Settings Manager

Release 3.0

Tong Zhang

INTRODUCTION

1	Introduction	3			
	1.1 Installation	3			
	1.2 Start up Application				
	1.3 GUI Overview				
2	Getting Started 1				
	2.1 App Launcher	11			
	2.2 Restore Settings	12			
	2.3 Save Settings	14			
3 Manage Accelerator Settings					
	3.1 Controls Environment	15			
	3.2 Initialize Device Settings View	16			
	3.3 Capture Device Settings				
	3.4 Cast Device Settings				
	3.5 Apply Device Settings				
4	Manage Snapshot Files	27			
	4.1 Load Snapshot Files	27			
	4.2 Working Directory				
5	Indices and tables	29			

Settings Manager is a PyQt5 application developed for the accelerator physics settings management. It is one of the high-level applications that built on top of PHANTASY framework.

INTRODUCTION 1

2 INTRODUCTION

CHAPTER

ONE

INTRODUCTION

Settings Manager features convenient loading and saving device settings, as well as value scaling to work with multi-charge state beam. Seamless integrated with other UI components, the user can initialize the device list by loading accelerator settings segment by segment. Columns of values show the information not limited to the current device settings. This application also features easy device accessibility by comprehensive filtering functionality, control knobs through context menu actions, etc.

1.1 Installation

Settings Manager is one of the GUI applications that built upon PHANTASY framework. The most straightforward installation approach is via apt install for Debian Linux OS. The way via pip install is under development.

Currently, Debian Buster is the supported Debian release, type the following commands in the Terminal to have it installed.

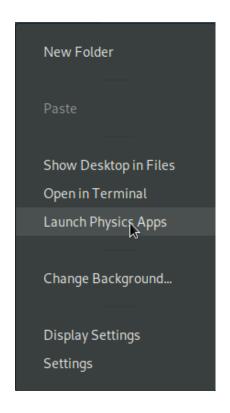
```
sudo apt update sudo apt install python3-phantasy python3-phantasy-ui python3-phantasy-apps
```

An ideal test environment is VirtualBox with the appliance (.ova file), which is a pre-built Debian Buster with FRIB controls repository configured, the ova file is hosted on lab shared drive: I:\simulations\physapps\physvms\develop-vmphy0-v7.ova, along with develop-vmphy0-v7.pdf as the guide manual.

1.2 Start up Application

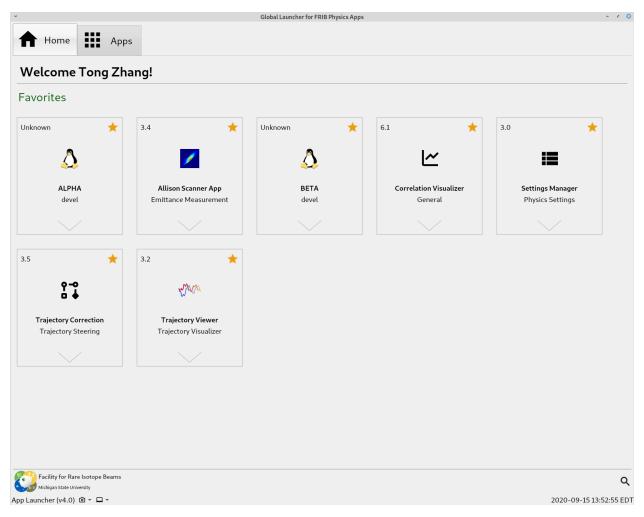
There are the two different ways to start up Settings Manager, through the global app launcher system or physics applications or console with command settings_manager.

App Launcher is another PyQt application developed to manage applications, from where the user can easily launcher any app by clicking. Right-clicking on the GNOME desktop will pop up the menu as the following image shows, click Launch Physics Apps will start up App Launcher.

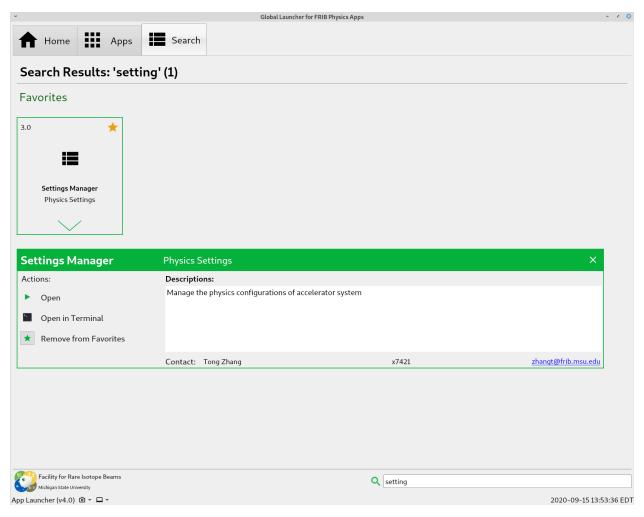


Note: For development (Accelerator Physics Department staff only), in the terminal, execute: $/files/shared/ap/run_apps.sh$

Below shows the main windows of App Launcher, as one can see, by default the welcome page will list all the *Favorites* which is a collection of all favored apps.

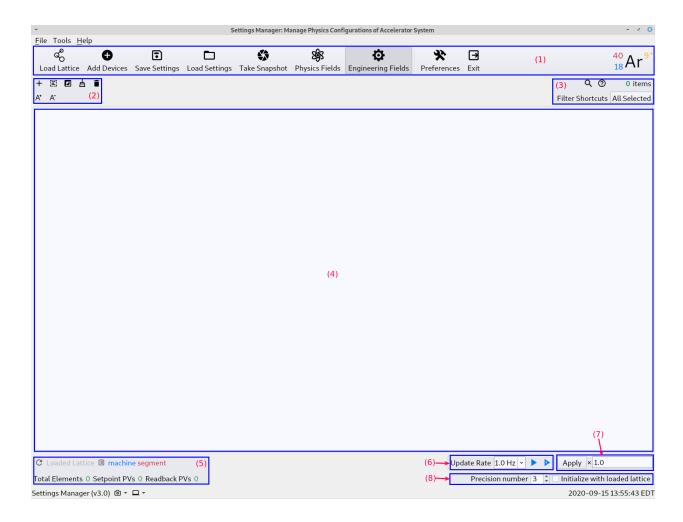


Input string in the right bottom *Search* box to filtering the app, or by hitting + to activate searching function, and pressing to deactivate.



On the card of Settings Manager, the top-left corner shows the version, top-right corner indicates it's a favorite app, click the star again will unfavor it, click the bottom-middle arrow button will open the details of the app, which provides various actions, and descriptions, as well as contact information.

Single click the app icon (in the middle of card) will launch the app, or choose other modes from the *Actions* list. The following image show the main windows of Settings Manager.



1.3 GUI Overview

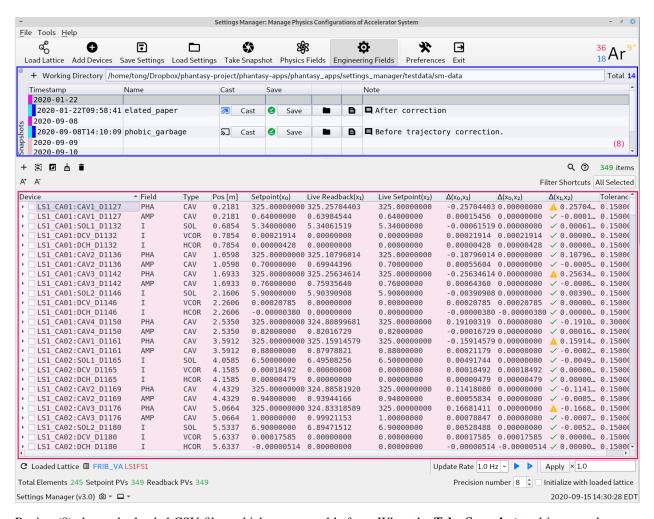
In the above figure, one can read numbered components as following:

- (1): Toolbar presents the buttons for major features
- (2): Convenient buttons for settings view (see (4))
- (3): Device filtering function
- (4): Treeview of device settings
- (5): Statistics
- (6): Value updating function
- (7): Set device settings with scaling factor
- (8): Complementary configurations for settings view

 $Press\ or\ through\ {\hbox{\tt Help}}\ {\hbox{\tt ->}}\ {\hbox{\tt Contents}}\ will\ open\ the\ help\ window\ for\ references}.$

After loaded data, below shows the typical how the main window looks like (Settings Manager version 3.x):

1.3. GUI Overview 7



Region (8) shows the loaded CSV files, which were saved before. When the **Take Snapshot** tool is pressed, a new entry will be added into this area. The user can use button *Cast* to reload different snapshot, and *Save* button to save the snapshot as a file, there is also button available to view the saved CSV file with system application, e.g. LibreOffice Calc.

The red box area shows the device settings, with columns:

- Device: Device name
- · Field: Dynamic field name defined in PHANTASY framework, which is used to control the device
- *Type*: Device type
- Pos: Global position along the accelerator beam line, in meter
- Setpoint, alias x0: Stored field setting, will change after loading settings from a file, or taking snapshot
- Live Readback, alias x1: Live field readback value
- *Live Setpoint*, alias x2: Live field setpoint value
- Δ (x0,x1): x0 x1, the discrepancy between stored setpoint and live readback
- Δ (x0,x2): x0 x2, the discrepancy between stored setpoint and live setpoint
- Δ (x1,x2): x1 x2, the discrepancy between live readback and setpoint
- *Tolerance*: Absolute discrepacy allowed for x1 and x2 to present green check status (see the left column), otherwise, show warning symbol

• Writeable: if the device is allowed to change, depends on Channel Access permission

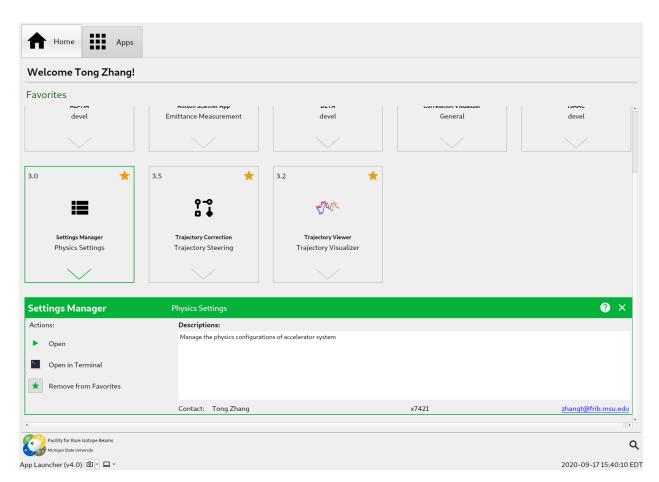
Click the column name tab will trigger the sorting feature, either sorting by alphabetic, numerical depends on the column data type, the *Device* column is sorting by the four digits after 'D', which stands for the longitudinal position along the accelerator.

1.3. GUI Overview 9

GETTING STARTED

This page presents the quick reference for the user to work with existing saved settings files with *Settings Manager*, to save and restore the machine states.

2.1 App Launcher

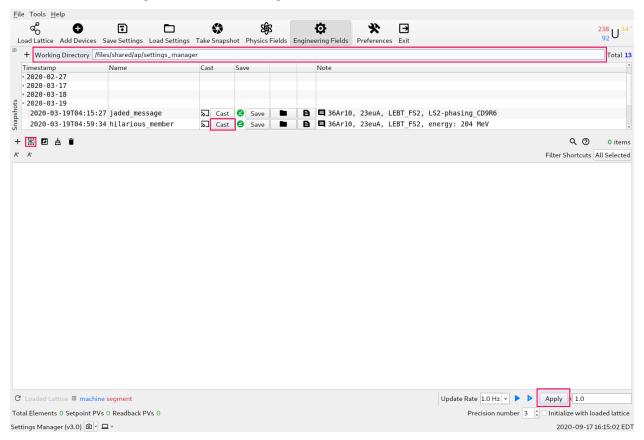


Run /files/shared/ap/run_apps.sh in the Terminal will reach App Launcher, from where Settings Manager can be seen in the Favorites tab, the user can:

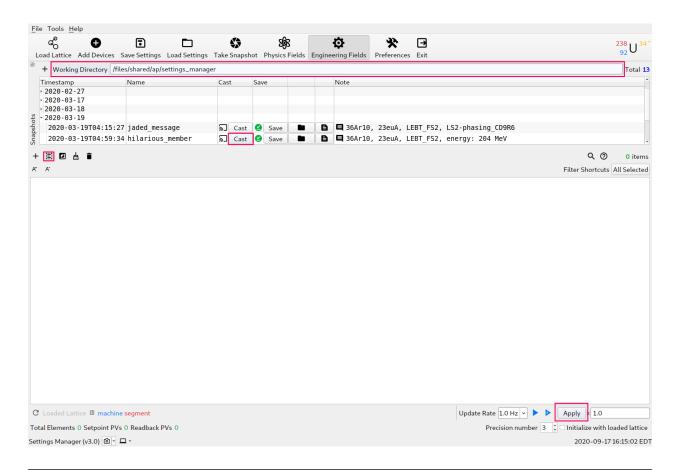
- Start up the app: click the icon, or through Actions
- Read the PDF documentation: click? mark icon in the app details window

2.2 Restore Settings

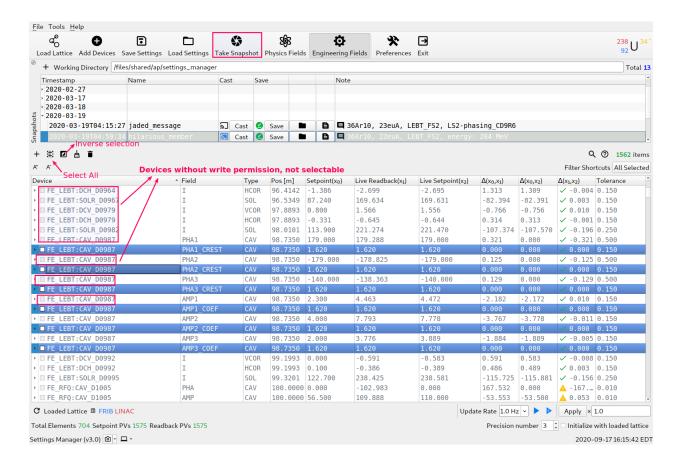
After starting up the app, the user should see all the loaded snapshots from the *Working directory*, by clicking any of them will show the settings data in the main settings tree view area.



And check all the devices, then push Apply button to do the restore. The text box right besides Apply button could be used to apply with the values of Setpoint(x0) multiplied by the input scaling factor.



Note: The default working directory is /files/shared/ap/settings_manager, where keeps all the saved settings files from *Settings Manager**. The user can find them in *Snapshots* windows. The *cast* button of each row can be used to load device settings. For now, please use *Note* column to append additional information for the dataset.



Note: The user can only select and check rows that can be controlled, i.e. with the write permission to the device. Multi-selection is supported with Ctrl, Shift keys, right-clicking on the selected rows will pop up the menu for check and uncheck rows.

2.3 Save Settings

Anytime the user pushes *Take Snapshot* tool will create a new entry in *Snapshots* window, from where the user can make note, save as a file, etc.

Note: After taking snapshot, if the user does not **save** with the *Save* button of the new snapshot entry, the snapshot data only persistent in the memory when the app is running, so always save *Valuable* data as you want.

Note: The user also can use *Load Settings* and *Save Settings* tools to do the same work, but this may not be managed by *Snapshots* window.

CHAPTER

THREE

MANAGE ACCELERATOR SETTINGS

This page shows how to initialize the device settings from scratch, and perform save and load settings, as well as set devices with settings. To do the following instructions, a running virtual accelerator is required.

Note: All the following operations are applied to EPICS driven accelerator.

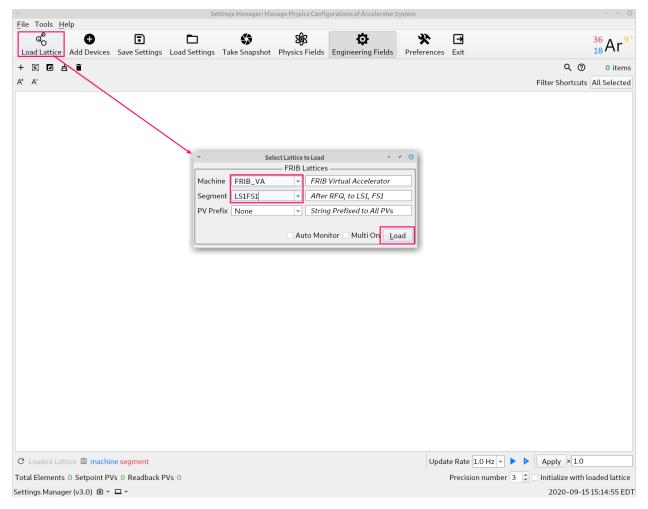
3.1 Controls Environment

Either virtual accelerator or real accelerator is driven by EPICS controls, to start a virtual accelerator to demonstrate the use of *Settings Manager*.

Select *Virtual Accelerator Launcher* from App Launcher, and choose machine *FRIB_VA* and segment *LS1FS1*, or executable the following line in Terminal:

phytool flame-vastart --mach FRIB_VA --subm LS1FS1

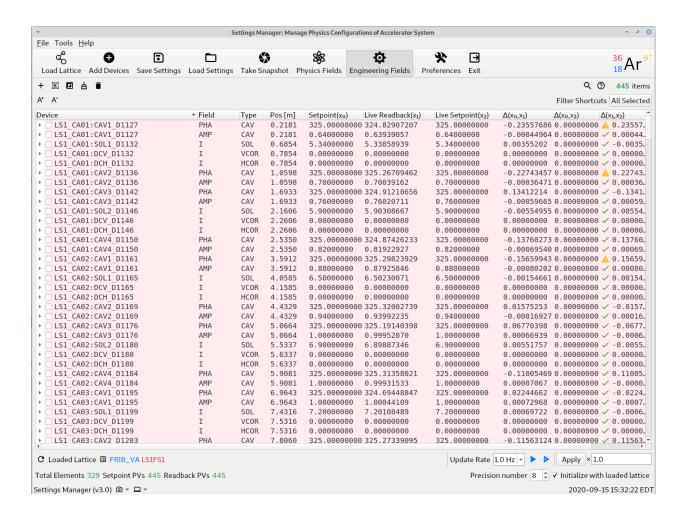
3.2 Initialize Device Settings View



Click *Load* button to load lattice *FRIB_VA/LS1FS1*. A message box will pop up which says the lattice is loaded, but no device settings to show. click *OK* to close the message box, and check the bottom right *Initialize with loaded lattice* option, after that, all the device settings (exclude diagnostic devices) should present in view area.

Note: As one may notice, the listed device values are only of 3 float precision, which is not enough for virtual accelerator, since all the values stand for physics units. Change the presicion number spinbox to the value of 8 should be good enough.

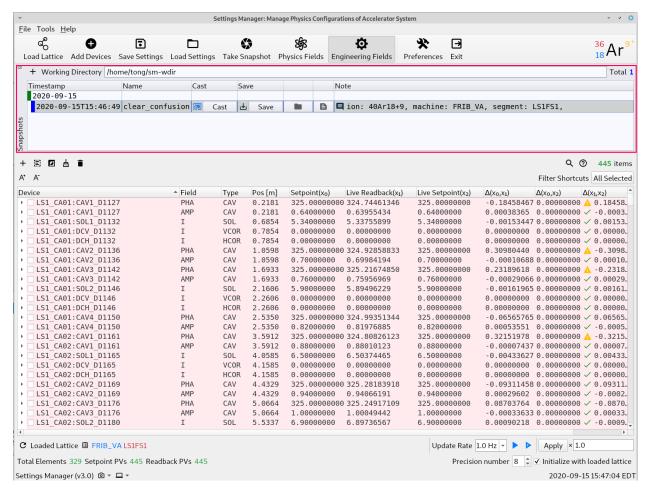
Below is the total 445 rows of settings in the loaded lattice.



3.3 Capture Device Settings

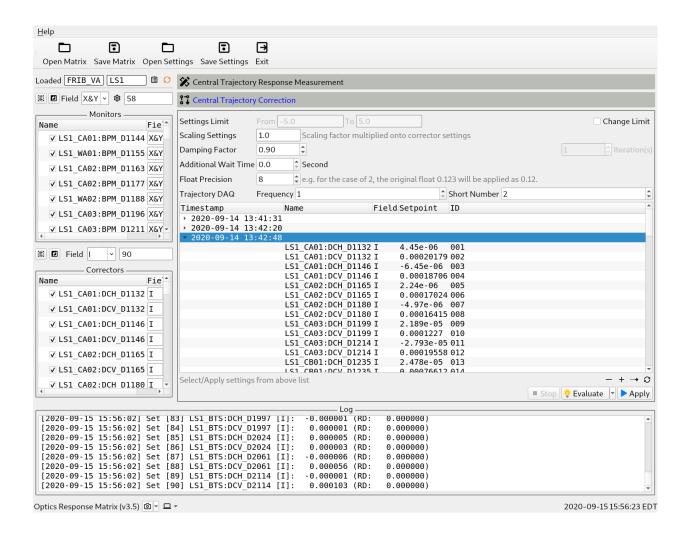
Beginning from now, the user can take snapshot of the accelerator by clicking *Take Snapshot*, the program will first update the settings view, then make a copy of all the data into memory, and show in the *Snapshots* dock window (just right below the Toolbar, it is relocatable).

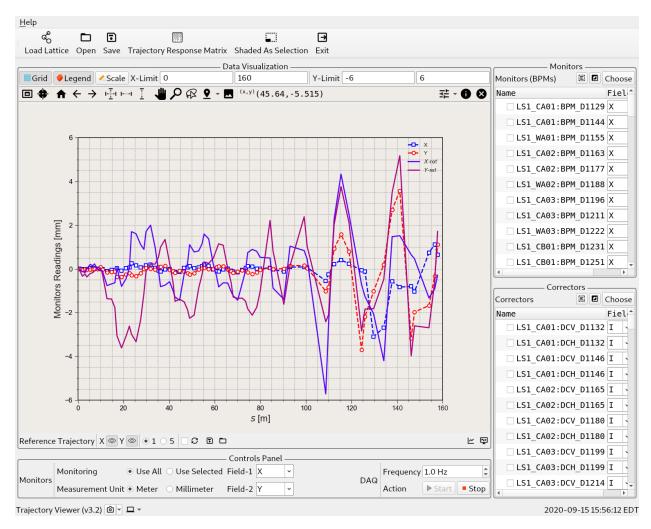
Let's say before doing the trajectory correction, the user wanna save all the device settings. After clicking *Take Snapshot*, a new entry appears in the *Snapshots*, see the following figure.



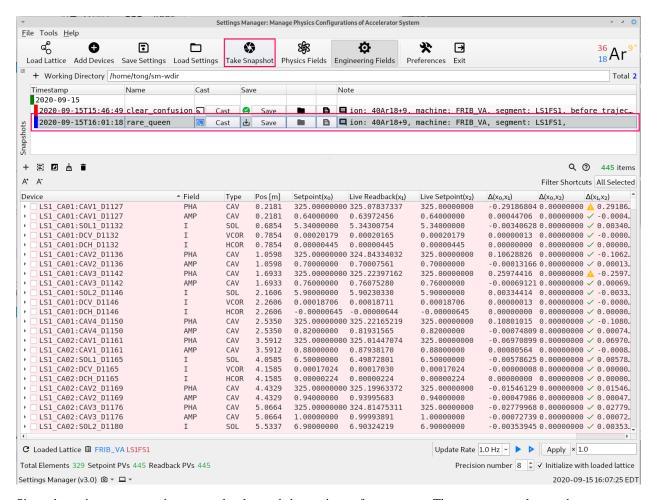
Now the snapshot is **ONLY** existing in the memory, the user has the option to save it to a file (do what as *Save* button guides), the *Note* area is filled with default message, the user may want input more useful information, here input "Before trajectory correction" as a memo, the *Name* area is also editable. The *Cast* column is lit with blue if current snapshot is showing in the main settings treeview, otherwise, it will be grayed out. Here we save this snapshot to a file.

Do trajectory correction with the apps *Trajectory Viewer* and *Trajecotry Correction*, could be found in the App Launcher, here skip the details, just post the figures below.





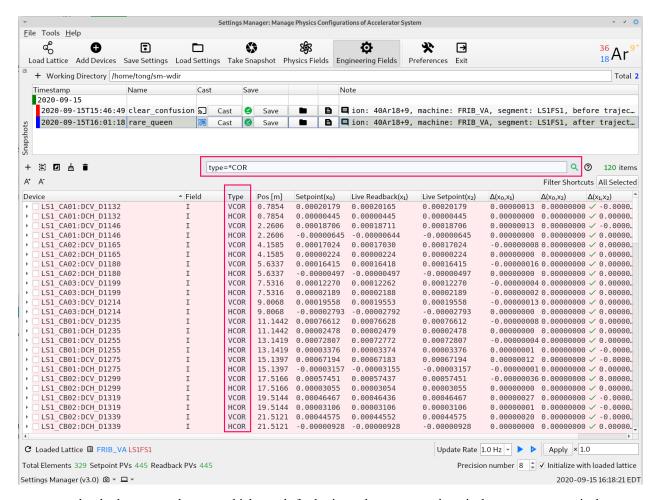
Now it is ready to take another snapshot of current device settings after trajectory correction. Following the same operation, click *Take Snapshot*, a new entry will append to the *Snapshots* window, shown name as *rare_queen* (random words trick), we may add note as "After trajectory correction", and save.



Since the trajectory correction app only changed the settings of correctors. The user may only see the correctors. *Settings Manager* was designed with comprehensive search function. Here is the simple guide.

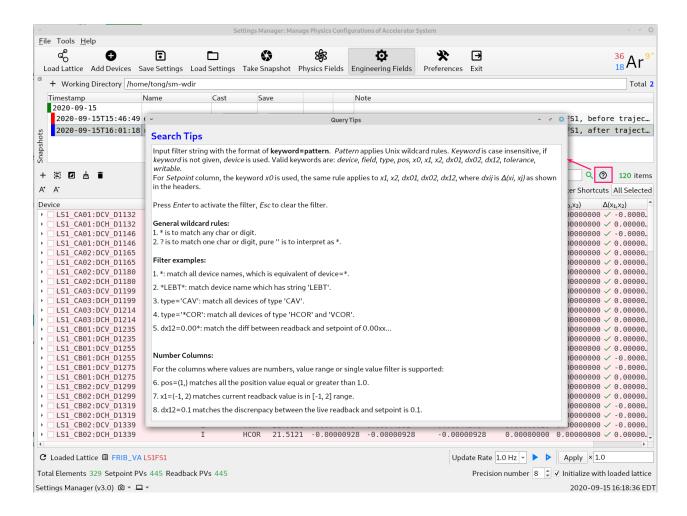
Click Search button to enable search function, or use short cut + to activate, and pressing to deactivate.

Type type=*COR in the search input box, press to start filtering.



*COR matches both HCOR and *COR, which stands for horizontal corrector and vertical corrector, respectively.

Click the question mark button right of search button will show the tips for searching.



3.4 Cast Device Settings

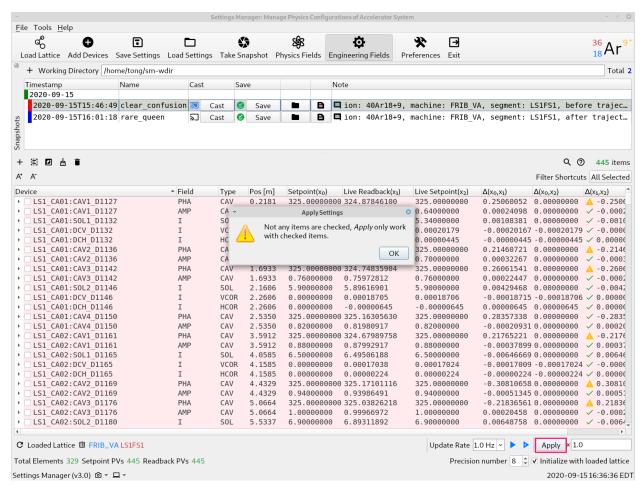
Now the user want load the device settings before trajectory correction, and restore the machine status.

- 1. Click the *Cast* button of snapshot of the name *clear_confusion*, which was noted "before trajectory correction", keep eye on the *casted* icon, should mark as blue.
- 2. Select and check the devices for settings apply, see next section.

3.5 Apply Device Settings

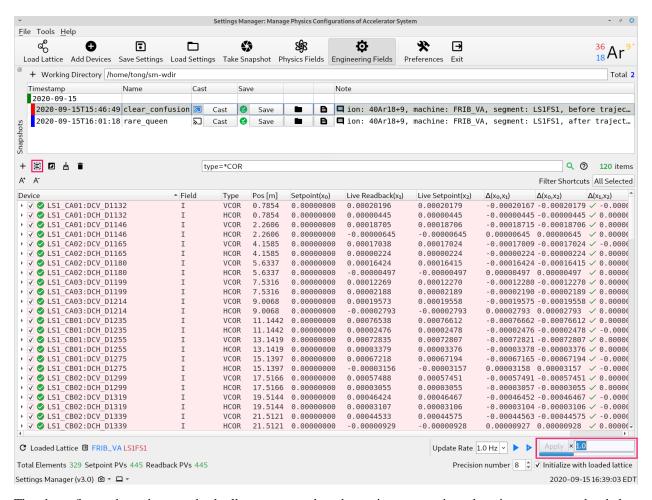
Note: Set devices with the settings shown in *Setpoint* (x0) column only work with the rows that the *Device* item is checked.

Now after loading a different snapshot by casting, the user directly go to press *Apply* button, the following message box may pop up.



Instead, check all correctors, and Apply again, it will set devices with new settings.

24



The above figure shows how to check all correctors, when the setting process is undergoing, a progress bar below *Apply* button shows the progress, meanwhile the applied device setting will be marked with green checked symbol.

If go to check the *Trajectory Viewer* app, the trajectory is back to the original.

Note: For trajectory correction, the *Trajectory Correction* app can do all the similar task as shown here, i.e. arbitrary switch between different set of corrector settings. While *Settings Manager* provides the general way to manage the machine states, which is much more robust and powerful.

Additional feature of Apply, before pressing this button, the user may multiple a additional scaling factor to the Setpoint (X0) column, the scaling factor is a single float number, or a valid mathematic expression, e.g. 12/18, 0.5, etc., for invalid input, 1.0 is used by default.

CHAPTER

FOUR

MANAGE SNAPSHOT FILES

For those already have saved settings files from *Settings Manager*, the device settings could be loaded in a pretty simple way, there is no need to load lattice and do the initialization any more starting version 3.0.

A comprehensive module for manage the snapshots have been developed to facilitate these features.

4.1 Load Snapshot Files

Settings Manager supports loading snapshot files by drag and drop. To do this, simply drag files and drop into the main window, it will present all the data in Snapshots window, then the user can load any of them by Cast button and Apply the checked items to change machine state.

4.2 Working Directory

By default, *Settings Manager* will load all of the saved files from *Working Directory*, and present in the *Snapshots* window, however, the user has the option to change the working directory in the *Preferences* dialog.

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search