K0TO Level 3 Software

Nikola Whallon

Online and Offline Software

- The purpose of the K0TO Level 3 (L3) system is to capture and store event data while the data acquisition (DAQ) system is online. The program l3_daq accomplishes this task and is documented in detail in the technical note “Level 3 of the K0TO DAQ System.pdf”
- However, I have also prepared a few pieces of software for use offline in analyzing event data and the Level 3 system. These include extract_event, find_clusters, csi_visualizer, and compression_ratio.

L3 Software Overview

- I have put together a suite of L3 software. The suite includes the following binary executables:
  - l3_daq
  - extract_event
  - find_clusters
  - csi_visualizer
  - compression_ratio
- and the following scripts:
  - make-install-binaries.sh
  - start_l3.sh
  - stop_l3.sh
  - l3_cluster.sh

L3 Documentation Overview

- The Level 3 software suite also includes the following documents:
  - “README.txt” files for each binary executable
  - “Level 3 of the K0TO DAQ System.pdf”
  - “K0TO Level 3 Software.pdf” (this document)
  - “K0TO Using Git.pdf”
  - “K0TO O(n) Clustering.txt”
  - “K0TO Run File Format 5.pdf”
  - “K0TO Run File Format 6.pdf”
  - “Compression Ratios.pdf”

Acquiring the Software

- The software (and documentation) is easily obtainable/maintainable via a git repository. To obtain the software, simply run the following command in a terminal:

  ```
  git clone https://USER@codeblue.umich.edu/git/daq
  ```

- USER is your codeblue user name (see the document “K0TO Using Git.pdf” on the Indico or K0TO web pages for more information on using git and codeblue).
- This presentation will give an overview of the software.

Installing and Dependencies

- The software should compile and run on most linux/unix machines with bash. There are, however, the following dependencies:
  - l3_daq requires libpthread and libpcap
  - csi_visualizer requires libsdl and libgl
- The easiest way to install the software is to run the script make-install-binaries.sh as root. This will attempt to put symlinks of the binary executables and scripts in ~/bin.
- Doing this allows the commands to be called directly from any directory, making working with the software easier and the DAQ control scripts more simple.

l3_daq

usage:

l3_daq runid

- l3_daq is the primary piece of Level 3 software. It collects, cuts, compresses, and stores k0to event data into k0to run files online.
- l3_daq is extensively documented in “Level 3 of the K0TO DAQ System.pdf,” and so little is said here...

start_l3.sh / stop_l3.sh

usage:

start_l3.sh runid   /   stop_l3.sh runid

- start_l3.sh and stop_l3.sh are scripts which help automate the process of starting and stopping l3_daq on multiple mandolin nodes.
- The scripts are called to run when the central K0TO DAQ system starts.
- It would be useful to include these scripts as well as l3_daq in the $PATH variable on mandolin nodes. That way, the central K0TO DAQ system calls them, it does not need to keep track of a full path.

(make-install-binaries.sh attempts to do this)

extract_event

usage:

extract_event filename event_number

- extract_event decompresses a single specified event in a compressed k0to run file, and stores it in a separate file.
- It will only attempt to process a k0to run file if the data format of the file is '6'.

find_clusters

usage:

find_clusters csimapping_file_path event_file_path

- find_clusters will perform the same clustering algorithm/function used by the online l3_daq program to find cluster information about the csi channels in the data of an event. The cluster information is then output to a .txt file.
- An uncompressed event file must be supplied, as well as a csimapping file used to determine which channels correspond to csi crystals.

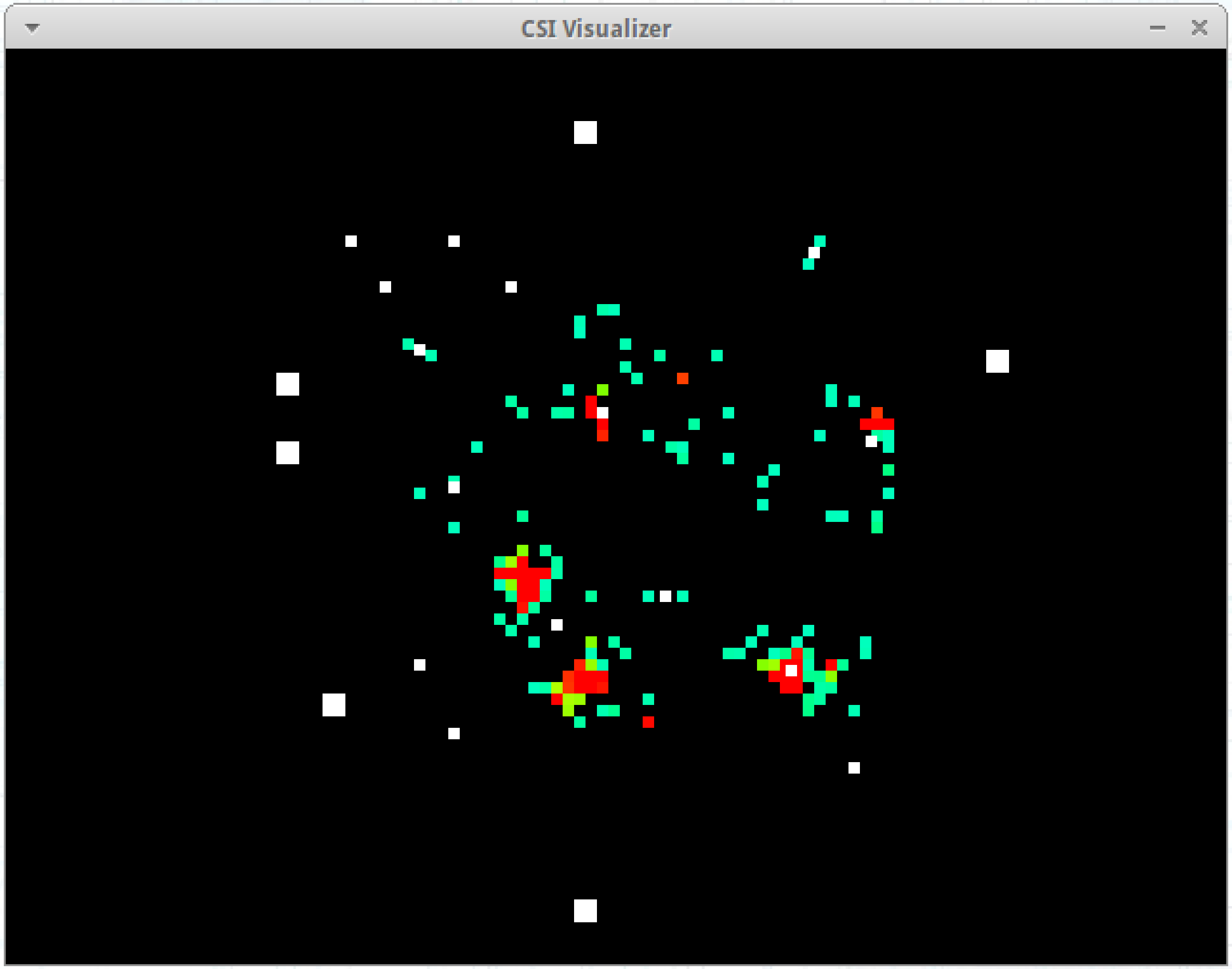
csi_visualizer

usage:

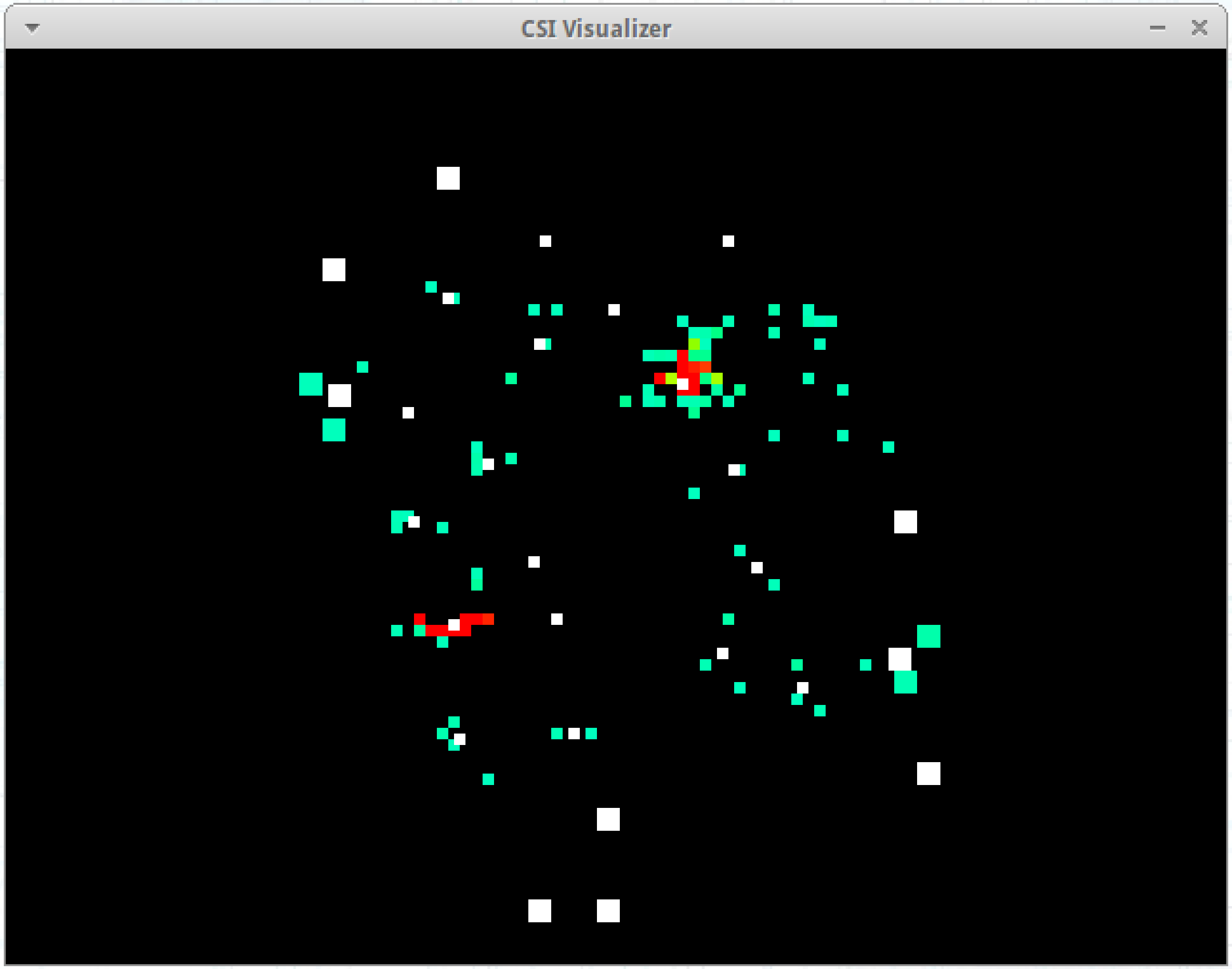
csi_visualizer filename

- csi_visualizer visualizes the cluster information output by find_clusters.
- Crystals in the visualization are colored to represent their energy. White points are actually not crystals, but the points of the center of energy of clusters.
- The following 3 slides show examples of this output.

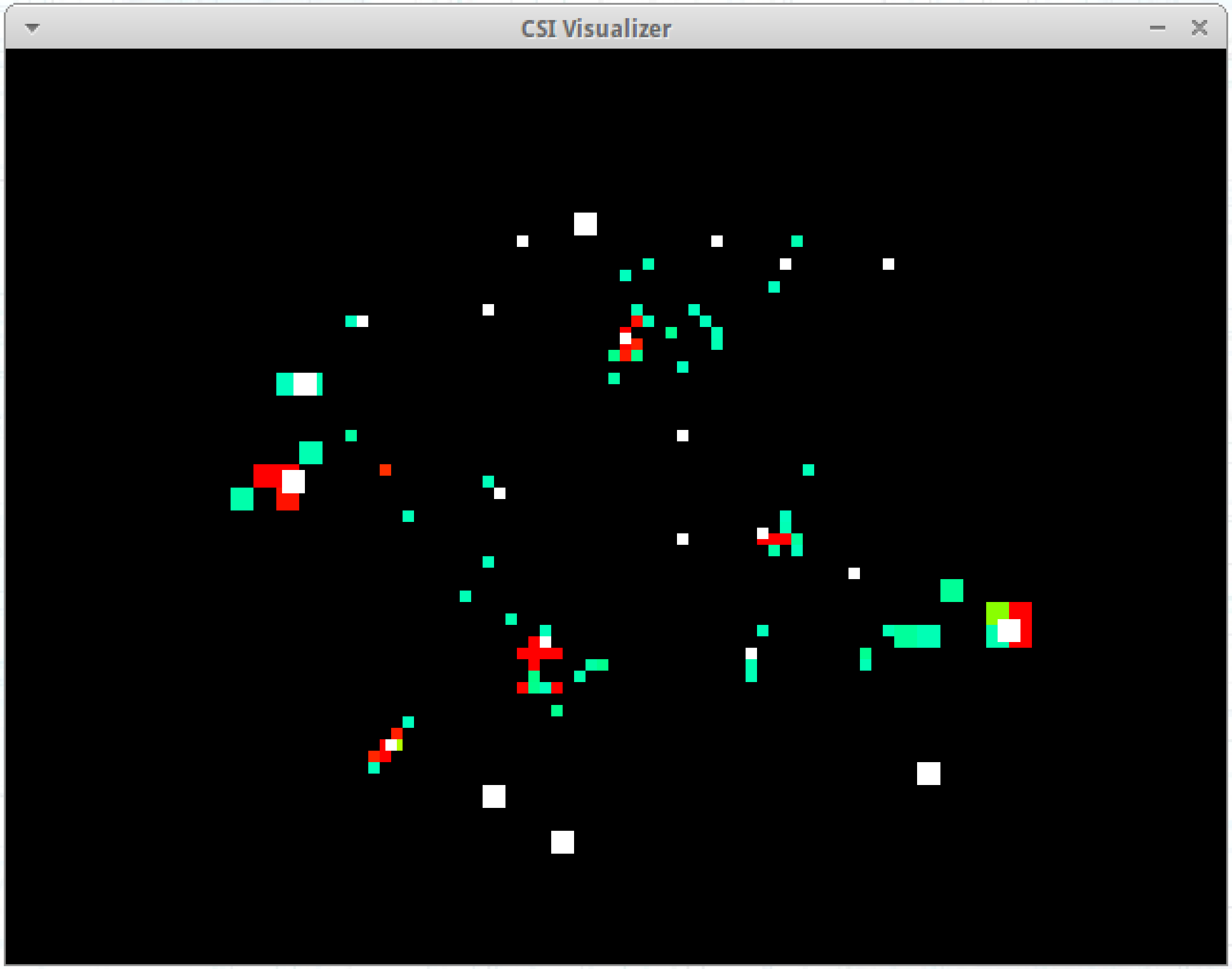
run16898node0file0event5



run16898node0file0event566



run16898node0file0event6793



Notes on Clustering

- The main use for find_clusters is to perform cluster analysis offline using the same functions for clustering used online.
- This will allow offline studies of how to best use the cluster information in an online L3 cut, which l3_daq is capable of doing.
- I have also documented a new clustering algorithm that has O(n)~O(n^(3/2)) complexity in the note “K0TO Order(n) Clustering.txt” However, the current algorithm works with no practical limitations.

l3_cluster.sh

usage:

l3_cluster.sh run_file_path event_to_extract

- l3_cluster.sh is a script that runs extract_event, find_clusters, and csi_visualizer in succession to quickly visualize the clusters in an event in a runfile.
- The first execution of the script will be slow, as the large run file is processed, but successive executions of the script (to look at other events, for example) are very fast.

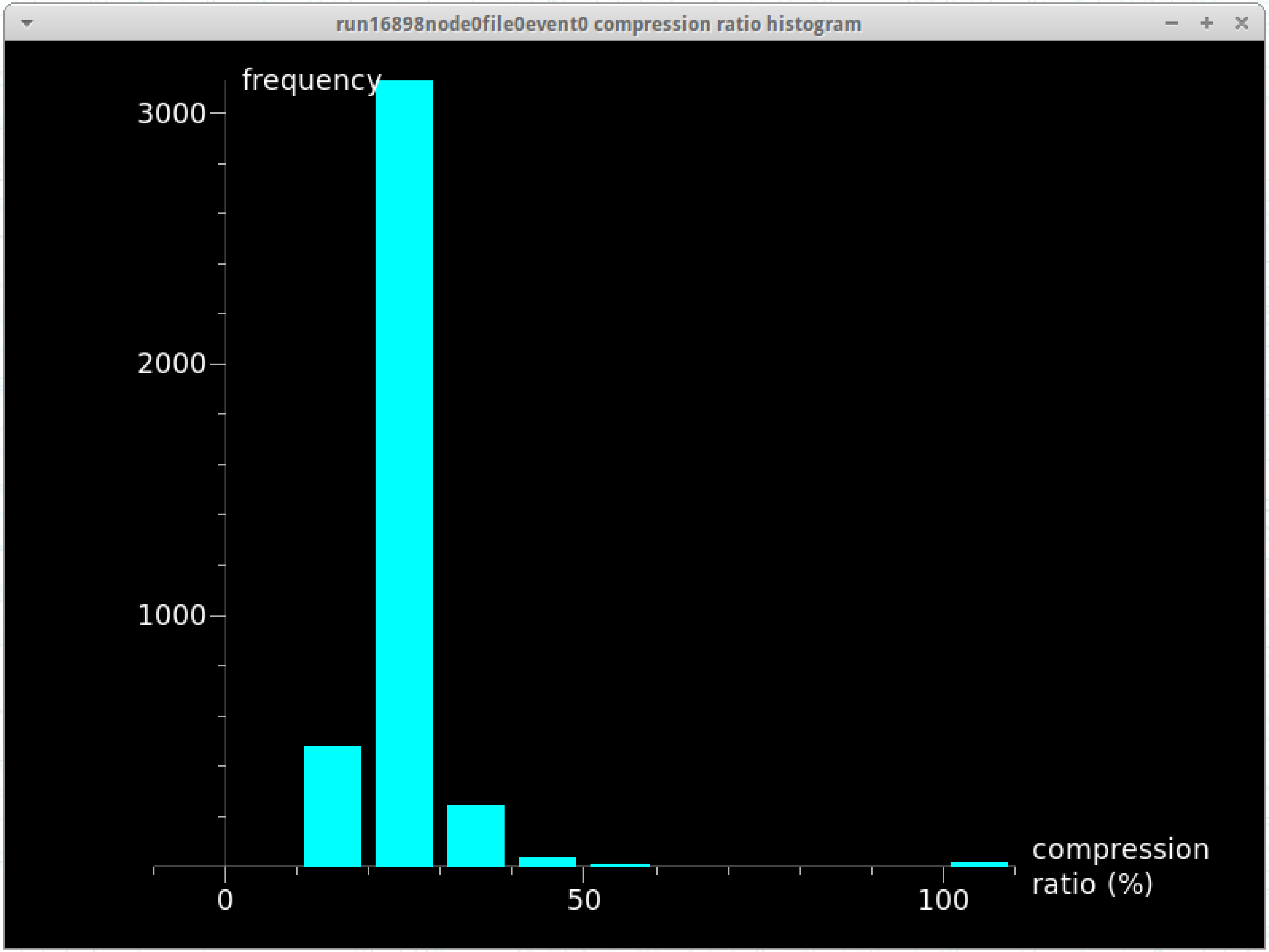
compression_ratios

usage:

```
compression_ratios filename event_number
```

- compression_ratios calculates the compression ratios of every channel of data in an event in a compressed k0to run file, and stores the information in a simply formatted .txt file.
- It will only attempt to process a k0to run file if the data format of the file is '6'.
- The .txt file can then be simply parsed to create histograms of the compression ratios, as shown on the next slide.

run16898node0file0event0 compression ratio histogram



Run Data Configuration

- The configuration of the k0to daq system (number of samples per channel per event, total number of channels per event, etc) is occasionally altered and tweaked.
- The l3_daq code has used a configuration file to know the format of incoming data to effectively parse and process.
- The offline programs I have worked on also require much of this information, as does Togawa-san's production code.
- In this suite, all programs have been using the same config.h file for this reason.
- I suggest all similar programs use the same configuration file, and data files be parsed according to their data format version number.

Conclusion

- This presentation aimed to give a brief summary of Level 3 software of the K0TO DAQ system.
- The software is simple, easy to use, and well documented. l3_daq is the essential piece of Level 3 software, and the other tools are useful for analyzing data offline.
- See the technical note “Level 3 of the K0TO DAQ System.pdf” for more detail on the program l3_daq.

Future

- In order to better test the cut/compress/etc functions in the online l3_daq program, it would be ideal if those functions exist within their own source files that can then be used by other programs.

  (Currently, the cut/cluster function, for example, is literally copied and pasted from the l3_daq code to the find_clusters code – not very elegant!)

- If it is desirable in future, the clustering algorithm described in “K0TO O(n) Clustering.txt” can be implemented.

(^_^) Thanks! :3