Pianofish User Guide

Guy Streeter

December 11, 2017

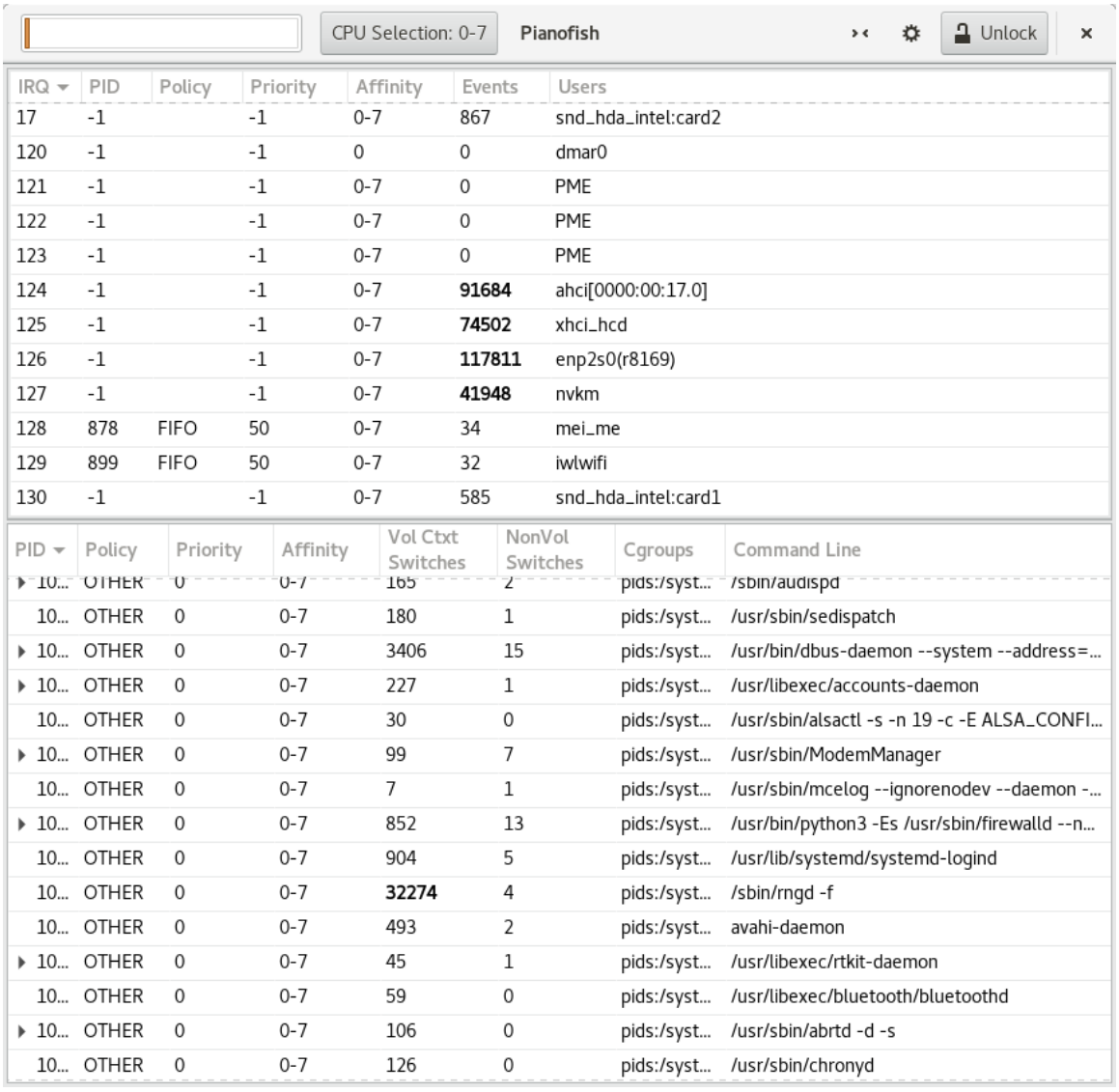
1 Introduction

Pianofish is a re-implementation of the *Tuna* application's Graphical User Interface. The main differences are a focus on representing the physical system more clearly and providing additional functions for distributing running applications to the processors. Users of the *Tuna* GUI will see some visual similarities with the main Pianofish window.

Note: This document describes Pianofish as it is implemented for Fedora®, and has been tested on Fedora 26 and 27.. Pianofish is also available for RHEL and CentOS 7[1]. Because of some differences the supporting packages, the RHEL or CentOS version behaves a little differently. This will be noted where it is applicable.

Figure 1 on the following page shows an example of the *Pianofish Main Window*. The two large panes are *IRQ* and *Process* displays, familiar to users of the *Tuna* GUI. The *Header Bar* is described in detail in Section 2 on page 3. The *IRQ* and *Process* panes described in section 3 on page 4 and section 4 on page 4, respectively.

[1]Fedora, Red Hat Enterprise Linux, and CentOS are trademarks of Red Hat, Inc.



IRQ	PID	Policy	Priority	Affinity	Events	Users
17	-1		-1	0-7	867	snd_hda_intel:card2
120	-1		-1	0	0	dmar0
121	-1		-1	0-7	0	PME
122	-1		-1	0-7	0	PME
123	-1		-1	0-7	0	PME
124	-1		-1	0-7	**91684**	ahci[0000:00:17.0]
125	-1		-1	0-7	**74502**	xhci_hcd
126	-1		-1	0-7	**117811**	enp2s0(r8169)
127	-1		-1	0-7	**41948**	nvkm
128	878	FIFO	50	0-7	34	mei_me
129	899	FIFO	50	0-7	32	iwlwifi
130	-1		-1	0-7	585	snd_hda_intel:card1

PID	Policy	Priority	Affinity	Vol Ctxt Switches	NonVol Switches	Cgroups	Command Line
10...	OTHER	0	0-7	165	2	pids:/syst...	/sbin/audispd
10...	OTHER	0	0-7	180	1	pids:/syst...	/usr/sbin/sedispatch
10...	OTHER	0	0-7	3406	15	pids:/syst...	/usr/bin/dbus-daemon --system --address=...
10...	OTHER	0	0-7	227	1	pids:/syst...	/usr/libexec/accounts-daemon
10...	OTHER	0	0-7	30	0	pids:/syst...	/usr/sbin/alsactl -s -n 19 -c -E ALSA_CONFI...
10...	OTHER	0	0-7	99	7	pids:/syst...	/usr/sbin/ModemManager
10...	OTHER	0	0-7	7	1	pids:/syst...	/usr/sbin/mcelog --ignorenodev --daemon -...
10...	OTHER	0	0-7	852	13	pids:/syst...	/usr/bin/python3 -Es /usr/sbin/firewalld --n...
10...	OTHER	0	0-7	904	5	pids:/syst...	/usr/lib/systemd/systemd-logind
10...	OTHER	0	0-7	**32274**	4	pids:/syst...	/sbin/rngd -f
10...	OTHER	0	0-7	493	2	pids:/syst...	avahi-daemon
10...	OTHER	0	0-7	45	1	pids:/syst...	/usr/libexec/rtkit-daemon
10...	OTHER	0	0-7	59	0	pids:/syst...	/usr/libexec/bluetooth/bluetoothd
10...	OTHER	0	0-7	106	0	pids:/syst...	/usr/sbin/abrtd -d -s
10...	OTHER	0	0-7	126	0	pids:/syst...	/usr/sbin/chronyd

Figure 1: Example Pianofish Main window

2 The Header Bar

The first item in the *Header Bar* is an aggregate CPU load indicator. Usage of all the the CPUs is averaged for this display.

The second item in the *Header Bar* indicates which CPUs are currently selected. The current *CPU selection* is used in actions of the *IRQ* and *Process* panes, described in their respective sections. Clicking the *CPU Selection* button brings up the *System View* window, containing a display of the system's hardware topology and an interface for changing the *CPU Selection*. The *System View* window is described in section 5 on page 7.

The next button in the *Header Bar* is the *Filter* button. This button (or the CTRL-F shortcut) opens a *Filter* entry at the bottom of the *Process* pane. *Pianofish* will filter the list in the *Proc*ess pane to match the expression the user enters in the filter. The filter text can be a regular expression or a Bash-style glob. The type of the filter expression can be changed in Preferences (Section 6 on page 10).

Next is the *Preferences* button. The user can also access *Preferences* from the *Application Menu* presented by the desktop manager, if supported. Settings are persistent across invocations of *Pianofish*, and are unique to each user.

The last button is the *Unlock* button. This initiates an authorization process between the *Pianofish* GUI and its privileged back-end service. The back-end will present a prompt for administrative credentials. When the authorized connection is established, the text of the button will change to `Lock`. If the user clicks the `Lock` button, it will close the session with the back-end service and change the text back to `Unlock`. The back-end service may also time out, in which case the button text will indicate it needs to be unlocked again. If the user starts an action that requires access to the back-end, Pianofish will automatically start the Unlock activity, if necessary.

Note: On RHEL or CentOS, there is no *Unlock* button. Instead, when the user starts *Pianofish*, the system immediately prompts for administrative credentials, and the Pianofish GUI runs as a privileged process. The user should be aware of security considerations involved when running a GUI application as a privileged process.

3 The IRQ Pane

The *IRQ* pane displays information about Interrupt Request (IRQ) handling. By default the information shown in the *IRQ* pane is refreshed on a configurable interval. If any of the values displayed for an IRQ has changed in the last refresh, *Pianofish* will present the new value in bold type.

Some interrupts are handled directly by the kernel. Those are indicated by a Process ID (PID) value of –1. Interrupts handled directly by the kernel also do not have a Scheduler Policy (indicated as an empty field) or a Priority (indicated by –1). IRQs handled by separate kernel threads have a PID, Scheduling Policy, and (if the Policy is one of the Real-time options) a Real-time Priority. All IRQs have a CPU affinity.

Note: The CPU Affinity of some IRQs may be unchangeable. Never change the CPU Affinity, Scheduler Policy, or Real-time Priority of any IRQ without a complete understanding of the consequences.

The user may change the attributes of an IRQ in two ways. To quickly set the CPU Affinity of an IRQ, click and drag the IRQ entry to the *CPU Selection* button in the *Header Bar* and drop it there. Pianofish will set the IRQ's CPU Affinity to the current *CPU Selection* value.

The user may also right-click on an IRQ entry. *Pianofish* will pop up a menu with two options. The second menu entry toggles refresh of the information in the *IRQ* pane. The first menu entry brings up the *Set IRQ Attributes* dialog (Figure 2). Here the user can change all three attributes (if they are available) of the selected IRQ.

After the user clicks **Done**, *Pianofish* will display a pop-up message to indicate the completion of the change, and the user can dismiss the pop-up by clicking anywhere in the *Main Window*.

4 The Process Pane

In the *Process* pane, *Pianofish* displays information about individual processes. It shows the Process ID, Scheduler Policy, Real-time Priority, and CPU affinity, as is shown in the *IRQ* pane for interrupt

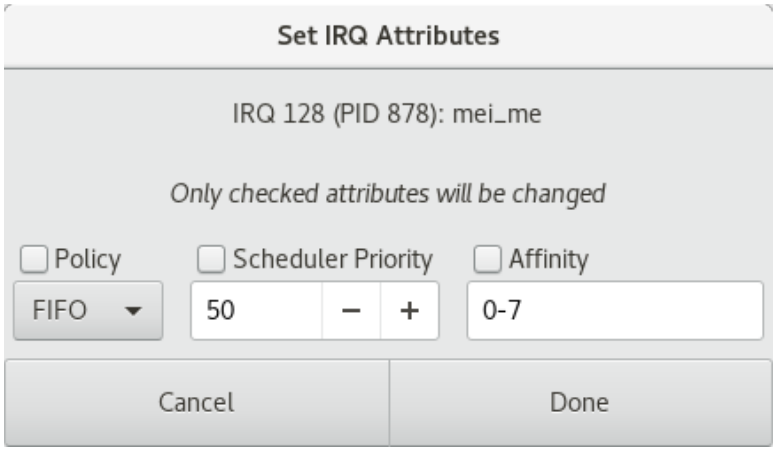


Figure 2: Example Set IRQ Attributes Dialog

handlers. Additionally, the Process Pane shows the count of Voluntary and Non-Voluntary Context Switches, the Cgroups, and the Command Line for each process. The user can alter this display in *Preferences*, hiding the kernel threads or the user-space processes, or the Cgroups column.

The *Process* pane shows values in bold if they have changed since the last refresh. It initially shows multi-threaded processes as a collapsed list. If the user selects a collapsed process thread list, Pianofish considers all the threads of that process to be selected. When a process thread list is expanded, individual threads can be selected. Multiple processes or threads and be selected using Shift-click or Control-click.

As with IRQs in the *IRQ* pane, the user can drag and drop selected processes from the *Process* pane to the *CPU Selection* button in the *Header Bar* to quickly set the CPU affinity of the selected processes or threads. *Pianofish* will confirm that the change has been completed by popping up a message box, and the user can dismiss the message by clicking anywhere in the *Main Window*.

If the user right-clicks on the selection in the *Process* pane, *Pianofish* will display a dialog offering three actions:

- Set process attributes
- Distribute tasks
- Toggle refreshing of the *Process* pane

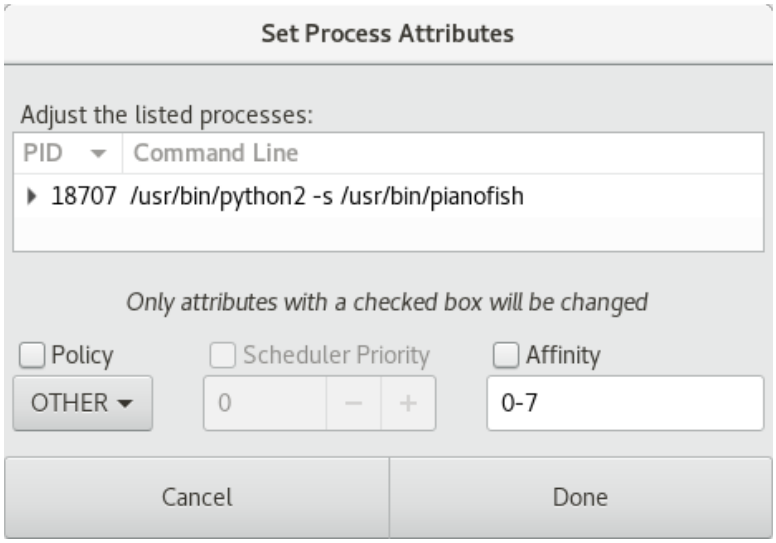


Figure 3: Example Set Process Attributes Dialog

4.1 Setting Process Attributes

Setting the process attributes is an action similar to the one offered for IRQs in the *IRQ* pane. The user can choose to change the Scheduling Policy, Real-time Priority, and CPU affinity of the selected processes or threads. Figure 3 is an example of the *Set Process Attributes* dialog. When the user clicks **Done**, *Pianofish* will apply the changes and notify the user of completion with a pop-up message.

Note: Changing the Scheduling Policy or Real-time Priority of processes can cause system instability. Making any changes in the attributes of kernel threads can disrupt the system. Be certain that you understand the consequences before making any changes

Distributing Tasks

Pianofish can distribute the CPU affinity of the selected processes or threads across CPUs. This distribution will attempt to keep the CPUs of a thread's affinity "close together", for instance making sure a thread's affinity setting will keep it on one core, socket, or

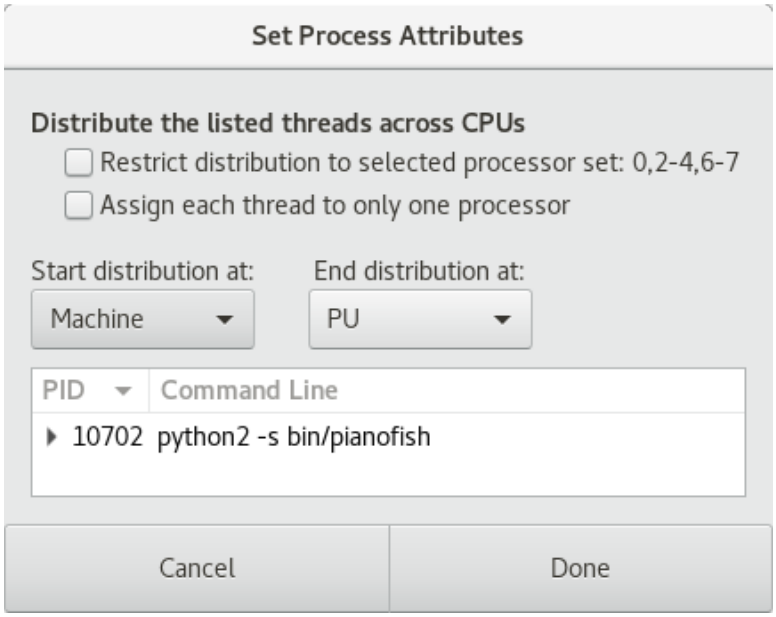


Figure 4: Example Distribute Tasks Dialog

NUMA node. Figure 4 is an example of the dialog box *Pianofish* displays when the user chooses to distribute tasks.

If the current CPU selection is not the entire set of CPUs, *Pianofish* will offer to restrict the distribution to the current selection. The user can also chose to assign only one CPU to each thread. Additionally, the user can choose the starting and ending group levels *Pianofish* will consider when calculating the distribution.

5 Choosing the CPU selection

Pianofish provides a view of the system that can help the user choose an appropriate CPU selection. Clicking the *CPU Selection* button in the *Header Bar* opens the *System View* window. An example of this window is shown in figure 5 on the following page. There are two panes in the *System View* window. The *Topology* pane show a hierarchical view of the objects in the system that contain CPUs, and allows them to be selected. The *PCI* pane show the system's PCI devices and an indication of the reach-ability of the devices from the selected CPUs.

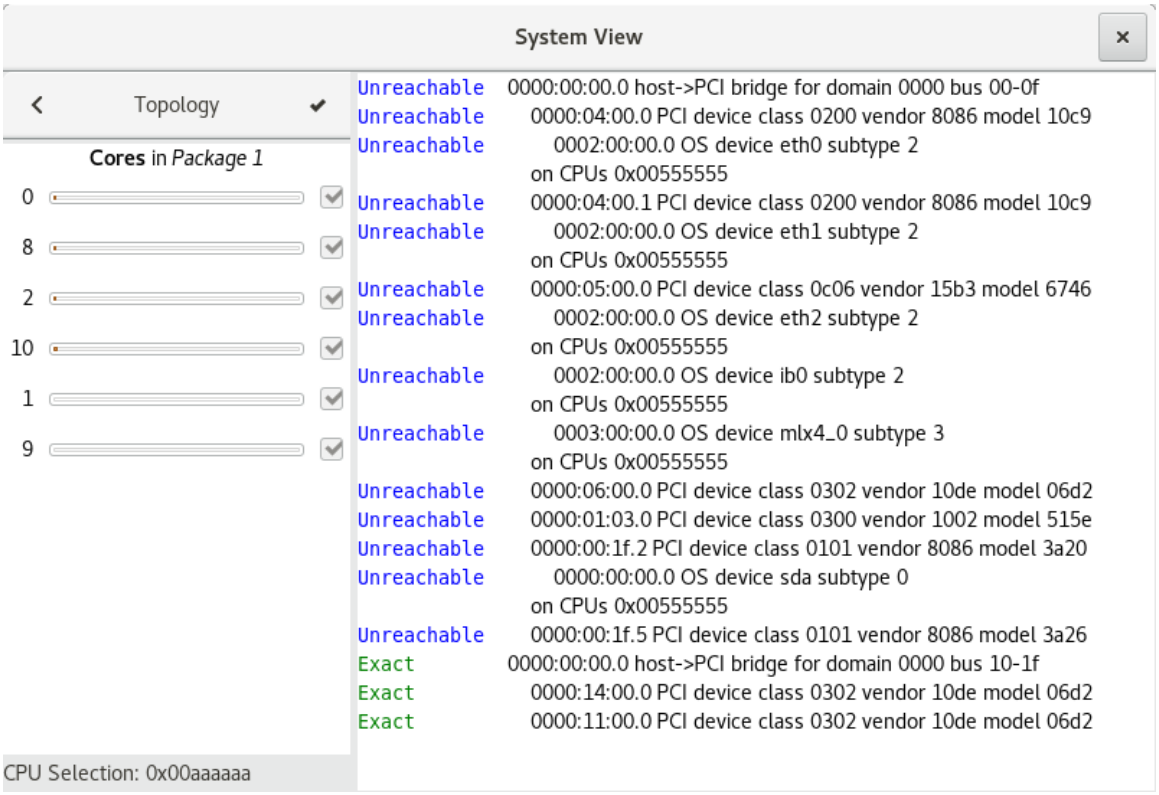


Figure 5: Example System View Window

5.1 The Topology Pane

When *Pianofish* first displays the *System View* window, the *Topology* pane displays the highest useful object level. This is usually the Machine or NUMA node level. The first line of the main body of the pane describes the current view level and object. The user can click on an item in the list to change the display to that item (unless the item is the lowest useful level). The back arrow in the pane's header bar allows the user to move the display up a level.

In the example shown in figure 5, the user has selected to view the package (usually equivalent to a processor socket) with the physical ID 1. The package contains cores with physical IDs 0 though 2 and 8 through 10. Pianofish shows these cores in the order in which the system enumerates them. The gauge for each core shows an average CPU usage for all of the processors in the core.

The check-box for each core indicates that all the processors contained in this core are in the current CPU selection. If a core had only some CPUs selected, *Pianofish* would show a dash instead of a check. If a core had no selected processors, the check-box would be blank.

To select or deselect an item in the *Topology* pane list, the user clicks the check-box in the *Topology* pane header. The *System View* will enter selection mode, and the user can click on the checkboxes of the list items to change whether the processors it contains are in the current CPU selection. The user can click the OK button to accept changes or the "Go Back" icon button to cancel the changes. The current *CPU Selection* takes effect immediately.

Processor selection can be done at any level. For instance when the user is viewing NUMA nodes in the Topology pane, clearing the check-box for a NUMA node deselects all the processors in that NUMA node.

5.2 The PCI Pane

The *PCI* pane shows some basic information abou PCI devices. Beside each PCI device is a word that describes the reach-ability and coverage of the current CPU selection. Here are the words and their meanings:

Exact The current CPU selection exactly matches the CPUs that can access this PCI device

Unreachable The currect CPU selection contains no CPUs that can access this device.

Reachable All the selected CPUs can reach the PCI device, but some other CPUs could as well

Mismatch Some of the selected CPUs can access the PCI device, and some of them cannot

The *PCI* pane is updated dynamically while the *Topology* pane is in selection mode. The user can see how selection changes affect the reach-ability of PCI devices before commiting a change in the CPU selection.

6 Preferences

The user can start the *Preferences* dialog by clicking the gear icon on the *Header Bar* of the *Main Window*, or by selecting *Preferences* from the application menu presented by the desktop manager, if available. An example of the *Preferences* dialog is shown in Figure 6 on the next page.

The Update Interval is the number of seconds between refreshes of the *IRQ* and *Process* panes. When *Pianofish* refreshes the panes, it displays information that has changed since the last update in bold text.

The *Process Filter* entry box can accept either a regular expression ot a bash-style glob. *Pianofish* applies the filter expression to the entire command-line of each process.

To reduce the amount of data displayed in the *Process* pane, The user can choose to show kernel threads, user threads, or both, and also whether to show the Cgroups column. Note that when the Cgroups column is turned on or off, Pianofish will re-draw the entire *Process* pane, which can take a little time.

Preferences are preserved between invocations of the *Pianofish* application, and every user has a unique preferences set.

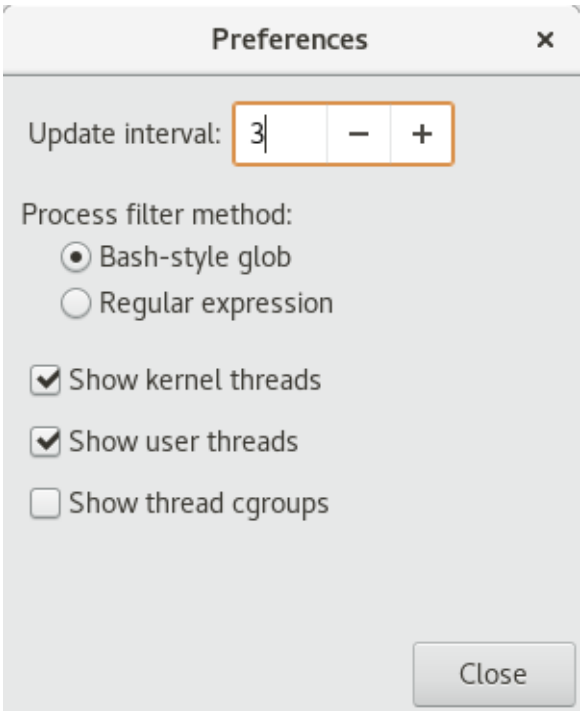


Figure 6: Example Settings Dialog