Summary of the tool and visualization:

There are 2 different tools here:

1. Gives cluster dependencies and cluster belongs to which concern (word list).
2. Gives which cluster is categorized to which concern.

These tool works good on any dependency file and cluster file generated by RELAX recovery method.

Tool 1 :

Producing visualization based on deps file and clusters file.

Name of the file is: **"txt2CSV-forarch.html"**

This tool basically converts clusters.rsf file and deps.rsf file into **CSV** format which is taken as input for visualization.

In a CSV file, Column A contains cluster dependent on column B and Column C contains concerns of column A.

Steps to run the tool:

Prerequisite: all files should be in download folder of the browser

1. Open "***txt2CSV-forarch.html***" file in firefox.
2. Enter name of the file that needs to be saved as in your local machine(ex – nbArch)
3. Choose cluster.rsf file and deps.rsf file (please select in order first is cluster.rsf then deps.rsf).
4. Upon choosing, the files are converted to CSV format and is downloaded to your local machine download folder of the browser.
5. Click on the link saying **"Click here to see visualization".**
6. This will open another html file name **"arch.html"** which contains the visualization of the file converted above. (output is as below for chukwa 0.4.0)

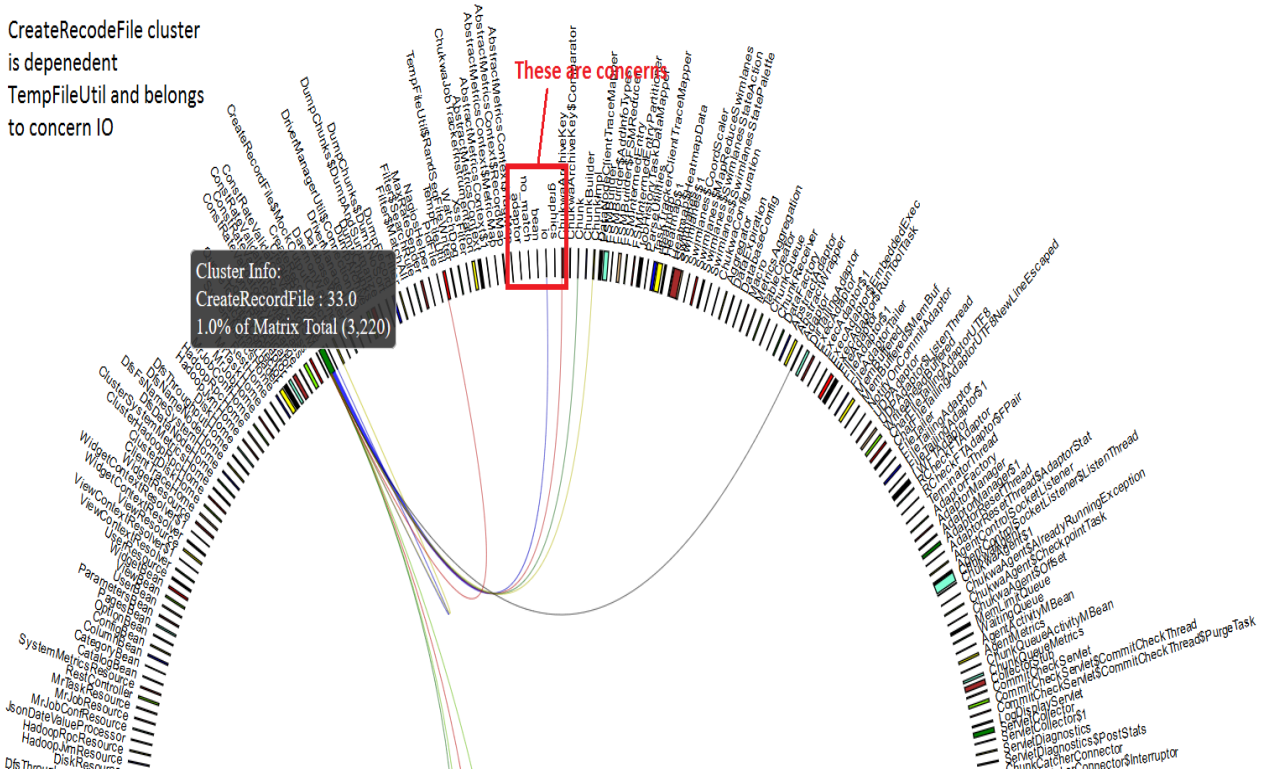


Fig: 1 - showing cluster is dependent on which another cluster

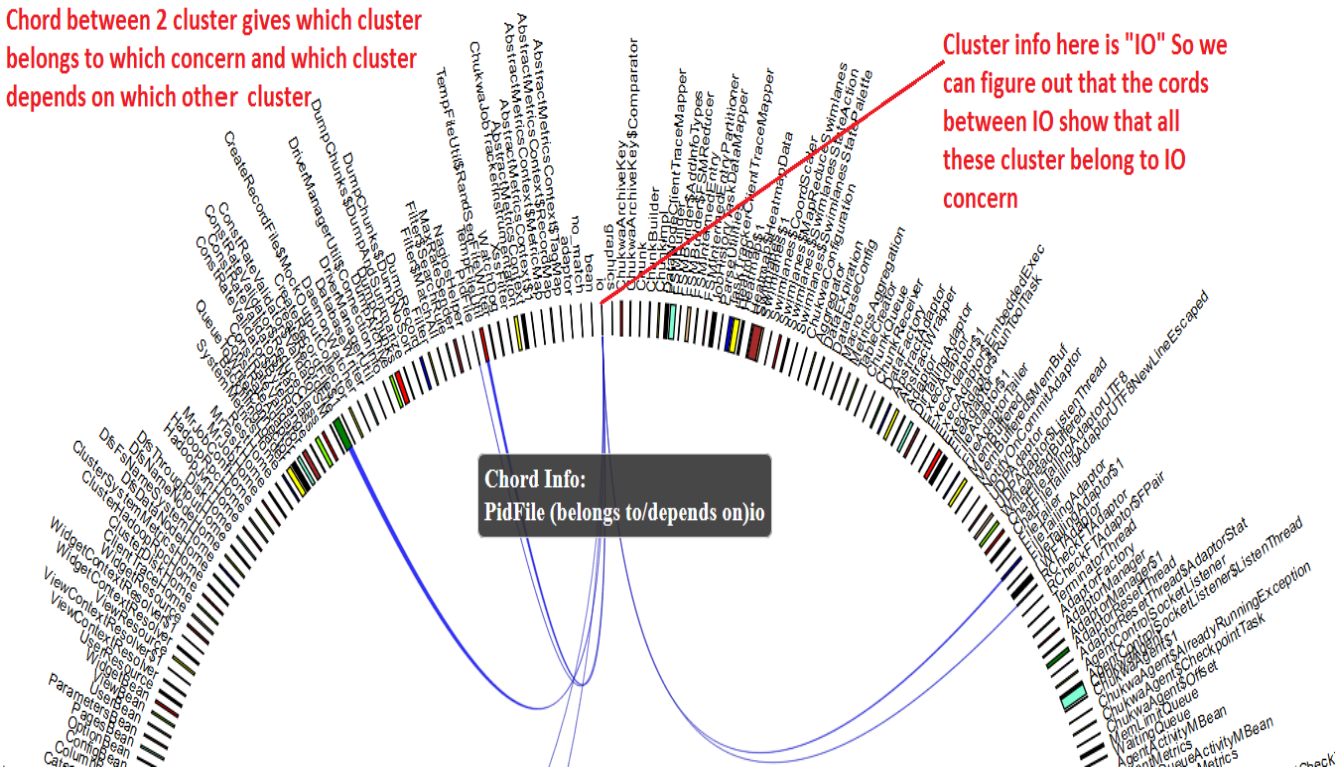


Fig 2: Showing each chord belongs to which cluster or depends on which another cluster.

From the above visualization, we can get to know about the architecture of the system.

- This visualization is related to the architecture of the system in a way to find out which cluster is dependent on the which cluster along with to which concern the dependent cluster belongs to.
    - For example in figure 1: CreateRecordFile cluster is dependent on TempFileUtil , ChukwaArchiveKey and many more (as shown in red,green,yellow lines). Also, CreateRecordFile belongs to cluster IO (as shown in blue line). This helps person understand more about the cluster components and their interaction between them.
- Hovering on the group (i.e., the bars next to names) it shows us how much percent of clusters of the whole is dependent on another cluster along with cluster name.
- Hovering on the chord (i.e., individual lines) shows which cluster belongs to which concern or which cluster is dependent on which another cluster.
- Hovering on the concern bars as shown in (Fig 3 below) shows which all clusters belongs to that concern for example hovering on “adaptor” concern shows chord to all clusters which belongs to that concern.

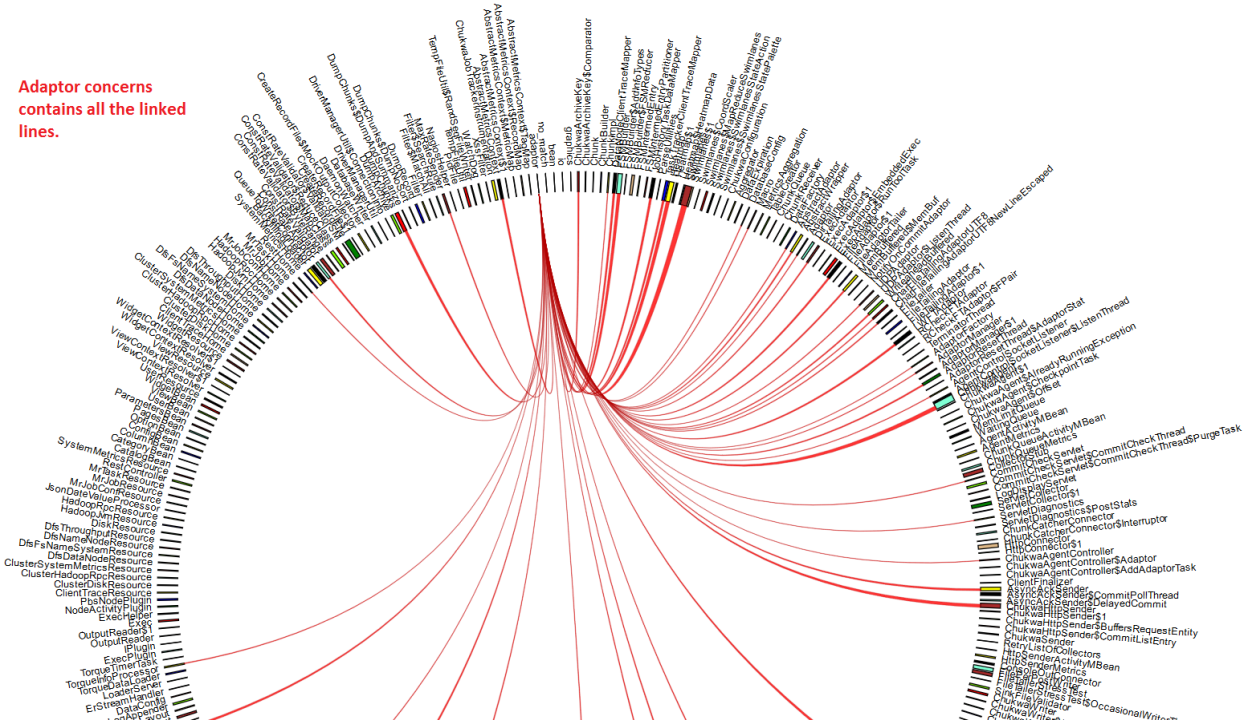
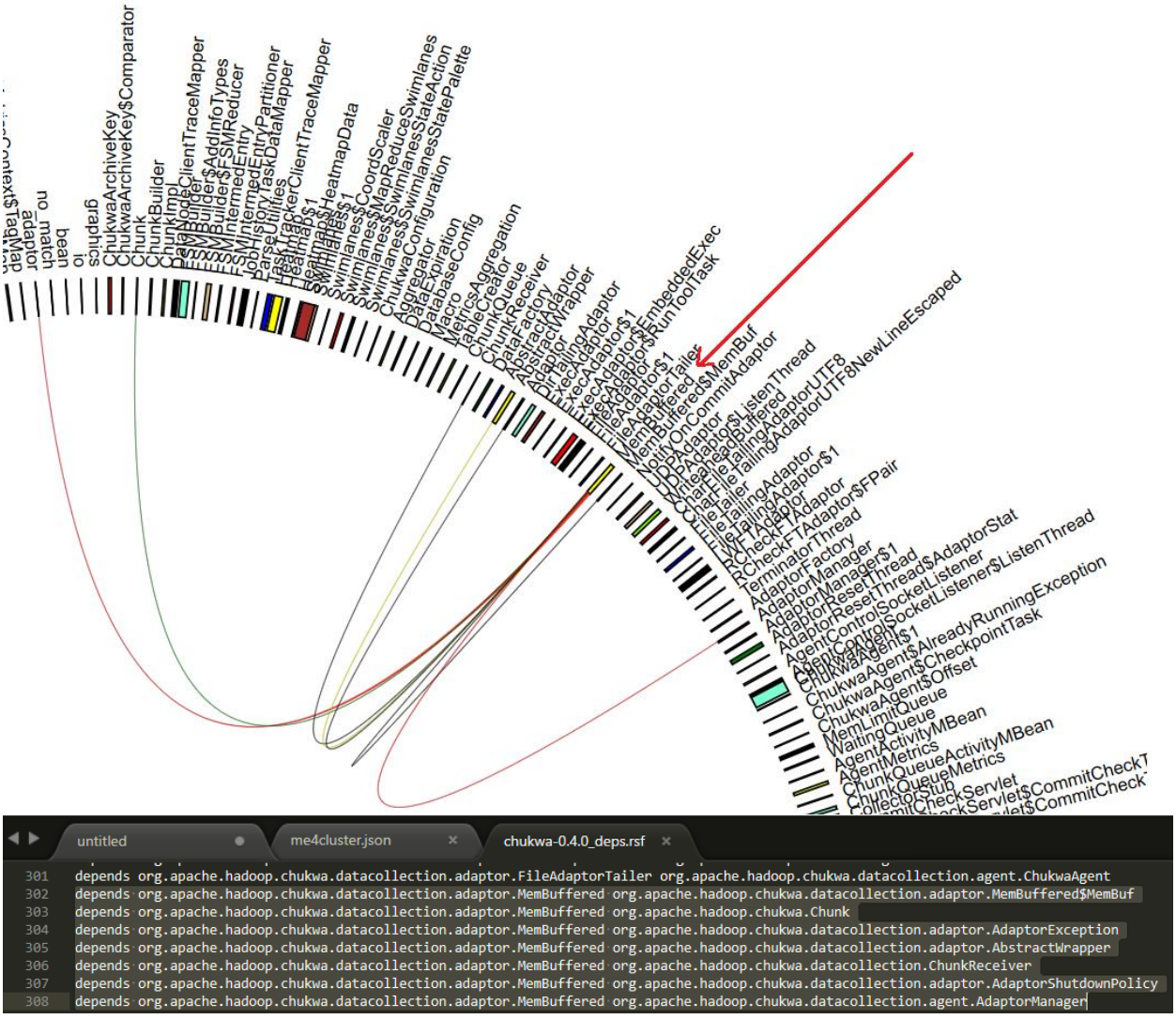


Fig 3: All clusters belonging to concern.

- The color of each concern can be modified based on the input depending upon the legends file's output (currently this is not implemented but can be extended easily).
- Since this visualization is completely dependent on deps file and cluster file, all the output data shown are correct and it can be easily understood by any developer or software engineer in the company.
    - Example: Suppose a new developer is joined to the system and wants to know about which java file extends or imports which another java/class file it is easy. In Fig -4: MemBuffered cluster is dependent on Chunk, ChunkReceiver, AbstractWrapper, MemBuffered$MemBuf and Adaptor Manager.



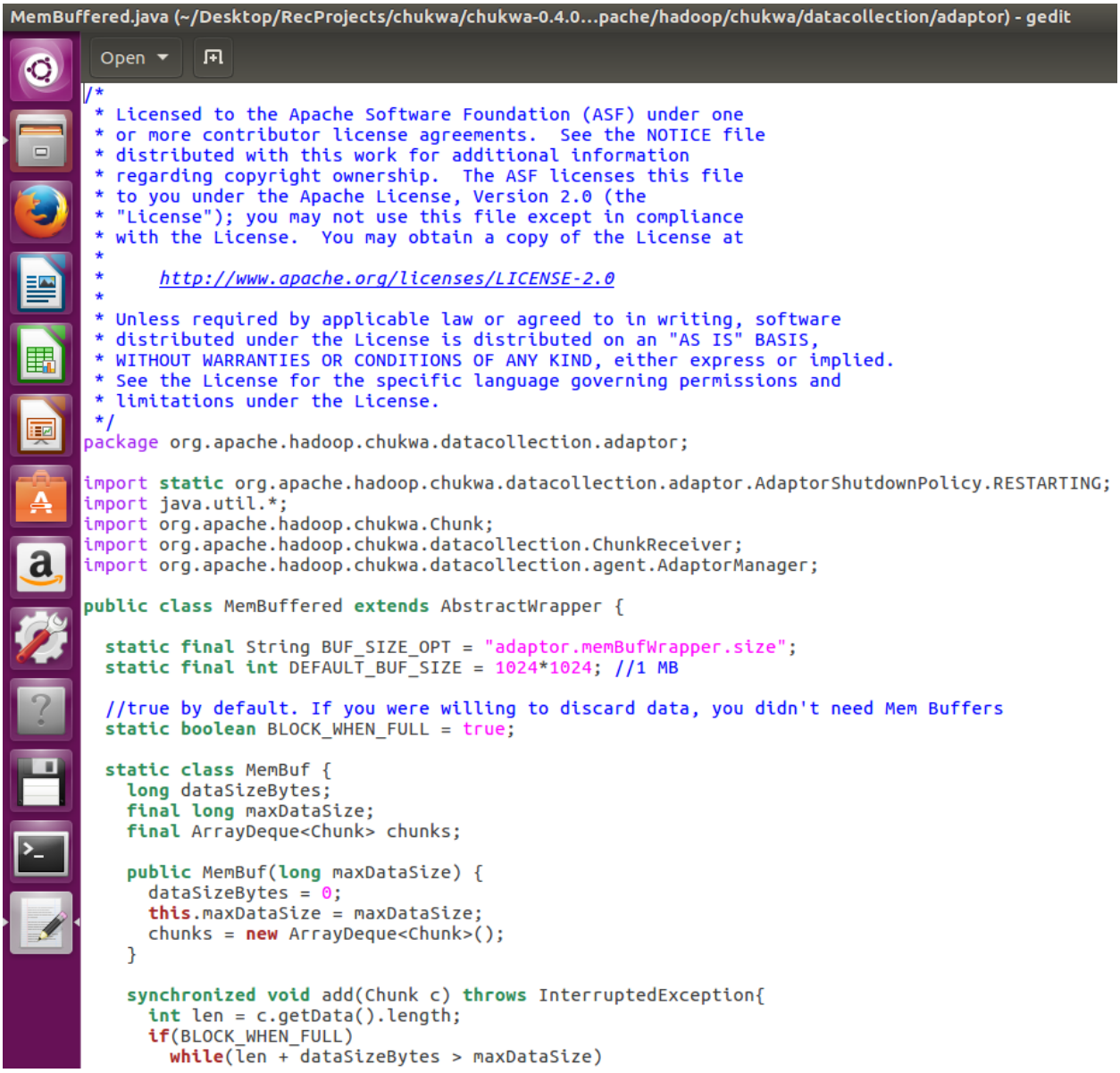


Fig 4 – showing MemBuffered cluster dependencies.

In this visualization for simplicity purpose I have reduced the path of file names (i.e., "**MemBuffered**" is written for "**org.apache.hadoop.chukwa.datacollection.adaptor.MemBuffered"**), we can also display full names (as shown in fig-5)

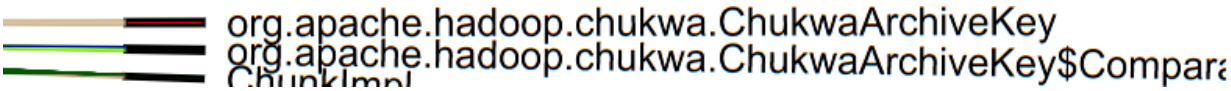


Fig 5 – showing full names instead of last word.

- This visualization works for any system which has clusters and deps as output file from the recovery, also it is easy to maintain as input file as it is in csv format. This can be scaled to different system just by converting them to csv format.

Here is the complete output for ME algorithm chukwa-0.4.0

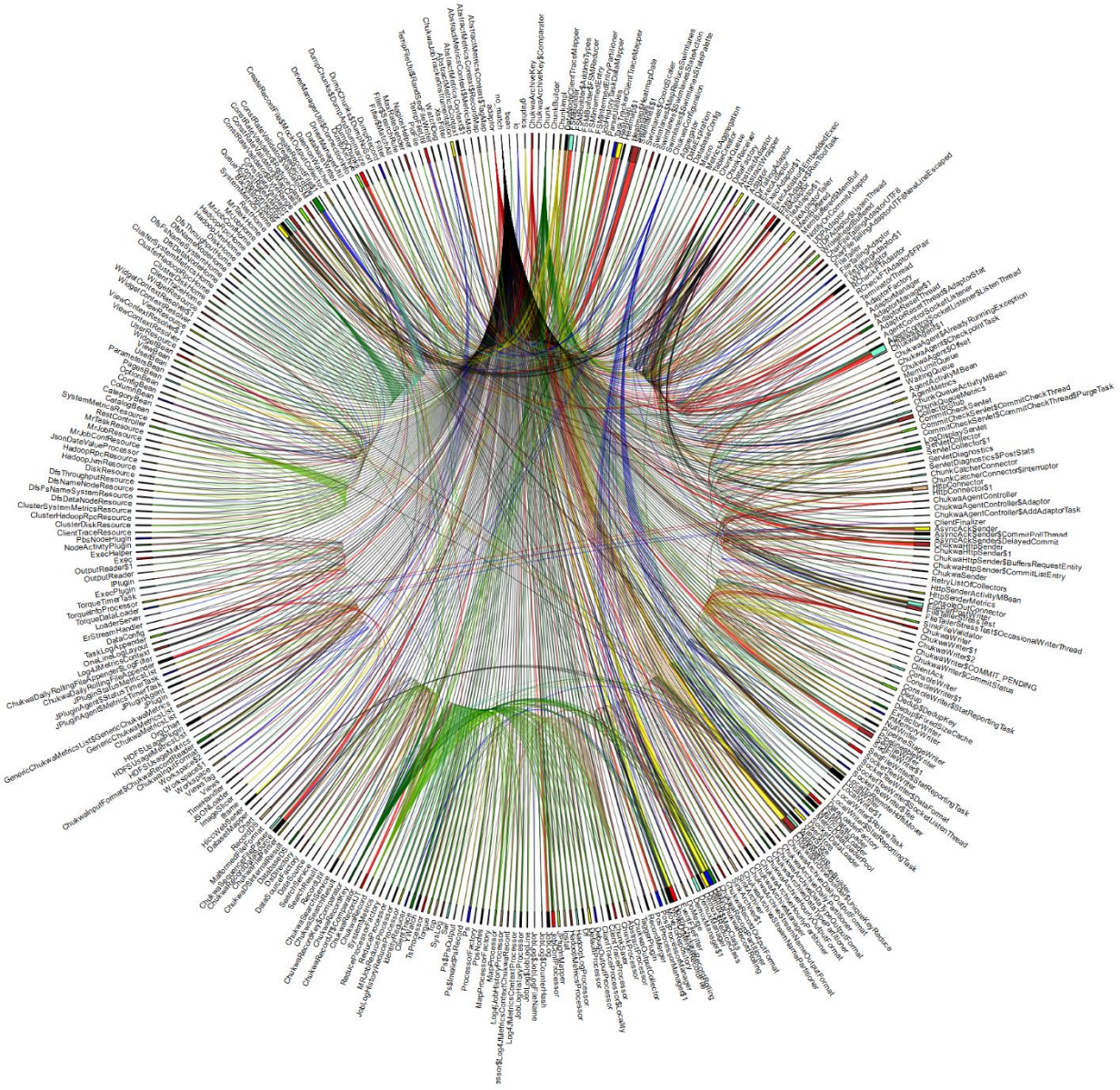


Fig 6: complete architecture picture of deps and clusters together using chord diagram

TOOL 2:

Visualization based on clusters.rsf file.

Currently in RELAX recovery method, we get clusters visualization but it is can be made human friendly, here is one example:

Name of the file to run is: "***txt2json-forhigh.html***"

This tool basically converts clusters.rsf into **JSON** format which is taken as input for visualization.

In a JSON file, it contains "name" and "imports" parameters with name having name of the concern and imports containing clusters.

Steps to run the tool:

Prerequisite: all files should be in download folder of the browser

1. Open "***txt2json-forhigh.html***" file in firefox.
2. Enter name of the file that needs to be saved-as in your local machine (ex – nbArch, need not mention any extension)
3. Choose cluster.rsf file.
4. Upon choosing, the files are converted to JSON format and is downloaded to your local download folder with name mentioned in step 2.
5. Click on the link saying **"Click here to see visualization".**
6. This will open another html file name **"cluster.html"** and parameter as name of the input file, which contains the visualization of the file converted above. (output is as below for chukwa 0.4.0)

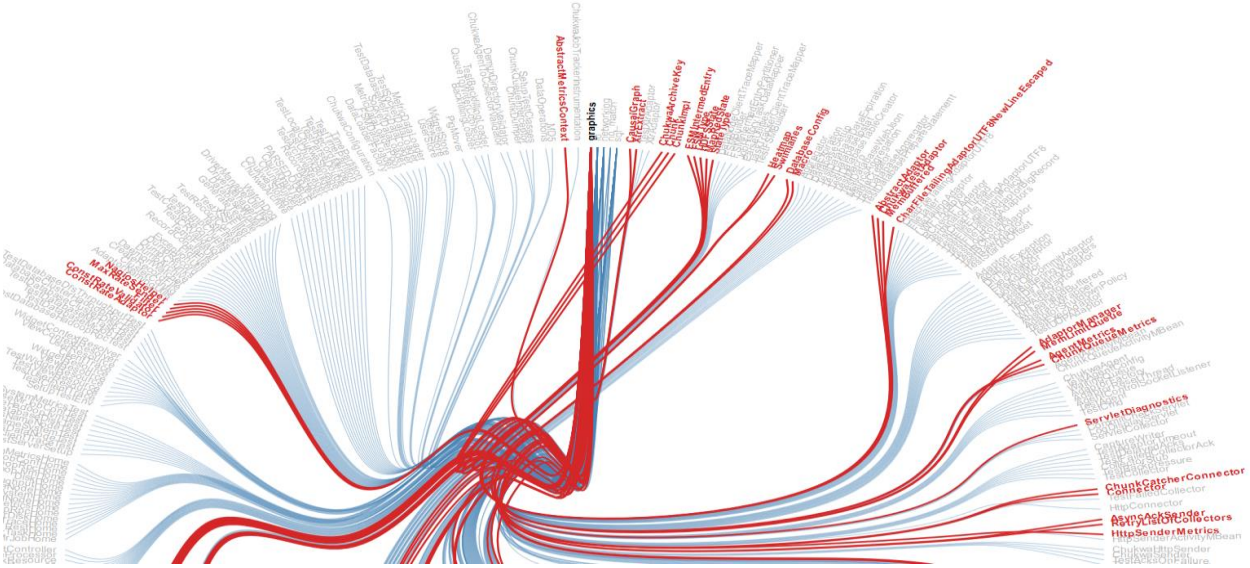


Fig-7: showing clusters belongs to concern.

From the above visualization, we can get to know about the architecture of the system.

- This visualization is related to the architecture of the system in a way to find out which cluster belongs to what concern.
    - For example, in figure 7 above: all the highlighted clusters in Red belong to concern “Graphics”
- Hover over the concern name, shows all the cluster that belongs to selected concern.
- Similarly hover over cluster shows to which concern selected cluster belongs.

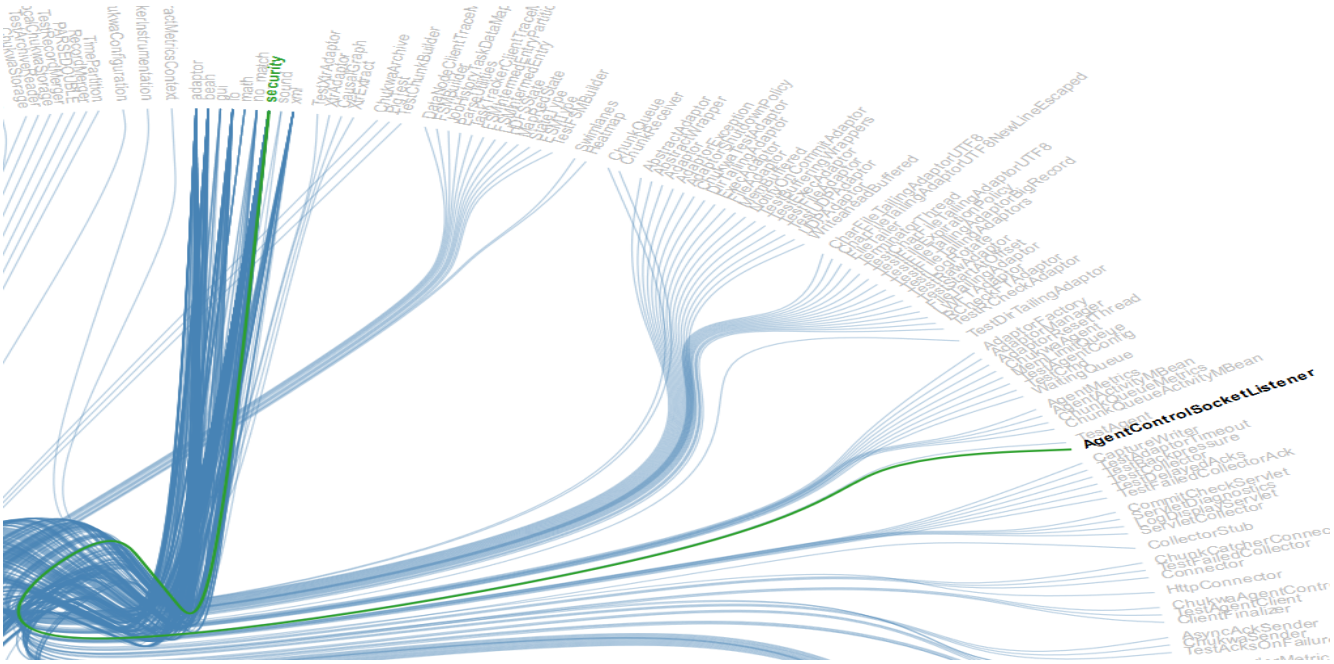


Fig-8: showing cluster belongs to security concern.

- This tool helps in understanding the system clusters and concerns relation.
- Here also the name is shortened to the last file name. we can still use complete name/path.
- This visualization works for any system which has clusters and deps as output file from the recovery also it is easy to maintain as input file is in JSON format. This can be scaled to different system just by converting them to JSON format. Any cluster.rsf files can be used to convert into JSON format.

Based on these visualizations one can get good overview of the system by understanding its dependencies along with concern category. By including these visualizations in RELAX recovery method would give more meaningful insight of whole architecture.

Here is the complete output for Chukwa-0.4.0

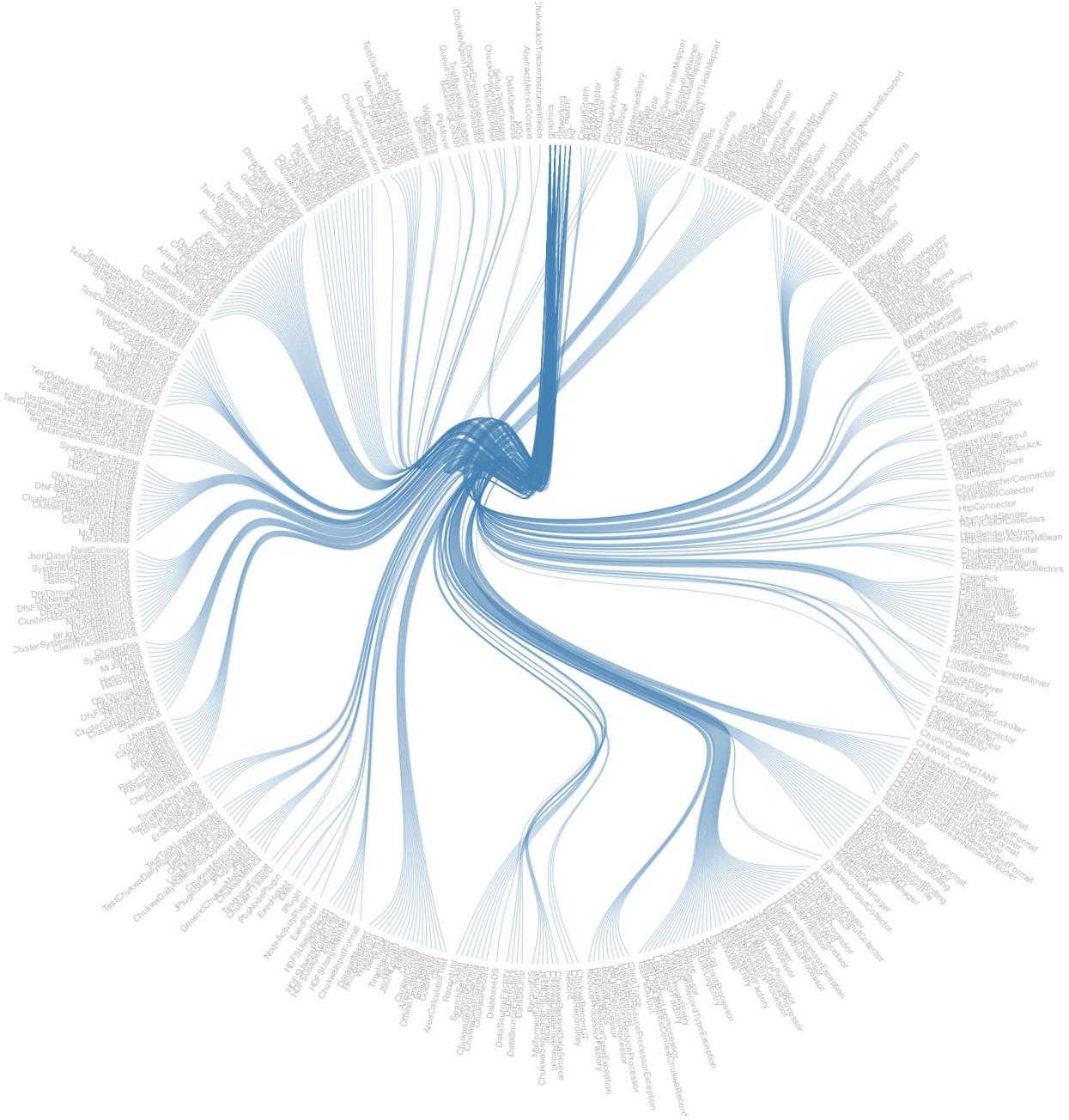


Fig-9: showing complete clusters using hierarchical edge diagram.