

Unified Hybrid Computing Infrastructure Management Architecture

Yung-Chin Fang, Jenwei Hsieh Ph.D., Victor Mashayekhi Ph.D. , Rooholamini Reza Ph.D.
Dell Inc.

{Yung-Chin_Fang; Jenwei_Hsieh; Victor_Mashayekhi;Reza_Rooholamini }@dell.com

Stephen L Scott Ph.D.
Oak Ridge National Laboratory
scottsl@ornl.gov

Chokchai Leangsuksun Ph.D.
Louisiana Tech University
box@coes.latech.edu

Xubin He Ph.D.
Tennessee Tech University
hexb@tntech.edu

Abstract

The unified management framework[1] specifically designed for large scale heterogeneous high performance computing cluster (HPCC) provides hardware independent and highly desired hardware/software independent infrastructure management interoperability. As mainstream high end computing architecture evolved from HPCC only to high availability cluster (HAC) embedded in HPCC and formed a hybrid cluster computing architecture, virtualization technology is also emerging into this hybrid architecture for higher level of availability. The unified management framework design is focused on HPCC and does not cover HAC and virtualization specific features or behavior and can cause conflicts in a hybrid computing management environment. A standardized, unified and efficient computing infrastructure management architecture that

covers HPCC, HAC and virtualization technology is in need for the quickly emerging hybrid computing architecture to sustain cluster performance and availability. In order to satisfy this hybrid computing trend, this paper proposes a unified hybrid computing infrastructure management architecture.

Keyword: large scale, management framework, industry standard, HA cluster, HPC cluster, virtualization technology, Hybrid computing architecture.

1. HAC and HPCC Comparison

Over the past decade, high performance computing cluster (HPCC) has evolved from an experimental computing architecture to a proven and widely adopted main stream high-end computing architecture. As HPCC gradually

became the chief high-end computing architecture in production data center facilities, the need for computer system, operating system and application availability became more and more important. Many types of high availability technologies were gradually incorporated into the HPCC architecture to provide levels of availability. Both the complexity of cluster computing software stacks and the scale of computing facilities grew rapidly in the past decade. The HPCC architecture-centric interface gradually started to block hybrid computing advancement. Therefore, a unified management architecture for hybrid computing infrastructure has become an urgent need.

Condition	HPCC	HAC	Result
Substance	Large Scale	Small Scale	Dissimilar
Service Passes Health Info Around	Not Desired	Must have	Dilemma
Service Consume Resources	Discouraged	Must Have	Dilemma
Node Down	Ignorable	Critical Condition	Ambiguity
HD Down	Ignorable for Diskless Configuration	Critical Condition	Ambiguity

Table 1. HAC/HPCC behavior and status comparison table

2. HAC/HPCC Behavior and Status Comparison

Table 1 shows the differences between HA and HPCC architecture. The hybrid computing architecture introduces management dilemmas and levels of ambiguity to computing infrastructure management due to dissimilar goals and scale. For example: an HA agent is required to monitor and pass certain information such as heartbeats to other nodes at certain intervals as an “alive” signal. This behavior is undesirable in a large scale HPCC environment due to shared resource constraint consideration. Contradictory engineering concerns such as this support the need for hybrid computing infrastructure management. Agents and services designed for HAC will consume computing resources on all nodes. On the other hand, HPCC users tends to allocate all available computing resources to computing tasks to reduce task turnaround time, meaning that the computing resources consumed by HA agents are a hindrance. HAC and HPCC specific requirements lead to hybrid computing system infrastructure management system design dilemmas. A single hung node in an HA cluster can change the overall cluster status from fully operational to critical, however a large scale HPCC environment usually will have multiple hung nodes at any given moment, a perfectly normal occurrence during the HPCC life cycle. A small number of hung nodes usually will not change the overall HPCC status from healthy to critical. The differences in the determination of

a critical state lead to overall hybrid cluster state ambiguity. Hard drive crashes in an HA cluster usually lead to cluster critical conditions and will trigger a preset failover process. On the other hand, hard-drive crashes in a diskless HPCC environment may be ignorable. This conflict leads to node and component level status ambiguity. Beside the fact that HAC and HPCC technologies are merging, virtualization technology is also entering the hybrid computing architecture for even higher levels of availability and load balancing capability. A unified computing infrastructure management framework which covers mainstream and emerging computing technologies for reliability, availability, serviceability, security and scalability is essential.

3. Dependency and Implication Analysis

Existing Dependencies	Implication
Power management HW dependency (STONOTH)	HW specific (i.e. PDU, etc.) binaries
Processor architecture dependency	Instruction set specific binaries
SM HW architecture dependency (LM, IPMI)	SM HW specific binaries
OS architecture dependency (Linux/windows)	OS architecture specific binaries
OS distro dependency (Redhat/Suse/etc)	OS distro specific binaries
OS version dependency	Kernel version specific binaries

Development tool dependency (java, perl)	Development tool specific code
Development tool version dependency (lib version dependency)	Development tool version specific binaries
SM CLI compatibility	CLI specific binaries

Table 2: implementation dependencies

Besides the dilemmas and levels of ambiguity, a hybrid computing infrastructure management framework should remove existing dependencies such as hardware, instruction set, systems management hardware (management processor) architecture, operating system architecture, operating system distribution, operating system version, development tool, development tool version and system management command-line interface. These dependencies are the road block for high performance hybrid computing architecture management.

A unified interface [1] for HPC cluster management introduces the concept of a single interface to manage heterogeneous hardware/software without adding a new agent. Other than the management console, no extra memory or processor cycles will be consumed. A unified interface architecture as many additional advantages. It removes the management software component and multi-management framework dependencies, it provides interoperability for very large scale environment management, it leverages existing in-band and out-of-band management implementations, and it makes

computing infrastructure management development hardware independent. Finally, a unified interface provides hardware-independent backward and forward interoperability, compatibility, manageability and usability. The unified interface architecture prototype has been proposed to Distributed Management Task Force[2] (DMTF) and has triggered the development of System Management Architecture for Server Hardware[3] (SMASH) specification. Currently, tier-one computer system vendors such as Dell[5] and HP[6] already support SMASH CLP [4] on their hardware platforms. Since the architecture was designed to provide interoperability, reduce resources consumption overhead and solve the knotty dependencies for HPCC computing environment at the time, the evolving HA embedded in HPCC hybrid computing architecture is not covered in the unified interface design scope. Hybrid computing architecture requires both higher levels of resource utilization and availability. The existing in-band system management components introduce dependency issues from hardware components, driver components, and cluster level service components. When combining heterogeneous hardware architecture and existing software implementations to the need for backward compatibility, the design of hybrid computing management architecture became much harder than before. Thus, to maintain hybrid computing component compatibility along with technology growth became a new challenge. In order to satisfy the rising HA requirement, to provide a solid

foundation for computing infrastructure management framework, and to reduce levels of dependencies, an architecture in figure 2 is proposed.

4. Achievement & Progress

In the overall systems management development process, certain services and components are standardized and implemented into the hardware, the firmware and the operating system to unify the overall operating environment. For example: an instrumentation service is standardized and the implementation is integrated into the operating system and the power management interface is implemented on the hardware, the firmware and software. The power management software service is also integrated into the operating system, and both kernel and user level system management services such as OpenIPMI are also integrated into the operating system. Figure 1 and 2 show the before and after comparison. The standardization and integration effort largely reduced all levels of dependencies. Industry standardization efforts lead by the DMTF System Virtualization, Partitioning, and Clustering (SVPC) Working Group are working on defining cluster states, and clarifying status ambiguity and conflicts. The goal is to have a unified and clearly defined set of profiles and definitions. SVPC is chartered to create profiles to address current and future requirements in the following areas: high availability in clusters of computer systems, scalability, simplified system management,

virtualization of ComputerSystems and the resources which comprise them or on which they are dependent, partitioning of computer systems and the resources which comprise them into multiple separate computer systems, and generic services in support of high availability and other automation services.

Next step is to investigate the common service routines across HAC, HPCC and virtualization technologies and the implementation techniques used. Also, to categorize the services, study the commonality, and start an initiative to establish a service routine common ground and merge DMTF SVPC profiles into the architecture to provide synchronized infrastructure management architecture for hybrid computing.

5. Future Works

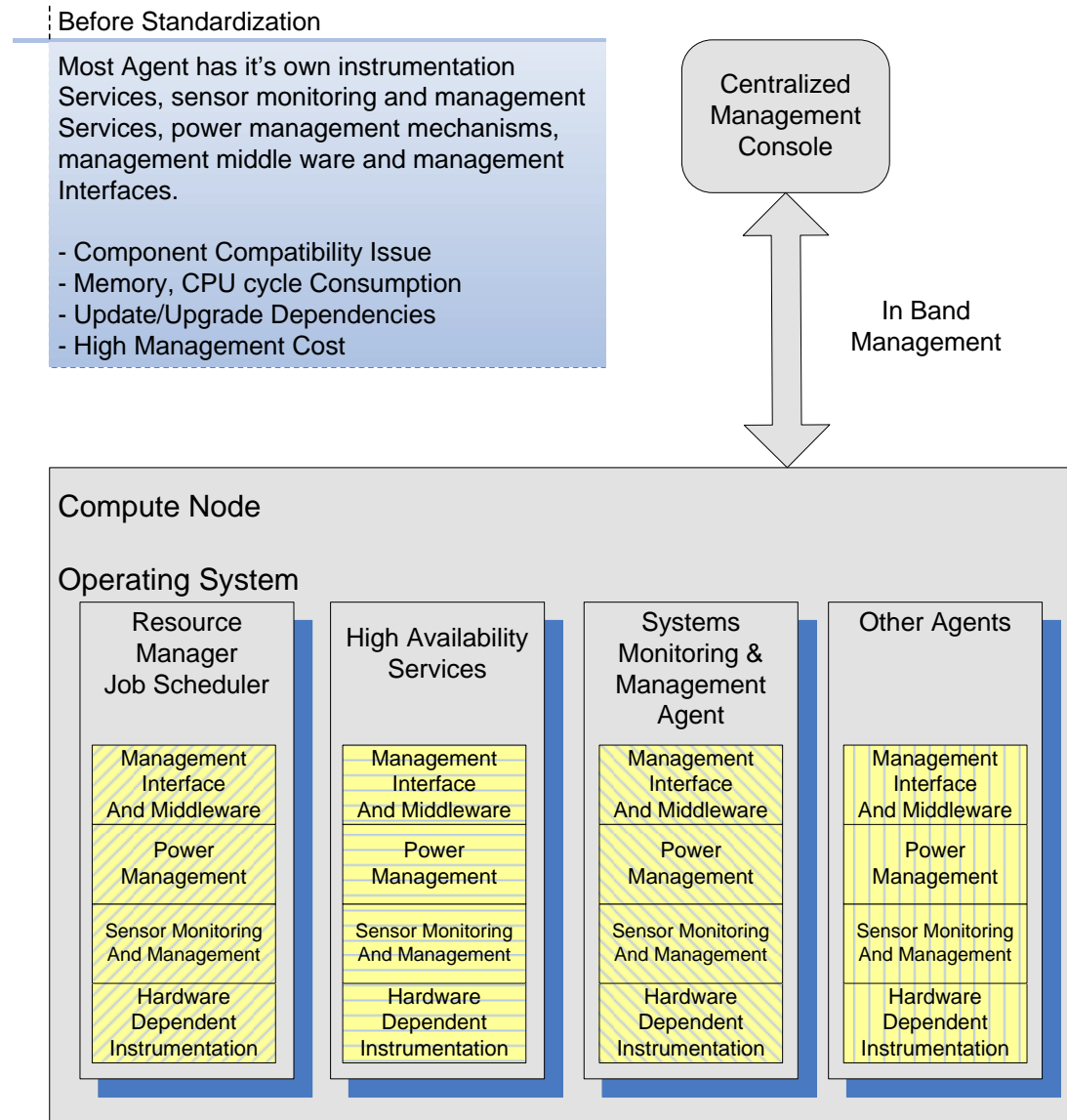


Figure1. Existing hybrid computing software stack

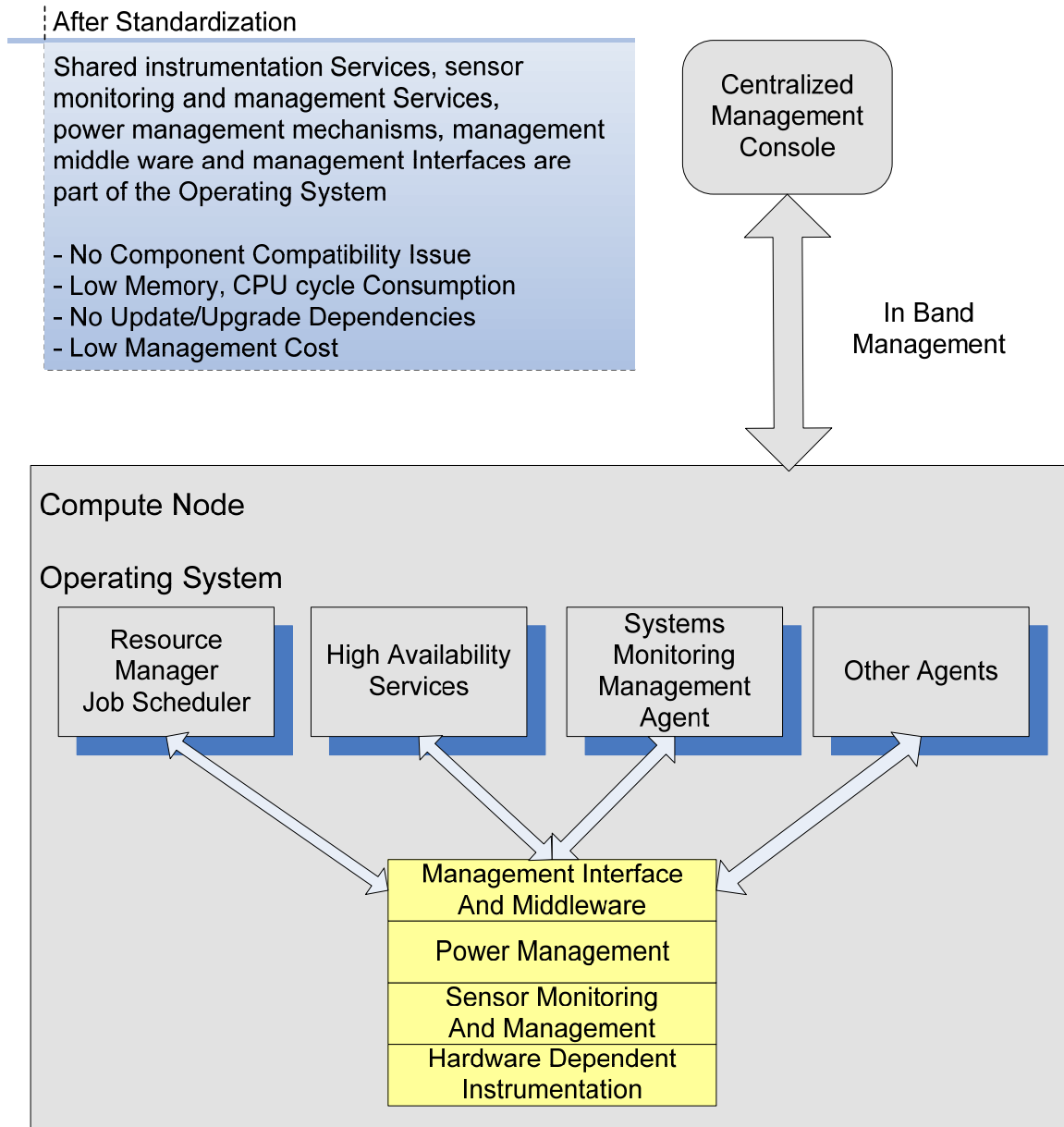


Figure 2. High dependency items are moved into OS

7. References

[1] “Unified Heterogeneous HPC Hardware Management Framework”, Yung-Chin Fang, LCI's 5th International Conference, Linux Clusters: The HPC

Revolution 2004, Austin, TX

[2] <http://www.dmtf.org/home>

[3] Systems Management Architecture for Server Hardware (SMASH) Initiative

[4] Server Management Command Line
Protocol Specification (SM CLP)

[5][http://support.dell.com/support/edocs/
software/smdrac3/drac5/OM5.1/en/ug/D
RAC5OM51.pdf](http://support.dell.com/support/edocs/software/smdrac3/drac5/OM5.1/en/ug/D
RAC5OM51.pdf)

[6][http://h18013.www1.hp.com/products
/servers/management/remotemgmt/lights
out100/index-g1.html?jumpid=reg_R100
2_USEN](http://h18013.www1.hp.com/products
/servers/management/remotemgmt/lights
out100/index-g1.html?jumpid=reg_R100
2_USEN)