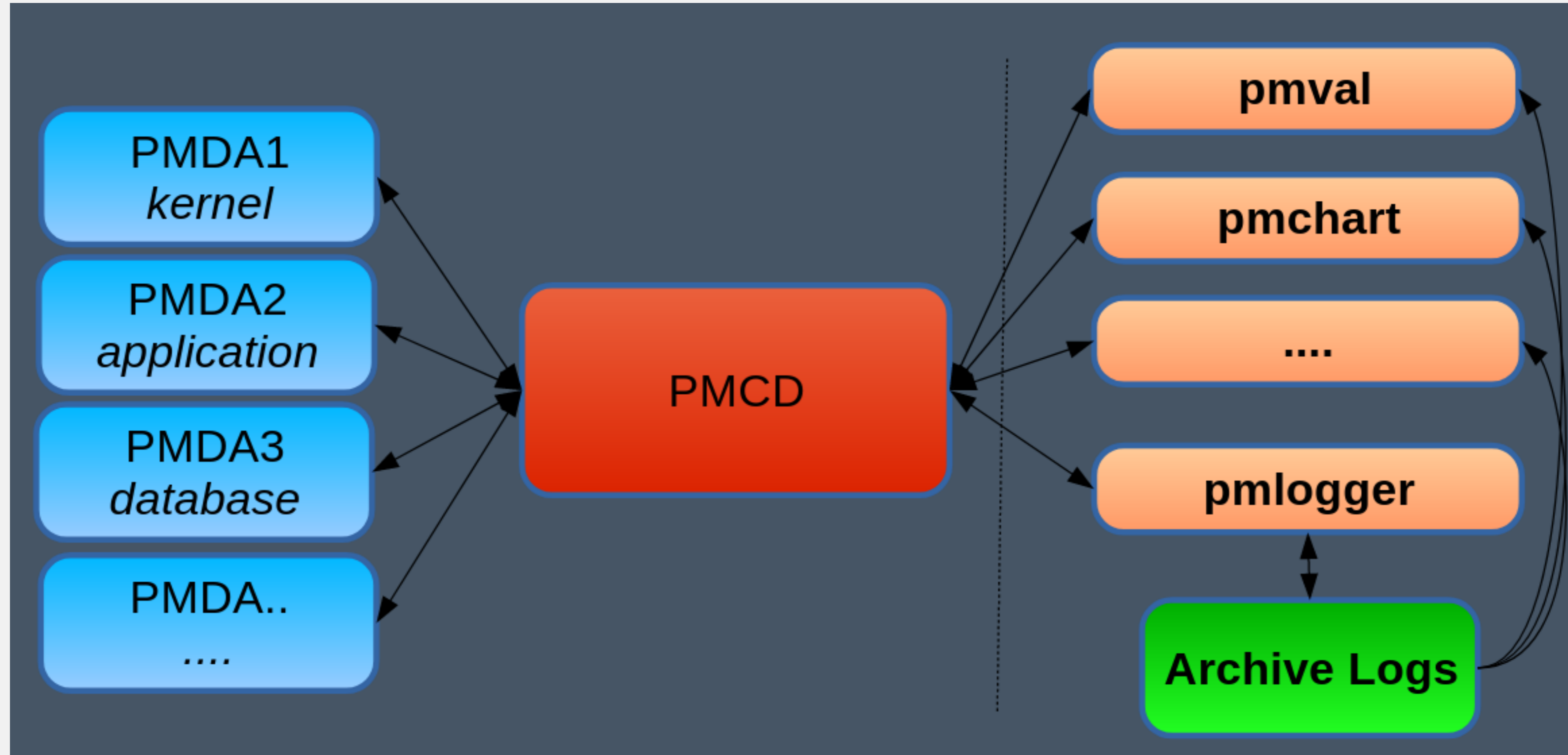


About me

- **Nikhil Kshirsagar (nkshirsa on freenode, email: nkshirsa@redhat.com)**
-
- **SSME with Storage, CEE, Red Hat**
-
- **15 years of experience, mostly development of enterprise storage and performance products.**
-
- **Use PCP for customer cases,**
- **Worked on some patches, 2 features, filed RFE's, etc.**

Performance Co-Pilot is a system level performance monitoring and management suite



Performance Metric Domain Agents (PMDAs) collect and export performance metrics

The Performance Monitoring Collection Daemon is deployed on the system(s) to be monitored

Client tools are deployed on user workstations

WHY PCP ?

- **Better than sysstat**
 - replaces iostat, dmstat, free, numastat, uptime, vmstat, mpstat, netstat, ipcs, ps, pidstat, atop, collectl, sar, tapestat, iotop, htop, top, systemtap, and many more
 - sysstat tools only allow live monitoring, so issues need reproducers. sar can only replay sadc binary files that were captured on the same version and platform/arch (pcp archives are portable)
- **Can replay archives at different sampling intervals than the collected intervals**
 - Metric values are appropriately interpolated except strings, and discrete ones.
- **Temporal index in archive files (.index) makes it easy to seek anywhere in the archive rapidly**
- **pcp-zeroconf package makes it easy for customers to install and monitor**
-
- **RFE for 7.6 install to have pcp installed by default**
 - - https://bugzilla.redhat.com/show_bug.cgi?id=1591135
- **Supported by Red Hat !**

What is pcp-pidstat -B?

- Reports states of processes live or historically
- Detailed reports show how much a process has spent in each state over time
- “All” reports on processes in current state (at time of sampling)
- Total time shown (USP !)
- Very useful to understand for example, how long has process “X” spent in state “Y” in the last few minutes/days/months/years/...
- Feel free to suggest additional features

Pending work in pcp-pidstat -B

- Transition matrix – how many times has a process changed states from x->y
- Sorting the processes according to time in particular state
- Disallow invalid process states after -B (pcp pidstat -B R,S,J)
- Recognize incorrect usage like -B detail abc, -B R abc , etc
- Username filters still to be implemented - just ignores the filter for now
 - Filter processes shown for a particular user
- Dont print headers if no samples to be printed – done
- Percentages spent in states ?
 - for eg. a process has been in D state 80% of the time in a time window.
- Add -z to pcp-pidstat, for timezone specification
 - vaibhav submitted <https://github.com/performancecopilot/pcp/pull/512>

Quick Demo