

LCLS-1 DAQ Tier-1 Troubleshooting

Contents:

- [Recovering from errors: restarting the DAQ](#)
- [Recovering from errors: restarting AMI \(the client\)](#)
- [Troubleshooting the DAQ](#)
 - [Ami does not work: the DAQ is fine after a restart, but I see no updates/detectors in ami](#)
 - [The DAQ shows "NO CONNECT" in procStat:](#)
 - [The DAQ shows an error message, requesting a restart, indicating a given IP as culprit](#)
 - [One of my DAQ devices has a problem \(damage,...\):](#)
 - [My ipimb has an issue:](#)
 - [My OPAL damages on every shot \(but configures\):](#)
 - [My <detector> damages often \(but configures\):](#)
 - [My BLD has an issue:](#)
 - [My detector runs without errors, but I can't see any beam:](#)
 - [My data does not seem to be moving well:](#)
 - [I have included a recorder source and the DAQ won't start:](#)
 - [The IOC recording seems to work fine, but my output file is of size 0:](#)
 - [The XTCav recording does not work:](#)
 - [Timetool](#)
 - [Acqiris](#)
 - [Rayonix:](#)
 - [Logfiles:](#)
 - [Logbook:](#)
 - [Technical note:](#)

Recovering from errors: restarting the DAQ

Use the "restartdaq <-w>" command. If you only want to stop the DAQ, call "stopdaq".

The command "wheredaq" will tell if you an instance of the DAQ is currently running (e.g. on the hutch-machine when you are in the control room).

Recovering from errors: restarting AMI (the client)

Use the "startami" command. This will start a second client if you are not on the main DAQ machine and it will restart the DAQ AMI client if run on the main DAQ machine (after asking for confirmation, so you will need to use a terminal). Should the server side be unhappy, you will need to restart the DAQ.

Troubleshooting the DAQ

Ami does not work: the DAQ is fine after a restart, but I see no updates/detectors in ami

1) Make sure that **all** dss nodes are selected in the daq partition. If you needed to take a node out due to problems, you need to edit the <hutch>.cnf file. You can use "serverStat <ip>" to get the name for the node that causes a problem and then edit the list of dss_nodes to exclude this node. If the problematic node is the last one, you might see that you have to reselect the Bld upon a restart. This means that the DAQ will make you allocate twice (the first time it'll fail with a complaint about a Bld change).

2) A second cause for this problem can be that the calib-dir is not mounted on the mon-nodes anymore. ami will look for the pedestal files here, so it will take a long time until it times out and will start, but w/o the detectors. You can check if that is the case by looking at an ami log file (e.g. for ami0-0) or ssh to a mon-node (daq-HUTCH-monXX) in use and check if you can see the pedestals files in /sdf/data/lcls/ds/<hutch>/<expname>/calib/<dettype>/.... If the calib directory is not visible from the mon-nodes, a temporary solution is to use a local version of the calib folder. To do so, copy the calib directory into ~<xxxopr>/tmp_calibdir/<hutch>/<exp>/. This can usually be done from a psana node, but you need to be a member of the xxxopr group to be allowed to write in ~xxxopr directories. Then in the <hutch>.cnf file, edit the ami-opts:

Comment the following lines:

```
if slow_readout_mode != 1:
    ami_opts = '-a only -R -L '+plugin_path+'/libtimetoolddb.so -E '+expname
else:
    ami_opts = '-a only -L '+plugin_path+'/libtimetoolddb.so -E '+expname
```

And uncomment:

```
#ami_opts = '-a only -R -L '+plugin_path+'/libtimetoolddb.so -E '+expname+' -c /cfs/home/opr/xppopr/tmp_calibdir'
```

Note that this is a temporary solution. Notify the ECS point of contact who will escalate or resolve the calib mount issue.

The DAQ shows "NO CONNECT" in procStat:

use "serverStat <DAQ device alias> <command>". "cycle" will power cycle the node with some time between off/on in the script. It will tie up the terminal, so if you have to deal with several nodes, you can also call "serverStat <DAQ device alias> off" and "serverStat <DAQ device alias> on" explicitly. Remember to wait a few second between off/on. After the script returns from turning the node(s) on, continue to **run "serverStat <ip/node name>" until both pings work. If you can ssh into the node(s), you can restart the DAQ.**

The DAQ shows an error message, requesting a restart, indicating a given IP as culprit

Use "serverStat <ip>" to check if both interfaces of the node in question are up. This script will also tell you which machine has the issue.

One/both of the pings fails: use "serverStat <ip/node name> cycle" to power cycle the machine. After the script returns, continue to run "serverStat <ip /node name>" until both pings work. If you can ssh into the node, you can restart the DAQ.

Otherwise: decide if you'd rather restart the DAQ and hope for the best. Power cycling a machine takes a few minutes.

The IP is a dss-node: here you have an additional option: you can edit the <hutch>.cnf file to take out the node: look for "dss_nodes = [...]" and take out the problematic node.

Depending on the data rate, you can run with 2 or 3 nodes (cspad + other detectors: 3 nodes, two EPIX: 2 nodes). As we run all the data into a single ami session and the best mapping allows max one ami node/dss node, you have less ami power if you have less dss nodes.

One of my DAQ devices has a problem (damage,....):

use "serverStat <DAQ device alias>" to check on the health of the node. Most likely it is prudent to power-cycle this node. Does this not help, you should power cycle the detector/camera/device itself as well.

This assumes that the detector has recently worked. Otherwise, if the detector configures, but does produce damage an all events, check the trigger timing settings here:

[Detector timing settings](#)

Should your detector not configure, it is either not turned on, needs power cycling or some cable is not patched correctly. Please let CDS folks know which detectors you would like to use so we can test them beforehand.

My ipimb has an issue:

Troubleshooting ipimbs is described on this page:[Troubleshooting for Controls IPIMB and Wave8s](#)

My OPAL damages on every shot (but configures):

The possible problems are that there is an issue with the EVR, the RCX boxes (data coming back from the camera) or the EDT card.

If no other physically triggered devices on the same EVR work either (other OPALs, Alviums, Zylas or Jungfraus), the problem is likely the EVR: as an expert, I would run evrsnoop. The second step is to power cycle the master (using serverStat or another method).

Should other triggered devices run, you can choose to look at the LEDs on the camera: if they are solid green, the boxes/fibers connecting them are not the issue. Should a camera have just run recently, it is likely NOT the RCX box.

The third suspect is the EDT card. Try to power-cycle the server: with luck, this will reset the card and make things work again. serverStat will take the DAQ alias as argument, no need to peruse the cnf-file.

My <detector> damages often (but configures):

One of the things to check here is the EVR: first, run 'evrsnoop' on the EVR host machine (typically dad-<instrument>-master).

The command is

```
/cds/group/pcds/dist/pds/current/build/pdsapp/bin/x86_64-rhel7-opt/evrsnoop -h
```

-h will list all options, you will need to specify which EVR with the '-r a/b/c' option and which event codes you want to check with the -o <code> option. I would usually use event code 40 as that should always come and you can check if the fiducial differ by 3. The output should look something like this:

After confirming that the EVR receives the event codes, check that the cable at the back of the card is properly seated and that the correct channel is running to your detector.

```
Received Fiducial 010353 Event code 40 Timestamp 11828
Received Fiducial 010356 Event code 40 Timestamp 11828
Received Fiducial 010359 Event code 40 Timestamp 11828
Received Fiducial 01035c Event code 40 Timestamp 11828
```

After confirming that the EVR receives the event codes, check that the cable at the back of the card is properly seated and that the correct channel is running to your detector.

My BLD misses a source:

You will have to fix this with the mask used in the bld-process in the cnf file.

The [address of the sources is here](#).

My BLD has an issue:

For LCLS-I, we use the last server in the list of used dss-nodes to run the BLD process. Check out of the data comes in, you can ssh to that node and call:

```
/reg/g/pcds/dist/pds/current/build/pdsapp/bin/x86_64-rhel7-opt/bldServerTest -i <Interface name> -a 239.255.24.<address>
```

To find the address of choice, look at "My BLD misses a source".

If you do not see any damage reported, but cannot see data in ami: make sure that see event code 140 - this (or a subset) produces the BLD (depending on the source).

To figure out what the interface name is, call `"/sbin/ifconfig -a"`. You want the name for the CDS interface. Recently we have seen cases where the BLD came over the FEZ interface only.

In 120 Hz operation, a difference in fiducials of 3 is expected. Some more detail can be found at the [BLD DAQ](#) page.

If data are not coming through, first check that the BLD source is properly selected in the DAQ partition.

A common problem with ebeam BLD is that the L3Energy calculation will not update if the accelerator has changed its beam energy configuration and not yet updated alarm limits on data that goes into the calculation. This should be evident in the "damageMask" of ebeam BLD. If so, call ACR.

For LCLS-II DAQ, see [L2SI DAQ BLD reception](#)

My detector runs without errors, but I can't see any beam:

Trusting that there actually are photons on the detector that could be seen, the second point are the timing settings. Find the values you should use here:

[Detector timing settings](#)

Go to the EVR configuration (with the exception of pgp triggered devices as the EPIX), if you are using aliases in your DAQ, it should be straightforward to find the right trigger channel. Look at all available EVR cards. For the run trigger setting for e.g. CsPad detectors, it might be best to contact your POC.

My data does not seem to be moving well:

The *Files-in-WAIT-state* table in the [PSDM-Mover-Tape](#) Grafana dashboard shows all files that are waiting for the movers to be transferred including files that are being transferred.

For any issues and question email and/or call the data-management group, depending on time-of-day and urgency

[Take out the nodes with the issue, assuming your problem is limited to a single node. If it's wider spread, it might warrant a call.]

I have included a recorder source and the DAQ won't start:

Most likely, the recorder process cannot connect to the node that you are trying to run the recording process on. This node depends on the source and is listed in your .iocrc file. You can use serverStat to check on the server health and power cycle if appropriate.

The IOC recording seems to work fine, but my output file is of size 0:

This means your recording process is up and you can connect to the PVs. If this PV does not have well-formed timestamps when you request this (necessary for cameras!), the data the process would want to record cannot be found as the timestamp us used to determine what should be recorded.

The XTCAV recording does not work:

First, please make sure the image in the camViewer updates. This means the IOC is up and running. The recorder process uses a special interface on a special machine, so in addition this this, that machine needs to be up (see recorder source and DAQ won't start). More details can be found here:

[LCLS1 XTCAV Controls Recorder Troubleshooting](#)

Timetool

Assuming your timetool camera (either an OPAL or now an Alvium works), some details that need to be set are described at [Timetool Troubleshooting](#)

Keep in mind that too large an ROI can lead to problems for 120Hz running (at least w/ an OPAL). We need to test if unpacking&processing the Alvium causes issues & when.

Acqiris

Some things to check if the Acqiris is causing problems:

- Reduce the total number of sample (especially at higher rates)
- Make sure the acquisition window (sampling*number of samples) is not too close to the trigger rate.
- Better cooling

Rayonix:

- [Rayonix MX\[340/170\]-HS](#) (expert information)
- Taking a dark
 - In the DAQ, "shutdown" first, then go to 'edit' -> 'rayonix'.
 - Make sure binning is what you are going to use to take data
 - Select on the menu 'Update background on config' and click 'apply'. Run the DAQ for few seconds.
 - Go back to the 'expert mode' and select on the menu 'keep current background' and click apply.
 - If this doesn't work go to the terminal where the Rayonix server is running (ask Silke), stop it (CTRL-X) and restart it after about 10 sec (CTRL-R). Restart the DAQ.
- 'rayonix' will start a session as huser on the Rayonix server. capXure can be run here and the DAQ process also runs in a procServ on this machine
- Trouble shooting: on Rayonix machine - skip first 3 lines if the DAQ process is already running.
 - `cd ~/slac`
`./startDaqInterface`
Now you can connect to the _active_ session using telnet localhost 30099.
This window behaves kind of like an IOC. ctrl+x to kill the process (necessary in several cases....), ctrl+R to restart things. It is not running auto restart as the detector interfaces sometimes needs more time to be available again than auto restart will give it. After the initial start I have found that typically I needed to restart the process (ctrl+x; wait; ctrl+R)
What you should see (after a bit) are either lines showing the detector status:
13_11:46:45:2015-Nov-13 11:46:45,[STATUS],[STATUS_DATA],-79.95,-79.35,-79.95,-79.75,-156.15,52.25,-860.95,-870.25,
-774.25,33.05,49.85,49.85,276,273,275,36,0,0,1200,00:00:04
or, if you are already running it in the DAQ, one line for each image you record:
12_15:30:30:2015-Nov-12 15:30:30,[VERBOSE],RxDetector::FrameAcquisitionFrameProcessor(83, 0x7f55d403bc20,0x7f55d42d2230)
- starting

12_15:30:30:FrameReady: Frame #84 size=7372800 timestamp=59989 ms
In DAQ config, go to Rayonix settings, choose binning you need (10 Hz at 2x2, 120 Hz at 10x10)
- Trouble shooting (includes controller reboot)
sometimes, the detector reports wrong status information and believe many state bits are flipping. You can see either that some of the temperature are reading unreasonably numbers without any change from the "usual" ones (you should see 4 times ~ -80 and 2 times ~-110)
13_11:38:45:2015-Nov-13 11:38:45,[STATUS],[STATUS_DATA],-79.95,-79.35,-79.95,-79.65,-109.25,
-109.95,34.05,42.65,45.45,34.75,49.85,49.85,276,273,275,37,0,0,1200,20:50:31

13_11:39:45:2015-Nov-13 11:39:45,[STATUS],[STATUS_DATA],-79.95,-79.35,-79.95,-79.65,-109.25,
-109.95,34.05,42.35,45.45,34.85,49.85,49.85,276,273,275,36,0,0,1200,20:51:31

13_11:40:45:2015-Nov-13 11:40:45,[STATUS],[STATUS_DATA],-79.95,-79.35,-79.95,-79.75,-156.15,52.05,-860.85,-870.25,
-774.15,34.75,49.85,49.85,276,273,275,37,0,0,1200,00:00:04

13_11:41:45:2015-Nov-13 11:41:45,[STATUS],[STATUS_DATA],-79.95,-79.35,-79.85,-79.65,-156.15,52.05,-860.85,-870.25,
-774.15,33.95,49.85,49.85,276,273,275,36,0,0,1200,00:00:04

To recover from that, you need to stop the DAQ process (ctrl+x in the telnet localhost 30099 terminal window) and open the Rayonix software cap xure.

Reboot the detector controller (there is a button for that). Re-enable cooling. You might have to quite this process and start again to see the now reasonable temperatures. Once this looks fine, you can quit this software and restart the DAQ process: (ctrl+R in the telnet localhost 30099 terminal window)

In the DAQ, "shutdown" first, then go to 'edit' -> 'rayonix'.
Make sure binning is what you are going to use to take data
Select on the menu 'Update background on config' and click 'apply'. Run the DAQ for few seconds.
Go back to the 'expert mode' and select on the menu 'keep current background' and click apply.
If this doesn't work go to the terminal where the Rayonix server is running (ask Silke), stop it (CTRL-X) and restart it after about 10 sec (CTRL-R). Restart the DAQ.

Logfiles:

Before calling the DAQ, you may want to look at the log files of the process with a problem. Be aware that some of them have concerning-sounding messages that are NOT actually concerning, so compare with log-files written during problem-free times.

Logfiles can be found at:

/reg/g/pcds/pds/<hutch>/logfiles/<year>/<month>

and have file name like:

<DAQ_start_time>_<hostname>"<process_name>.log

e.g.

08_15:17:03_daq-mec-mon03:ami1-0.log

Logbook:

The DAQ is capable of storing a snapshot of a list of EPICS PVs in the Logbook when a run is recorded. The "offlineobs" process must be included (and not commented out) in the DAQ cnf file in order for this feature to work. The list of PVs to be recorded is specified by the -V option of offlineobs. In the xpp.cnf example below, the list of PV names is found in logbook.txt.

If some of the desired EPICS PVs are recorded in the Logbook while others are missing, make sure that logbook.txt (in this example) includes all of the PV names of interest.

xpp.cnf

```
misc_path = '/reg/g/pcds/dist/pds/xpp/misc'
procmgr_config_base = [
    . . .
    # MONITORING
    { id:'offlineobs', flags:'sp', cmd:pdsapp_path+'/offlineobs -g -P XPP -E '+expname+' -L '+misc_path+'/.
    offlinerc -V '+misc_path+'/logbook.txt'},
    . . .
]
```

Use 'caget' or 'pvget' to verify that a PV is available (and spelled correctly).

Remember that you must run the DAQ with recording enabled in order to update the Logbook.

Technical note:

"serverStat" at this moment works in all hutches when using the machine name or IP, but the "DAQ alias" interpretation feature might not quite work. We hope to improve on this soon.

[Expert Troubleshooting \(limited permissions\)](#)