Performance of Clouds

Issues and Research Directions

Eleni D. Karatza
Department of Informatics
Aristotle University of Thessaloniki
Greece

**2nd International Conference on
Cloud Computing and Services Science
(CLOSER - 2012)
Porto, Portugal
April 18-21, 2012**

Presentation Structure

- Grid Computing vs. Cloud Computing
- Grid Issues
- Clouds Issues
- Performance Evaluation
- Scheduling in Clouds
- From Cloud to Sky Computing
- Conclusions and Future Direction

Grid Computing vs. Cloud Computing (1/3)

- Parallel computers used in HPC are not always sufficient to cope with resource intensive scientific and commercial applications.

- Grids have emerged as an important infrastructure for serving demanding jobs and evolved to become the basis of Cloud computing.

Grid Computing vs. Cloud Computing (2/3)

- Computational and data grids and clouds are **large-scale distributed systems used for serving very large and complex applications**.

- Grids and Clouds performance became more important due to the tremendous increase of users and applications.

Grid Computing vs. Cloud Computing (3/3)

Important issues that must be addressed:

- Efficient scheduling,
- Resource management,
- Load balancing,
- Energy efficiency,
- Reliability,
- Security and Trust,
- Cost,
- Availability,
- Quality of Service.

Grid Issues (1/4)

- The main idea of Grid Computing:

  To use a large number of distributed high-performance computational resources while minimizing the related operating costs in order to solve complex and computationally demanding problems that practically could not be solved on a single resource.

Grid Issues (2/4)

- In such a dynamic, distributed computing environment, where resource availability varies dramatically, efficient resource allocation and job scheduling are essential.

- Grid scheduling manages the selection of the appropriate sites and resources for jobs, the allocation of jobs to specific resources and the monitoring of jobs execution.

Grid Issues (3/4)

- A grid system following a hierarchical architecture is organized at multiple levels:
  - At the grid level, a grid scheduler selects the appropriate sites for jobs
  - At the in-site local level, local schedulers allocate jobs to specific resources.
- Scheduling techniques should perform efficiently across several metrics that represent both:
  - user and system goals.

Grid Issues (4/4)

- The usage of energy has become a major concern for grid and cloud computing since the price of electricity has increased dramatically.
- **The energy consumption** is a metric aiming at diminishing the energy consumption of computations.
- This metric is always used in multi-criterion optimization problems, otherwise all the jobs would be scheduled sequentially on the most energy efficient machine.

Clouds Issues (1/18)

Cloud computing evolves from grid computing:

- It provides users the ability to lease computational resources from its virtually infinite pool for commercial, business, and scientific applications.

If cloud computing is going to be used for HPC, sophisticated methods must be considered for both parallel job scheduling and VM scalability.

Furthermore, high-speed, scalable, reliable networking is required for transferring data within the cloud and between the cloud and external clients.

Clouds Issues (2/18)

R. Buyya’s definition of Clouds

"A Cloud is a type of parallel and distributed system consisting of a collection of inter-connected and **virtualised** computers that are **dynamically provisioned** and presented as one or more unified computing resources based on **service-level agreements** established through **negotiation** between the service provider and consumers.”

Clouds Issues (3/18)

- Clouds may offer the **Infrastructure as a Service (IaaS)** where users have access to the execution environment through Virtual Machines (Amazon, Nimbus).
- Clouds offer many services - sometimes they are also referred to as **XaaS**: "**everything as a service**".
- Cloud providers such as Amazon with the EC2 (Amazon Elastic Compute Cloud) platform propose new services.
- It is now possible to rent high-speed clusters with the "Cluster Compute" instance of EC2 and run HPC applications with good performance.

Clouds Issues (4/18)

- Clouds were mostly used for simple **sequential applications**. However, recent evolutions enables the HPC community to run **parallel applications** in the Cloud.

- Good management policies can provide great improvements on different metrics:

  - maximum utilization of the resources,
  - faster execution times, and
  - better user's satisfaction (QoS guarantees).

Clouds Issues (5/18)

- Users can have access to a large number of computational resources at a fraction of the cost of maintaining a supercomputer center.
- A user can receive a service from the cloud without ever knowing **which** machines rendered the service, **where** it was located, or how many redundant copies of its data there are.
- The term “cloud” appears to have originated with depiction of the Internet as a cloud hiding many servers and connections.

Clouds Issues (6/18)

Cloud computing is a paradigm in which computing is moving from personal computers to large, centrally managed datacenters – **Questions:**

- How does cloud computing differ from Grid computing, and other previous models of distributed systems?
- What new functionalities are available to application developers and service providers?
- How do such applications and services leverage pay-as-you-go pricing models and rapid provisioning to meet elastic demands ?

Clouds Issues (7/18)

- **Similarities:**
  - Grids and Clouds aim at reducing the **costs** of computing and thus they enable more users to perform computations.
- **Differences:**
  - Contrary to Grids, Clouds resources can be accessed **on-demand** for any amount of time with the illusion of an infinite pool of resources.
  - Unlike most Grids, some Cloud architectures let users deploy their own **operating systems** using Virtual Machines to have a total control on their environment.

Michael Armbrust et als. Above the Clouds: A Berkeley View of Cloud Computing. Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley, Feb 2009.

Clouds Issues (8/18)

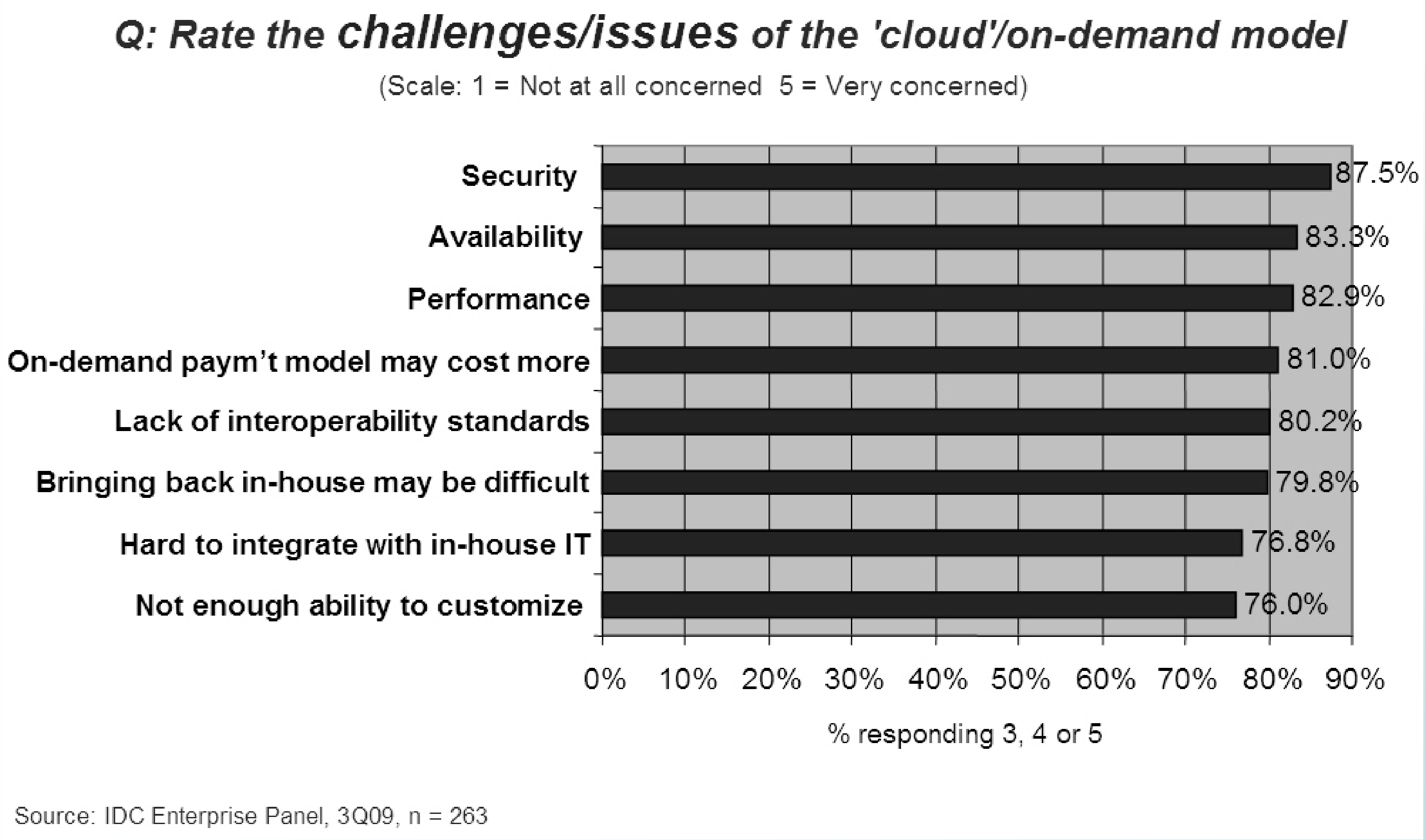


Fig. 1. Cloud Issues (Source: IDC Survey, 3Q09)

http://blogs.idc.com/ie/?p=730

Clouds Issues (9/18)

- The cloud model utilizes the concept of **Virtual Machines** (or VMs) which act as the computational units of the system.
- Depending on the computational needs of the jobs being serviced, new VMs can be **leased** and later **released** dynamically.
- It is important to study, analyze and evaluate both the performance and the overall cost of different scheduling algorithms.

Clouds Issues – Scheduling (10/18)

- The scheduling algorithms must seek a way to maintain **a good response time to leasing cost ratio.**
- Users requirements for **quality of service** (QoS) and specific system level objectives such as **high utilization**, **cost**, etc. have to be satisfied.
- Furthermore, **data security** and **availability** are critical issues that have to be considered as well.

Clouds Issues – Privacy and Trust (11/18)

- A significant barrier to the adoption of cloud services is that users fear **data leakage** and **loss of privacy** if their sensitive data is processed in the cloud.

- The privacy of data has to be ensured - Users have to be reassured that their data will not be inadvertently released to others.

- Cryptographic techniques for enforcing the **integrity** and **consistency** of data stored in the cloud have to be studied.

Clouds Issues – New Techniques (12/18)

Realizing cloud computing benefits requires new techniques for:

- managing shared data in the cloud,
- fault-tolerant computation,
- protecting privacy,
- scheduling, and sharing resources among applications,
- communication,
- billing.

Doug Terry, Microsoft, *Chairman, ACM Tech Pack Committee on Cloud Computing, 2011,* http://techpack.acm.org/cloud/

Clouds Issues – Programming Models (13/18)

Cloud computing platforms offer computing on demand but differ in the flexibility and functionality that they provide to programmers.

- **MapReduce** is a programming model and an associated implementation for processing and generating large datasets that is amenable to a broad variety of real-world tasks.

- Users specify the computation in terms of a map and a reduce function, and the underlying runtime system automatically parallelizes the computation across large-scale clusters of machines.

J. Dean and S. Ghemawat, 2008, “MapReduce: simplified data processing on large clusters”, Commun. ACM 51:1, 2008, 107-113.

Clouds Issues – Mobile Devices (14/18)

Questions:

- How do mobile devices at the edge of the network interact with cloud-based services to effectively manage data and computation on behalf of users?
- How cloud computing will augment applications on mobile devices, and vice versa, particularly for context-aware interaction.
- What new functionalities are available to application developers and service providers?

Clouds Issues – Computing Paradigms (15/18)

Today's computing represents the intersection of three broad paradigms for computing infrastructure and use:

(1) **Owner-centric** (traditional) HPC;

- resources are locally owned, with private access

(2) **Grid computing** (resource sharing);

- resources are both locally and externally owned

(3) **Cloud computing** (on-demand resource/ service provisioning).

- resources can be either externally owned (public cloud), or internally owned (private cloud).

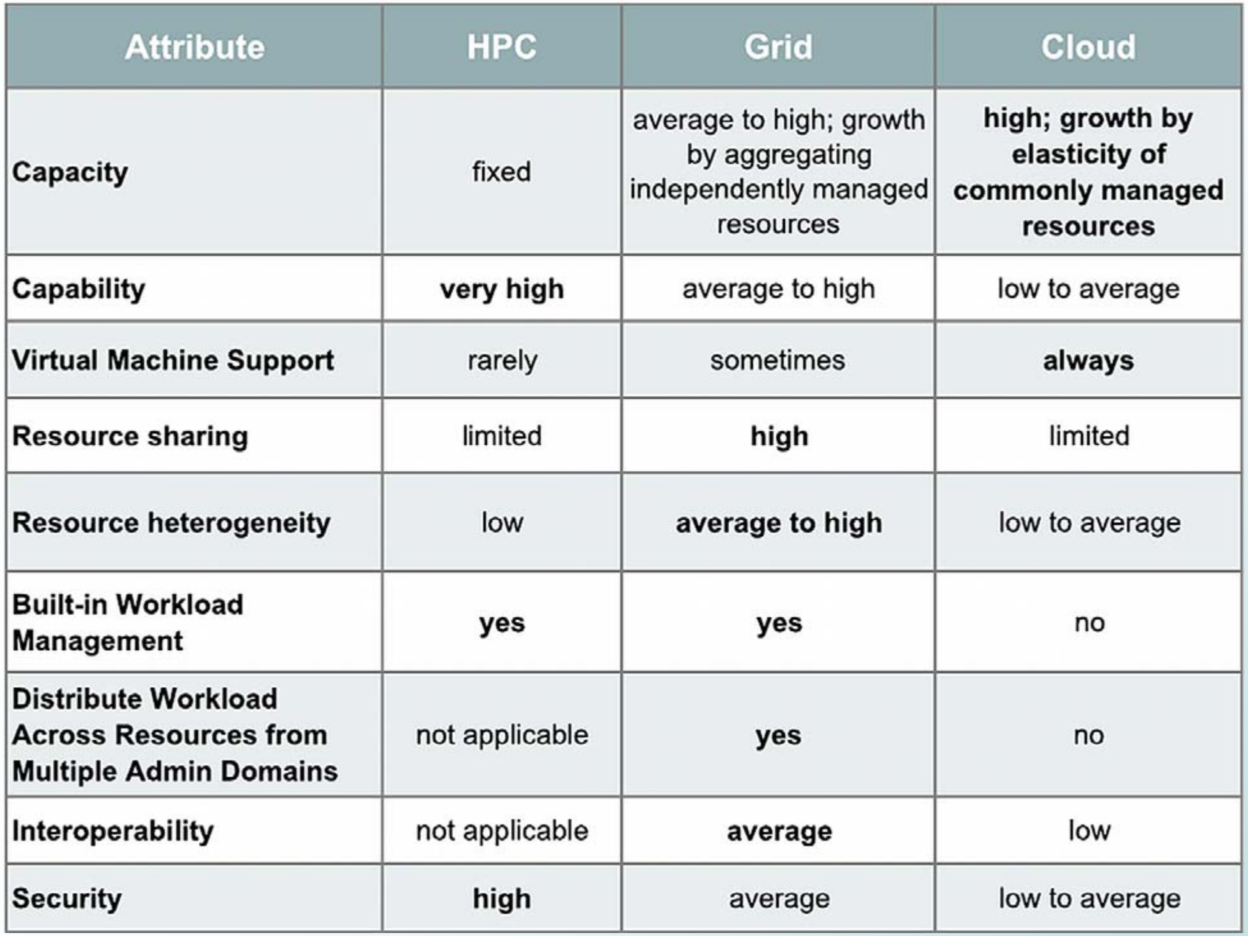
Clouds Issues - Computing Paradigms (16/18)

Each paradigm is characterized by a set of **attributes** of the resources making up the infrastructure and of the applications executing on that infrastructure.

- **G. Mateescu, W. Gentzsch, C. Ribbens**

  "Hybrid Computing—Where HPC meets Grid and Cloud Computing", Future Generation Computer Systems 27 (2011) 440–453

Clouds Issues (17/18)



Attribute	HPC	Grid	Cloud
Capacity	fixed	average to high; growth by aggregating independently managed resources	**high; growth by elasticity of commonly managed resources**
Capability	**very high**	average to high	low to average
Virtual Machine Support	rarely	sometimes	**always**
Resource sharing	limited	**high**	limited
Resource heterogeneity	low	**average to high**	low to average
Built-in Workload Management	**yes**	**yes**	no
Distribute Workload Across Resources from Multiple Admin Domains	not applicable	**yes**	no
Interoperability	not applicable	**average**	low
Security	**high**	average	low to average

Source: G. Mateescu et al. / Future Generation Computer Systems 27 (2011) 440–453

Fig.2. Comparison of Attributes

Clouds Issues - Hybrid Computing (18/18)

- Each of the three major computing paradigms has its strengths and weaknesses.

- The motivation for **hybrid computing** is to combine all three paradigms so that strengths are maintained or even enhanced, and weaknesses are reduced.

Grid and Cloud Performance – Workloads (1/8)

- Workloads consist of single task jobs and parallel jobs.
- Parallel jobs may consist of:
  - **independent tasks** which can execute on any processor and in any order
  - **tasks which need to frequently communicate with each other** – they start essentially at the same time and execute for the same amount of time (***gangs***).
- Appropriate scheduling techniques should be employed in each job type case.

Grid and Clouds Performance – Simulation (2/8)

Performance Evaluation -Simulation

- The performance evaluation of grids and clouds is often possible only by simulation rather than by analytical techniques, due to the complexity of the systems.
- Simulation can provide important insights into the efficiency and tradeoffs of scheduling in large-scale heterogeneous distributed systems, such as grids and clouds.
- Synthetic workloads – Traces from real systems.

Grid and Clouds Performance - Scheduling (3/8)

Scheduling manages:

- the **selection** of resources for a job,
- the **allocation** of jobs to resources and
- the **monitoring** of jobs execution.

Grid and Clouds Performance – Queueing (4/8)

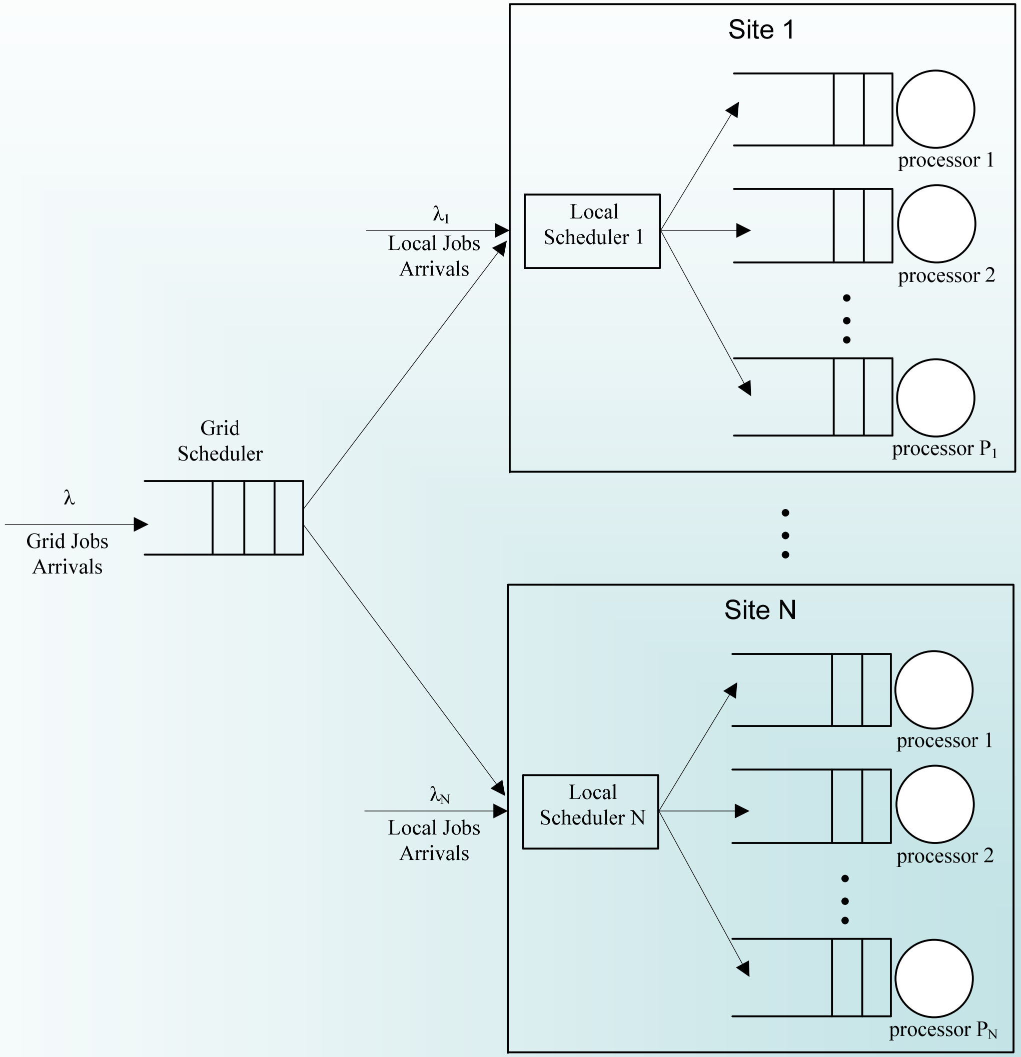


Fig. 3: A grid system model

Grid and Clouds Performance – Energy Conservation (5/8)

- In complex distributed systems, **energy conservation** is an important issue.

- Energy conservation can take place at multiple levels:
  - server level,
  - cluster level,
  - site level, and
  - grid broker level.

Grid and Clouds Performance – Energy Conservation (6/8)

Common practices to conserve energy include:

- **Shutting down idle servers**

  The idle power consumption of a server is about 50-60% of its peak power. Turning off the idle servers – versus the recovery time interval required.

- **Use of dynamic voltage scaling (DVS)**

  The voltage is increased or decreased, depending on the system load.

- **Virtualization**

  Many jobs often need only a fraction of the available computational resources. These jobs can be run within a virtual machine (VM) leading to significant increases in overall energy efficiency.

Grid and Clouds Performance– Energy Efficiency (7/8)

Processor characteristics

- Processors are characterized by:
  - Performance capabilities
  - Power characteristics

- Two classes of processors:
  - High performance (HP) processors
  - Energy efficient (EE) but slower processors.

Grid and Clouds Performance – Environment (8/8)

Data Centers - GreenCloud

- Power consumption of data centers has large impacts on environments.
- Especially, Cloud providers need a high amount of electricity to run and maintain their computational resources in order to provide the best service level for the customer.
- Researchers are seeking to find effective solutions to make data centers reduce power consumption while guarantee the performance from users' perspective.

Liang Liu et als. “GreenCloud: a new architecture for green data center”, *ICAC-INDST’09,* June 16, 2009.

Cloud Scheduling – Gang Scheduling (1/34)

- **Gang Scheduling** is an efficient job scheduling algorithm for time sharing, already applied in parallel and distributed systems.

- Gangs are parallel jobs that consist of tasks that are in **frequent communication** and therefore must execute both simultaneously and concurrently.

Cloud Scheduling – A Performance Study (2/34)

Ioannis A. Moschakis and Helen D. Karatza, “Evaluation of Gang Scheduling Performance and Cost in a Cloud Computing System”, J. of Supercomputing, Springer, DOI 10.1007/s11227-010-0481-4.

- The model utilizes the concept of VMs which act as the computational units of the system.

- Initially the system includes no VMs but, depending on the computational needs of the jobs being serviced new VMs can be leased and later released dynamically.

Cloud Scheduling – The Simulation Model (3/34)

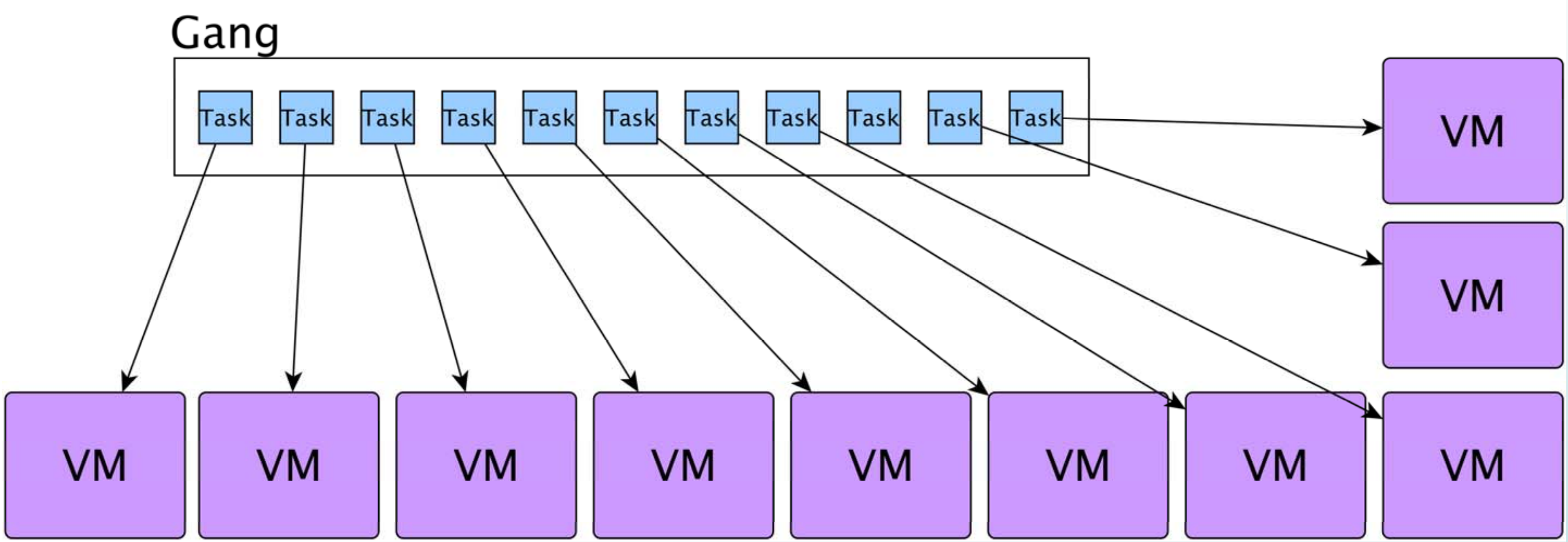


Fig. 4: A gang model

Cloud Scheduling – The Simulation Model (4/34)

- The simulation model consists of a single cluster of VMs connected with a *Dispatcher VM (DVM)*.
- Initially the system leases no VMs so the cluster is empty.
- Depending on the workload at any specific moment the system has the ability to lease new VMs up to a total number of *Pmax* = 120.

Cloud Scheduling – The Simulation Model (5/34)

- Each VM incorporates its own task waiting queue where the tasks of parallel jobs are dispatched by the DVM.

- The DVM also includes a waiting queue for jobs that where unable to be dispatched at the moment of their arrival due to either *inadequacy of VMs* or due to *overloaded VMs*.

Cloud Scheduling – Jobs Characteristics (6/34)

Jobs fall in two different categories of size based on probabilities.

The job entry point for the system is the DVM. If the degree of parallelism of a job is less than or equal to the number of the available VMs, the job is immediately dispatched.

The allocation of VMs to tasks is handled by the DVM which employs the *Shortest Queue First (SQF).*

Cloud Scheduling – The System Model (7/34)

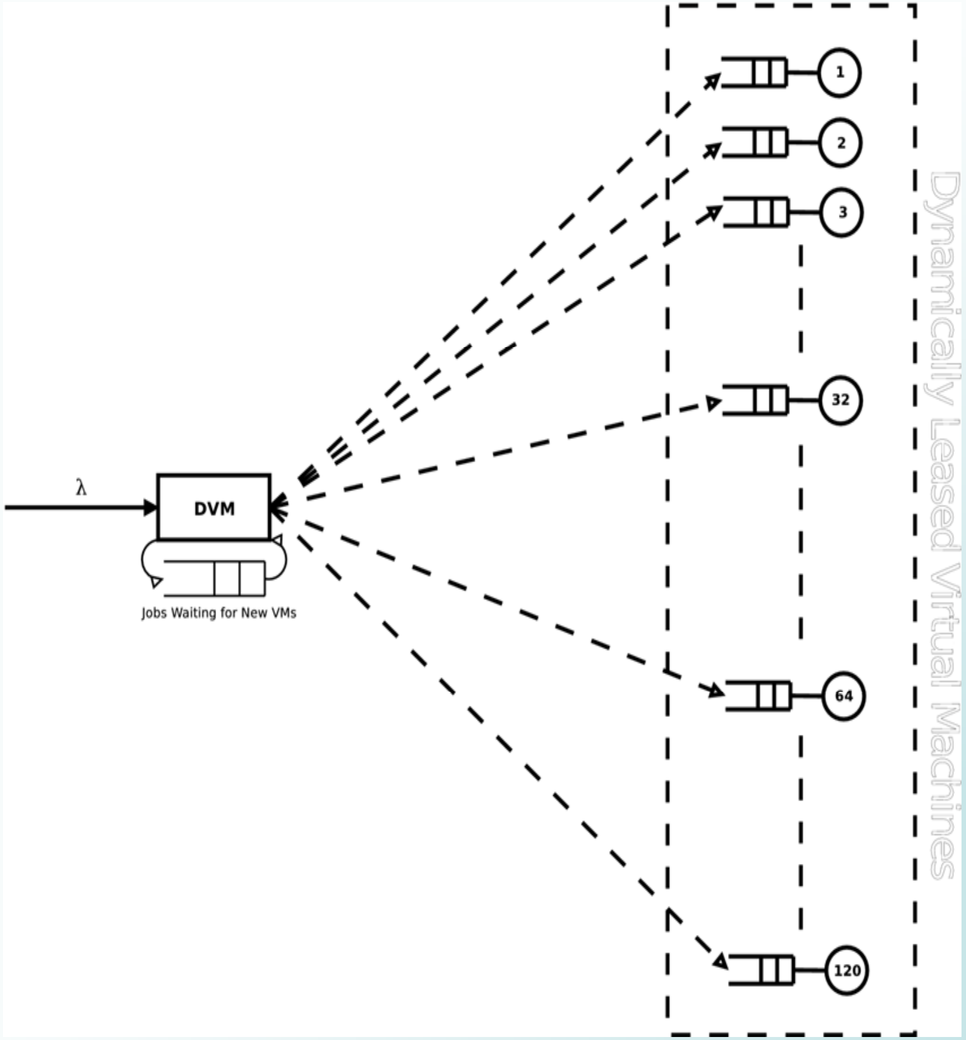


Fig. 5. The cloud queueing model

Cloud Scheduling – Dispatching (8/34)

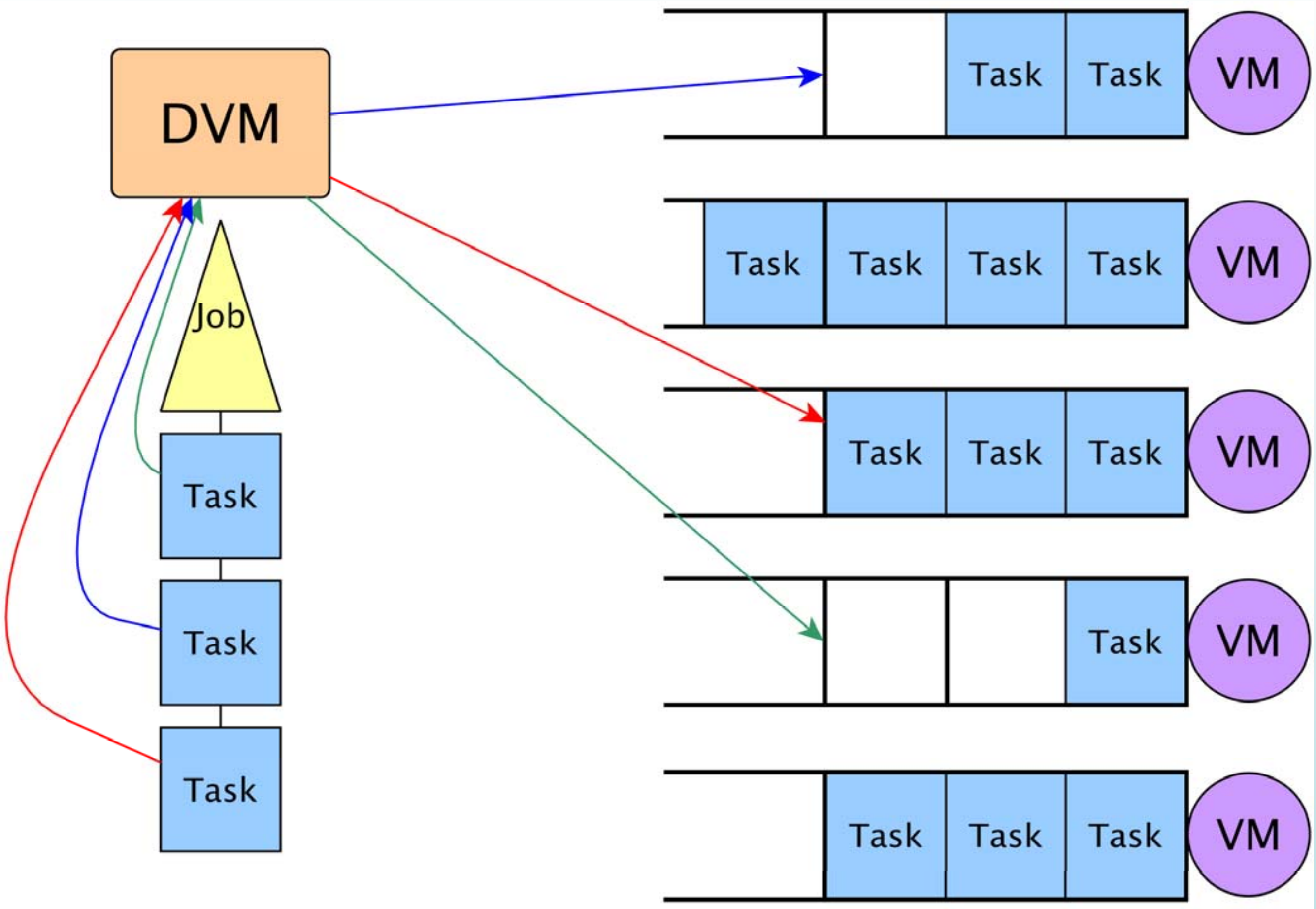


Fig. 6. Gang tasks dispatching

Cloud Scheduling – Policies (9/34)

- **Adaptive First Come Fist Served (AFCFS):**
  - Attempts to schedule a job (gang) whenever processors assigned to its tasks are available.

- **Largest Job First Served (LJFS):**
  - Tasks are placed in increasing job size order in processor queues. That is, tasks that belong to larger jobs (gangs) are placed at the head of the queues.

Cloud Scheduling - VM Handling (10/34)

- **VMs Lease / Release**

  The *Cloud* provides users with the ability to quickly up-scale or sub-scale their available resources.

  The addition of more VMs is accomplished through a virtual machine cloning process which involves the replication of a single initial state that all new virtual machines share.

  In this system model a delay is introduced which refers to the time that the VM cloning process will take to create a stated number of new VMs.

Clouds Scheduling - Inadequate VMs (11/34)

The ***lease/release* cycle** of VMs happens dynamically while the system is in operation.

– Inadequate VMs. When a large job arrives and the system has an inadequate amount of VMs to serve the job, then the newly arrived job enters the waiting queue of the DVM and waits while the system provisions for new virtual machines.

This procedure obviously involves a certain delay that refers to the real world delay of cloning a virtual machine and inserting it in the VM cluster..

Cloud Scheduling - Overloaded VMs (12/34)

– **Overloaded VMs.** When a new job arrives the system checks the *Average Load Factor (ALF)* of the available VMs:

$$ALF = \frac{\sum_{i=1}^{P_l} t_i}{P_l}$$

where t_i is the number of tasks currently assigned to VM i and P_l is the number of VMs leased by the system at that moment.

Should *ALF* surpass a certain threshold, the system provisions for new VMs equal to the degree of parallelism of the arriving job.

Cloud Scheduling - Releasing VMs (13/34)

VMs can also be released when they are not needed.

VMs are released only if certain criteria are met:

- The VM is currently idle and the VM’s task waiting queue is also empty.
- The removal of the VM from the system will not cause a new shortage of VMs, for jobs that are waiting in the DVM’s queue for new VMs to be leased.

Cloud Scheduling – Performance / Cost (14/34)

- The use of the *Cloud* is “**cost- associative**”:

  One pays only for the computing time which is equivalent to the total lease time of virtual machines.

- *Cost to performance efficiency* view.

- *Total lease time* (TL) of virtual machines while the system is in operation:

$$LT = \sum_{i=1}^{P_{tot}} T_{lease(i)}$$

Cloud Computing – Parameters (15/34)

- In order to compare LJFS and AFCFS **cost-performance wise**, the following metrics are used:

  Average and Weighted, Response and Waiting Time in conjunction with Slowdown metrics.

  Cost-Performance Efficiency (CPE) metric to evaluate the gain in response time in relation to the cost.

Cloud Scheduling – Simulation Results (16/34)

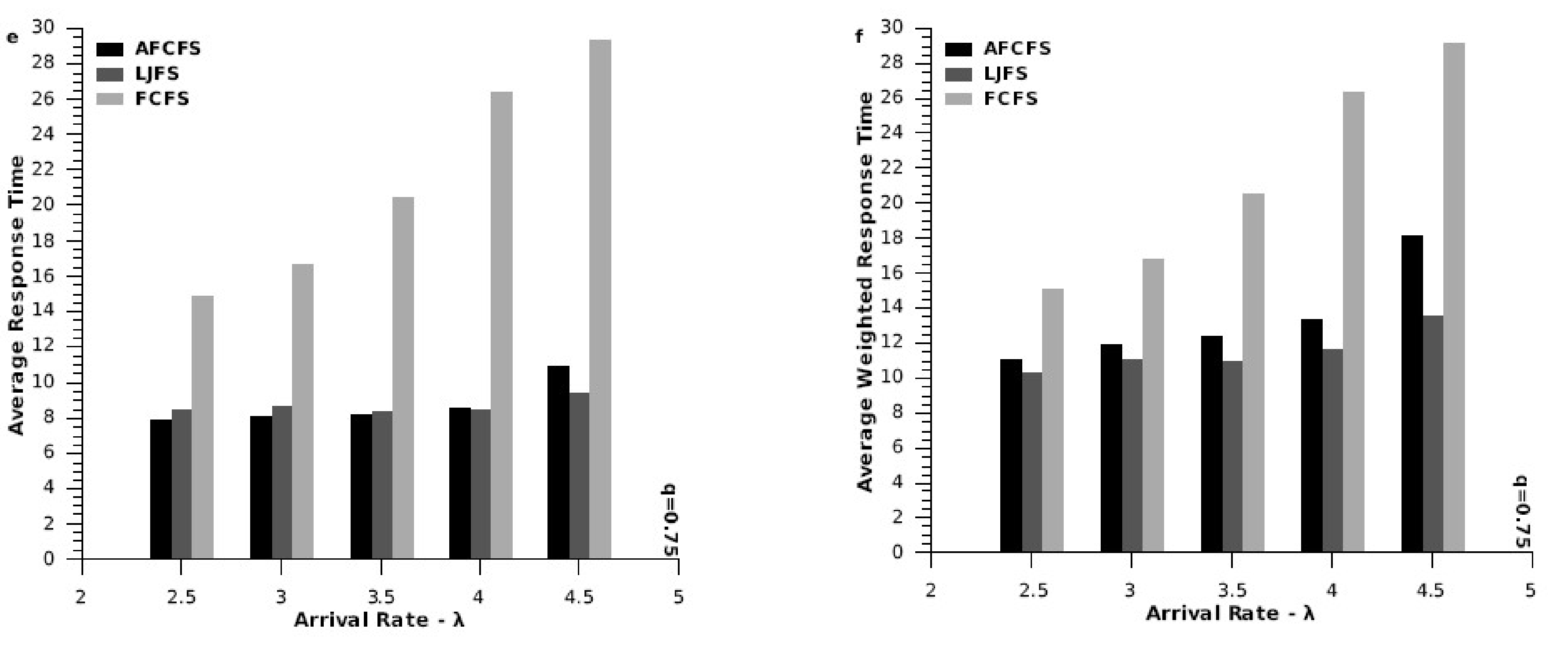
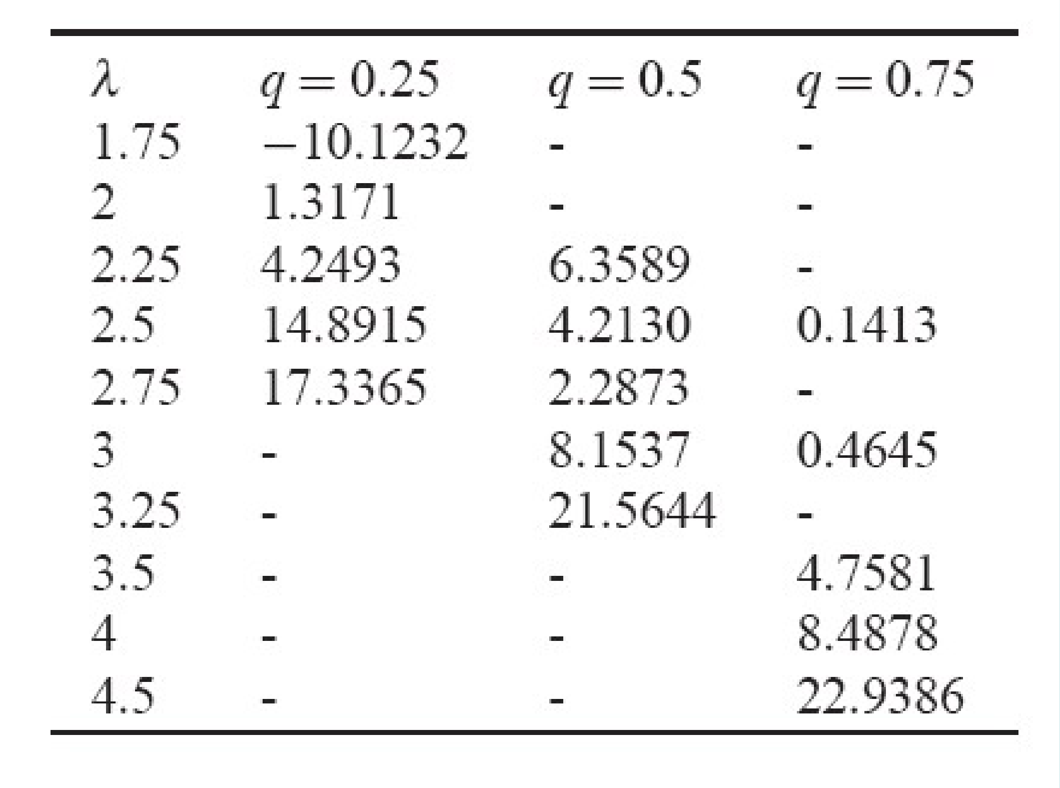


Fig. 7. Average Response Time & Average Weighted Response Time

Cloud Scheduling – Simulation Results (17/34)

Table 1 Cost-to-Performance Efficiency AFCFS-LJFS



λ	$q = 0.25$	$q = 0.5$	$q = 0.75$
1.75	-10.1232	-	-
2	1.3171	-	-
2.25	4.2493	6.3589	-
2.5	14.8915	4.2130	0.1413
2.75	17.3365	2.2873	-
3	-	8.1537	0.4645
3.25	-	21.5644	-
3.5	-	-	4.7581
4	-	-	8.4878
4.5	-	-	22.9386

Cloud Scheduling - Migrations-Starvation Handling (18/34)

- **I. Moschakis and H.D. Karatza**, “Performance and Cost evaluation of Gang Scheduling in a Cloud Computing System with Job Migrations and Starvation Handling”, Proceedings of ISCC 2011, June 28-July 1, 2011, Corfu, Greece, pp. 418-423.

Migration and Starvation Handling systems are incorporated
to deal with job fragmentation.

Cloud Scheduling - Migrations-Starvation Handling (19/34)

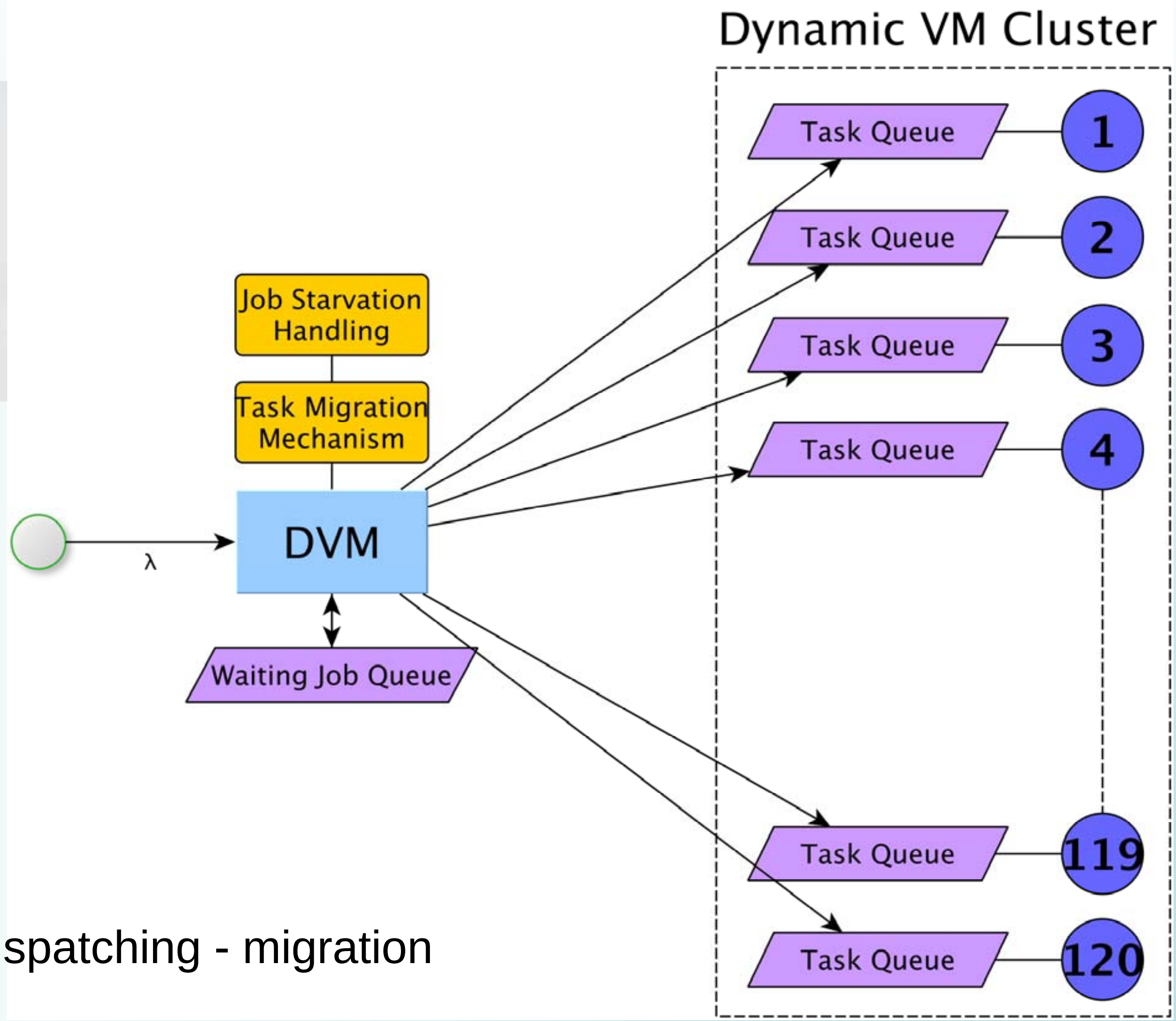


Fig. 8. Gang tasks dispatching - migration

Cloud Scheduling - Migrations-Starvation Handling (20/34)

- Frequently in gang scheduling processors remain idle while there are waiting tasks in their queues. This problem is called **fragmentation**.
- By using migrations we can solve the fragmentation problem.
- But we also have to regulate migrations carefully in order to avoid excessive overheads.

Cloud Scheduling - Migrations-Starvation Handling (21/34)

Job Migration

Two different migration strategies were used, each producing different amounts of overhead:

1. **First Fit (FF)** - This method selects the first job from those that fit and provides an easier implementation with less overhead.

2. **Best Fit (BF)** - This method selects the best job from those that fit but also incurs higher overhead.

Cloud Scheduling - Migrations-Starvation Handling (22/34)

Starvation Handling

Starvation Handling is implemented with a priority queue in conjunction with migrations.

- When a job's eXpansion Factor (X_{factor}) breaches the **Starvation Threshold**, the job is considered **starved**, and it is served in priority.
- Since “starved” jobs can migrate without limitations, the X_{factor} plays an important role in the regulation of migrations.

Cloud Scheduling - Migrations-Starvation Handling (23/34)

Cost versus Performance Efficiency

- Without migrations:

  LJFS was consistently more cost-efficient than AFCFS.

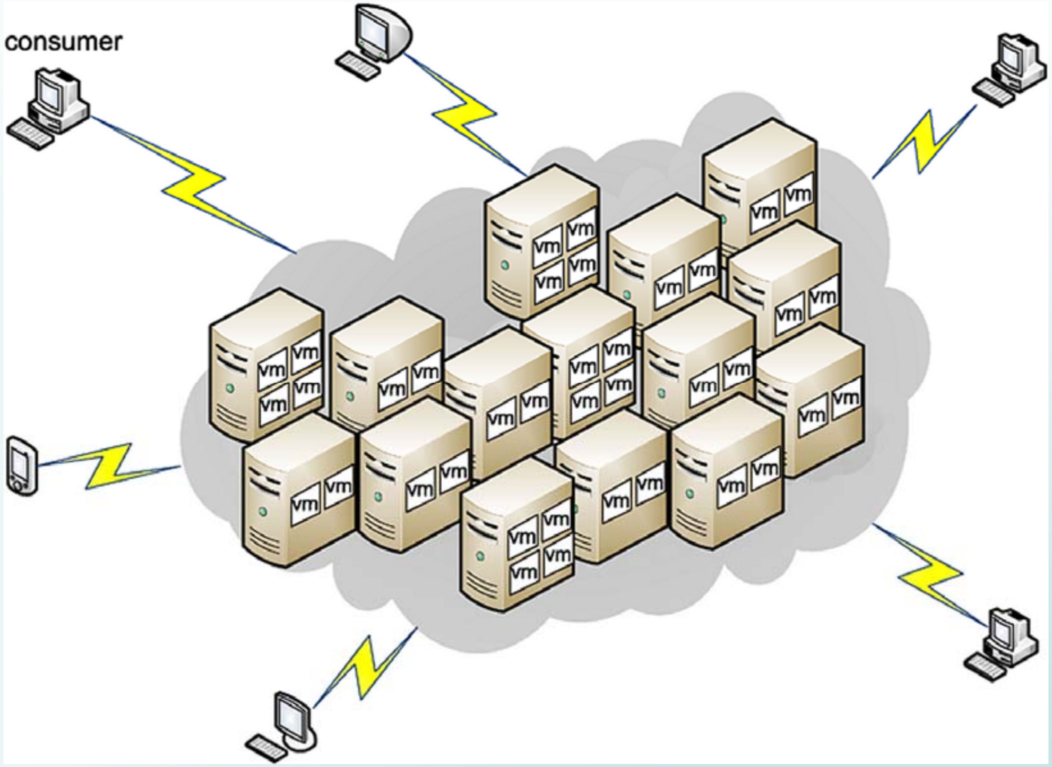
- With migrations:

  The superiority of LJFS over AFCFS is only marginal.

Cloud Scheduling- Task Consolidation (24/34)

- **Lee and Zomaya** (2010):

  "Task consolidation is an effective method to increase resource utilization and in turn reduces energy consumption".



Source: **Lee and Zomaya, Journal of Supercomputing, 2010**.

Fig. 9. A Cloud

Cloud Scheduling- Task Consolidation (25/34)

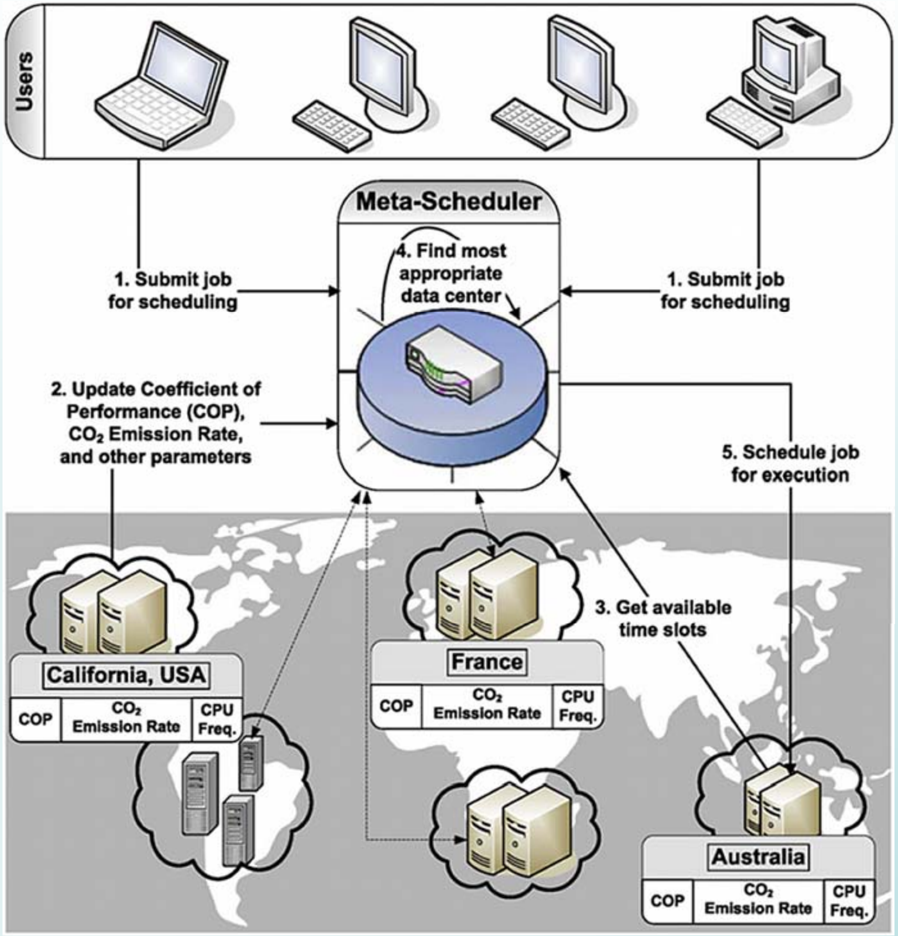
- **Lee and Zomaya (2010**) present two **energy-conscious** task consolidation heuristics, which aim to maximize resource utilization and explicitly take into account both active and idle energy consumption.

- Their heuristics assign each task to the resource on which the energy consumption for executing the task is explicitly or implicitly minimized without the performance degradation of that task.

  Y.C. Lee and A.Y. Zomaya, Energy efficient utilization of resources in cloud computing systems, Journal of Supercomputing, 2010.

Cloud Scheduling – Environment (26/34)

Garg, Yeo, Anandasivam and Buyya, Environment-conscious scheduling of HPC applications on distributed Cloud-oriented data centers, J. Parallel Distrib. Comput. 71 (2011) 732–749


Users
Meta-Scheduler
1. Submit job for scheduling
4. Find most appropriate data center
1. Submit job for scheduling
2. Update Coefficient of Performance (COP), CO₂ Emission Rate, and other parameters
5. Schedule job for execution
3. Get available time slots
California, USA
COP
CO₂ Emission Rate
CPU Freq.
France
COP
CO₂ Emission Rate
CPU Freq.
Australia
COP
CO₂ Emission Rate
CPU Freq.

Source: Garg, Yeo, Anandasivam and Buyya, J. Parallel Distrib. Comput. 71 (2011) 732–749

Fig. 10. Data Centers

Cloud Scheduling – Environment (27/34)

- The meta-scheduler interprets and analyzes the service requirements of a submitted application and decides whether to accept or reject the application based on the availability of CPUs.

- Its objective is to schedule applications such that the **carbon emissions can be reduced** and the profit can be increased for the Cloud provider, while the Quality of Service (QoS) requirements of the applications are met.

Cloud Scheduling – Management of Power (28/34)

Petrucci, V. Carrera, E.V. Loques, O. Leite, J.C.B. Mosse, D., Optimized Management of Power and Performance for Virtualized Heterogeneous Server Clusters, CCGrid 2011, pp. 23 – 32.

An approach for power and performance management in virtualized server clusters: **Petrucci et als. 2011**.

Contributions:

- a way of modeling power consumption and capacity of servers even under heterogeneous and changing workloads
- an optimization strategy based on a mixed **integer programming model** for achieving improvements on power-efficiency while providing performance guarantees in the virtualized cluster.

Cloud Scheduling – Management of Power (29/34)

- Costs related to application workload balancing and also switching due to frequent and undesirable turning servers on/off and VM relocations are addressed.

- The experiments reveal that the approach conserves about 50% of the energy required by a system designed for peak workload scenario, with little impact on the applications’ performance goals.

Cloud Scheduling – Hybrid Metaheuristic (30/34)

M. Mezmaza, et al. 2012, “A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems”, Journal of Parallel and Distributed Computing, Vol. 71, Issue 11, 2011, pp.1497-1508.

- Scheduling **precedence-constrained parallel applications** on heterogeneous computing systems like cloud computing infrastructures: M. Mezmaza, et al., 2012.
- A parallel bi-objective hybrid genetic algorithm is proposed that takes into account, not only **makespan**, but also **energy consumption**.
- Focus on the **island parallel model** and the **multistart parallel model**.

Cloud Scheduling – Hybrid Metaheuristic (31/34)

The new method is based on dynamic voltage scaling (DVS) to minimize energy consumption.

- In terms of **energy consumption**, the obtained results show that this approach outperforms other previous scheduling methods by a significant margin.
- In terms of **completion time**, the obtained schedules are also shorter than those of other algorithms.

The energy consumption is reduced by 47.5% and the completion time by 12%.

Cloud Scheduling – Machine Learning (32/34)

Berral, J. L., et. al. “Towards energy-aware scheduling in data centers using machine learning”, 1st International Conference on Energy-Efficient Computing and Networking, Passau, Germany, 2010.

In order to obtain an **energy-efficient data center**, **Berral et al. 2010** propose a framework that provides an intelligent consolidation methodology using different techniques such as:

- turning on/off machines,
- power-aware consolidation algorithms, and
- machine learning techniques to deal with uncertain information while maximizing performance.

Cloud Scheduling – Cost-wait Trade-offs (33/34)

User point of view: “The cloud provides an inexhaustible supply of resources, which can be dynamically claimed and released”.

- **Genau and Gaussa 2011**, present how billing models can be exploited by provisioning strategies to find **a trade-off** between fast/expensive computations and slow/cheap ones for independent sequential jobs.
- They study strategies based on classic heuristics for online scheduling and bin-packing problems, with the double objective of minimizing the wait time of jobs and the monetary cost of the rented resources.

- **S. Genaud and J. Gossa**, “Cost-wait Trade-offs in Client-side Resource Provisioning with Elastic Clouds”, Cloud 2011.

Cloud Scheduling – Inter-cloud Schedulers (34/34)

- **S. Sotiriadis, N. Bessis, N. Antonopoulos**, Towards inter-cloud schedulers: A survey of meta-scheduling approaches, 2011 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing.

- As the number of resource consumers is increasing significantly, the capacity-oriented clouds require coming together and **agreeing on common acting behaviours** for improving the quality of service, hence providing an overall optimal load allocation.

- **Sotiriadis et al. 2011** present a state-of-the-art review with a particular focus on the adoptability of current meta-schedulers for managing workloads, towards the **inter-cloud era**.

From Cloud to Sky Computing (1/3)

Sky with Clouds !

Grid Computing:
Aggregation of distributed heterogeneous resources

Sky Computing:
Aggregation of distributed heterogeneous Clouds.

Source: Keathey et als., IEEE Internet Computing

Fig. 11. Sky Computing

K. Keahey, M.Tsugawa, A.Matsunaga and J.Fortes, Sky Computing, IEEE Internet Computing, Vol.13, no. 5, 2009, pp. 43-51.

From Cloud to Sky Computing (2/3)

- In the past, site owners couldn’t trust a remote resource because they had no control over its configuration.
- Now that clouds let users control remote resources, however, this concern is no longer an issue.
- Combining the ability to trust remote sites with a trusted networking environment, a virtual site can now exist over distributed resources.

K. Keahey, M.Tsugawa, A.Matsunaga and J.Fortes:

“Because such dynamically provisioned distributed domains are built over several clouds, this kind of computing is called ***sky computing*****.”**

From Cloud to Sky Computing (3/3)

- In order to have different Clouds compatible together, standards are being developed and also users develop software compatible with multiple Cloud platforms.

  R. Buyya, R. Ranjan, and R. Calheiros. InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services. LNCS, Vol. 6081 pp. 13–31, Springer Berlin / Heidelberg, 2010.

  B. Rochwerger et als. The RESERVOIR Model and Architecture for Open Federated Cloud Computing. IBM Journal of Research and Development, 53(4), 2009.

  D. Nurmi et als. The eucalyptus open-source cloud-computing system, in:
  Proceedings of Cloud Computing and Its Applications, 2008.

Conclusions and Future Directions (1/3)

Advances in processing, communication and systems/middleware technologies had as a result new paradigms and platforms for computing.

- Grid computing has effectively addressed many integration, security, and heterogeneity aspects arising from larger-scale virtual organizations.
- The Cloud computing paradigm promises on-demand scalability, reliability, and cost-effective high-performance.
- Proper scheduling in Cloud environments can be effective from performance, energy and cost perspectives.

Conclusions and Future Directions (2/3)

- Our perception of computing is changing constantly.
- The rise of Cloud computing presents a new opportunity for the evolution of computing.
- Maybe, in few years computers will be nothing more than thin-clients, and all our processing will be done on the Clouds.

Conclusions and Future Directions (3/3)

- However, multiple issues have to be addressed before Clouds become viable for large scale processing like HPC.

- Security and availability will need the improvement of existing technologies, or the introduction of new ones, in order to achieve scalability that spans very large numbers of nodes.

Thank you

Email: karatza@csd.auth.gr