Cluster-wide Performance Monitoring Working Group

General structure of a monitoring system

- Monitoring agent (for compute and management nodes)
- Transport protocol (HTTP, MessageQueues, Shared-FS)
- Aggregation agent (Intermediate routers in tree)
- Storage agent (Logfiles, Databases, …)
- Visualization agent (Webpage, PDFs, …)
- Analysis agent (MachineLearning, …)

- Different systems use different monitoring stacks
- Systems installed by the vendors might not be what you need

General questions

- What metrics? For what purpose?
  - Collect anything is not feasible
  - Which group needs which metrics?
- Who controls/feeds the monitoring?
  - Always on? User can control detail level?
  - User can add own application data to monitoring
    (-> Ubiquitous Performance Analysis)
- Who should see the data in the end?
  - Users: Just overview (red, yellow, green), Timeline data of jobs
  - Admins: Timeline? Notifications created by analysis agents?
  - Managers: Just overview? Convert to $?

What do we need

- System logs
  - Important for resilience
  - If a job crashes should the backtrace be added to system log
- Generic interfaces (Cray has them but hard to talk (NDA))
  - APIs to exchange components to fit personal needs
- Tracking of system configuration changes
- How to isolate users/jobs (authentification)

Actions based on monitoring data

- All system config manipulations should be revert back after jobs
- Node health checks (energy consumption, temperatures)
  - Regulate cooling based on monitoring data
- Job placement might affect job performance
  - All are using one I/O node, not the array of them
  - Jobs around the regarded job, correlation between jobs
  - Find most/least efficient nodes
- SC18 Paper: Better integration of GPU jobs in the scheduling system
  - Big jobs on reliable nodes, small jobs on less reliable nodes
- Analysis might be complex and creates (sometimes) more (temp) data then the original data
- GeoPM measure energy/power and control system state

Future

- Github repo for future collaboration RRZE-HPC/DFG-PE
  - Write me your GitHub acc.: Thomas.Gruber@fau.de
  - TODO: Paper collection in wiki
  - Paper: Modernizing Cray Systems management using redfish API on next generation Cray hardware
- PowerStack has layers (generic interfaces between layers) to measure/combine energy/power data.
  - What can we learn from the data?
  - Can we feed that into resource managers for actions?