

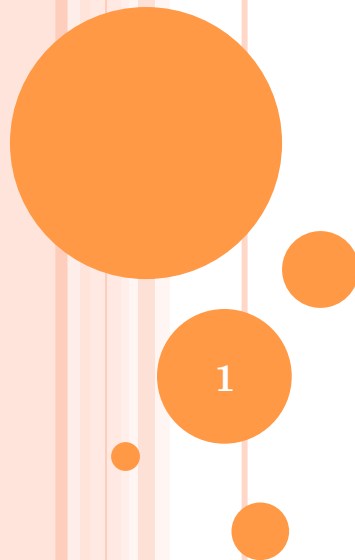


RELIABILITY IN CLOUD COMPUTING SYSTEMS: SESSION 1

Dr. Bahman Javadi

School of Computing, Engineering and Mathematics
Western Sydney University, Australia

WESTERN SYDNEY
UNIVERSITY





TUTORIAL AGENDA

- Session 1: Reliability in Cloud Computing
 - 11:00-12:30
- Session 2: Failure Trace Archive (FTA)
 - 14:00-15:30
 - Visit FTA website and download the following:
 - <http://fta.scem.uws.edu.au>
 - Grid' 5000 traces
 - LANL traces
 - Matlab scripts
 - Requirement: Matlab with Statistical Toolbox



OUTLINE: SESSION 1

- Introduction and Background
- Resource and Service Failures
- Cloud Reliability
- Case Study: Failure-aware Hybrid Cloud Architecture
- Results
- Conclusions
- Issue, Challenges and Open Questions



DEFINING RELIABILITY

- Reliability means different things to different people.
 - A researcher defines reliability as accurate web site content. The more up-to-date and impartial the information, the more reliable it is.
- The term **reliability** refers to the ability of a computer-related hardware or software component to consistently perform according to its specifications

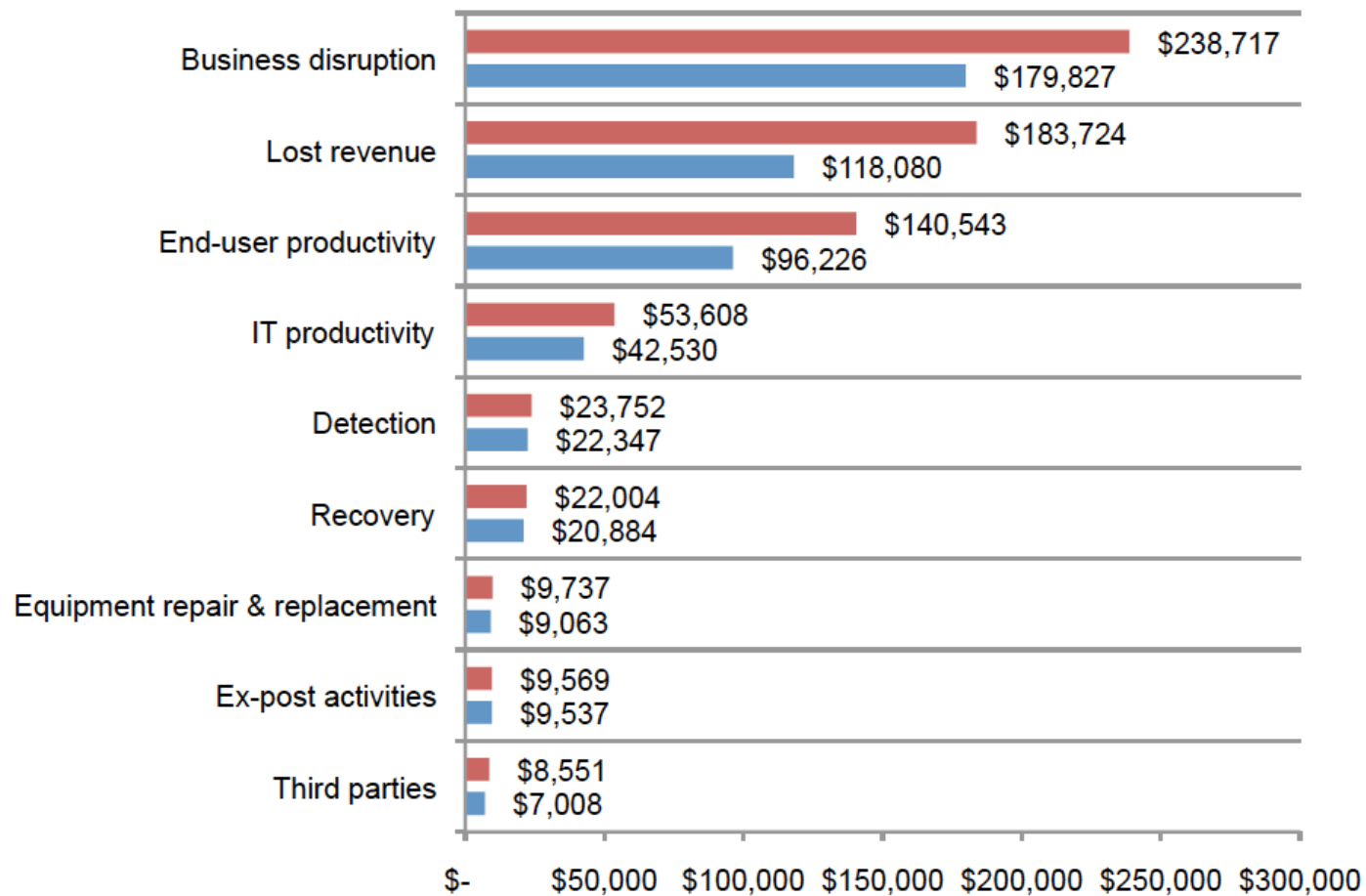


RELIABILITY IN CLOUD COMPUTING

- Reliability is a critical challenge in Cloud Computing environments.
- Reliability have huge impact on service providers
 - Business Disruption
 - Lost Revenues
 - Customer Productivity Loss



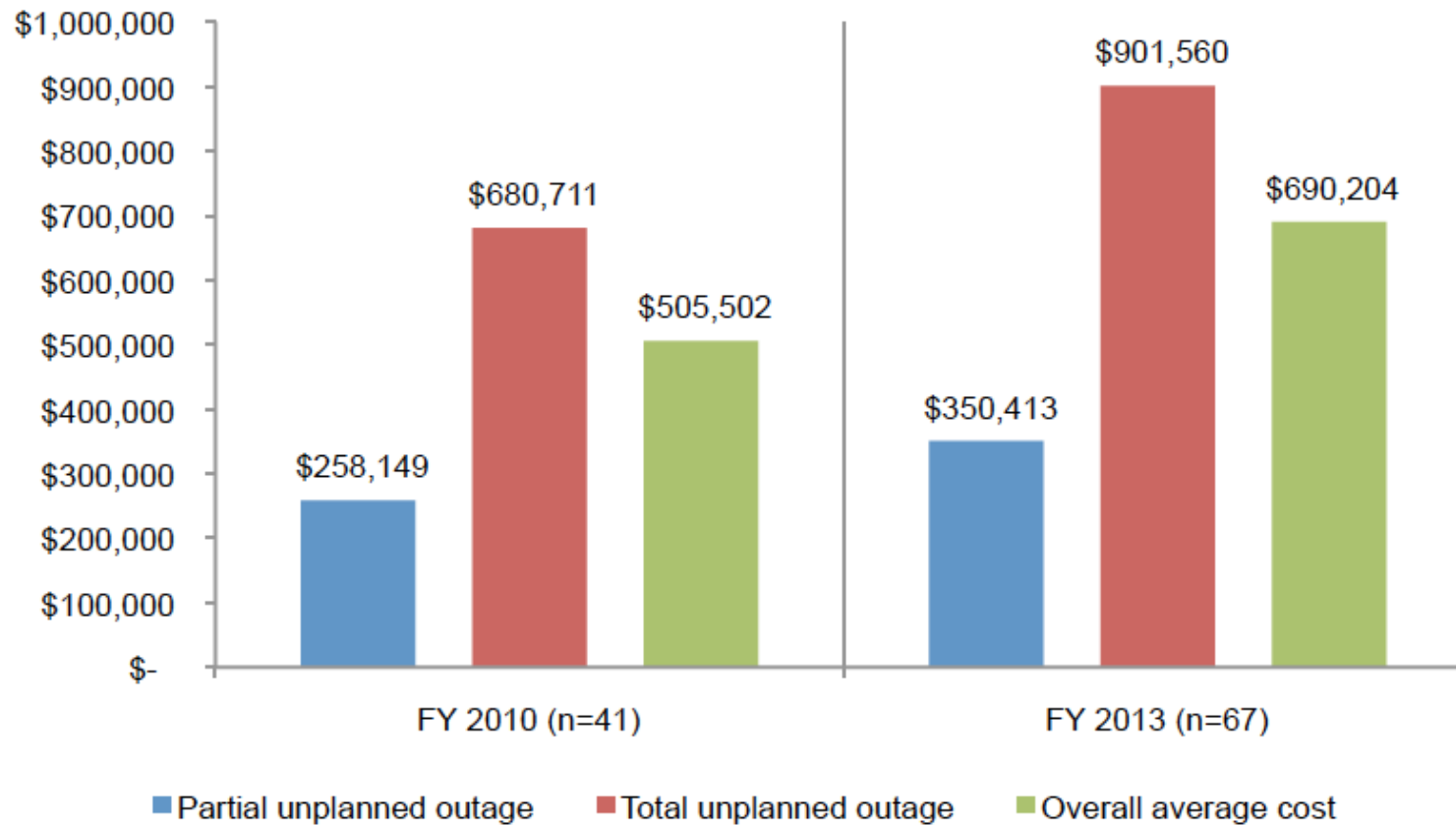
IMPACT OF CLOUD OUTAGE



* Ref: Calculating the Cost of Data Center Outages, Ponemon Institute© Research Report



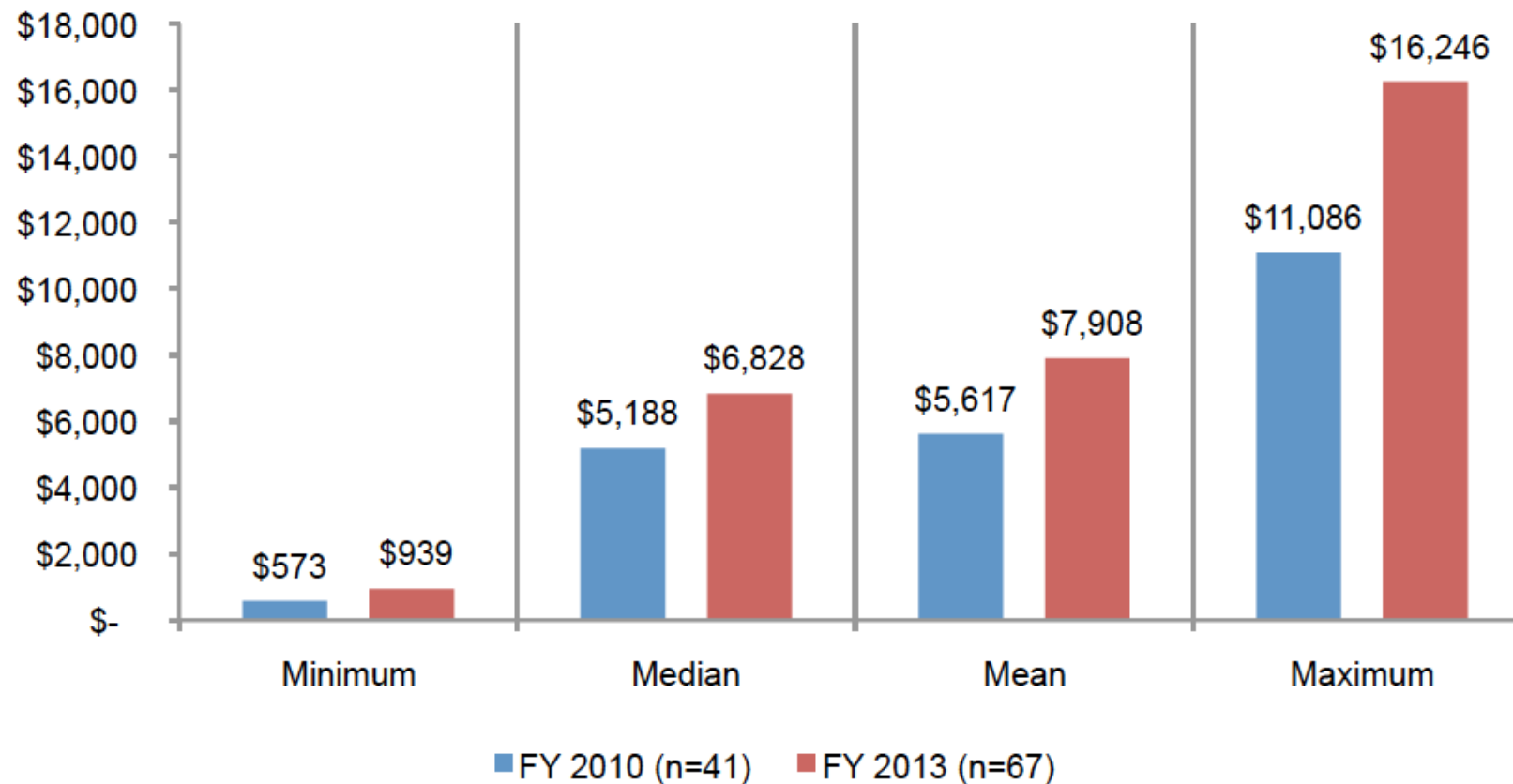
COST OF CLOUD OUTAGE



* Ref: Calculating the Cost of Data Center Outages, Ponemon Institute© Research Report



COST PER MINUTE



* Ref: Calculating the Cost of Data Center Outages, Ponemon Institute© Research Report

RESOURCE AND SERVICE FAILURES

- A **failure** is an event that makes a system fail to operate according to its specifications.
 - Service Failure
 - Resource Failure
- Service Failure is a service unavailability caused by resource failures.
- Resource Failure is resource unavailability caused by an incident.



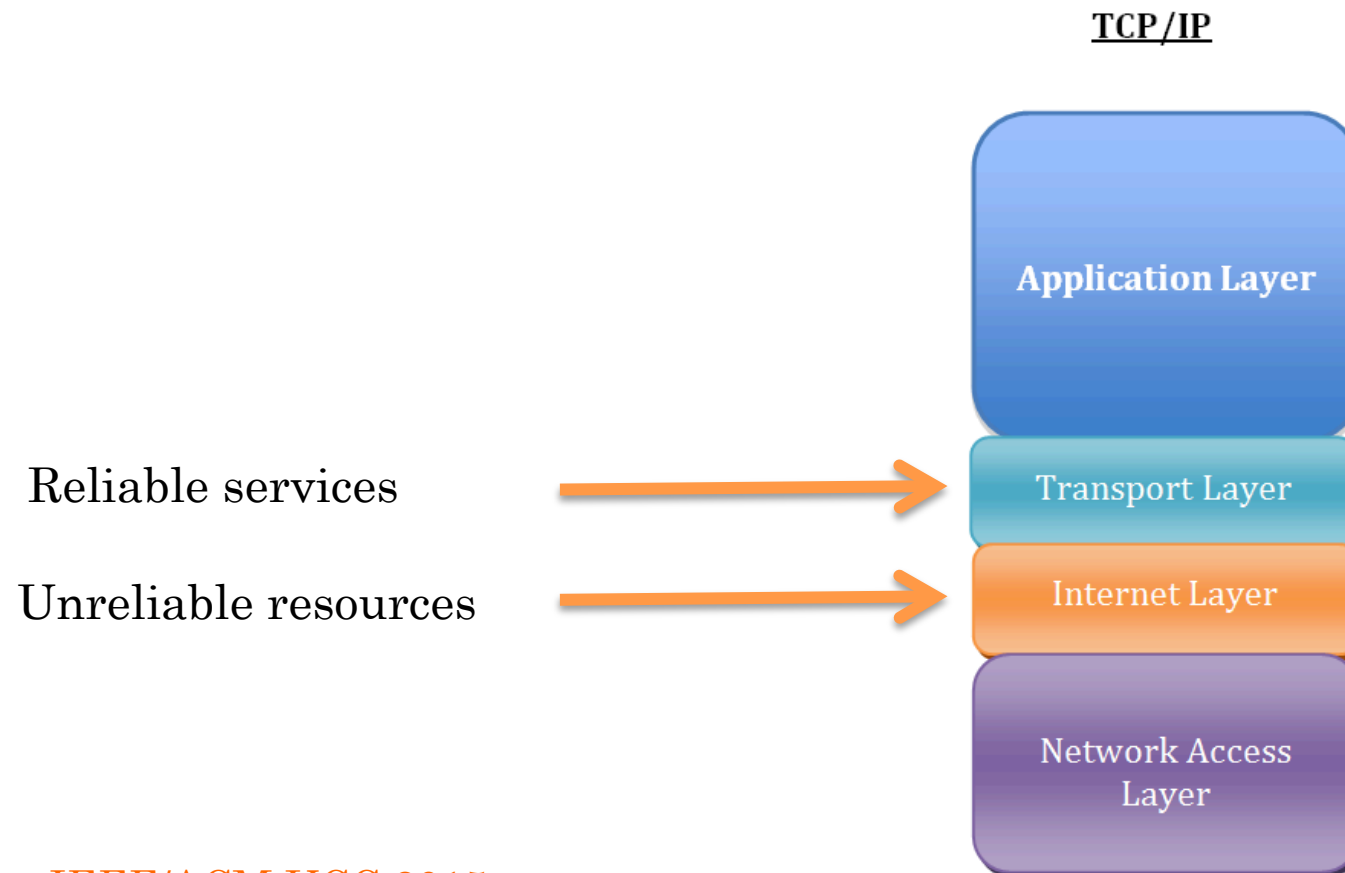


RESOURCE FAILURES

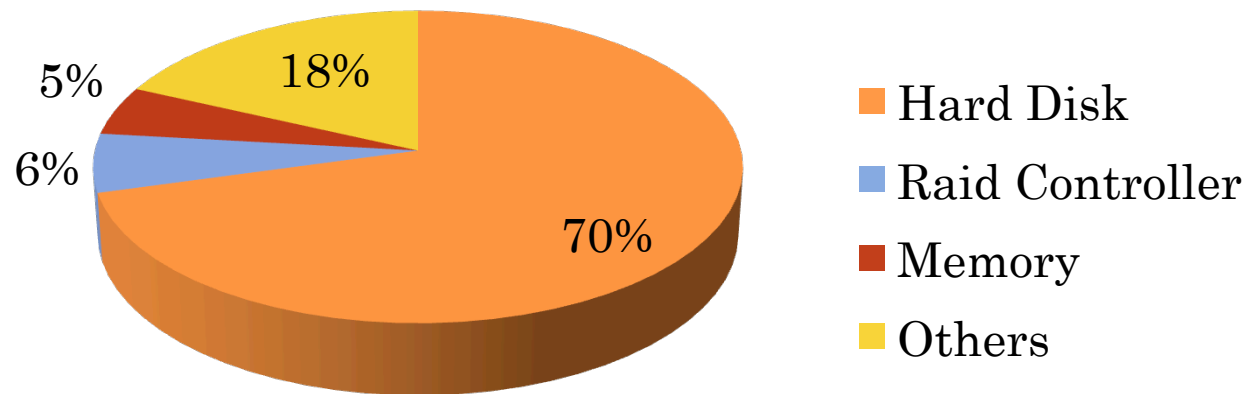
- Resource failure is inevitable where failure is the norm rather than exception.
 - $AFR = 8\%$
 - $N = 100,000$ nodes
 - $F = 22$ node failures per day

CLOUD SERVICE RELIABILITY

- Providing *Reliable Service* over unreliable resources



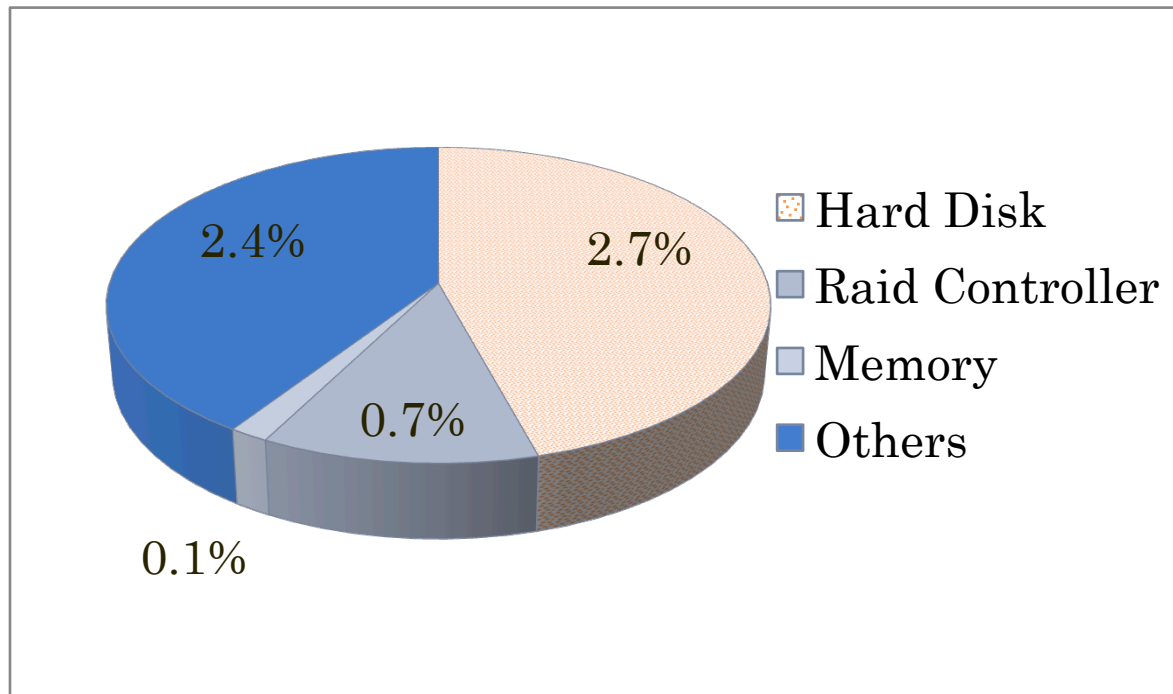
CLASSIFYING FAILURES OF SERVER



Hard disks are the not only the most replaced component, they are also the most dominant reason behind server failure!!

Vishwanath, Kashi Venkatesh, and Nachiappan Nagappan. "Characterizing cloud computing hardware reliability." Proceedings of the 1st ACM symposium on Cloud computing. ACM, 2010.

FAILURE RATE FOR COMPONENTS



The cost of per server repair (which includes downtime; IT ticketing system to send a technician; hardware repairs) is \$300. This amounts close to 2.5 million dollars for 100,000 servers.

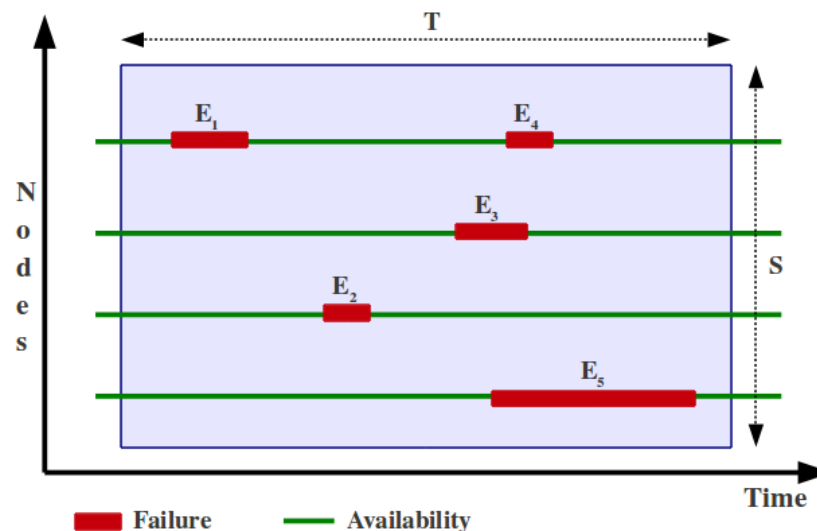
Vishwanath, Kashi Venkatesh, and Nachiappan Nagappan. "Characterizing cloud computing hardware reliability." Proceedings of the 1st ACM symposium on Cloud computing. ACM, 2010.



FAILURE CORRELATION

- Correlation in Failures → *overlapped failures*
 - Spatial
 - Temporal
- Spatial correlation means multiple failures occur on different nodes within a short time interval.
- Temporal correlation is the skewness of the failure distribution over which means failure events exhibit considerable **autocorrelation** at small time lags, so the failure rate changes over time.

FAILURES IN SERVICE



- The sequence of overlapped failures

$$H = \{F_i \mid F_i = (E_1, \dots, E_n), T_s(E_{i+1}) \leq T_e(E_i)\}$$

- Downtime of the service

$$D = \sum_{\forall F_i \in H} (\max\{T_e(F_i)\} - \min\{T_s(F_i)\})$$



CHALLENGE IN CLOUD RELIABILITY

○ Resource Redundancy

- Have more backup resources to tolerate the failures
 - Power consumption
 - Hiding failures

○ Checkpointing

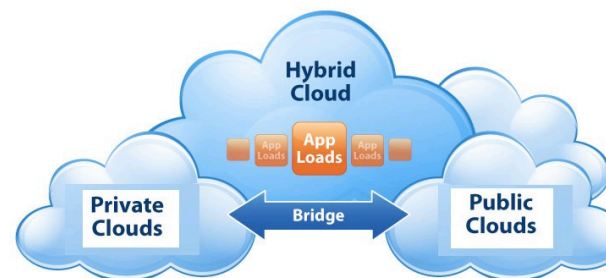
- Keep saving the state of applications to recover from the failures
 - Overhead
 - Complexity

○ Failure Model

- Adapting the knowledge of failure pattern into resource provisioning and resource management

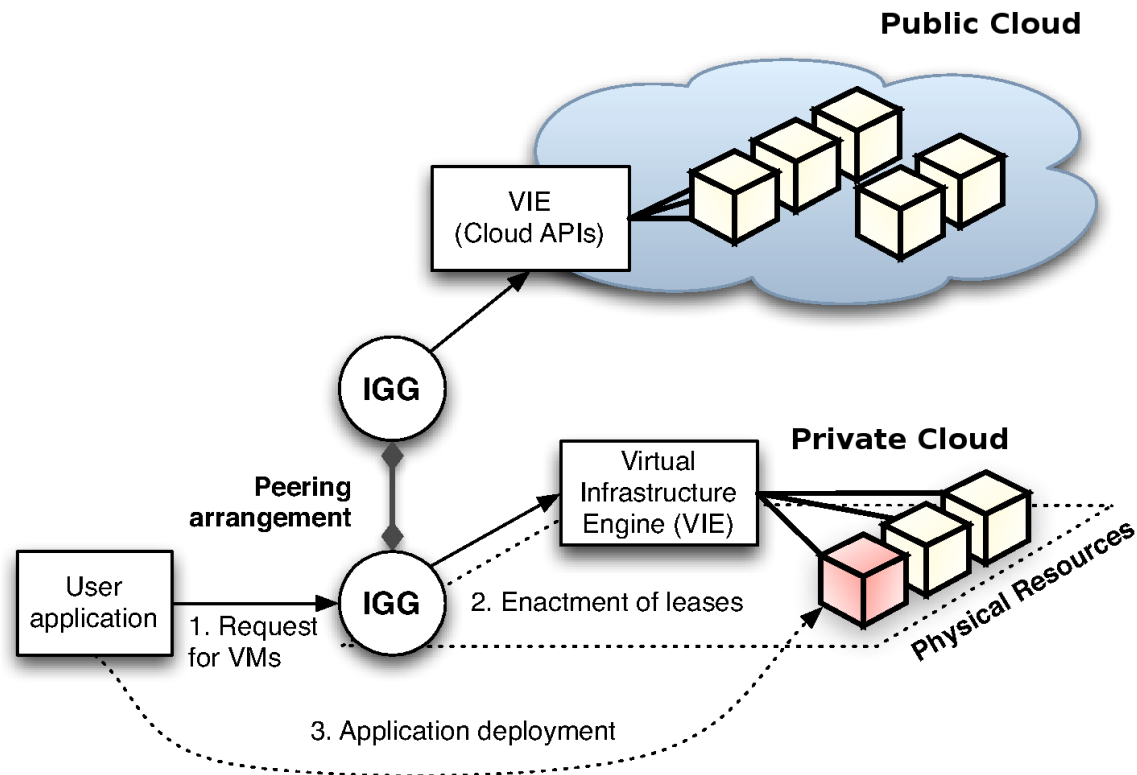
CASE STUDY

- Hybrid Cloud Systems
 - Public Clouds
 - Private Clouds (Less Reliable)



- Resource Provisioning in Hybrid Cloud
 - Users' QoS (i.e., deadline)
 - Resource failures
- Taking into account
 - Workload model
 - Failure characteristics
 - Failure correlations
 - Failure model

HYBRID CLOUD ARCHITECTURE



WORKLOAD MODEL

○ Scientific Applications

- Potentially large number of resources over a short period of time.
- Several tasks that are sensitive to communication networks and resource failures (*tightly coupled*)

○ User Requests

- Type of virtual machine;
- Number of virtual machines;
- Estimated duration of the request;
- Deadline for the request (optional).





PROPOSED APPROACHES

- Knowledge-free Approach
 - *No Failure Model*
 - Using failure correlation
 - Three brokering policies
- Knowledge-based Approach
 - *Failure Model*
 - Generic resource provisioning model
 - Two brokering policies (cost-aware)
- Workload model
 - Request size
 - Request duration



PROPOSED POLICIES

○ Size-based Strategy

- **Spatial correlation** : multiple failures occur on different nodes within a short time interval
- *Strategy*: sends wider requests to more reliable public Cloud systems
- Mean number of VMs per request
 - P_1 : probability of one VM
 - P_2 : probability of power of two VMs

$$\bar{S} = P_1 + 2^{\lceil k \rceil} (P_2) + 2^k (1 - (P_1 + P_2))$$

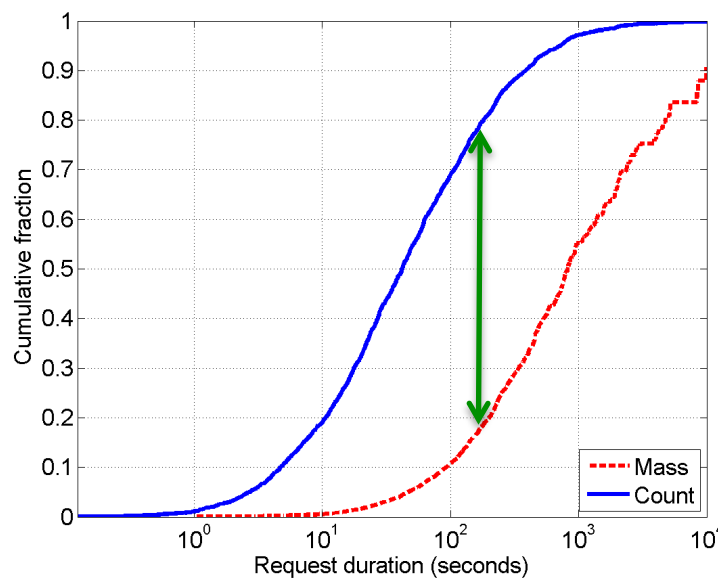
- Request size: two-stage uniform distribution (l, m, h, q)

$$k = \frac{ql + m + (1 - q)h}{2}$$

PROPOSED POLICIES (CONT.)

○ Time-based strategy

- **Temporal correlation**: the failure rate is time-dependent and some periodic failure patterns can be observed in different time-scales
- **Request duration**: are *long tailed*.



- The mean request duration
 - Lognormal distribution in a parallel production system

$$\overline{T} = e^{\mu + \frac{\sigma^2}{2}}$$



PROPOSED POLICIES (CONT.)

○ Area-based strategy

- Making a **compromise** between the size-based and time-based strategy
- The mean area of the requests

$$\overline{A} = \overline{T} \cdot \overline{S}$$

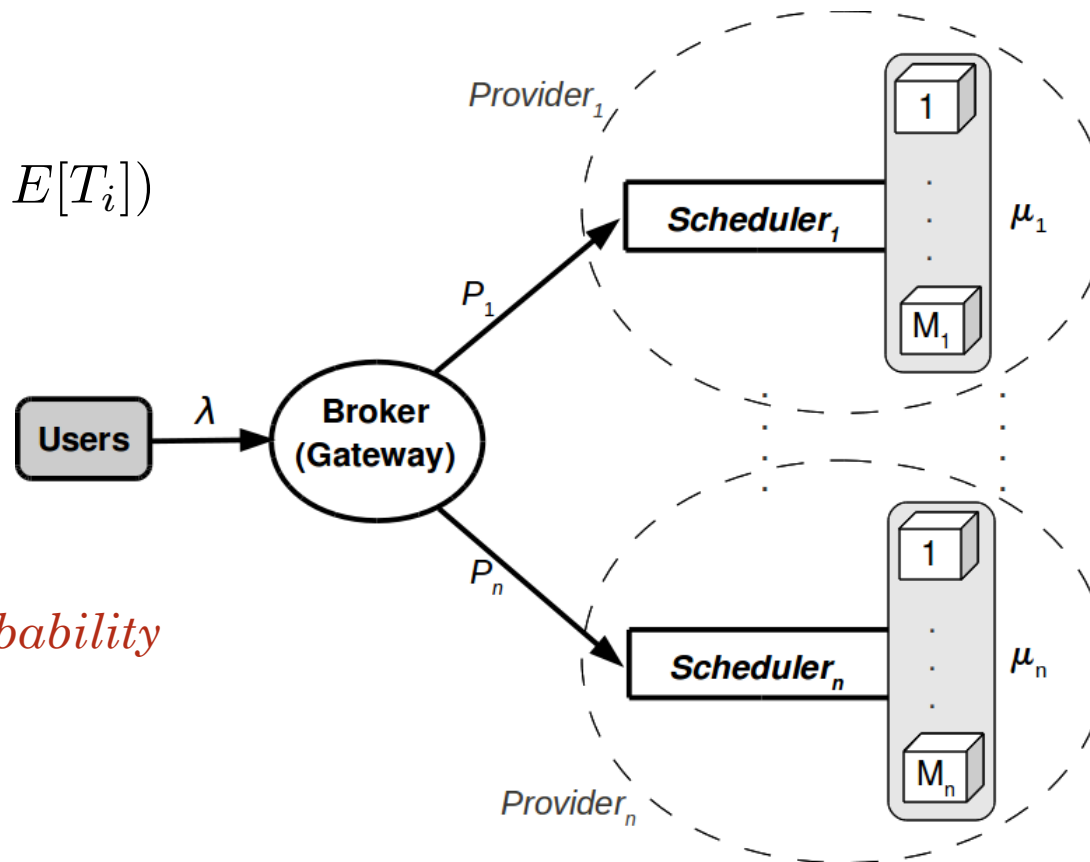
- This strategy sends long *and* wide requests to the public Cloud,
- It would be more conservative than a *size-based* strategy and less conservative than a *time-based* strategy.

KNOWLEDGE-BASED APPROACH: GENERIC RESOURCE PROVISIONING MODEL

- Model based on routing in distributed parallel queue

$$\min \sum_{i=1}^n (K_i \cdot E[T_i])$$

P_i : routing probability



MODEL PARAMETERS

- Using Lagrange multipliers methods, we obtained the routing probability as follows:

$$P_i = \frac{\mu_i}{\lambda} - \frac{\sum_{i=1}^n \mu_i - \lambda}{\lambda} \cdot \frac{\sqrt{K_i \eta_i}}{\sum_{i=1}^n \sqrt{K_i \eta_i}}$$

- Private Cloud service rate

$$\mu_s = \left(\frac{\bar{W}}{M_s \cdot \tau_s} \frac{t_a + t_u}{t_a} + L_s \right)^{-1}$$

- Public Cloud service rate

$$\mu_c = \left(\frac{\bar{W}}{M_c \cdot \tau_c} + L_c \right)^{-1}$$

ADAPTIVE POLICIES

- Adaptive with Random Sequence (ARS)
 - Routing probabilities (P_i)
 - Dispatch using *Bernoulli* distribution
- Adaptive with Deterministic Sequence (ADS)
 - Routing probabilities (P_i)
 - Dispatch using *Billiard* sequence

$$i_b = \min_{\forall i} \left\{ \frac{X_i + Y_i}{P_i} \right\}$$





SCHEDULING ALGORITHMS

- Scheduling the request across private and public Cloud resources
- Two well-know algorithms where requests are allowed to leap forward in the queue
 - Conservative backfilling
 - Selective backfilling

$$XFactor = \frac{W_i + T_i}{T_i}$$

- VM Checkpointing
 - VM stops working for the unavailability period
 - The request is started from where it left off when the node becomes available again

PERFORMANCE EVALUATION

- CloudSim Simulator
- Performance Metrics

- Deadline violation rate

- Slowdown
$$Slowdown = \frac{1}{M} \sum_{i=1}^M \frac{W_i + \max(T_i, bound)}{\max(T_i, bound)}$$

- Cloud Cost on EC2

$$Cost_{pl} = (H_{pl} + M_{pl} \cdot H_u) C_n + (M_{pl} \cdot B_{in}) C_x$$

- Workload Model

- Parallel jobs model of a multi-cluster system (i.e., DAS-2)

Input Parameters	Distribution/Value
Inter-arrival time	Weibull ($\alpha = 23.375, 0.2 \leq \beta \leq 0.3$)
No. of VMs	Loguniform ($l = 0.8, m, h = \log_2 N_s, q = 0.9$)
Request duration	Lognormal ($2.5 \leq \mu \leq 3.5, \sigma = 1.7$)
P_1	0.02
P_2	0.78

PERFORMANCE EVALUATION (CONT.)

○ Failures from Failure Trace Archive (FTA)

- <http://fta.scem.uws.edu.au/>

• Grid'5000 traces

- 18-month
- 800 events/node

Parameters	Description	Value (hours)
t_a	Mean availability length	22.25
σ_a	Std of availability length	41.09
t_u	Mean unavailability length	10.22
σ_u	Std of unavailability length	40.75

○ Synthetic Deadline

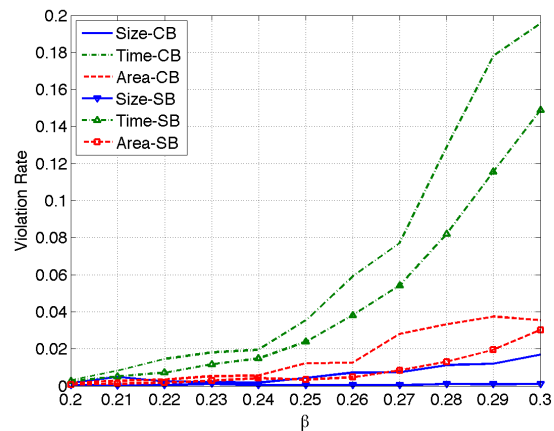
$$d_i = \begin{cases} st_i + (f \cdot ta_i), & \text{if } [st_i + (f \cdot ta_i)] < ct_i \\ ct_i, & \text{otherwise} \end{cases}$$

- f : stringency factor
- $f > 1$ is normal deadline (e.g., $f=1.3$)

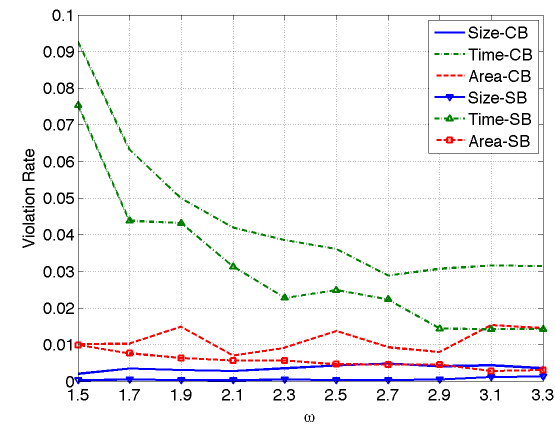
○ $M_s = M_c = 64$

SIMULATION RESULTS

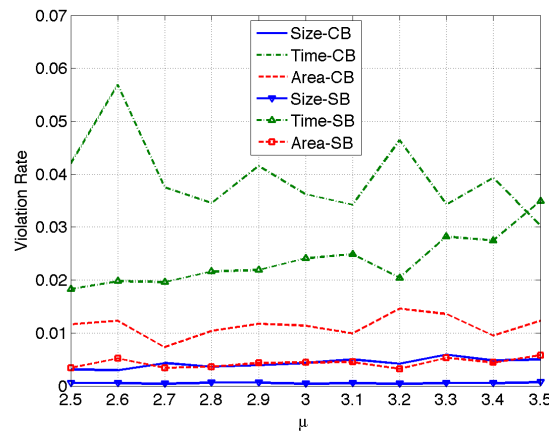
○ Violation rate (knowledge-free policies)



Request arrival rate



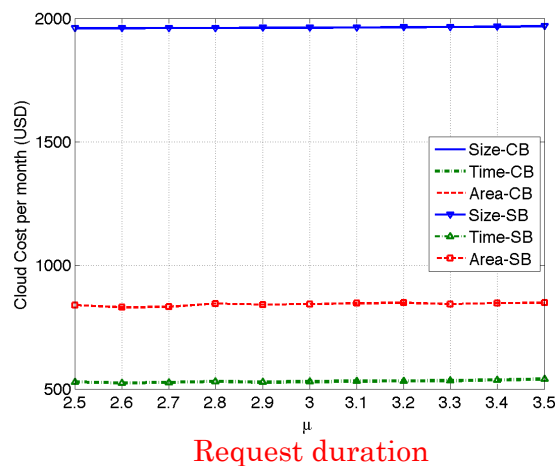
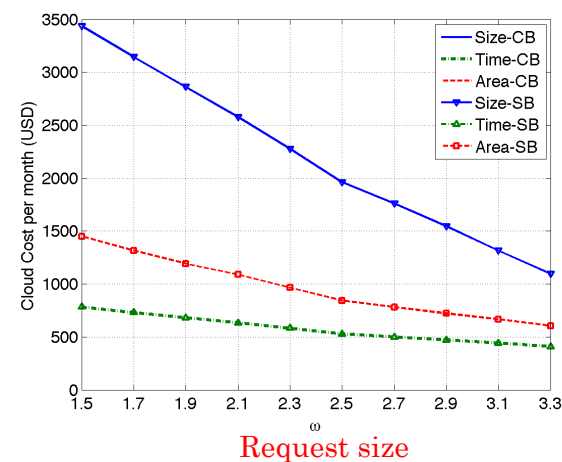
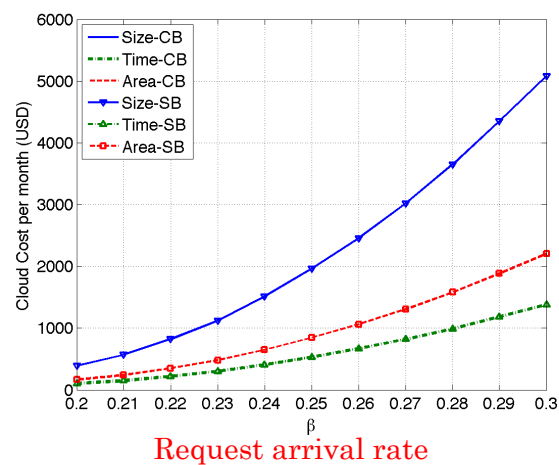
Request size



Request duration

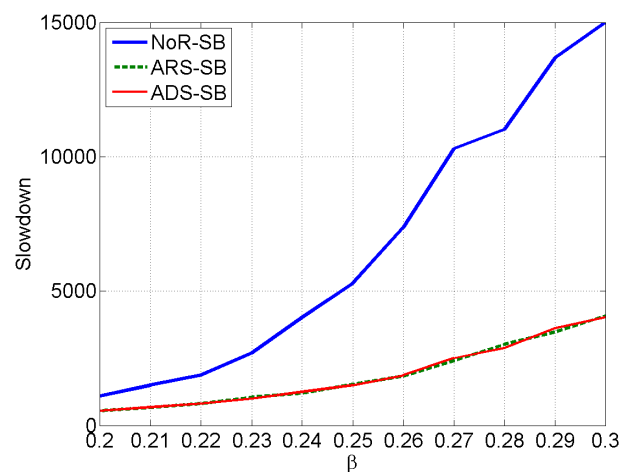
SIMULATION RESULTS (CONT.)

Cloud Cost on EC2 (knowledge-free policies)

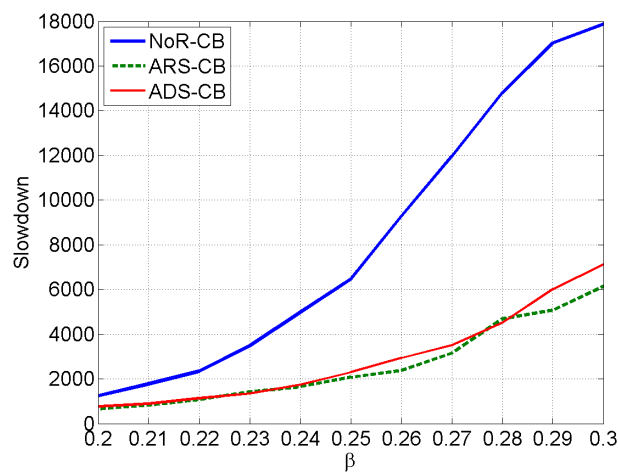


SIMULATION RESULTS (CONT.)

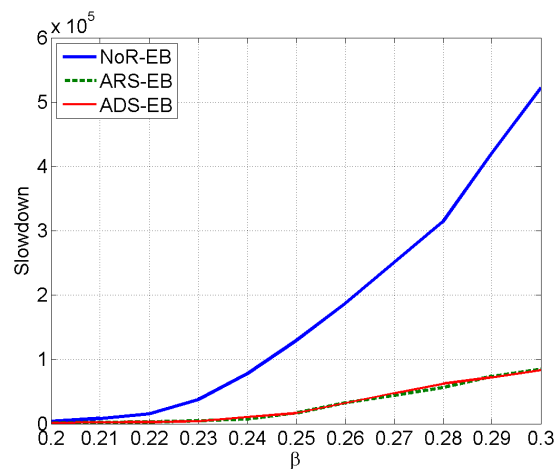
Slowdown (Knowledge-based policies)



Request arrival rate (SB)



Request arrival rate (CB)



Request arrival rate (EB)



DISCUSSIONS

- Improve performance and reliability of hybrid Cloud
 - Knowledge-free approach: 32% in terms of deadline violation and 57% in terms of slowdown while using 135\$/month on EC2
 - Knowledge-based approach: 4.1 times in terms of response time while using 1200\$/month on EC2



CONCLUSIONS

- Reliability is a critical challenge in Cloud Computing environments and needs more attention.
- Resource failures are inevitable, however we need to manage them to provide reliable services
- We need more reliability-aware resource provisioning policies for Cloud system



OPEN QUESTIONS

- Recourse Failures vs. Energy Consumption for Cloud Systems
 - How they are related?
- Reliability-as-a-Service (RaaS) in Cloud Computing
 - Providing reliability on demand based on the users' requirements (e.g., Amazon Spot Instances)
- Cost Model for Resource Failures in Cloud Systems
 - Repair Replacement



REFERENCES

- Bahman Javadi, Parimala Thulasiraman, Rajkumar Buyya, “*Enhancing Performance of Failure-prone Clusters by Adaptive Provisioning of Cloud Resources*”, *Journal of Supercomputing*, 63(2) (2013) 467-489.
- Bahman Javadi, Jemal Abawajy, and Rajkumar Buyya, “*Failure-aware Provisioning for Hybrid Cloud Infrastructure*”, *Journal of Parallel and Distributed Computing*, 72 (10) (2012) 1381-1331. . **ScienceDirect TOP25 Most Downloaded Articles**
- Bahman Javadi, Jemal Abawajy, and Richard O. Sinnott , “*Hybrid Cloud Resource Provisioning Policy in the Presence of Resource Failures*” 4th IEEE International Conference on Cloud Computing Technology and Science (CloudCom 2012), Taipei, Taiwan, December 2012, pp. 10-17. **Best Paper Finalist Award.**
- Bahman Javadi, Parimala Thulasiraman, Rajkumar Buyya, “*Cloud Resource Provisioning to Extend the Capacity of Local Resources in the Presence of Failures*”, 14th IEEE International Conference on High Performance Computing and Communications (HPCC-2012), Bradford, UK, June 2012, pp. 311-319.



Thank You

