

CS1980 - Polkadot

September 23, 2020 Meeting

What Data to Pull from API

- <https://polkadot.js.org/apps/?rpc=wss://rococo-rpc.polkadot.io#/chainstate>
 - Using this link, we can see all of the different functions that are available
 - To view which parachains are connected:
 - * Chain State, Storage, query = parachains, function = heads()
 - returns a vector of parachains with their ID and their *most recent block hash*(if I understood correctly)
 - These different queries can be converted into code
 - For now, we mainly want to focus on which parachains are connected and their head
- At a high level, the goal for right now is:
 1. Show each new block as they come in (on the relay chain)
 2. Show which parachains are connected to the relay chain
 - At least their ID, and show their name(if possible)
 3. Show number of inbound/outbound messages for each parachain
 - Likely will be 0 most of the time due to a lack of activity, that is okay
 4. Click on a parachain to link to the Polkadot explorer for that parachain
 5. Any time a message is passed, show some indication that *something* has happened on the relevant parachains
 6. Have text-based update box showing what is happening (alongside the visual representation)
- Further down the road
 - Show arrow(or some other animation) from parachain to parachain when XCMP occurs
 - * Click on the arrow to show details about that message

Anime.js

- Kirsten recommended this for doing our data visualization
- Lightweight JS animation library
- It works with CSS properties, SVG, DOM attributes and JavaScript Objects.