European Network for co-ordination of policies and programmes on e-infrastructures

Cloud Computing Economics:
An evidence-based approach for Research Applications

Deliverable Lead:	JISC
Related Work package:	WP3: Implementation of Joint Activities
Author(s):	Carlo Daffara
Dissemination level:	Public
Submission date:	18/01/2013
Project Acronym:	e-InfraNet
Website	e-infranet.eu
Call:	FP7-INFRA-2009-3.1
Project Number	246651
Instrument:	Coordination action (CA) – ERA-NET
Start date of Project:	01/Jan/2010
Duration:	36 months

Project funded by the European Commission within the Seventh Framework Programme		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Acknowledgements

This report has been developed within the project "European Network for co-ordination of policies and programmes on e-infrastructures" (e-InfraNet.eu). The project is an ERA-NET co-funded by the 7th Framework Programme of the European Commission.

The members of the consortium are:

Participant organisation name	Short Name	Country
Higher Education Authority	HEA	IE
Higher Education Funding Council for England	HEFCE (JISC)	UK
SURF	SURF	NL
Foundation for Science and Technology	FCT	PT
Latvian Academy of Sciences	LAS	LV
Ministry of Economy and Competitiveness	MINECO	ES
CSC - IT Center for Science	CSC	FI
National Information Infrastructure Development Institute	NIIFI	HU
The Scientific and Technological Research Council of Turkey	TUBITAK	TR
The Israel-Europe R&D Directorate	ISERD	IL
The General Secretariat for Research and Technology	GSRT	GR
Department of Economy, Science and Innovation, Flemish Government	EWI	BE

Author(s):	Carlo Daffara

Table of Contents

Figures

Tables

List of abbreviations

CapEx	Capital Expenditure
CPU	Central Processing Unit
GPU	Graphics Processing Unit
HPC	High Performance Computing
IaaS	Infrastructure as a Service
OpEx	Operation Expenditure
PaaS	Platform as a Service
PUE	Power Usage Effectiveness
SaaS	Software as a Service
SLA	Service Level Agreement
TCO	Total Cost of Ownership

Executive Summary

Cloud Computing is among the most rapidly adopted technologies by private companies and large Public Administrations. Despite the huge exposure, the vagueness of the term "cloud" and the lack of measurable data on the real economics of cloud usage makes it difficult to create a sensible model to evaluate whether a transition to a cloud environment makes sense for research organizations.

We have thus identified a set of specific aspects that introduce differences in the adoption and management economics of cloud compared to traditional clusters, grids and other research-oriented IT infrastructure. Those aspects have been applied to a set of models that take into consideration the use of private clouds in addition to the more common scenario of public clouds, deriving some potential recommendations for use in research applications, especially for the "IaaS"—Infrastructure as a Service—approach, that is currently dominating.

Among the recommendations and findings:

- Clouds in general provide a potential increase in flexibility and reduction of management costs for varying workloads, especially for "desktop computing", workloads that require no or little preservation after the end of the experiment or for jobs that may be repeated frequently. The "self-service" model is in particular one of the most cited advantages of the cloud model for scientists and researchers, along with the flexibility in selecting the most appropriate software stack for performing the research tasks among the large (and growing) library of software components that are publicly available. This flexibility is especially visible in fast-changing environments, where traditional cluster installations may be slow in updating compilers, libraries and in the adoption of last-generation programming paradigms like Hadoop. This advantage **is not related** to where the processing happens, and is evident in private as well as public clouds;
- Moving to an infrastructure that is based on reusable virtual machines increases initially the effort necessary to prepare and run jobs on a cloud, when compared to a more constrained cluster environment where the majority of the management effort has been already performed by system administrators. However, the relative standardization among hypervisors and the reliance on open source libraries and tools provides the opportunity for large scale collaboration among researchers and institutions, creating libraries of reusable "active components" embedded in VMs that can be reused more or less like black boxes, already adapted to a cloud environment and its properties;
- Public clouds provide a substantially neutered and abstract machine model that can have substantial differences in terms of performance when compared to the hardware that can be provisioned internally. This "equivalence paradox" requires special attention—especially with workloads that may be latency sensitive like MPI jobs, or with strict I/O requirements. While this is not a generalized observation, in some instances there may be substantial differences in the performance of virtualized instances within public clouds—both within different jobs, and even in the context of the same job—introducing a variability that must be taken into consideration when comparing execution economics. A common error, in this sense, is comparing a generic cpu/hour cost per core between an internally provisioned hardware system and an external public cloud: the comparison is in many instances biased by a factor 2 to 10.

- In the majority of submitted jobs storage is not a parameter that changes in a major way the economics of cluster versus clouds. However, the lack of tuned parallel file systems in public cloud environments can be a major factor both in terms of VM engineering and performance wise;
- While limited in adoption at the moment, higher-abstracted services like PaaS and SaaS may introduce substantial savings through the reduction of management and setup costs. An example can be found in the Hadoop-based MapReduce services offered by several commercial companies; the use of abstracted services is expected to be leveraged by newly developed applications—which means that it will still require some time before maturity.

As a general observation we can certainly identify public cloud computing as a relevant and potentially important tool for a large percentage of research jobs, especially with short temporal execution or that require substantial variability, in institutions that do have limited internal computing resources configured in the traditional cluster model. For institutions that already do have internal IT resources or that have a certain degree of flexibility in managing said resources, a partial or total conversion of traditional cluster and grids into private clouds may provide most of the same advantages that may be obtained through public clouds (namely flexibility, fast experiment turnaround, access to pre-existing and modern software libraries) along with an overall lower computational cost—even taking into account management costs and the up-front effort for creating a suitable VM environment.

The use of standards-based private cloud infrastructures may also provide the needed flexibility in case of jobs that exceed available resources through cloud bursting (the use of external public clouds for the exceeding capacity) or by pooling individual institutional private clouds into a federated cloud that can provide the necessary resources in a federated way, following the extremely successful approach already in use for GRIDs.

1 Cloud economics: a survey

1.1 Introduction to cloud computing: what cloud are we looking at?

There are few terms that are so widely used and at the same time so vaguely defined. Ian Foster with great humour pointed out that according to the dictionary one of the meaning of the word cloud is “to make obscure or indistinct; confuse” [1]. In the last few years, “cloud” has been used as an attribute of traditional ERP systems that can provide HTML output, to online storage systems, Infrastructure-as-a-service or remote terminal access services. No company or service provider seems privy to demonstrate cloud capabilities, or to be left behind this seemingly sweeping tide that is changing IT infrastructures in any market. The reality is much simpler: “cloud” is a term that defines services—of varying abstraction level—that can be provided through a network, scalable, self-service, flexible and measured [2]. There is no implicit definition of how access is mediated (despite its widespread use, HTML is not a defining part of it) but the essential point is that it provides an inherent flexibility in how the service is provisioned, consumed and commissioned. NIST recently presented a final version of their definition of what is cloud computing [3] that will be used as a basis here as well:

- **On-demand self-service:** A user can unilaterally provision computing capabilities, such as server time and network storage;
- **Broad network access:** Capabilities are available over the network and accessed through standard mechanisms;
- **Resource pooling:** The provider’s computing resources are pooled to serve multiple users using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to user demand;
- **Rapid elasticity:** Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand;
- **Measured service:** Cloud systems automatically control and measure the requested resources and their consumption.

1.2 Locality, Management and Abstraction: classifying cloud computing and current research IT

These characteristics are at the moment presented to current cloud users through three main service models of increasing abstraction and an orthogonal spectrum of locality patterns. The service models are:

- **Infrastructure as a Service (IaaS):** The capability provided to the user is to provision processing, storage, networks, and other fundamental computing resources where the user is able to deploy and run arbitrary software, which can include operating systems and applications. The user does not manage or control the underlying cloud infrastructure but only the deployed VMs.
- **Platform as a Service (PaaS)**. The capability provided to the user is to deploy onto the cloud infrastructure user-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The user does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

- **Software as a Service (SaaS):** The capability provided to the user is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g. web-based email), or a program interface. The user does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or individual application capabilities, outside of limited user-specific configuration settings.

While the locality and management patterns (also called deployment models) are:

- **Private cloud:** The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple users (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.
- **Community cloud:** The cloud infrastructure is provisioned for exclusive use by a specific community of users from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.
- **Public cloud:** The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.
- **Hybrid cloud:** The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g. cloud bursting for load balancing between clouds). From the economic point of view, hybrid clouds can be considered a linear superposition of private and public clouds.

While the definitions are generic and cover a wide range of possible combination, business-owned and managed public clouds and internally-managed private cloud are the most common combinations, and are expected to prevail in the midterm as well [4,5]. A comparable classification for current, non-cloud resources used in research environment can be derived from [6] by analysing the management model adopted for the computational resource. It is possible to identify two separate classes:

Scale class	IT resource type	# cores	Single use or limited sharing	Wide Sharing	Centrally managed
Scale 1	Desktop/Workstations	<20	✓		
	Local clusters	<100	✓		
Scale 2	Institutional clusters	>100		✓	✓
	Shared GRIDs	>1000		✓	✓

Table 1—Scale class classification of clouds

Desktop, workstations and local clusters share a similar value in terms of cost per core/hour, and share the same management properties: resources are more or less reserved for single use (or minimally shared within a small group of users), are locally managed and flexible (re-provisioning and software changes are done without the need for external approval). On the other hand, larger scale structures like clusters and GRIDs (both local and shared across institutions) tend to have a substantially lower price per core/hour, are generally reused across a larger group of users and tend to be more limited in terms of operating system, libraries and other software infrastructures. The flexibility of Scale 1 resources is balanced with a higher maintenance and support costs compared to Scale 2, despite the fact that most of these costs are usually not accounted for, being part of the day-to-day activities of the researchers managing the cluster or workstation.

In the rest of the paper, we will use only the scale 1 and 2 classification, simplifying the model and removing irrelevant details.

1.3 Cloud computing: potential benefits

Cloud computing has been heralded as a form of universal optimizer, capable of reducing cost and increasing flexibility in almost every aspect of IT. As for most technologies with seemingly magical properties, the reality is somehow less extraordinary – cloud computing is effectively capable of introducing some changes in how IT is consumed and priced, but in some cost centres the differences are limited or not significant. Most of the economic approach based on ROI, TCO, business agility [7][8][9][10] are inherently difficult to transpose to a research environment, lacking an easy-to-measure criteria like profit. Some parallels (for example between employee productivity and researcher's paper production) are inherently dangerous.

If we start from the definition of the cloud and internal, traditional computational platform, it is clear that several differences exist: among them, pricing linearity (that allows for optimal resource reservation when variable loads are required), very small time to usable service (thanks to self-provisioning and limited "queue effects") and apparently infinite resources (that gives the user the choice between time and cost—for example, allocating a large number of cores for a short period or a small number of cores for a longer time frame).

Among the most cited differences [11],[12],[13],[14] are:

Potential Benefit	Description
Economies of scale	Large public clouds enjoy economies of scale in equipment purchasing power and management efficiencies. Savings may be passed on to users, and will increasingly be so as competition in the sector increases over time.
Increased security	Thanks to the cloud provider's security effort, overall security is increased
Improved utilization & efficiency	The use of virtualisation improves efficiency and utilisation due to sharing of pooled resources and through better workload balance across multiple applications.
Increased availability	Another benefit of being based on grid computing is that applications can take advantage of a high availability of architecture that minimises or eliminates planned and unplanned downtime, improving user service levels and business continuity.
Elastic scalability	Grid computing provides public and if outsourced) private cloud with elastic scalability; that is, the ability to add and remove computing capacity on demand. This is a significant advantage for applications with a highly variable workload or unpredictable growth, or for temporary applications.
Fast deployment	Application deployment is greatly accelerated because both public and private cloud can provide self-service access to a shared pool of computing resources, and because the software and hardware components are standard, re-usable and shared.
Simpler to manage	Public clouds may require fewer IT personnel to manage and administer, update, patch, etc. Users rely on the public cloud service provider instead of an internal IT department.
No queuing effects	Cloud computing does not face the traditional delays and complexities related to job submission and scheduling

Table 2—Cited differences for cloud

To create a realistic model of cloud adoption costs and benefits we must first of all check whether the identified advantages are real, how much an effect they have in the specific world of research, and what can be done to maximize them.

1.4 Scale

The concept of “cloud economies of scale” is the commonly held idea that large and very large data centres do have a substantial economic advantage compared to small and mid-size structures, and this improved economy of scale can be transferred to the cloud end-users. Hamilton [15] demonstrates that there are substantial cost advantages moving in scale from mid-size (up to 1000 servers) to very large (more than 50000 servers) on parameters like personnel costs, storage and networking:

Technology category	Medium	Very large	Differential
Network ($ per Mbit / sec / month)	95	13	7.1x
Storage ($ per GB / month)	2.2	0.4	5.7x
Administration (Servers / admins)	140	>1000	7.1x

Table 3—Economies of scale

However, this reduction in costs is partially compensated by the substantial costs and complexity that is inherent in a flexible, large scale structure that must provide reliable services to a heterogeneity of users. As an example, while it is true that bandwidth in and out of a datacentre receives a substantial discount at scale, bringing the same bandwidth to the interfaces of each server has a cost that increases quadratically with the number of active ports, negating most of predicted economic advantages:

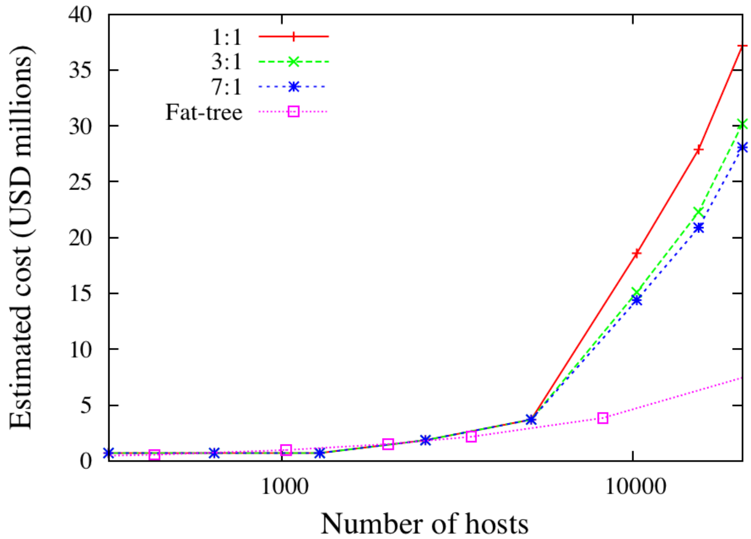

Figure 1—Bandwidth subscription rate vs.cost

Al-Fares et al. [16] in fact estimate that the cost of providing a 3:1 oversubscribed 1GB of bandwidth to 20000 servers is around 37M$, bringing the cost of the networking infrastructure in the same range of the server hardware. A 10GB infrastructure with a fat-tree topology (the most efficient up to now) would cost 690M$, making it largely unsustainable.

This is one of the reasons for the strong push towards flexible and self-managing software defined networks that can provide a degree of automatic locality between virtual machines and their physical execution environment reducing the cost and effort for providing reliable transport between machines and across the datacentre.

One of the aspects of scale that clearly persist is the procurement advantage for hardware and networking gear: the scale at which large cloud providers procure hardware allows for a substantial discount compared to smaller research infrastructures; some cloud providers even use custom assembly and components to further reduce their costs and increase energy efficiency, an option that is clearly outside of the reach of most local datacentres. Hamilton estimates that economy of scale is reached in the 45000-50000 server range, in installations that require between 12MW and 25MW [17]. The scale effect is also partially ameliorated by the delay between price reduction in the free marketplace and procurement process, that for very large scale orders is substantial. While pricing efficiency (measured as price/core, price/GB of delivered storage and so on) constantly improves both in terms of processing power and in computation per watt, the large scale procurement process introduces an inherent delay that partially reduces this effect. The reality is that most of the advances in cost efficiency that are measurable on the market are not passed to the end users; as noted by Wittman [18], the storage costs for Amazon offering has been reduced by 36% from 2006 to 2012, while hardware cost for a comparable reliable storage system decreased by an order of magnitude. A similar finding was part of the Magellan report [19]: “the cost of a typical compute offering in Amazon has fallen only 18% in the last five years with little or no change in capability…This translates into a compound annual improvement of roughly 4%. Meanwhile, the number of cores available in a typical server has increased by a factor of 6x to 12x over the same period with only modest increases in costs.”

In conclusion, for the purpose of running research applications, comparing a small/medium size datacentre (typical of most research institutions) with a very large cloud installation we can conclude that scale does have a limited impact on overall economics.

1.5 Increased security

Security is another widely debated aspect—with the popular claim that clouds provide a reduction in costs related to security and assurance. As an example, Amazon in their product literature claim that: “[A] ... direct cost for enterprises running their own data center is ensuring the confidentiality, integrity, and availability of business critical data. Examples of security costs for enterprises include capital expenditures for network security devices, security software licenses, staffing of an information security organization, costs associated with information security regulatory compliance, physical security requirements, smart cards for access control, and so on.” [46] While certainly true that cloud providers have spent a substantial effort in providing a secure infrastructure, it is important to recognize the fact that this effort does not extend to the images that are executed inside of the cloud—thus still requiring an investment in providing protection that is not substantially different between a traditional research infrastructure and the cloud, both public and private. The real difference between public clouds and traditional research infrastructures is actually the same that exists between scale 1 (self-managed, multiple images and largely self-service effort) and scale 2 (centralized, single operating system image, limited or no flexibility for installation and provisioning of personalized software and libraries). The move towards self-supplied VMs can actually decrease security: “In the current situation, i.e. without VMs provided by users or virtual organisations, the operations teams have full control over the software running on all machines. They make sure that no software with (in their eyes) unacceptable problems or … threaten the stability of the infrastructure or the integrity of the data. The currently used grid site security models rely on the assumption that machines in the infrastructure can be trusted. Introduction of VMs implies that the operations teams no longer have full control over the installed software and that they will have to reconsider the existing security models to accommodate the new and untrusted components in the infrastructure” [20].

Data management requirements vary widely between experiments, with some requiring long-term archival, while other create their datasets on the fly, and discard them after job completion. In this sense, data requirements are not substantially different between traditional research infrastructures and cloud computing.

An additional point specific to public cloud computing is the aspect of privacy and data protection, in observance of EU rules. Apart from medical, military and other privacy sensitive research, the majority of data sets pose no real risk of damage if accidentally divulged, and the risk posed by external providers is not substantially different between public and internal IT providers. The availability of public cloud resources physically located within EU borders is also sufficient to satisfy most of the legal requirements even of more stringent legislation [22],[23],[65].

1.6 Improved utilization and efficiency

A popular belief related to the economics of cloud computing is the idea that clouds are inherently more “efficient”, that is that clouds allow a higher utilization of resources. One of the reasons for this is related more to the use of virtualisation, that through hardware abstraction it reduces the inefficiencies in allocation that are common in structures with different hardware platforms with preferential access.

A common example is traditional managed hosting, where the provider has multiple offerings with different size and processor models, and the customer select one of the available model. In this approach, the hosting provider needs to have a minimum number of each server available on request, independent from the actual usage. In this scenario, the provider will necessarily incur a cost related to the percentage of servers that are not selected, and thus idle. In fact, managed server providers are among the most enthusiastic adopters of cloud computing, since it allows them to reduce this inefficiency to a minimum since the customer is not allowed to select among models—but only access a virtualized set of resources, that will be offered by the available physical machines.

In research infrastructure efficiency is substantially higher, with resources in Class 2 experiencing 80% or higher utilisation [19]; common grid software infrastructure are designed with high performance and near-optimal job schedulers that guarantee a very high use of local resources, while the use of standardized job scripts provide an abstraction mechanism that introduces efficiencies in use similar to those of virtualisation.

In large scale public clouds utilization is more complex to measure, and depends on the actual share of potentially usable physical machines. Very few measurements exist, but at least one demonstrable reference put public clouds like Amazon EC2 at utilisations of 7 to 15% [24].

1.7 Increased availability

One of the much touted advantages of cloud infrastructures is the capability of recovering from faults and in general the increased resilience thanks to the strong decoupling of services, that are then offered through a high-availability interface. For example, most cloud providers offer storage that is virtualized and abstracted through a specific interface or is directly attached through the hypervisor; in both cases, the management backplane is capable of monitoring the health and status of the storage infrastructure, and nearly instantly provide a transparent replacement in case of a physical or software failure. Most cloud systems allows as well for the automated restart of failed virtual machines, substantially reducing the downtime and the effort necessary to maintain the offered service.

It should be noted that most grid software infrastructures do offer similar capabilities, with restart of failed jobs and full fault tolerance at the file system level (in distributed file systems like Lustre, Gluster or XtreemFS); at the same time, in some instances the software deployed on the cloud may need modification or external scheduling services to be able to hide potential faults. In this sense, both private and public cloud are capable of improving somehow the reliability and resiliency of the software that is deployed on top (for an evaluation of the reliability of job submission and execution in cloud, see section 1.11).

1.8 Elastic scalability

Transparent and linear scalability is one of the most cited benefits of private clouds—the capability to add on-demand, with seemingly infinite capability and with pricing proportional to resources used, with the result that it is possible to set up an experiment at a very low upfront costs and at the same time scale resources higher or lower without apparent limits. One of the most famous examples is an experiment performed by the New York Times: "The New York Times has a large collection of high-resolution scanned images of its historical newspapers … They wanted to process this set of images into individual articles in PDF format. Using 100 EC2 instances, they were able to finish the processing in 24 hours for a total cost of $890 ($240 for EC2 computing time and $650 for S3 data transfer and storage usage, storing and transferring 4.0TB of source images and 1.5TB of output PDFs)" [25]

A more extreme example is Animoto, a provider of online photo services, that scaled in 48 hours from 50 to 3500 server [26]; such an enormous variation in scale is effectively possible in an economical way only through public cloud computing or in large scale, federated community grids.

The value for scalability is related to the concept of variability of loads: a fixed load that require no substantial variability in load or hardware requirements will be served more cheaply from a fixed, internally provisioned asset (for the explicit economic measurements, see section 2.5). A related aspect that is usually mentioned in commercial adoption of cloud computing is the shift from capital expenditure (CapEx) to operating expense (OpEx) that can be observed when moving from an internal IT infrastructure to public clouds. The pricing linearity implies that an enterprise can commit and pay for cloud resources through periodic invoicing, freeing capital that would be otherwise fixed in initially bought assets. This aspect is actually less relevant for research institutions, where provisioning is usually performed through public grants that are structured around a medium or long term project; the "pay as you go" model, usually based on credit card payments, may be even impossible to adopt for a public institution like a university; it is in fact not uncommon to find mention of cloud spending performed using the individual researcher's credit cards, to be later reimbursed (with some difficulty, given the lack of transparency) by the institution.

1.9 Fast Deployment

Another area where a difference is clearly visible is the ease and flexibility in creating a personalized environment. In [19] it is mentioned that "HPC centres typically provide a single system software stack, paired with purpose built hardware, and a set of policies for user access and prioritization. Users rely on a relatively fixed set of interfaces for interaction with the resource manager, file system, and other facility services. Many HPC use cases are well covered within this scope; for example, this environment is adapted for MPI applications that perform I/O to a parallel file system. Other use cases such as high-throughput computing and data-intensive computing may not be so well supported at HPC centres. For example, computer scientists developing low level runtime software for HPC applications have a particularly difficult time performing this work at production computing centres."

The wide adoption of class 1 computing by scientists is in fact related to the flexibility in creating a personalized, ad-hoc computing environment without the need for external resources or approval. This flexibility is in fact one of the most commonly cited reasons for use of public clouds by researchers: "The majority of grid users find the grid environment too complex. U sers might be unfamiliar with the operating system installed on the grid, or the user rights to install specific software, or they might lack technical skills. VMs provide an isolation layer between user application and this complex environment. Applications running inside a VM can be deployed without modification on a local desktop, a batch cluster, a grid site, or multiple grid sites. Consequently, an end-user starting from scratch on a VM significantly reduces the overhead of these different scaling steps." [20]

Again, the cloud does not introduce a totally different environment from the economic point of view, but more a mix between scale 1 and scale 2—the flexibility of a personal workstation, but with the resource level of large scale grids and clusters typical of scale 2.

Deployments based on VMs—when compared with a traditional Scale 2 infrastructure—show substantial reductions in the time and effort necessary to provision a new service. Using a model based on 3 stages of transition from physical to cloud we find that delivery times change substantially from one stage to the other [27]:

Stage	Average Delivery time
Legacy (rackmount); full physical deployment	6-8 weeks
Virtualized, manual image provisioning	1-3 weeks
Cloud, self-service provisioning	15-30 minutes

Table 4—Comparison of provisioning lead times

An important aspect is the conversion of existing software platform to cloud-ready images, an effort that can be substantial and involve in part a form of re-engineering both to take advantage of the inherent properties of scalability of the cloud, and to compensate for potential unavailability of resources when provisioned dynamically: "There were a number of lessons learned. The goal was to build a VM image that mirrored the STAR reconstruction and analysis workflow from an existing system at NERSC. Building such an image from scratch required extensive system administration skills and took weeks of effort. The image creator needs to be careful not to compromise the security of the VMs by leaving personal information like passwords or user names in the images… Using cloud resources was not a turnkey operation and required significant design and development efforts." [21]

On the other hand, the widespread availability of cloud infrastructure and the emergence of standardized interfaces allows for a substantial degree of reuse, both inter-experiment and intra-institution. There are already in place VM marketplaces that provide pre-built or base images on which to build a specific software installation, substantially reducing both the time to execution and the effort necessary; examples of such efforts are the EU EGI Unified Middleware Distribution or the Amazon AWS Marketplace [28]; a potential for sharing efforts in the creation and maintenance of such images can contribute substantially to reduce each individual job set-up cost, as well as increase security and reliability.

1.10 Simpler to manage

Underlying the assumptions that are inherent of cloud infrastructures is the idea of "self-service", as well as the idea of at least a partial autonomic management—the fact that individual nodes are able to assess and evaluate state, change parameters and perform actions to maintain functionality and maximize efficiency. In this sense, the cloud introduces a model that takes the flexibility of a class 1 system (autonomous management of software and libraries) along with the infrastructure and amount of resources typical of class 2 (large number of cores, storage, memory and special devices like GPUs). Some estimates place a reduction in management and administration costs between 36 and 45% between a traditional IT infrastructure and a cloud-based one [2].

The reduction in management cost is not, however, a universal factor. Some TCO studies improperly assume for public cloud computing that there is no or very limited management: "By eliminating physical infrastructure, there is no need or minimal cost to manage a server" [57] or mix the cost structure of IaaS with the abstract model of PaaS: "no need to install Tomcat, Java and J2EE environment; and no need to update software". The reality is that while there is no need for physical management (identifying and replacing non-functional parts with spares, for example) the operating system instance that is used in the public cloud needs the same effort of a similar instance installed locally.

1.11 No queuing effects

The issue of queue effects—the delay and barriers introduced by the scheduling system and the limits on usable resources—is actually not substantially different between public clouds and class 2 systems. Queues could have a negative effect, with research not being attempted or being scaled to maximise the chances of running in a reasonable time or the inability to complete work to a specified or preferred timing (for example, related to a publication or conference deadline).

The majority of grid systems are subject to a gate process that schedule requests and try to provide an optimal allocation of jobs to existing resources [47][48]. This allocation may introduce delays and sometimes even outright rejections, requiring a job resubmission to complete the requests that are not successfully completed. In public cloud systems, the large pool of available resources and the simplicity of the allocation process hides most latencies and creates the appearance of a largely "infinite" resource; this illusion however disappears when the job allocated requires the instantiation of a substantial number of cores. As an example, in [47]: "Our experience with the Amazon Web Services environment is that a variety of transient failures can occur, including an inability to access the user-data passed in during image startup, failure to properly configure the network, failure to boot properly, and other performance perturbations, including intermittent virtual machine hangs. … In general, we found that the software creating the virtual cluster cannot assume that it will always acquire all of the requested resources. Allocating 128 or more cores at once is not always practical, and results in indefinite hangs and costly idling of resources if the request cannot be fulfilled."

While the probability of successful job completion using a public cloud is higher than the equivalent probability in a grid (with some job abort rate as high as 10% [58]), a more restricted environment like a private cloud can still introduce submission limits that may break the illusion of a reliable and infinite computing resources. As mentioned in section 1.7, system administration and developers must take into account the probability that some of the resources may not be immediately available, exactly like in a traditional grid environment.

1.12 Myth and realities of cloud computing for research

We can summarize the previous points by saying that most (but not all) of the advantages of cloud computing are inherent both in the private and public models, since they depend on standardization of interfaces, self-management of the nodes and self-provisioning of resources. Some aspects, like utilisation, security and scale are not really changed in a substantial way between a traditional research infrastructure and a cloud (both private and public), despite the many claims from commercial providers; this will help us in simplifying the model that will be used in the second part of this report to evaluate the true costs of a transition to a cloud environment.

2 The economic impact of cloud adoption and use

2.1 Defining the context: varying vs. invariant variables

As mentioned in the previous section, not all the parameters that influence the economics of cloud IT are really different in a substantial way between a traditional research infrastructure and the cloud (be it private or public); in this sense, we can provide an at-a-glance model to see what changes and what is roughly equivalent, using as a reference the eGep cost evaluation framework developed in the context of the EU MODINIS programme [51]:

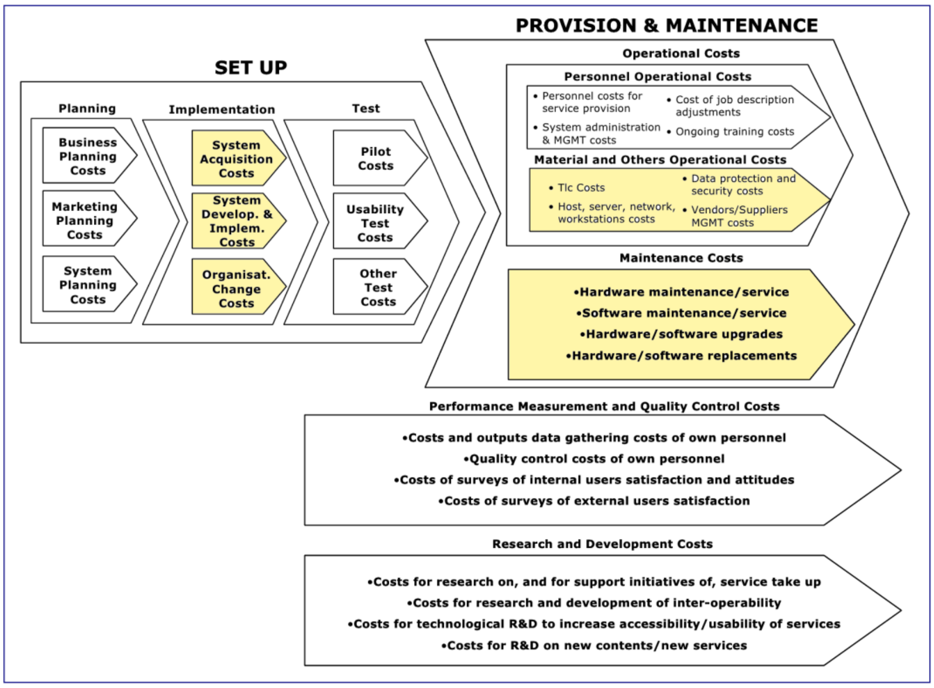

Figure 2—eGep cost evaluation framework

The yellow areas are those where it is possible to observe a substantial difference—both in terms of CapEx vs. OpEx, provisioning speed or flexibility—compared to the previously defined computing classes.

Our model will take into account the following generalized model of a computational experiment in a public research institution:

Phase	Traditional IT	Cloud
Experiment setup	Data preparation, software set-up, creation of job or task description files if necessary	Creation or instantiation of the necessary VMs
Experiment submission	Submission through standard GRAM interfaces, workflow or portals	Self-service provision of the VMs and resources necessary for the experiment execution
Queue wait/execution	Submission of the job to the scheduler; if queued, wait for the allocation of the necessary resources	Unless resource limited, the execution starts immediately and the user can observe/steer the process
Result collection	Extract result, save or delete non-relevant data files	Extract result, dispose of no longer necessary VMs, save or dispose of non-relevant data files

Table 5—generalized model of a computational experiment

For each phase, it will be evaluated whether the underlying execution model justify a substantial difference in cost or convenience, and such difference will be estimated using the available data. This approach provides a reasonable approximation of real costs and at the same time allows for a simplification and reduction in the number of parameters that need to be evaluated to assess the relative costs of a traditional versus clouds.

2.2 Set-up costs: Recurring and non-recurring

The cost for creating the necessary set-up for a computational job is extremely difficult to assess, given the extreme variability both in job size, complexity and the possibility of reuse of some components. In a traditional grid infrastructure, the job submission is based on a standardized set of execution instructions (for example, submitted through a grid portal or a standardized interface like JSDL or OGSA-BES), eventually coupled with a set of prepared applications that are delivered on-demand by the grid administrators. The tightly coupled approach (hardware, operating systems, drivers, libraries, execution engines) guarantees the highest possible performance, but at the same time is substantially restrictive; if a user requires a different operating system, or library set, the job simply can't be submitted—and usually the researcher resorts to the use of his personal workstation or a small-scale cluster, bearing the full cost of management and environment preparation along with the effort necessary to execute the planned job. In a cloud environment, the user is entitled to a set of pre-built images and user-submitted images, reducing the cost for creating an appropriate execution platform, with the trade-off of reduced potential maximum performance (see 2.4).

To evaluate the portability and the potential benefit of bringing existing code sets to the cloud, Frey and Hasselbring [52] recently introduced the idea of Cloud Environment Constraints (CECs), a set of constraints specific to the targeted cloud environment that can prevent the execution (or impede efficiency) of an application moved to a cloud environment. For example, a Java application may use types and classes not available in a constrained PaaS environment like Google's App Engine, making the application unexecutable without a substantial engineering effort. A simplified version of CloudMig hierarchy will help in:

- L0 Cloud incompatible: The application cannot be executed at all.
- L1 Cloud compatible: The application can be executed.
- L2 Cloud aligned: The execution context, utilized cloud services, or the migrated software system itself were configured to achieve an improved efficiency in resource use or scalability without pervasively modifying the software system.
- L3 Cloud optimized: The software system was pervasively modified to enable automated exploitation of the cloud's elasticity, for example through recoding to increase the level of parallelization. An evaluation was conducted to identify system parts which would experience an overall benefit from substitution or supplement with offered cloud services. These substitutions and supplements were performed.

If an application is in the L0 class, it will simply be unable to run in the targeted cloud environment; in this case, no economic consideration can be advanced—the user is restricted to the legacy IT environment until the application is not redesigned. An application that is redesigned or adapted to move from L1 (barely running) to L2 or L3 receives an additional advantage in terms of ease and speed of setup, increased efficiency, scaling abilities that—when reused across experiments—can reduce substantially both the recurring costs and the cloud utilization.

While legacy applications ported from a traditional grid environment would probably fit in class L1 (or even in L0, if they need specific hardware features that are not properly abstracted or virtualized by the cloud orchestrator), a new breed of tools and programming paradigm are more or less designed to be executed in a cloud environment; for example, the popular MapReduce model (used in the Apache Hadoop environment, among others) is already available in prebuilt VM images that are ready to be executed in a cloud environment, with full scaling and fault tolerance. Applications that are more cloud-oriented will require little or no adjustment in the pre-execution phase and will adapt and extract the maximum efficiency from the cloud infrastructure in the execution phase. In this case, if an application is planned to be reused substantially, it may be sensible to invest in bringing it to L2 or L3 level, a trade-off between one-time redevelopment or re-engineering costs and the reduction or elimination of recurring costs [61].

The cost necessary to adapt an application is highly dependent on its internal structure. The most common programming models are:

Name	Scenario	Coupling
Task Model	Independent bag of tasks applications	
Thread Model	Multi-threaded applications	Partial/High
MapReduce Model	Data intensive applications	
PSM	Parameter sweeping applications	
Workflow	Workflow applications	
MPI	Message passing applications	High
Actors	Distributed active objects / agents	Partial/High

Table 6—Common programming models

Applications with limited coupling (limited need for high-speed, low latency interprocess communication) will be simpler and easier to adapt to a cloud environment, and are thus ideal candidates for such an effort. For some models (MapReduce, Workflow, Task model) it is possible to reuse existing software platforms, already cloud-enabled, thus further reducing the translation cost.

On the contrary some code classes - like MPI - require a substantial low-latency communication capability and specific hardware or software features like RDMA that may not be exposed or efficient enough in cloud toolkits. In this case, the application will be necessarily restricted to a non-cloud infrastructure, or a specific porting activity will be necessary (usually at substantial cost, and losing the advantage of code maturity), These applications will run in emulation on a cloud—but at such a substantial performance loss that the execution itself may not be economical or feasible (some examples will be presented in section 2.4).

2.3 2.3 PaaS/SaaS

A substantial difference in the experiment setup phase can be observed when instead of the more commonly used IaaS—where an entire machine is virtualized and used for the implementation of a specific task—the experimenter adopts a higher-level abstraction in the form of PaaS or SaaS. Examples of such applications are Cornell's RedCloud [53] and their Matlab-as-a-Service, the many Hadoop/MapReduce services available from commercial providers like Amazon or the SaaS services like BLAST offered by Yahoo or Microsoft.

The use of a higher-order abstraction substantially reduces the cost and effort of creating, managing and orchestrating a set of VMs to provide the needed service, thus reducing both the costs and the complexity of the task. In fact, the area exposed to the need of management by the end user is substantially smaller in PaaS compared to IaaS, and even more so in SaaS (accompanied by an equivalent loss of control):

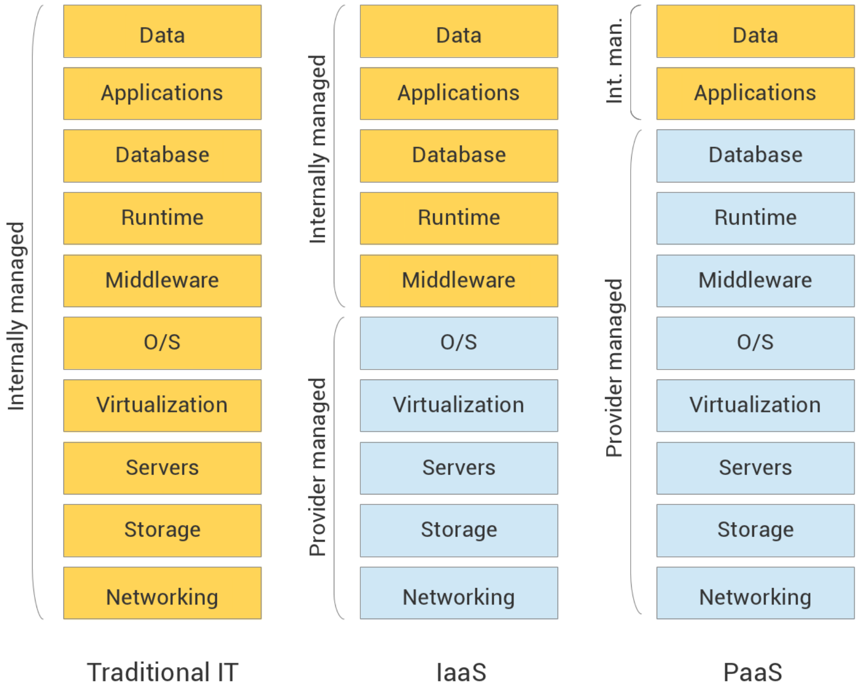

Figure 3—Traditional IT vs. IaaS vs. PaaS

Submitting a task to a PaaS service require limited knowledge of the underlying computational infrastructure and automates several aspects like scaling, resource provisioning and error management. However the user at the same time has a reduced visibility and control, limiting for example what version of a service or library may be used, or forcing a specific database choice; the use of a PaaS also severely limit the possibility to perform application or platform-specific tuning. A potential workaround is the execution of a PaaS layer on top of an internal IaaS private cloud – for example CloudFoundry [54] or OpenShift [55], providing an opportunity for direct tuning and adaptation to the internal hardware properties of the private cloud.

2.4 Measuring costs: estimating resource equivalence

An important point is related to the poorly defined concept of "core". While perfectly defined in terms of parallelism, it does not convey enough information on performance—with the result that it may be not appropriate to compare the processing power of an internal infrastructure with that of a virtualized public cloud.

The issue has been widely researched before; [19] provide a set of benchmark and comparisons between scientific applications executed within the Amazon public cloud and a traditional grid/cluster infrastructure, with results that show how both the virtualization layer and the less performing networking infrastructure impact negatively on several scientific codes[1]:

[1] On pure benchmarks (like HPCC, VASP or IMB) the slowdown was even more sensible – from 10 to 22 times.

Code	Algorithm	Slowdown
CAM	Navier Stokes CFD	3.05
MILC	Conjugate Gradient, sparse matrix; FFT	2.83
IMPACT-T	PIC, FFT component	4.55
MAESTRO	Block structured grid multiphysics	5.75

Table 7—Performance impact

The recent introduction of HPC-specific computing models from Amazon is a welcome addition, but does still not completely cover the gap with a dedicated cluster. The 10Gb networking layer provided still has substantially higher latencies compared to dedicated networks like Myrinet or Infiniband; the lack of in-band parallel file systems that can be tuned to the specific file access pattern is another factor that introduces a performance loss, especially with file sets larger than 1TB [19]. The impact of virtualization on the performance of HPC codes has been widely studied [[29][30][31][32][33] and the research shows that tuning the parameters of the hypervisor can substantially increase the efficiency of the execution, even if the code itself is tuned for bare metal (for example, taking for granted the exclusive access to the processor cache, and its size). The effect of contention and the latency introduced by the hypervisor reduces the performance of public cloud instances, without the possibility to reduce—through tuning and scheduling—the negative effects of virtualization. The slowdown is more visible in codes that require very low latency, like MPI applications, or where the contention between VMs introduces cache pollution; other codes (MapReduce, task model) will suffer a much more limited slowdown. To compensate in the following economic estimation for the difference in performance between an external cloud like Amazon, we have assumed that an EC2 core has an overall slowdown of 2, while the execution of codes on a private cloud incurs a slowdown of 1.25 since the hypervisor is under direct control of the cloud administrators, and thus it is possible to tune the VM execution parameters to reduce the overhead introduced by the hypervisor [72],[73],[74].

At the moment there is no large scale private cloud software toolkit that directly support the low-latency and direct I/O virtualization necessary for HPC; this is among the necessary improvements that need to be implemented to be able to effectively leverage the current cluster-level hardware within a cloud computing framework.

2.5 A framework for cost estimation in research institutions

There is no shortage of research on the economics of computing and datacentres, and this helps us in identifying some common estimates of both fixed and varying costs; we will start by considering the generalized Class 2 environment, assumed to be composed of rack mount servers, hosted in a managed environment for which space, power, cooling, networking and management needs to be provided with continuity. The cost centres for such a configuration are:

- Servers (hardware), with an average 3yr depreciation period
- Networking Equipment, with average 4yr depreciation period
- Power distribution and cooling
- Power
- Other infrastructure costs

Several estimates of the relative relevance of each category have been presented in the past—among them, a survey by Amazon's James Hamilton [17]:

- Servers: 57%
- Networking equipment: 8%
- Power distribution and cooling: 18%
- Power: 13%
- Other infrastructure: 4%

The data is in line with alternative surveys like that of Greenberg [44] that places server costs at 45% and power related costs at 15%.

It is clear that hardware (servers and network equipment) dominate with an aggregate cost of 65%, followed by power-related costs (31%). This is in line with the findings from Patel and Shah [43] that identified a cost per rack (with a 10KW consumption) at a price per month adjusted for inflation of around $13,000/month, perfectly in line with the cost for the CINN hardware mentioned in the JISC report [7] of $13,555. Management and maintenance costs are limited—if we assume a full-time employee managing 140 servers or around 3 full racks of server as per [17], we can infer that the addition of full management for our infrastructure increases costs by between $900 and $1,100—less than 10% of our estimated monthly cost.

Power consumption is the second largest cost centre—both as direct server consumption and lost in cooling and power distribution. A substantial effort is being done to implement “green” schedulers, that for example reduce the number of active nodes that are not required, or through the individual server cpu-throttling capabilities; the amount of power saved using these last techniques is however limited, since the power consumption when a server is not loaded (or with negligible load) is between 50% and 80% of full-load consumption[41][42]. Assuming a nearly ideal capability to partition servers in fully loaded and null-loaded and an average utilization of 50%, a green scheduler (shutting down unneeded servers) would save around 15% of costs, while server power throttling only around 5% [62][63].

By taking the estimate of $13,000 per month for a 10KW rack, we can estimate the cost per server and per core assuming the use of off-the-shelf hardware and networking gear. Assuming an average consumption of 425W [42] per server we can host around 280 core per rack within the allocated power consumption, obtaining a cost per core of 0.06$/core-hour for a fully managed infrastructure.

We can assume this as the real cost for a small size installation, between 1 and 5 racks; we can now compare our measurement with the data obtained from other sources, assuming a cost scaled to 100% utilization—so that we can later have a baseline number to scale back for lower utilization percentages. For data referenced in the literature calculated at a different utilization, cost at 100% was obtained through a linear approximation.

System	$/Core-hour, 100% utilization	$/Core-hour, 75% utilization
Hopper [19]	$0.018	$0.025
Montero-Llorente [31]	$0.04	$0.05
Magellan (overall) [19]	$0.04	$0.053
Class 1 server/workstation [7]	$0.046	$0.06
Cornell RedCloud [53]	$0.058	$0.078
Our estimate	$0.06	$0.085
Amazon cc1.4xl, resv. instance[2]	$0.062	$0.086
Amazon cc1.4xl	$0.093	$0.12
CINN [7]	$0.15	$0.18

Table 8—Cost per core

Amazon does offer two alternatives to the linear pricing model: the "reserved instance", where the payment of an initial one-time fee allows for a substantial discount on the price of consumed resources, and the "spot market", where the user on unused Amazon EC2 resources—and is allowed to run those instances for as long as their bid exceeds the current Spot Price. The table lists an estimated pricing for a reserved instance, but not that of spot instances, given the large variability and the lack of guarantees offered. A parameter that was not included in the table is the added cost related to the quantization of billing—most cloud provider use the "billable hour" abstraction, with partial instance-hours consumed billed as full hours. This added cost, specific to public clouds, can be estimated by modelling the start and stop of an EC2 instance as uniformly distributed within the hour, and in general can be expected to add the cost equivalent to an instance-hour for each activated instance. For the majority of jobs this is not really significant; for massively parallel jobs that start and stop in a short period, a simple time-based scheduler may reduce the effect by aligning the instance activation with the first seconds of the next Amazon billable hour.

There are two clear outliers—Hopper, a 153216 compute cores, 217 TB memory cluster that can be considered an example of "very large scale system"—thus reaching the point, already mentioned in 1.4, where scale substantially changes the per-core economics, and the CINN cluster—a relatively small but modern system with Infiniband, GPGPU processing and solid state disks, that has a slightly higher cost per core-hour compared with our estimates, and should be considered the high mark for the overall cost estimates.

[2] An evaluation of pricing of public cloud providers has been performed on a basis of the excellent work by Hawtin [7], adjusting pricing to may 2012. Amazon remains the only cloud provider with HPC-specific nodes, and given the competitive pricing was chosen as the benchmark for public cloud costs.

The estimates are based on pure core costs—not taking into account the performance gap mentioned in 2.4. If we consider a (very conservative) slowdown of 2 compared to a tuned private cloud slowdown of 1.25, we end up with the equivalent, adjusted cost per cpu-hour:

System	**Adj. $/Core-hour, 100% ut**	**Adj. $/Core-hour, 75% ut**
Hopper [19]	$0.023	$0.031
Montero-Llorente [31]	$0.05	$0.063
Magellan (overall) [19]	$0.05	$0.066
Class 1 server/workstation [7]	$0.058	$0.075
Cornell RedCloud [53]	$0.072	$0.098
Our estimate	$0.075	$0.11
Amazon cc1.4xl, resv. instance	$0.12	$0.172
Amazon cc1.4xl	$0.19	$0.25
CINN [7]	$0.19	$0.23

Table 9—Cost per core (adjusted)

For the data cost, we can estimate the average amount of data stored permanently or near-permanently using data from the Magellan survey that found that more than 50% of submitted jobs required a storage pool bigger than one TB, with other two classes (10 to 100GB and less than 10GB) around 18% of jobs, with the remaining 10% require around 1TB of data, and the remaining 10% requiring storage of less than 1GB. A rough estimate of costs put the average storage utilization between 1 and 2TB per experiment.

[19] reports that of all submitted jobs, 48% was in the order of hours, 14% in the order of minutes and roughly the same amount was in the order of days (the remainder was marked as “highly variable”). Using the available data and assuming the use of an external cloud provider like Amazon, we can infer that the time for which data needs to be maintained for the job execution is around 9 hours, with a cost per job of roughly $2 assuming the use of Amazon's EBS at $0.10 per GB-Month (for comparison, an internally provided storage services is $0.007 per GB-Month, one order of magnitude less than EBS). For the average job execution the storage costs is thus negligible; transaction cost (both into and out cloud) are equally limited.

Utilization is among the single most important parameters—increasing utilization clearly improves the economics of a private cloud infrastructure. While HPC clusters average extremely high utilization rates (around 85% in many instances), private clouds may have lower rates due to the need to migrate jobs to run in VMs or for performance or convenience reasons. Approaches to increase this rate are already appearing in literature; among them the use of a backfill scheduler that takes advantage of unused cloud capacity to offload jobs from a traditional HPC cluster [64]; an approach that can push utilization to nearly 100%, with a reduced impact on performance (less than 7%).

In institution where utilization is low (or is predicted to be low), a potential solution may be the adoption of an hybrid cloud; that is, maintaining a smaller, private cloud that can send jobs to a public cloud for peak demand. The need to transfer data back and forth may hamper this approach; transferring 1TB of data through a public network may require several hours—and given the fact that a large number of computation are planned for a limited execution time, this mixed approach may be unsustainable. A better approach may be the creation of community clouds across institutions; by pooling resources and users, it is possible to raise utilization without requiring additional investments or wasting over-provisioned resources. In this sense, the standards adopted and used by the cloud infrastructure become extremely important, and can become a bottleneck if not properly chosen. As mentioned in 1.4, private clouds can adopt new hardware much faster than large scale public cloud providers; as an example, most new generation servers do employ SSD drives for storage tiering, an offer that is still unavailable in Amazon (it has been however recently announced by other cloud providers like CloudSigma). In this sense, given the rapid progress in computational efficiency in commercial servers, the cost ratio between private clouds and public clouds will probably increase, lowering the cost per core-hour in private clouds built with more recent hardware.

2.6 Summary of main findings

It is clear that for a substantial range of applications, cloud (both private and public) provide a compelling proposition even outside of the pure cost per computation. The self-service capabilities, the reuse of virtual machines that contain full infrastructure that are under the complete user control—there are several advantages that can be only partially captured in an economic analysis. The transition towards virtualization and cloud is foremost a change in how computational resources are accessed and managed, and this change will require not only new technology but a new mindset as well—the idea that management of the infrastructure will largely disappear in the background, leaving the researcher with a set of active black boxes with which to compose new experiments.

		Scale 1	Scale 2	Public Clouds	Private Clouds
Fixed	Structural	0.06-0.1$/cpuh	0.05-0.08 $/cpuh	0	Same as Scale2
Per job	Services	0	0	0.12-0.19$/cpuh	0
	Data related	negligible	0.007-0.01$/GB	0.1$/GB/month	Same as Scale2
	Devel. costs	Moderate (1)	High (2)	Low/very low (3)	Low/very low (3)
	Sched. constraint	None	Moderate to high	None	Depends on resources

Table 10—Costs summary

(1) Management and creation of the development environment + software development.
(2) Constrained environment, single OS, limited flexibility in setting up alternative environments.
(3) After creation of a set of common VM environments

2.7 Evolution of clouds

Cloud computing is rapidly evolving: from IaaS initial predominance the market is moving towards more abstract services like PaaS and SaaS. In this sense the research landscape is structurally different from the traditional business environment; apart from IaaS (private and public) there is a substantial interest in public SaaS, mainly centered on the Hadoop ecosystem with offerings from many commercial actors like Microsoft, Yahoo, Amazon and IBM.

While it's clear the role of private and public IaaS, and the growing role of pure public offerings devoted to SaaS, a possible future evolution will probably increase the role and importance of PaaS offerings, both public and private, thanks in large part to the growing importance of open source platforms, especially when explicit efforts for integration between the IaaS and PaaS will allow for a degree of autonomic management—freeing the user from the burden of allocating, checking and planning in advance the allocation of resources for a specific job execution.

PaaS oriented towards scientific applications may provide a substantial reduction in the cost of developing new applications; leveraging existing scientific codes as libraries or toolboxes, in a form similar to existing environments like R or Matlab. Some combinations will probably remain less used in the future: for example, hybrid SaaS and PaaS face substantial hurdles in the synchronization and exchange of data between the private and public clouds, leading to a “data gulf” that requires substantial effort and research to be overcome—for example, new partitioning schemes that can take advantage of distributed filesystems across private and public clouds.

As for standards, the IaaS landscape is at the moment dominated by de-facto standards like Amazon's API. One of the main selling point of several private cloud toolkits is the capability to operate like EC2, and transfer VMs from public to private and back. It is not clear, however, whether using an interface that is based on a cloud infrastructure with goals and purposes that are not clearly oriented towards research is really worthwhile; at the same time, implementing a standard that is controlled by a single entity may create a form of lock-in. Several other standardization efforts exist, like OCCI (Open Cloud Computing Interface) or the Open Virtualization Format (OVF), and thanks to the fact that the majority of private clouds are currently based on open source software toolkits like OpenNebula [66], OpenStack [67] or CloudStack [68], these are actually becoming a substantial main driver of adoption for these open standards. Some can act as “brokers” and mediate between different cloud infrastructures, while others can directly connect EC2 as a remote resource. However, there is no current open source infrastructure that implements the full Amazon service APIs, and given its impressive rate of evolution and expansion it will be probably not even imaginable to have a complete, fully independent reimplementation of the core services like EC2 or S3. It is advisable to evaluate the current platforms, assess the degree of support for both Amazon's standard and true open standards to provide guidance on APIs that can be used in a way that does not force lock-in towards a single vendor.

The adoption of interoperable open source platforms does have the additional advantage of facilitating the exchange of reusable components and VMs between institutions, to share the cost of developing and maintaining commonly used images, in a way similar to what has been done with science-oriented Linux distributions, like Scientific Linux or Rocks [59].

3 Bibliography

[1] Foster, I. ‘Cloud, Grid, what's in a name?’ - available at http://ianfoster.typepad.com/ blog/2008/08/cloud-grid-what.html Retrieved 15/4/2012

[2] M. Chetty, R. Buyya, “Weaving computational grids: how analogous are they with electrical grids?” Computing in Science and Engineering 4 (4), 2002

[3] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” National Institute of Standards and Technology, vol. 53, no. 6, p. 50, 2009.

[4] A. Perilli, “Building Private Clouds : Why , How , When and Who,” Gartner Report, 2012

[5] J. Staten, L. E. Nelson, O. Glenn, and N. M. Hayes, “Market Overview : Private Cloud solutions, Q1 2011,” Forrester report, 2011

[6] J. Hammond “Developer Trends and the Cloud,” Forrester report, 2010.

[7] Hawtin, R., Hammond, M. “Cost analysis of cloud computing for research” JISC report, 2012

[8] B. C. Tak, “To Move or Not to Move : The Economics of Cloud Computing,” Proceedings of HotCloud11, 2011.

[9] R. Harms and M. Yamartino, “The economics of the cloud,” Technology, November, pp. 2004-2004, 2010.

[10] Klems, M., Nimis, J. and Tai, S. 2009. “Do Clouds Compute? A Framework for Estimating the Value of Cloud Computing” Designing E-Business Systems. Markets, Services, and Networks, Lecture Notes in Business Information Processing, 22, p. 110.

[11] T. Alford and G. Morton, “The Economics of Cloud Computing Analyzed,” Cloud C0mputing Journal, pp. 1-10, 2011.

[12] C. Gong, J. Liu, Q. Zhang, H. Chen, and Z. Gong, “The Characteristics of Cloud Computing,” 2010 39th International Conference on Parallel Processing Workshops, pp. 275-279, 2010.

[13] B. Armbrust et al., “A view of cloud computing,” Communications of the ACM, vol. 53, no. 4, pp. 50-58, 2010.

[14] S. P. Mirashe and N. V. Kalyankar, “Cloud Computing,” Communications of the ACM, vol. 51, no. 7, p. 9, 2010.

[15] Hamilton J. “Cloud computing economies of scale” Keynote, Self-Managing Database Systems, Shanghai 2009

[16] M. Al-Fares, A. Loukissas, and A. Vahdat, “A scalable, commodity data center network architecture,” Proceedings of the ACM SIGCOMM 2008 conference on Data communication - SIGCOMM ’08, p. 63, 2008.

[17]James Hamilton, Overall Data Center costs. http://perspectives.mvdirona.com/2010/09/18/OverallDatacentreCosts.aspx

[18] Wittman, A. “Does the cloud keep pace with Moore's law?” InformationWeek, http://www.informationweek.com/news/cloud-computing/infrastructure/232601509 retrieved 8/5/2012

[19] K. Yelick et al., “The Magellan Report on Cloud Computing for Science The Magellan Report on Cloud Computing for Science,” US Department of Energy (DOE), 2011.

[20] BiG GRID dutch e-science grid, “Virtualization of worker nodes working group report”, available at https://wiki.nbic.nl/images/f/f6/ProgressReport-1.0.pdf retrieved 15/5/2012

[21] P. V. Beek, M. V. Driel, S. Klous, R. Starink, and R. Trompert, “BiG Grid Virtualization of worker nodes Working group progress report Table of Contents,” pp. 1-35.

[22] Mckinsey report, clearing the air on cloud computing. http://images.cxotoday.com/cxoimages/storyimages/matter101157.pdf

[23] Jeffrey K., Neidecker-Lutz B., “the future of cloud computing ” Expert Group Report on Cloud Computing, European Commission, 2010

[24] H. Liu, A. T. Labs, and S. Jose, “A Measurement Study of Server Utilization in Public Clouds.” 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC)

[25] Hilley D., “Cloud Computing: A Taxonomy of Platform and Infrastructure-level Offerings ”.CERCS Technical Report , Georgia Institute of Technology 2009

[26] Werner Vogels. A Head in the Cloud: The Power of Infrastructure as a Service. Talk at Stanford

University for Stanford EE Computer Systems Colloquium (EE380), April 2008.

[27] K. Schroeder, “CITEIS Gen2 – Cisco Internal Private Cloud,” 2011, http://www.slideshare.net/Ciscodatacentre/citeis-gen2-ciscos-internal-private-cloud (retrieved 21 april 2012)

[28] Amazon AWS Marketplace,

https://aws.amazon.com/marketplace

[29] M. D. D. Assunção and A. Costanzo, “Evaluating the cost-benefit of using cloud computing to extend the capacity of clusters,” Proceedings of the 18th ACM international symposium on High performance distributed computing HPDC 09, vol. 256, no. 3, pp. 141-150, 2009.

[30] C. Vecchiola, S. Pandey, and R. Buyya, “High-Performance Cloud Computing: A View of Scientific Applications,” 2009 10th International Symposium on Pervasive Systems, Algorithms, and Networks, pp. 4-16, 2009.

[31] R. Moreno-vozmediano, R. S. Montero, and I. M. Llorente, “Supplemental Material : Multi-Cloud Deployment of Computing Clusters for Loosely-Coupled MTC Applications .,” vol. 22, no. 6. IEEE, 2011, pp. 924-930.

[32] A. Iosup, S. Ostermann, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema, Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing, vol. 22, no. 6. IEEE, 2011, pp. 931-945.

[33] S. Ostermann, A. Iosup, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema, “An Early Performance Analysis of Cloud Computing Services for Scientific Computing,” Computing, vol. 22, no. December, pp. 115-131, 2010.

[34] Anhalt, F., Primet P.V., “Analysis and Experimental Evaluation of Network Data-Plane Virtualization Mechanisms” International Journal on Advances in Intelligent Systems, vol 2 no 4, 2009,

[35] I. Foster, Y. Zhao, I. Raicu, and S. Lu, “Cloud Computing and Grid Computing 360-Degree Compared,” 2008 Grid Computing Environments Workshop, vol. abs/0901.0, no. 5, pp. 1-10, 2008.

[36] C. Architecture, “Cloud Computing Cloud Infrastructure,” Architecture, vol. 3, no. 10, 2010.

[37] A. Khajeh-Hosseini, I. Sommerville, J. Bogaerts, and P. Teregowda, “Decision Support Tools for Cloud Migration in the Enterprise,” 2011 IEEE 4th International Conference on Cloud Computing, pp. 541-548, Jul. 2011.

[38] A. Khajeh-Hosseini, I. Sommerville, J. Bogaerts, and P. Teregowda, “Decision Support Tools for Cloud Migration in the Enterprise,” 2011 IEEE 4th International Conference on Cloud Computing, pp. 541-548, Jul. 2011.

[39] M. Woitaszek and H. M. Tufo, “Developing a Cloud Computing Charging Model for High-Performance Computing Resources,” 2010 10th IEEE International Conference on Computer and Information Technology, no. Cit, pp. 210-217, 2010.

[40] Misra S.C., Mondal A. “Identification of a company's suitability for the adoption of cloud computing and modelling its corresponding Return on Investment” Mathematical and Computer modelling, 2010

[41] Newcombe, L. “Data centre Fixed to Variable Energy Ratio metric DC-FVER”. BCS chartered institute for IT, 2012

[42] Chheda R., Shookowsky D., Stefanovich S., Toscano J. “Profiling Energy Usage for Efficient Consumption” The architecture Journal, Jan 2009

[43] C. D. Patel and A. J. Shah, “Cost Model for Planning, Development and Operation of a Data Center,” vol. 107, pp. 1-36, 2005.

[44] A. Greenberg, J., D. A. Maltz, and P. Patel, “The Cost of a Cloud : Research Problems in Data Center Networks.”

[45] R. Moreno-Vozmediano, R. S. Montero, and I. M. Llorente, “Multicloud Deployment of Computing Clusters for Loosely Coupled MTC Applications”, vol. 22, no. 6. IEEE, 2011, pp. 924-930.

[46] Amazon, “The economics of AWS cloud vs. owned IT infrastructure”, http://aws.amazon.com/whitepapers/ 2009

[47] K. R. Jackson et al., “Performance Analysis of High Performance Computing Applications on the Amazon Web Services Cloud,” 2010 IEEE Second International Conference on Cloud Computing Technology and Science, pp. 159-168, 2010.

[48] I. Foster, Y. Zhao, I. Raicu, and S. Lu, “Cloud Computing and Grid Computing 360-Degree Compared,” 2008 Grid Computing Environments Workshop, vol. abs/0901.0, no. 5, pp. 1-10, 2008.

[49] X. W. Street and X. Chen, “The Comparison Between Cloud Computing and Grid Computing,” System, vol. 11, no. 46, pp. 72-75, 2010.

[50] D. Ghoshal, R. S. Canon, and L. Ramakrishnan, “I/O performance of virtualized cloud environments,” Proceedings of the second international workshop on Data intensive computing in the clouds - DataCloud-SC ’11, p. 71, 2011.2-75, 2010.

[51] European Commission, “eGovernment Economics Project (eGEP) Expenditure Study Final Version,” 2006.

[52] Frey S., Hasselbring W. “The CloudMIG Approach: Model-Based Migration of Software Systems to Cloud-Optimized Applications ”, International Journal on Advances in Software, vol 4, 2011

[53] RedCloud – Cornell University Center for Advanced Computing website: http://www.cac.cornell.edu/RedCloud/ (retrieved 28 may 2012)

[54] CloudFoundry project. www.cloudfoundry.com/about

[55] Openshift project. https://openshift.redhat.com/app/

[56] Hammond, M., Hawtin, R. “Cloud computing for research final report” JISC report, 2010

[57] Han Y, “Cloud Computing: Case Studies and Total Costs of

Ownership ”.Information technology and libraries, December 2011

[58] Berten V., Jeannot E., “Modeling Resubmission in Unreliable Grids: the Bottom-Up Approach”. In Euro-Par 2009, Parallel Processing - Workshops, volume 6043 of LNCS, pages 150– 161, Delft, The Netherlands, August 2009

[59] P. M. Papadopoulos, “Extending clusters to Amazon EC2 using the Rocks toolkit,” International Journal of High Performance Computing Applications, vol. 25, no. 3, pp. 317-327, 2011.

[60] E. Deelman, G. Singh, M. Livny, B. Berriman, and J. Good, “The cost of doing science on the cloud: The Montage example,” 2008 SC International Conference for High Performance Computing Networking Storage and Analysis, vol. C, no. November, pp. 1-12, 2008.

[61] S. Patidar, D. Rane, and P. Jain, “Challenges of software development on cloud platform”. IEEE, 2011, pp. 1009-1013.

[62]Barroso, L. A. and Hölzle, U. “The Datacentre as a Computer: An Introduction to the Design of Warehouse-Scale Machines”. Morgan & Claypool Publishers. 2009

[63] L. Newcombe, P. Latham, and V. Smith, “Data centre Fixed to Variable Energy Ratio metric An alternative to useful work metrics which consumption.”

[64] P. Marshall, K. Keahey, and T. Freeman, “Improving Utilization of Infrastructure Clouds,” 2011 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp. 205-214, May 2011.

[65] Jaeger, P., Lin, J., Grimes, J. and Simmons, S. 2009. Where is the cloud? Geography, economics, environment, and jurisdiction in cloud computing. First Monday, 14, 5.

[66] OpenNebula: Sotomayor, B; Montero, RS; Llorente, IM & Foster, I. “An Open Source Solution for Virtual Infrastructure Management in Private and Hybrid Clouds”. IEEE Internet Computing, Special Issue on Cloud Computing, October 2009

[67] The Openstack project. http://openstack.org/

[68] The CloudStack project http://cloudstack.org/

[69] C. N. Höfer and G. Karagiannis, “Cloud computing services: taxonomy and comparison,” Journal of Internet Services and Applications, vol. 2, no. 2, pp. 81-94, 2011.

[70] Z. Hill and M. Humphrey, “A quantitative analysis of high performance computing with Amazon’s EC2 infrastructure: The death of the local cluster?,” 2009 10th IEEEACM International Conference on Grid Computing, pp. 26-33, 2009.

[71] M. Böhm, S. Leimeister, C. Riedl, and H. Krcmar, “Cloud Computing and Computing Evolution,” in Cloud Computing Technologies Business Models Opportunities and Challenges, S. Murugesan, Ed. CRC Press, 2010, pp. 1-28.

[72] Praveen G., “Analysis of Performance in the Virtual Machines Environment”, International Journal of Advanced Science and Technology Vol. 32, 2011

[73] Luszczek, P. Meek E., Moore S., Terpstra D., Weaver V.M., Dongarra J. “Evaluation of the HPC Challenge Benchmarks in Virtualized Environments” 6th Workshop on Virtualization in High-Performance Cloud Computing, Bordeaux,, 2011.

[74] Tikotekar A, Vallee G., Naughton T., Ong H., Engelmann C., Scott S., “An Analysis of HPC Benchmarks in Virtual Machine

Environments” Euro-Par 2008 Workshops - Parallel Processing

[75] M. Armbrust, A. Fox, R. Griffith, Anthony D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, M. Zaharia, “Above the clouds: a berkeley view of cloud computing”, Technical report, 2009.

[76] P. Brebner and A. Liu, “Performance and Cost Assessment of Cloud Services,” in Service-Oriented Computing, ser. Lecture Notes in Computer Science, E. Maximilien, G. Rossi, S.-T. Yuan, H. Ludwig, and M. Fantinato, Eds. Springer Berlin/ Heidelberg, 2011, vol. 6568, pp. 39–50.

[77] Klems, M., Nimis, J. and Tai, S. 2009. “Do Clouds Compute? A Framework for Estimating the Value of Cloud Computing” Designing E-Business Systems. Markets, Services, and Networks, Lecture Notes in Business Information Processing, 22, p. 110.

[78] Delgado V. “Exploring the limits of cloud computing ” Master of Science Thesis , KTH University Stockholm, Sweden 2010