

Quick Performance

User’s Manual

cpe 631

Kyle Ray

April 30, 2018

Contents

[Goal of Quick Performance 2](#_Toc512548373)

[Software Dependencies 2](#_Toc512548374)

[What is Available in Quick Performance? 3](#_Toc512548375)

[Topology Tab 3](#_Toc512548376)

[Perf STAT Tab 4](#_Toc512548377)

[Performance Group Selection Tab 5](#_Toc512548378)

[Performance Metric Selection Tab 6](#_Toc512548379)

[Feature Control Tab 7](#_Toc512548380)

[CPI Stack Tab 8](#_Toc512548381)

[Application Output Tab 9](#_Toc512548382)

[Perfscope 9](#_Toc512548383)

[How to Use Quick Performance 10](#_Toc512548384)

# Goal of Quick Performance

Quick Performance is meant to be a simple to use performance tool. It uses pre-existing tools, such as LIKWID and PERF, and wraps them in an easy to user interface so that a user can gather, rather quickly, performance data on their application or algorithm without any prior knowledge of instrumentation tools or profilers.

# Software Dependencies

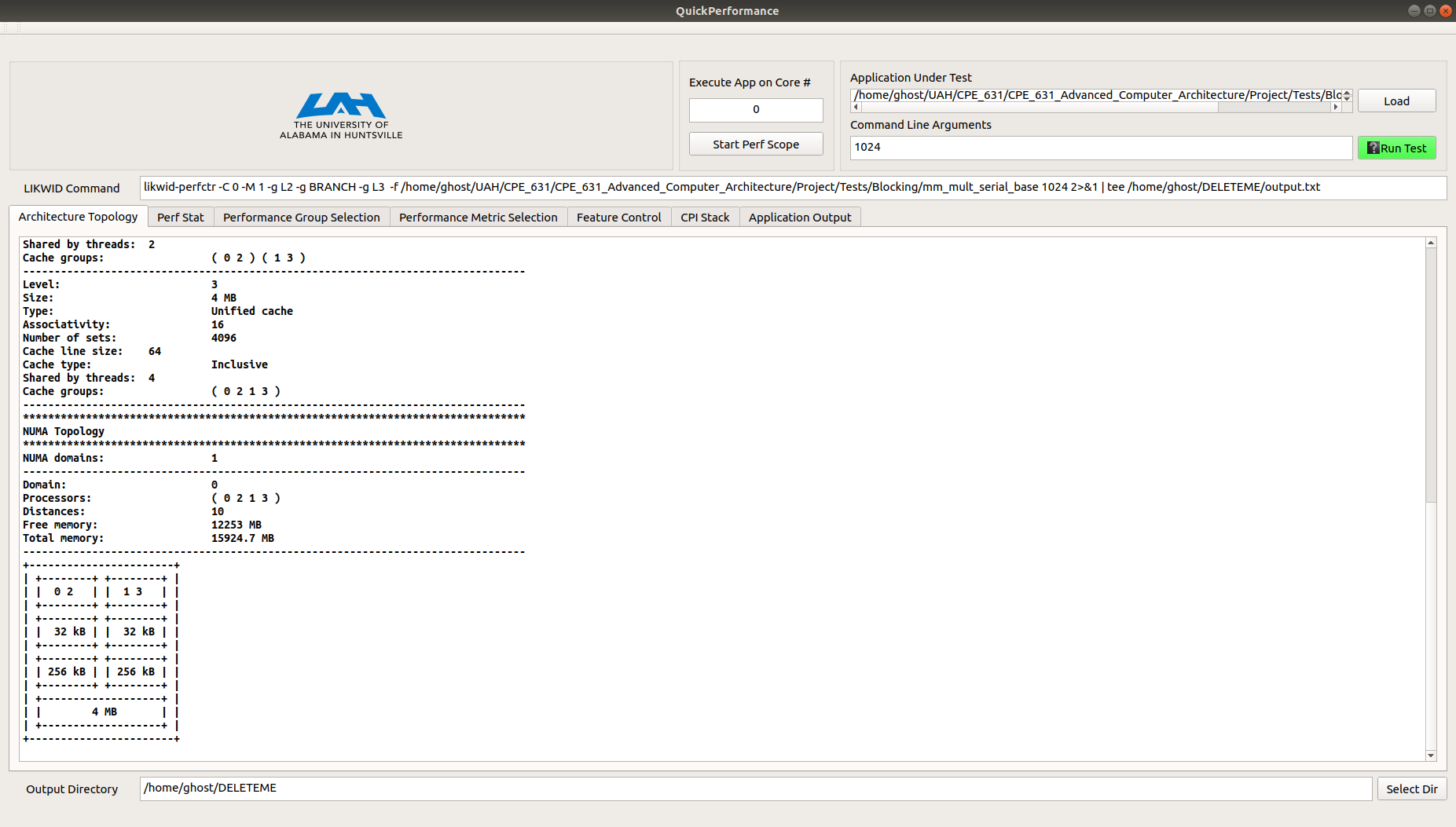
Quick Performance uses a few tools to provide feedback to the user and these tools will need to be pre-installed for Quick Performance to work.

1. LIKWID (Like I Know What I’m Doing)
   1. “Easy to use yet powerful performance tools for the GNU Linux operating system”
   2. Releases can be found at: <https://github.com/RRZE-HPC/likwid/releases>
   3. Follow the installation instructions found on the site
2. PERF
   1. Can be installed via linux package manager (part of linux-common-tools)
3. Python
   1. Standard on Linux should be pre-installed, if not it will reside in the distributions repository

# What is Available in Quick Performance?

## Topology Tab

Upon opening the Quick Performance application, you will be greeted with the “Architecture Topology Tab” which is the result of the likwid-topology command ran on startup. The information is displayed in plain text so that you will have a quick representation of the system architecture.

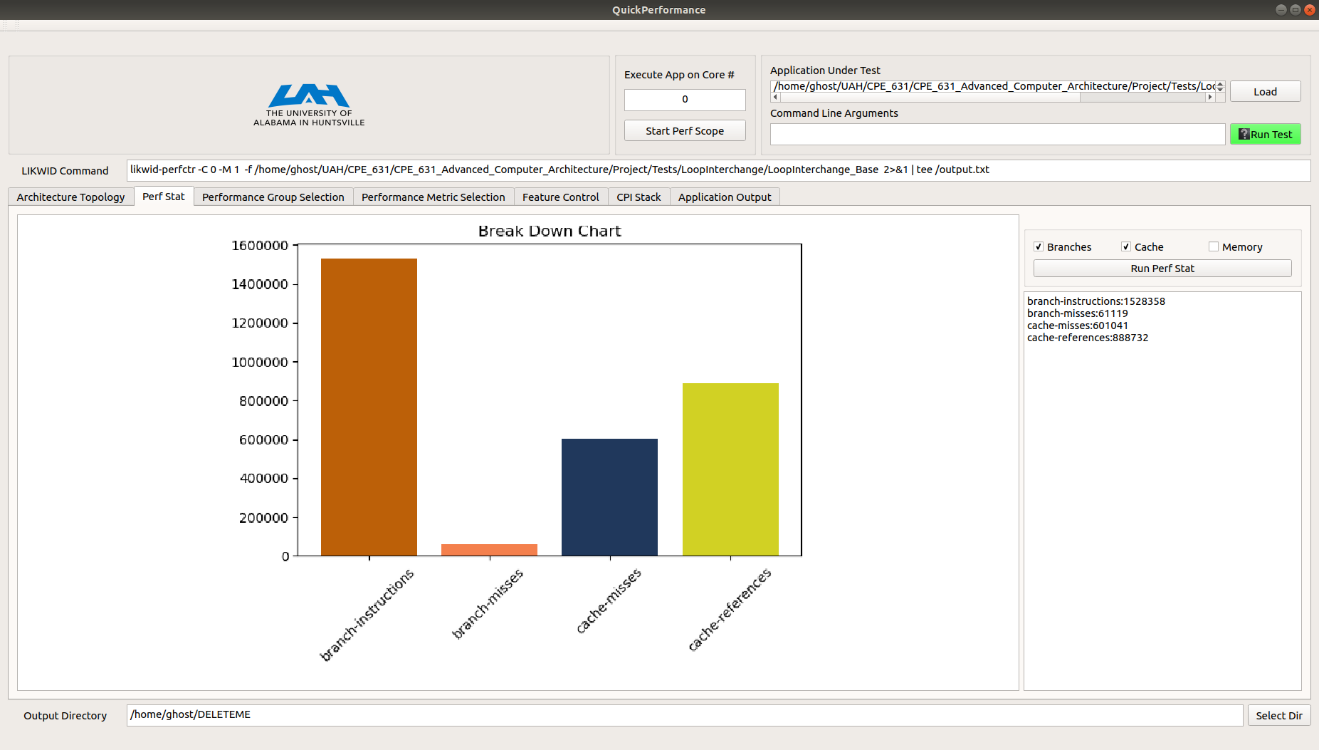


**Figure 1: Quick Performance - Architecture Topology**

This is meant to give a quick overview of the underlying computer architecture to hopefully aid in the software development process if one is seeking to increase performance via software optimization techniques for a certain architecture.

## Perf STAT Tab

The second tab in the list is a stand-alone process to give the user immediate feedback on branching, caching, or memory. Allows the user to choose a combination of the above metrics, or choose them all, and proceeds to utilize perf stat to gather information and display the counts as well as create a graphical bar graph to visualize the data. Just make your selections and press the “Run Perf Stat” button to generate results.



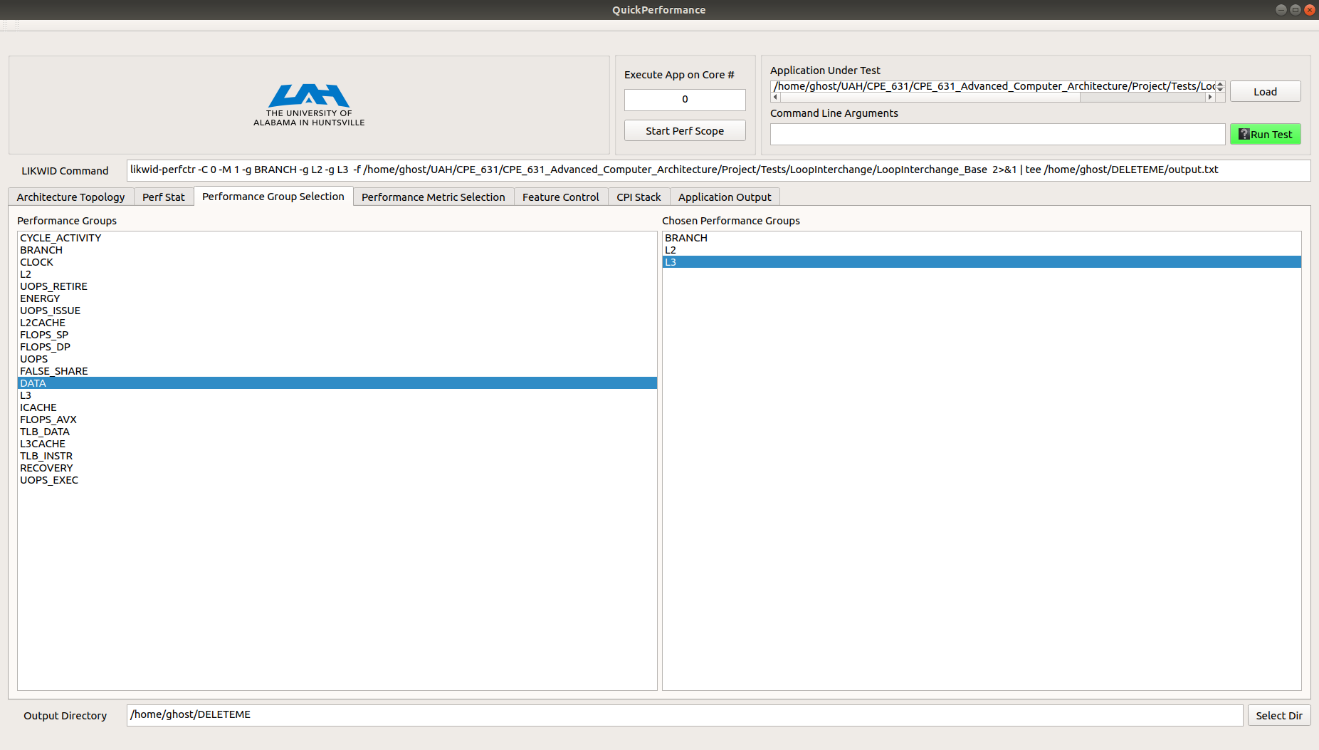
**Figure 2: Quick Performance - Perf Stat Tab**

Note: If you do not see results after pressing the “Run Perf Stat” button, try the following.

1. Run Quick Performance from the terminal to allow for error information from stderror
2. Check that perf is installed on the current machine
3. Modify the perf\_event\_paranoid file
   1. sudo sh -c ‘echo 0 >/proc/sys/kernel/perf\_event\_paranoid’
   2. This will update the paranoid file to allow for CPU monitoring
   3. To make the change permanent you can update for boot settings to do this on startup.

## Performance Group Selection Tab

The Performance Group Selection Tab and the following tab, Performance Metric Selection, allow the user to select from pre-defined scripts to measure certain metrics or the ability to choose from a list of supported events. Double clicking a metric will add it to the chosen performance groups list. If more than one performance group is selected, LIKWID will perform a round-robin method of multiplexing the groups and when they are counted. While it is possible to test with more than one performance group at a time, caution should be used because of the round-robin multiplexing short-running measurements can carry large statistical errors.



**Figure 3: Quick Performance – Performance Group Selection**

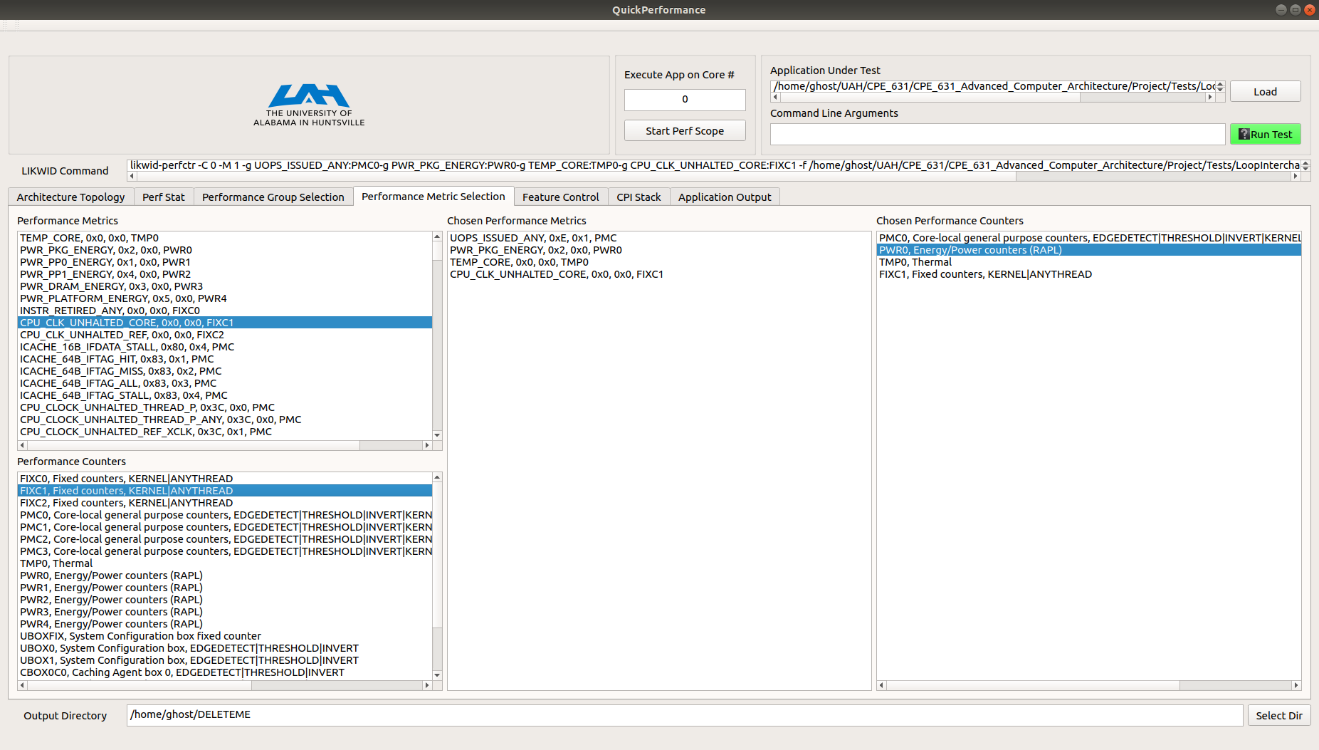
Note: Choosing performance groups will disable the following tab, Performance Metrics, as Quick Performance cannot guarantee that specific counters aren’t already being used by a performance group.

It is possible to create your own performance group. Just follow the examples given in the groups section for your architecture, located in the LIKWID install directory. Once it is in this directory Quick Performance will read it in and you will be able to use it within the application.



## Performance Metric Selection Tab

The Performance Metric Selection Tab will allow the user to select from a list of events that are available for their architecture. Double clicking a metric will add it to the chosen performance metrics list. When choosing a metric, the user will have to choose an appropriate counter to monitor that metric. To choose a counter, match the last string in the metric name with the name of a counter. For example, the metric *UOPS\_ISSUE\_ANY,0xE, 0x1, PMC* will need to be paired with a general performance monitoring counter (PMC) such as *PMC0, Core-local general-purpose counters*. Refer to the figure below for a set of metrics with their appropriate counters.

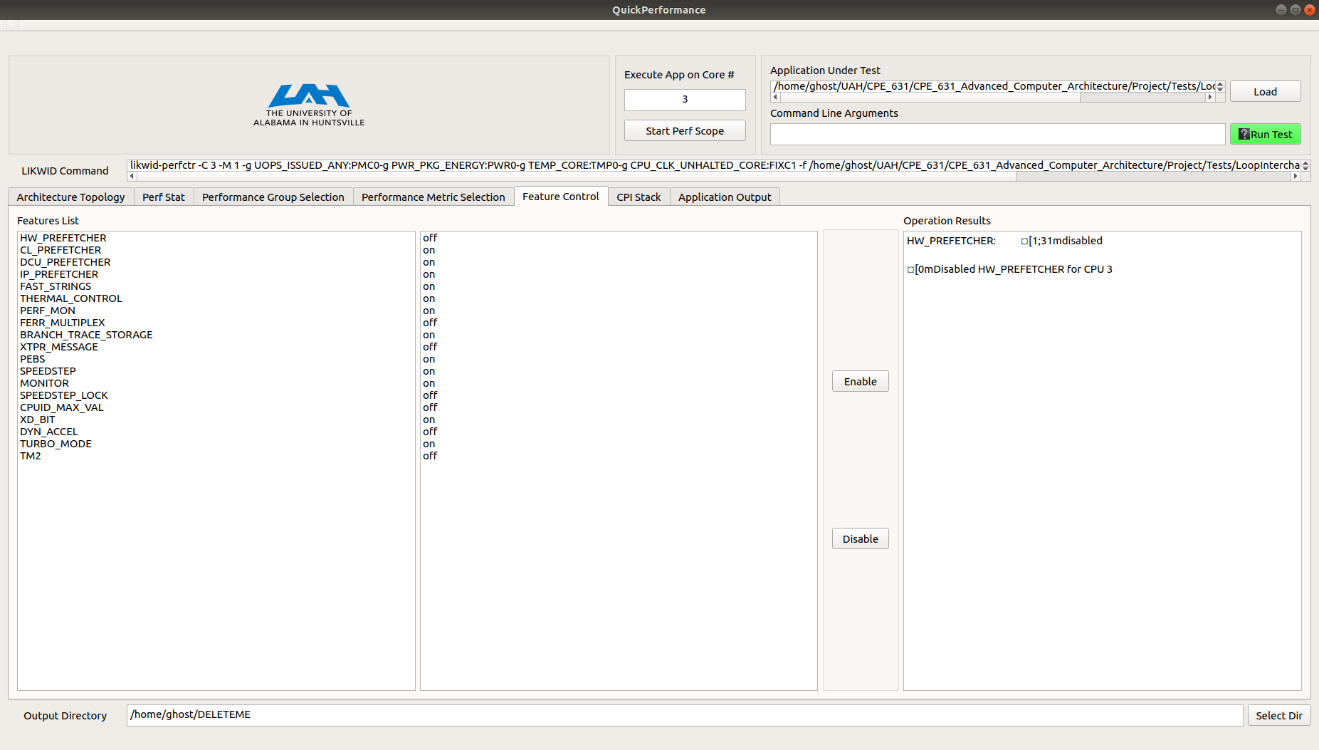


**Figure 4: Quick Performance – Performance Metric Selection with Counter Selection**

Note: It is possible to monitor as many metrics as you have counters available. When choosing performance metrics, the Performance Group Selection Tab will be disabled because Quick Performance cannot guarantee that performance groups won’t try and use the same counters selected by the user.

## Feature Control Tab

The Feature Control Tab will allow the user to enable/disable certain features that are available on the CPU. To enable/disable a feature just select it from the features list and hit either of the buttons Enable or Disable to toggle the feature, the results of the operation will be displayed in the Operation Results text box. This can be used in conjunction with the “Execute App on Core #” edit box, changing the number value in this box will pull up the features for that CPU core.

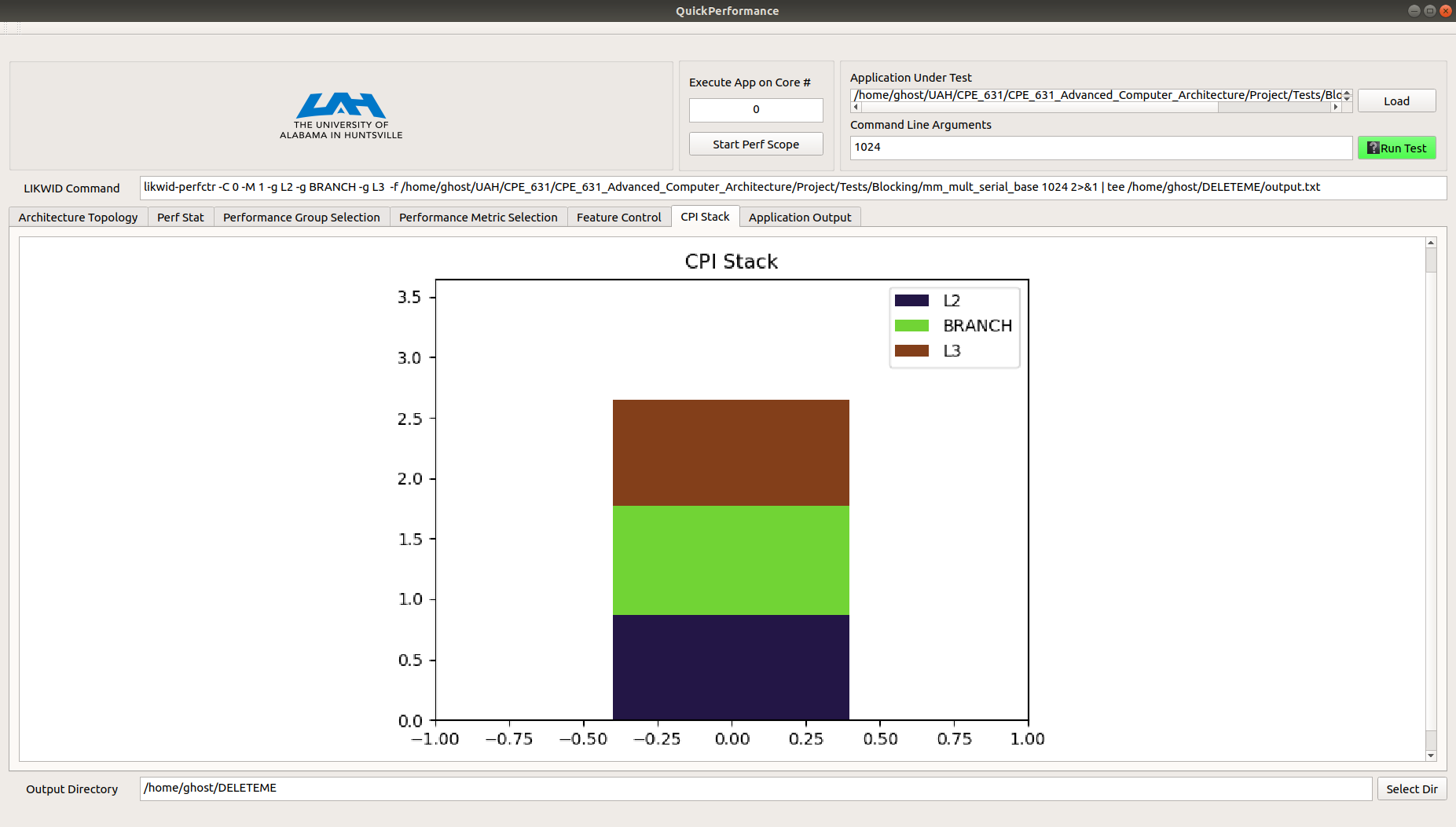


**Figure 5: Quick Performance – Feature Control**

In the image above, we have elected to turn off the hardware pre-fetcher for the CPU core number 3.

## CPI Stack Tab

The CPI Stack Tab is used to show the overall CPI of a performance group. An image will be generated after the user elects to “Run Test” with their current selections from the other tabs. This will run the LIKWID command and pass the data to a python script which in turn will process it into a stacked bar graph displaying the CPI for each performance group category.

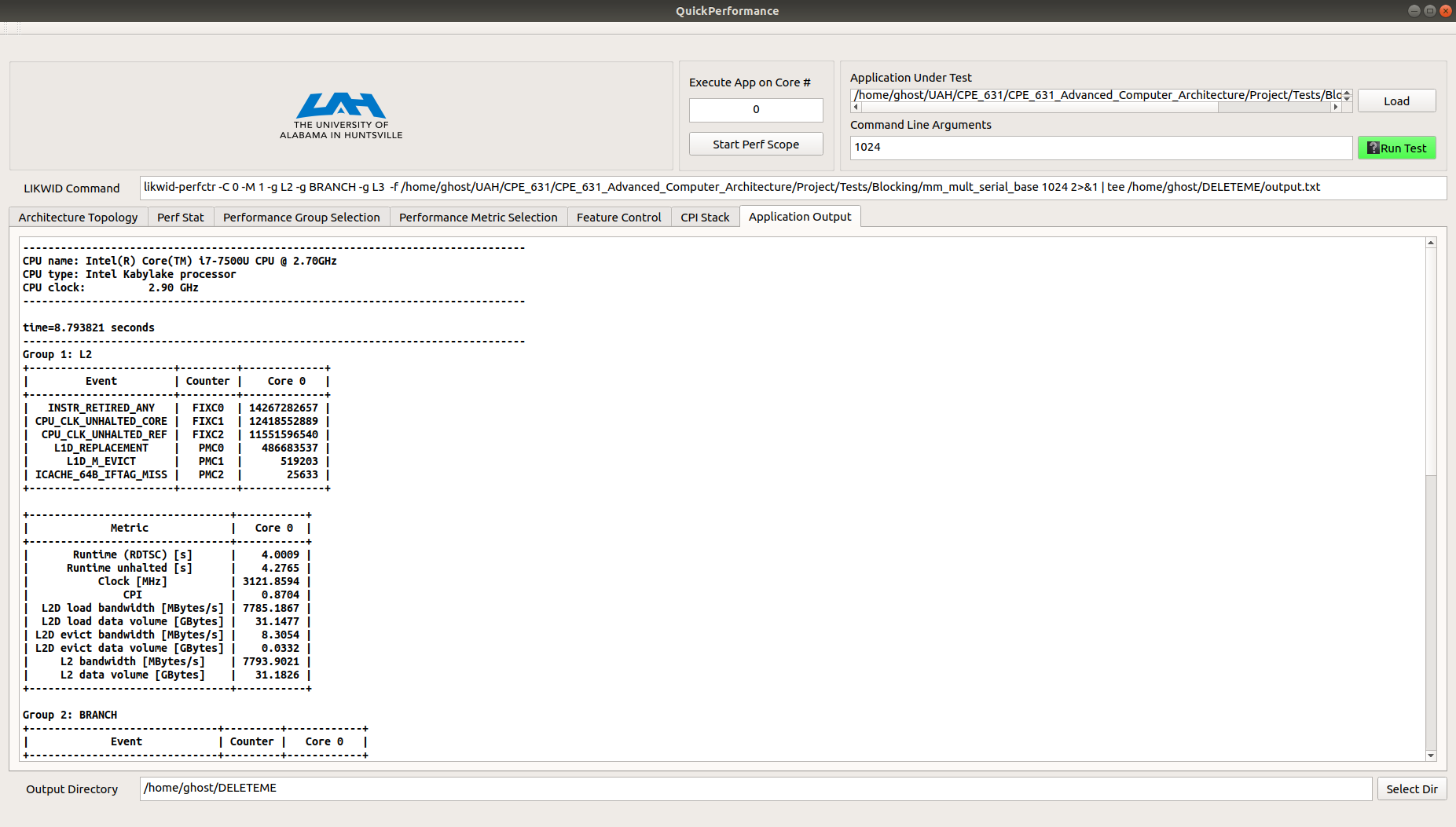


**Figure 6: Quick Performance – CPI Stack**

**TODO:** Because of recent findings within LIKWID, this tab does not function correctly and should be removed if no solution is found to correct the error. Currently displays the overall CPI for the program and not for each individual performance group as originally thought.

## Application Output Tab

The Application Output Tab will automatically pull up upon completion of the test and will display the results of the test. The contents of this tab are directly dependent on the choices that are made in the other selection tabs. The header will include some CPU information followed by the output of your application and the counter results. All data will be saved to the selected output directory as “output.txt”.



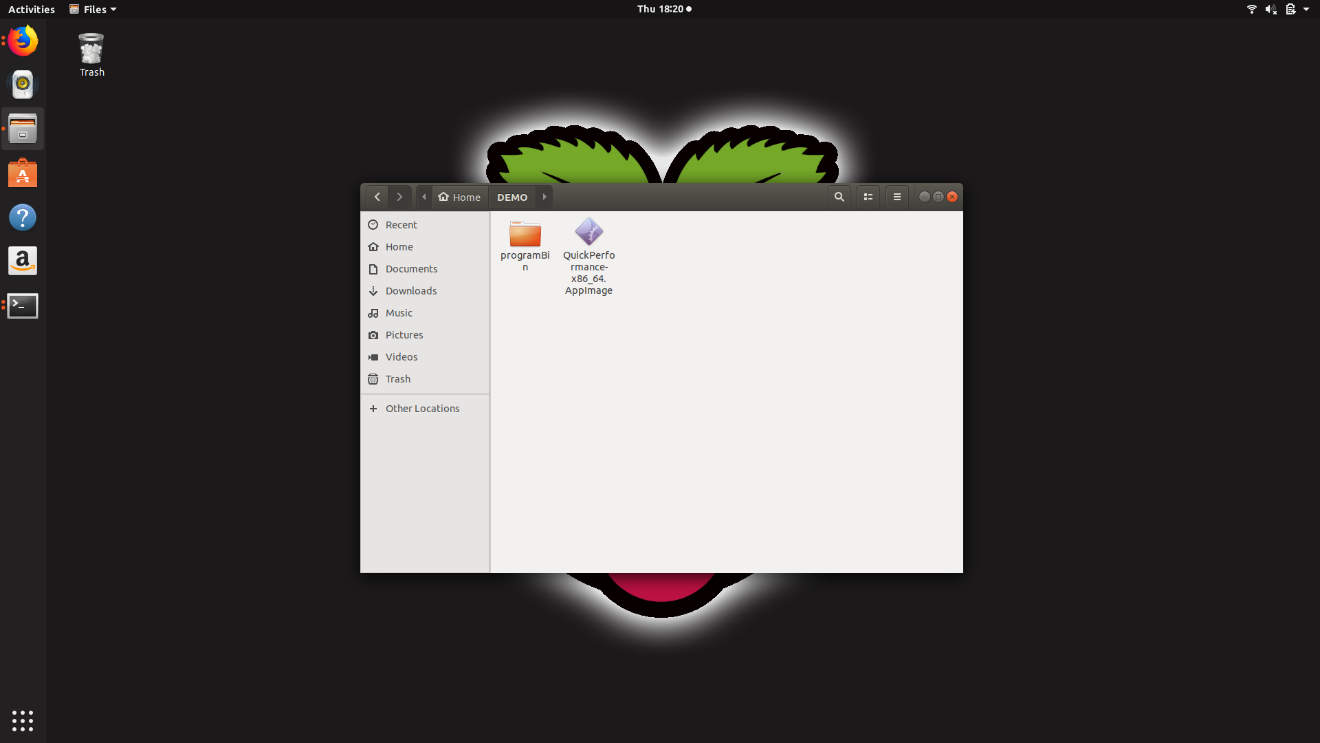
**Figure 7: Quick Performance – Application Output**

## Perfscope

This feature is currently being integrated and developed and is not guaranteed to work. Pressing the “Start Perf Scope” button will trigger the LIKWID perfscope to pull up gnuplots and plot power and energy consumed for the current CPU in real time.

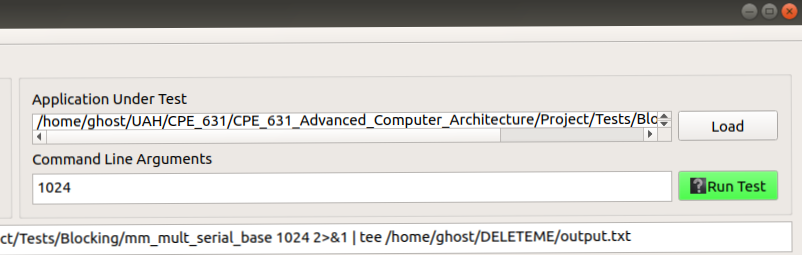
# How to Use Quick Performance

Quick Performance is compiled into an AppImage, which contains all the necessary libraries to run the application.



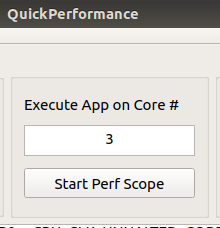
**Figure 8: Quick Performance AppImage and ProgramBin Folder Layout**

To run Quick Performance simply call the AppImage from the command line or double click the AppImage to launch the application. Upon launch you will be greeted with the Topology Tab, see above for description. From here you can elect to just make a quick run using the Perf Stat tab or to move on to the LIKWID tabs and select from performance groups or more specific metrics. While interacting with Quick Performance you may notice that the LIKWID command will update based on your selections. This is placed here so that it is possible to copy out and be used externally if desired, note that editing of the command inline is not supported. Once you have made your selections you will need to load your application executable by using the “Load” button. If your application requires any command line arguments to be run, then please provide them in the Command Line Arguments text box, refer to the figure below.



**Figure 9: Load Application and Command Line Arguments**

If you would like to change features, you can do so before running a test via the Feature Control Tab. The “Execute App on Core #” text box will allow you to select the CPU core to pin, execute, and monitor your application on.



**Figure 10: Core Selection**

It should be noted that the current implementation of Quick Performance does not support multi-threaded applications.