

SBOLExplorer

A visualization of the Synthetic Biology Open
Standard Library

Russell Kennington
Jiahui Chen

Table of Contents:

SBOLEplorer - Project Proposal	3
Background and Motivation	3
Visualization Design	4
Prototypes	6
Project Schedule	8
SBOLEplorer - Process Book	9
Overview and Motivation	9
Related Work	9
Questions Our Project Answers	9
Data	10
Exploratory Data Analysis	11
Design Evolution	12
Implementation	15
Progress	16
Evaluation	26
Mock-Up Diagrams	28

SBOLExplorer - Project Proposal

Russell Kennington - u0946645 - arussellk@gmail.com

Jiahui Chen, u0980890 - jiahui.chen@utah.edu

<https://github.com/arussellk/dataviscourse-pr-sbolexplorer>

Background and Motivation

This project is motivated by the research work of Michael Zhang, a mutual friend and student at The U. The data we are visualizing is a hierarchy of synthetic genetic data. The Synthetic Biology Open Standard (SBOL - <http://sbolstandard.org/>) defines a data format which describes genetic building blocks and their relationships. SBOLExplorer will provide a way to view and navigate SBOL data to give researchers deeper understanding and more efficient discovery of their data.

Project Objectives We want to answer:

What subparts compose this genetic part?

An SBOLExplorer user can search for a part like “BBa_E0041”, see relevant search results, click on a result, and view “BBa_E0041” and its child components. The ability to quickly search and navigate SBOL data will help researchers make better use of their SBOL data.

How do these subparts work together to make this genetic part?

SBOLExplorer’s hierarchical tree will have a genetic sequence visualization which shows where child parts fit together to make a genetic component. Visualizing the size and location of child components in the parent genetic part could help researchers see new relationships.

What is the detailed information for this genetic part?

SBOLExplorer provides both a tooltip and an expanded view for genetic part details.

Providing detailed data about a genetic part will let researchers make full use of their SBOL data.

Data.

Our data will be provided by Michael Zhang. Similar data can be found through SynBioHub (e.g. https://synbiohub.org/search/BBa_E0041). If necessary, we will export static JSON files instead of working against a live API.

Data Processing.

We do not expect to perform substantial data cleanup. We may need to manually label the start and stop indices of child components in relation to parent sequences.

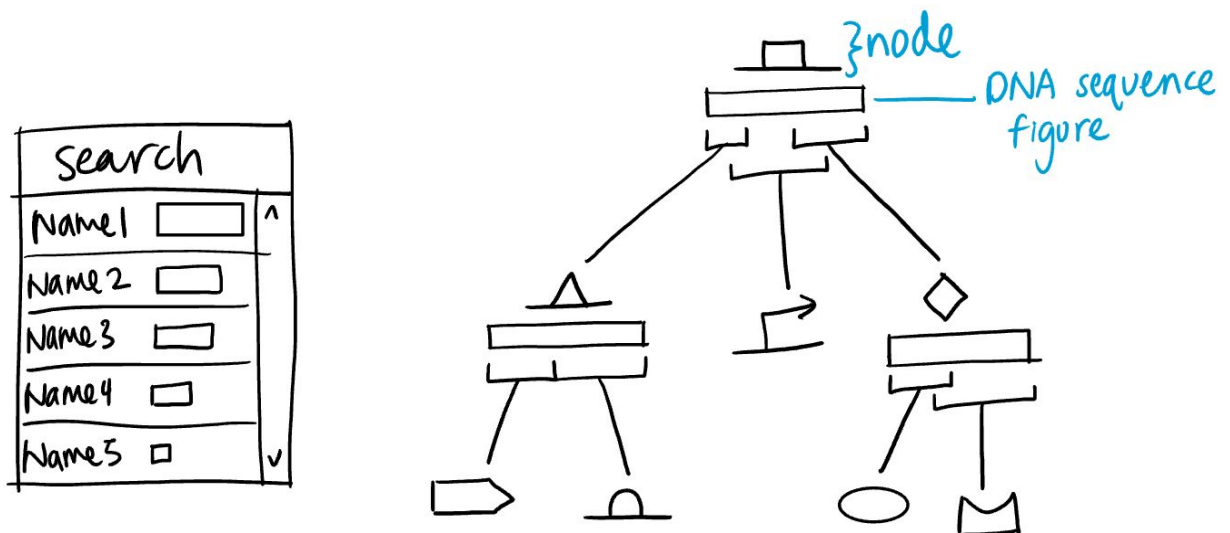
For example:

```
{  
  sequence: "AAAABBBBCCCC"  
  children: [  
    {  
      sequence: "AAAA",  
      parentStart: 0,  
      parentStop: 4,  
    },  
    ...  
  ]  
}
```

Visualization Design

We will have a main visualization consisting of a tree with each node representing a genetic part. Each node will be encoded with a quantitative channel that indicates its popularity (page rank). Each edge will represent a dependency: the parent node's DNA sequence includes each child node's DNA sequence. Each node will also show a figure that represents its DNA sequence, and this figure is marked according to which portions of it consist of its children's DNA sequences. The tree will display data dictated by a search dialogue box that searches for genetic parts. Here's a sketch of a broad overview:

Overview:

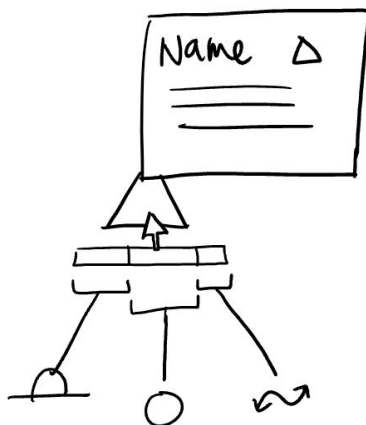


Additional views include:

Popularity indicator (page rank) within the search box, one bar for each genetic part in the search results. This lets users quickly discern if SBOLExplorer has found relevant results and how many of the results are worth viewing.

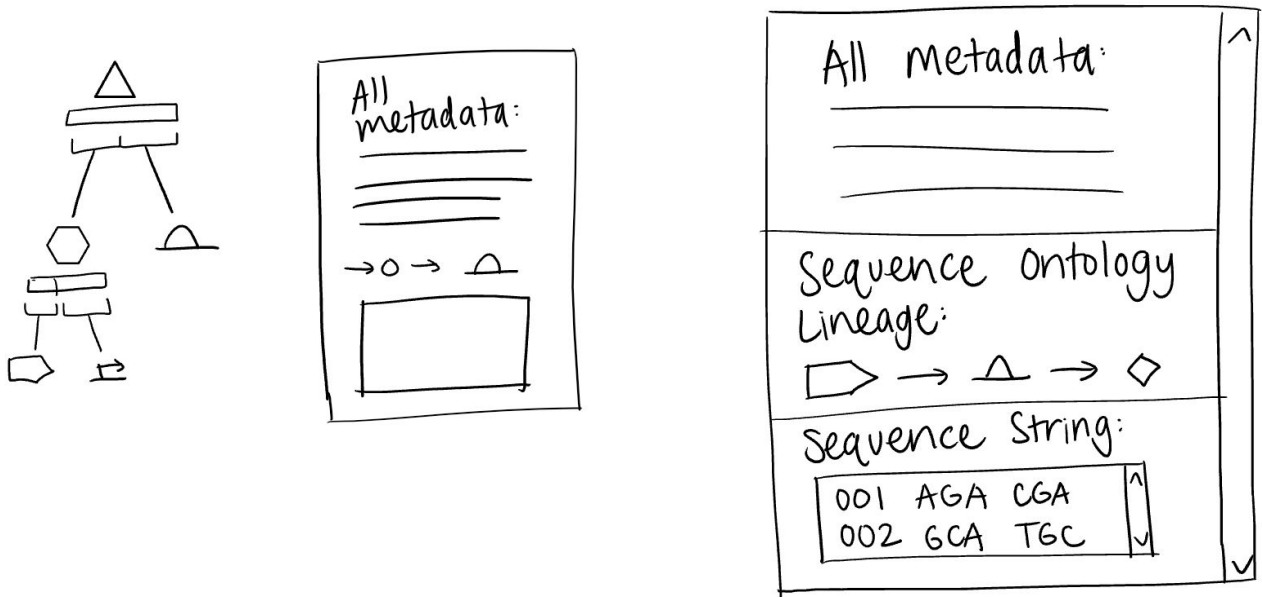
search	
Name1	<input type="checkbox"/>
Name2	<input type="checkbox"/>
Name3	<input type="checkbox"/>
Name4	<input type="checkbox"/>
Name5	<input type="checkbox"/>

Basic info box that displays when hovering over any node/genetic part. The contents of this tooltip may change as we get feedback from future SBOLExplorer users.



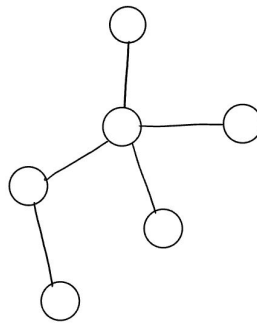
Name	
Description:	
<hr/>	
<hr/>	
<hr/>	

Thorough info box showing all of a genetic part's metadata when the genetic part's node is clicked.



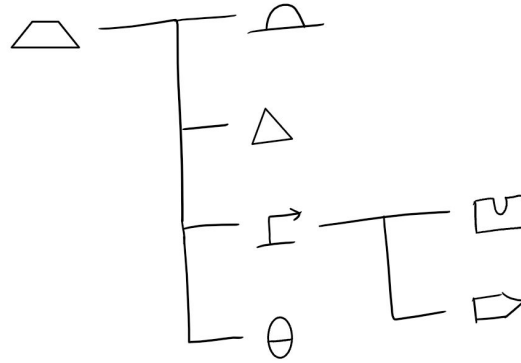
Prototypes

Graph - The initial idea was to display the data in a simple graph where each node represented a genetic part and each edge represented a DNA sequence dependency:

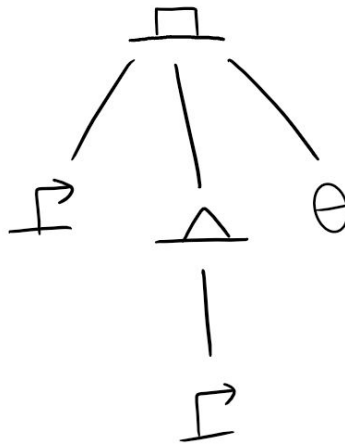


We decided this visualization doesn't convey the most information for this data, as the dependencies should be shown in a more hierarchical manner and should have more influence on the positioning of the nodes.

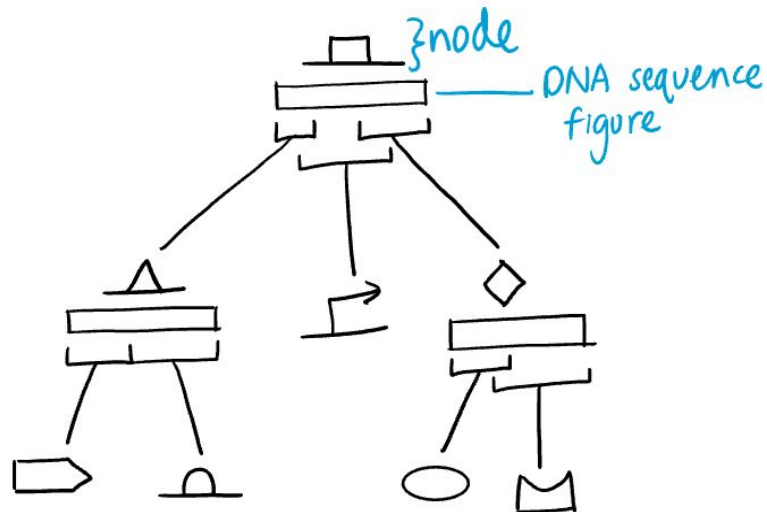
Left to Right Tree - Inspired by filesystem hierarchies, we moved to a left-to-right tree visualization to show dependencies more clearly. We also decided to use the glyphs for each node (the glyphs are part of a field-wild system of symbols for genetic parts and given in the data) instead of one shape for all nodes to show more information on first glance.



Top Down Tree - We decided to do a top-down tree instead of a left-to-right tree because according to Michael a top-down tree would be more readily understood by biologists.



Final visualization design: Top down tree with DNA sequence figure - this is the same as the top-down tree, but each node also has a horizontal bar figure representing the genetic part's DNA sequence. This figure will indicate what partitions of the DNA sequence consist of its children's DNA sequences. We decided to add this DNA sequence figure because Michael said a visual representation of how the sequence dependencies manifest would be useful and isn't available anywhere currently. We thought this would make our visualization more unique and much more useful.



Must-Have Features.

- Search box with the ability to click on a result and render a tree.
- Tree rendering with minimal tooltip to tell you the name of a node.
- Full details in detail view on click.

Optional Features.

- SBOL glyphs as nodes in tree.
- DNA sequence visualization.
- Tree expand/collapse.
- Search relevance horizontal bars.
- Improved tooltip on hover.

Project Schedule

- Week of Oct. 29: Project Setup, Data Gathering
- Week of Nov. 5: Search, Basic Tree Display,
- Week of Nov. 12: Hover and click boxes, tree sequence figures
- Week of Nov. 19: Polish Tree, Tree Collapsing
- Week of Nov. 26: (Nov. 30 due) Buffer

SBOLExplorer - Process Book

Russell Kennington - u0946645 - arussellk@gmail.com

Jiahui Chen, u0980890 - jiahui.chen@utah.edu

<https://github.com/arussellk/dataviscourse-pr-sbolexplorer>

Overview and Motivation

The Synthetic Biology Open Standard ([SBOL](#)) defines a data format that represents synthetic genetic building blocks, which may be combined to create genetic constructs or larger, parent synthetic genetic building blocks. This project, SBOLExplorer, aims to visualize this dataset of synthetic genetic parts as well as the inheritance relationships between them. SBOLExplorer will provide a platform to view and navigate SBOL data to give researchers a deeper understanding of genetic building block relationships, a way to explore the composition of certain genetic parts, and a more efficient discovery of their data.

Related Work

SBOL Standard Data Format: <http://sbolstandard.org/>

SynBioHub: <https://synbiohub.org/>

A searchable repository of the SBOL genetic parts and designs comprised of different combinations of them.

SBOLGraph Library: <https://github.com/udp/sbolgraph>

A library that manipulates SBOL data in a graph form, we use it to get our visualization data.

Questions Our Project Answers

These questions originally motivated our visualization designs as well as dictated what visualizations we used:

What subparts compose the selected genetic part?

How do these subparts work together to make this genetic part?

What is the detailed information for this genetic part?

After the peer review of the project, we had the additional question:

Would be useful to be able to download a genetic part's data from the visualization?

The answer is no, so we didn't add anything to our visualization that would enable a download.

Data

Our data is sourced from the API that SynBioHub's data is sourced from. An example data point can be found through SynBioHub (e.g.

https://synbiohub.org/public/igem/BBa_K1407008/1).

There are two forms of data we will be working with:

Search result data

Tree data

The **search result data** can be obtained through SynBioHub, though we have already saved a sample search result for offline use. The search data looks like this:

```
[
  {
    "_id": "104450",
    "_index": "part",
    "_score": 0.0025428662,
    "_source": {
      "description": "green fluorescent protein derived from jellyfish
Aequorea victoria wild-type GFP (SwissProt: P42212",
      "displayId": "BBa_E0040",
      "graph": "https://synbiohub.org/public",
      "keywords": "BBa E0040",
      "name": "GFP",
      "pagerank": 0.00031315985719595617,
    },
  },
  ...
]
```

Notice that there is a score attribute on a search result. This will be used to create the relevance bar in the search results list view.

The **tree data** is received through the <https://github.com/udp/sbolgraph> library, which queries the API for a specific tree and returns a graph object for each genetic part that's queried. Some data processing has been implemented to make the returned graph better for our purposes. Additionally, we have already saved sample trees for offline use. A sample tree looks like this:

```
{
  "uri": "https://synbiohub.org/public/igem/BBa_B0030/1",
  "sequence": "attaaagaggagaaa",
}
```

```
"children": [],  
"range": [63, 77]  
},
```

Note that a tree has a range attribute consisting of a start and end value. This range indicates where this node fits into the sequence of its parent node. The range will be used to create the subsequence indicators in our tree.

More data processing may be needed as we discover ways our visualization can be improved. We are using TypeScript and have good knowledge of the library which means we can quickly pull in data which we need.

Exploratory Data Analysis

There's currently no visualizations of SBOL data and we did initial data analysis by consulting Michael Zhang, who's thesis project is the creation of SynBioHub. He knows the data well because he's worked with it extensively and also belongs to a research group that is heavily involved with the creation and maintenance of SBOL.

We gained crucial information about the size and scope of genetic parts which is important to visualization decisions regarding navigation and information density:

- Average number of levels of inheritance for a genetic part:

 - 3, maximum is 6

- Average number of child genetic parts that a parent genetic part is comprised of:

 - 3, maximum is 10 but when the number of child parts is high the total inheritance levels are low.

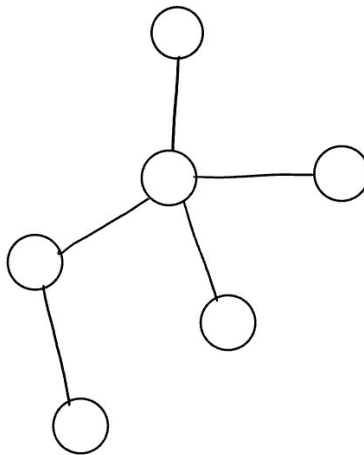
Knowing these 2 facts led us to decide to not include navigation in our visualization as the information for each genetic part will not be incredibly dense. We think that scaling the visualization will be enough to show the information for all genetic parts.

Design Evolution

We did not deviate from the project proposal. Below is the evolution of our main visualization's design and all the additional visualizations we will include.

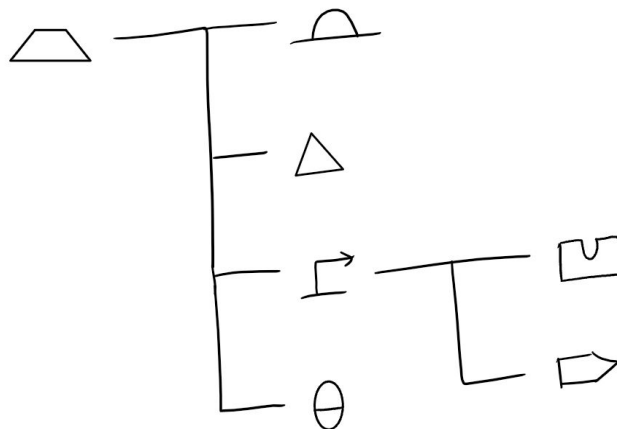
Main Visualization:

Prototype 1: Because the most important aspect of our data is the relationship between data points, we initially thought of a graph visualization for our main visualization of genetic parts where nodes represent genetic parts and edges represent inheritance relationships between them:

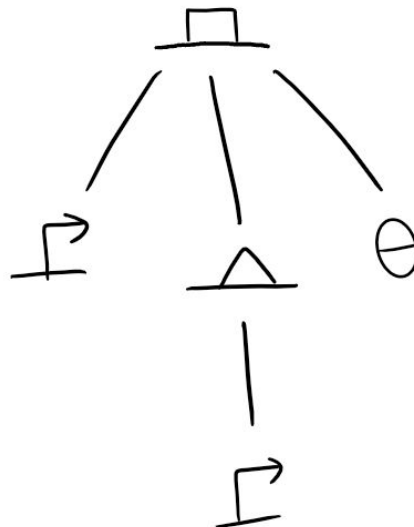


We decided a graph visualization would not explicitly show inheritance/dependency relationships, so we moved on to consider tree visualizations.

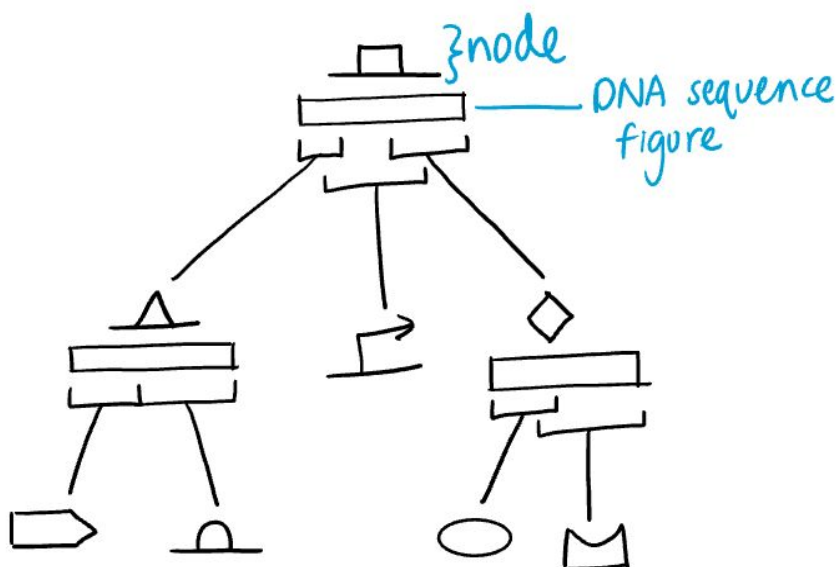
Prototype 2: Left to Right Tree - We first considered a left-to-right tree visualization to show dependencies more clearly. We also decided to use the glyphs for each node (the glyphs are part of the SBOL standard and indicate a certain type of genetic part) as the marks instead of the same shape for all nodes to show more information on first glance.



Prototype 3: Top Down Tree - We decided to do a top-down tree instead of a left-to-right tree because according to Michael a top-down tree would be more readily understood by biologists, and we want our visualization to be as useful and understandable as possible for our target audience.

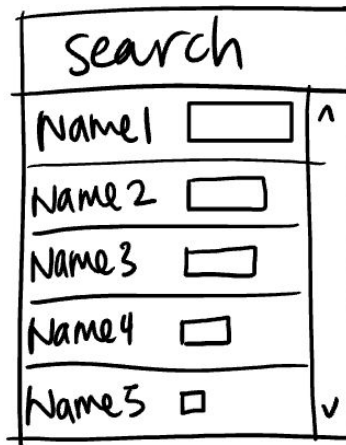


Final main visualization design: Top down tree with DNA sequence figure - this is the same as the top-down tree design, but each node also has a horizontal bar figure representing the genetic part's DNA sequence. This figure will indicate what partitions of the DNA sequence consist of its children's DNA sequences. We decided to add this DNA sequence figure because a visual representation of how the sequence dependencies are structured would be useful to researchers and isn't currently available anywhere. We thought this would make our visualization more unique and much more useful.

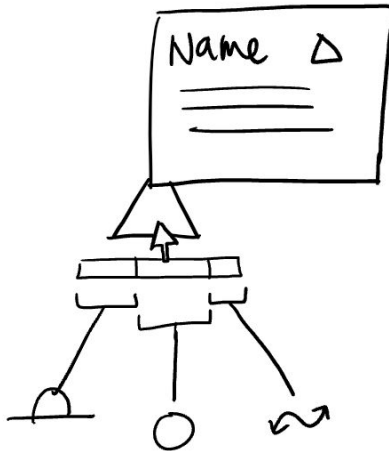


Additional Visualizations:

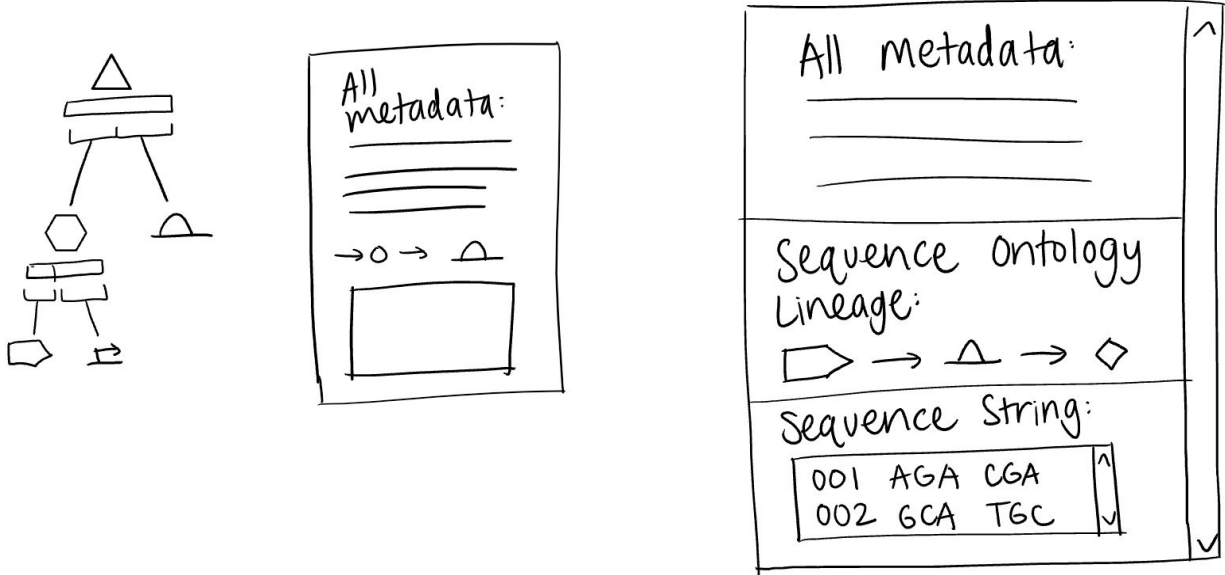
Search Box with Popularity Indicator - Our visualization will have a search functionality, where a string can be input and the search results will be all the SBOL parts that the query returns. Within the search results display box, there will be one bar for each genetic part in the search results indicating its popularity (page rank, which is included in each data point). This lets users quickly discern if SBOLExplorer has found relevant results and how many of the results are worth viewing.



Basic Info Box - A box that displays when hovering over any node/genetic part. It will show brief information such as full name, the glyph, and version.



Thorough Info Box - This box shows all of a genetic part's metadata when the genetic part's node is clicked.



Implementation

Currently we have implemented:

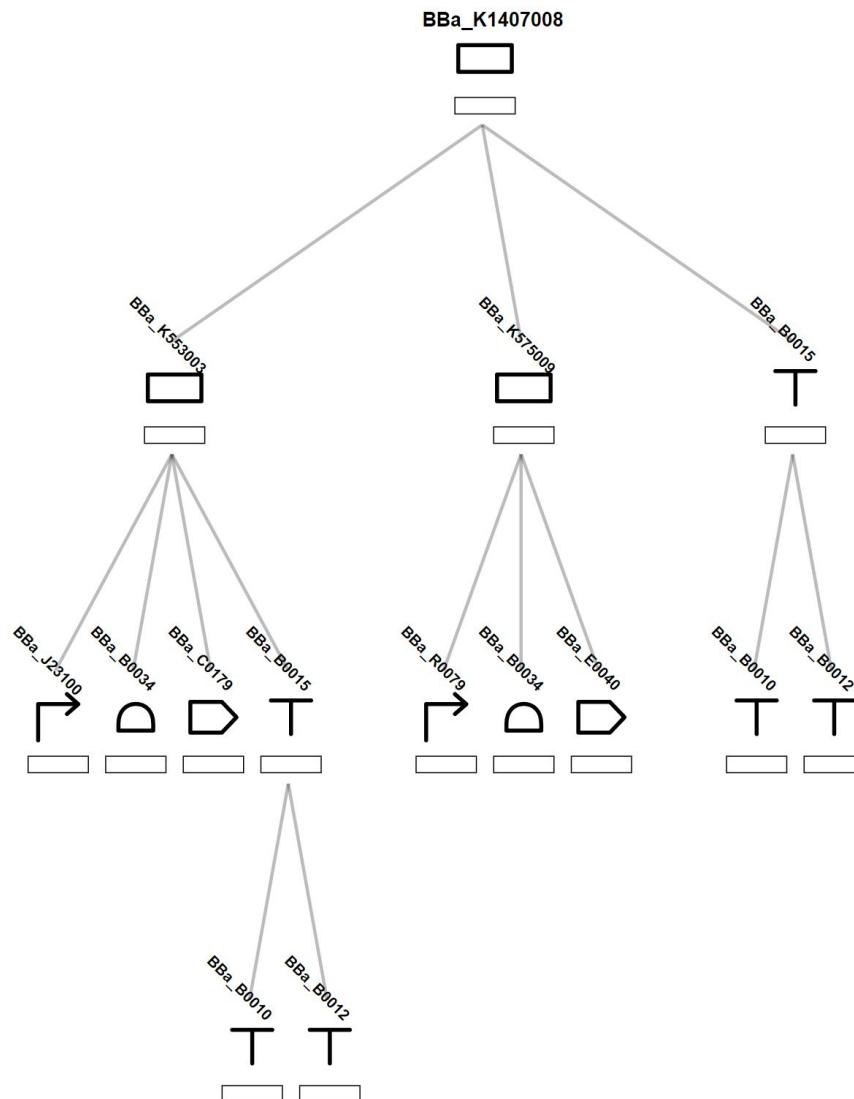
A working project that uses an SBOL library with Webpack
Local Search Result and Tree data as well as code that can query it from the data API

Implementation Plans:

A detailed mock-up of our project plans, including interaction annotations are the last 3 pages of this process book (pages:

Progress

The basic **tree** implementation looks like this:



The most difficult implementation detail of the basic tree was getting all the glyphs to show up, and correspond to each data point. All the glyphs are SVGs in the `/build/glyphs` folder. An empty rectangle under each node represents the DNA sequence visualization.

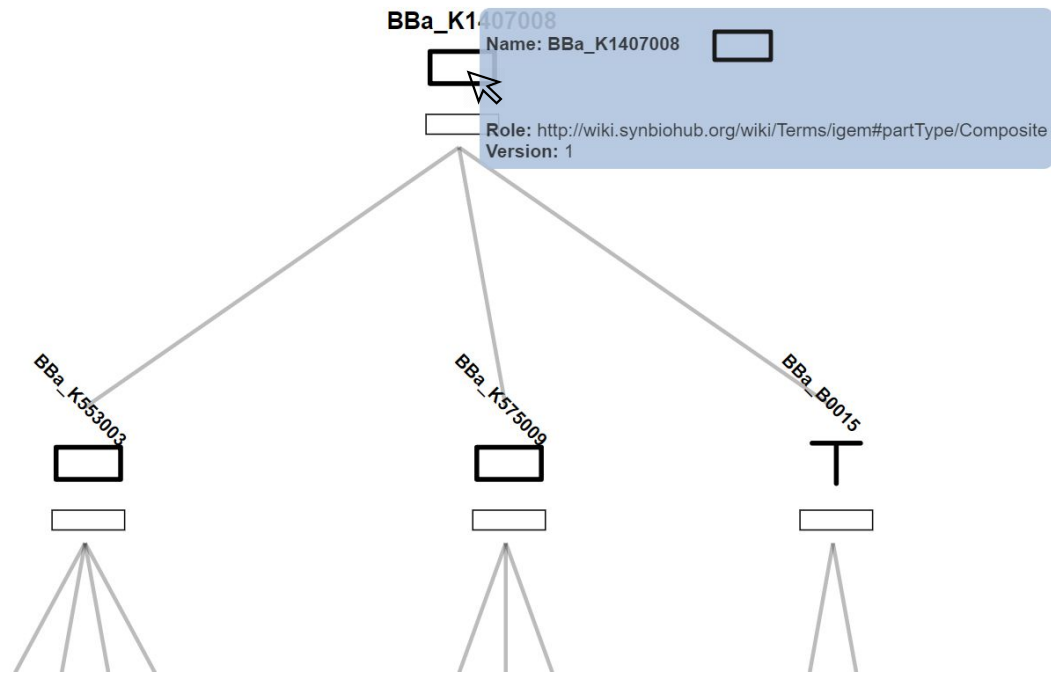
The **search** panel is the leftmost component in the DOM, and looks like this:

Search

BBa_E0040 green fluorescent protein derived from jellyfish Aequorea victoria wild-type GFP (SwissProt: P42212)
BBa_E0240 GFP generator
BBa_E0032 enhanced yellow fluorescent protein derived from A. victoria GFP
BBa_E0840 GFP generator
BBa_E0030 enhanced yellow fluorescent protein derived from A. victoria GFP
BBa_J04450 RFP Coding Device
BBa_E0020 engineered cyan fluorescent protein derived from A. victoria GFP
BBa_K145015 GFP with LVA tag
BBa_E0022 enhanced cyan fluorescent protein derived from A. victoria GFP
BBa_K093005 RFP with RBS

The list of search results changes when a new query is entered. This search box does not include the bars that will visually indicate the popularity (Page Rank) of the search result.

The **hover info box** appears when a user hovers their mouse over the glyph of a node, and looks like this:




This hover box is the finished hover box, and no further implementation will be done involving it.

The **click info box** is the rightmost panel of the DOM and looks like this:

BBa_K1407008

BBa_K1407008
Version 1



Source:
https://synbiohub.org/public/igem/BBa_K1407008/1

Description: constitutive production of LasR+LasR/PAI1
Inducible promoter +RBS(B0034)+GFP+Terminator

Type: <http://www.biopax.org/release/biopax-level3.owl#DnaRegion>

Role:
<http://wiki.synbiohub.org/wiki/Terms/igem#partType/Composite>

Sequence:

```
ttgacggctagctcagtcctaggtacagtgcctagctactag
agaaagaggagaaatactagatggccttggtgacggttt
tcttgagctggaacgctcaagtggaaaattggagtgagc
gccatcctccagaagatggcgagcgaccttgattctcga
gatcctgttcggcctgtgcctaaggacagccaggactacg
agaacgccttcacgtcggaactacccggccgctggcg
cgagcattacgaccgggctggctacgcgagggtcgaccc
gacggtcagtcactgtacccagagcgtagtccgattttctg
ggaaccgtccatctaccagacgcgaaagcagcagagttc
ttcgaggaagcctcgccgcccggcctggtgtatgggctga
```

The box changes when any glyph in the tree is clicked to show the metadata of that node. This info box is the finished info box, and no further implementation will be done involving it.

The overall visualization with the tree at a basic stage looks like this:

SBOLE Explorer

Search

BBa_K1407008

BBa_E0040

green fluorescent protein derived from jellyfish Aequorea victoria wild-type GFP (SwissProt: P42212)

BBa_E0240

GFP generator

BBa_E0032

enhanced yellow fluorescent protein derived from A. victoria GFP

BBa_E0840

GFP generator

BBa_E0030

enhanced yellow fluorescent protein derived from A. victoria GFP

BBa_J04450

RFP Coding Device

BBa_E0020

engineered cyan fluorescent protein derived from A. victoria GFP

BBa_K145015

GFP with LVA tag

BBa_E0022

enhanced cyan fluorescent protein derived from A. victoria GFP

BBa_K093005

RFP with RBS

BBa_B0015

BBa_B0015
Version 1

T

Source: https://synbiohub.org/public/igem/BBa_B0015/1

Description: double terminator (B0010-B0012)

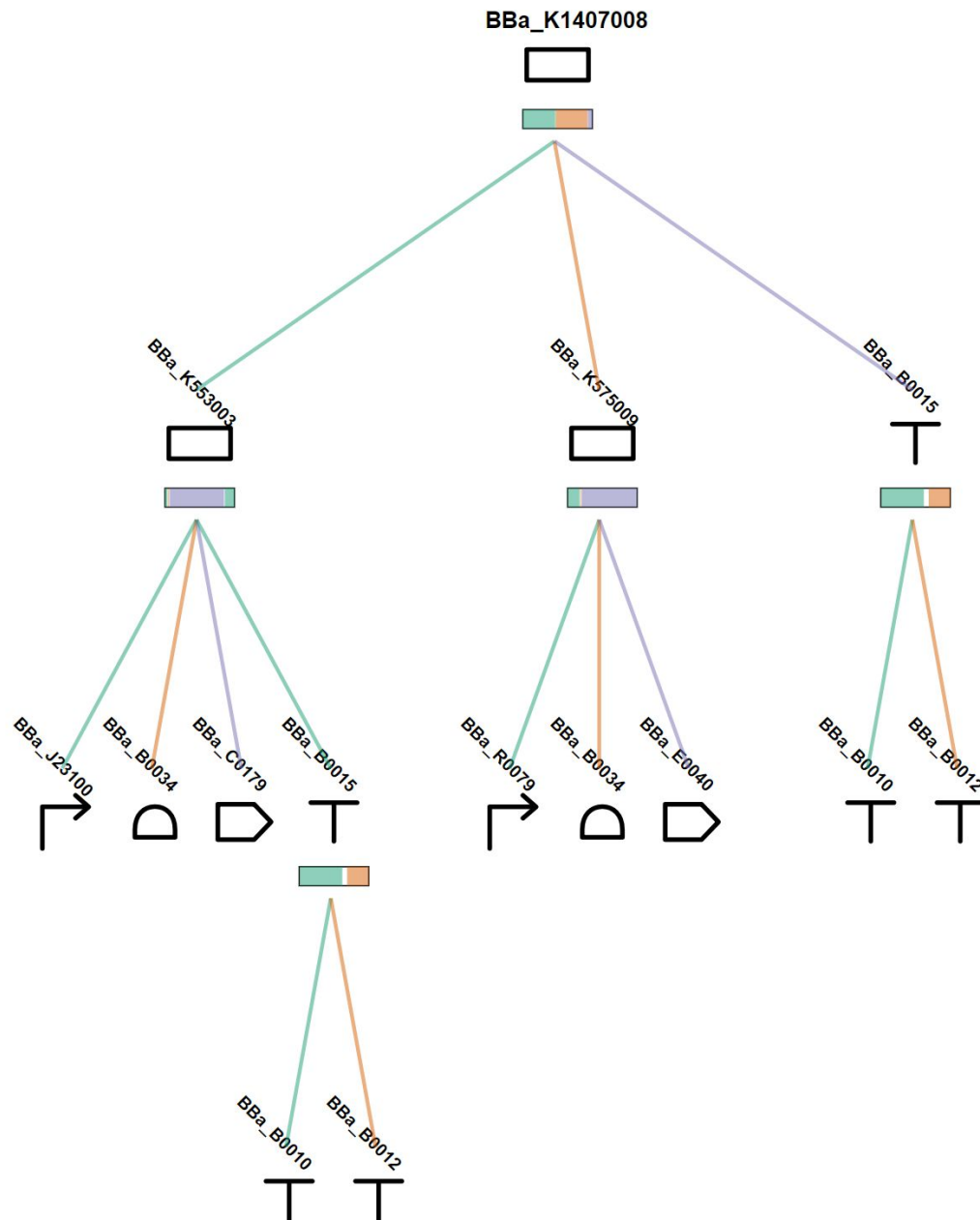
Type: <http://www.biopax.org/release/biopax-level3.owl#DnaRegion>

Role: <http://identifiers.org/sc/0000141>

Sequence:

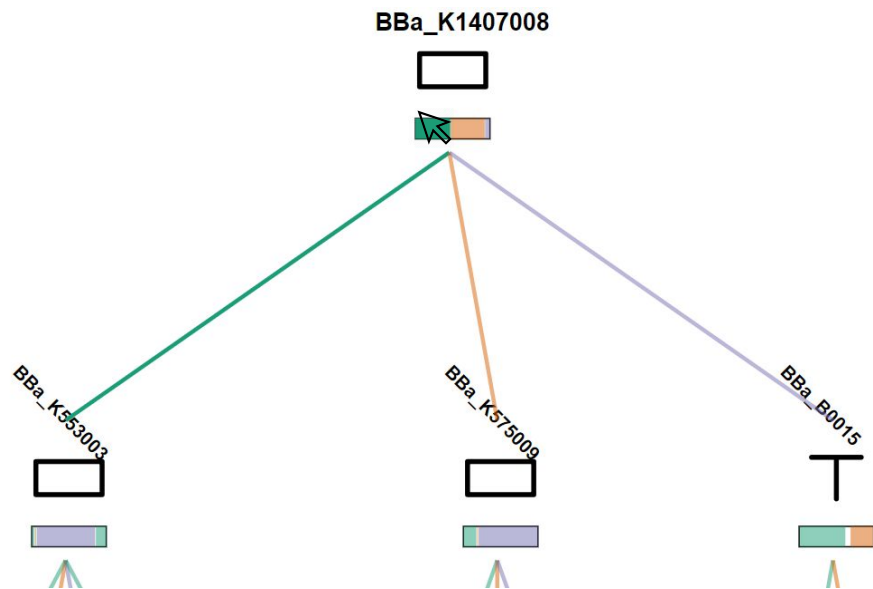
```
ccaggcctcaataaaacgaaggctcagtcgaagactggg
ccttctgtttatctgtgttgcggtaacgctctctactaga
acactgctcaccctcggtggcctttctgcgtttata
```

The **tree** with DNA sequence visualized looks like this:

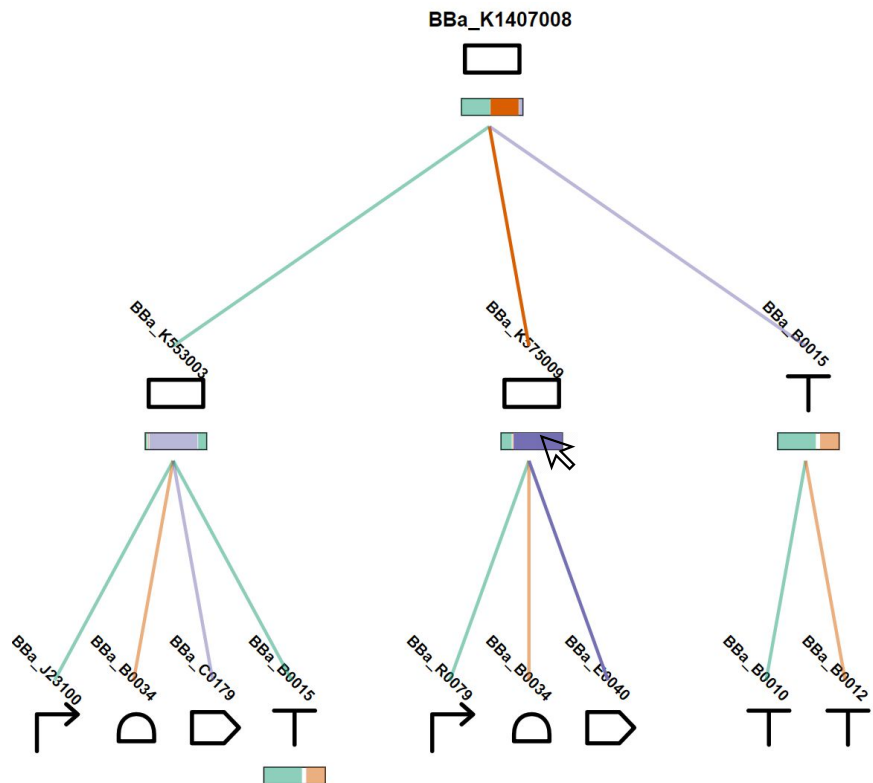


The rectangles under each glyph are colored according to where the children nodes' DNA sequences exist in the parent node's DNA sequence. The white space indicates areas where the DNA codons in the DNA sequence are not any defined genetic part. When the DNA sequences of children overlap in the parent sequence, the colors overlap. The edges are correspondingly color coded to show which children comprise which part.

Edge and rectangle portion highlighting also exists. When a portion of the DNA sequence rectangle is hovered over, it and its corresponding edge are highlighted by the color darkening:



If the region highlighted has multiple parents, then all edges and portions from parent nodes are highlighted:



The **search panel** with popularity (PageRank) visualization looks like this:

Search

gfp	
BBa_E0040 green fluorescent protein derived from jellyfish Aequorea victoria wild-type GFP (SwissProt: P42212)	<div><div></div></div>
BBa_E0240 GFP generator	<div><div></div></div>
BBa_E0032 enhanced yellow fluorescent protein derived from A. victoria GFP	<div><div></div></div>
BBa_E0840 GFP generator	<div><div></div></div>
BBa_E0030 enhanced yellow fluorescent protein derived from A. victoria GFP	<div><div></div></div>
BBa_J04450 RFP Coding Device	<div><div></div></div>
BBa_E0020 engineered cyan fluorescent protein derived from A. victoria GFP	<div><div></div></div>
BBa_K145015 GFP with LVA tag	<div><div></div></div>
BBa_E0022 enhanced cyan fluorescent protein derived from A. victoria GFP	<div><div></div></div>
BBa_K093005 RFP with RBS	<div><div></div></div>

Each search result's rectangle represents its popularity/PageRank. This PageRank is predetermined by SBOLEplorer's database and is a field in each search result's datapoint.

These bars will serve as a visual indicator to how relevant each search result is and can help users decide which tree they want to look at.

Finishing Touches:

The project's website was updated so that the top of the page included links to the process book, video demo, and a project description:

SBOLExplorer

SBOLExplorer - The Synthetic Biology Open Language Explorer

You will create a public website for your project using GitHub pages or any other web hosting service of your choice. The web site should contain your interactive visualization, summarize the main results of the project, and tell a story. Consider your audience (the site is public) and keep the level of discussion at the appropriate level. Your process book and data should be linked from the web site as well. Also embed your interactive visualization and your screen-cast in your website. If you are not able to publish your work (e.g., due to confidential data) please let us know in your project proposal.

The Synthetic Biology Open Standard (SBOL) defines a data format that represents synthetic genetic building blocks, which may be combined to create genetic constructs or larger, parent synthetic genetic building blocks. This project, SBOLExplorer, aims to visualize this dataset of synthetic genetic parts as well as the inheritance relationships between them. SBOLExplorer will provide a platform to view and navigate SBOL data to give researchers a deeper understanding of genetic building block relationships, a way to explore the composition of certain genetic parts, and a more efficient discovery of their data.

Process Book

TODO: insert link to process book here

Data

Our data is sourced from SynBioHub. Through SynBioHub, users can view data about a genetic component by visiting its reference page (8Ba_I5610 example reference). Notice that SynBioHub does not have a way to view multiple layers of a genetic component tree. SBOLExplorer was created to fill this need.

Demo



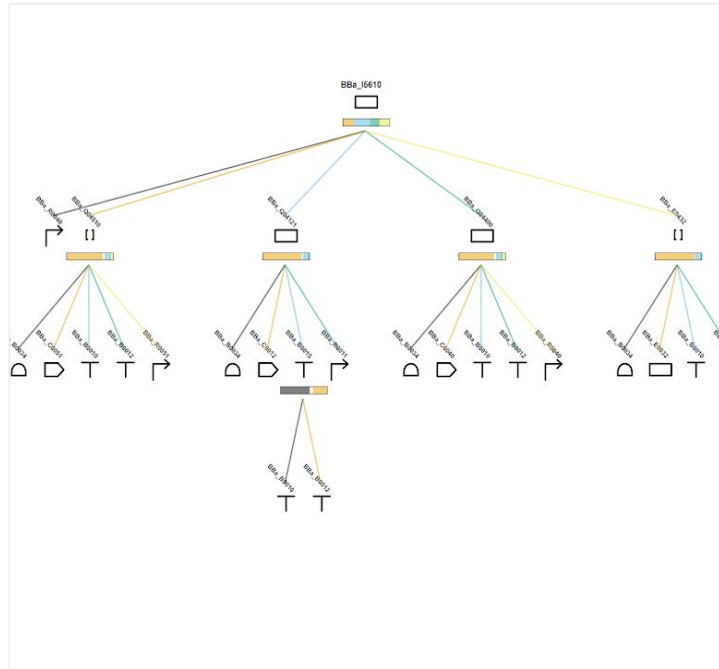
The visualization was also organized into panels, and the tree visualization is zoomable and pannable:

Visualization

Search

rbs

BBa_B0034	<input type="checkbox"/>
RBS (Elowitz 1999) -- defines RBS efficiency	
BBa_J70591	<input type="checkbox"/>
RFC12 Elowitz RBS	
igem_BBa_B0034	<input type="checkbox"/>
RBS (Elowitz 1999) -- defines RBS efficiency	
BBa_I724002	<input type="checkbox"/>
Standard Elowitz Repressilator with degradation tag	
BBa_I5610	<input type="checkbox"/>
Elowitz Repressilator: QPI design	
BBa_I13711	<input type="checkbox"/>
Tet promoter, Elowitz RBS, CheB	
BBa_I724005	<input type="checkbox"/>
Elowitz repressilator with added degradation tag	
BBa_I13721	<input type="checkbox"/>
CheB under Lac control with Elowitz RBS	
BBa_I724006	<input type="checkbox"/>
Standard Elowitz Repressilator with Degradation Tag Addition	
BBa_I13625	<input type="checkbox"/>
Cascaded Lac (medium) and Tet (elowitz) QPIs	



BBa_I5610

BBa_I5610

Version 1



Source: https://synbiohub.org/public/igem/BBa_I5610/1

Description: Elowitz Repressilator: QPI design

Type: <http://www.biopax.org/release/biopax-level3.owl#DnaRegion>

Role: <http://identifiers.org/so/SO:000804>

Sequence:

```

tcctatcagtgatagagattgacatccctatcagtgatagagatact
gagcactactagagaagaggagaaatactagatgacacaaaa
aagaaccattaacacaaagacagcttgagacgcacgicgctt
aaagcaattttgaaaaaaagaaaatgaactggcttatccagg
aatctgtcgcagacaagatgggatgggagcagcagcggttgg
gctttatttaatgcatcaatgattaaatgcttataacgcccattgc
ttgcaaaattctcaagtttagcgttgaagaatttagccctcaatcg
ccagagaatctacgagatgatgaagcggtagtatgcagccgtc
acttagaagtgagtatgagatccctgtttttctcatgttcaggcagg

```

Evaluation

What we learned from our data:

Many genetic parts have a bunch of child data points, but then none of these data points are actual genetic parts so they should not be nodes in our graph. This is because many genetic parts have DNA sequences that are comprised of DNA sub-sequences that are not a defined genetic part. This was misleading when we were initially creating the tree, as some trees had children where data fields that defined the glyph and other attributes were empty yet there was a sequence.

How did you answer your questions?

What subparts compose the selected genetic part?

The tree visualization innately shows which children genetic parts each genetic part/node has. It's clear to see each genetic part's children as there are edges from the parent genetic part to all its children.

How do these subparts work together to make this genetic part?

The DNA sequence visualization that's a part of the tree visualization answers this question. Each rectangle under the genetic part's glyph represents the genetic part's DNA sequence, and has portions of it colored according to what child genetic part's DNA sequence exists there in the parent's DNA sequence. Therefore, the DNA sequence rectangle of each node shows where its subparts are and how much each subpart comprises.

What is the detailed information for this genetic part?

The hover and click info boxes display all the metadata of each genetic part. All details of a genetic part can be accessed by searching for it, displaying the tree at which it's the root, and clicking on its glyph. The click info box also links to the data source on SynBioHub, which has even more links and extended information about the part.

How well does your visualization work, and how could you further improve it?

Our visualization works well. It could be improved by adding tree expansion and collapsing to handle extremely large trees. Currently, the dataset we're working with doesn't have trees large enough to warrant that.

We could add more dynamic scaling of glyphs and DNA sequence rectangles. For example: the more nodes in a row, the smaller sized the glyph and rectangle become.

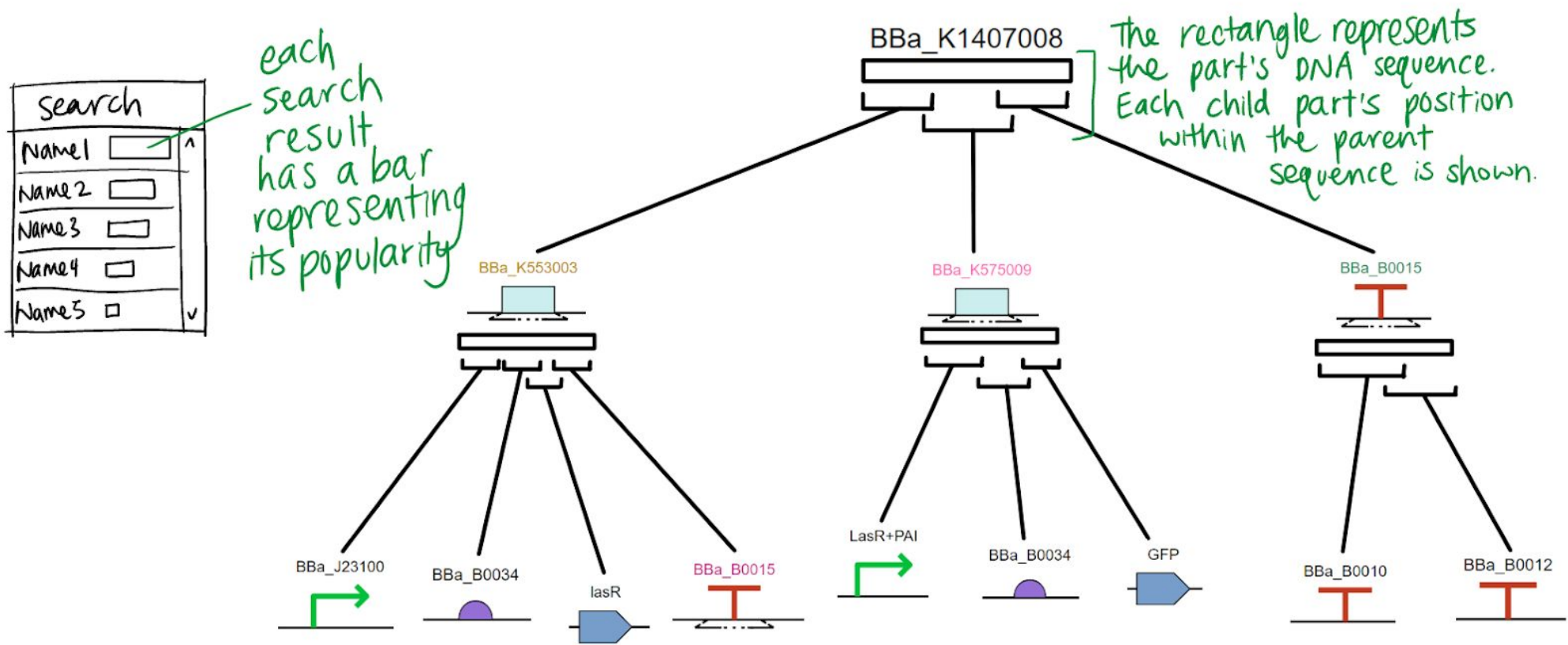
A really great addition to this visualization would be a magic lense on the DNA sequence rectangles, where when hovered over the DNA sequence rectangles will expand and show the labeled regions of where each child sequence exists in the parent sequence.

Overall Visualization Goal:

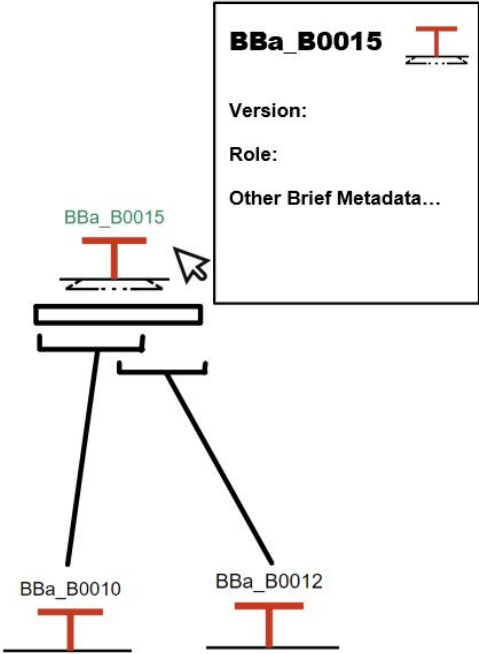
A tree of the data point/genetic part: https://synbiohub.org/public/igem/BBa_K1407008/1

The JSON of this data point is located at:

https://github.com/arussellk/dataviscourse-pr-sboexplorer/blob/master/src/data/trees/BBa_K1407008-with-range.json



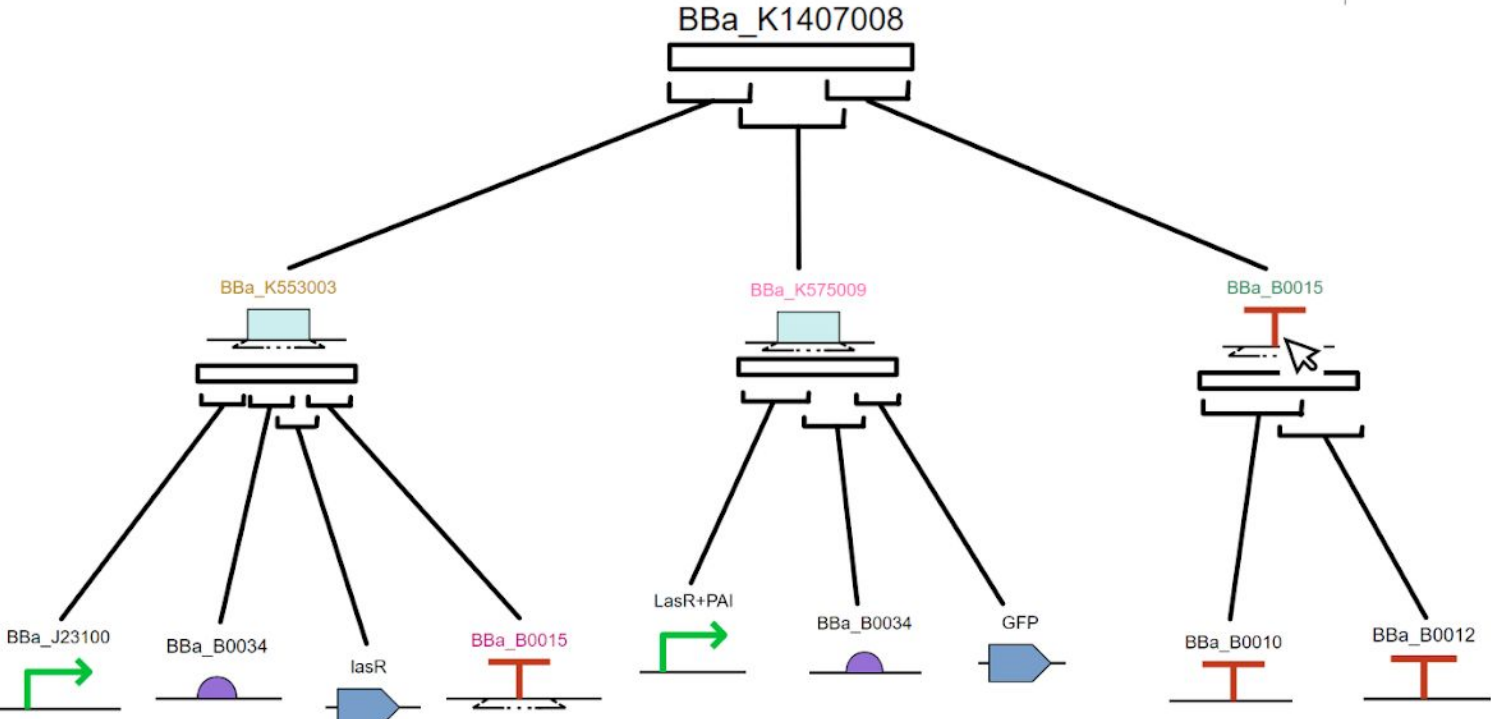
Tree Interaction: Hover



When a node in the tree is hovered over, an infobox containing brief metadata shows up.

Tree Interaction: Click

When a node in the tree is clicked on, a large, scrollable info box showing all of the genetic part's metadata appears on the page



BBa_B0015

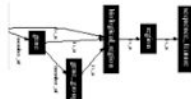
Version:

Role:

All Metadata...

Sequence Ontology:

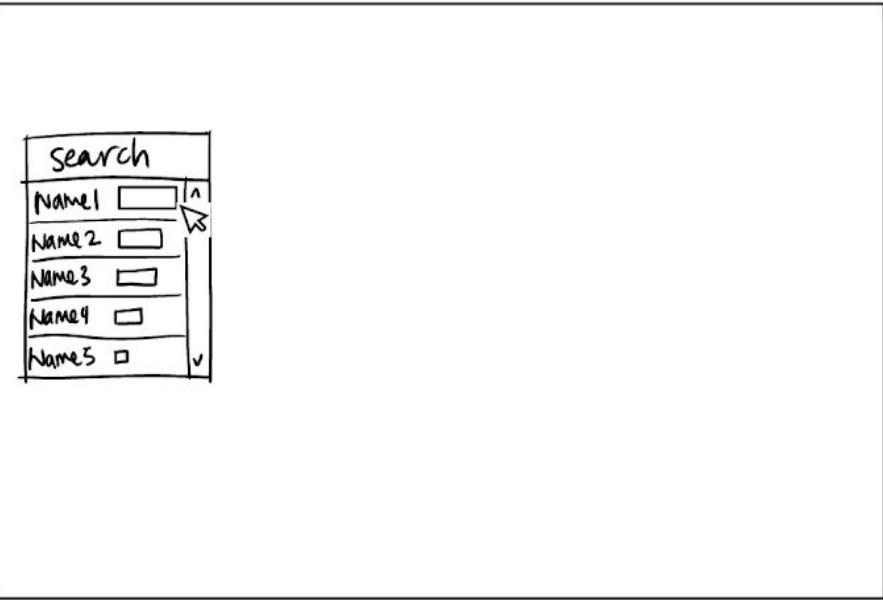
Sequence:



```
ttgacggcta gtcagtcct aggtacagtg
agaaatacta gatggccttg gttgacggtt
agtggaaaat tggagtggag cgccatcctc
tggattctcg aagatcctgt tcggcctggt
acgagaacgc cttcatcgtc ggcaactacc
tacgaccggg ctggctacgc gcgggtcgac
ccagagcgta ctgccgattt tctgggaacc
agcagcacga gttcttcgag gaagcctcgg
ctgaccatgc cgctgcatgg tgctcgcggc
```

Search Interaction:

When a user clicks on a search result, the selected genetic part is passed into the sbolgraph library, which gives us the genetic part's tree data.



The tree of the selected part renders in the area to the right of the search box. If a tree is already rendered, the tree of the selected part will replace it.

