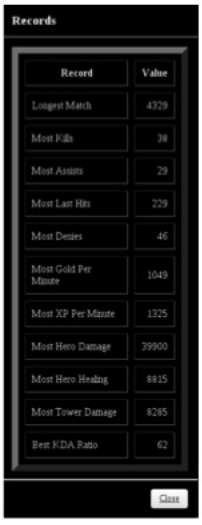


Figure 35: Design Process

Designing the Records

Benjy Levin



Motivation

Players love to view their historically best scores in various categories. This not only serves as a nostalgic/boastful moment to reminisce on great games past, but could assist in analyzing why said game was so exceptional and what could one learn from it.

Design Process and Evolution:

1. The records started off in the basic form as a table inside a modal that displayed various record-worthy categories and the value associated with those categories.
2. This was accomplished by building a record JSON that is updated by traversing through the matches array of the current filtered data selection, and then updating each field accordingly.
3. We felt that although the table provides the information desired, it is not aesthetically pleasing. So we opted to go for something a bit more visually stimulating and informative.

Figure 36: Design Process

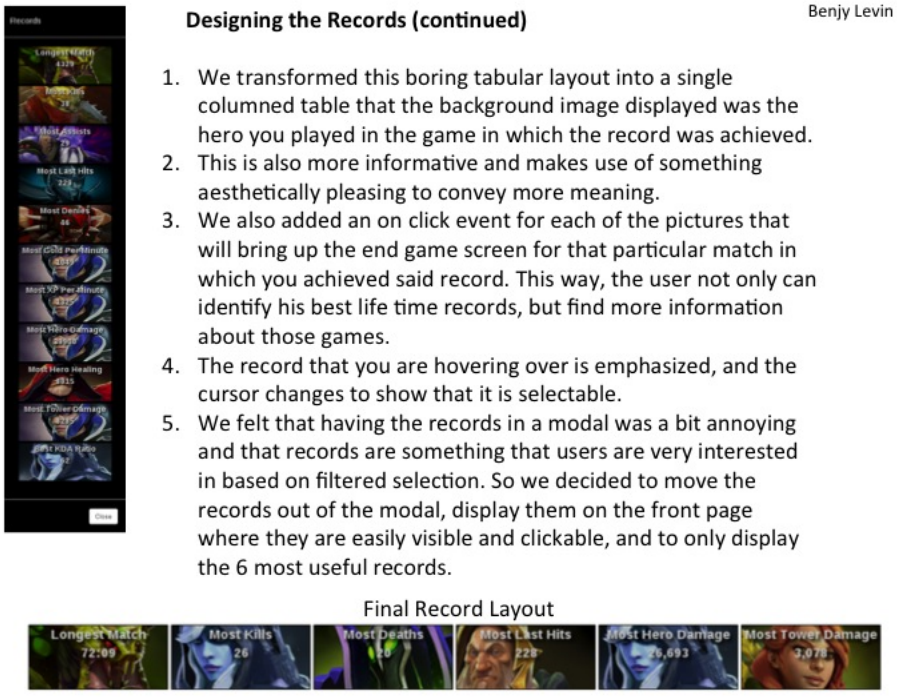
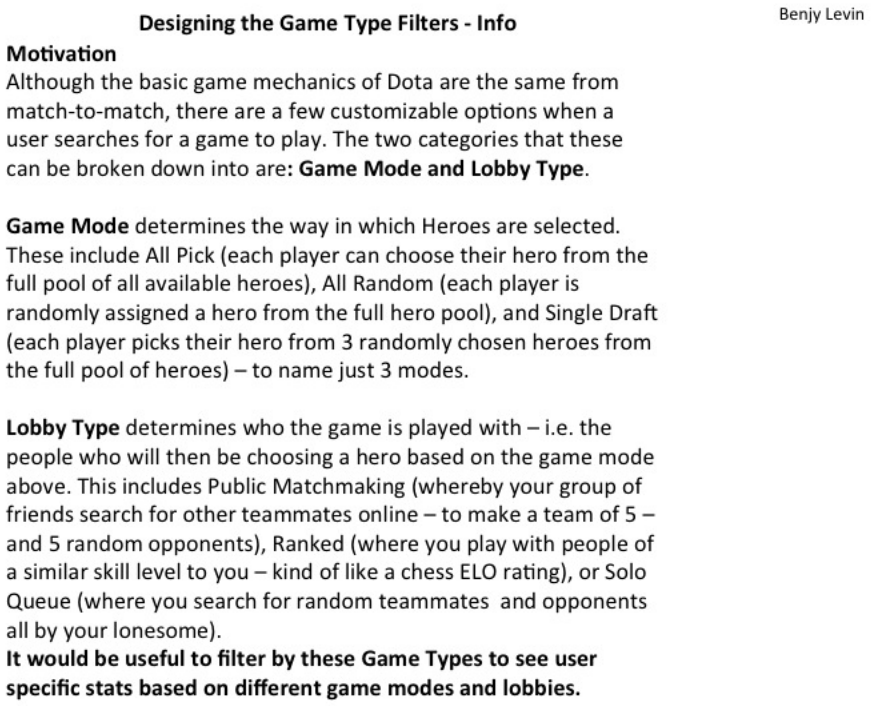


Figure 37: Design Process



Designing the Game Type Filters - Info

Benjy Levin

Motivation

Although the basic game mechanics of Dota are the same from match-to-match, there are a few customizable options when a user searches for a game to play. The two categories that these can be broken down into are**: Game Mode and Lobby Type**.

Game Mode determines the way in which Heroes are selected. These include All Pick (each player can choose their hero from the full pool of all available heroes), All Random (each player is randomly assigned a hero from the full hero pool), and Single Draft (each player picks their hero from 3 randomly chosen heroes from the full pool of heroes) – to name just 3 modes.

Lobby Type determines who the game is played with – i.e. the people who will then be choosing a hero based on the game mode above. This includes Public Matchmaking (whereby your group of friends search for other teammates online – to make a team of 5 – and 5 random opponents), Ranked (where you play with people of a similar skill level to you – kind of like a chess ELO rating), or Solo Queue (where you search for random teammates and opponents all by your lonesome).

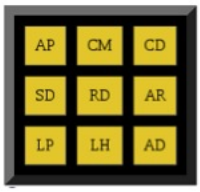
It would be useful to filter by these Game Types to see user specific stats based on different game modes and lobbies.

Figure 38: Design Process

8 Implementation

Please see figures.

Designing the Game Type Filters Benjy Levin



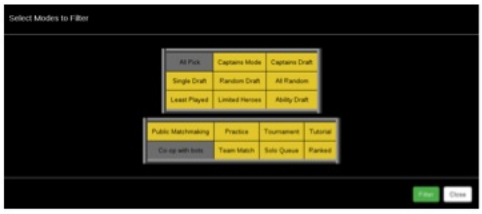
Design Process and Evolution:

1. We originally just placed a square table to filter by Game Mode (AP = All Pick, etc.) on the main graph page. On click, it would update the filtered data based on the user's toggled selection.
2. Although this was functional, it was not ideal as it lacked the functionality of filtering by Lobby Type, and was not user friendly for those who do not understand the abbreviation.

3. We expanded the names in full, and added the functionality to filter by lobby types.
4. However, having this on the main page was very cluttered and aesthetically unpleasing, so we moved it into a modal (like the hero filters) and styled it further.

Figure 39: Design Process

Designing the Game Type Filters (continued) Benjy Levin



5. This was aesthetically much more pleasing. We also changed it such that the filtering only occurred on click of the filter button, which prevented some of the graphs displaying incorrect info if the user clicked on different modes in quick succession. The webpage saves the state of the selection, so even if the user makes a selection and then CLOSES the modal without filtering, upon reopening, the most recently applied filter selection is still displayed.

6. However, we felt that the filter layout, and styling did not fit in well with the rest of our webpage, so we restyled the table layout to be more in line with the webpage's layout and the hero filter modal. We also added titles to the 2 different filtering categories. We also decided to default to a few modes unselected, since they are not of immediate interest, and may skew the "typical" user information shown on load (i.e. Practice matches or Tutorial games).

Figure 40: Design Process

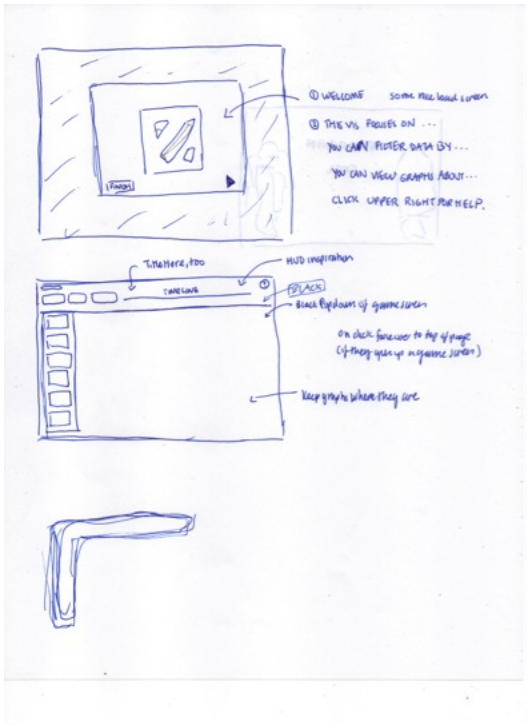


Figure 41: Design Process

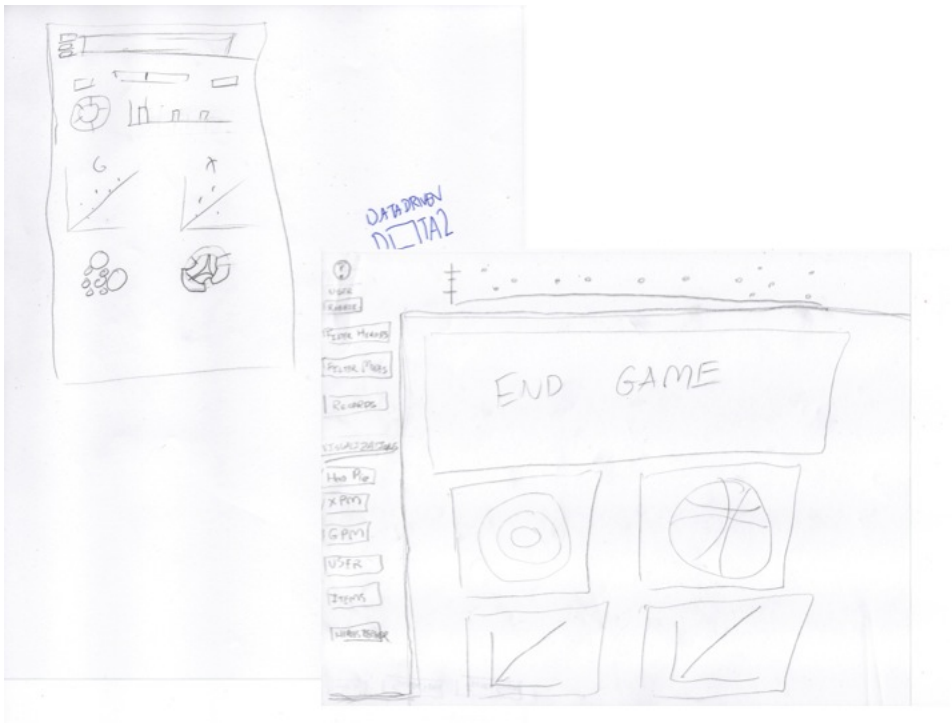


Figure 42: Design Process

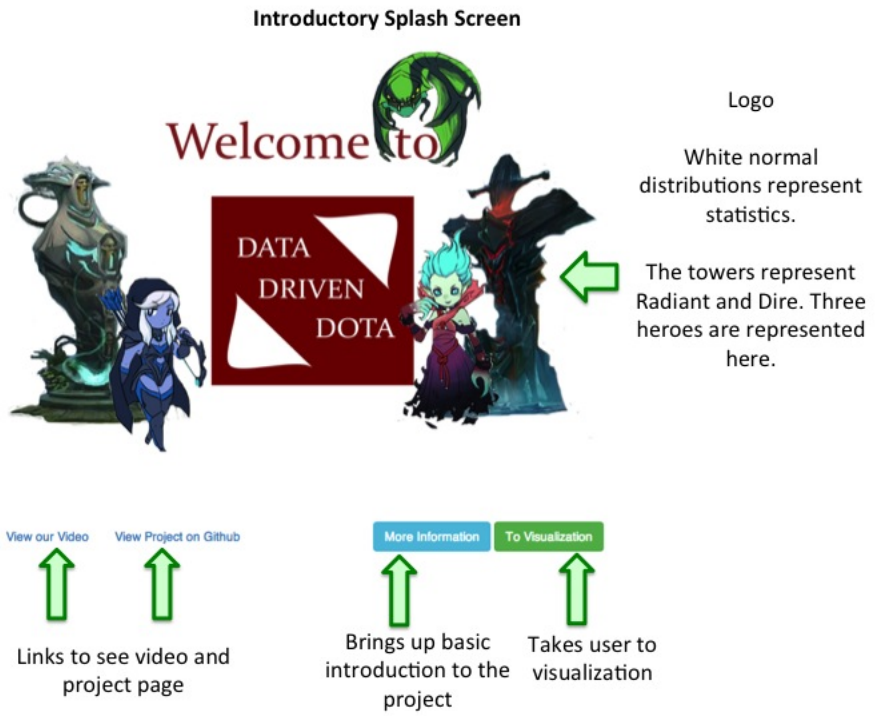


Figure 43: Final Feature Description

9 Evaluation

We learned a lot about our Dota 2 play using our visualization. Here are some examples:

1. Expensive, but not good? Some items offered in the game that are quite expensive are not necessarily very good- in fact, you often have a low winrate when you use them (Eye of Skadi). **Items Bar Chart**

2. Cheap and cost-effective, but not good? The most cost effective item in the game, Iron Branch, has a low winrate. Why? It is usually only purchased early game, so if you end the game with it, you probably lost quite early and didn't get a chance to replace it with a better, more expensive item. Same thing occurs with an item like Boots of Speed- key in the early game, but always replaced as soon as possible.

3. Soooo overpurchased! Some items have a reputation of being good, and seem like cool items, but in actuality are not very great. An example is Dagon- there are five levels, and it is an item that deals a huge amount of damage in one shot. Cool, right? Actually the winrate with a Dagon is quite low.

   Hero Sunburst

4. Wow, very uneven! Often a mark of a good player is balance- being able to play a diverse array of Dota heroes well. From our data, we see that the pro players in our dataset have this quality, whereas we (mere mortals) don't. Despite thinking that we play a diverse array of heroes, we, in actuality, have pretty high hero bias. This could be holding back our ability to get better.

5. Similar item builds On the same hero, people often build the same items. This is easy to do, because you can follow your in-game guide, or follow your usual playstyle. However, building the same items could indicate that the player is not adapting to the situation in game. This could be an area where players look to improve and diversify.

   Hero Chord Diagram

6. Hero Coordination Our Hero Chord Diagram recovers many of the common, popular hero pairings, such as Phantom Lancer-Keeper of the Light, and Crystal Maiden-Juggernaut.

   User Bubble Diagram

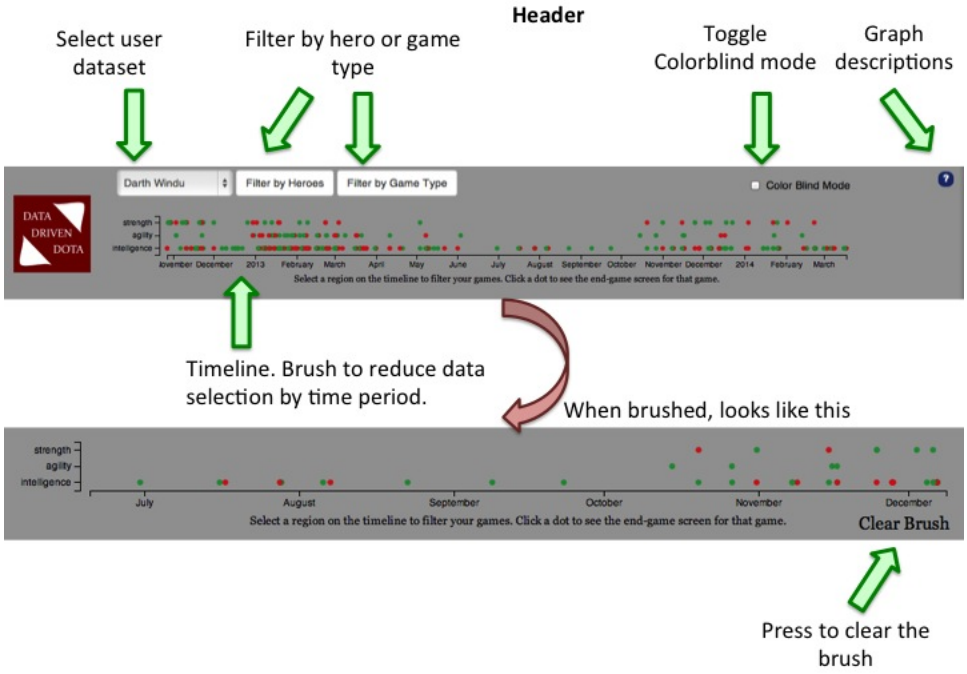


Figure 44: Final Feature Description

7. Playing with Friends Our diagram reveals that playing with friends is sometimes detrimental- for example, if your friends are not as good as you are, it is often difficult to win in a game with them.

8. Valve Matchmaking Valve forms games with people of similar skill-level, so the games will be fun and hopefully balanced. This means that all players do not have an equal chance of being matches with all other players. Thus, it is possible that a player is often matched with the same random player from somewhere else in the world. We were not sure if this possibility was high, given Dota's large playerbase, but from this graph can see that there are a number of people that hvae been matched randomly more than once.

XPM/GPM Graphs

9. Comparing to the Average One surprising thing these graphs indicated was that a player could be player worse than average and still win, and play better than average and still lose. Originally when we color coded the games by winrate, we thought it would obviously be do well = win, do badly = loss, but clearly it is about who else is on your team and how well they are doing. For example, if you are not playing very well this game, but the other four members of your team are having an exceptionally good game, it is possible for the team to still win.

10. What does Average mean? It's hard to conceptualize what an average game is like. However, our graph allows users to see game endscreens of an "average" game, so they can see how they are doing. This also allows users to see where they are getting better- when their average GPM and XPM increase, it is clearly a trend of improvement.

11. How do GPM and XPM relate? When users highlight a game in the GPM graph, it also highlights a game on the XPM graph. This led to an interesting observation for us- it is possible to have high GPM but relatively low XPM, and vice versa. For example, if you are gaining a lot of experience but not killing other heroes, you have a high XPM but low GPM. If you are managing to hunt down heroes and kill them, that is a lot of gold but not necessarily a lot of experience, since you may be moving alone through a jungle.

We were able to answer our original questions quite well:

1. How can I compare how I played in different games? For example, I know I've played ten different games with the hero Lina. How can I figure out which game I played the best in? Currently, the only thing you can do is compare the different end game screens, but this involves pulling up one, closing it, pulling up the next one, etc. Our visualization aims to allow users to compare whatever games they want next to each other. For example, users can easily compare who they played with, their XPM/GPM, and what items they purchased.

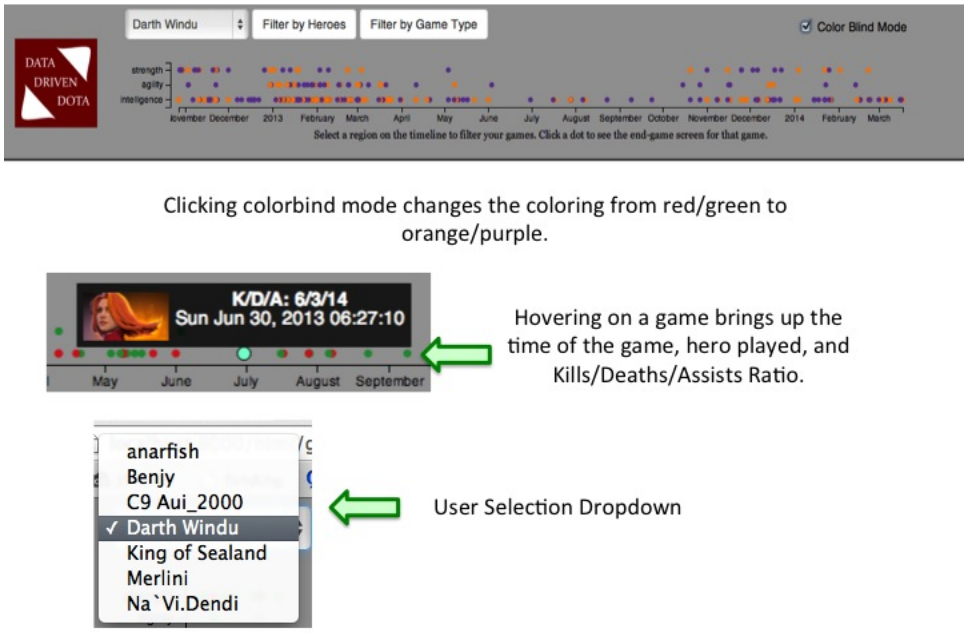


Figure 45: Final Feature Description

2. How can I see if I improved over time? This is not possible to do, currently. Our visualization is very timeline focused- user games are plotted on a timeline, and users can brush over the timeline to narrow the visualization down. For example, if a user wanted to compare his play over two years, that is possible with our visualization.

3. How can I see if my game strategy was effective? Our visualization allows users to compare different strategies. For example, users can filter the dataset down so it only shows games with Lina, then compare their winrate with each item. Users can look at their winrate playing with their friends. Users can compare their chosen ability orders between different games by clicking on winning vs. losing GPM/XPM graphs.

4. How can I tell what heroes I play best? Users can evaluate "playing best" by looking at the GPM/XPM graphs, allowing them to see when they are playing better than average. Users can see which heroes they have the highest winrate on.

Our visualization works quite well. There are some small issues:
Given time, we would make some changes.

1. The one statistic we did not choose to focus on was KDA- Kills/Deaths/Assists Ratio. We chose this originally because in many ways, KDA is highly overrated. This is because different heroes play different roles- some heroes are good at setting up a kill but not necessarily killing itself, some heroes are better at sacrificing themselves so other members on their team can survive, etc. so KDA is not a great measure of how well you are playing this game. However, KDA is over-cared about in the Dota 2 community, so if we had time, we would make a KDA streamgraph.

2. Seeing only winning and only losing games could be potentially useful. Given more time, we would have implemented a "click to see only winning games" and "click to see only losing games" functionality. This would allow users to contrast their wins and losses more effectively.

Figure 46: Final Feature Description

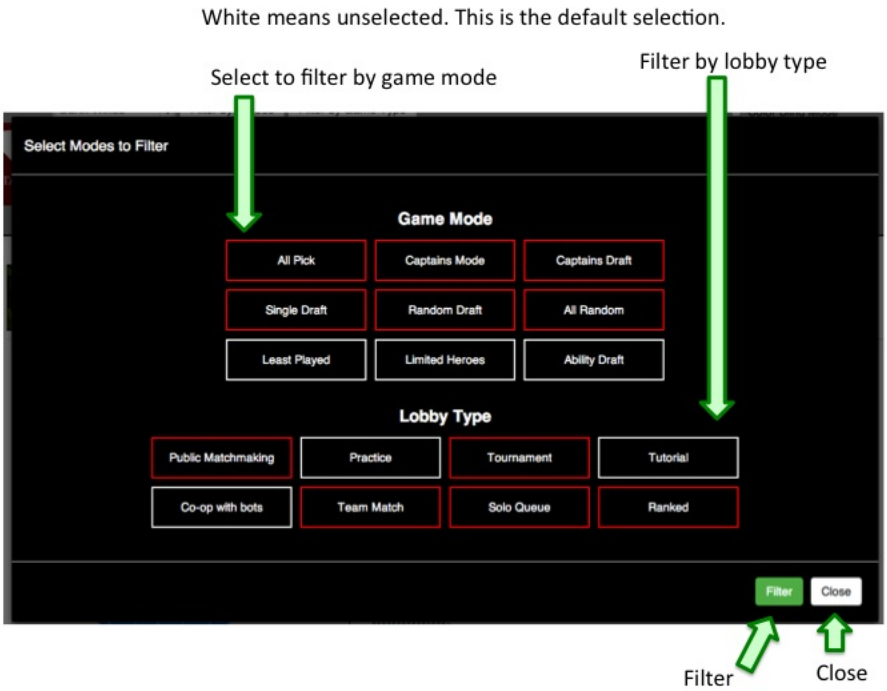


Figure 47: Design Process

Question mark on click brings up graph descriptions for user.

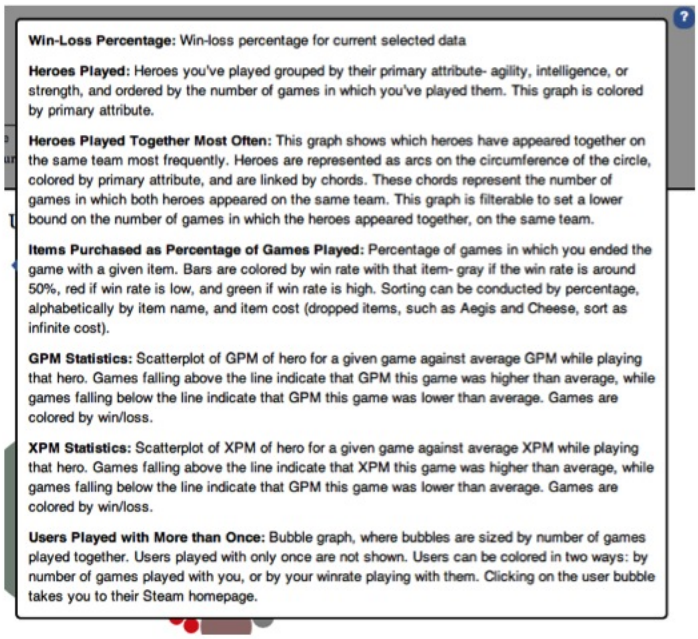
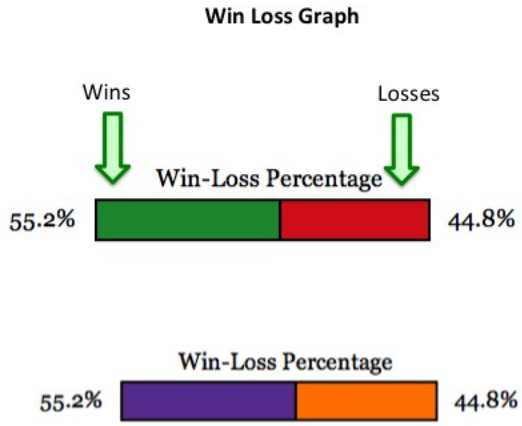


Figure 48: Final Feature Description



Clicking colorbind mode changes the coloring from red/green to orange/purple.

Figure 49: Final Feature Description

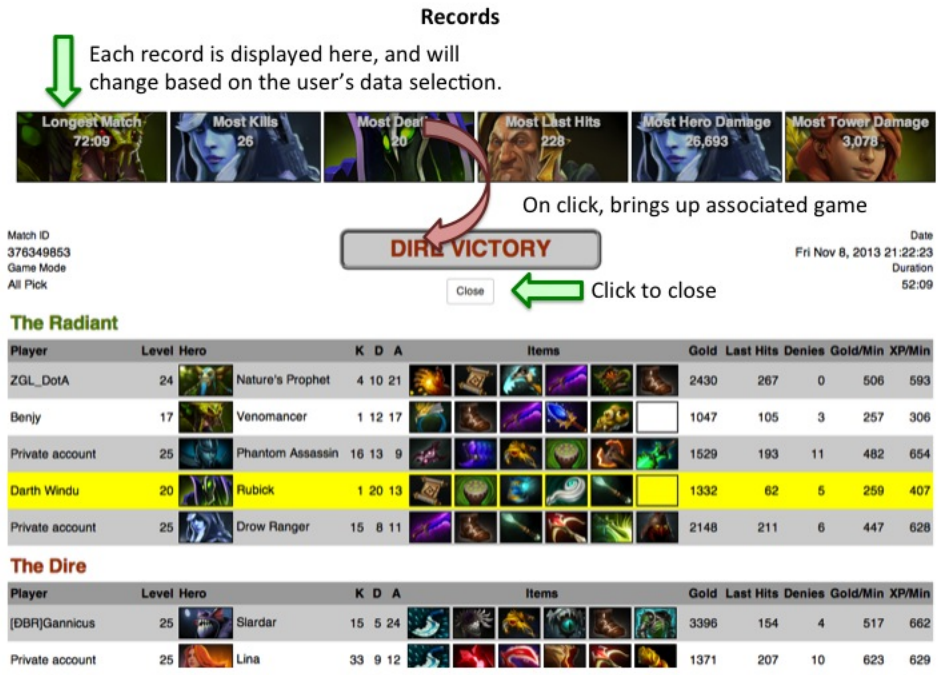


Figure 50: Final Feature Description

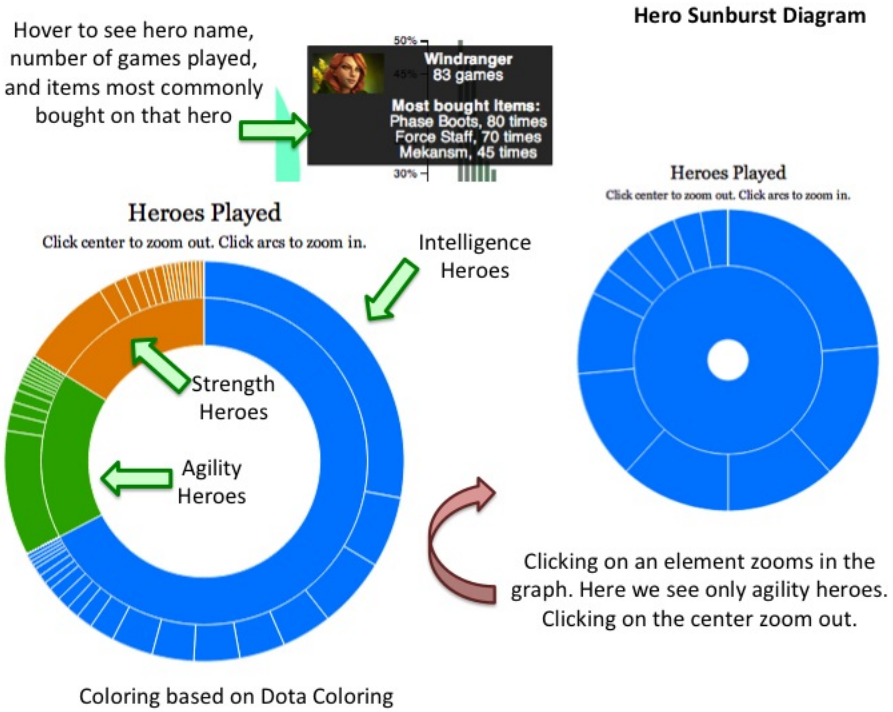


Figure 51: Design Process

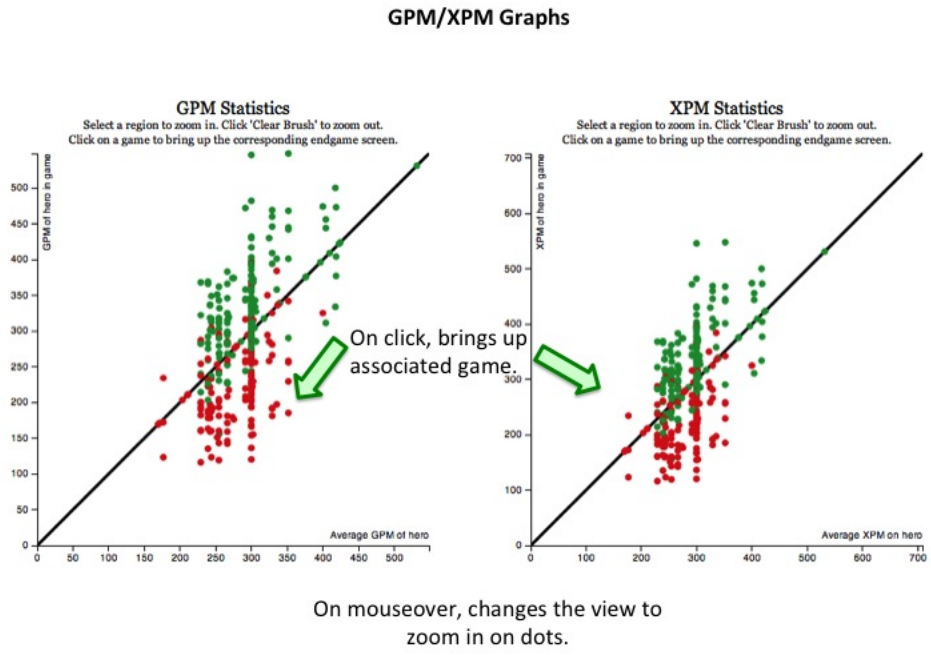


Figure 52: Final Feature Description

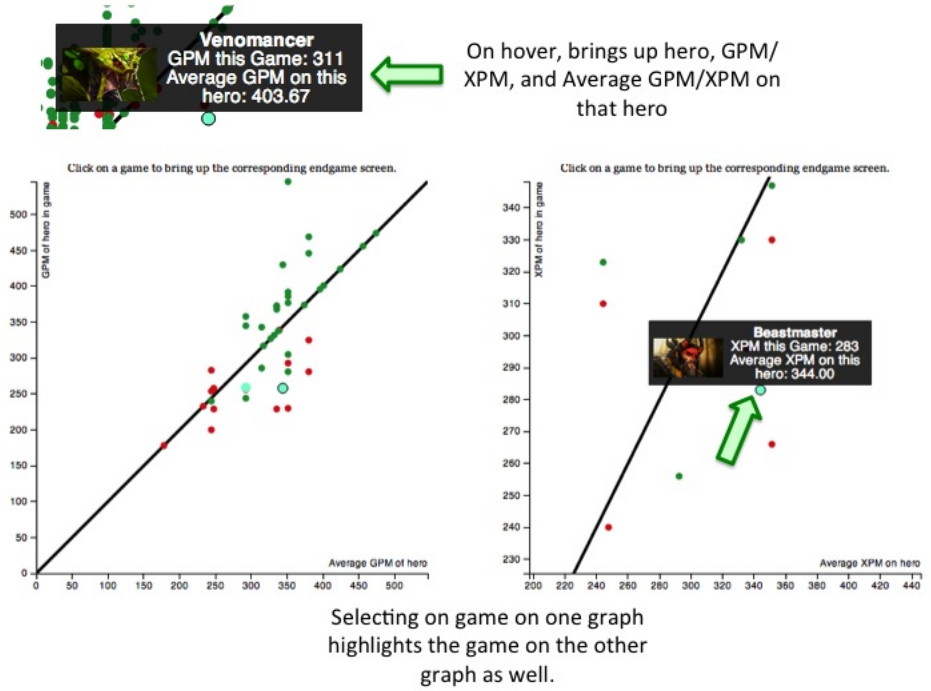


Figure 53: Final Feature Description

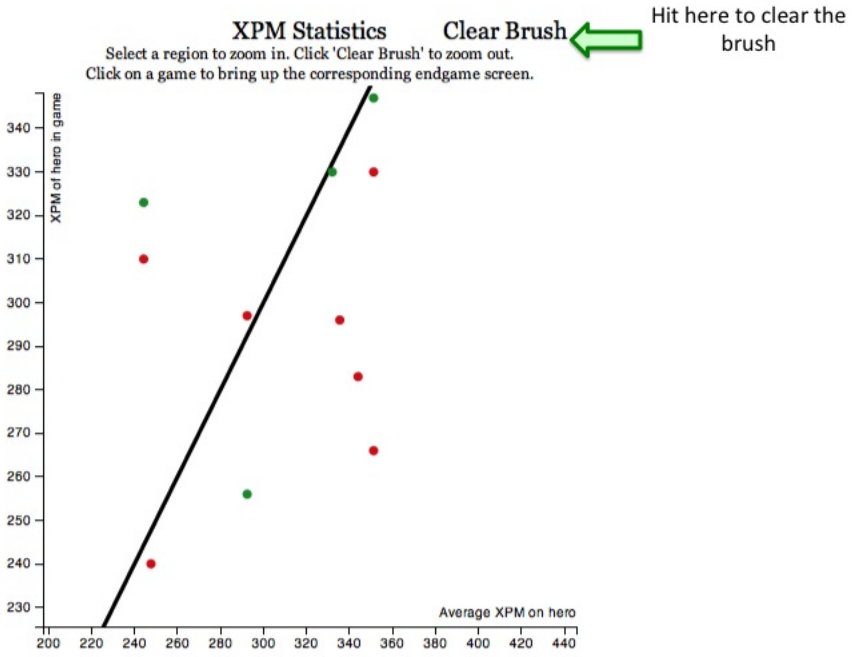


Figure 54: Final Feature Description

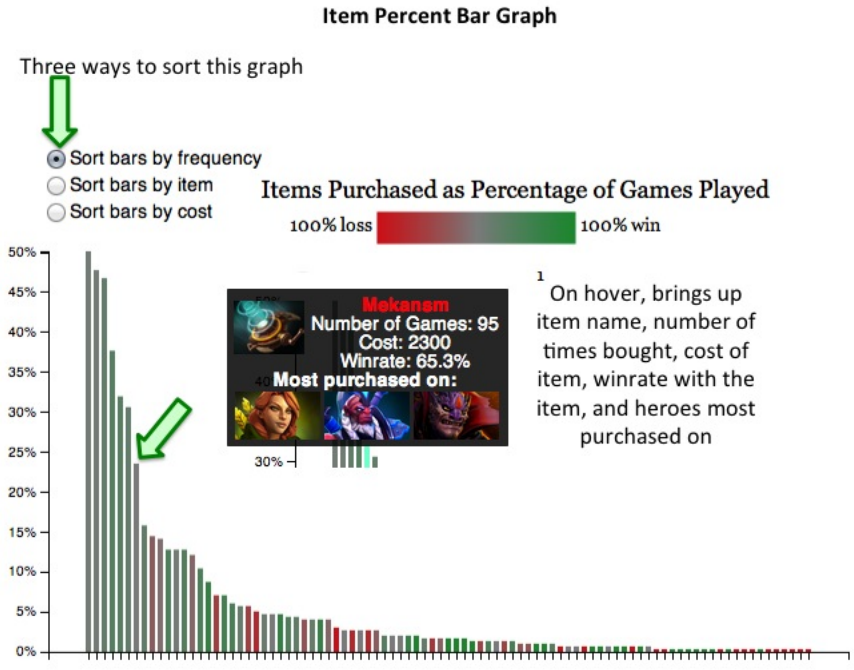


Figure 55: Final Feature Description

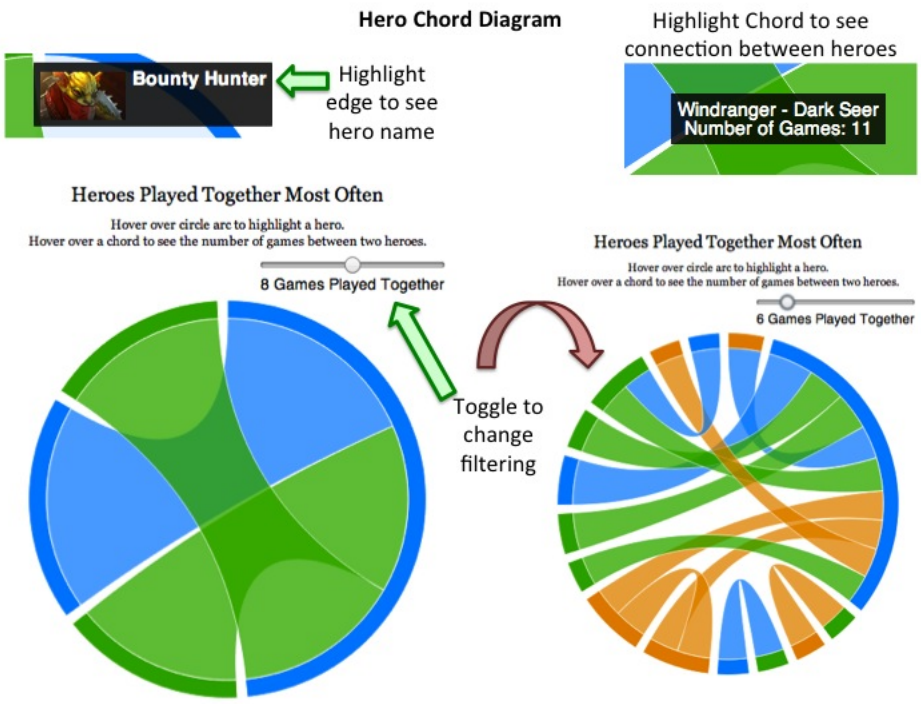


Figure 56: Final Feature Description

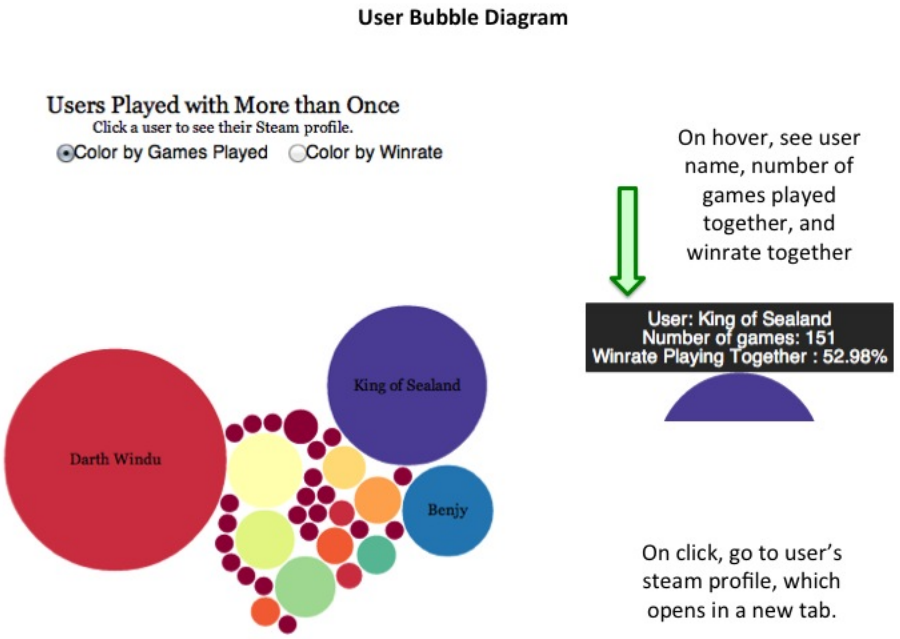


Figure 57: Final Feature Description

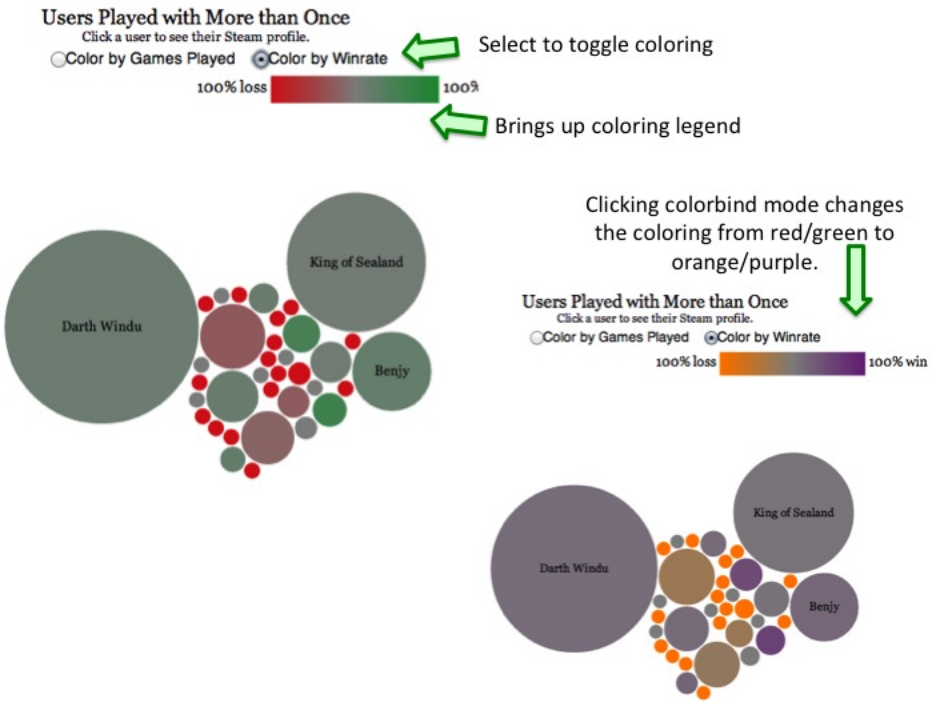


Figure 58: Final Feature Description