

CS 171 Design Studio 1 - Github Commit Graph

In this design studio you will sketch multiple solutions to visualize the commit graph and the branches of a git repository. This design studio is synchronized with Homework 2. You must submit your (and your groups) solutions with Homework 2 and then choose one solution to implement as part of Homework 2. There is no need to hand in anything at the end of this design studio.

PART 1 - ANALYSIS

Individual; Time: 20 minutes

Take a look at various Github Network Graphs. Here are some examples:

<https://github.com/Caleydo/caleydo/network>

<https://github.com/mbostock/d3/network>

<https://github.com/CS171/CS171.github.io/network>

- Think about how these networks are different. Analyze the “dimensions” of these networks. What are the relevant attributes (e.g., commits, users, branches, commit size, etc.) of these representations? What other attributes could be relevant in this graph? Write a list of all the attributes your visualization could show.

Commits, users and branches are relevant in this graphs. Commit sizes and relationships between branches will be a good addition.

- Are there different roles, i.e., different types of users who might want to achieve different things? Write a list of user roles.

Various users will have varying needs for the visualisation. Project managers, contributors, project users and visitors.

- Think about which tasks a user of your visualization might want to achieve. Write down a list of tasks.
 1. Most recent commits.
 2. Most active repos.
 3. Highest contributors to repos.
- Identify one role that you want to design your visualization for. Prioritize your task and attribute lists based on this role's needs.

I would design a visualization for a fast way to see the relationship between various commits and branches.

PART 2 - SKETCHING

Individual; Time: 30 minutes

Design two alternative visual representations for the Github Network. You should design for an interactive system, i.e., you should not assume that you have to fit all content onto paper.

Here are some questions to consider:

- Decide on which visual variable to use for which attributes of the visualizations. Remember the strengths and weaknesses of visual variables we discussed in class. Use the strongest visual variable for the most important attributes of the data.
- Do you think it is necessary to represent every single commit as a separate node? Could you think of ways to aggregate this?

It's not necessary to represent every single node. For instance lone nodes could be deleted or combine with others.

- Do you think that every contributor needs a "row", as on the default network view on github? Could you think of a smarter way to summarize those?

No. Not every contributor needs a row. A linked graph or a bar graph would be handy.

- Is a node-link diagram the appropriate representation? Or should you consider alternative graph representations?

Yes it is appropriate.

PART 3 - Group Reflection

Group of up to 5 people; Time: 40 minutes, or as long as you like. You are invited to spread out through the building, e.g., to the lobbies outside, on the 2nd and third floor.

Take your analysis and ideas and share it with up to 4 of your fellow students. Discuss your priorities and your designs. Do you find a consensus? Come up with one visualization that you agree is ideal.