FIT2079 – DATA VISUALISATION PROJECT

An interactive walk through films and their actors

OCTOBER 27, 2014

JAMES ASHWORTH
26164035

Table of Contents

Concept

Origin

The original concept for the visualisation came from a few places (seen in Appendix 1). First was an image on "Whedon's World", showing where fans of Joss Whedon's work might know each actor from Much Ado About Nothing (2012). The second was an implementation (in D3), hosted on the New York Times, showing connections between directors and actors that often worked together, "Constellations of Directors and Their Stars". Finally, there was a demonstration of a similar concept using WebCoLa, which did the majority of what I was expecting in terms of functionality, but I felt that the presentation was sub-optimal. Called "Incremental exploration of a large graph", it is the only dynamic example here.

Whedon's World

This visualisation is very close to what I desired, but is static. The key element I wanted to incorporate was the ability to go from a film to an actor to see all credits, and the ability to go from an actor to a film to see all cast.

Constellations of Directors and Their Stars

This visualisation has a number of flaws that I wanted to avoid: when highlighting a connection between a director and an actor (representing a film), the resulting tooltip is often obscuring the director, the actor, or both; the hit box to highlight a connection can be fiddly; there is no way to differentiate the films without highlighting the connection; and the sidebar and main bar scroll at different rates, which is unhelpful.

Incremental exploration of a large graph

The visualisation allows the user to perform a walk through films and actors, but leaves behind the paths unfollowed, causing a massive amount of clutter. The dynamic positioning of elements to avoid collisions also causes a jitter, which is not helpful. The tooltips are also very sparse in terms of information given.

Design

From Whedon's World, I chose the idea of a central circle with radiating lines. I also wanted to ensure that the information displays did not obscure relevant information, so tooltips needed to not overlap with visual elements as much as possible. In order to make it obvious which records you are looking at, a numbering system around the exterior elements would be useful. In addition, a breadcrumb system allowing you to browse backwards and forwards along your path would increase the usability, and highlighting to show the record you arrived from.

Operation

In order to make it interactive, I wanted to be able to choose an exterior circle and have it move into the middle, and have the visualisation reform around it. Hovering over a circle should also mark it in some way to indicate the location of the mouse, and display additional information, so it was decided to have the circle expand and a tooltip appear. The breadcrumbs indicate a continuity "jump", so no transition was created. A search function was also desired, so, as it also represents a jump, it should be in a similar location. As the breadcrumbs were deigned to be in the top left and right, the search functions were also placed at the top. Given that the search buttons were going at the top, placing the next / previous buttons close by was a logical choice.

Implementation

From the beginning, this project needed to be interactive and able to operate on a remote data-set (namely the records at themoviedb.org), so a system able to utilise the API was necessary – as such, R / Shiny were dismissed as being unsuitable. D3 appeared to be a suitable choice, and the decision was made to proceed until either the project was complete or some incompatibility necessitated a reconsideration. A breakdown of the functions can be found in Appendix 2.

The initial step is to get the configuration information – via JSON from themoviedb.org – and set up the basic web-page – namely the two search forms, the header and sub-header, the previous and next buttons, the tooltip, and the SVG canvas.

The next step is to interface with the API again to retrieve the initial data-set in the JSON format, and parse it into a more manageable array. This is done in three steps: the initial retrieval and parsing of the central record (The Avengers); the retrieval of the cast array; and the iteration through the cast array to retrieve and parse the surrounding records (Robert Downey Jr., Chris Evans, Mark Ruffalo, etc.). Each circle is positioned at the same time, using polar notation converted to Cartesian coordinates to calculate the location of the centre of each circle.

Once the data has been fully parsed, each circle and line is created on the canvas, then all are faded in. Events are added to anything that should do things, and then the system waits for user action. The actions themselves are as follows (in no particular order):

Mouse over a circle

When the user mouses over a circle, it grows to show more of the image contained within (unless it's the central circle) and displays the tooltip to give the information.

Mouse out of a circle

When the user mouses out of a circle, it returns to its original size and the tooltip disappears.

Click on a circle

When the user clicks a circle, it grows (if it was not already at maximum size), moves to the middle, and the cast record (if a movie) or acting history (if an actor) is retrieved and iterated through to create a new outer ring. The previous central record is then placed on the breadcrumb 'stack'.

Click on previous / next buttons

When the user chooses previous or next, if there were more than 20 records, the circles rotate to either show the previous or next 20 records, or to show the first or last 20 records.

Click on previous / next breadcrumbs

When the user chooses a breadcrumb, the system resets to that point (with no animation).

Search for a movie / an actor

When the user searches for a movie or actor, the search request is sent to themoviedb.org, and the first result is used to set the central circle. The system is then reset as if this were the initial record.

Analysis

Layout Stability

The principle of layout stability states that if the information displayed is not being changed, the display itself should not change. The WebCoLa visualisation has an annoying tendency to jitter even when there is no user interaction, which can give the impression that the display is somehow readjusting to take account of new information, which is not the case. This visualisation, on the other hand, does not move any information around without express user interaction.

Additionally, any changes in the information on display should be accompanied by some form of visual cue, generally an animation. The WebCoLa visualisation is better about that, but there is an appreciable delay as it determines additional connections and can occasionally be adding new links several seconds after it seems to have settled down. With this visualisation, there are four ways to change the displayed information: looking at additional records for the current display; choosing an external record to move to; switching records through breadcrumbs; and searching for new records. Each is analysed separately below.

Additional Records

When choosing to view additional records, the animation to make this clear treats the visualisation as a cross section through a spiral. Records rotate up / down and off the plane of visualisation as they leave the current scope, and more records are pulled in from above / below to replace them. The effect is that of a spinning wheel with a boundary between records 1 and 20, where records appear and disappear. This can be observed in the screenshot “Rotating” in Appendix 3.

Moving to External Record

When choosing an external record, all other records are faded out and the chosen record is moved to the centre, at which point the new external records are faded in. This can be observed in the screenshots “After Click” and “After Move” in Appendix 3.

Breadcrumbs

When navigating through breadcrumbs, because of the jump nature, the animation is not smooth – rather, the entire system resets – all visual elements are removed, and replaced with the previous iteration, and the breadcrumb images are updated.

Search

When searching, as with breadcrumbs, there is a jump, so the animation is as before – a complete reset.

Layout Structure

Considerations of layout structure tell us that any design should be compact – wasted space is not useful. WebCoLa is possibly ahead of this visualisation in this regard. The spokes of the wheel do not currently provide any additional information, and the packing of the circles is not great. That being said, aesthetically, I find the current number of circles to look nice, rather than fewer, larger circles which would pack more information in.

The aspect ratio is another element of layout structure to be assessed. In this instance, WebCoLa and this visualisation both have shortcomings: WebCoLa doesn’t adjust its layout to suit any aspect ratio, and with a certain sequence of decisions, the system can become significantly wider than tall, or vice versa; by contrast, this visualisation is constrained to a square. Generally speaking, I feel that a square is unlikely to cause a significant wastage of screen real-estate, so I believe this to be acceptable.

Edges should be of uniform length if they represent the same connection identity. In WebCoLa, because of the dynamic layout, edges are often stretched or compressed to fit, whereas the fixed layout of this visualisation lends itself to this criterion.

Edges also should not cross one another, or as little as possible. WebCoLa's dynamic layout does do its best to avoid crossings, but is not perfect. Once again, a fixed layout ensures adherence.

Edges should have high angular resolution – they should not lie tangent or even close to it with elements they exit. Both WebCoLa and this visualisation have edges that coincide at their centres, and therefore cannot have a higher angular resolution.

Elements themselves should not overlap. Again, WebCoLa's dynamic layout prevents overlap (using a spring-force model, by the looks of things), and the fixed layout ensures no overlap on this visualisation.

Interaction

Hovering over records should have a visual cue to represent that the record is under inspection. We cannot move the record, as that is used to represent selection, and likewise, blinking the record could imply a data change, so instead we increase the size of the circle while the mouse remains over it. WebCoLa also has a (slight) visual cue, in that the exterior of the elements show their tendency towards new connections when we hover – in the event that we're looking at an already expanded node, there is a delay while we await the tooltip to indicate a response, which is suboptimal.

Additionally, when a record is under inspection, we should show additional information. This is achieved through the use of a tooltip which is displayed adjacent to the record, displaying information such as the name of the film or actor, and the character that links the actor to the film. WebCoLa, on the other hand, only displays the name of the actor or the name of the film.

Gestalt

Records that are connected are grouped together. In terms of this visualisation and WebCoLa, there is not a massive amount of relevance to this concept, given that they represent the cast or credits for a film or an actor.

Objects that act together are grouped together. WebCoLa has no removal or animation of objects, so does not utilise this. This visualisation, on the other hand, fades out deselected objects, and rotates all exterior records, to reinforce that they are grouped.

We are good at recognising symmetry, so it was important that for results with fewer than 20 children, they be equally spaced. WebCoLa, with its dynamic positioning, does not create any kind of symmetry.

Further Work

Element Shape

It would be useful to be able to tell at a glance what any particular element is (film or actor) by shape. Additionally, while the circle is fine for headshots, given that people have roughly spherical heads, posters are inherently rectangular, and thus would suit a more angular representation. That being said, given that is not possible to directly connect a film to a film, or an actor to an actor, this is not to be considered a high priority, and is difficult to achieve due to the nature of item selection in D3.

Recognition of Posters

It is sometimes difficult to recognise a movie poster, unless it is particularly iconic. Having to hover over each film to identify it is inconvenient. Therefore, it would be useful to have a fixed element attached to the film with its name. However, the positioning to keep it legible is not immediately obvious with the current structure, so it would require significant work.

Ordering

The actors are ordered by billing for any given film, but the ordering of the credits for an actor is indeterminate. It would be useful were the ordering to be based on rating, or year of release, but to do this would require a reasonable amount of post-processing and larger films are already somewhat slow to load. Investigation into the API may yield a better way of ordering, but for now, it remains random.

Incompatibility

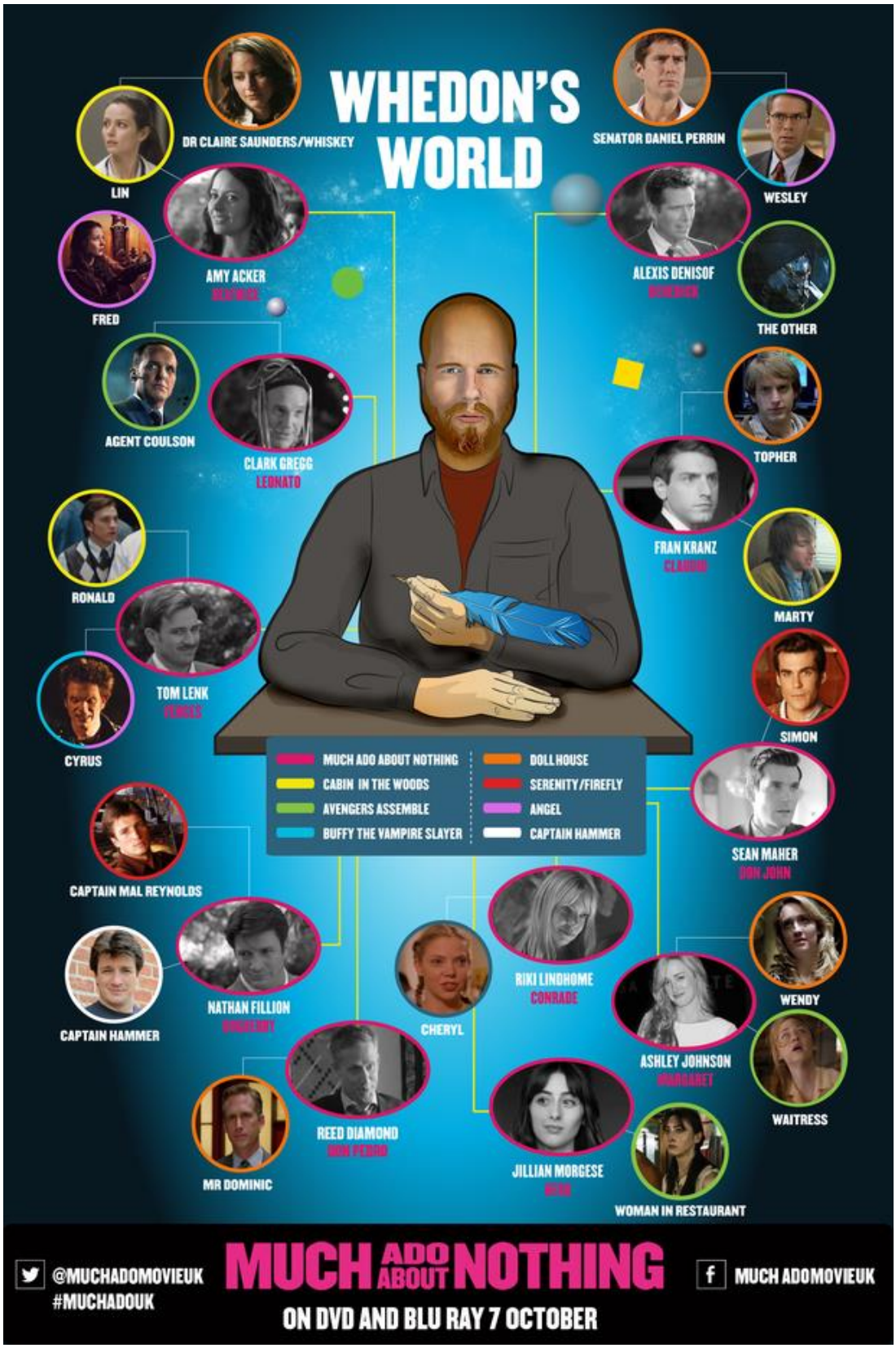
This visualisation currently only works in Chrome and Firefox – Internet Explorer and Safari both break when they encounter D3's remove function. There does not appear to be a work-around for this, and the code would require a complete rewrite to avoid this, as well as adding significant complexity. As such, the suggestion would be to add a wall for IE and Safari, to prevent access and redirect to a different browser.

External Links

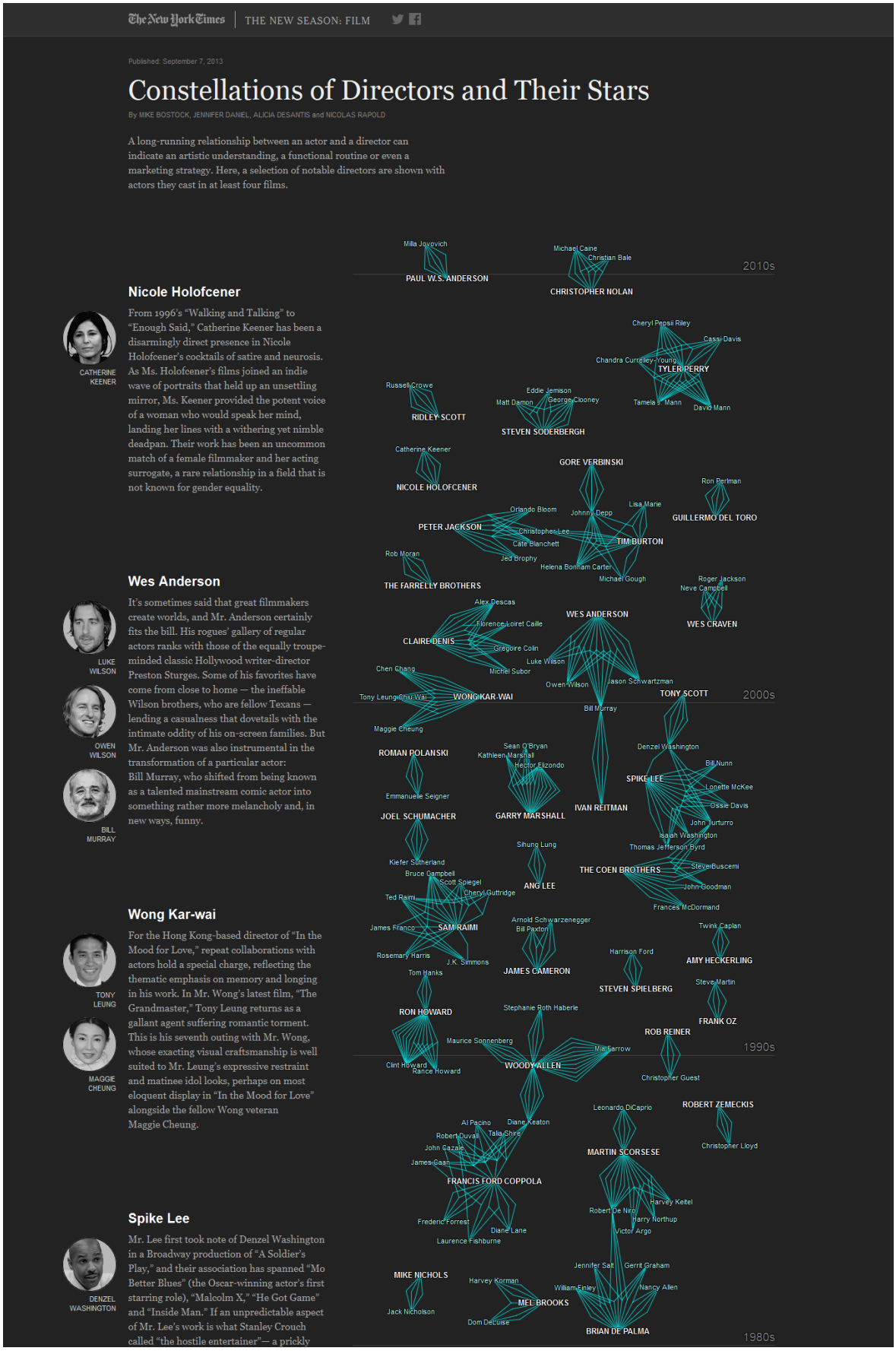
Ideally, there would be some way to go from a representation in the visualization to the page on theMovieDB.org that the information is pulled from. Given that left-click is for selection, and right-click is generally slaved to context menus, middle-click would be the ideal choice. However, despite multiple attempts, SVG has stubbornly resisted hijacking the middle-click. This would probably be the most difficult to implement, but requires the least modification of existing code.

Appendices

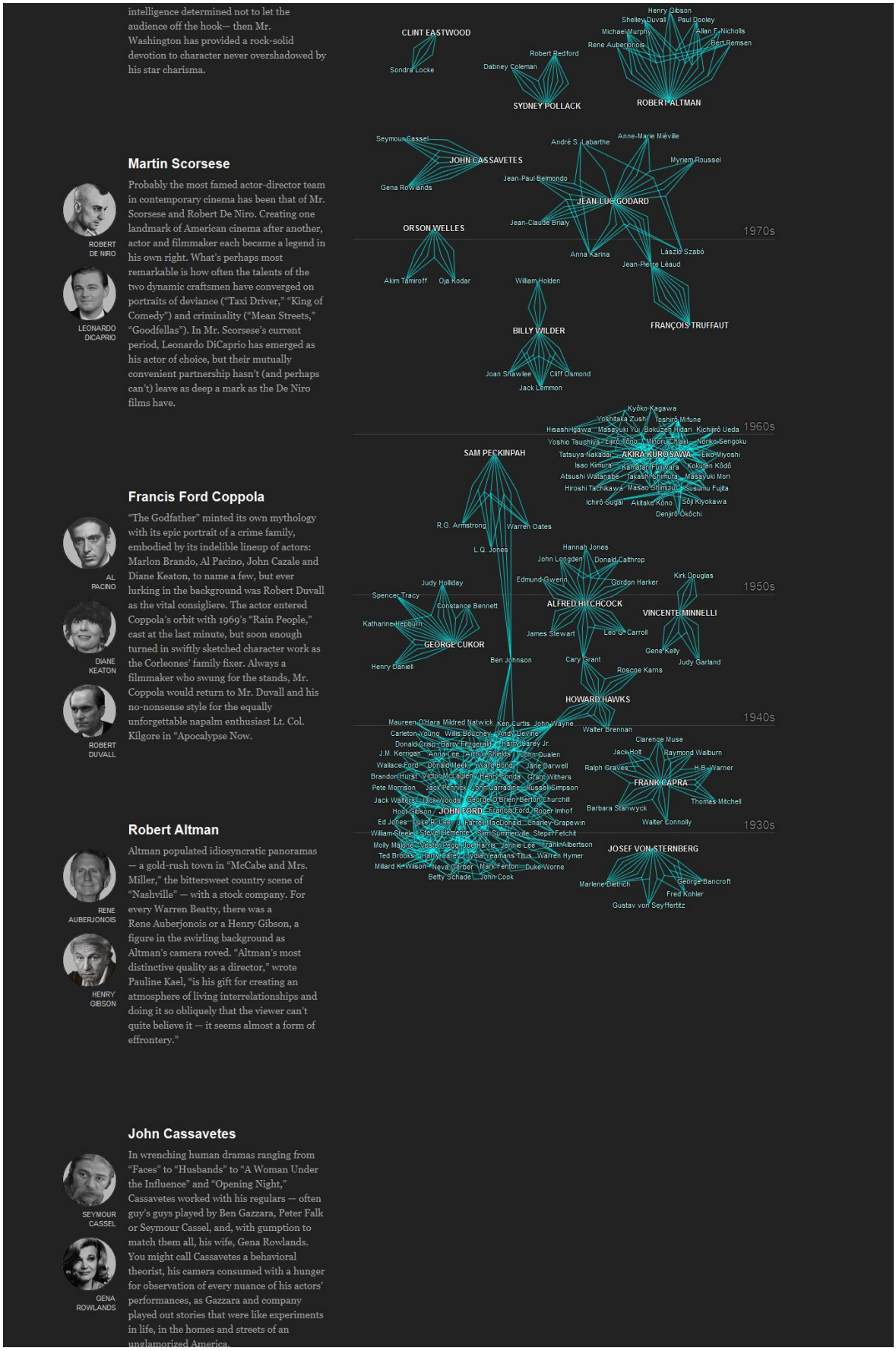
Appendix 1: Inspiration



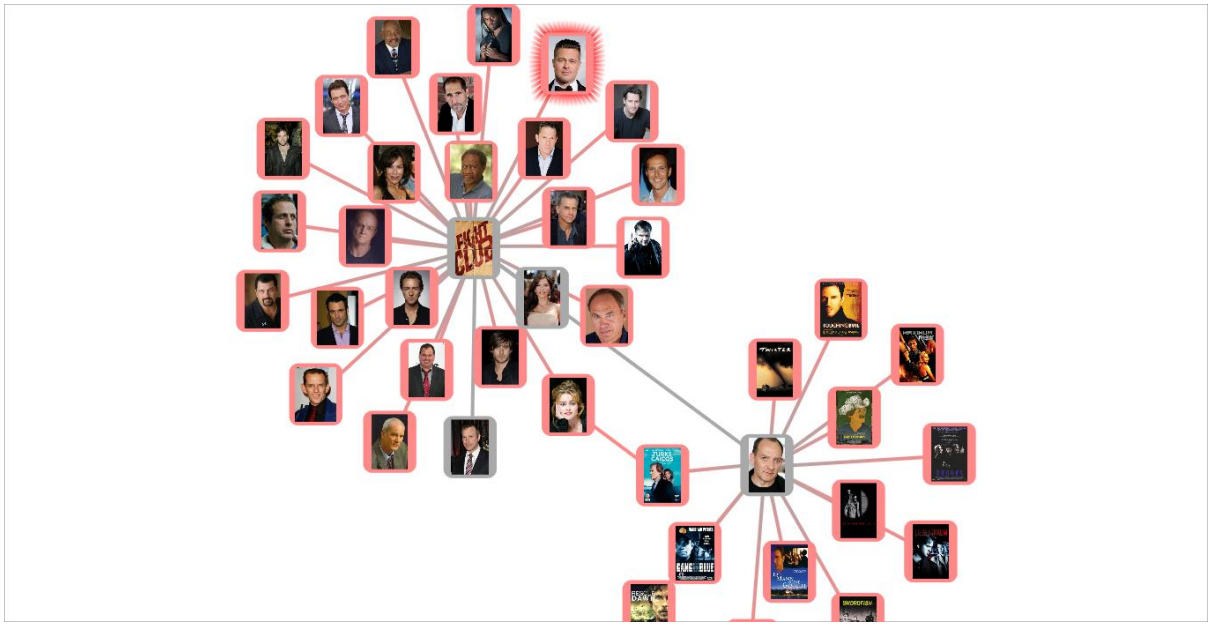
http://www.digitalspy.com.au/movies/at-the-movies/a519608/joss-whedon-the-whedonverse-in-one-handy-infographic.html



http://www.nytimes.com/newsgraphics/2013/09/07/director-star-chart/



http://www.nytimes.com/newsgraphics/2013/09/07/director-star-chart/



http://marvl.infotech.monash.edu/webcola/examples/browsemovies.html

Appendix 2: Functional Breakdown

Global:

- Configuration variables
- Search fields
- Header
- Previous / next button
- Subheader
- Tooltip
- SVG canvas
- Previous / next breadcrumbs
- Initial configuration request

dataCreate:

- Disables buttons
- Clears information
- Resets wheel position
- Resets data
- Calls dataSetup

dataSetup:

- Requests the information for the central record
- Calls centerParse
- Calls dataExtend

centerParse:

- Creates central tooltip
- Creates central image
- Names the page
- Enters the header
- Calls makePrevNextPatterns
- Resets the parse counter

makePrevNextPatterns:

- Creates the previous / next breadcrumb patterns

dataExtend:

- Requests the information about the surrounding records
- Determines the number of circles
- Calls dataCreate2

dataCreate2:

- Create placeholders for the surrounding records
- Calls dataRequest
- (If there are no surrounding records, calls setup)

dataRequest:

- Requests the information for the surrounding records
- Calls dataParse

dataParse:

- Creates surrounding tooltips

Creates surrounding images
Once all circles are parsed, calls setup

setup:
Calls onceMore
Calls doFadeIn
Calls updateDisplayBox

onceMore:
Creates the lines
Creates the circles
Puts circles on top of lines

doFadeIn:
Resets counters
Calls fadeIn

fadeIn:
Fades in the circles
Fades in the lines
Enables the buttons
Calls visibilityUpdate
Calls addEvents

updateDisplayBox:
Fills the subheader

visibilityUpdate:
Ensures that only circles that should be visible are visible
Circles that are about to become hidden are semi-transparent
Circles that are being unhidden are semi-transparent

addEvents:
Adds mouseover event to circles to show the tooltip and call increase
Adds mouseout event to circles to hide the tooltip and call decrease
Adds click event to circles to remove events and call doSelect
Adds click events to breadcrumbs to call onPrevious and onNext respectively

increase:
Grows the circle to double size and moves the tooltip

decrease:
Resets the size of the circle

doSelect:
Prevents any button clicks
Calls increase
Clears header and subheader
Sets up breadcrumbs
Creates the master record
Calls beginSelect

beginSelect:
- Resets the counter
- Calls fadeOut

fadeOut:
- Removes the numbers
- Fades out the lines
- Fades out the (non-selected) circles
- Calls moveToCentre

moveToCentre:
- Smoothly moves the selected circle into the middle
- Clears out extraneous patterns
- Sets selected circle and friends to center circle
- Resets counters
- Calls dataCreate

onPrevious:
- Sets up breadcrumbs
- Moves breadcrumb to master record
- Clears out everything
- Calls dataCreate

onNext:
- Sets up breadcrumbs
- Moves breadcrumb to master record
- Clears out everything
- Calls dataCreate

movieSearch:
- Requests information for search information
- Calls searchReturn

actorSearch:
- Requests information for search information
- Calls searchReturn

searchReturn:
- Informs the user that no results were found; or
- Clears out the search boxes
- Sets up the breadcrumbs
- Sets up the master record from the search return
- Clears out everything
- Calls dataCreate

rotateNext:
- Prevents any button clicks
- Calls doRotate

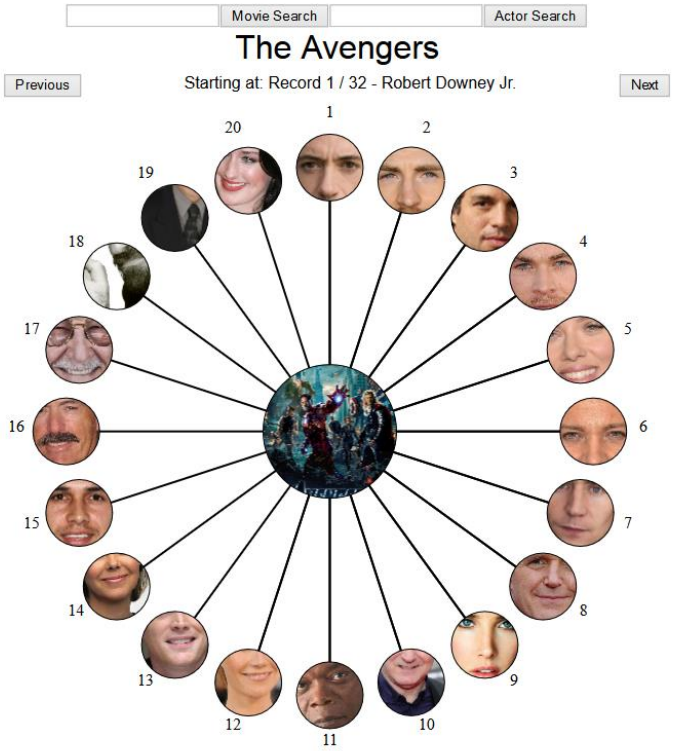
rotatePrevious:

Prevents any button clicks
Calls doRotate

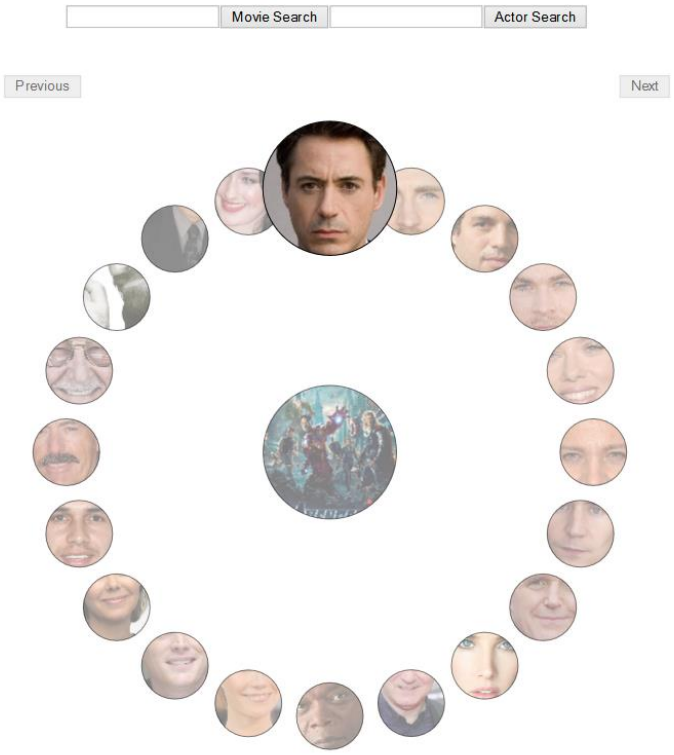
doRotate:
Removes all click events
Fades out the lines
Rotates the circles
Calls positionUpdate
Calls visibilityUpdate
Calls updateDisplayBox
Fades the lines back in
Enables button clicks
Calls addEvents

positionUpdate:
Recalculates the locations
Updates the circles with new locations

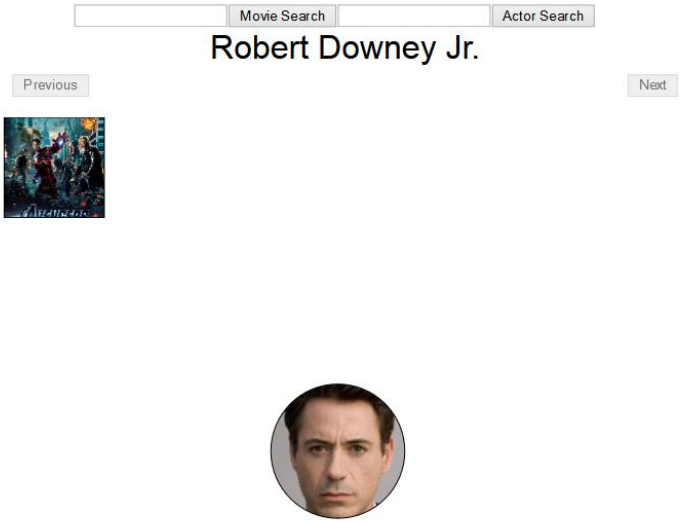
Appendix 3: Screenshots



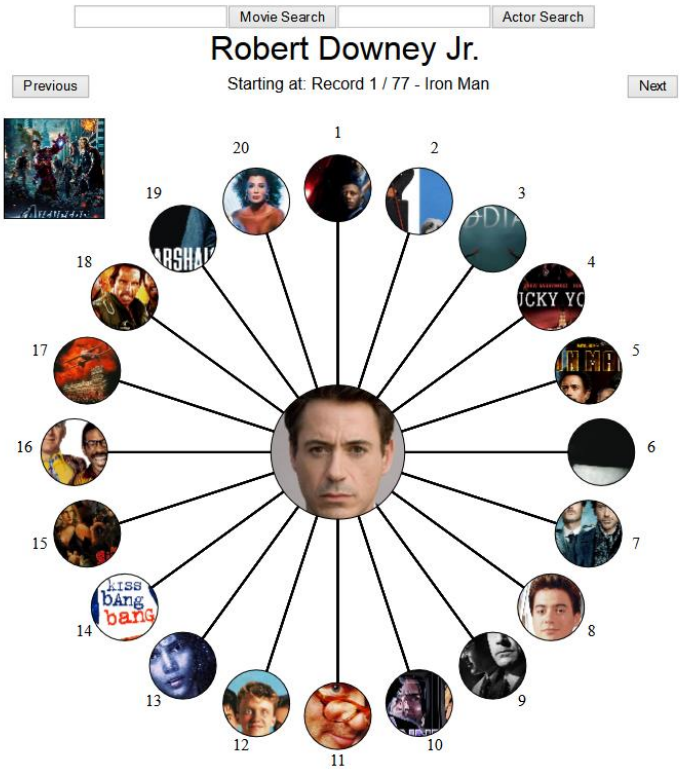
Initial View



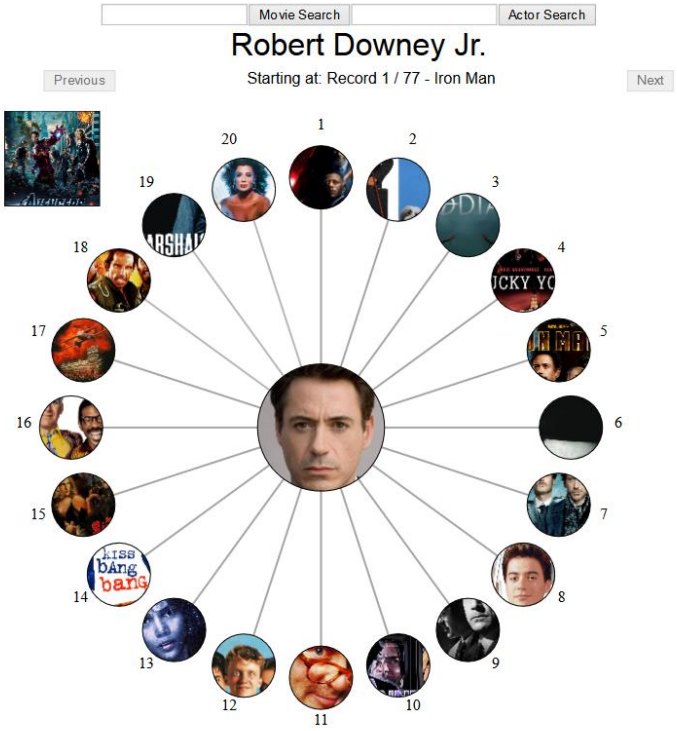
After Click



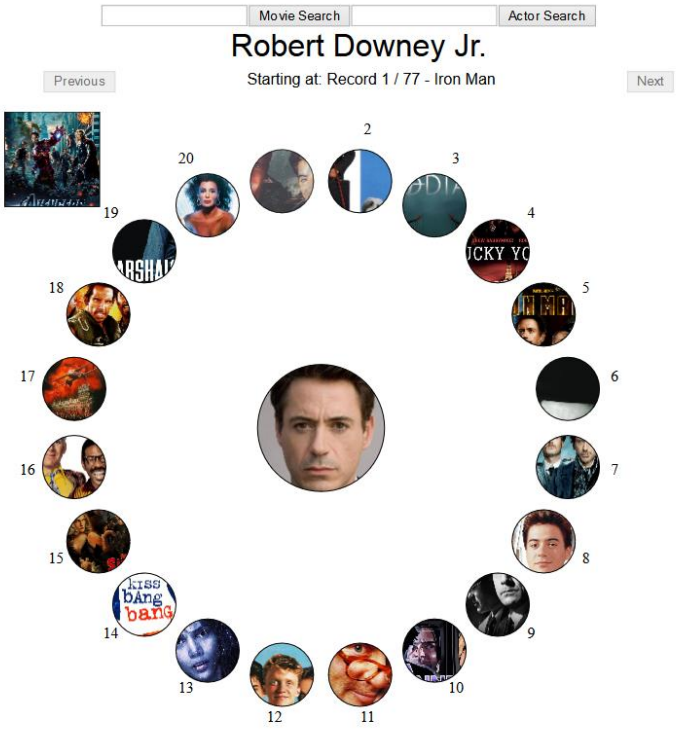
After Move



Awaiting Input



After Next Click – Fade Out in Preparation for Rotation



Rotating – Observe Overlay of Circles 1 and 21 Through Transparency