Introduction

Harvard SEAS is a strange place.

Think about it. A young school, not fully detached yet from FAS and GSAS, no official departments yet with pseudo-departments in the concept of "teaching areas", teaching areas that don't directly match the degrees offered, and often no clear path from your first day in the door to the sheepskin at the end. It's often said that SEAS is in startup mode -- and that's definitely reflected in its structure, or lack thereof.

The SEAS Academic Visualizer (SEASAV) is an attempt to give structure where there's very little. It focuses on *connections* -- how research interest areas connect to degrees, how degrees connect to teaching areas, how teaching areas connect with faculty. The hope is that by drawing these connections in a clear and visual manner, we can funnel visitors to more in-depth content and help them understand how SEAS works.

Project History

The SEAS Academic Visualizer was attempted once before, in the spring of 2013, through hiring an outside freelancer, but the Office of Communications wasn't quite satisfied with the results. SEASAV 1.0 was a tabular depiction of our content; when you clicked on a name in one column, the other columns would close up to show only content related to that item. It worked, but it didn't give a feel for the interconnectedness of SEAS at a glance, and it only connected items in adjacent columns -- which meant that specifically that Faculty were not connected to degree programs, a potentially useful connection to know

In Fall 2013 we observed that CS 171 would be offered in the Spring of 2014 and available to Extension School students. We were ready to try again, this time with all the design and implementation in-house.

About the Developer

As the primary developer for the SEAS public-facing website, it was decide that I would take the class and apply what I'd learned to SEASAV. I'm a web developer with nearly 15 years of experience in everything from good old hand-tagging of XHTML to dealing with XML/XSLT/XPATH-based content management systems to developing custom Drupal modules. I had had some experience with javascript and jQuery, but none with d3.

More interesting, perhaps, is that my BS in Computer Science came with a rider -- Studio Art. Art taking a good portion of my free time (everything from professional pet portraiture to volunteer traditional feature-film animation), the combination was perfect for a foray into data visualization.

Meet the Team

While this project might not be owned by an in-class group due to its nature as a work-related project, there is still a team involved. For the purpose of the class, it might be best to think of

this team as my clients -- they provide me with feedback, data, and good questions. While we'll seek feedback from a larger group after the completion of the semester, the core group is as follows:

- Paul Karoff, Executive Director of Communications, SEAS: Project sponsor. Bounces ideas and helps clear the way for obtaining data.
- Eliza Grinnell, Communications Project Manage, SEAS: Design sounding board and local branding expert. Source for SEAS-related graphics.
- David Hwang, Assistant Dean for Education, SEAS: Master of data relating to SEAS programs and teaching.

The team will be kept up to date on all progress for the duration of the project and will determine when it's ready to go live on the SEAS web site.

So what *are* We Trying to do here?

Put simply, we're trying to take a confusing place and make it make sense. It can take years for people on the inside to understand SEAS structure -- prospective students don't have that kind of time. We have a limited shot at their attention and time, and we need to use it as best we can. Pages of text, argued and fought over and lengthened repeatedly by various stakeholders aren't going to do it.

We need something that says right out: We have an unusual, interdisciplinary structure -- but you CAN find what interests you at SEAS, and we can help you find it.

Our Data

The SEAS Website

We're sourcing our data for SEASAV from the public SEAS website (http://www.seas.harvard.edu/). Most of this content is fairly organic and maintained by hand through Drupal's editing interface, but some of it -- the directory content in particular -- is sourced from Active Directory. Rather than pulling content from multiple sources, since Drupal already consolidates the information, we've settled on it to be our single source.

Everything that will be shown in SEASAV exists somewhere on the site already, though many of the connections are hidden from the public, or perhaps used for features such as related news or grouping directory entries into sub-listings.

Currently, the major sections of the site that relate to SEASAV are:

- **Individual directory entries:** Each member of the SEAS community has one of these and it shows your typical directory information -- name, phone, location, and so on. Faculty entries are a little more fleshed out, and contain text that indicates their *research interests* (selected from an approved list), their *primary teaching area* and other pertinent

information. Connections between faculty and research and teaching can be done from these pages, but only on a person-by-person basis.
http://www.seas.harvard.edu/directory/pfister

- **Teaching Areas**: These are the pseudo-departments. Each has their own portion of the site with basic information about the area. These pages are very much informational, and not always consistent or up to date. For the most part, these pages are only concerned with their subject, so connections are not obvious, though there may be pointers to the degree programs, and each area has a "People" listing where you can see Faculty that teach in that area. http://www.seas.harvard.edu/computer-science

- **Research Interests:** Right now this appears on the site only as a listing of interests and the faculty who have selected them (and, of course, on each individual's directory page). This isn't particularly useful, and is on the list of items to improve, but for now, it is what it is. http://www.seas.harvard.edu/faculty-research/research

- **Degree Programs:** Two listings appear on the site -- one for undergrads and one for grad students -- and the listings feed visitors off to small sites for each degree. This content is also in flux, but in general it can be assumed that each program will have its own small site. The program sites definitely do not connect well with any other information on the site, though an attempt was made on the undergraduates' listing page to try and simplify the connection between basic groups of interests and the appropriate degrees by making the degree listing a table of interests against concentrations. This small thing made a major difference in visitors' understanding of the degree, and we feel it bodes well for the inclusion of the visualization.
http://www.seas.harvard.edu/academics/undergraduate (listing),
http://www.seas.harvard.edu/programs/computer-science (program)

As you can see by browsing the above pages, there's a lot that can be done to clarify how all the pieces of SEAS fit together.

Data Manipulation

Drupal makes it easy to categorize content based on tagging or content type, and we're doing a little of each. Programs and Teaching Areas both have their own special content types; Faculty are identified by content type and a set of values in a particular field; Research interests are a structured taxonomy. Connections between content types are made via tagging and extrapolated from tagging (e.g. Person A is tagged with Area B and Interest C, so Area B and Interest C should connect). Summary content for the detail view is pulled from various fields, most typically the body summary, but in the case of the Faculty images and titles are pulled as well.

Data manipulation is done through a custom Drupal module that pulls content from Drupal's core database. More details on how this is done can be found in a later section.

The First Steps

When the project began, essentially all we knew was that we didn't want to use the tabular design -- everything else was up for grabs. Eliza had some favorites among public visualizations (including the China Project), though nothing specifically fit what we had. It was pretty clear that we were going for a network graph solution, but which type wasn't yet clear.

I gathered the data early, so when Homework 2 was assigned, and I saw that I had a chance to throw some rough data at it early. This can be seen in http://theredsetter.com/cs171/snapshots/simple_graph.html -- there are no labels, here, though you can color by category, a distinction that remains in later versions; this was simply a chance for me to see what might work and what might not.

It became clear at this point that we were looking at something similar to a radial graph; nothing else would fit the bill.

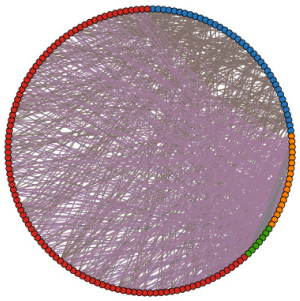
At this point I sat down and started thinking of possible variations on the theme. One of the first that came up, given that we had four categories, was to make the graph in the form of a rectangle -- bigger categories like the Faculty and Research interests on the top and bottom and Areas and Degrees along the sides. This had some major issues, though, not the least being that it wasn't extensible -- we were stuck with four categories forever.

So I moved on to the true radial graph, where I stayed. There are still some problems with this solution -- it won't scale indefinitely -- but it presents our information in the best way possible, and with some work on the interactive side, scaling issues can be mitigated.

The Process

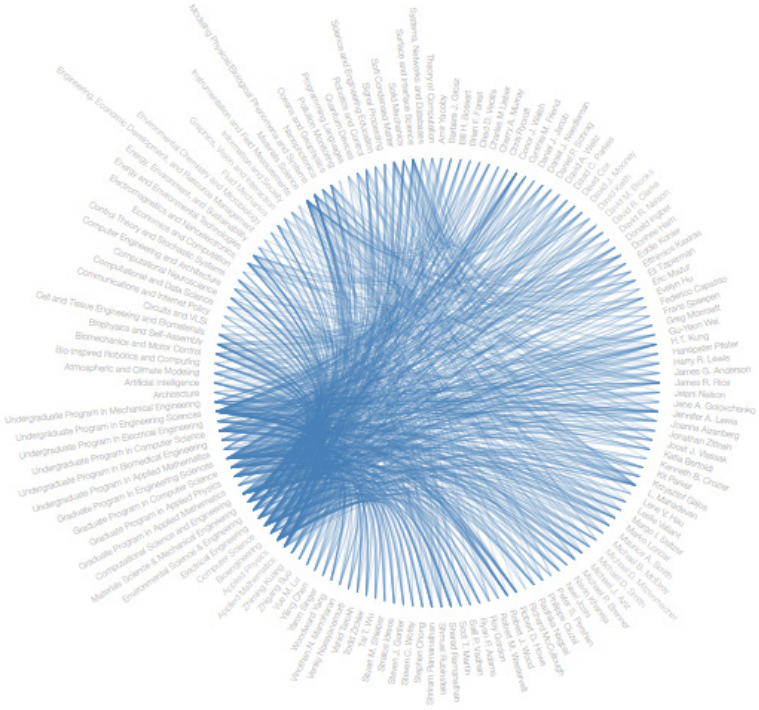
To allow my non-technical team access to review early designs I took "snapshots" on days that I'd worked on the project -- this allowed them to be able to look back on previous steps and see the latest stable code without needing to understand git. This had the interesting side effect that I have a record of the changes I've made out where everyone can see it. Below are links and screenshots of most of the stages of the project along with some notes (extremely similar stages or stages with bugfixes only have been left out).

- http://theredsetter.com/cs171/snapshots/simple_graph.html
  My first look at the data in graph form. I simply stole HW2 and threw some very, very raw data at it. You can do any of the variations that were available for HW2 (with the exception of node size) to see what I saw when I first played with the data. Not very exciting, but this was the first indication that the radial graph was going to be what I needed. The data here is downright wrong, as well; it was not filtered as well as it should be and the result was some nodes with no connections (if you're



not connected to anything else we're not interested in you in this graph).

- http://theredsetter.com/cs171/snapshots/test20120228.html
  A first shot at applying the data to the radial graph code at http://bl.ocks.org/mbostock/7607999. This is essentially a vanilla version of the graph; no indication of category, no labels, and only very minor CSS tweaks. But it made it clear that I was headed in the right direction. At this point in the game there was a major mistake in the data -- I had each connection doubled, which was unnecessary and in some browsers on some machines really slowing things down.



- http://theredsetter.com/cs171/snapshots/test20120304.html
  This is starting to look more like the final -- we now have labels (though not what you'd call pretty labels) and are using the pie chart to make it clear what's in what category.

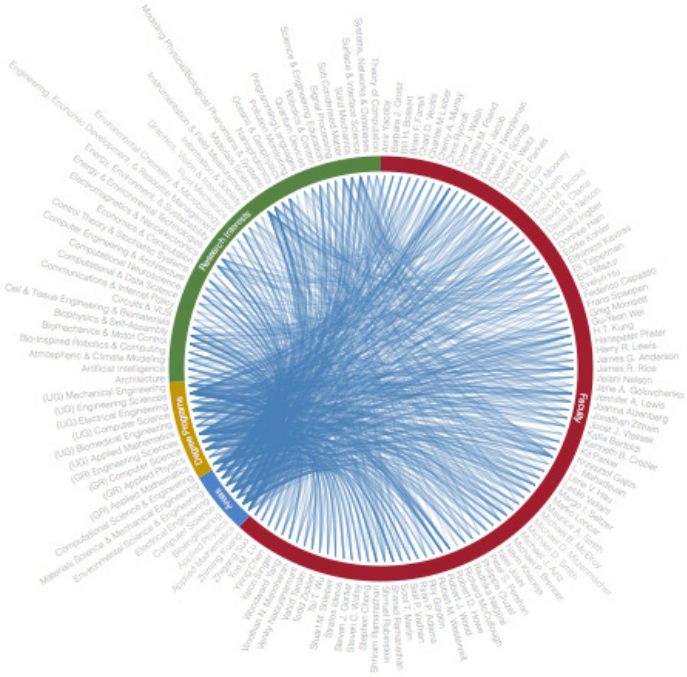
- http://theredsetter.com/cs171/snapshots/test20120307.html
  This takes the previous version and applies an arc to the text -- much, much better looking, easier to read, and avoiding the overlapping text issue. Now we're starting to see that we're going to have issues with node title length, and the next couple versions were fighting with possible solutions to that issue as well as perfecting the node title animating to the center of the screen.

- http://theredsetter.com/cs171/test20120318.html

Here we see the solution to the long-title issue -- title truncation. At this point we're not perfect yet; the only way you can read a full title is to click it and get the detail view. The detail view is fully realized at this point, though the data included in it is still sparse -- improvements to the Drupal module producing the data are still necessary at this point.

- http://theredsetter.com/cs171/snapshots/test20120324.html
  We see the data starting to improve -- descriptions are available for most items, Research Interests excepted. Also, experimentation with having the circle size respond to screen size has begun. Larger screens get a larger circle here, and therefore more room for node titles. There is a lower limit to circle size; smaller screens may have to scroll; that's an unfortunate side effect of the amount of data. This will never work well on a phone, for example.

- http://theredsetter.com/cs171/snapshots/test20120326.html and
  http://theredsetter.com/cs171/snapshots/test20120326b.html
  20120326 is a stable version of 20120326b without the rotation trick added for the b version. There was a request from the team to allow category rotation to the left of the screen to aid in reading node titles in the category of interest, and this was the first attempt. It does work, though the code was refined for the final, and one bug which persisted to near the end (rotate to Research Interests and have a look at the faculty names near the top of the screen to see it) was introduced here.



- http://theredsetter.com/cs171/snapshots/test20120328.html

A very broken first attempt at filtering! While the in-page search was all of five lines of code and extremely simple, the filtering gave me multiple days of hair-tearing. Here we do have the node data exiting correctly, but we hit the end of the day without cornering the filters. Tooltips show up here for the first time as well, made using the title element. These had issues; they were ugly, and the delay was too long for people to notice them.

- http://theredsetter.com/cs171/snapshots/test20120331.html
  Here we see some buggy first attempts at animations for the filters. The filters themselves do work fairly well here, but the category animation is very flakey. Ironically animation had to be abandoned in the end anyway, because full link animation was too much for the average computer to take (and I did have working code at one point). With the node/link animation abandoned it made sense to also discard the category animation and just live with the lack of transition.

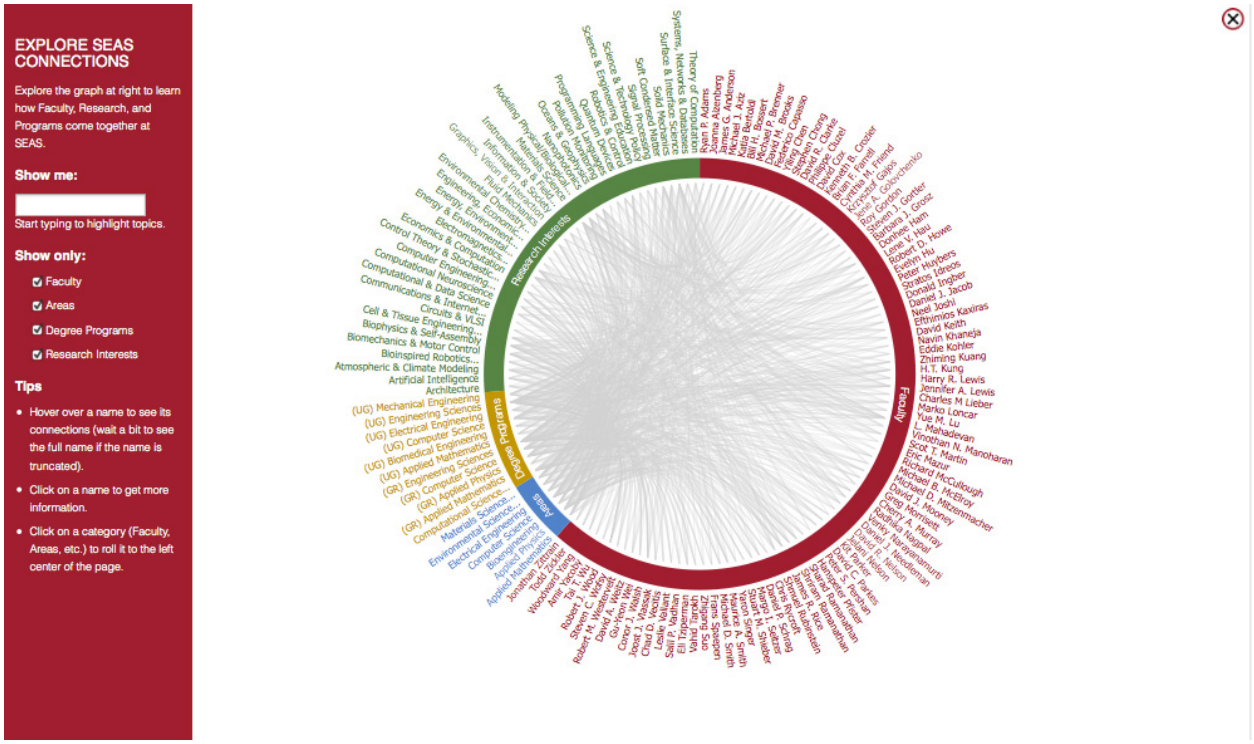
- http://theredsetter.com/cs171/snapshots/test20120401.html and
  http://theredsetter.com/cs171/snapshots/test20120403.html
  Primarily code cleanup and organization.

- http://theredsetter.com/cs171/snapshots/test20120404.html
  Borrowed a Windows machine to experiment with IE9; while no browser but Chrome was required for the class project, SEASAV has to work on all major browsers before we can be comfortable releasing it publicly. Making it work on IE meant that I had to drop the use of SVG ForeignObject and go with a div that's positioned over the SVG graph rendering. Calculations were a little tricky for this, and I believe that this version still has some centering issues on very small screens, but these were later corrected.

- http://theredsetter.com/cs171/snapshots/test20120407.html
  Very few behavioral changes here, but the JSON data has been improved again, this time including more information in the Faculty detail view. Also some CSS tweaking here, especially as it relates to the detail view.

- http://theredsetter.com/cs171/snapshots/test20120410.html
  Bugfixes and implementation of Design Studio feedback. In particular, this phase improved search highlighting, increased node text readability, refined the tooltips, and colored the links according to their target category when highlighted. Introduced a bug where the link coloring was lost in the detail view.

- http://theredsetter.com/cs171/snapshots/test20120416.html
  Fix link coloring loss introduced on 4/10. Fixed text orientation bug introduced with rotation implementation and added code to recalculate text orientation on filter.



Team Feedback on Early Implementations

The team met in March to review a very rough implementation of the design (http://theredsetter.com/cs171/snapshots/test20120317.html). At that time, the following feedback was collected:

- Filters are needed; a visitor should be able to see all the connections or do one-to-one connections between categories (added 3/28)
- An in-page search would be useful (added 3/28)
- Do we want to add a Centers category? (future feature)
- Program connections need to be formalized; we'll begin by tagging Programs with Teaching Areas and Research Interests and using that to associate them with Faculty. This is subject to change. (future feature)

Design Studio Feedback

Design Studio feedback has been touched on in the design studio deliverables, but the following bear mentioning again:

- Improve highlighting on in-page search (added 4/10)
- Improve node text readability (font changed 4/16, tooltips added 3/28 and improved upon 4/10)
- Free rotation (future feature)
- Connections within categories (future feature)
- Color the links according to target node (added 4/10)
- Category tooltips (future feature)
- Add course data (possible future feature; may be way too much content to deal with)

Implementation

FInal Design

The final design is a heavily modified blend of two existing web examples, http://bl.ocks.org/mbostock/7607999 and https://gist.github.com/mbostock/3887235. It is a radial chart with hierarchical edge bundling to make the connections easier to follow, plus a donut chart used not as a chart, but as an easy way to provide labels for the categories.

Detail views are available on click, and tooltips aid in reading rotated text on the radial graph. Filters and in-page search are also available.

Drupal Module

The Drupal module has two parts, one to build and display the JSON feed and the other to create a Drupal block for use on the public SEAS site. At the time of this writing, the Drupal module is not yet live, so what you will see on the project website is standalone D3/XHTML and is slightly different from what will actually appear on the SEAS website.

The JSON portion of the module is fairly simple. It consists of several database queries (the hope is to trim this down eventually, contingent on simplifying some of our content types) to pull detail information and connections between data items. It is not, as it stands now, written to be able to be extended to non-SEAS information, though it might be fun to modify it in the future to work with the Views API and therefore be able to handle arbitrary data.

The block portion of the module creates a block and ensures that D3 and the visualization javascript are not loaded in pages that do not contain said block. Additional controls exist in the block javascript to prevent the block from being displayed in browsers that do not support certain behaviors and to make it play nice with Drupal core javascript.

D3 Code

The D3 code is an adaptation of the typical radial graph. Nodes and links are created based on the JSON data and displayed on the screen.

Several modifications exist, however:

- A donut chart was added to act as category labeling
- The donut chart labels, when clicked, will rotate the entire graph, allowing for areas of interest to be pulled to the left side of the page and closer to horizontal for easier reading.
- To accommodate long node titles, the titles have been automatically truncated
- To make rotated/truncated node titles easier to read, tooltips have been added containing the entire title
- Tooltips are carefully positioned to avoid overlapping the center circle, links, and detail view
- A detail view was added when a node is clicked; the node title moves to the center of the circle and is expanded to a short explanation. More details can be found by following a

link in the detail view.
- Filters were added to exit/enter data categories -- these were initially to be transitioned, but the complexity of the graph resulted in browser crashes; they are now just immediate changes, no pretty animations.
- In-page search was added; typing a string the in the box will result in matching node titles being highlighted.

Evaluation

Being familiar with the data going in, there wasn't much for me to personally learn from looking at it from a different angle, but I am hoping that it will allow for outside visitors to be able to learn more about SEAS in a manageable fashion. Feedback from the Design Studio group was already promising in this way (one of the group members was a SEAS student, so her feedback on how things fit together was especially helpful).

Future Considerations

Public projects like these are never done. Some things I'd like to eventually include in the visualization that are not there now, are:

- Drag to rotate (instead of click to rotate; offers somewhat more control)
- Inclusion of Centers as a category
- Inclusion of Alumni/Careers as a category
- Continued improved handling of small screens
- Tablet support

Conclusion

The SEASAV is already looking to be a success. Test audiences seem to like it, get it, and appreciate the insight that the visualization gives that our more traditionally structured web pages cannot.

Related Sites

All code, including snapshots can be found at http://theredsetter.com/cs171/, the project website.

The git repository for the project is https://github.com/uzuri/cs171-project-mazur-rebecca.

The dev version of what will eventually be the public visualization can be found at http://seasdev.prod.acquia-sites.com/#vis. Note that because this is a dev site, the visualization may not be available at all times.

Eventually, the visualization will live on http://www.seas.harvard.edu/, most likely available from the Home page, Academics, Faculty & Research, and other related pages.

Resources

Much of the basic heirarchical edge bundling code is borrowed from

http://bl.ocks.org/mbostock/7607999; the pie/donut chart used for labels came from here: https://gist.github.com/mbostock/3887235, both by Mike Bostock.