Exascale Power Management

With

Power API and Redfish

Prasanth Kurian, Vinanti Phadke, Anusha M S

Enterprise Group

{[anusha.m-s@hpe.com](mailto:anusha.m-s@hpe.com), prasanth.kurian@hpe.com, vinanti.phadke@hpe.com}

Abstract

*The challenge of balancing between power and performance is now well established. While research in this area is well underway, the ability to monitor and manage power and energy has been evolving. This is a behemoth challenge in the field of High Performance Computing (HPC) more so when designing an Exascale supercomputer. With every generation of hardware, more power measurement and control capabilities are made available. Software will need to make appropriate choices about how to allocate the available power budget given many, sometimes conflicting considerations. For these reasons, specifications for Power API for power management was been defined by Sandia Labs in collaboration with a consortium of hardware vendors including HPE. It is a standardized power measurement and control API for exascale systems. It enables high-frequency measurement and it includes a rich and granular level metrics gathering interface and it also targets diverse requirements from software vendors .It provides flexibility to the implementer of Power API to use wide variety of devices like RAPL, PAPI, Power Insight etc. Redfish is a DMTF standard, built on modern tool chain (HTTPS, JSON and OData), for managing multimode server via a RESTFUL interface. By integrating Power API with Redfish we strive to achieve a solution which will provide immense flexibility and portability for the implementers.*

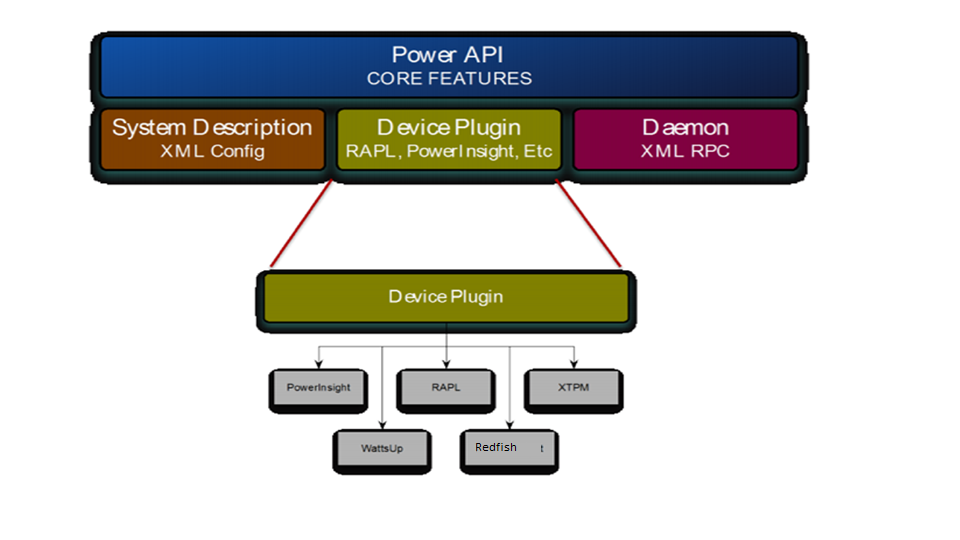
Problem statement

Though Power API currently supports a wide spectrum of devices but most of them are proprietary and not hardware agnostic. Redfish is an open standard and is vendor neutral is expected to be a mandate for all the hardware being used by large accounts like Federal/National labs in near future. Redfish can be supported by wide variety of devices starting from chipsets to servers. Power API offers both in-band and out-of-band communication with devices which reduces stealing the CPU cycles.

System wide power management in a cluster is a tedious task, however using Power API the power management of the entire cluster could be done from a single point. All the desired power configurations can be pushed by an agent to the individual Redfish compliant devices in a seamless way.

Our solution

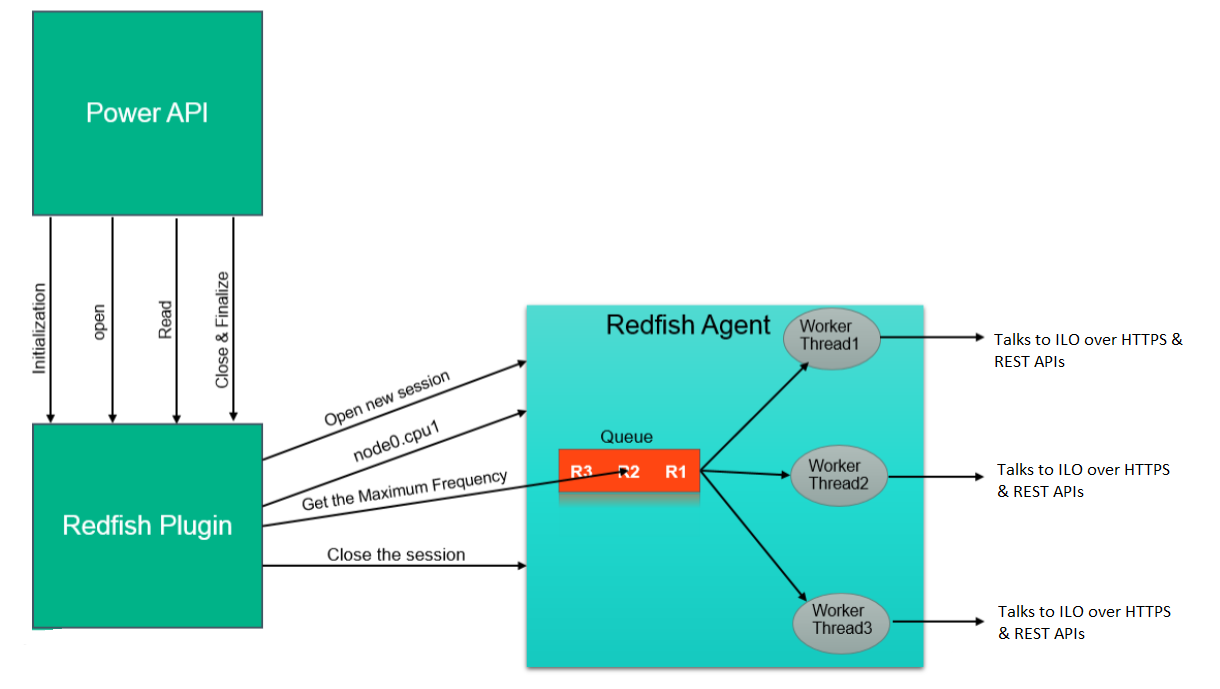
Power API core features are exposed to the various clients looking for Power Management across small or large cluster. A device plugin is developed, in current scenario Redfish plugin, which communicates between Power API and the master node. This master node talks to the underlying servers which conforms to Redfish Standard. [Refer Figure 1 for Reference Implementation of the same.]



|  |
| --- |
| *Figure-1: Reference Implementation* |
|  |
|  |

Refer Figure -2 for Schematic flow of our Implementation. In the figure we have three main entities. 1) Power API exposed to user interface and which can talk to redfish plugins 2) Redfish Plugin which can talk to underlying redfish complaint systems via Redfish agents and it can talk to multiple agents 3) Redfish Agents which is light weight daemon listening over TCP/IP . It receives node, device and attributes information from the plugin and converts the received information into REST URI’s. It can also talk to local machine via a channel interface driver (in-band) and remote machine securely over HTTPS (out-of-band).

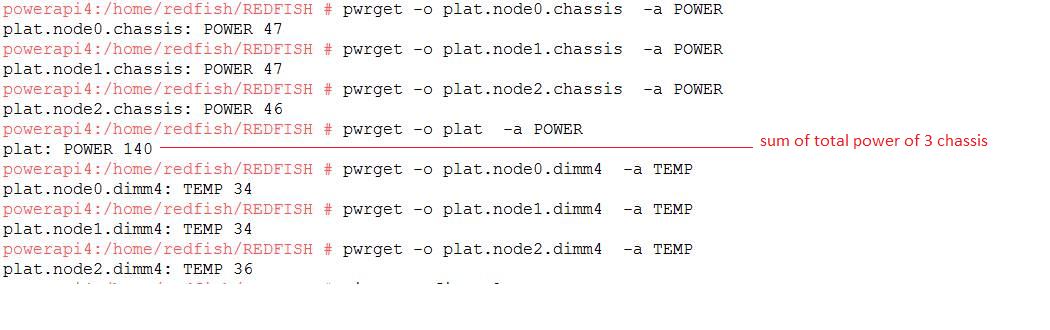
A connection is initialized with an agent and a session is created. Power API opens the connection and send the node and device information to desired agent. The plugin makes the read call for attributes like voltage, power, and temperature to an agent to read from ILO (In the case of HPE Servers) for the corresponding device. The result is sent back from an agent to plugin and finally session is closed.



|  |
| --- |
| *Figure-2:Schematic Workflow* |
|  |
|  |

Current status

The initial prototype has been developed and currently the tool can query some of the attributes related to temperature and power. Power API is also integrated with the monitoring tool called Ganglia. For further enhancements we are collaborating with Sandia Labs. Currently we are working towards setting up a Test Bed for measuring different factors like Latency and accuracy.



|  |
| --- |
| *Figure-3: Textual output* |
|  |
|  |

|  |
| --- |
| *Figure-4:Ganglia Interface* |
|  |
|  |



References

[1] <http://powerapi.sandia.gov/docs/PowerAPI_SAND_V1.4.pdf>

[2] <http://www.sandia.gov/~jhlaros/Laros_23_PowerInsight.pdf>

[3] <https://www.dmtf.org/standards/redfish>

[4] http://www.sandia.gov/~jhlaros/publications/MeasuringRealPowerUsage.pdf