

**Version**

**1.0**

JEFFERSON LAB

---

Data Acquisition Group

# JEventViewer User's Guide

JEFFERSON LAB DATA ACQUISITION GROUP

# **JEventViewer User's Guide**

---

Carl Timmer  
[timmer@jlab.org](mailto:timmer@jlab.org)

24-Jan-2014

© Thomas Jefferson National Accelerator Facility  
12000 Jefferson Ave  
Newport News, VA 23606  
Phone 757.269.7365 • Fax 757.269.6248

---

# Table of Contents

**1. Evio Event-Viewing Gui.....3**

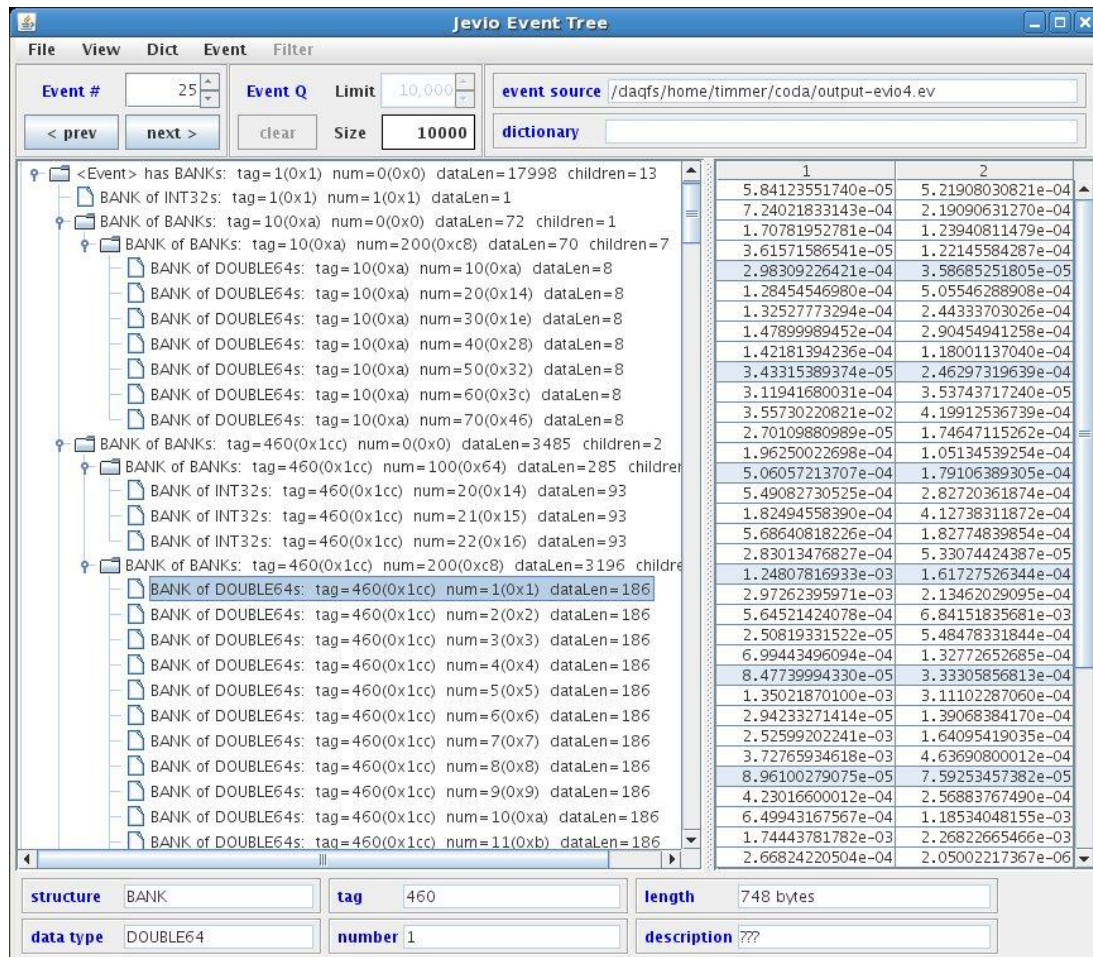
## 1. Evio Event-Viewing Gui

There is a graphical user interface for looking at EVIO format files event-by-event, although it can also look at any file as a list of 32 bit integers. To run it simply execute:

```
java org.jlab.coda.eventViewer.EventTreeFrame
```

while making sure this class or the jar file JEventViewer-1.0.jar is in your CLASSPATH environmental variable. The following is a screen shot of the gui.

Figure 1.1: Event-viewing gui



## JEVIO EVENT-VIEWING GUI

Here's a quick list of all the available, user-selectable features:

- valid event sources are files, cMsg messages, and ET buffers
- fast compare ability for data from different events
- when receiving events through cMsg or ET, they can be filtered based on their CODA event type (physics, control, etc) and trigger type if physics event
- view integer data as hex or decimal
- select dictionary from event source or from separate file containing dictionary
- view the dictionary being used
- export any evio file in xml format (MUST BE TESTED)
- split between the event tree panel and the data panel may be vertical or horizontal
- can add or remove data columns
- view the contents of any file as 32 bit hex integers

Starting with the middle of the gui first, the left side shows a tree structure diagram of the whole, single evio event being viewed. Notice that the type of each evio structure is given (bank, segment, tagsegment), along with the type of data it contains, tag, num, size, and # of children. Tag and num are shown in decimal and hex. If a dictionary is being used, the dictionary name is displayed instead of the corresponding structure type, data type, tag, and num values.

The right side, on the other hand, shows the data of any selected bank, segment, or tagsegment that contains a primitive data type. The number of columns can be set in the "View" menu. Integers can be displayed in hex or decimal.

A fast compare feature is able to compare data from different events. If the current event is changed while viewing the data of its selected structure, and if the new event has a structure with the same hierarchy of tags that the previous selection had, it too is automatically selected. This facilitates comparing the same structure in each successive event by simply hitting the "next" event button.

A dictionary can be loaded from a separate xml format file, or it can come embedded in an evio format file or buffer (cMsg, ET). The viewer allows the user to switch, in the "Dict" menu, between the different dictionaries if more than one is available. Any dictionary being used can be displayed instead of the data.

Selecting an ET system or a cMsg server as an event source, in the "Event" menu, brings up other menus to allow the proper connections to be created and maintained. The only assumptions made are that in a cMsg message, the evio data is contained in the byteArray field. Any dictionary is first looked for in the evio data and if none is found, it is looked for in a String payload item called "dictionary".

The box in the upper left (under the row of menu buttons), "Event #", shows the event currently selected (in this case 25) and allows the user to navigate to the desired event.

## JEVIO EVENT-VIEWING GUI

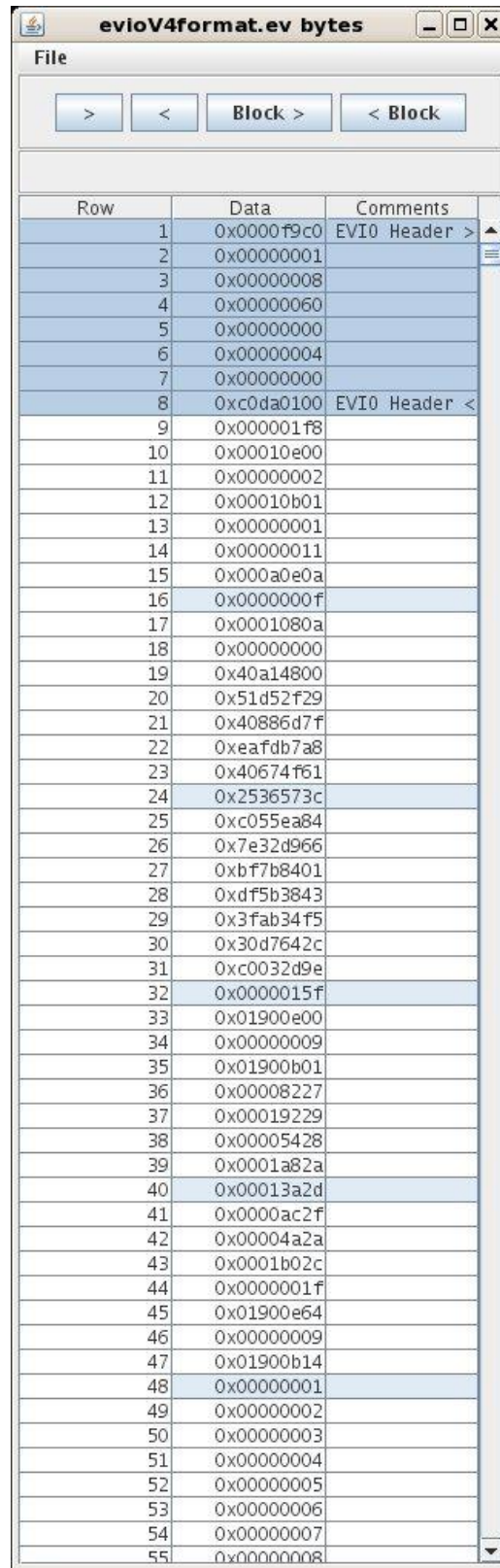
The box to its right, “Event Q”, shows different things depending on if the data source is a file, cMsg messages, or ET events. For files it shows the total number of events (in this case 10,000). For cMsg messages and ET events, on the other hand, events are continually arriving. In this case, “Size” shows the number of events currently in an internal queue. “Limit” allows the user to set the size of this internal queue, while “Clear” will remove all events currently in the queue. Once this queue is full, nothing else is added. The “Event #” controls can be used to switch between events in the queue.

Switching between the different event sources can be done in the “Event” menu item. When selecting a cMsg or ET source, the “Filter” menu is enabled. With this menu, the user can choose to look at control, partially-built physics, physics events, or any combination as well as the selecting the run type of interest.

Notice that above the data, there are boxes containing the event and dictionary sources. Beneath the data are boxes containing information about the selected data structure.

In addition to looking at evio events, this viewer is useful when debugging evio-related applications by looking at files directly. The data in a file are viewed as 32-bit integers in hex. This can be done with the “File” menu item selecting the “View File Bytes” option and producing the following window:

Figure 1.2: General file viewing window



The screenshot shows a window titled "evioV4format.ev bytes" with a "File" menu. Below the menu are four buttons: ">", "<", "Block >", and "< Block". The main area contains a table with three columns: "Row", "Data", and "Comments". The table lists 55 rows of hex data. Rows 1 and 8 are highlighted in blue. Row 1 has the comment "EVI0 Header >" and Row 8 has the comment "EVI0 Header <".

Row	Data	Comments
1	0x0000f9c0	EVI0 Header >
2	0x00000001	
3	0x00000008	
4	0x00000060	
5	0x00000000	
6	0x00000004	
7	0x00000000	
8	0xc0da0100	EVI0 Header <
9	0x000001f8	
10	0x00010e00	
11	0x00000002	
12	0x00010b01	
13	0x00000001	
14	0x00000011	
15	0x000a0e0a	
16	0x0000000f	
17	0x0001080a	
18	0x00000000	
19	0x40a14800	
20	0x51d52f29	
21	0x40886d7f	
22	0xeaadb7a8	
23	0x40674f61	
24	0x2536573c	
25	0xc055ea84	
26	0x7e32d966	
27	0xbf7b8401	
28	0xdf5b3843	
29	0x3fab34f5	
30	0x30d7642c	
31	0xc0032d9e	
32	0x0000015f	
33	0x01900e00	
34	0x00000009	
35	0x01900b01	
36	0x00008227	
37	0x00019229	
38	0x00005428	
39	0x0001a82a	
40	0x00013a2d	
41	0x0000ac2f	
42	0x00004a2a	
43	0x0001b02c	
44	0x0000001f	
45	0x01900e64	
46	0x00000009	
47	0x01900b14	
48	0x00000001	
49	0x00000002	
50	0x00000003	
51	0x00000004	
52	0x00000005	
53	0x00000006	
54	0x00000007	
55	0x00000008	

## REVISION HISTORY

Endianness of the data can be switched using the “File” menu and comments can be added by hand in the right-most column. The arrows move the view up or down by one screen’s worth. The block buttons hop to the next or previous evio-format block header. When an evio block header is found, it is labeled and selected as you see above.