This is an objective to create a read-only Stream Viewer UI (SVU). It will be a more visual way of displaying streams in lieu of the row-based “Streams” tab.



This Stream Viewer will eventually be grown to a write-enabled UI where a user can drag arrows and start streams. That was the original intent of the bounty, however it proved too nuanced for an all-at-once attempt. As a result the bounty will be reduced to $5k. After completion, another $5k bounty for the write-enabled version will be provided. If this bounty’s taker believes additional compensation is merited for either/both phases, please provide reason and the comp may be adjusted accordingly.

In constructing the visualizations, the (SVU) will attain a list of the user’s streams. We will use an overly complicated example to flesh out as many edge cases as possible.

Example: the user is streaming with DAI->ETH, ETH->DAI, USDC-> IDLE, DAI->IDLE, USDC->DAI, DAI->MKR, WETH->USDC/WETH

List: [ [DAI, WETH] , [WETH, DAI ], [USDC, IDLE] , [DAI, IDLE], [USDC->DAI], [DAI->MKR], [WETH->USDC/WETH] ] - ordered by volume (would be in USD, so for example, .25 ETH/mo. Is obviously greater than 100 DAI/mo.)

The interface will then iterate down the list (the ordering does not matter) and produce the visualization in this fashion:

Note! This viz construction is not additively responsive. As in, if the user starts a new stream, we don’t try to figure out what current box to draw a line from an all. Rather, the iteration runs again from the top and the logic captures the change.

[DAI, WETH]

1. Get input token (left token)

Example: Gets DAI

1. See if any other checked pairing has DAI as an input

Example: None (first pairing being checked)

1. See if any other checked pairing has DAI as an output

Example: None (first pairing being checked)

1. Get output token (right token)

Example: Gets WETH

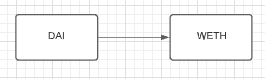
1. See if any other checked pairing has WETH as an input

Example: None (first pairing being checked)

1. See if any other checked pairing has WETH as an output

Example: None (first pairing being checked)

1. Generate viz



[WETH, DAI]

1. Get input token (left token)

Example: Gets WETH

1. See if any other checked pairing has WETH as an input

Example: None

1. See if any other checked pairing has WETH as an output

**Example: [DAI->WETH] - line origin point will be that WETH box**

1. Get output token (left token)

Example: Gets DAI

1. See if any other checked pairing has DAI as an input

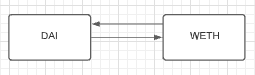
**Example: [DAI->WETH] - line end point will be that DAI box**

1. See if any other checked pairing has DAI as an output

Example: None

1. Generate viz

Line goes back from WETH to DAI



[USDC, IDLE]

1. Get input token (left token)

Example: Gets USDC

1. See if any other checked pairing has USDC as an input

Example: None

1. See if any other checked pairing has USDC as an output

Example: None

1. Get output token (right token)

Example: Gets IDLE

1. See if any other checked pairing has IDLE as an input

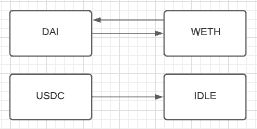
Example: None

1. See if any other checked pairing has IDLE as an output

Example: None

1. Generate viz

Since it’s totally exclusive from the current visualization, a separate pairing is rendered below



[DAI, IDLE]

1. Get input token (left token)

Example: Gets DAI

1. See if any other checked pairing has DAI as an input

**Example: [DAI->WETH] - line origin point will be that DAI box**

1. See if any other checked pairing has DAI as an output

Example: None

1. Get output token (left token)

Example: Gets IDLE

1. See if any other checked pairing has IDLE as an input

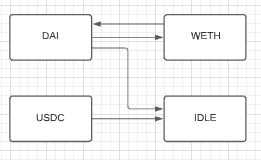
Example: None

1. See if any other checked pairing has IDLE as an output

**Example: [USDC->IDLE] - line end point will be that IDLE box**

1. Generate viz

Line connects from DAI box to IDLE box



[USDC, DAI]

1. Get input token (left token)

Example: Gets USDC

1. See if any other checked pairing has USDC as an input

**Example: [USDC->DAI] - line origin point will be that USDC box**

1. See if any other checked pairing has USDC as an output

Example: None

1. Get output token (left token)

Example: Gets DAI

1. See if any other checked pairing has DAI as an input

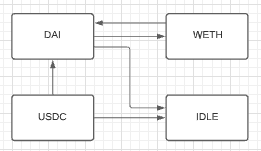
Example:  **[DAI->WETH] - line end point will be that IDLE box**

1. See if any other checked pairing has DAI as an output

**Example: [USDC->IDLE] - line end point will be that IDLE box**

1. Generate viz

Line connects from DAI box to IDLE box



[DAI, MKR]

1. Get input token (left token)

Example: Gets DAI

1. See if any other checked pairing has DAI as an input

**Example: [DAI->WETH] - line origin point will be that DAI box**

1. See if any other checked pairing has DAI as an output

Example: **[WETH->DAI] - line origin point will *still* be that DAI box**

1. Get output token (left token)

Example: Gets MKR

1. See if any other checked pairing has MKR as an input

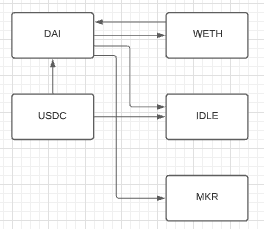
Example: None

1. See if any other checked pairing has MKR as an output

Example: None

1. Generate viz

MKR box is rendered in second column from DAI and placed on the bottom of the column



[WETH, USDC/WETH]

1. Get input token (left token)

Example: Gets WETH

1. See if any other checked pairing has WETH as an input

**Example: [WETH->DAI] - line origin point will be that WETH box**

1. See if any other checked pairing has WETH as an output

Example: **[DAI->**WETH**] - line origin point will *still* be that** WETH **box**

1. Get output token (left token)

Example: Gets USDC/WETH (the rexLP)

1. See if any other checked pairing has USDC/WETH as an input (impossible, you can’t take USDC/WETH as an input for any stream market)

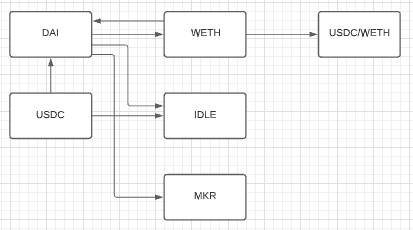
Example: None

1. See if any other checked pairing has USDC/WETH as an output (possible, you *can* have multiple tokens outputting the same rexLP)

Example: None

1. Generate viz

Since WETH is on the second row and this is a new stream market not in an existing column, a new column must be constructed to the right of WETH.



The design for this UI will follow Josef’s Figma Concept here: <https://www.figma.com/file/Wbr92wkrSLLVY3u5EHwft4/Ricochet?node-id=111%3A902>

* The lines will vary in color based on the input token (so a line from a DAI box should be gold/orange, from USDC it’ll be blue, etc)
* The user’s balance of the token should be displayed on the token’s box
* The stream rate should be shown on the line with additional details upon hover



* The size of the stream relative to the others should also be indicated visually by the girth of the connecting arrow

LucidChart behind screenshots here:

<https://lucid.app/lucidchart/e3f548ce-0772-494a-875b-8417786aa3ef/edit?invitationId=inv_ace9923d-ab2e-4ebc-8de0-421673dacd8a>

My hope is that these example would suffice as enough instructional substance to get started. If additional details are required, please let me know.