

**Subject:** Re: Raw data copy to cache

**From:** Christopher Larrieu <larrieu@jlab.org>

**Date:** 05/27/2016 10:11 AM

**To:** David Lawrence <david.lawrence.nn@gmail.com>

**CC:** Paul Mattione <pmatt@jlab.org>, Sandy Philpott <philpott@jlab.org>, Curtis Meyer <curtis.meyer@cmu.edu>, "Mark M. Ito" <marki@jlab.org>, Chip Watson <watson@jlab.org>, Graham Heyes <heyes@jlab.org>, Ying Chen <ychen@jlab.org>

Okay, this should happen automagically henceforth.

On 05/27/2016 09:52 AM, David Lawrence wrote:

Hi Chris,

I would make it slightly more restrictive to:

hd\_rawdata\_?????\_00[01234].evio

(don't forget file "0")

Regards,  
-David

On May 27, 2016, at 8:41 AM, Christopher Larrieu <[larrieu@jlab.org](mailto:larrieu@jlab.org)> wrote:

So is it accurate to say any hall D raw data file matching \*\_00[1234].evio should be auto-cached?

Chris

On 05/26/2016 05:42 PM, Paul Mattione wrote:

The first five files of every run: ~100 GB per run (~ 2hrs). They will be written to folders like:

/mss/halld/RunPeriod-2016-02/rawdata/Run010873/

and will have names like \*\_000.evio, \*\_001.evio, ... \*\_004.evio

- Paul

On May 26, 2016, at 1:54 PM, Christopher Larrieu <[larrieu@jlab.org](mailto:larrieu@jlab.org)> wrote:

I do already have a mechanism to auto-cache files as they are written to tape. I just need a clear rule for deciding which files to include.

On 05/26/2016 11:42 AM, David Lawrence wrote:

Hi Sandy,

Here's a follow up to our discussion on copying Hall-D raw data to cache disk at the time it is written to tape. I'll recap a little to bring others up to speed.

1. Prior to the last run, it was suggested we write raw data to the write-through cache rather than to the staging disk in order to get it on tape. The benefit would

be that we would save reading from tape just a short time later to get it into cache

for our "offline monitoring" jobs.

2. We decided not to change anything so close to the beginning of the run and to explore this during the summer, prior to the Fall 2016 run. (i.e. now)

3. After further discussion within Hall-D we realized that what we really need is

just the first 5 files for each run to be cached since that is all the offline monitoring looks at anyway.

4. We also don't want to churn through all of the cache disk space, displacing other files, or filling up quotas by writing the entire data set when only a small fraction needs to be available on cache right away.

5. Our thought is to keep writing raw data to the staging disk as we are, but have

some other piece of software on your end move the first 5 files of a run over to the cache disk to make it immediately available.

6. As you noted, the pinning system will probably play an important role here. The different colors of pins will require some thought be put into making the system robust.

So, hopefully this gives an idea of what we are wanting to do. You said Chris may

already have a mechanism to do this sort of thing. I'm thinking we can explore this via e-mail thread first and arrange a meeting later if need be.

Regards,  
-David