Hall C HV CSS User Manual

Tyler Lemon

June 6, 2019

Contents

I. Overview

The Hall C HV CSS screens are built in Control Systems Studio (CSS), an Eclipse-based collection of tools created to help control large-scale systems using EPICS channel access. Part of CSS is an extension called “Best Operator Interface (OPI) Yet”, or CSS-BOY. CSS-BOY utilizes the tools available in Eclipse to create graphical user interfaces (GUIs, or screens). Because Eclipse is Java-based, extensions and scripts can be executed from CSS’s runtime environment, allowing developers to customize the environment to the needs of the control system.

Before the development of Hall C’s HV CSS screens, all HV monitoring and controls were done through Tcl/Tk scripts. These scripts created GUIs upon execution for users to control/monitor Hall C’s high voltage systems. CSS was selected as a replacement for Tcl/Tk given CSS’s native ability to communicate over EPICS and the ability to generate static screens that are stored locally and do not need to be re-generated every time they are opened.

Hall C’s HV CSS screens, accessible using a start-up script on any Hall C subnet PC (Part II), provide monitoring and controls capabilities for all HMS and SHMS detectors (Part III). The CSS screens can be updated at any time by modifying its configuration file and re-running the script developed to generate the screens (Part IV). The CSS controls system also includes the ability to backup and restore parameters for HV channels (Part V).

II. Start-up script

A shell script was created to open the screens in the appropriate workspace and in runtime mode. The shell script, called *go_hallc-hv*, is located at /home/cdsg/ on *cdaql2* and contains the command below to open the screens.

```
/home/cdsg/tlemon/css-4.5.0/css -data /home/cdsg/CSS-Workspace/HV_release2/ -nosplash
```

Components of the command are as follows:

```
/home/cdsg/tlemon/css-4.5.0/css
```

- CSS executable with path to its location on *cdaql2*.

```
-data
```

- Command line option allowing user to specify which workspace to open.

```
/home/cdsg/CSS-Workspace/HV_release2/
```

- Path to workspace containing CSS-BOY screens and scripts.

```
-nosplash
```

- Option to suppress CSS's default splash screen as it loads.

If the user is on Hall C counting house PCs, a second executable has been created called *go_newhv* that utilizes secure shell to allow *go_hallc-hv* to be executed from remote hosts.

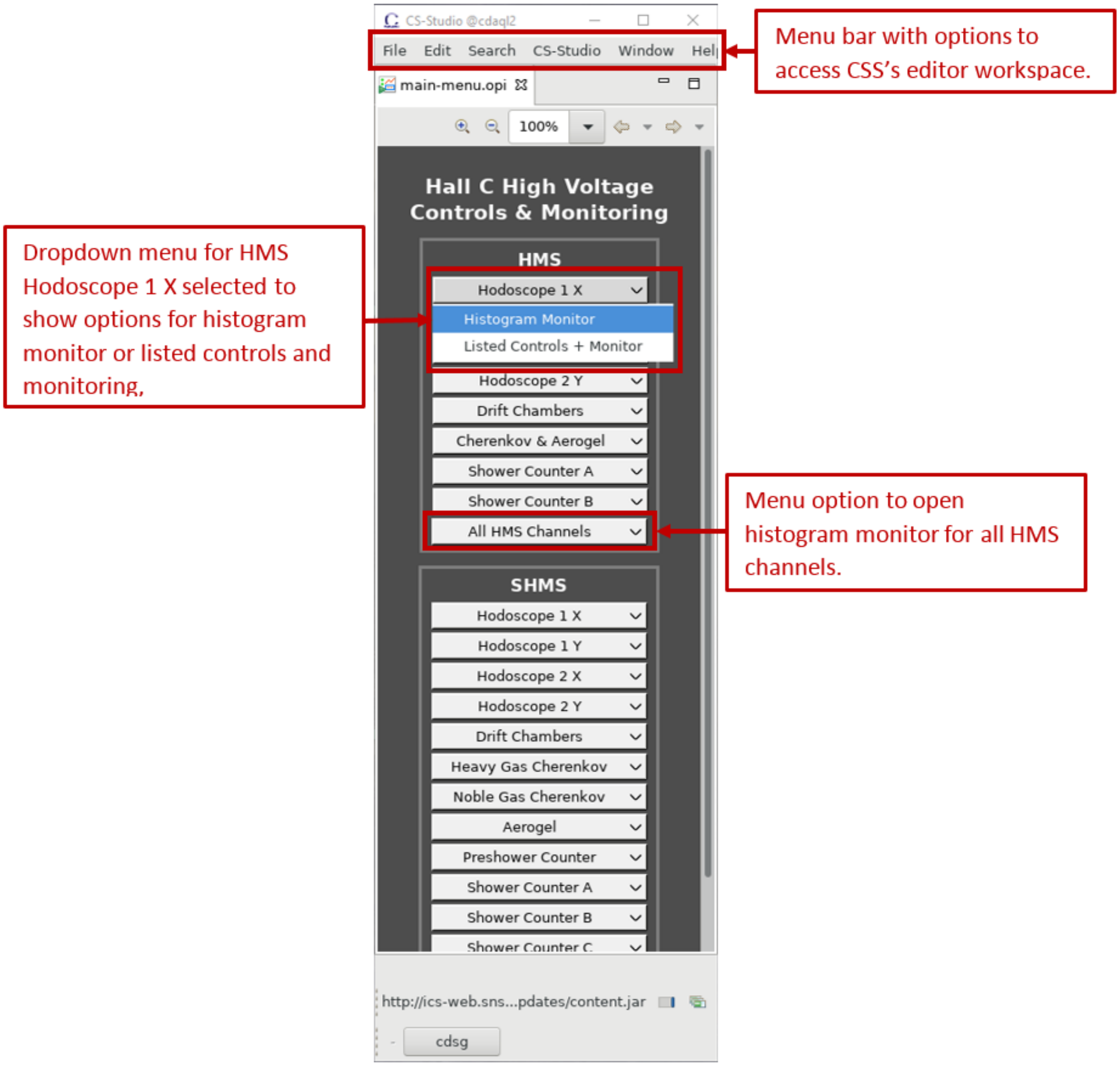
Currently, *go_hallc-hv* only allows one user to use the screens at a time. The next version of the start-up script that will alleviate this issue is under development.

III. CSS Screen Description

i. Main Menu

When the *go_hallc-hv* is executed, CSS opens and displays the main menu in runtime mode. If this is not the case, steps to resolve this issue are in Part VI. On the main menu, all detector systems are listed as dropdown boxes that allow the user to open either a histogram monitoring screens or a listed view controls and monitoring screen. There are also menu options to open histogram monitoring screens for all HMS channels or all SHMS channels.

From this main menu, if needed, the user can access CSS's editor workspace using the menu bar at the top of the screen.



Menu bar with options to access CSS's editor workspace.

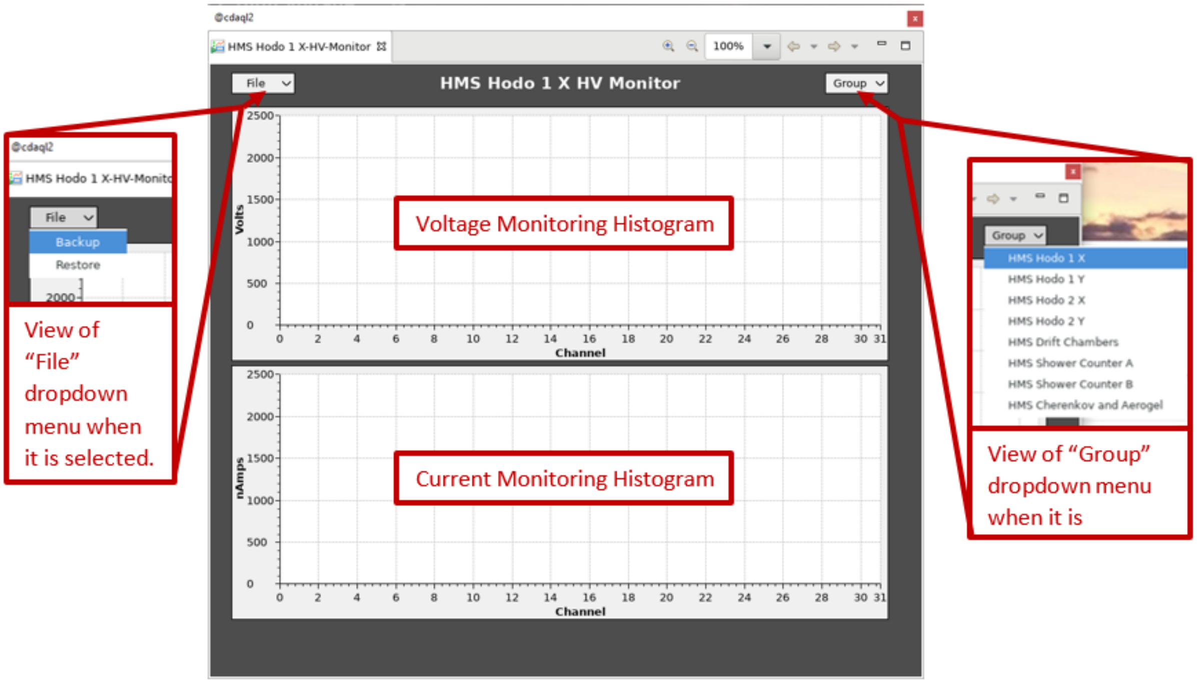
Dropdown menu for HMS Hodoscope 1 X selected to show options for histogram monitor or listed controls and monitoring,

Menu option to open histogram monitor for all HMS channels.

ii. Histogram Monitor Screen

For each detector on HMS and SHMS, one of the options for the screen to select from the main menu is the histogram monitor. This screen plots all of a detector's channels' voltage and current in a bar plot, or histogram, view. This screen is used only for monitoring and has no control functionality. When the screen is running and channels are on (unlike screenshot below from when all channels were off), the histogram will show one bar for each channel. These screens are used to visually check that all channels are on and there are no channels that are significantly lower or higher than the rest.

From each histogram monitoring screen, there are two additional dropdown menus. These two dropdown menus are displayed on every histogram and list-view screen. One menu, labeled "File", on the left opens the backup and restore GUIs. The second menu, labeled "Group", on the right opens list-view monitoring and controls for any detector in the same spectrometer.

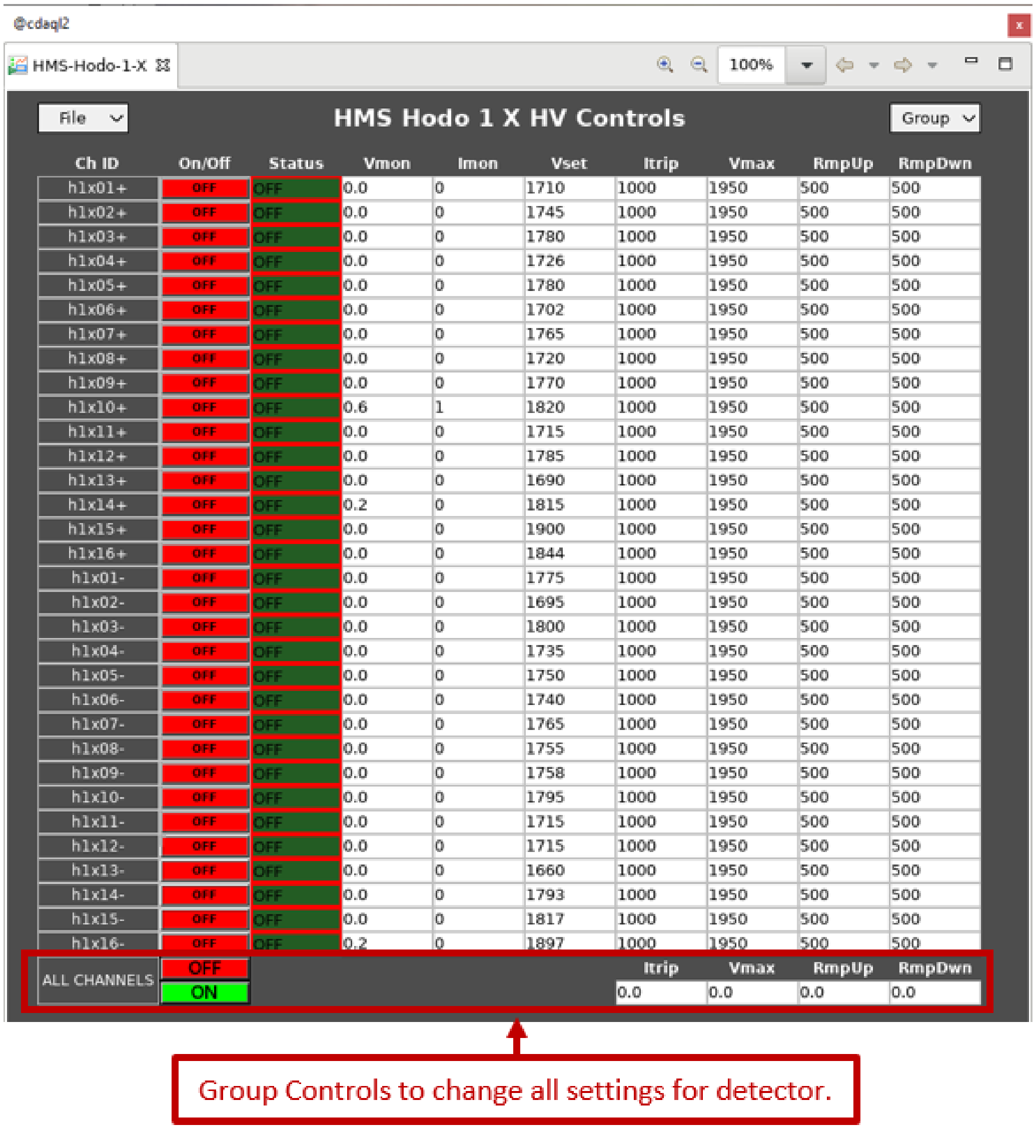


iii. List Monitor and Controls Screen

The second screen available for each individual detector is a list view of controls and readbacks for each channel. Each channel is displayed in its own row with its ID name (Ch ID), on/off control (On/Off), status readback (Status), voltage readback (Vmon), current readback (Imon), voltage setpoint readback/control (Vset), current trip point readback/control (Itrip), software voltage limit readback/control (Vmax), ramp up rate readback/control (RmpUp), and ramp down rate readback/control (RmpDwn).

In addition to individual channel controls, there are controls at the bottom of the screen to set values for all channels displayed. There are group controls for channel on/off, current trip point, software voltage limit, ramp up rate, and ramp down rate. These controls utilize CSS's ability to embed Java or Python scripts in a control to change all channels' parameters from one control. When these controls are used, a confirmation message is displayed requiring the user to confirm that they want to change that parameter for all channels of that detector.

Also on the list view screens are the same two "File" and "Group" dropdown menus for accessing other detectors and the backup/restore GUIs.



HMS Hodo 1 X HV Controls

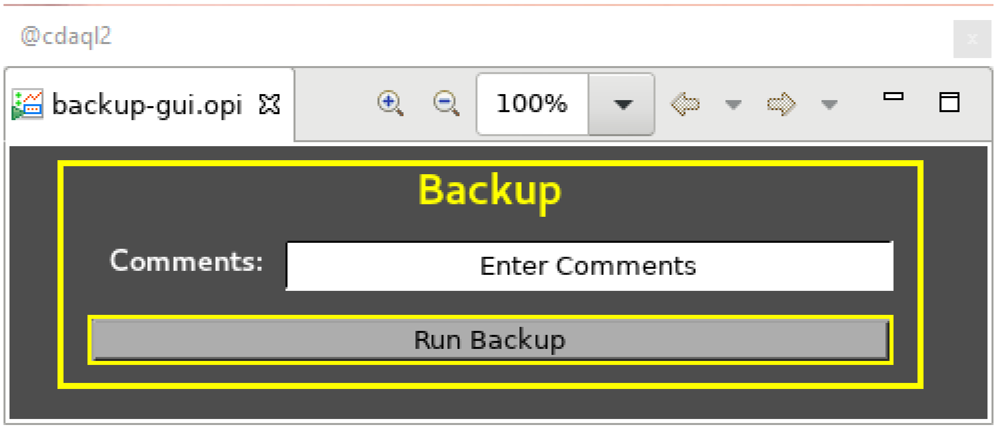
Ch ID	On/Off	Status	Vmon	Imon	Vset	Itrip	Vmax	RmpUp	RmpDwn
h1x01+	OFF	OFF	0.0	0	1710	1000	1950	500	500
h1x02+	OFF	OFF	0.0	0	1745	1000	1950	500	500
h1x03+	OFF	OFF	0.0	0	1780	1000	1950	500	500
h1x04+	OFF	OFF	0.0	0	1726	1000	1950	500	500
h1x05+	OFF	OFF	0.0	0	1780	1000	1950	500	500
h1x06+	OFF	OFF	0.0	0	1702	1000	1950	500	500
h1x07+	OFF	OFF	0.0	0	1765	1000	1950	500	500
h1x08+	OFF	OFF	0.0	0	1720	1000	1950	500	500
h1x09+	OFF	OFF	0.0	0	1770	1000	1950	500	500
h1x10+	OFF	OFF	0.6	1	1820	1000	1950	500	500
h1x11+	OFF	OFF	0.0	0	1715	1000	1950	500	500
h1x12+	OFF	OFF	0.0	0	1785	1000	1950	500	500
h1x13+	OFF	OFF	0.0	0	1690	1000	1950	500	500
h1x14+	OFF	OFF	0.2	0	1815	1000	1950	500	500
h1x15+	OFF	OFF	0.0	0	1900	1000	1950	500	500
h1x16+	OFF	OFF	0.0	0	1844	1000	1950	500	500
h1x01-	OFF	OFF	0.0	0	1775	1000	1950	500	500
h1x02-	OFF	OFF	0.0	0	1695	1000	1950	500	500
h1x03-	OFF	OFF	0.0	0	1800	1000	1950	500	500
h1x04-	OFF	OFF	0.0	0	1735	1000	1950	500	500
h1x05-	OFF	OFF	0.0	0	1750	1000	1950	500	500
h1x06-	OFF	OFF	0.0	0	1740	1000	1950	500	500
h1x07-	OFF	OFF	0.0	0	1765	1000	1950	500	500
h1x08-	OFF	OFF	0.0	0	1755	1000	1950	500	500
h1x09-	OFF	OFF	0.0	0	1758	1000	1950	500	500
h1x10-	OFF	OFF	0.0	0	1795	1000	1950	500	500
h1x11-	OFF	OFF	0.0	0	1715	1000	1950	500	500
h1x12-	OFF	OFF	0.0	0	1715	1000	1950	500	500
h1x13-	OFF	OFF	0.0	0	1660	1000	1950	500	500
h1x14-	OFF	OFF	0.0	0	1793	1000	1950	500	500
h1x15-	OFF	OFF	0.0	0	1817	1000	1950	500	500
h1x16-	OFF	OFF	0.2	0	1897	1000	1950	500	500

ALL CHANNELS	OFF	Itrip	Vmax	RmpUp	RmpDwn
	ON	0.0	0.0	0.0	0.0

Group Controls to change all settings for detector.

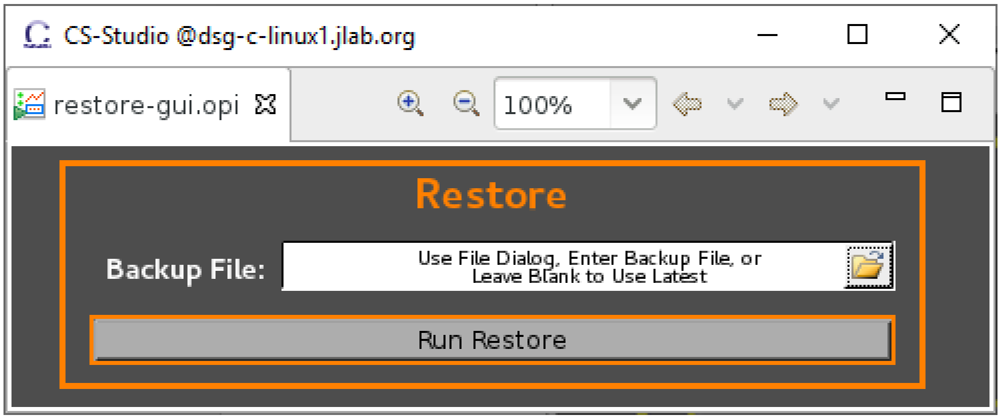
iv. Backup GUI

If the user selects “Backup” from the “File” menu, the backup menu is opened. This screen contain a text box that can be used to enter comments into the file created and a Boolean control button that the user must click to initiate the backup. Further details on how the backup is performed are in Part V.



v. Restore GUI

If the user selects “Restore” from the “File” menu, the restore menu is opened. This screen contain a dialog box that can be used manually enter the file to perform the restore from or to open a file explorer window to select the file to restore from. The user may also leave the dialog box empty to automatically perform the restore using the latest backup file. Further details on how the restoration is performed are in Part V.


CS-Studio @dsg-c-linux1.jlab.org
restore-gui.opi
100%
Restore
Backup File:
Use File Dialog, Enter Backup File, or
Leave Blank to Use Latest
Run Restore

IV. Running Script to Generate Screens

The script to generate all CSS-BOY screens for high voltage monitoring and controls, called *TCL2CSS*, is Python based and takes advantage of CSS-BOY's OPI file type being a variation of XML that can be read and edited as text. *TCL2CSS* uses "widget templates" ("widget" is CSS's name for any item placed on to a screen) to format and place controls and indicators on a CSS-BOY screen. Widget templates are text constants extracted from an OPI file with key parameters, such as a widget's x-position, y-position, height, width, and PV name, replaced with keywords that can then in turn be replaced by script. These PV names and properties are parsed by *TCL2CSS* from two configuration files, HV.hvc and HV.group, which are from the old Tcl/Tk high voltage monitoring system. *TCL2CSS* cannot generate CSS-BOY screens without these two files.

TCL2CSS is executed via command line and allows the user to declare which directory contains HV.hvc and HV.group and which directory to write the resulting CSS-BOY files to. *TCL2CSS* also has the ability to generate channel mapping reference documentation and EPICS Alarm Handler (ALH) configuration files based on input options. Complete usage information on *TCL2CSS* is below.

```
python tcl2css.py [-m] [-a] dir [write]
```

```
[-m]
```

- Optional argument to output channel_map, group_map, and HV.alhConfig.

```
[-a]
```

- Optional argument to output *only* HV.alhConfig.
- If option is used, no other files will be generated.

```
dir
```

- Mandatory argument for directory containing HV.hvc and HV.group used to make the CSS-BOY screens.

```
[write]
```

- Optional argument for directory to write resulting CSS-BOY files to.
- If argument is not used, resulting files will be written to user's current working directory.

V. Backup and Restore Script

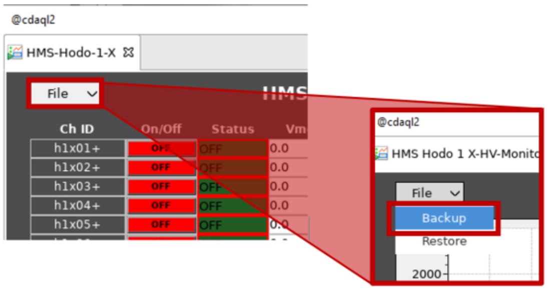
The backup and restore functionality of the HV CSS-BOY screens is performed by a Python script that uses the *pyepics* module (an interface to the Channel Access library for Python) to read/write values from/to PVs. In order to use the backup/restore functions of the CSS screens, *pyepics* must be installed on the PC being used (See Part VI for help installing *pyepics*).

There is one script for both backup and restore functions. Which function to use is automatically passed to the backup/restore program by CSS depending on the GUI used to execute the script.

Rather than allowing users to select the detector to backup or restore, when the backup/restore program is executed in backup mode, all HMS and SHMS high voltage channels are backed up. This ensures that there is always a backup of all channels that can in turn be used by *TCL2CSS* to rebuild all CSS-BOY high voltage controls and monitoring GUIs.

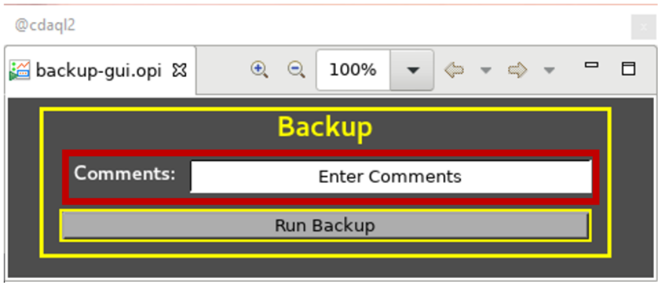
i. Backup Procedure

1. Open any detector’s controls/monitoring screen.
2. Use “File” drop down menu to select “Backup”.

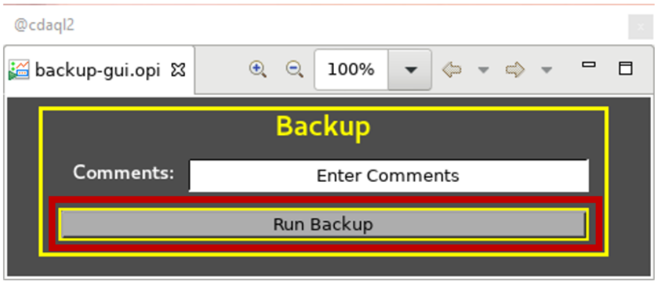


Ch ID	On/Off	Status	Vm
h1x01+	OFF	OFF	0.0
h1x02+	OFF	OFF	0.0
h1x03+	OFF	OFF	0.0
h1x04+	OFF	OFF	0.0
h1x05+	OFF	OFF	0.0

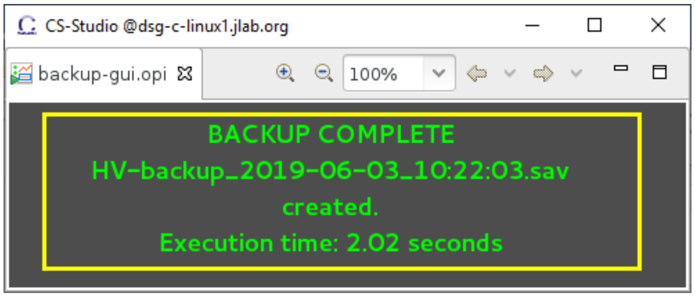
3. Enter any comments into the “Comments” text box on the backup GUI.



4. Click “Run Backup” button.



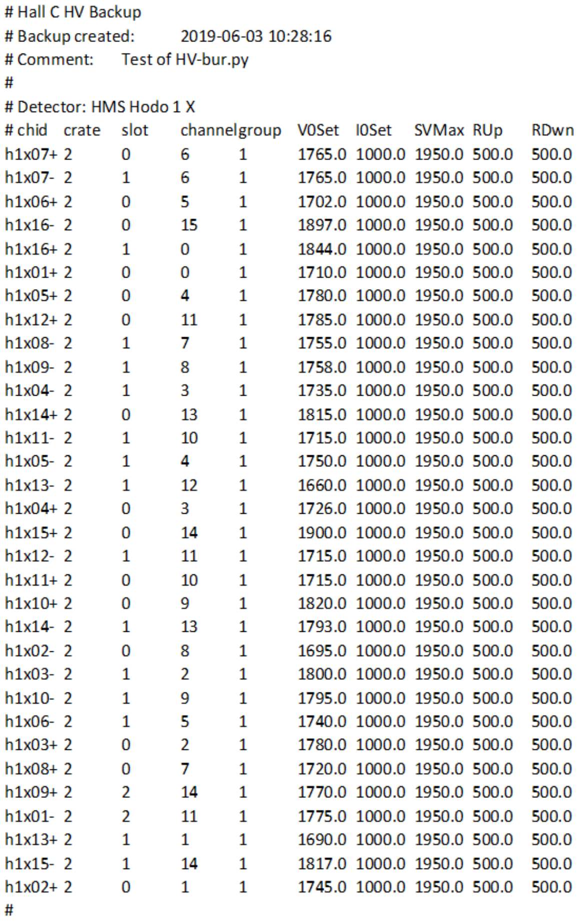
Upon successful backup, the GUI prints the success message with the name of the resulting backup file and the time required to complete the backup.



BACKUP COMPLETE
HV-backup_2019-06-03_10:22:03.sav
created.
Execution time: 2.02 seconds

ii. Backup File Format

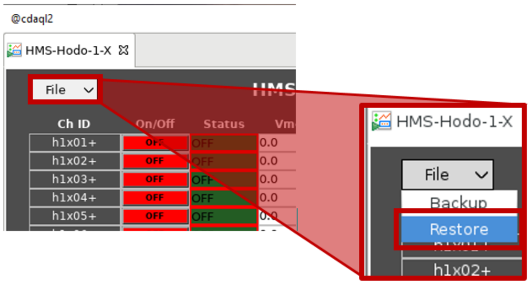
The resulting backup file's name is automatically contains the date and time of the backup to avoid any ambiguity about which file is most recent and eliminates the possibility of overwriting a previous backup file. The backup file contains the data and time of the backup, any comment entered by the user, and all channels' settings. The backup data is formatted into rows based on channel with the channel's properties separated by tabs. This creates a document that is readable in any text editor. Any line in the backup file that starts with a "#" character is ignored by the backup/restored program, allowing users to comment out channels or to add other comments between channels. Below is a portion of the backup file showing the overall header for the file and the backup data for HMS Hodoscope 1 X.



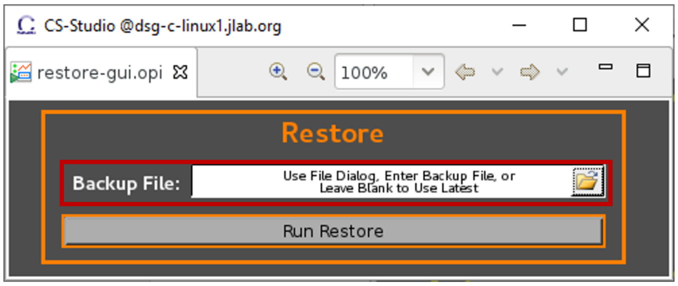
```
# Hall C HV Backup
# Backup created:       2019-06-03 10:28:16
# Comment:      Test of HV-bur.py
#
# Detector: HMS Hodo 1 X
# chid   crate  slot   channelgroup  V0Set  I0Set  SVMax  RUp    RDwn
h1x07+ 2       0      6      1      1765.0 1000.0 1950.0 500.0  500.0
h1x07-  2      1      6      1      1765.0 1000.0 1950.0 500.0  500.0
h1x06+ 2       0      5      1      1702.0 1000.0 1950.0 500.0  500.0
h1x16-  2      0      15     1      1897.0 1000.0 1950.0 500.0  500.0
h1x16+ 2       1      0      1      1844.0 1000.0 1950.0 500.0  500.0
h1x01+ 2       0      0      1      1710.0 1000.0 1950.0 500.0  500.0
h1x05+ 2       0      4      1      1780.0 1000.0 1950.0 500.0  500.0
h1x12+ 2       0      11     1      1785.0 1000.0 1950.0 500.0  500.0
h1x08-  2      1      7      1      1755.0 1000.0 1950.0 500.0  500.0
h1x09-  2      1      8      1      1758.0 1000.0 1950.0 500.0  500.0
h1x04-  2      1      3      1      1735.0 1000.0 1950.0 500.0  500.0
h1x14+ 2       0      13     1      1815.0 1000.0 1950.0 500.0  500.0
h1x11-  2      1      10     1      1715.0 1000.0 1950.0 500.0  500.0
h1x05-  2      1      4      1      1750.0 1000.0 1950.0 500.0  500.0
h1x13-  2      1      12     1      1660.0 1000.0 1950.0 500.0  500.0
h1x04+ 2       0      3      1      1726.0 1000.0 1950.0 500.0  500.0
h1x15+ 2       0      14     1      1900.0 1000.0 1950.0 500.0  500.0
h1x12-  2      1      11     1      1715.0 1000.0 1950.0 500.0  500.0
h1x11+ 2       0      10     1      1715.0 1000.0 1950.0 500.0  500.0
h1x10+ 2       0      9      1      1820.0 1000.0 1950.0 500.0  500.0
h1x14-  2      1      13     1      1793.0 1000.0 1950.0 500.0  500.0
h1x02-  2      0      8      1      1695.0 1000.0 1950.0 500.0  500.0
h1x03-  2      1      2      1      1800.0 1000.0 1950.0 500.0  500.0
h1x10-  2      1      9      1      1795.0 1000.0 1950.0 500.0  500.0
h1x06-  2      1      5      1      1740.0 1000.0 1950.0 500.0  500.0
h1x03+ 2       0      2      1      1780.0 1000.0 1950.0 500.0  500.0
h1x08+ 2       0      7      1      1720.0 1000.0 1950.0 500.0  500.0
h1x09+ 2       2      14     1      1770.0 1000.0 1950.0 500.0  500.0
h1x01-  2      2      11     1      1775.0 1000.0 1950.0 500.0  500.0
h1x13+ 2       1      1      1      1690.0 1000.0 1950.0 500.0  500.0
h1x15-  2      1      14     1      1817.0 1000.0 1950.0 500.0  500.0
h1x02+ 2       0      1      1      1745.0 1000.0 1950.0 500.0  500.0
#
```

iii. Restore Procedure

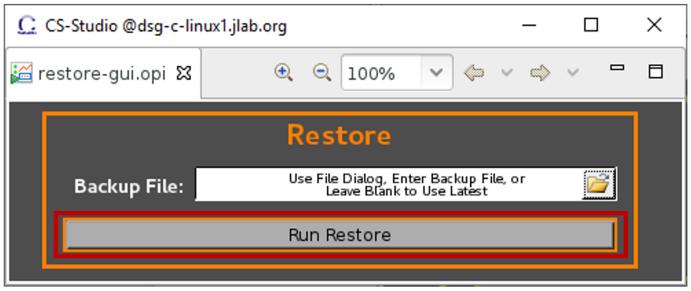
1. Open any detector's controls/monitoring screen.
2. Use "File" drop down menu to select "Restore".



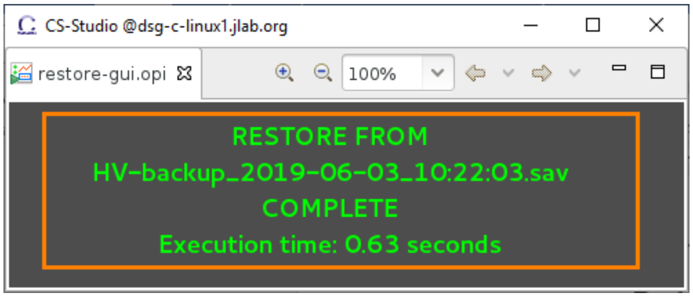
3. Select file to restore from. Can be done in three ways:
    a. Use file dialog by clicking icon of folder.
    b. Type name of backup file into text box.
    c. Leave text box blank (do neither option above) to restore from latest file.


CS-Studio @dsg-c-linux1.jlab.org
restore-gui.opi
100%
Restore
Backup File:
Use File Dialog, Enter Backup File, or
Leave Blank to Use Latest
Run Restore

4. Click "Run Restore" button.


CS-Studio @dsg-c-linux1.jlab.org
restore-gui.opi
100%
Restore
Backup File:
Use File Dialog, Enter Backup File, or
Leave Blank to Use Latest
Run Restore

Upon successful backup, the GUI prints the success message with the name of the file used in the restoration and the time required to complete the backup.



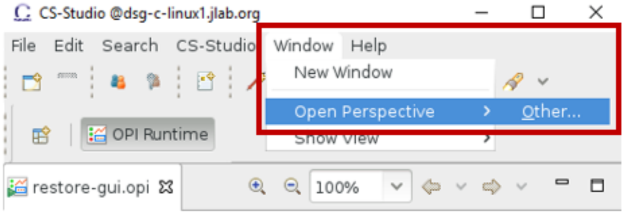
```
RESTORE FROM
HV-backup_2019-06-03_10:22:03.sav
COMPLETE
Execution time: 0.63 seconds
```

VI. Common issues and troubleshooting steps

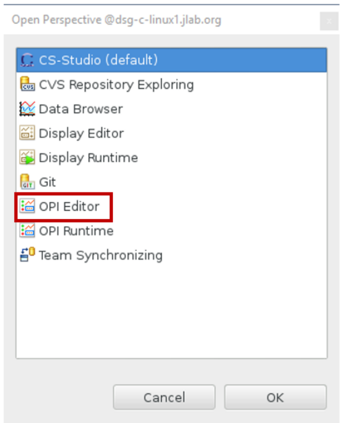
i. Start-up script does not open to main menu

If the last screen closed is not the main menu screen, the next time the start-up script is executed to open the monitoring and controls screens, the screen displayed will not be the main menu. Regardless of the screen displayed, to get back to the main menu, the user should follow the steps below.

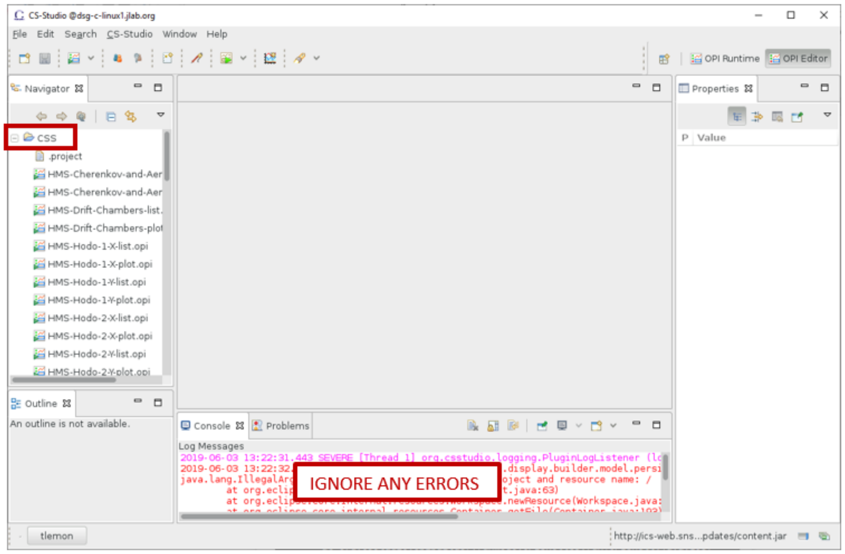
1. In top tool bar, go to Window – Open Perspective – Other…



2. Select "OPI Editor"

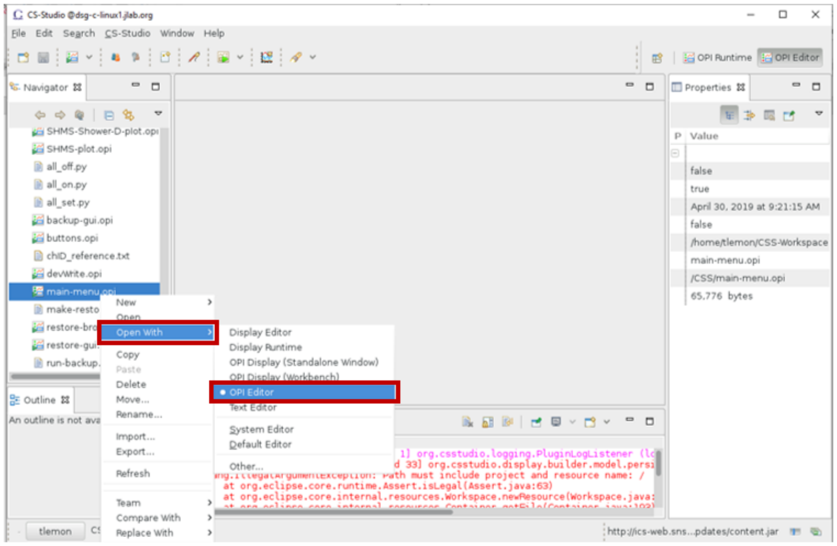


3. In window that opens, expand CSS folder in Navigator panel on left of window.
    a. Any errors in console pop-up tab at bottom of window can be ignored.



```
Log Messages
2019-06-03 13:22:31.443 SEVERE [Thread 1] org.csstudio.logging.PluginLogListener (lo
2019-06-03 13:22:32.                                  .display.builder.model.persi
java.lang.IllegalArg                                  oject and resource name: /
        at org.eclip                                  t.java:63)
        at org.eclip                                  .newResource(Workspace.java:
```

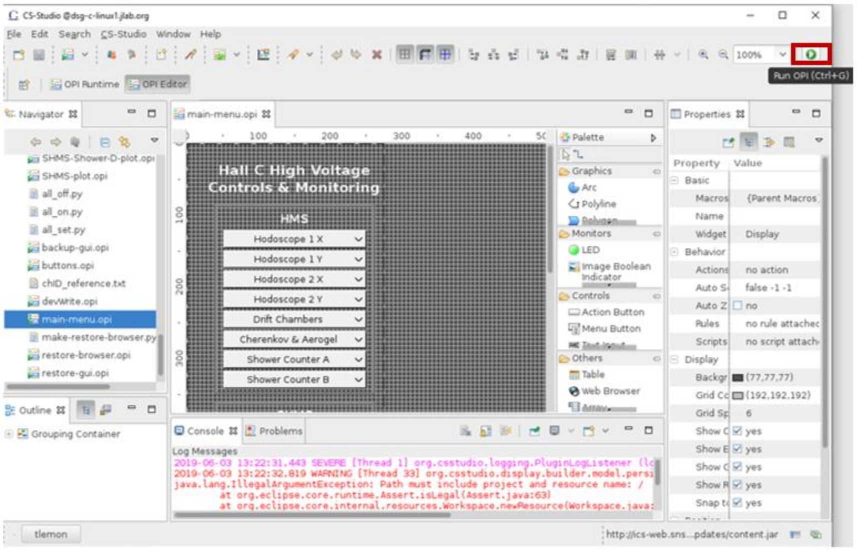
4. Right-click on “main-menu.opi”, select Open With – OPI Editor



P	Value
	false
	true
	April 30, 2019 at 9:21:15 AM
	false
	/home/tlemon/CSS-Workspace
	main-menu.opi
	/CSS/main-menu.opi
	65,776 bytes

```
1] org.csstudio.logging.PluginLogListener (lc
d 33] org.csstudio.display.builder.model.persi
ang.IllegalArgumentException: Path must include project and resource name: /
 at org.eclipse.core.runtime.Assert.isLegal(Assert.java:63)
 at org.eclipse.core.internal.resources.Workspace.newResource(Workspace.java:
```

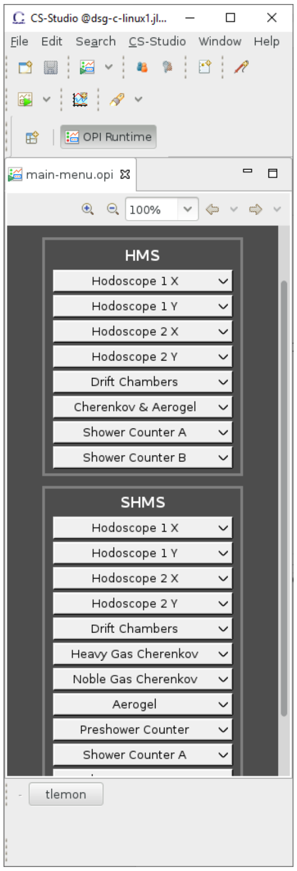
5. Click run button on top tool bar or hit Ctrl+G.



Property	Value
Basic	
Macros	{Parent Macros
Name	
Widget	Display
Behavior	
Actions	no action
Auto S	false -1 -1
Auto Z	□ no
Rules	no rule attachec
Scripts	no script attach
Display	
Backgr	(77,77,77)
Grid Cc	(192,192,192)
Grid Sp	6
Show C	☑ yes
Show E	☑ yes
Show G	☑ yes
Show R	☑ yes
Snap tc	☑ yes

```
Log Messages
2019-06-03 13:22:31.443 SEVERE [Thread 1] org.csstudio.logging.PluginLogListener (lc
2019-06-03 13:22:32.819 WARNING [Thread 33] org.csstudio.display.builder.model.persi
java.lang.IllegalArgumentException: Path must include project and resource name: /
        at org.eclipse.core.runtime.Assert.isLegal(Assert.java:63)
        at org.eclipse.core.internal.resources.Workspace.newResource(Workspace.java:
```

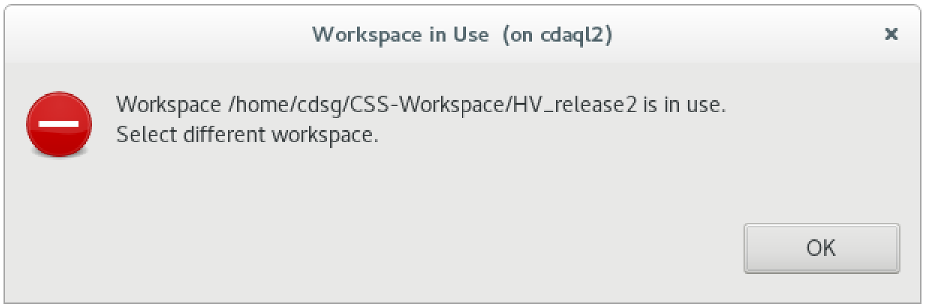
6. Close OPI Editor Perspective while leaving screen that opened after step 6 running.



CSS screens can now be used as normal after this step.

ii. Workspace cannot open because it is already in use

With the current start-up script for CSS, only one instance of the screens can be used at a time. If the screens are already open, the dialog box below will appear.

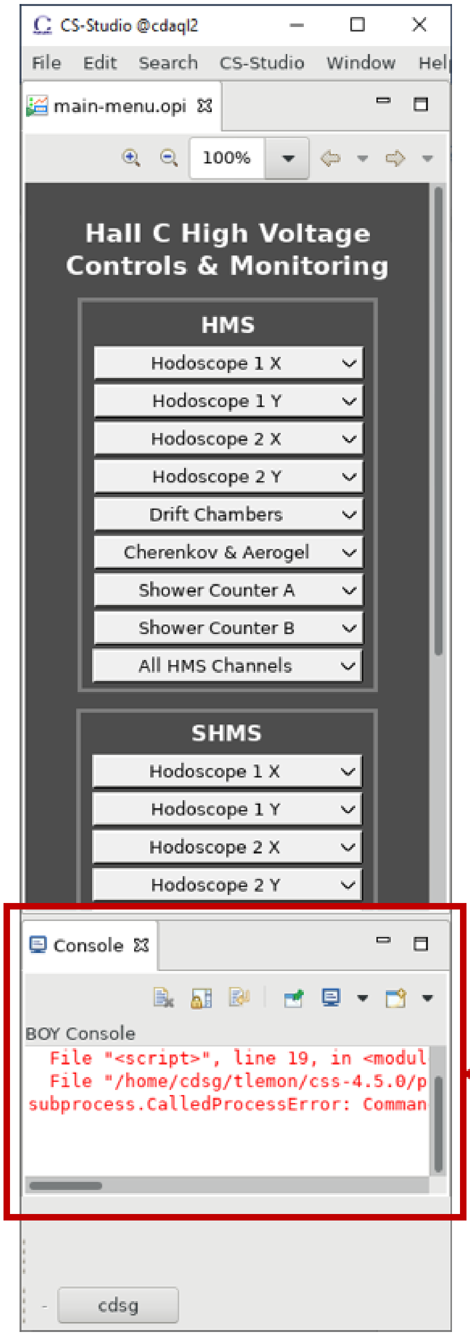

Workspace in Use (on cdaql2)

Workspace /home/cdsg/CSS-Workspace/HV_release2 is in use.
Select different workspace.

Currently there is no resolution for this error besides finding the instance of the screens already in use and close them. As mentioned in Part II, there is a version of the start-up script under development that will alleviate this issue.

iii. CSS errors

When an error occurs in CSS, it will display the error in a pop-up console tab at the bottom of the workspace's main screen. Typically, this main screen for Hall C CSS-BOY HV system will be the main menu.



```
  File "<script>", line 19, in <modul
  File "/home/cdsg/tlemon/css-4.5.0/p
subprocess.CalledProcessError: Comman
```

Console pop-up tab displayed due to an error.

The following are errors known to occur and how to resolve them.

a. Script returns error

When calling scripts from CSS, as done by for the backup/restore GUIs and the group controls, there is a possibility that calling the script will result in an error. The error displayed in the console tab will notify the user of which script the error occurred in and the approximate line in the script that caused the error. The user can then use this information to debug the script. Below is an example error received when executing the backup functionality of the backup/restore script from the backup GUI.

```
1  2019-06-03 10:24:05 ERROR: Error in  on Action Button_1.
2  Traceback (most recent call last):
3  File "<script>", line 19, in <module>
4  File "/home/cdsg/tlemon/css-4.5.0/plugins/org.python.jython_2.7.0.release.jar/Lib/subprocess.py", line 551, in check_output
5  subprocess.CalledProcessError: Command '['python', '/home/tlemon/CSS-Workspaces/dev/CSS/HV-bur.py',
   'backup', '/home/tlemon/CSS-Workspaces/dev/CSS/']' returned non-zero exit status 2
```

The message displayed when the error is the result of a failed script tends to have two parts. The first (Line 1 in example above) states the date and time of the error and the widget that was activated to trigger the script. The second part (lines 2-5) gives the source of the error in the called script.

b. Resource Exception: Resource called is out of sync in CSS

If a file called by CSS, such as a script embedded in a widget or an image, is modified in any way and then opened in CSS quickly after modification, CSS will display an error stating that the source file is out of sync with the workspace (e.g., developing a script in your preferred IDE, while running the CSS screen that calls it and then using the widget that calls the script). The solution to this error is to simply wait a few seconds before performing the action that triggered the error. CSS automatically refreshes files in its workspace, but if the file was modified using an external program, it takes a few seconds. Below is an example error received when the source file is out of sync with the CSS workspace.

```
1  2019-06-03 12:57:47 ERROR: Failed to register tcl2css_V2.py.
2  org.eclipse.core.internal.resources.ResourceException: Resource is out of sync with the file system:
   '/CSS/tcl2css_V2.py'.
```

Line 1 states the time and date of the error with the source. Line 2 states the actual error and the file out of sync.

iv. Installing required software/packages.

a. *pyepics*

To install *pyepics*, the following terminal command should be used:

```
pip3 install --trusted-host pypi.org --trusted-host files.pythonhosted.org --user pyepics
```

Components of the command are as follows:

```
pip3 install
```

- Calls *pip3* (*P*ackage *I*nstaller for *P*ython 3) with the argument to install a module.

```
--trusted-host pypi.org --trusted-host files.pythonhosted.org
```

- Adds pypi.org (a common repository for Python modules) and pythonhosted.org (site that hosts documentation from modules' authors) as a trusted host.
- This option may not be necessary, but with the proxies set up on Hall experimental subnets for network security, declaring these two sites as trusted hosts prevents *pip* from not being able to connect to download the necessary files to install *pyepics*.

```
--user
```

- Option installs module for only the user currently logged in.
- This option can be omitted if sudo or root privileges are used to install the module.

```
pyepics
```

- Declares *pyepics* as the module for *pip* to install.

b. *pip*

If user needs *pip* installed to be able to then install *pyepics*, the command below should be used.

```
sudo yum install python3-pip
```

This command calls *yum*, the primary tool for maintaining/installing Red Hat Enterprise Linux (RHEL) software packages from official RHEL repositories, as a sudo user to install *pip* for Python 3.