Multidimensional, Interactive Graph Visualization

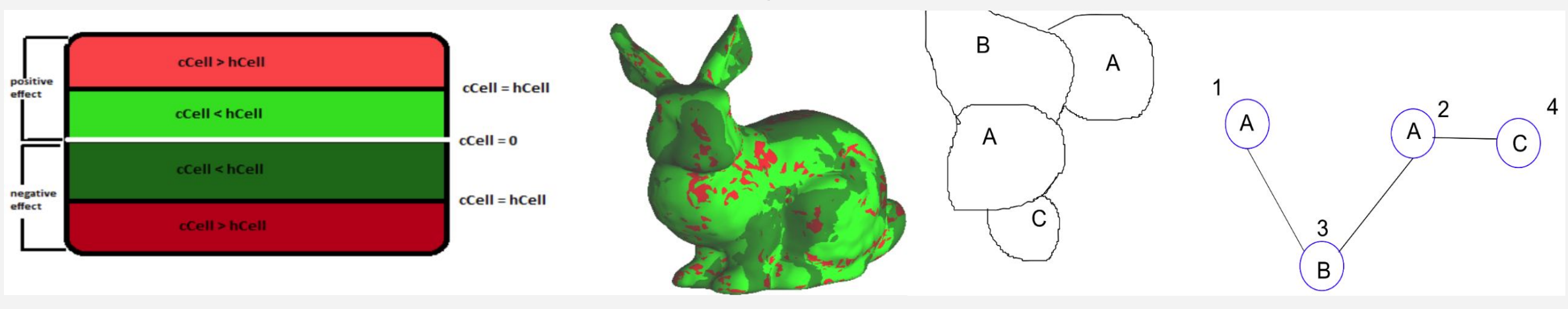
By: Cymon Dillon

Introduction

Internship: My internship was called "Computer Graphics", and it was at The Kelley Engineering Center at Oregon State University. I came into it not knowing a ton about the topic, but I love math and I am fascinated with computer science.

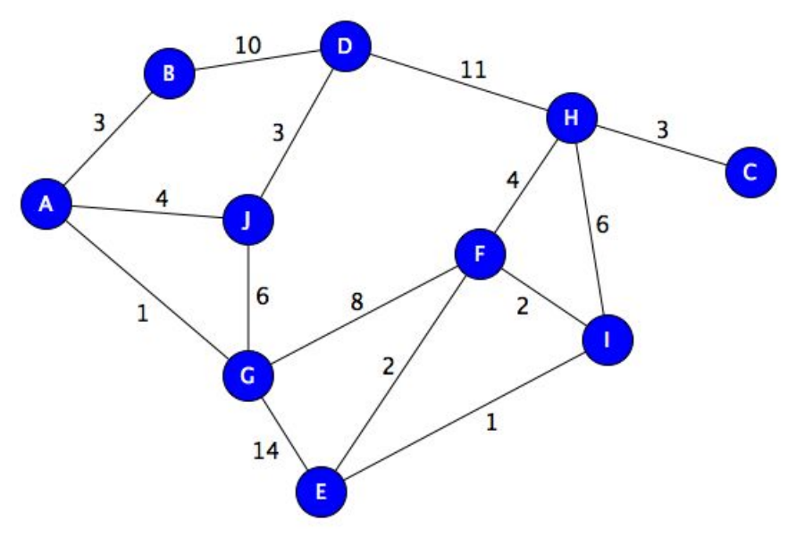
Graph Visualization

Graph visualization is the visual representation of the nodes and edges of a graph. (Linkurious.com) In the pictures below, you can see how nodes and edges are being used to represent area in a shaded 3D object.



Task:
Creating a Graphing Tool

- Make a tool that can create node-edge graphs
- Tool must be able to:
  - Create/remove/merge nodes
  - Contain several settings for nodes
  - Be able to zoom and have several levels of abstraction

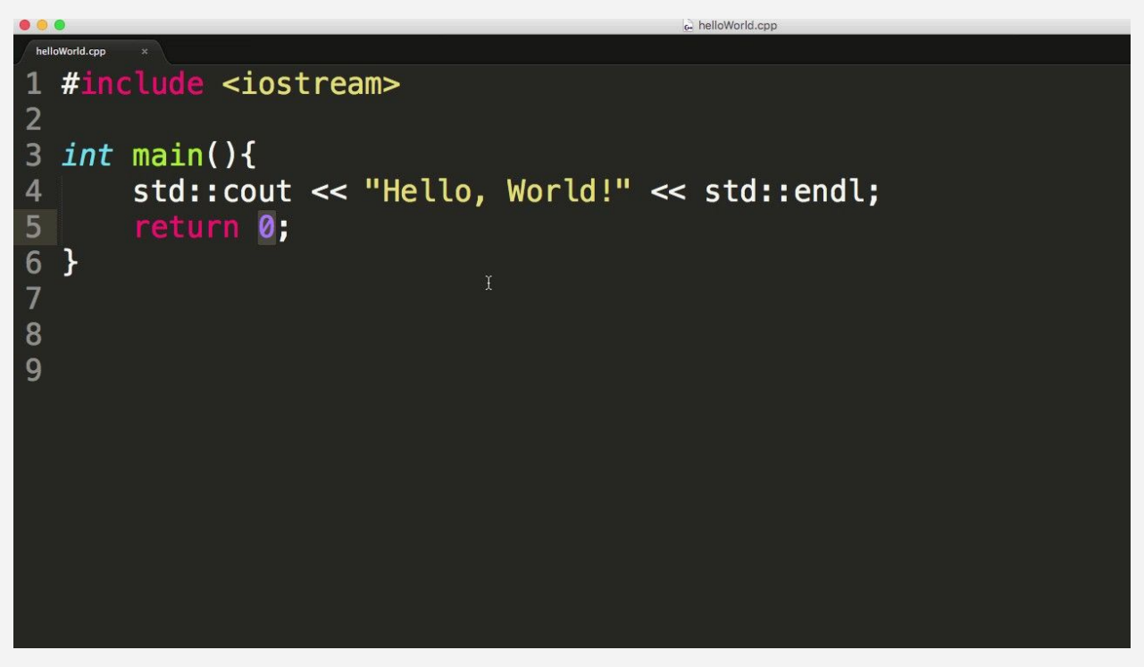


Methods

- Learning C++
- Researching OpenGL
- Researching Shaders
- Learning How to use Qt
- Designing an Interface
- Implementing C++ Graphing Libraries

Learning C++

C++ is a general-purpose programming language. I learned C++ from several online tutorials in both written form and from Youtube.



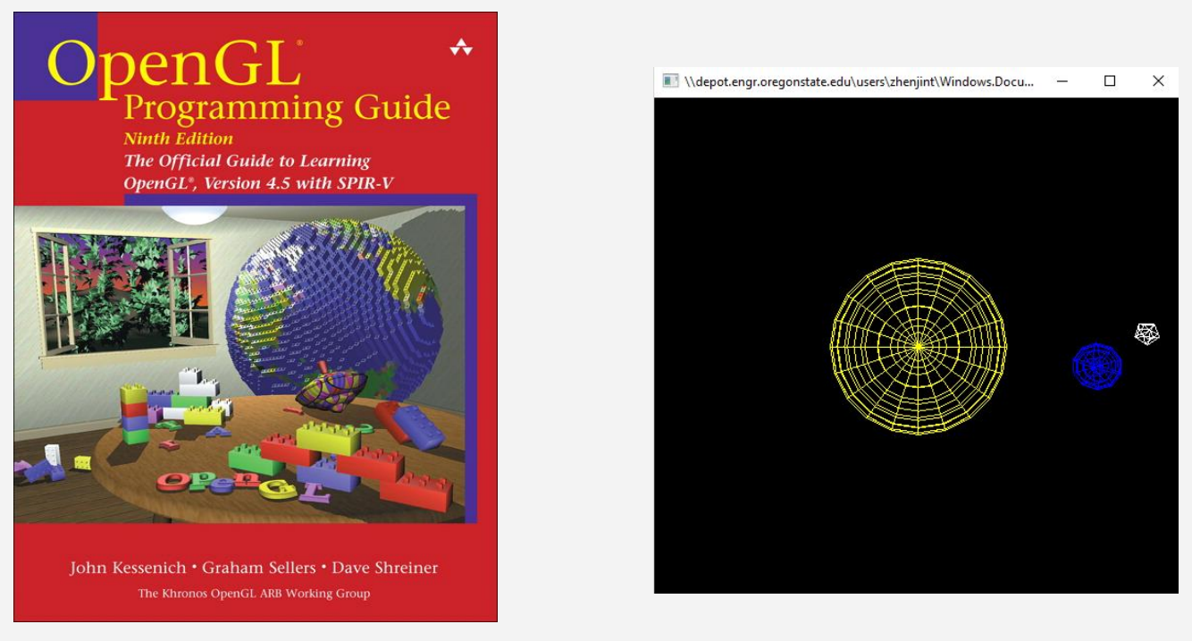
helloWorld.cpp

```cpp
#include <iostream>

int main(){
    std::cout << "Hello, World!" << std::endl;
    return 0;
}
```

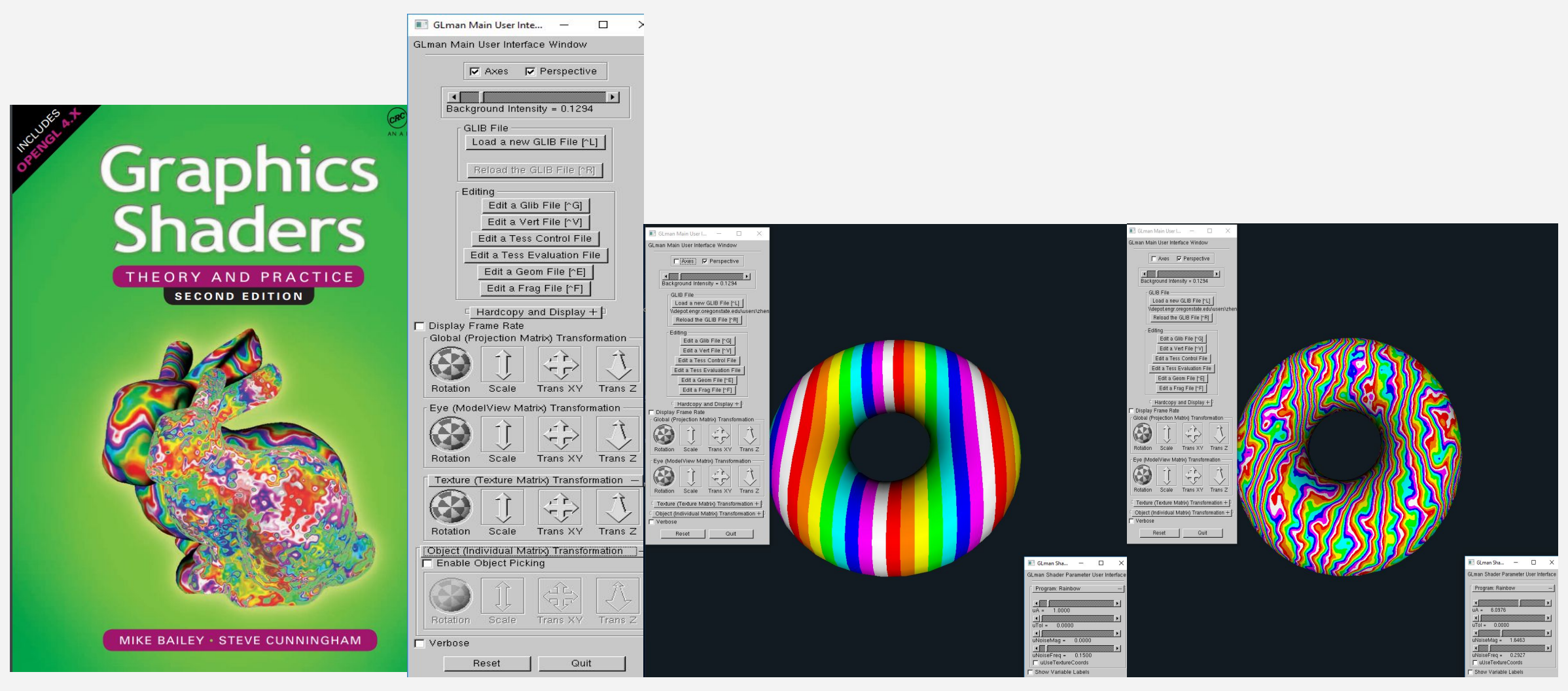
Researching OpenGL and Shaders

OpenGL- An application programming interface used for rendering 2D and 3D vector graphics. It uses several libraries to create the different shapes. The most important skills to learn are how to create shapes, transform the shapes, and, most importantly, creating and using the camera. I learned OpenGL by reading the OpenGL Programming Guide. Here is an example of a scene that the book gave. It originally showed two spheres, one orbited around the other, and then I added a smaller sphere to orbit around the medium sized sphere and also changed their color.


OpenGL Programming Guide
Ninth Edition
The Official Guide to Learning OpenGL, Version 4.5 with SPIR-V

Shaders-Shade and change the appearance of a 3D object. I used a tool called GLMan to be able to change the shader and the scene without closing the scene and manualing inputting a transformation. The shader pipeline is shown below, which demonstrates how all of the types of shaders interact.

I learned about shaders from reading the "Graphics Shaders: Theory and Practice" textbook shown below. An example of a shader is shown in the two pictures of the donut shape. One has rainbow stripes on it, but the other one has some variables changed (sliders shown bottom right corner) which changes its ultimate appearance.

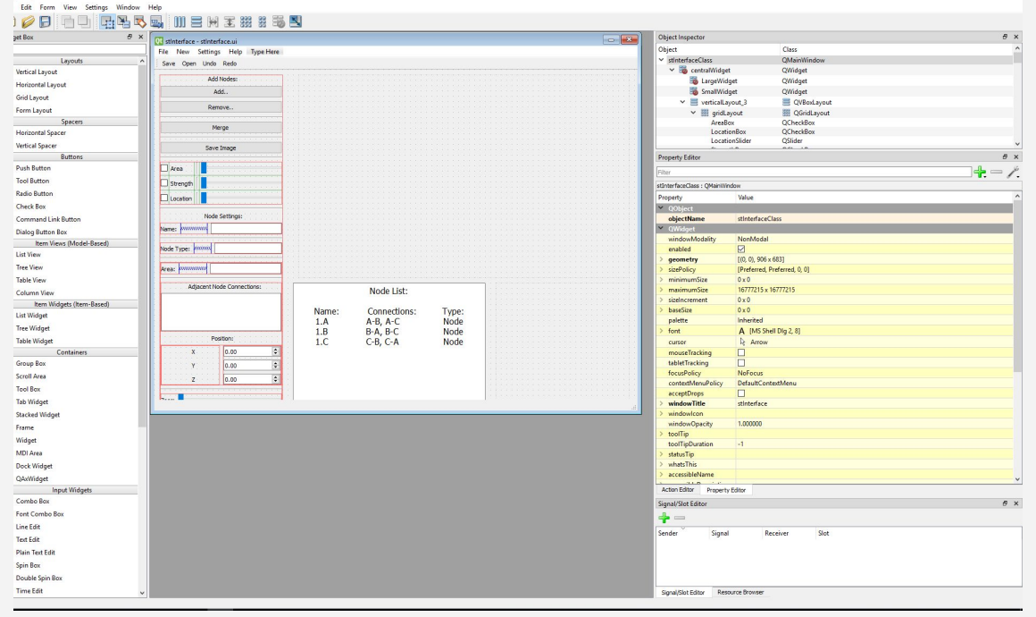


Designing the Interface

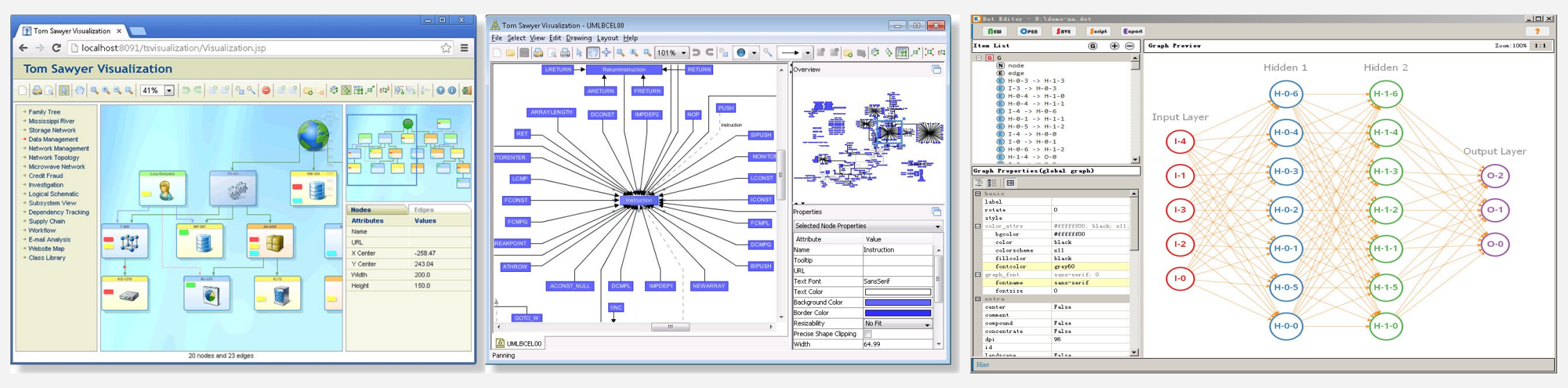
Steps:

- Look at Other Interfaces
- Decide Which Functions I Need
- Design My Own Interface in Qt

Qt is an application framework and widget toolkit used for creating embedded graphical user interfaces. I used the Qt designer to design the interfaces, and the "Signal and Slots" functionality to add functions to the different projects. I learned Qt through a series of Youtube tutorials and experimenting with the different widgets in the designer. The picture shown is the interface of the Qt designer.



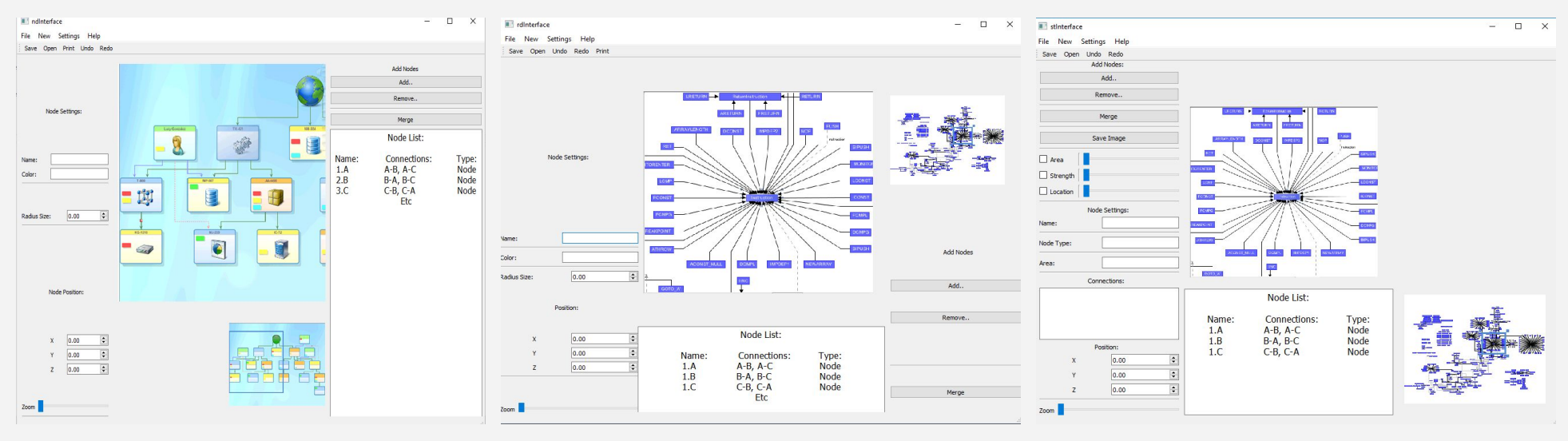
Looking at Similar Interfaces



Looking For:

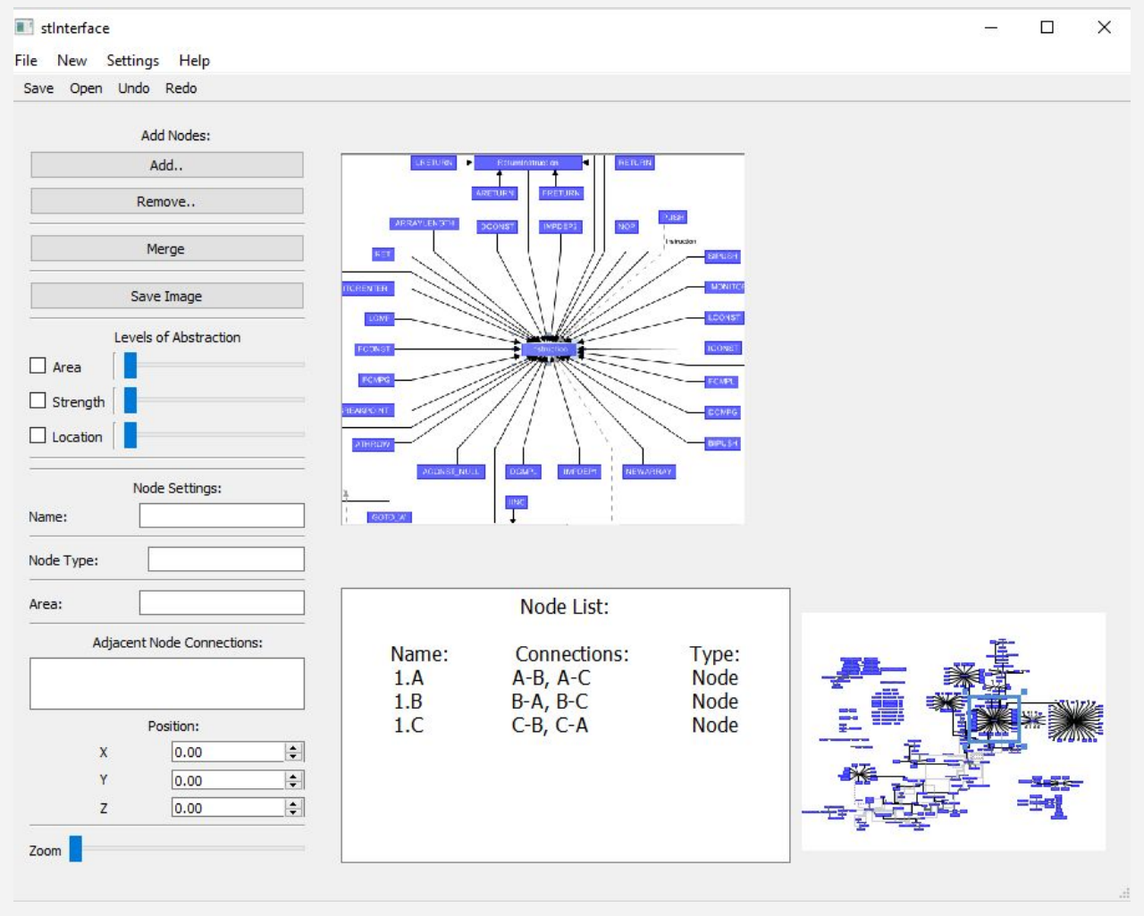
- Functions
  - What each interface includes in order to output a graph
- Windows
  - What each window displays
- Layouts
  - How it places each part in the interface

Designing My Own Interface



- Each are built with the same sets of buttons and widgets
- Pictures taken from the interfaces that I researched are where the graph will be

My Interface



Add Nodes:

Levels of Abstraction

☐ Area
☐ Strength
☐ Location

Node Settings:

Name:

Node Type:

Area:

Adjacent Node Connections:

Position:

X 0.00
Y 0.00
Z 0.00

Node List:

Name:	Connections:	Type:
1.A	A-B, A-C	Node
1.B	B-A, B-C	Node
1.C	C-B, C-A	Node

- **Functions**
  - Add
    - Add- Adds a node where the mouse is clicked
    - Remove- Removes a node that is clicked
    - Merge- Merges two selected nodes
    - Save Image- Saves both images shown on both widgets
  - Level of Abstraction Sliders- Changes the different levels of abstraction in the large widget
  - Settings-
    - Name- Shows a selected nodes name
    - Type- Shows a selected nodes type
    - Area- Shows a selected nodes area
    - Adjacent Node Connections- Shows the connections that a selected node has to other nodes
  - Widgets
    - Large Widget- Allows editing and shows the nodes names and their connections
    - Small Widget- Shows the entire graph with a box around what you are viewing in the large widget
  - List- Shows a list of all nodes and their connections
  - Zoom- Controls how zoomed in the large widget is. Zooming out combines nodes in groups into one nodes and does the opposite while zooming in.

Researching and Implementing Graphing Libraries

Stanford Network Analysis Platform (snap)-

Boost-

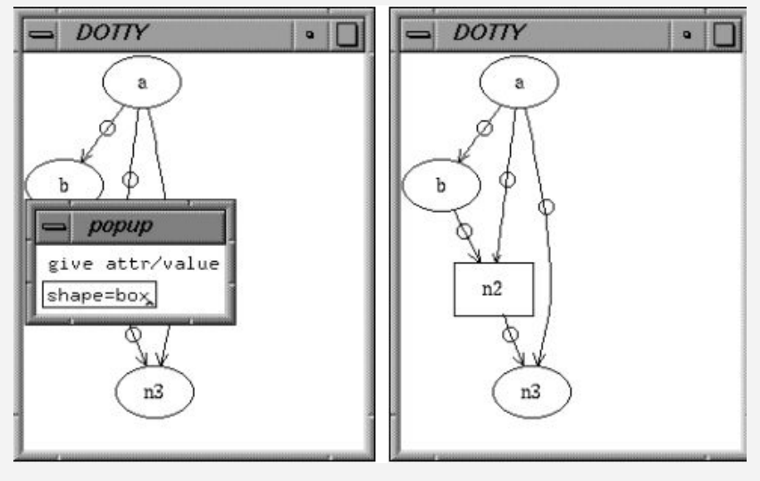
Igraph-

Graphviz-

Tulip-

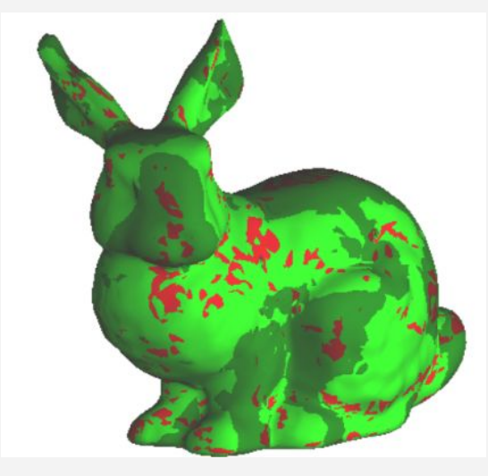
Implementing C++ Graphing Libraries

I am now trying to implement a tool that is a part of the Graphviz library called dotty. It allows a user to click to create a node and drag the middle mouse button between two nodes to create an edge. The dotty interface is shown below.



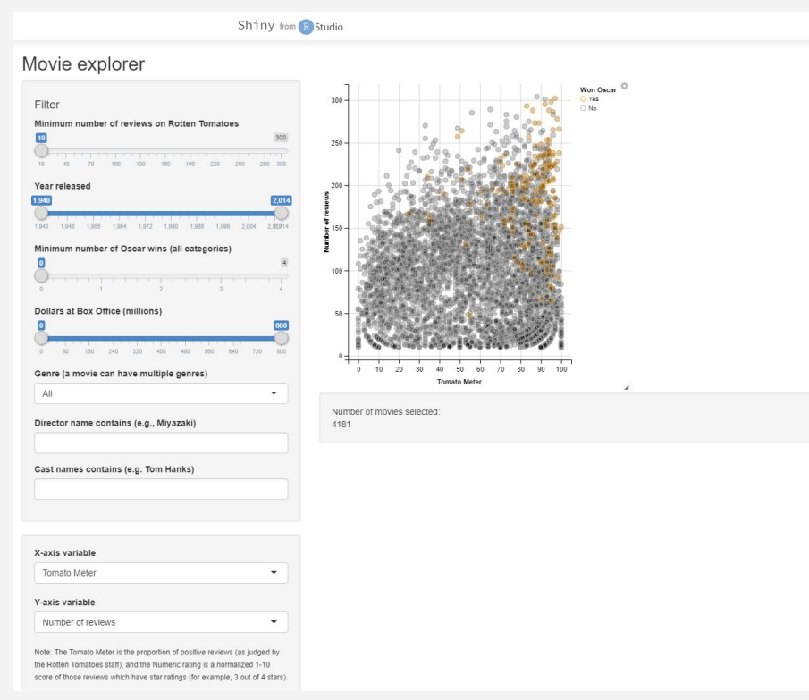
For the Rest of the Internship

For the remainder of the internship, I'll continue to work to get as far as I can in the tool in the limited time that I have left. I want to finish adding functions to the tool, and maybe try to add a shader to a 3D object to visualize data.



The Future of Graph Visualization

The invention of computers and the internet has revolutionized data visualization because of the large amount of data, graphs that can be animated, and graphs that can now be interactive. (Medium.com) This example allows a user to change the graph's data depending on several variables. This tool can be found at http://shiny.rstudio.com/gallery/movie-explorer.html



Acknowledgements

- Mentors:
  - Dr. Yue Zhang-School of EECS
  - Dr. Eugene Zhang-School of EECS
  - Fariba Khan-School of EECS
  - Jinta Zheng-School of EECS

ASE

- **Teacher Monitor-** Cara Benfield

Oregon State University

References:

- Brown, Nick. "The Past and Future of Data Visualization – Nick Brown – Medium." *Medium*, Augmenting Humanity, 21 Sept. 2015, medium.com/@uptownnickbrown/the-past-and-future-of-data-visualization-d58a4b0af6b0.
- "Graph Visualization: Why It Matters." *Linkurious*, 15 May 2018, linkurio.us/blog/why-graph-visualization-matters