INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

CLOUD COMPUTING: AN ANALYSIS

Thakur Ramjiram Singh Lecturer Dept. of C.A., TIT Bhopal trrsingh@gmail.com

Sachin Kamley Lecturer Dept. of C.A., SATI Vidisha skamley@gmail.com

Sushil Kumar Verma Lecturer Dept. of C.A., SATI Vidisha sushilverma81@gmail.com

ABSTRACT

Cloud computing was originally designed for dealing with problems involving large amounts of data and/or compute-intensive applications. Today, however, Clouds enlarged their horizon as they are going to run both scientific and business applications supporting scientists, professionals and end users. To face those new challenges, Cloud environments must support adaptive knowledge discovery and data mining applications by offering resources, services, and decentralized data analysis methods. This paper presents an analysis that data storage, performance, data security, and efficiency can go hand in hand with cloud reliability.

KEYWORDS: *Cloud computing, economies of scale, performance, reliability, security, virtual storage.*

1 Introduction

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

A scalable distributed computing environment in which a large set of virtualized computing resources, different infrastructures, various development platforms and useful software's are delivered as a service to customers as a pay-as-you-go manner usually over the Internet is called cloud computing.

Cloud computing service have been broadly classified at three levels—as relevant to IT managers (Infrastructure as a Service, IAAS), developers (Platform as a Service, PAAS), or end users (Software as a Service, SAAS). However, scientists and business enterprises have found it difficult to choose the right cloud services for leveraging the cloud's disruptive economies of scale, elastic computational power or storage. Reference architectures for the cloud have been particularly lacking, as have specifications of the key interactions between various cloud services and deployment configurations. This is likely due to the organic emergence of the cloud. Ten years ago, the notions behind cloud conflicted with best practices of exotic and expensive hardware and software, as opposed to low-cost, commodity alternatives. But, as Google, Yahoo, and other web innovators have shown, those days are over; clouds can scale economically using commodity hardware and specialized cloud software. The time is ripe for explicit software engineering focuses when considering the cloud. While clouds are new, several best practices and specific lessons from the world of cloud Computing have a great potential for adoption. Domain Specific Software Architectures (DSSA) when applied to grids enables the articulation of service component interactions both at configuration time and at runtime. Looking into key examples from Amazon web services (IAAS), Microsoft Azure and Google App Engine (PAAS), and Gmail or Hotmail email service (SAAS) to explain the power of applying DSSA. In cloud computing service DSSA could represent as a helping hand in comparative analysis for data storage, security, and performance.

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

Scientific research is faced with the challenge of marshaling and analyzing large volumes of data from a variety of instruments and simulations, the resource capacity for which is beyond the reach of most scientists. Cloud computing has the potential to advance data intensive sciences by providing shared, pay as you go access to large scale storage and compute nodes co-located at global datacenters that offer economies of scale. While cloud computing democratizes on demand resource access for the broader e- Science community, the promise of scalability should not be at the cost of increased application complexity and rewrite overhead. Also, the ability to work with familiar desktop tools is as important to scientists as the need to off load resource heavy tasks to the cloud. Cloud computing introduces new challenges for the execution of scientific applications, which typically require specific hardware and software configurations. Currently, Virtual Machine Managers (VMMs) such as Eucalyptus or Open Nebula pave the way to manage VMs in elastic Cloud infrastructures, providing moderate support for VM contextualization (e.g. IP allocation, hot-plugging disks at boot time, etc.). However, the process of turning a VM into a Virtual Appliance (VA) that encapsulates the entire hardware and software configuration for an application to run successfully is still fairly manual. Scientific applications typically rely on numerical libraries, databases, web services, system packages, etc. whose installation and configuration can sometimes be automatically performed. Therefore, we envision semi-automatic procedures to contextualize VMs for scientific applications, where users and/or developers would provide a declarative description of their applications and its dependences . Then, contextualization software would resolve the dependences and provide the appropriate environment to install each requisite. However, installing complex software might not be automated so easily. This would lead to a set of Pre-Contextualized VMs (PCVMs) where the user (or the VM provider) would have installed this kind of software. These PCVMs could be stored in VM providers (such as Amazon S3) and a VM Repository Service would store metadata information about each VM.

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849
http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

This would allow PCVMs sharing among different scientists, thus enhancing collaboration. These PCVMs would certainly be finally configured at boot time to produce a self-contained VA that allows the execution of the application. This paper describes how various parameters are useful for efficient data storage, retrieval and performance. The conditions are same, as now the future being uncertain we must start practicing new standards and protocols to sustain.

2 Related work

In a future where data will not only come from classical computing systems as it does today, but also from millions of sensors and mobile devices, the need for energy-efficient large-scale data computation will explode**.** While Cloud computing APIs, mechanisms and abstractions are rapidly maturing, perhaps the most significant open issue is performance -- the concern is not necessarily whether we can write my application to run in the cloud, but rather how the application will perform if we do so, and it is extremely challenging to determine the lowest-cost, scalable software architecture that is able to predictably meet the performance requirements .The success of next-generation cloud computing infrastructures will depend on how effectively these infrastructures will be able to instantiate and dynamically maintain computing platforms, constructed out of cloud resources and services that meet arbitrarily varying resource and service requirements of cloud consumer applications. Typically, these applications will be characterized by Quality of Service (QoS) requirements, such as timeliness, scalability, high availability, trust, security, specified in so-called Service Level Agreements (SLAs); i.e., legally binding contracts that state the QoS guarantees an execution environment, such as a cloud based computing platform, has to provide its hosted applications with. There should be a middleware architecture that enables SLA-driven dynamic configuration, management and optimization of cloud resources and services, in order to respond effectively to the QoS requirements of the cloud customer applications. Cloud computing is taking shape, shifting the

ISSN (ONLINE) : 2230-8849
http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

focus from pure computing typical of the grid toward data distribution. The focus shift imposes several constraints in the management of computational resources assigned to a cloud. If a node of a grid is hopefully used during computations, state and applications for the cloud may have more dynamic behavior with respect to computational resources. The ability of manage efficiently the farm running cloud services is important both for granting high availability and power efficiency. Cooling and power consumption have become a major issue and current infrastructure for managing clusters is still focused on distributing computations rather than packing computations and managing nodes. A fundamental building block for any cloud network is an efficient and reliable communications infrastructure, e.g., a publish/subscribe service. However, the organization of a cloud can be very complex and may consist of arbitrary numbers and sizes of smaller microclouds and services running on many different physical networks. The capacity and costs of data transfer within and beyond an individual micro cloud network could differ by several orders of magnitude. Besides that, the whole cloud system is usually powered by ever-changing and often erratic user behavior, which makes it difficult, if not impossible, to predict any upcoming usage patterns and trends. Failing to account for such heterogeneous nature of the clouds could render the existing publish/subscribe methods inefficient or even inoperable? Developers have a wide range of platforms to choose from in creating cloud-based applications.

Google App Engine

Google App Engine is in a preview release. It is more oriented to testing out the concept and the tools than building mission critical applications. Google is currently giving developers sample accounts with 500 megabytes of storage, 200 megacycles of CPU per day, and 10 gigabytes of bandwidth per day. This should allow most applications to serve about 5 million page views per

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849
http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

month. In the future, Google plans to keep the basic accounts for free, while charging for additional resources. Google's offering is similar to Amazon, but it does not provide a set of standalone services like Amazon's S3 for storage, EC2 for hosting. The Google offering bundles everything into one package. One of the downsides of the Google App engine is that developers are limited to Python, although Google plans to add other programming languages in the future .

BigTable, Amazon's EC2 and Etelos

A key component of the Google App Engine is BigTable, which has some key distinctions from a traditional database. It is fast and extremely large scale, which enabled by a sparse distributed multi-dimensional map, rather than traditional database rows and columns. Google App Engine has a SQL-like syntax called "GQL". Select statements in GQL can be performed on one table only. GQL intentionally does not support the "Join" statement. **Amazon's EC2** is a commercial service that allows companies to rent computers to run their own computer applications. Customers rent out virtual machines (VMs) through a web services interface. These can be launched and terminated on demand. The platform uses Xen virtualization of one of three sizes ranging from 1.7 gigabytes to 7.5 gigabytes of memory and 850 gigabytes of storage. **Etelos**: Etelos provides a cloud computing platform for building and distributing apps built in PHP, JSP, Python and other languages.

3 Technical Approaches: Vulnerabilities and Threats with the clouds

Cloud computing introduces new security threats and vulnerabilities that are not present in traditional IT environments. Current IaaS technologies lack adequate security mechanisms to handle these new threats and risks, potentially exposing information stored in the cloud to the

ISSN (ONLINE) : 2230-8849
http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

service providers, attackers with Internet access, and all the other users of the cloud . For the user to have full trust in the Private Virtual Infrastructure and individual components of the system, he must implicitly trust two components: Trusted Platform Module and data factories. Without trust in these components, he cannot have assurance in the security of his information in the cloud. It is reasonable to expect the information owner can trust these components as they are trustworthy devices. A TPM is an explicitly trusted component of a computer system . A TPM is implemented as an integrated circuit that is physically attached to a platform's motherboard. The TPM has well-defined commands that allow software running on the platform to control it. Because the TPM is implemented in hardware and presents a carefully designed interface, it is resistant to software attacks. An Endorsement Key certificate allows anyone to validate that a transaction signed by a TPM is genuine.

4 Implementation

Cloud Datacenter: A cloud datacenter is a network of virtual servers that allows a company to move all of its corporate data assets into the cloud. The only IT the company maintains on-site is data terminals, laptops, netbooks, or mobile computing devices for their employees to access the cloud datacenter.

Trust in the Cloud: Trust in cloud computing is more complex than in a traditional IT scenario where the information owner owns his own computers . The information owner has an inferred trust in the platform from a social trust relationship with the service providers.

Social Trust: Social trust is established based upon reputations, previous interactions, and contractual obligations. There are two critical social trust relationships that must be established

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

in cloud computing from the perspective of the information owner: service provider trust and cloud user trust. Social trust cannot be measured, but is important to build confidence that an entity is holding up its end of a contract. Service provider trust lies in the relationship between customer and vendor . If the provider has a good reputation, then there is sufficient reason for customers to trust the provider. A vendor that has questionable service or ethics would not be as trustworthy as a vendor with excellent service and ethics.

Technical Trust: In a cloud computing environment, multiple entities must trust the cloud services; the user of the cloud service or information owner, the provider of the cloud service, and third parties. A third party is an outside entity that is providing service to or receiving services from either the user or service provider. The cloud trust model is based on transitive trust, which is the notion that if entity A trusts entity B and entity B trusts entity C, then entity A trusts entity C. This property allows a chain of trust to be built from a single root of trust. There two basic sources of trustworthiness in a cloud: information owner trust and hosting platform trust. By combing these two sources of trust, virtual environment trust can be established.

Cryptographic Primitive Trust: We assume that standard cryptographic protocols and algorithms are in place and not compromised. Protocols such as Secure Socket Layer and Transport Layer Security are required to provide secure links and transmission of data between the service provider and client. Both symmetric and asymmetric encryption primitives are required to protect data confidentiality and must be able to decrypt cipher text when required. We also assume cryptographic hash algorithms such as SHA-1 provide sufficient resistance to collisions that weaken the cryptographic strength of the hash.

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

Authenticated Public Key Channel: Trusted Platform Module public key certificates are published in a location that is accessible by users of the cloud. The public key certificates allow users to validate the genuineness of any TPM with which they transact . The authenticated public key certificate channel prevents TPM spoofing by providers ensuring that all transactions with second party TPM are confidential.

Service Provider Trust: A cloud service provider is trusted to provide a certified hypervisor and services that enable the client to verify the configuration of the provider's platform . The service provider should publish the configuration of their platforms and provide attestation signatures that clients can use to verify the configuration of the host platforms. We assume that the service provider does not lie about their configuration or there are other mechanisms to verify the configuration of the host platforms.

Known Vulnerabilities: The largest vulnerability in cloud computing is that data are processed on a machine that is owned by an entity different from the information owner. In the case of IaaS, the virtual machine is owned by the information owner and the host platform is owned by the service provider. Since the information is stored and processed on the service provider's machine, the information owner does not have full control of the data. Utility cloud computing is, multi-tenancy with other users also using the same hardware resources, which introduces the risk of exposing sensitive information to unauthorized users. Information owners do not control the hardware resources used to operate on the information and must rely on the virtualization to provide the security needed to protect their information. The shared components that make up the underlying IaaS fabric (*e.g.,* CPU, caches, storage, etc.) were typically not designed to offer strong isolation for multi-tenancy. A hypervisor addresses this gap by mediating access between guest operating systems and the physical computing resources; however, hypervisors

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

have exhibited flaws that have enabled attackers to gain inappropriate levels of control of the platform.

	Hybrid Cloud	SaaS	Remote Desktop	Client / Server	Mainframe
	2010	2000	1995	1980	1970
Scalability					
Millions of users	✓	✓	-	-	-
Central deployment	✓	✓	✓	-	✓
Multi client access	✓	✓	-	-	-
Applications scale automatically	✓	-	-	-	-
Operational Cost					
Required amount of servers	LOW	HIGH	HIGH	LOW	HIGH
Network bandwidth usage	LOW	MEDIUM	HIGH	LOW	LOW
Network latency	LOW	HIGH	HIGH	MEDIUM	LOW

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

User Experience					
Rich UI Applications	✓	-	✓	✓	-
Drag and Drop	✓	-	✓	✓	-
Fast user response time	✓	-	-	✓	✓
Applications that work together	✓	-	✓	✓	-
Application Development					
Rapid application development	✓	-	-	-	-
Visual IDE	✓	-	✓	✓	-
XML Web Service Orchestration	✓	-	-	-	-
Built in database / repository	✓	-	✓	✓	-
Multiple application support	✓	-	✓	✓	-
Native collaboration	✓	-	-	-	-
Single sign-on / keying	✓	-	-	-	-
Scalable grid platform	✓	✓	-	-	-

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849
http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

Comprehensive component library	✓	✓	-	-	-

5 . Discussions and Findings

The security aspect can be strengthened by understanding the risks associated with the clouds. A security perspective can help us to estimate the risk thereby beefing security and other points in mind. Clouds can store vast data, the mechanism of retrieval, storage, and security concerns should be given paramount importance . By advancement in technology and research, cloud can provide a better platform, infrastructure, quality of service and on demand services.

6. Conclusion

Data storage, retrieval and security will be the main concern in future, clouds can provide a better platform and ease of access .new technological arena will create much simplifications and data can be treated as a service, from the paper it can be better understood that concerns if arise then solutions will also be automatically generated.

7 References

[1] J. M. Brandt, B. J. Debusschere, A. C. Gentile, J. R. Mayo, P. P. P´ebay, D. Thompson, and M. H. Wong. OVIS-2: A robust distributed architecture for scalable RAS. In IEEE International Parallel and Distributed Processing Symposium (IPDPS): Workshop on System Management Techniques, Processes, and Services (SMTPS), 2008. URL https://ovis.ca.sandia. gov/mediawiki/images/6/60/Ovis-ipdps08.pdf.

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

---

[2] S. Chakravorty, C. L. Mendes, and L. V. Kal´e. Proactive fault tolerance in MPI applications via task migration. In Lecture Notes in Computer Science: International Conference on High Performance Computing (HiPC), volume 4297, pages 485–496, 2006. URL http://www.springerlink.com/content/ 9q840u6310467255/.

[3] K. Charoenpornwattana, C. B. Leangsuksun, A. Tikotekar, G. R. Vall´ee, and S. L. Scott. A neural networks approach for intelligent fault prediction in HPC environments. In High Availability and Performance Workshop (HAPCW), in conjunction with the High-Performance Computer Science Week (HPCSW), 2008. URL http://xcr.cenit.latech.edu/ hapcw2008/program/papers/101.pdf.

[4] Cluster Resources, Inc., Salt Lake City, UT, USA. TORQUE Resource Manager documentation, 2009. URL http://www.clusterresources.com/torque.

[5] Cray Inc., Seattle, WA, USA. Cray XD1 computing platform documentation, 2007. URL http://www. cray.com/products/legacy.html.

[6] J. T. Daly. A higher order estimate of the optimum checkpoint interval for restart dumps. Future Generation Computing Systems (FGCS), 22(3):303–312, 2006.

[7] J. T. Daly. ADTSC nuclear weapons highlights: Facilitating high-throughput ASC calculations.Technical Report LALP-07-041, Los Alamos National Laboratory, 2007. URL http://www.lanl.gov/orgs/adtsc/publications/ nw highlights 2007/ch13/13 2daly facilitating.pdf.

[8] C. Du and X.-H. Sun. MPI-Mitten: Enabling migration technology in MPI. In IEEE International Symposium on Cluster Computing and the Grid (CCGrid), pages 11–18, 2006. URL http://ieeexplore. ieee.org/xpls/abs all.jsp?arnumber=1630790.

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

---

[9] E. N. M. Elnozahy and J. S. Plank. Checkpointing for peta-scale systems: A look into the future of practical rollback-recovery. IEEE Transactions on Dependable and Secure Computing (TDSC), 1(2):97–108, 2004.

[10] E. N. M. Elnozahy, R. Bianchini, T. El-Ghazawi, A. Fox, F. Godfrey, A. Hoisie, K. McKinley, R. Melhem, J. S. Plank, P. Ranganathan, and J. Simons. System resilience at extreme scale. Technical report, Defense Advanced Research Project Agency (DARPA), 2008. URL http://institutes.lanl.gov/resilience/docs/Toward%20Exascale%20Resilience.pdf.

[11] C. Engelmann, G. R. Vall´ee, T. Naughton, and S. L.Scott. Proactive fault tolerance using preemptive migration. In Euromicro International Conference on Parallel, Distributed, and network-based Processing (PDP), pages 252–257, 2009.

[12] S. Fu and C.-Z. Xu. Exploring event correlation for failure prediction in coalitions of clusters. In IEEE/ACM International Conference on High Performance Computing, Networking, Storage and Analysis (SC), pages 1–12, 2007.

[13] R. Gioiosa, J. C. Sancho, S. Jiang, and F. Petrini. Transparent, incremental checkpointing at kernel level: A foundation for fault tolerance for parallel computers. In IEEE/ACM International Conference on High Performance Computing and Networking (SC), page 9, 2005. URL http://hpc.pnl.gov/people/ fabrizio/papers/sc05.pdf.

[14] Hewlett-Packard Development Company, L.P., Palo Alto, CA, USA. Managing Serviceguard – Fifteenth edition, 2007. URL http://docs.hp.com/en/ -90122/B3936-90122.pdf.

[15] A. Litvinova. Reliability availability and serviceability framework engine prototype. Master's thesis, Department of Computer Science, University of Reading, UK, 2009.

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

---

[16] M. L. Massie, B. N. Chun, and D. E. Culler. The Ganglia distributed monitoring system: Design, implementation, and experience. Parallel Computing, 30(7):817–840, 2004.

[17] H. Meuer, E. Strohmaier, J. J. Dongarra, and H. Simon. Top 500 list of supercomputer sites, 2009. URL http://www.top500.org.

[18] D. L. Mills. The Network Time Protocol (NTP) distribution, 2009. URL http://www.eecis.udel.edu/□mills/ ntp/html/index.html.

[19] A. B. Nagarajan, F. Mueller, C. Engelmann, and S. L. Scott. Proactive fault tolerance for HPC with Xen virtualization. In ACM International Conference on Supercomputing (ICS), pages 23–32, 2007. URL http://www.csm.ornl.gov/□engelman/ publications/nagarajan07proactive.pdf.

[20] National Energy Research Scientific Computing Center (NERSC), Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA, USA. Current and past HPC system availability statistics, 2009. URL http: //www.nersc.gov/nusers/status/AvailStats.

[21] S. L. Scott, C. Engelmann, G. R. Vall´ee, T. Naughton, A. Tikotekar, G. Ostrouchov, C. B. Leangsuksun, N. Naksinehaboon, R. Nassar, M. Paun, F. Mueller, C. Wang, A. B. Nagarajan, and J. Varma. A tunable holistic resiliency approach for high-performance computing systems. Poster at the 14th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP) 2009, Raleigh, NC, USA, 2009. URL http://www.csm.ornl.gov/□engelman/ publications/scott09tunable.pdf.

[22] M. Seager. Operational machines: ASCI White. Talk at the 7th Workshop on Distributed Supercomputing (SOS) 2003, 2003. URL http://www.cs.sandia.gov/ SOS7/presentations/seager white.ppt.

INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING AND BUSINESS SYSTEMS

ISSN (ONLINE) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

---

[23] J. Stearley and A. J. Oliner. Bad words: Finding faultsin Spirit's syslogs. In IEEE International Symposium on Cluster Computing and the Grid (CCGrid): Workshop on Resiliency in High Performance Computing (Resilience), 2008. URL http://xcr.cenit.latech.edu/ resilience2008/program/resilience08-3.pdf.

[24] X.-H. Sun, Z. Lan, Y. Li, H. Jin, and Z. Zheng. Towards a fault-aware computing environment. In High Availability and Performance Workshop (HAPCW), in conjunction with the High-Performance Computer Science Week (HPCSW), 2008. URL http://xcr.cenit. latech.edu/hapcw2008/program/papers/104.pdf.

[25] A. Tikotekar, G. Vall´ee, T. Naughton, S. L. Scott, and C. Leangsuksun. Evaluation of fault-tolerant policies using simulation. In IEEE International Conference on Cluster Computing (Cluster), 2007.

[26] K. Uhlemann, C. Engelmann, and S. L. Scott. JOSHUA: Symmetric active/active replication for highly available HPC job and resource management. In IEEE International Conference on Cluster Computing (Cluster), 2006. URL http://www.csm.ornl. gov/□engelman/publications/uhlemann06joshua.pdf.

[27] G. R. Vall´ee, K. Charoenpornwattana, C. Engelmann, A. Tikotekar, C. B. Leangsuksun, T. Naughton, and S. L. Scott. A framework for proactive fault tolerance. In International Conference on Availability, Reliability and Security (ARES), pages 659–664, 2007. URL http://www.csm.ornl.gov/□engelman/ publications/vallee08framework.pdf.