Mox Job Profiling

A Grafana dashboard containing graphs of job resource usage over time is available at:

https://job-profiling.hyak.uw.edu

The dashboard can be reached from the campus network, or via Husky OnNet VPN. Login is via UW NetID, and it is available to all Hyak users.

The data for the dashboard is collected from Slurm utilizing Slurm's InfluxDB Profile Accounting Plugin. Note that the plugin stores the profiling data in a buffer on each node, sending data to the profiling database only when the buffer fills or a task ends. Therefore, dashboard data will arrive in chunks and can lag as much as 10 minutes behind real time. Note that for multi-node jobs, data from different nodes may arrive at different times.

Example Jobs

Long running checkpoint job – can see it bouncing between nodes. Weird diagonal lines are when it lands on a node it was previously running on so the lines connect.
https://job-profiling.hyak.uw.edu/d/U3WlCDRZz/job-profiling?orgId=1&from=1555830000000&to=1556311168417&var-job=717716&var-host=All&var-step=All&var-task=All

Long running multi-node job that sure seems like it is wasting a lot of resources (only one node well utilized, others hardly doing anything):
https://job-profiling.hyak.uw.edu/d/U3WlCDRZz/job-profiling?orgId=1&from=1555688820227&to=1556311917176&var-job=760793&var-host=All&var-step=All&var-task=All

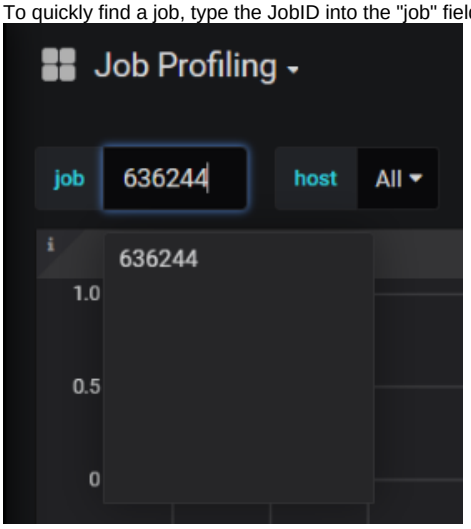
A job doing 3MB/s writes for a bit:
https://job-profiling.hyak.uw.edu/d/U3WlCDRZz/job-profiling?orgId=1&from=1556168110138&to=1556168468615&var-job=636230&var-host=All&var-step=All&var-task=All

Job Profiling Graphs

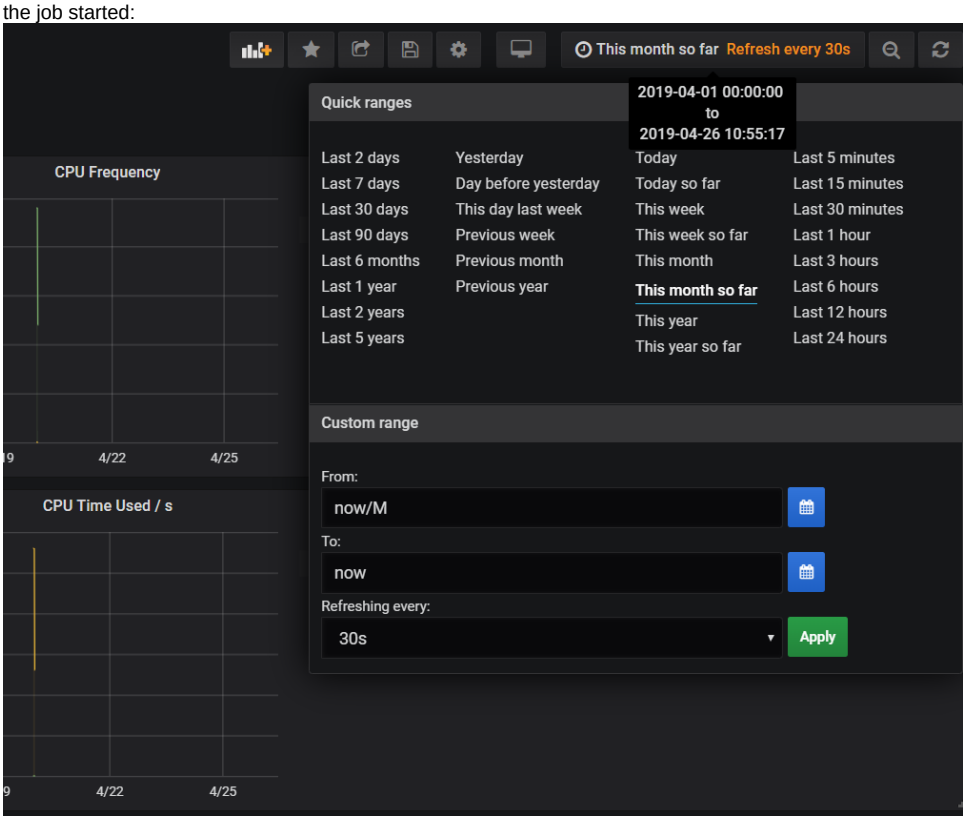
CPU Frequency	The average CPU frequency of CPUs allocated to the task. Note that Intel processors can scale up core frequencies when there are idle cores and there is thermal and power overhead to do so. Also note that certain operations, like waiting for IO, can cause a core to run at a lower frequency even though it is utilized.
CPU Time Used per second	The average amount of total CPU Time consumed per second by the task during the (30 second) profile sample. A fully utilized CPU core will consume 1s of CPU Time per second. A step that is fully utilizing all cores on a 28 core node would consume very close to 28 CPU seconds per second.
CPU Utilization	The total CPU utilization of the task. A value of 1.0 represents one fully utilized core. A step fully utilizing 28 cores would show utilization very close to 28.
Memory RSS	The Resident Set Size, which in practice, is the amount of physical memory consumed by the task. This is a stacked graph, so the values of the individual steps will be graphed atop one another so that the height of the top line of the graph represents the total physical memory consumed by the job tasks being displayed in the graph.
VMSize	The virtual memory usage of the task, which represents all memory allocated to a task, including memory that has been written out to swap, and memory that has been allocated but not consumed. This is a stacked graph, so the values of the individual steps will be graphed atop one another so that the height of the top line of the graph represents the total virtual memory consumed by the job tasks being displayed in the graph.
Pages	The number of pages of memory being used by a task. A memory page is a fixed-length contiguous block of virtual memory, described by a single entry in the memory page table.
Data Written and Data Read to/from Filesystem per second	The average amount of data written to or read from mounted filesystems per second over the (30 second) profile sample.

Tips

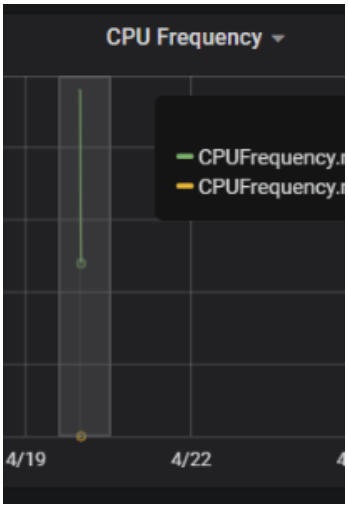
- To quickly find a job, type the JobID into the "job" field and hit enter (jobs will not always show up in the drop-down):



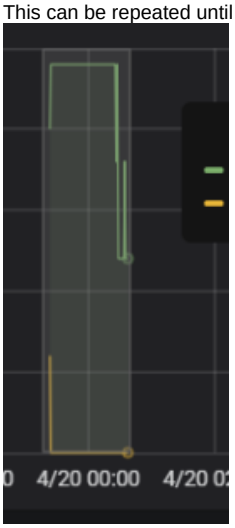
- To easily find profile data for jobs that are no longer running, click on the time range in the top right and select a time window that begins before the job started:

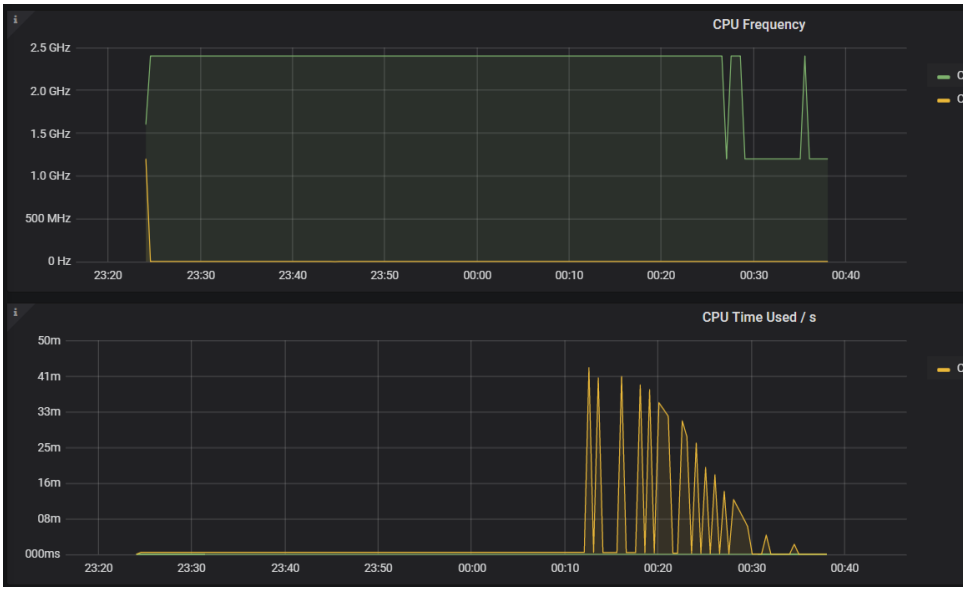


....and then draw a box around the data appears to "zoom in" on when the job was running:

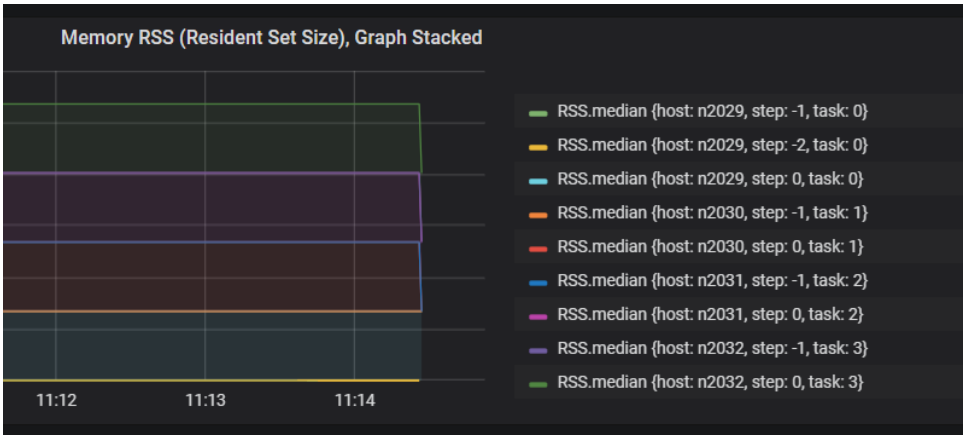


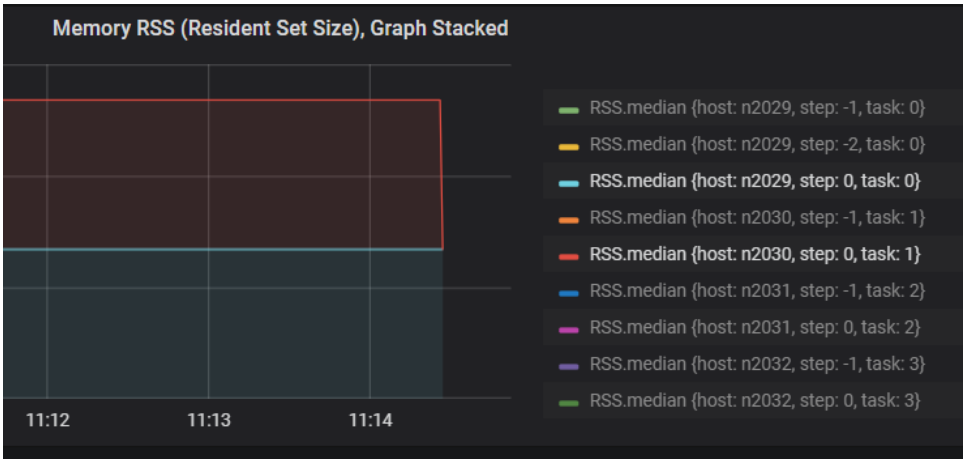
This can be repeated until a useful view of the data is achieved:



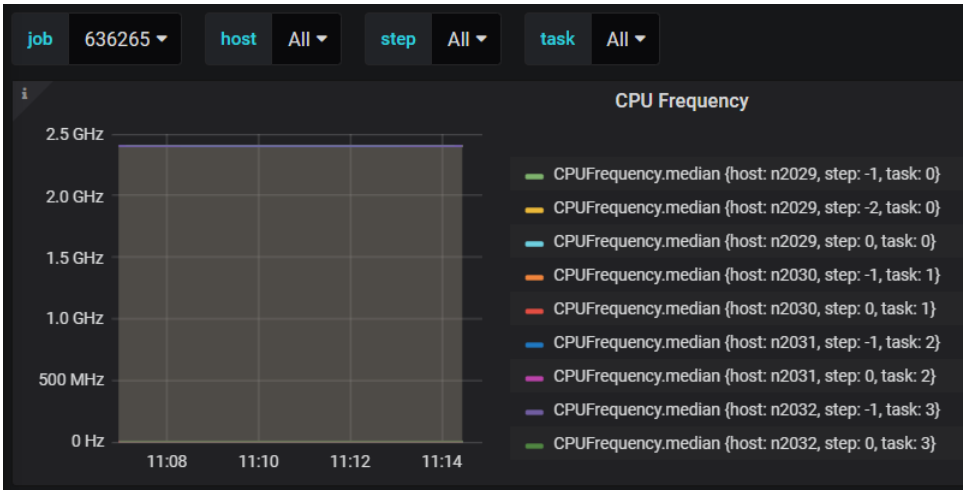


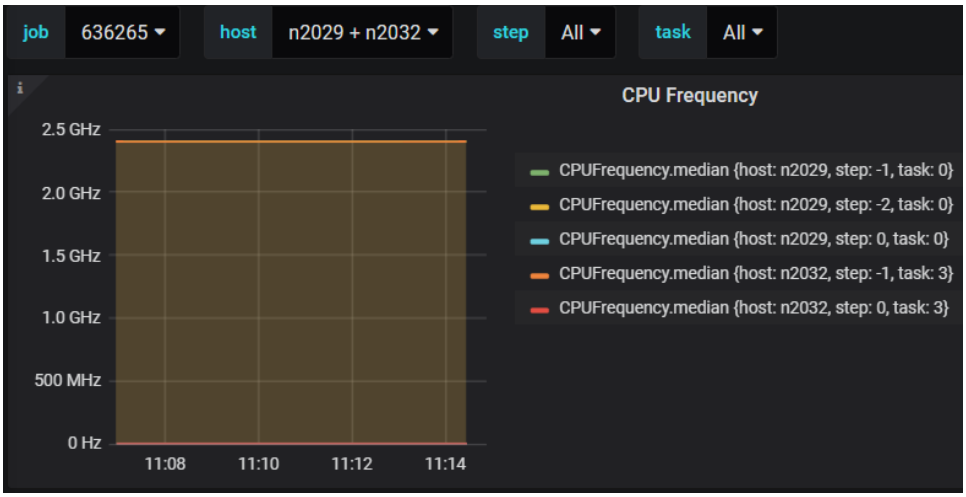
- For jobs with many nodes/tasks/steps, the graphs can be limited to particular data series by clicking to select them in the legend, multiple series can be selected by holding down CTRL and clicking :





Hosts/Steps/Tasks can be completely filtered by selecting specific series labels in the corresponding drop-down:





- Long running checkpoint jobs can have a large number of tasks, as a new series is created every time the job starts on a new node after being re-queued due to preemption or after 4-5 hours of runtime. If a job is re-queued on a node it previously ran on, you may see a line that connects where it stopped on the node to where it started again on the same node. This is artifact of the graphing, and should not be interpreted to mean that the job was active on that node for the entire time period implied by the graph.