Try experimental React UI

■ Enable query history

sum by (mode, instance) (rate(node_cpu_seconds_total{job="node"}[1m])) Load time: 239ms Resolution: 14s Total time series: 16 Execute - insert metric at cursor -**Console** <u>Graph</u> **>>** Res. (s) 1h Until stacked 0.8 0.6 0.4 0.2 15:45 16:00 16:15 16:30 ✓ □ {instance="172.31.95.36:9100",mode="user"} ✓ □ {instance="172.31.95.36:9100",mode="system"} ✓ □ {instance="172.31.95.36:9100",mode="steal"} ✓ □ {instance="172.31.95.36:9100",mode="softirq"} ✓ □ {instance="172.31.95.36:9100",mode="nice"} √ □ {instance="172.31.95.36:9100",mode="irq"} ✓ □ {instance="172.31.95.36:9100",mode="iowait"} ✓ □ {instance="172.31.95.36:9100",mode="idle"} ✓ □ {instance="172.31.83.151:9100",mode="user"} ✓ □ {instance="172.31.83.151:9100",mode="system"} ✓ □ {instance="172.31.83.151:9100",mode="steal"} ✓ □ {instance="172.31.83.151:9100",mode="softirq"} ✓ □ {instance="172.31.83.151:9100",mode="nice"} ✓ □ {instance="172.31.83.151:9100",mode="irq"} √ □ {instance="172.31.83.151:9100",mode="iowait"} ✓ □ {instance="172.31.83.151:9100",mode="idle"} Remove Graph node_memory_MemFree_bytes Load time: 173ms Resolution: 14s Total time series: 2 node_memory_MemFree. Execute <u>Graph</u> <u>Console</u> Res. (s) 1h Until **>>** stacked $ec2-44-211-69-249.compute-1.amazonaws.com: 9090/graph?g0.range_input=1h\&g0.expr=sum\ by\ (mode\%2C\ instance)\ (rate(node_cpu_seconds_total\%7Bjob\%3D"node"\%7D\%5B1m\%5D))\&g0.tab=0\&g1.range_input=1h\&g0.expr=sum\ by\ (mode\%2C\ instance)\ (rate(node_cpu_seconds_total\%7Bjob\%3D"node"\%7D\%5D)$

Add Graph