Engineered for a Difference in High Performance Computing (HPC): Why IBM Power Systems Lead in Performance, Reliability, Availability, and Serviceability

Sponsored by IBM

Srini Chari, Ph.D., MBA

December, 2009

chari@cabotpartners.com

Introduction – the Mainstreaming of HPC Has Put Considerable Focus on Reliability

In today's climate, companies must innovate with flexibility and speed in response to customer demand, market opportunity, regulatory changes, and competition. High performance computing (HPC) helps companies achieve the speed, agility, insights, and sustained competitive advantage to deliver innovative products, increase revenues, and improve operational performance. Scientists, engineers, and analysts in many smart enterprises rely on HPC to solve challenging problems in engineering, manufacturing, finance, risk analysis, revenue management, and in the life and earth sciences.

With this mainstreaming of HPC, in addition to application performance, users demand systems that deliver excellent reliability, availability, and serviceability (RAS) – just as these systems are increasingly built with thousands of components (processor, memory, disk, switch, power supplies, and so on) to deliver the needed performance and functionality. This is a challenging engineering design problem as failures grow with the number of components. It's crucial for today's petascale and tomorrow's exascale systems to possess the RAS characteristics approaching today's mission-critical business systems. Merely scaling up systems designs with "commodity" components will just not cut it.

Most parallel HPC applications have multiple processes that run concurrently on many nodes with communication over high-speed interconnects. A single failure in one of these processes can cause an outage in the entire application, requiring the user to restart the application from the previously checkpointed state. This hampers a user's productivity, diminishes enterprise value, and increases the annual true cost of ownership (TCO). In fact, a 1% reduction in system availability could result in a loss of several million dollars of bottom-line benefits at enterprises that rely on large-scale production HPC environments.

What's needed, to maximize application uptime, are holistic HPC systems designs and reliability-aware software solutions[1] that are engineered for outstanding RAS while delivering the quantum increase in performance and scale needed for tomorrow's HPC applications.

Today, IBM delivers a rich portfolio of HPC solutions spanning systems, software, and services to the marketplace based on sustained technology investments to enhance functionality, performance, energy-efficiency, and RAS across its portfolio of systems and technology offerings. These investments in semiconductor processor technology and architecture, packaging, power and cooling, system software, and RAS features that are largely a part of the mission-critical System z mainframes have been systematically implemented across the IBM HPC portfolio and next generation energy-efficient data centers. Also, the General Parallel File System (GPFS), initially developed by IBM for the HPC market, is the key enabling file-system underpinning IBM's recently announced Smart Analytics cloud and large storage clouds.[2] These symbiotic technology investments differentiate IBM from its competitors in the HPC markets of today and the future as customers continue to demand performance, scale, RAS, and energy-efficiency.

Through in-depth research and interviews, this article highlights how IBM's HPC solutions, particularly the IBM Power Systems, are engineered to improve RAS for HPC.

---

[1] Raju Gottumukula, *et. al*, "Reliability-Aware Resource Allocation in HPC Systems", IEEE International Conference on Cluster Computing, 2007, pp 312-321.

[2] www.ibm.com/cloud

Outstanding RAS and Application Uptime is Key to an Enterprise's Bottom-line Benefits

The current economic downturn and the escalating energy and people costs for HPC, will force companies to reevaluate how they can maximize their return on IT investments. They will need smarter approaches to reduce costs, manage complexity, improve productivity, reduce time to market, and enable innovation. Simply put, organizations must and will carefully examine the business (value and costs) case of HPC investments. RAS and application uptime have a major impact in the business case of an HPC solution.

A cost-value framework for HPC investments: The escalating costs of building and operating data centers are not due to IT capital costs. They are primarily because of increasing energy, facilities, and other operational costs. Evaluating systems solely on IT acquisition costs and price/performance is seriously flawed. Moore's law continues to persist at the processor level and every 18 months or so, the computational performance delivered by new generation of systems continues to more than double at roughly the same price point. IT acquisition costs as a fraction of the Total Cost of Ownership (TCO) are expected to decline in the future. The TCO over several years, say 3, must be assessed in order to make objective cost decisions while evaluating various solution options. But the TCO alone is inadequate. What's needed is a framework of inter-related drivers and associated metrics that examine the total costs incurred and the value delivered by HPC solutions including the significant effects of RAS on HPC application uptime and time to results.

Value

- Business Value: e.g. customer revenues, new business models, compliance regulations, better products, increased business insight, and new breakthrough capability,
- Operational Value: e.g. faster time to results, more accurate analyses, more users supported, improved user productivity, better capacity planning,
- IT Value: e.g. improved system utilization, manageability, administration, and provisioning, scalability, *reduced downtime*, access to robust proven technology and expertise.

Costs

- Data Center Capital e.g. new servers, storage, networks, power distribution units, chillers, etc.
- Data Center Facilities e.g. land, buildings, containers, etc.
- Operational Costs: e.g. labor, energy, maintenance, software license, etc.
- Other Costs: e.g. deployment and training, *downtime*, bandwidth, etc.

Enterprises must evaluate the costs and value of outstanding RAS for HPC within this broad cost-benefit framework. They must maintain a focus on maximizing application uptime and performance – not just for one application but a collection of workloads typical in production HPC environments. This is particularly crucial for HPC applications which are often large-scale parallel applications that execute over several hours or even days, greatly increasing the probability of failure during execution.

In addition to contributing to the substantial increase in TCO, system failures often translate to a many fold loss in business, operational, and IT value. If users have to restart their HPC applications several times during execution, they would not only get frustrated with their HPC infrastructure and support (deteriorating user-experience and hampering innovation), but would also realize lower productivity (engineers may have to wait several days before restarting critical product development simulations), loss of revenues (a one day of delay in bringing out a new drug to market in the pharmaceutical industry could result in a loss of $3M/day[3]), and often be unable to complete their time and mission-critical simulation as is the case of the daily operational weather forecast[4].

[3] Swami Subramaniam, "Productivity and attrition: key challenges for biotech and pharma", Drug Discovery Today, Volume 8, Issue 12, 15 June 2003, Pages 513-515.

[4] Srini Chari, "Reliable and Accurate Weather and Climate Prediction: Why IBM® Resilient High Performance Computing (HPC) Solutions have an Overwhelming Lead", White Paper, Cabot Partners, September 2009, ftp://ftp.software.ibm.com/systems/deepcomputing/IBMinWeatherAndClimate.pdf

Like Today's Enterprise Business Systems, RAS is becoming Very Critical for HPC Systems

In the 1990s, 99% availability was considered acceptable for enterprise business computing in the industry. This is no longer the case; most businesses measure availability by the number of 9's. The following figure depicts the system availability, by function and use, ranging from 90% (one 9 for personal computing) to 99.9999% (six 9's for military and defense applications). Most HPC clusters have significantly less availability compared to today's entry business systems[5].

9's	Availability	Downtime/Year	Examples
1	90.0%	36 days, 12 hours	Personal Computers
2	99.0%	87 hours, 36 min	Entry Level Business
3	99.9%	8 hours, 45.6 min	ISPs, Mainstream Business
4	99.99%	52 min, 33.6 sec	Data Centers
5	99.999%	5 min, 15.4 sec	Banking, Medical
6	99.9999%	31.5 seconds	Military Defense

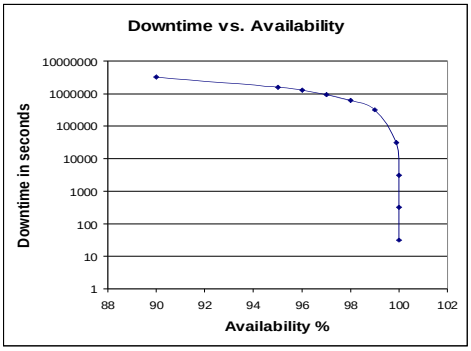

Figure 1: The downtime reduces exponentially with incremental increases in availability: (Source: Stephen L. Scott and Christian Engelmann, "Advancing Reliability, Availability, and Serviceability for High Performance Computing", Oak Ridge National Laboratory, April, 2006)

Large scale HPC systems availability with commodity components is unacceptable: Scalable parallel clusters with thousands processors are the dominant platform for HPC now and in the near future. Research on the reliability and availability of HPC clusters is very recent and ongoing[6, 7, 8, 9]. This research indicates that the existing reliability of large-scale HPC clusters was limited by a Mean Time Between Failure (MTBF) in the range of 6.5-40 hours depending upon the maturity of the installation. The most common causes of failures were processor, memory, and storage errors with hardware contributing to over 50% of the failures, software another 20%, and the remaining causes were unaccounted for. However, by extrapolation, these studies also indicate that a one petaflop system (built with today's components) would have an MTBF of approximately 1.25 hours. Clearly, this failure trajectory is unacceptable as even larger HPC systems are built and deployed. Increasingly, many production HPC applications (although there are several major bioinformatics codes that don't support checkpoint/restart yet) are designed with checkpoint and restart capability to ensure the successful completion of the long-running job despite expected failures.

Mere checkpoint and restart will not work for production HPC: HPC application strategies using only checkpoint and restart to circumvent failures will become woefully inadequate and impractical. Although, most parallel HPC applications, inherently long-running, have the capability to checkpoint and restart to recover from job failures, systems with lower availability will not only increase the true cost of ownership (TCO) but also will not satisfy the time and mission-critical requirements demanded by production HPC users. What's crucial for production HPC parallel applications is the elapsed time between submission of a parallel job and its successful completion despite unplanned outages. So it's critical to have a holistic view of reliability and availability spanning the application, system (hardware and software), and even the

[5] Stephen L. Scott and Christian Engelmann, "Advancing Reliability, Availability, and Serviceability for High Performance Computing", Oak Ridge National Laboratory, April, 2006.
[6] Raju Gottumukkala, et. al., "Reliability Analysis of HPC Clusters", http://xcr.cenit.latech.edu/hapcw2006/program/papers/hapcw_rel_analysis.pdf
[7] Arun Babu Nagarajan, et. al., "Proactive Fault Tolerance for HPC with Xen Virtualization", ICS'07, June 18-20, 2007.
[8] C. H. Hsu and W. C. Feng, "A power-aware run time system for high-performance computing", Proceedings of the 2005 ACM/IEEE conference on Supercomputing, SC'2005.
[9] I. Philp, "Software failures and the road to a petaflop machine", 1st Workshop on High Performance Computing Reliability Issues, Proceedings of the 11th International Symposium on High Performance Computing Architecture, IEEE, 2005.

administrative policies of the HPC data center with a view of minimizing failures and maximizing application uptime.

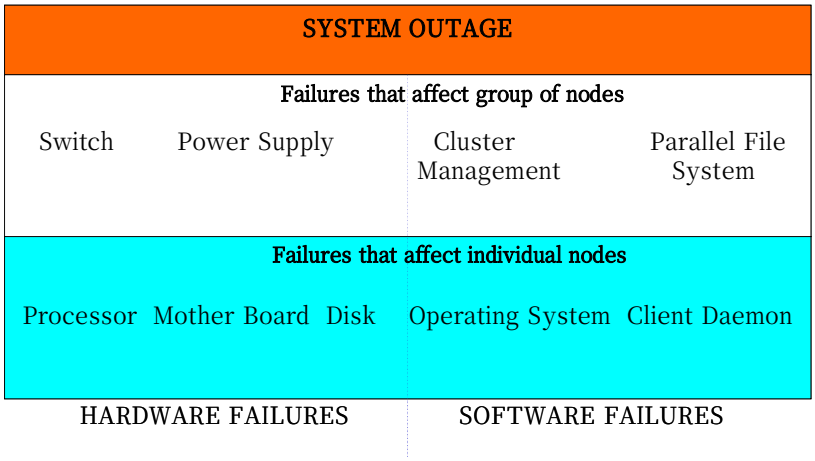

Figure 2: Failures in a HPC system (Source: Raju Gottumukkala, et. al., "Reliability Analysis of HPC Clusters", http://xcr.cenit.latech.edu/hapcw2006/program/papers/hapcw_rel_analysis.pdf

A holistic focus on RAS for HPC is needed: While *a priori* prediction and analysis of failures and reliability are just beginning to appear, *a posteriori* approaches based on log file information and end-user surveys validate that most large-scale HPC clusters have significantly less availability, between 95% to 99%[5,10] - lower than today's enterprise business computing environments.

What's required for HPC is a comprehensive approach to RAS engineering and the use of associated systems typically prevalent in today's business systems. Proactive reliability-aware systems management and resource scheduling[7, 11] solutions must be coupled with checkpoint and restart capabilities to minimize downtime and failure effects. These software approaches must be built on systems that are engineered for very high-levels of reliability and resilience typical in today's business systems. Reliable server hardware and operating system is the foundation and bedrock upon which mainstream production HPC applications must rest – just as the engine is at the core of a reliable and high performance automobile. But even a 1% difference in availability has a substantial business impact especially for production HPC centers.

A Small Decrease in Availability Translates to a Big Loss of Business Value in Productivity

Failures and downtime in HPC systems have a deleterious and substantial impact on the "time to results", TCO, and indeed overall enterprise business value. Here, we quantify some of these impacts for typical large-scale HPC environments.

How processor failures prolong "time to results": For simplicity and to illustrate these concepts, we consider only the case of a processor failure. Let us also consider a long-running (100 hours of job time – a little over 4 days) parallel HPC application running on a cluster with ***k*** homogenous processors, for example, a climate model or a crashworthiness analysis typical in the automotive industry. Let us assume that the peak performance of this large-scale system is 100 teraflops. If any of the ***k*** processors fails during execution, then the job has to be restarted from the previously stored check pointed state. This process must be continued until the job terminates normally after 100 hours of job time. If there are no failures i.e. 100% availability, the elapsed time would be equal to the job time of 100 hours. This would constitute the best case scenario. Any decrease in availability would translate to an unplanned job outage necessitating a job restart after the system has recovered.

[10] Alan Simpson, Mark Bull, and Jon Hill, "Identification and Categorization of Applications and Initial Benchmarks Suite", PRACE Consortium Partners, 2008.
[11] N. R. Gottumukula, et. al., "Reliability-Aware Resource Allocation in HPC", IEEE International Conference on Cluster Computing, 2007.

For illustrative simplicity, we further assume that the system recovery time after a failure is 1 hour. Furthermore, we assume that the Mean Time Between Failure (MTBF) is directly a function of availability, and that the failure for processor ***i*** is an independent event occurring mid-stream during execution. So, for a cluster with 99% availability, the first failure would occur exactly at 50 hours into the job execution with the next one occurring 50 hours after the system recovery from the first failure and so on. For a system with 97% availability, the first failure would occur at 25 hours into job execution, the next failure would occur 25 hours after recovery from the first failure and so on.

We further assume that the Mean Time To Recovery (MTTR) after failure for the system is 1 hour in all cases. The parallel job can restart only after the system has recovered. This job restart time ($t_{job\_restart}$) is the elapsed time between the job restart and system recovery after a failure. This job restart time could be large in many practical situations as the user may be unaware of a job failure unless the user is continuously monitoring job execution – something not practical 24 by 7. Most end-users that we interviewed indicated that for long-running jobs, it would be several hours if not the next day before they realized that their job terminated prematurely due to a system failure. After this, they would still have to restart the job from the last checkpoint and continue with job execution. The job restart time, $t_{job\_restart}$, accounts for the wasted elapsed time between a job restart and system recovery after a failure. When a failure occurs, the job computational process between the failure and the most recent checkpoint must be repeated. This adds to the wasted elapsed time. Hence, the wasted elapsed time accumulates three components: MTTR, $t_{job\_restart}$, and the rolled-back job computing time from the last checkpoint.

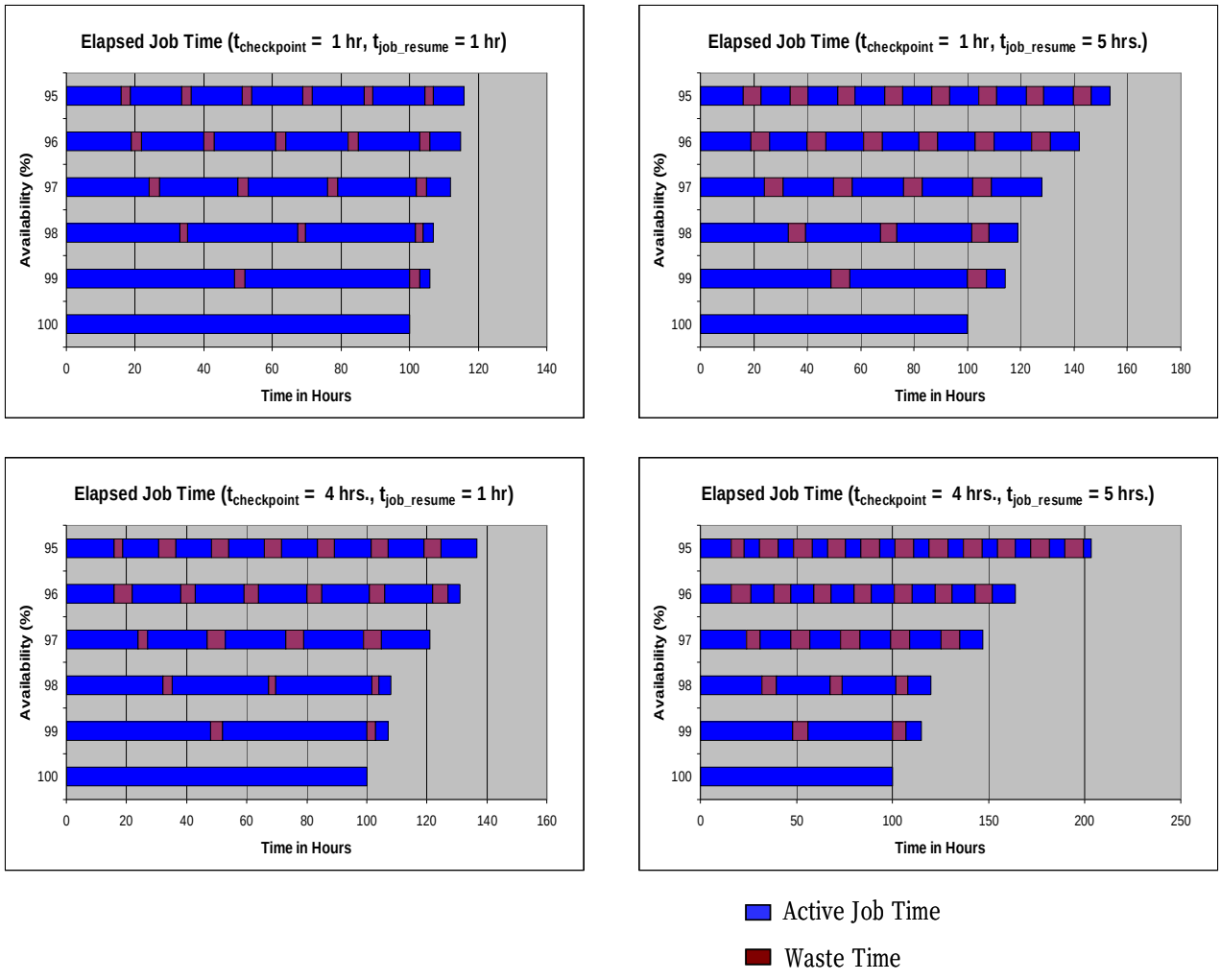

Figure 3: Effect of System Availability on the Elapsed Parallel Job Time (Longer Bars are Worse)

We study the impact of availability (95% - 100%) on the elapsed job time. The previous figure depicts four representative cases; the time for checkpoint ($t_{checkpoint}$) of 1 hour and 4 hours, and for a $t_{job\_restart}$ of 1 hour and 5 hours respectively. The wasted time around a failure is indicated in red and the productive job time is depicted in blue. The best scenario without failure (100% availability) completes in 100 hours in all cases. The other bars in each case depict increasing elapsed time for systems that have decreasing availability down to 95%. In fact, for the case with $t_{checkpoint}$ = 4 and t_{job_resume} = 5, the elapsed job time ("time to results") for a system with 95% availability is over double the best case scenario. Here the active job time is almost equal to the wasted time. This translates to a parallel job efficiency of just under 50% - clearly demonstrating the deleterious impact of reduced system availability on end-user productivity.

How reduced system availability increases TCO: Beyond the negative impact on the "time to results", reduced availability increases TCO in many ways. Here we highlight a few ways. First, the I/O job time increases considerably if a HPC parallel job is check pointed more frequently in order to reduce wasted roll-back time. So there is a practical trade-off between the time required to roll-back the computation at failure, and the I/O costs for check pointing. A smaller $t_{checkpoint}$ will reduce the roll back time but increase I/O costs, as the size of the interim data files that need to be stored could be very large for HPC. Secondly, the people and serviceability costs rise with decreasing availability and increasing failures. Lastly, reduced availability and slower "time to results" reduce the effective utilization of all HPC data center assets (hardware, facilities and people) and increase the TCO for a given workload.

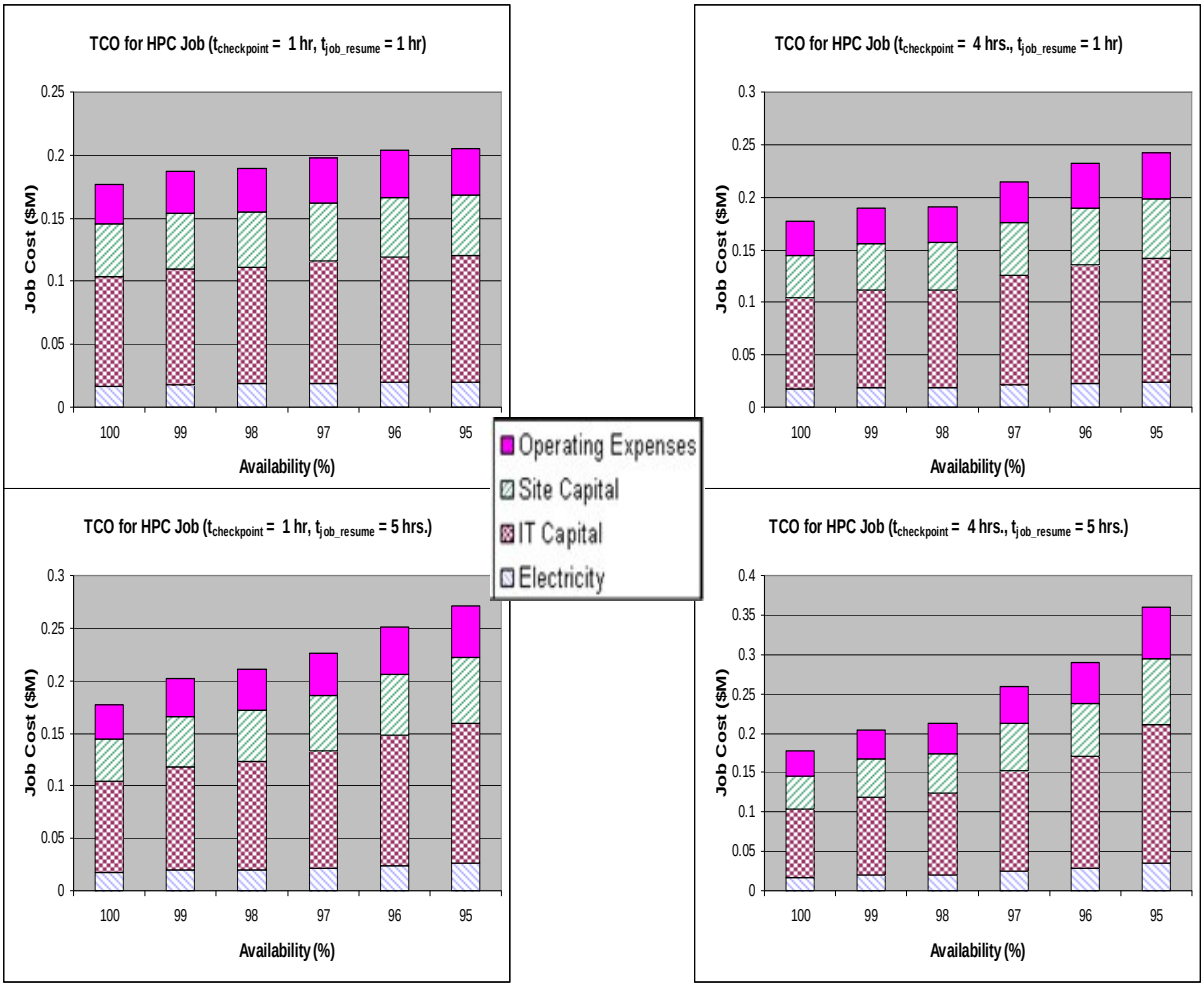

Figure 4: Effects of Availability on TCO for a 100 Teraflop System (Taller Bars are Worse)

The previous figure depicts the increases in TCO for job completion for our illustrative example on a 100 teraflop HPC system with quad-core x86 processors. As before, we consider the same four cases depicting a variation in check pointing intervals and job restart times. The TCO model[12], based on an Uptime Institute model customized for HPC, accounts for electricity, IT capital, site capital, some people-related operational expenses[13]. It does not consider other costs including direct downtime costs, software licensing, maintenance, application specific people costs, and typical administrative costs. All these additional costs would further increase the TCO for systems with lower availability.

The best scenario without failure (100% availability) translates to a per job cost of about $170K. The other bars in each case depict increasing job costs for systems that have decreasing availability down to 95%. In fact, for the case with $t_{checkpoint} = 4$ and $t_{job\_resume} = 5$, the per job costs for a system with 95% availability is over double the best case scenario. In a production setting, these parallel jobs are repeated several times before product designs or scientific recommendations are finalized. It is easy to see how these incremental job costs could easily add up to several million dollars on systems with just a small decrease in availability.

The further loss of enterprise business value could be immense: Beyond increased TCO and prolonged time to results, reduced systems availability could further diminish business value in several other ways as HPC mainstreams. First, frequent failures frustrate innovative users and researchers who not only suffer productivity losses but also become less innovative in the process; substantially impacting innovation across the business, and diluting HPC's strategic value in the enterprise. Second, prolonged time to results for every single HPC job has an accretive effect as many hundreds of related simulations must be performed during the design and development of new products or services. This will substantially increase design and development cycle time of new products or services for the enterprise, translating to revenue losses in the millions. Third, the time and mission critical nature of the simulation (as in the case of weather prediction) may make systems with lower availability unpractical. Lastly, as HPC gets increasingly deployed as a service or a cloud in a centralized environment, frequent outages could seriously harm the service provider's reputation. This has recently plagued both Amazon and Google cloud services.

Performance and reliability are crucial for HPC: As HPC mainstreams, what's required are extremely fast and exceptionally reliable systems to guarantee that operational production workloads will run to completion with minimal unplanned outages and interrupts. Winning HPC solutions must possess this combination of sustained high performance + reliability/availability + utilization. Over the last decades, IBM has continued to make substantial investments in semiconductor processor technology and architecture (www.power.org), packaging, power and cooling, system software, and RAS. This has served IBM very well in commercial IT markets. As HPC mainstreams, and as HPC users demand reliability attributes common in today's business systems, we believe IBM is very well positioned to expand its leadership in very large-scale HPC systems. The Blue Waters project[14] is just one recent example of IBM's marketplace success in this segment based on the soon to be available Power7 architecture.

The Key Elements of the IBM HPC Solutions Portfolio

IBM offers a wide array of HPC solutions through its multi-core processor systems, large storage systems, support for a broad range of operating systems, visualization, innovative applications, middleware and partner ISVs with proven expertise and deep industry presence. IBM has the leading portfolio[15] of HPC architectures, systems, and software ranging from the System x® Cluster 1350™, Blades, iDataPlex®, Power Systems®, and Blue Gene® with support for a range of operating systems including Linux®, AIX®, and Windows® together with cluster management software, a high-performance shared-disk clustered file system - General Parallel File System (GPFS™), and optimized scientific and engineering libraries. In addition, IBM has a worldwide technical staff of domain experts to help HPC customers migrate and optimize their applications on the IBM HPC portfolio to solve their largest and most challenging problems.

---

[12] Jonathan Koomey, "A Simple Model for Determining True Total Cost of Ownership for Data Centers", White Paper, The Uptime Institute, 2007.
[13] Srini Chari, "A Total Cost of Ownership Study (TCO) Comparing the IBM Blue Gene/P with Other Cluster Systems for High Performance Computing", Cabot Partners White Paper, November 2008, http://www-03.ibm.com/systems/resources/tcopaper_finalfinal_2008.pdf
[14] The Blue Waters Project, http://www.ncsa.illinois.edu/BlueWaters/
[15] The IBM Deep Computing Portfolio, http://www-03.ibm.com/systems/deepcomputing/index.html

IBM Power Systems have an overwhelming lead in performance, RAS, and utilization

We believe that over the past decade, IBM, particularly with Power-based supercomputers, has been able to deliver supercomputers, associated HPC solutions, and other complementary IT infrastructure solutions with the best mix of sustained performance, and reliability/availability, and utilization[4]. Here we demonstrate IBM's Power Systems advantage in RAS first for general enterprise computing, and then for large scale HPC gleaned from several recent RAS surveys.

AIX on Power Systems lead in server reliability for enterprise computing: Recent 2008 and 2009 results contained in the ITIC Global Server Hardware & Server OS Reliability Survey indicate that Power Systems with AIX deliver the best RAS of UNIX, Linux, Windows choices with the least amount of downtime (15 minutes per year), the fewest unscheduled outages (less than one outage per year), and the fastest patch time (11 minutes to apply a patch) [16,17]. The survey did not include mainframes, which probably would have taken the top spot. Also, not included are IBM Blue Gene systems which, in our opinion, would have also come up high especially in the context of HPC reliability. The following figure clearly demonstrates the lead in RAS that AIX on Power has over other systems including x86 systems across a broad range of business applications and installations.

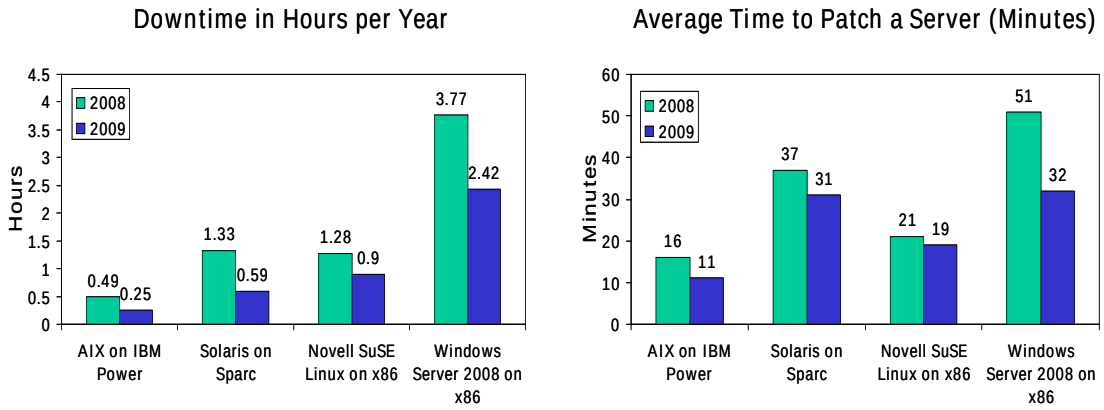

Figure 5: Downtime and Patch Time for Some Systems (Lower is Better)

Power-based supercomputers have the best mix of performance, RAS, and utilization: Large scale parallel supercomputers must meet several simultaneous yet somewhat conflicting objectives (failures and downtimes typically grow with the number of processors, leading to poorer utilization) of sustained performance, excellent RAS, and high utilization levels. So it's important to examine availability and utilization normalized with respect to performance. We examine recent survey[10] data obtained from the largest European supercomputers, which we believe to be the most comprehensive recently published study of HPC application workload performance, system utilization, and system availability. The following figures plot the Linpack[18] peak performance (RMax) of these systems versus reported availability which vary from 95% to 100% and utilization levels which vary from 20% (Galera) to 95% (Neolith).

For visual clarity, the data is depicted in two different ways: a bar chart sorted in descending order of RMax and a bubble chart. The Jugene IBM Blue Gene system is about three times the second highest performing system (Mare Nostrum – IBM Power JS21). These systems are the best systems for large scale supercomputing; possessing very high levels of absolute performance, availability for that performance, and utilization. Both systems are based on the Power architecture.

[16] "IBM Power servers most reliable in recent survey", Network World, July, 2009, http://www.networkworld.com/news/2009/071409-ibm-power-servers.html
[17] ITIC 2009 Global Server Hardware and Server OS Reliability Survey, ftp://ftp.software.ibm.com/common/ssi/sa/wh/n/pol03058usen/POL03058USEN.PDF
[18] www.top500.org

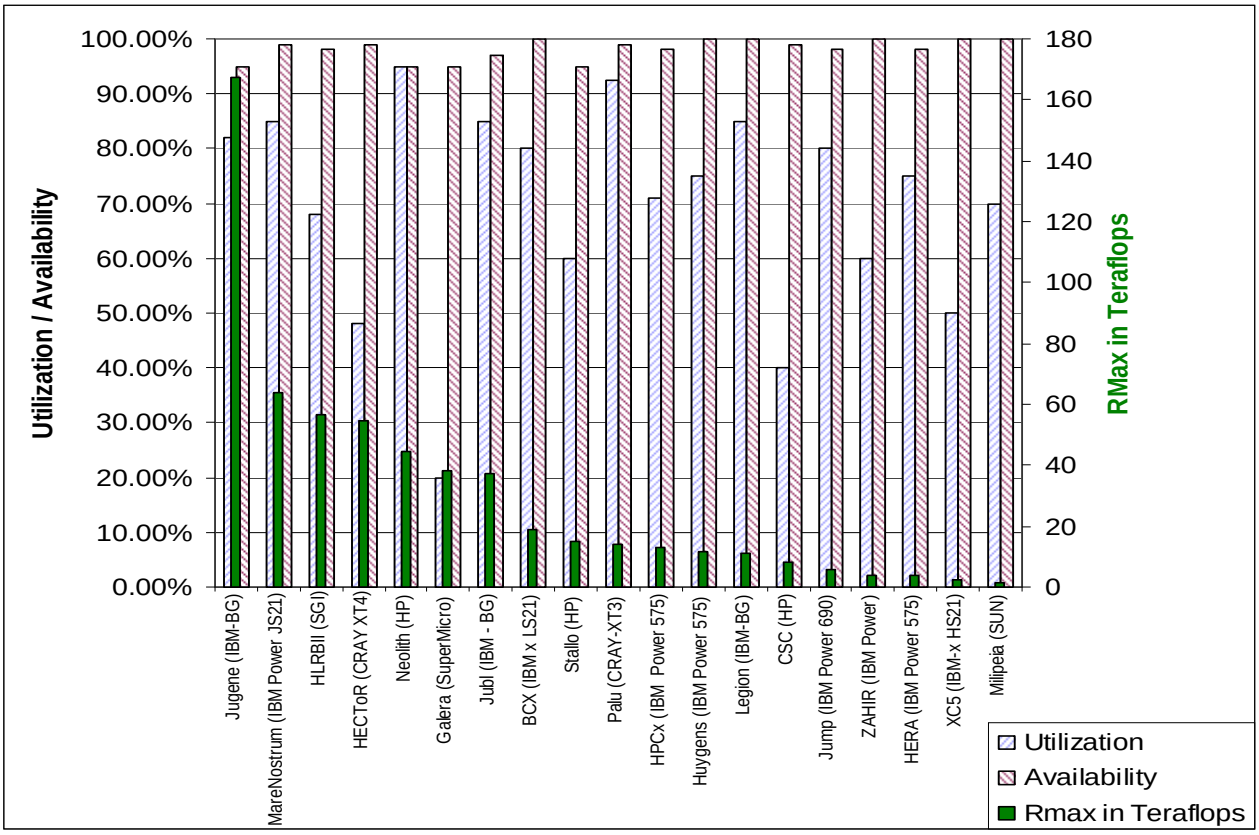

Figure 6: Linpack Performance vs. Availability for Various European Supercomputers.

In the following figure, the IBM supercomputers are colored in shades of blue with the Power based Blue Gene (darkest), Power based p575, p590, and JS21 (medium), and x86 based clusters (lightest). Likewise, HP systems are colored in shades of green, the CRAY systems are colored in purple, and other systems (SGI, SUN, and Supermicro) are in red. The IBM portfolio of supercomputers leads in this combined metric. For very large scale systems (over 100 teraflops), the IBM Blue Gene (Jugene) is the only system in this tier and exhibits good availability and utilization. Between, 50 and 100 teraflops, the IBM Power-based JS21 (MareNostrum) is the clear leader. For smaller systems below the 50 teraflops threshold, several IBM systems lead: the LS21 (BCX), p575 (Huygens), and the Blue Gene (Legion).

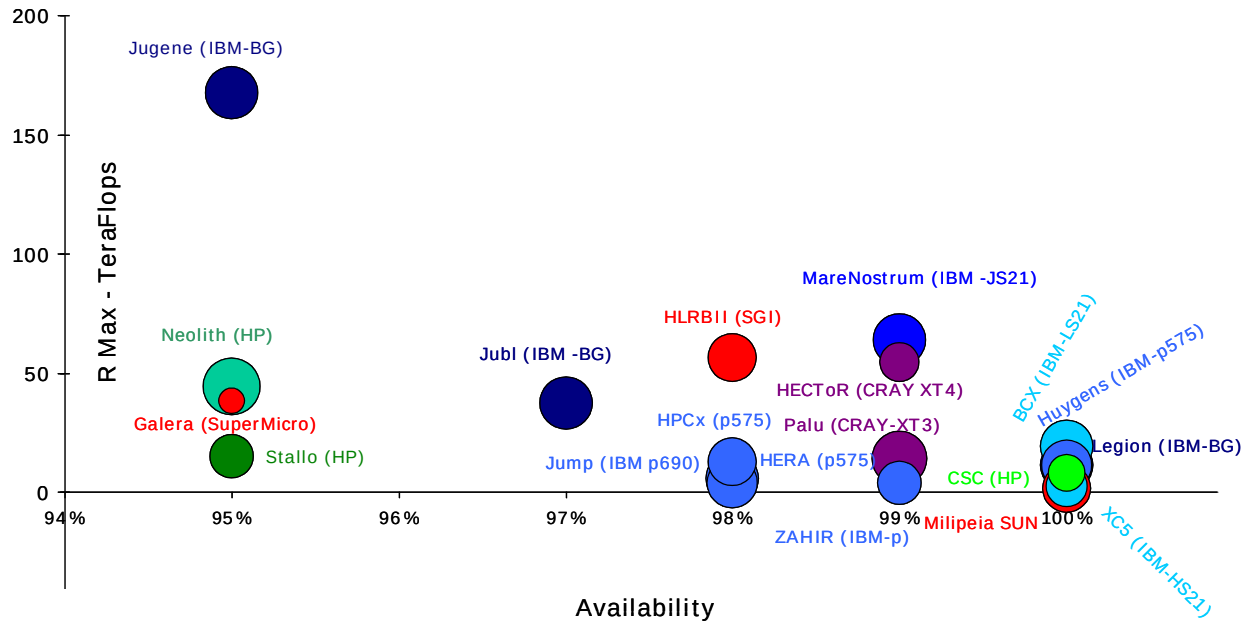

Figure 7: Linpack Performance vs. Availability for Various European Supercomputers - Larger Bubble Sizes Imply Greater System Utilization.

Furthermore, if we were to graphically extrapolate this data, it is clear that the highly energy efficient Blue Gene is on the best RAS trajectory if larger systems approaching multiple petaflops are built. This, we believe, clearly demonstrates IBM's leadership in the combined metric of performance, RAS, and utilization for large scale supercomputers. In fact, most of the top systems in each tier are IBM Power-based supercomputers – many of them with GPFS.

Engineered for a Difference: Why IBM Power Systems Lead in RAS

IBM RAS engineers are constantly making incremental improvements in hardware designs and associated software to ensure that these systems deliver the highest levels of reliability, availability, serviceability, and energy-efficiency. Here, we highlight just a few key RAS capabilities in areas ranging from servers, OS, and other software such as parallel file systems.

Highly Available Power Servers: To minimize planned and unplanned outages with a focus on maximizing application uptime, IBM's Power Systems RAS philosophy[19] is based on a holistic, well organized architectural and engineering approach to:

1. Avoid problems, where possible, through a well-engineered design with components with outstanding **reliability**,
2. Recover or retry the operation if a problem occurs to improve system **reliability** and **availability**,
3. Diagnose the problem and reconfigure the system as needed, to enhance **availability** and automate **serviceability**, and,
4. Initiate a call to repair and service automatically for quick **serviceability**.

Outstanding Reliability: Systems are packaged and built with highly reliable multi-core components. To further enhance system reliability, components that have the highest probability for failure are identified early in the server design process and the system is designed so that these components can recover from any intermittent errors or can fail over to redundant components if necessary. Automated retry for error recovery

[19] Jim Mitchell, *et. al.*, "IBM Power Platform Reliability, Availability, and Serviceability (RAS): Highly Available IBM Power Systems Servers for Business-Critical Applications", June, 2009, ftp://ftp.software.ibm.com/common/ssi/sa/wh/n/pow03003usen/POW03003USEN.PDF

of: failed instructions (through the Power6 Processor Instruction Retry), failed data transfers in the I/O subsystem, or corrupted cache data (overwriting data in a cache using correct copies stored elsewhere in the memory hierarchy), enhance systems reliability and availability. Redundancy can duplicate a function (dual I/O connections between the Central Electronics Complex (CEC) and an I/O drawer), or provide N+1 capability (increasing the speed of the remaining variable speed fans if one or more fans were to fail, maintaining adequate operational cooling until hot-plug repair can be effected). Energy-efficiency is further enhanced in the Power chip by separating circuits with high voltage operations into their own power supply rails, dramatically reducing the energy consumed by the rest of the chip. Also, processor clocks are dynamically tuned off and on as required by the execution environment, and instructions are pipelined in parallel to minimize the number of stages needed for execution. This reduces execution time or energy consumption or both while improving overall system availability.

Improving Availability: Power systems use an instrumented First Failure Data Capture (FFDC) method to detect and report fault details to a service processor to identify the root cause of failure when it first occurs. Using FFDC, the service processor has extensive knowledge and intelligence to identify and predict failure patterns, and can take proactive steps to prevent catastrophic failures that may result in overall system downtime due to unplanned outages. Self-healing actions to effect error correction, repair, or component replacement can be initiated by FFDC and the service processor, even before a system failure occurs. If a failure does occur, FFDC information can be used in the restart procedure to isolate and remove the failing component, allowing the system or other partitions to continue operation, perhaps in a degraded mode, while waiting for a scheduled repair for the failed component. This not only improves availability but also enhances serviceability.

Enhancing Serviceability: The Hardware Management Console (HMC is an option in a Power System) includes a wealth of improvements for service and support including automated install and upgrade, and concurrent maintenance and upgrade for hardware and firmware. HMC provides a focal point for service receiving, logging, tracking system errors, and if enabled, forwarding problem reports to IBM Service and Support.

Unique AIX RAS features: AIX has many unique RAS features that are still unavailable in the various Linux flavors[19]. These include run-time diagnostics and dynamic tracing of fault detection and isolation, some PCI component de-allocation of failing components, and some serviceability options. However, over time, we expect that many of these RAS features would also be supported on Linux on Power Systems[20].

The IBM General Parallel File System (GPFS): GPFS is a high performance shared-disk parallel file management solution that provides fast, reliable access to a common set of file data, online storage management, scalable access, and tightly integrated information lifecycle tools capable of managing petabytes of data and billions of files. Heterogeneous servers and storage systems can be added to and removed from a GPFS cluster while the file system remains online, simplifying management, serviceability, and enabling 7x24 operations. When storage is added or removed the data can be dynamically rebalanced to maintain optimal performance without any impact to system availability.

GPFS offers proven reliability, is installed on thousands of nodes across many HPC industries, and has become the foundation of several IBM cloud storage offerings. GPFS can be configured to eliminate single points of failure. A GPFS file can transparently failover and can be configured to automatically recover from node, storage and other infrastructure failures. GPFS provides this functionality by supporting data replication to increase availability in the event of a storage media failure, multiple paths to the data in the event of a communications or server failure, and file system activity logging, enabling consistent fast recovery after system failures. In addition, GPFS supports point-in-time snapshots and provides an online backup to protect from user errors.

[20] Satya Sharma, "Power System Software: Linux and the Power Platform", IBM Analyst Briefing, August, 2009.

Why are IBM Power Systems Well-Positioned to Provide Outstanding RAS for Future Extreme Scale HPC Environments?

We believe that IBM, with the Power Systems, is uniquely well-positioned to extend its leadership in the extreme-scale HPC market for the following reasons:

1. Today, IBM has the broadest portfolio of HPC solutions with the flexibility to offer the best mix of sustained performance, reliability, and utilization,
2. With the Power 7 and associated enhancements in the operating system (AIX and Linux), IBM is expected to significantly enhance its performance and RAS leadership on the Power 7 in the near future,
3. IBM's Programmable Easy-to-use Reliable Computing System (PERCS)[21] was selected by the Defense Advanced Research Projects Agency (DARPA) to provide systems designs and associated software that must support the eventual scaling of sustained computation to 10 petaflops. This system will be based on the Power architecture and will enable IBM to leverage this large government investment to satisfy expected future needs of extreme-scale HPC environments,
4. The Blue Waters system[14] that will be installed at the National Center for Supercomputing Applications (NCSA: One of NSF's largest supercomputer installations) will be based on the Power7 architecture,
5. The Power Hypervisor[19], GPFS, and IBM virtualization and cloud technologies could be packaged together to provide the foundation for reliability-aware software solutions needed to deliver the quantum increase in performance and scale for tomorrow's HPC applications,
6. Recently, IBM announced an exascale initiative[22] which we expect to be based on the Power architecture,
7. IBM also recently announced initiatives in cloud computing and business and predictive analytics[23] with several billion dollars of investment to build integrated (hardware, software, and services) solutions. These initiatives will require systems with outstanding RAS. GPFS and IBM Power Systems are already part of IBM's Smart Analytics[2] systems for the cloud. Many of IBM's future investments in these areas will also help the extreme-scale HPC environments of tomorrow.

[21] "PERCS", IBM Almaden Research Center, http://www.almaden.ibm.com/StorageSystems/projects/percs/
[22] http://www.ibm.com/ibm/ahead/supercomputers/index3.shtml
[23] IBM to Acquire SPSS Inc. to Provide Clients Predictive Analytics Capabilities, http://www-03.ibm.com/press/us/en/pressrelease/27936.wss