Cray Advanced Platform Monitoring and Control

CAPMC, CUG 2015:
Steven J. Martin ...... (stevem@cray.com)
David Rush .... (rushd@cray.com)
Matthew Kappel ...... (mkappel@cray.com)

Safe Harbor Statement

This presentation may contain forward-looking statements that are based on our current expectations. Forward looking statements may include statements about our financial guidance and expected operating results, our opportunities and future potential, our product development and new product introduction plans, our ability to expand and penetrate our addressable markets and other statements that are not historical facts. These statements are only predictions and actual results may materially vary from those projected. Please refer to Cray's documents filed with the SEC from time to time concerning factors that could affect the Company and these forward-looking statements.

Introduction

- **Overview of CAPMC**
  - Availability
  - Functionality
  - Architecture
- **Applets**
  - Quick walkthrough of the API
- **Near-term roadmap for CAPMC**
  - In-band controls
  - Additional “Platform” use cases
  - As always, **roadmap is subject to change**…

CAPMC Overview

- **Cray Advanced Platform Monitoring and Control**
  - Cray SMW 7.2.UP02 and CLE 5.2.UP02, release in Oct-2014
  - XC30 and XC40 systems

- **Cray Advanced ~~Power~~ Platform Monitoring and Control**
  - Use of CAPMC planned for much more than just power

- **1st CAPMC release enables**
  - Power-aware scheduling and resource management

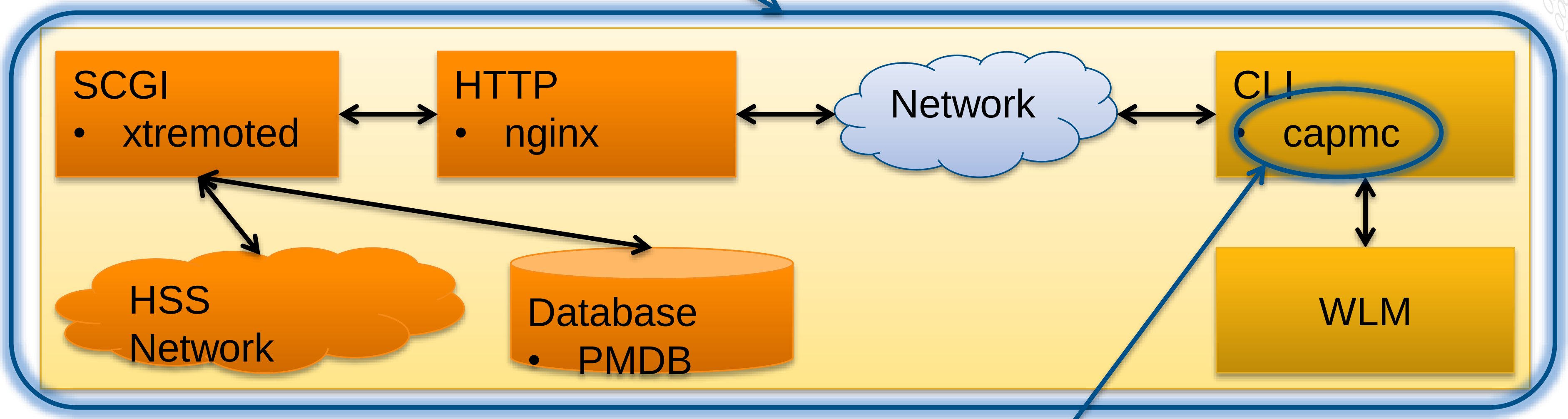
CAPMC Functionality

- Access to system- and cabinet-level power data
- Access to node-, job-, and app-level energy data
- Control of node-, job-, and app-level power capping
- Control to power on and off idle nodes

Cray supplying monitoring & control capabilities

Enabling WLM partners to innovative & manage policy

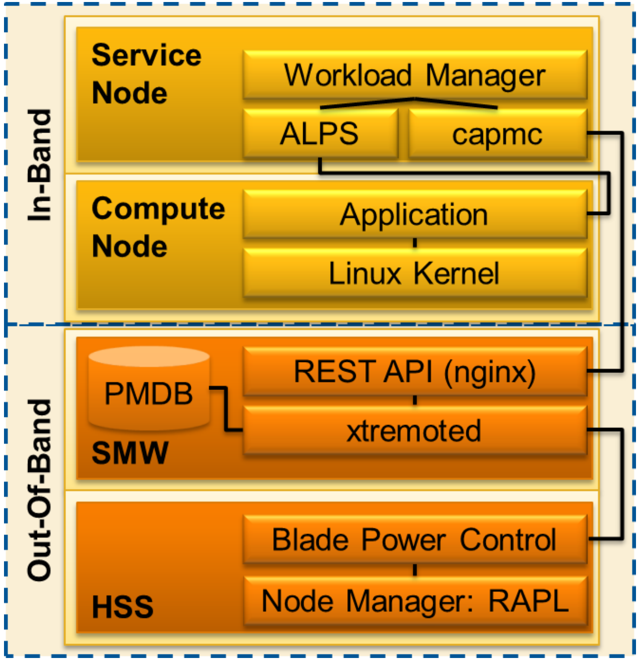
CAPMC Architecture



lower case “capmc” is the Python command line tool

CAPMC Architecture

- **Python CLI (capmc)**
  - Clients installed on select service nodes
  - Enable integration with 3rd party WLM software
- **REST API**
  - JSON data interface(s)
  - Nginx (pronounced engine-x) web server
- **Access control and security**
  - SSL & X.509
- **SMW Backend**
  - Implementing out-of-band monitoring and control functions



CAPMC Applets: System-Level Monitoring

- **get_system_power [-s start_time] [-w window]**
  - Returns system-level power data
    - Minimum, average, and maximum power for the requested time window

Time Format: 'yyyy-mm-dd hh:mm:ss'

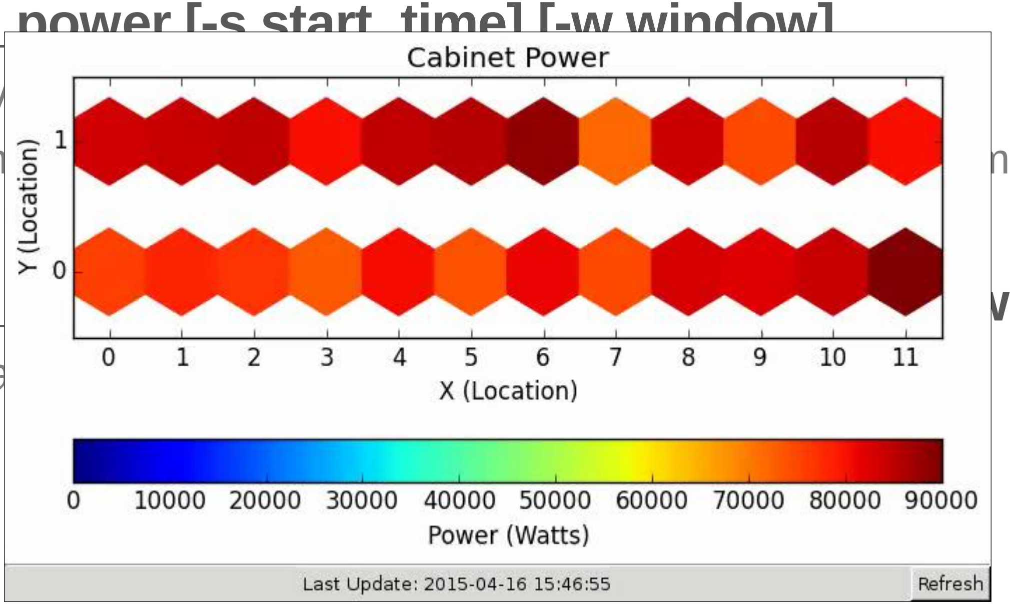
- **get_system_power_details [-s start_time] [-w window]**
  - Returns cabinet-level data for all cabinets in the system

Time in seconds

CAPMC Applets: System-Level Monitoring

- **get_system_power [-s start_time] [-w window]**
  - Returns sy
    - Minimum ne window

- **get_system_ w]**
  - Returns ca



Last Update: 2015-04-16 15:46:55

Use Case: (From our PM workshop earlier this week)
Video playback 40X real time, 24 cabinet system running HPL

CAPMC Applets: Node-Level Monitoring

- **get_node_energy_stats [-s start_time] [-e end_time] \ [--nids nid_list] [--apid apid] [--jobid job_id]**
  - Returns statistics for node-level energy (fixed size response)
- **get_node_energy [-s start_time] [-e end_time] \ [--nids nid_list] [--apid apid] [--jobid job_id]**
  - Returns node-level energy data (one record for each node)
- **get_node_energy_counter -t time [--apid apid] [--jobid job_id] \ [--nids nid_list]**
  - Returns raw accumulated energy counter data (one record for each node)
  - Multiple calls needed, raw counters used for delta calculations

Given an apid, CAPMC can use start_time, end_time, and the nid_list from the PMDB

CAPMC Applets: Node-Level Monitoring

- **get_node_energy_stats [-s start_time] [-e end_time] \ [--nids nid_list] [--apid apid] [--jobid job_id]**

WLM Use Case:

- Supporting interactive user queries on power/energy of their job(s)
- Tracking app-, or job-level power/energy to enable dynamic power scheduling

Additional use cases covered in our paper

- **get_node_energy_counter -t time [--apid apid] [--jobid job_id] \ [--nids nid_list]**
  - Returns raw accumulated energy counter data (one record for each node)
  - Multiple calls needed, raw counters used for delta calculations

CAPMC Applets: Node Power ON | OFF

- **node_on --nids nid_list**
  - Turn-on nodes and boot Linux making them ready to run jobs
- **node_off --nids nid_list**
  - Shutdown Linux and power off the nodes
- **node_rules**
  - Returns information to the WLM w/respect to node on/off operations
  - Allows system admin to establish constraints
- **node_status [--nids nid_list] [--filter 'opt|opt|opt|...']**
  - Returns current status for requested nodes
  - Allows WLM to poll for status of nodes it has powered on/off
  - Filters: show_all, show_off, show_on, show_halt, show_standby, show_ready, show_diag, show_disabled

nid_list: ‘1,3,9-11, 100-300'

CAPMC Applets: Power Capping

- **get_power_cap_capabilities [--nids nid_list]**
  - Returns power capabilities per node-type, for requested nodes

- **get_power_cap [--nids nid_list]**
  - Returns current power cap settings, one record per node

- **set_power_cap --nids nid_list [--node watts] [--accel watts]**
  - Set power cap settings

CAPMC Applets: Power Capping

- **get_power_cap_capabilities [--nids nid_list]**
  - Returns power capabilities per node-type, for requested nodes

WLM Use Case:

- Power capping at job launch
- Dynamic power capping at application, job, or system-level
  - Adjust power cap up/or down within limits in ***get_power_cap_capabilities***
  - Respond to external site conditions or changes in workload priorities
- Scheduling for system power/cooling limitations
  - Power capping as a way to implement power as a consumable resource

CAPMC Roadmap

- **Proposed new in-band features**
  - Dynamic c-state limiting
  - Dynamic p-state limiting

Working with ACES on new in-band controls enabled by the HPC PowerAPI

- **Proposed new "Platform" controls**
  - Configuration controls for future blades and processors
  - Enable WLM to configure nodes to match job-level requirements
  - Support WLM orchestration of hardware reinitialization
    - As required to activate requested changes

Q&A

Steven J. Martin …… (stevem@cray.com)

David Rush ………….. (rushd@cray.com)

Matthew Kappel …… (mkappel@cray.com)

Additional Resources

Man Page

- capmc (8)
- http://docs.cray.com/cgi-bin/craydoc.cgi?mode=Show;q=;f=man/smwm/72/cat8/capmc.8.html

“Monitoring and managing power consumption on the Cray XC30 system”

- **Cray S-0043-72**
- **http://docs.cray.com/books/S-0043-7203/S-0043-7203.pdf**

Legal Disclaimer

Information in this document is provided in connection with Cray Inc. products. No license, express or implied, to any intellectual property rights is granted by this document.

Cray Inc. may make changes to specifications and product descriptions at any time, without notice.

All products, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.

Cray hardware and software products may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Cray uses codenames internally to identify products that are in development and not yet publically announced for release. Customers and other third parties are not authorized by Cray Inc. to use codenames in advertising, promotion or marketing and any use of Cray Inc. internal codenames is at the sole risk of the user.

Performance tests and ratings are measured using specific systems and/or components and reflect the approximate performance of Cray Inc. products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance.

The following are trademarks of Cray Inc. and are registered in the United States and other countries: CRAY and design, SONEXION, URIKA, and YARCDATA. The following are trademarks of Cray Inc.: ACE, APPRENTICE2, CHAPEL, CLUSTER CONNECT, CRAYPAT, CRAYPORT, ECOPHLEX, LIBSCI, NODEKARE, THREADSTORM. The following system family marks, and associated model number marks, are trademarks of Cray Inc.: CS, CX, XC, XE, XK, XMT, and XT. The registered trademark LINUX is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Other trademarks used in this document are the property of their respective owners.

CRAY®

THE SUPERCOMPUTER COMPANY