Computing (2011) 91:75–91
DOI 10.1007/s00607-010-0120-1

Cloud computing for small research groups in computational science and engineering: current status and outlook

Hong-Linh Truong · Schahram Dustdar

Received: 15 February 2010 / Accepted: 3 September 2010 / Published online: 25 September 2010

Abstract Cloud computing could offer good business models for small computational science and engineering (CSE) research groups because these groups often do not have enough human resources and knowledge to manage the complexity of computational and data infrastructure for their research, while cloud computing aims to eliminate that complexity from the user. In this paper, we have analyzed current status of supporting tools for small CSE groups to utilize cloud computing. Although cloud computing is perceived as an interesting model, we have identified several issues that prevent a wide adoption of cloud computing from small CSE research groups. We also present recommendations for addressing these issues in order to attract small CSE groups to utilize cloud computing.

Keywords Cloud computing · Computational science and engineering · Programming support · Infrastructure-as-a-service · Platform-as-a-service · Software-as-a-service

Mathematics Subject Classification (2000) 68-02 · 68U01 · 68M14

1 Introduction

Computational science and engineering (CSE) involves several fields, such as computer science, applied mathematics, bioinformatics, biomechanics, meteorology, and

---

Communicated by C.H. Cap.

H.-L. Truong (✉) · S. Dustdar
Distributed Systems Group, Vienna University of Technology,
Argentinierstrasse 8/184-1, 1040 Vienna, Austria
e-mail: truong@infosys.tuwien.ac.at

S. Dustdar
e-mail: dustdar@infosys.tuwien.ac.at

computational material sciences [47]. In many cases, CSE research requires a strong knowledge about and infrastructure of high performance computing and large-scale data storage systems. Cloud computing is relevant to CSE research groups, in particular those with a small number of people and with limited knowledge about high performance computing and data systems. As pointed out in [8,11,22,25], cloud computing refers to both hardware, software and their infrastructure delivered as services over the Internet. In the literature, three main classes of cloud computing providers are currently recognized: IaaS (Infrastructure as a Service) which provides only machines and storage systems for any purpose, PaaS (Platform as a Service) which provides platforms for developing applications, and SaaS (Software as a Service) which provides available software which can be used as or be composed for applications [37,59]. These classes of cloud providers could potentially offer solutions to meet different needs of CSE groups. However, we need to understand how current cloud computing offers and promises are relevant to small CSE research groups, whether small CSE research groups can utilize these offers or not, and what support needs to be provided for small CSE research groups.

Motivated by the above-mentioned questions, in this paper we conduct a detailed analysis of the state of the art of cloud computing offers for small CSE research groups. By focusing on high-level platform and programming support, and supporting tools and services for cost evaluation and software developments, we have identified several concerns that should be addressed. For example, cost estimation and monitoring tools are not well supported, platform and programming support for CSE in cloud systems are poor, and generic CSE services have not been widely deployed in the cloud under the SaaS model. Based on our findings, we present several recommendations for eliminating these concerns in order to attract more small CSE groups to use cloud computing.

The rest of this paper is organized as follows: Section 2 discusses related work. Section 3 gives an overview of relevance to and requirements from small CSE research groups. Section 4 identifies main concerns for the adoption of cloud computing in small CSE groups. In Sect. 5 we present our recommendations to cloud computing providers and researchers. We conclude the paper and outline our future work in Sect. 6.

2 Background and related work

Recently, cloud computing has been increasingly used in solving business and e-science problems. Although there is a confusion on the definitions of cloud computing, cloud computing is considered as an emerging model which aims at allowing customers to utilize computational resources and software hosted by service providers [8,11,22,25], thus shifting the complex and tedious resource and software management tasks typically done by the customers to the service providers. To date, many cloud computing providers exist to offer different functionalities. However, as we will discuss in detail in Sect. 4, only a few providers could be used by CSE groups; these providers focus on IaaS features by providing machines and storage systems.

With respect to the utilization of cloud computing for CSE, CSE scientists have started using cloud computing for CSE applications, such as [23,30], and they have provided various feedback on experimenting cloud computing for scientific applications. However, these works focus on concerns related to particular applications. Many computer scientists have evaluated particular cloud systems, such as Amazon [26,43]. However, they do not analyze a wide perspective of cloud computing for small CSE research groups.

Buyya and his colleagues have discussed the limitation of cloud computing in terms of market-orientation [11]. However, they do not discuss benefits of cloud computing for the CSE's market. A detailed report has been made about cloud computing [8] which presents various limitations of cloud computing. While it does not concentrate on CSE issues, it presents various obstacles that cloud customers could face, thus these obstacles are also valid for CSE groups. That report considers generic properties of clouds but does not specifically analyze current support of IaaS, PaaS, and SaaS providers for small CSE research groups.

With respect to cost estimations for scientific applications in cloud computing, [8,22] give generic cost comparison discussion and [16] provides experimental work on costs for workflows in cloud computing. Both raise important questions on the complexity of cost models when using cloud computing. We share similar questions through our discussion but we focus on understanding how current tools can be used to evaluate costs for CSE applications.

In their short paper, Wang et al. [61] have summarized some recent cloud systems, such as OpenNebula, Amazon, and Nimbus, and discussed functionalities of clouds, key features and enabling technologies. However, they do not analyze limitations of clouds for scientific computing and how to overcome these limitations. A special issue discussing how cloud computing can be used for sciences is given in [55]. This issue includes discussion about important scientific algorithms in the cloud, such as based on MapReduce. Especially, in this issue, Sterling and Stark have discussed how cloud computing fulfills the needs of main requirements from high performance computing (HPC), for example, capability, capacity and collaborative demands [54]. We do not examine the type of HPC applications but focus more on how cloud computing could support CSE scientists from general software development aspects. In a recent report on the Open Cirrus testbed [9], eight cloud computing testbeds are also briefly compared based on their types of research, approach, participants and distribution.

What we have observed is that computer scientists have discussed and demonstrated many aspects of cloud computing for scientific applications, while CSE scientists have not really taken the advantage of cloud computing due to several reasons that we will present in later sections.

3 Cloud relevance to and requirements from CSE groups

With respect to software development and execution, CSE scientists have quite different needs from their business counterpart. CSE applications have some distinguishable properties, as also studied in [2,35], with respect to programming languages (mainly C/C++, Fortran, Java, Perl, and Python), high performance computing systems, parallel programming models (e.g., MPI, OpenMP, and MapReduce), and computational libraries (e.g., BLAS, LAPACK, ParMETIS, and Trilinos).

Unlike big CSE research groups which own strong computing and storage infrastructure and/or have supporting teams of compute scientists and IT professionals, small CSE research groups[1] face many challenges. First, they need computational resources which are currently not enough. Second, they also do not have knowledge and staff to manage complex computational resources and storage systems. Third, they also do not want to invest a lot of time on managing infrastructures. The management is typically done by PhD students who want to concentrate on their own research rather than setup and manage (high performance) computing systems. Some groups even have their own services deployed for a wide use because these services are useful for research. However, these groups do not want to release the source code of their services due to many reasons, e.g., competitive research.

As promised by the benefits of cloud computing [50], cloud computing could offer a great chance for small CSE groups to deal with these problems. Generally, the CSE scientists[2] found that cloud computing is very relevant. In our discussion, we found the following findings describing how cloud computing is relevant to CSE groups:

- *Reducing management cost*: obviously CSE scientists appreciate how cloud computing could help them to reduce the operation and management cost. In particular, management cost is important as, with cloud computing, CSE scientists and PhD students can focus on their own research, instead of dealing with the complexity of high performance computing systems. This benefit also meets the main slogan of the cloud computing industry.
- *Reducing resource cost and fostering research proposal acceptance*: the benefit of cloud computing is not only due to the simple management but it is also related to research funding and proposals. Interestingly, some funding agents have asked CSE scientists why they do not use resources from others and it is very hard to get funds for supporting the setup and maintenance of CSE infrastructures.
- *Improving application capability sharing*: many research results can be exposed as a service in cloud environments that can be used by other research groups.
- *Improving data sharing*: CSE research groups need to share data with other collaborators and the data can be easily shared if it is in cloud. Many large-scale datasets which are hard to manage by a single group could be put into clouds for various research groups.
- *Supporting reproducible research results*: this is probably one of the most important factors when considering cloud computing for CSE. The issue of reproducible research results is critical in science, generally, and in CSE, particularly. However, it is hard, if not impossible, to reproduce the research result when different experimental testbeds are used. In fact, in many cases, it is impossible to setup similar testbeds, let alone to reproduce similar experiments. With cloud computing, this issue can

---

[1] Specifically our discussion is based on small CSE groups in Austria. Most Austrian CSE groups are small and they work independent. A typical size of a group is from 5 to 30 people, including professors, post-docs, PhD/master students, and supporting staffs. Due to the organizational structure of Austrian universities, research groups are very independent. CSE groups need computational infrastructure and most of them own a small cluster (4–32 cores), apart from shared computational resources provided by the universities and research institutions.

[2] In this section, in most cases, CSE scientists mean those participating in our discussion in [48].

Table 1 Major types of cloud computing providers from the list of 100 cloud players [27]

Type	Focus	Number
IaaS	Compute as a service by providing machines	11
	Storage as a service, Web hosting	11
PaaS	Web application development	10
	SaaS integration	2
SaaS	CSE	0
	Enterprise computing	4
	Collaborative working environments, virtual desktops, social network	8
Others	Tool development, software and hardware providers, support services	46

Types are classed based on the product information

(partially) be solved as different groups could utilize similar configurations of cloud computing infrastructures.

However, cloud computing is a rather new model and not all CSE scientists are aware of and understand its benefits as well as its limitations. To further support CSE scientists to investigate the offers of cloud computing, we have studied the list of 100 players in the cloud ecosystem [27] and Table 1 presents the types of cloud computing providers in this list.[3] Obviously, CSE has not been well supported by existing cloud providers. While generic IaaS cloud systems are popular that may be used by CSE scientists, they lack many features required by CSE groups (see Sect. 4 for further discussion). On other hand, PaaS and SaaS for CSE are not well developed.

4 Raised issues

4.1 Cost issues

The first issue is that it is hard to understand and determine all elements of the actual, full cost model when using cloud computing. This cost model is necessary for scientists to decide when and under which form cloud computing offers can be utilized. CSE scientists need to estimate if the total cost of using cloud computing is smaller than that of their own infrastructure. The second issue is the lack of tools to monitor and analyze pay-as-you-go costs associated with runtime use of machines, storage systems, and networks for particular applications. It is needed for short term planning and on-demand resource allocation at runtime, e.g., to determine when cloud resources should be added into existing, local resources in order to perform a large-scale experiment. As studied, major factors of the cost of using cloud computing would

---

[3] Note that from the list some providers have been acquired and merged. Furthermore, many providers offer different types of services and information about some providers is not clear. Therefore, we choose to count only providers when information is clear. Our evaluation method is mostly relied on the analysis of existing documents. For a small number of systems, we were able to use demonstration/trial offers to (partially) evaluate them.

Table 2 Existing tools for determining costs

Tool	Capability		Scope				Note
	Estimation	Monitoring	Machine	Storage	Network	Application	
Amazon [19]	+	+	+	+	+		Only for machine, storage and network use
Walker et al. [60]	+			+			Comprehensive models for estimating cloud storage costs
Truong et al. [57]	+	+				+	Estimating and online monitoring costs for applications

be computing resources, storage systems, and data transfer, but the total cost is related to many factors and activities, such as computing resources, storage, data transfers, security, and data cleaning [7, 16].

While several papers have discussed and reported the cost of performing scientific applications in the cloud, these reports are either for specific applications [16], general discussion [7, 15], or comparing different cloud systems [34]. Table 2 presents existing tools that can be used to estimate, monitor and analyze costs of using clouds. On the one hand, while cloud service providers give some basic information about costs and tools for determining computation and data transfer costs, they are trivial. Using information provided by the provider, it is very difficult to calculate the cost of an application because scientists need to map the computation and data transfer models associated with their CSE applications to primitive prices of CPUs, storage size and network transfer. On the other hand, we lack cost estimation and monitoring tools that can provide detailed cost factors associated with different activities of scientific applications.

4.2 Software development, deployment and execution issues

Software development, deployment and execution support is crucial to scientists. Based on our study in Table 1, we further study how existing cloud players, including also software development providers not in the list mentioned in [27], support the development and execution of CSE applications.

With respect to IaaS, first of all, many scientists want to know more about detailed configurations of clouds, such as network topology and what could happen behind the scene. Such configurations could help scientists to understand better how they should port their applications to the cloud and how to optimize their applications. Understanding what happens behind the scene also helps scientists to decide whether they should use the cloud or not, and, if used, for which phases in their experiments. For example, many applications are involved with sensitive data. When more computing and storage resources are needed, sensitive data could be stored, cached and transferred among different nodes in the cloud. The scientists have to comply with different rules applied to sensitive data, thus they must know, for example, where the data is stored and what is the retention policy of the data. Unfortunately, current providers publish almost no information about detailed topology and possible situations related to data. Unclear information about the topology also creates confusion about the performance issue because sometime it is impossible to determine the location of services. Second, in order to use cloud computing, we need to classify which types of applications are suitable for clouds. Although, several papers have reported on the use of cloud for different scientific applications, most cloud providers do not explicitly present the types of application classes they support. Third, many CSE applications are parallel applications based on MPI, OpenMP, embarrassingly parallel, and workflow models. Therefore, we need good support for parallel applications. However, currently cloud computing providers offer very little support of such parallel applications. Tables 3 and 4 present deployed IaaS and platforms in our study. Clearly, there is a lack of strong support for CSE applications.

With respect to PaaS, we need simple ways of programming on the cloud by utilizing different programming environments and tools. However, currently PaaS for CSE applications is not in the focus of cloud computing providers. When a CSE scientist utilizes the cloud, he/she does not have to manage the machines. Nevertheless, with respect to software development and execution, apart from what the scientist has to do in his/her systems, she/he has to deal with the preparation of packages and machine configuration which is not simple at the moment. While many systems provide tools to manage the infrastructure, there is a lack of tools for wrapping scientific codes, packaging and deploying them to the cloud. It is important to have such tools because developing code in cloud systems needs to deal with different virtual machines and complex scientific libraries in a pay-as-you-go fashion. Other issues are application monitoring and performance optimization which are important and well supported in today's high performance systems. First, so far there have been a few discussion on how to monitor and optimize the performance of CSE applications in the cloud. When CSE applications are executed in the cloud, they run atop virtual machines. This requires a new class of performance monitoring and analysis tools for virtual machines. Second, the question of whether we should optimize the performance of CSE applications or not in the cloud is still open. Cloud providers often claim that the user can obtain unlimited resources the user wants. So does the elastic resource provisioning solve the performance problem? And what would be the hidden relationship between the cost and the resource usage in this way, if we ignore the performance optimization. Table 5 summarizes application development support of products and systems that we studied. A clear observation is that most systems support deployment utilities and system monitoring. (Automatic) deployment utilities are a must for application development in the cloud as the user does not want to deal with the complexity of deployment and installation of software in clouds, in particular, when the user relies on different IaaSs. However, such utilities in most studied systems are not for CSE applications.

With respect to SaaS, many CSE applications or part of CSE exploratory processes could be provided as a SaaS. However, currently, SaaS cloud computing providers have not focused on this sector.

Table 3 Overview of programming model and language support in IaaS studied

Systems	**Programming models**					**Programming languages**					**Notes**
Name	OpenMP/MPI	Web Services	MapReduce	Condor	Workflows	C/C++	Fortran	C#/.NET	Java	Scriptings	
Amazon EC2 [19]	o	o	+	o	o	o	o	o	o	o	Provides computing infrastructure. However, some public Amazon AIMs include images for scientific applications.
Appnexus [4]	o	o	o	o	o	o	o		o	o	Provides computing infrastructures based on that different OS images can be installed.
ElasticHosts [20]	o	o	o	o	o	o	o	o	o	o	Provides computing infrastructures based on that virtual machines can be installed.
Eucalyptus [42]	o	o	o	o	o	o	o		o	o	Provides a public cloud for research. It also includes software for setting cloud infrastructure.
Joyent [31]	o	o	o	o	o	+	o		+	o	Supports computing infrastructures based on Open Solaris.
Science Clouds/Nimbus [32]	+	o	+	o	o	o	o		o	o	Supports scientific and educational projects to experiment with cloud computing.
Skytap [51]	o	+	o	o	o	o	o	o	o	+	Supports virtual labs of pre-built virtual machines, databases, and software test tools.

+ and *o* denote *native support* (included in the system) and *user-specific support* (performed by the user based on his/her requirements), respectively. Most features in systems studied are marked as user-specific support because these systems mainly provide machines. As a result, the user has to perform several manual activities in order to prepare the right machine images for his/her applications

Table 4 Overview of programming model and language supports in platforms studied

Systems	**Programming models**					**Programming languages**					**Notes**
Name	OpenMP/MPI	Web Services	MapReduce	Condor	Workflows	C/C++	Fortran	C#/.NET	Java	Scriptings	
3tera AppLogic [1]		+			o						Enables the execution of cloud applications with a focus on scalable Web applications and services.
Appistry CloudIQ [3]		o			+	+	o	+	+		Includes cloud manager and engines which are used to managing and executing applications.
Apprenda SaaSGrid [5]		+						+			Supports C#/.NET based application development
Aptana Cloud [6]		+							+	+	Supports the development and execution of PHP, MySQL, Apache and Aptana Jaxer applications.
Boomi AtomSphere [10]					+				+		Supports the composition of processes of application integration, data integration, and B2B integration.
Gigaspaces [28]		+				+		+	+	+	Supports Web and enterprise applications, and parallel processing managed by the user.
Google App Engine [29]		+							+	+	Supports the development and execution of Web applications.
LongJump [36]		+			+				+		Supports Web applications for enterprises.
Microsoft Azure [38]		+			+				+		Support cloud application development based on Windows technologies (.NET, Windows Live, etc.)
Parabon Frontier [44]			+		+				+		Supports batch jobs and includes several financial forecasting and data mining to biological and nano-scale modeling and simulations.

+ and *o* denote *native support* and *user-specific support* (carried out by the user based on his/her requirements), respectively

Table 5 Overview of application development and monitoring utility support in systems and products studied

System	Development environment	Utilities			
		Wrapper tool	Deployment tool	System monitoring	Application monitoring
Appistry CloudIQ [3]	Microsoft Visio and Eclipse plug-in	+	+	+	
Apprenda SaaSGrid [5]	Microsoft Visio	+	+	+	
Aptana Cloud [6]	Eclipse-based Aptana Studio		+	+	
Boomi AtomSphere [10]	Web browsers		+		+
Gigaspaces [28]	Specific IDE and Eclipse plug-in		+	+	
Google App [29]	Specific SDK and Eclipse plug-in		+	+	
LongJump [36]	Eclipse plug-in and Web browsers			+	
Microsoft Azure [38]	Microsoft Visio	+	+	+	+
Parabon Frontier [44]	Eclipse plug-in and specific SDK	+	+	+	+

5 Enhancing CSE support in cloud systems

Based on our detailed study about current offers of cloud computing to small CSE research groups, in the following, we present specific recommendations for cost evaluation, IaaS, PaaS and SaaS.

5.1 Cost evaluation tools

With respect to the cost estimation, cloud computing providers and research communities should offer application-specific cost models. Scientists should be provided with cost models associated with application models, e.g., cost models for OpenMP, MPI, and workflows, and these models should allow the scientists to utilize their vast knowledge about their application requirements gained before the use of cloud computing. Furthermore, a scientific experiment might include different phases, each has a different application model and might or might not be executed in the cloud. Thus, it is interesting to have composite cost models in order to decide which phases should be executed by using the cloud. To date, good models for cost estimation are missing. Also there is no integrated cost evaluation tool that presents the correlation and difference among costs reported from multiple views, e.g, cost estimation, application-specific cost monitoring, and infrastructure-specific cost monitoring.

5.2 IaaS for CSE

Currently, IaaS typically supports the user to specify the number of machines and provides different images of different operating systems. We believe that current support is too low level and does not encourage CSE researchers to use cloud computing. We recommend IaaS for CSE to provide the following features:

5.2.1 Cluster and reproducible machine images and templates

CSE typically do not use a single machine. Most cloud providers offer single machine images, but not cluster machine images. Although some cluster machine images are provided through community/research effort and third parties, such as in Amazon EC2 [19] and Nimbus [32], this kind of support has not been on the main focus of cloud providers. However, cluster machine images are important for CSE applications. Such images require much more complex configurations which can consist of front nodes, compute nodes, double/virtual communication interfaces, network/distributed file systems, etc. Some research initiatives have focused on the provisioning of cluster images, such as [40], but their results need to be investigated and integrated into native support of IaaS providers. Furthermore, these images should be stored, creating reproducible images for supporting reproducible research experiments. It is even better if cloud providers and CSE communities provide machines including ready-to-use scientific images (e.g., like Amazon AMI scientific Linux 5.2 which includes mpiBlast from the AMI public site). Another interesting topic is to create images suitable for different cluster topologies. Hence, useful tools similar to the Elastra system [21] will simplify the task to define suitable infrastructure images for CSE application substantially. An other important point is that such images should be searchable and associated with their deployed software and performance information. This is particularly important for sharing and reproducible images among collaborative teams, e.g., in Elastic-R [12], as well as for dynamic resource deployment and provisioning, e.g., in Cafe [39].

5.2.2 Compilers and scientific libraries

Most cloud computing providers do not include a rich set of compilers for CSE applications, such as Fortran and Python besides C/C++ and Java. Scripting languages, such as Python, have increasingly played an important roles in scientific applications [41]. They are also well supported in Web applications in many cloud systems. Thus, we think that increasing the testing and deployment of such scripting languages for CSE would also attract CSE scientists. Scientific libraries used for CSE applications, such as MPICH, FFTW, LAPACK, ScaLAPACK, and BLAST, are typically installed manually and maintained by the user. It is important that these libraries are tested and included in machine and cluster images. New tools that are able to support the assembly and building scientific images suitable for different clouds are highly expected. In this respect, a good example in the business domain is the CohesiveFT service [13] which allows the developer to build and customize his/her software stack for different clouds.

5.2.3 Automatic dynamic resource provisioning

Certain CSE applications take a long time to finish and during their execution, resources are used differently. Although current resource acquisition in cloud systems can be elastic, in most cases, this has to be done manually by the user. An interesting point is how cloud systems provide facilities for acquiring new resources and preparing images in an automatic way so that these facilities can be utilized and integrated into CSE applications for the optimization of the resources used by CSE applications. The AutoScaling feature of the Amazon EC2 is a good starting point in this direction when it allows resources to be changed based on pre-defined triggers. However, it is still not clear how it can be easily integrated into CSE programming models and how it can be used together with the automatic deployment of images. In particular, it is very challenging to provide automatic dynamic resource provisioning mechanisms that allow scientists to use their own resources and add new cloud resources when needed, even though some underlying cloud techniques have enabled this, such as in the OpenNebula toolkit [53], by using virtual network and cloud interfaces to other IaaS providers. We believe that a systematic way to support the integration of resource provisioning, and images deployment and management into programming models is important for long-run applications.

5.3 PaaS for CSE

Generally, PaaS provides existing tools for the developer to write and deploy cloud applications. Examples of PaaS are the Aptana [6], the Appirio Cloud Connectors

[14], and the Bommi integration components [10]. Typically, a PaaS will provide a Web programming portal, a set of available components that can be composed, and libraries/tools that can be used easily. Providing PaaS for CSE means that cloud providers should support ready-to-use platforms for scientists to develop and test their applications. Thus, PaaS support should be concentrated on the following points:

5.3.1 Generic integrated IDEs for software development

Cloud computing providers offer many good IDEs for supporting the development of Web applications, but not for CSE applications. Similar to IDEs for Web applications, CSE scientists need good programming development IDEs. These IDEs will go beyond current IDEs for scientific applications, such as Eclipse for parallel programs [62] or Eclipse for workflows [46], because they have to support also packaging and runtime deployment of code and elastic control in the cloud. Furthermore, they have to be able to support different back-end cloud systems. While the use of multiple clouds is possible [33], we are not aware of software development tools for multiple clouds.

5.3.2 Supporting workflows, exploratory, interactive programming models

A great number of CSE applications follow the workflow model [17] that need not only high performance computing systems but also large-scale data storage ones. Both requirements are met by cloud computing, but the main challenge lies in the interconnection between computing services and data services, especially when they are executed in different clouds. To date, various cloud providers have jointly offered complementary services, for example, computing infrastructure and data storage center. However, the workflow programming model has not been well supported by cloud systems. In particular, the workflow support could also focus on the programming of CSE applications based on cross cloud providers, for example, handling biodata from one provider while using computing resources from another provider.

5.3.3 PaaS for testing CSE applications

This would be highly interesting because very often scientists develop and test their applications in local and small systems before they start to run their applications in large-scale production high performance systems. While they may be granted to access such systems (due to collaboration contract or relationship), it is not efficient to use these systems for development and testing (due to usage constraints). Interesting examples in the business domain are the SOASTA test platform [52] and Skytap [51].

5.4 SaaS for CSE

Providing SaaS for CSE is especially interesting for many domains. For example, sharing application capabilities using Web services and Web portal is very fine when scientists do not want to share the application's source code and when their applications are computation-intensive and non-interactive with simple input data. Similar to current SaaS support for the business domain, we recommend CSE SaaS providers to offer the following types of services:

5.4.1 Basic CSE services

SaaS providers for CSE could investigate existing, well-established CSE applications and provide them as services. Some have been already deployed in cluster environments, such as Wien2K [63] (electronic structure calculations of solids), Dynamite [18] (predicts protein motions), and VAMP/VASP [58] (ab-initio quantum-mechanical molecular dynamics). Many potential applications could be provided such as ParaView [45] (data analysis and visualization application). Typical applications described in [49] would be good candidates.

5.4.2 Mashup portal for connecting diverse CSE services

A good example of mashup portal for business applications is the Process Factory [24]. For CSE, we could develop similar connectors to connect different CSE services. Existing research results in Grid workflows [56] can be leveraged to provide new tools to deal with CSE services and service connectors in the cloud.

6 Conclusion and future work

In this paper, we have studied the state of the art of how cloud computing providers support small computational science and engineering (CSE) research groups. While many cloud providers exist, they mainly focus on supporting machine infrastructure, data hosting, Web applications development, and SaaS for business applications. IaaS/PaaS/SaaS providers could be utilized by CSE groups but these providers have not provided enough support for CSE customers with respect to programming models, supporting tools and proved working applications. The current landscape of the cloud computing ecosystem neglects potential CSE customers and we call for cloud computing providers as well as researchers to address found open issues.

While we recommend some solutions to cloud computing providers and researchers, whether cloud computing providers will put their effort more on supporting CSE is clearly dependent on business factors. It is, however, of interest for the CSE community to address the above-mentioned challenging issues in order to foster our CSE research, in particular for small groups, to enhance data and application sharing and to ensure reproducible experiments. In our future work, we will develop a service-oriented PaaS platform for data intensive applications because this topic has a good combination of cloud strength—data and computation support—and would free many hard jobs that scientists have to face when dealing the complexity of computational infrastructures and storage systems.

Acknowledgments The work in this paper is motivated from our discussion with around 40 Austrian scientists who gathered to discuss the current status of CSE research and collaboration [48]. They, most from the three main universities in Vienna, have indicated an interesting but hesitant attitude in adopting cloud computing for CSE research. We thank all participants in the Vienna CSE workshop 2009 for their fruitful discussion.

References

1. 3tera AppLogic (2009) http://www.3tera.com/AppLogic/. Accessed 8 July 2009
2. Ahern S, Alam S, Fahey M, Hartman-Baker R, Barrett R, Kendall R, Kothe D, Messer O, Mills R, Sankaran R, Tharrington A, White JB III (2007) Scientific application requirements for leadership computing at the exascale. Technical report, Oak Ridge National Laboratory. http://www.nccs.gov/wp-content/media/nccs_reports/Exascale_Reqms.pdf
3. Appistry CloudIQ Platform (2009) http://www.appistry.com/products. Accessed 8 July 2009
4. Appnexus (2009) http://www.appnexus.com/. Accessed 2 July 2009
5. Apprenda SaaSGrid (2009) http://www.apprenda.com/. Accessed 8 July 2009
6. Aptana Cloud (2009) http://www.aptana.com/cloud. Accessed 8 July 2009
7. Armbrust M, Fox A, Griffith R, Joseph AD, Katz R, Konwinski A, Lee G, Patterson D, Rabkin A, Stoica I, Zaharia M (2009) Above the clouds: A Berkeley view of cloud computing. Technical report. University of California at Berkeley, URL http://www.berkeleyclouds.blogspot.com/2009/02/above-clouds-released.html
8. Armbrust M, Fox A, Griffith R, Joseph AD, Katz RH, Konwinski A, Lee G, Patterson DA, Rabkin A, Stoica I, Zaharia M (2010) A view of cloud computing. Commun ACM 53(4):50–58
9. Avetisyan AI, Campbell R, Gupta I, Heath MT, Ko SY, Ganger GR, Kozuch MA, O'Hallaron D, Kunze M, Kwan TT, Lai K, Lyons M, Milojicic DS, Lee HY, Soh YC, Ming NK, Luke JY, Namgoong H (2010) Open cirrus: a global cloud computing testbed. Computer 43:35–43. http://www.ieeecomputersociety.org/10.1109/MC.2010.111
10. Boomi AtomSphere (2009) http://www.boomi.com/. Accessed 9 July 2009
11. Buyya R, Yeo CS, Venugopal S (2008) Market-oriented cloud computing: vision, hype, and reality for delivering it services as computing utilities. In: 10th IEEE international conference on high performance computing and communications, vol 0, pp 5–13. http://www.ieeecomputersociety.org/10.1109/HPCC.2008.172
12. Chine K (2010) Open science in the cloud: towards a universal platform for scientific and statistical computing. In: Furht B, Escalante A (eds) Handbook of cloud computing, Springer, USA, pp 453–474. http://dx.doi.org/10.1007/978-1-4419-6524-0_19, ISBN 978-1-4419-6524-0
13. Cohesive FT (2009) http://www.cohesiveft.com/. Accessed 2 July 2009
14. Connectors AC (2009) http://www.appirio.com/products/cloudconnectors.php. Accessed 16 June 2009
15. de Assuncao MD, di Costanzo A, Buyya R (2009) Evaluating the cost-benefit of using cloud computing to extend the capacity of clusters. In: HPDC '09: Proceedings of the 18th ACM international symposium on high performance distributed computing, ACM, New York, NY, USA, pp 141–150. http://www.acm.org/10.1145/1551609.1551635
16. Deelman E, Singh G, Livny M, Berriman B, Good J (2008) The cost of doing science on the cloud: the montage example. In: SC'08: Proceedings of the 2008 ACM/IEEE conference on supercomputing, IEEE Press, Piscataway, NJ, USA, pp 1–12. http://www.acm.org/10.1145/1413370.1413421
17. Deelman E, Gannon D, Shields M, Taylor I (2009) Workflows and e-science: an overview of workflow system features and capabilities. Future Gener Comput Syst 25(5):528–540. http://dx.doi.org/10.1016/j.future.2008.06.012
18. Dynamite (2009) http://www.s12-ap550.bioch.ox.ac.uk:8078/dynamite_html/collect_data_v1.5.html. accessed: 16 June 2009
19. EC2 AECCA (2009) http://www.aws.amazon.com/ec2/. Accessed 2 July 2009
20. ElasticHosts (2009) http://www.elastichosts.com/. Accessed 2 July 2009
21. Elastra (2009) http://www.elastra.com/. Last access: 16 June 2009
22. Erdogmus H (2009) Cloud computing: Does nirvana hide behind the nebula? IEEE Software 26(2):4–6. http://www.ieeecomputersociety.org/10.1109/MS.2009.31
23. Evangelinos C, Hill C (2008) Cloud computing for parallel scientific HPC applications: feasibility of running coupled atmosphere-ocean climate models on Amazon's ec2. In: The first workshop on cloud computing and its applications (CCA'08). http://www.cca08.org/papers/Paper34-Chris-Hill.pdf
24. Factory TP (2009) http://www.theprocessfactory.com/. accessed: 16 June 2009

25. Foster IT, Zhao Y, Raicu I, Lu S (2009) Cloud computing and grid computing 360-degree compared. CoRR abs/0901.0131
26. Garfinkel SL (2007) An evaluation of amazon's grid computing services: Ec2, s3 and sqs. Technical report. TR-08-07, Harvard University. http://www.ftp.deas.harvard.edu/techreports/tr-08-07.pdf
27. Geelan J (2009) Cloud computing: 100 players in the cloud computing ecosystem. http://www.cloudcomputing.sys-con.com/node/770174
28. GigaSpaces (2009) http://www.gigaspaces.com/. Accessed 8 July 2009
29. Google App Engine (2009) http://www.code.google.com/appengine/. Accessed 9 July 2009
30. Hoffa C, Mehta G, Freeman T, Deelman E, Keahey K, Berriman B, Good J (2008) On the use of cloud computing for scientific workflows. In: IEEE international conference on eScience, vol 0, pp 640–645. http://www.ieeecomputersociety.org/10.1109/eScience.2008.167
31. Joyent (2009) http://www.joyent.com/. Accessed 2 July 2009
32. Keahey K, Freeman T (2008) Science clouds: early experiences in cloud computing for scientific applications. Cloud computing and applications
33. Keahey K, Tsugawa M, Matsunaga A, Fortes J (2009) Sky computing. IEEE Internet Comput 13:43–51. http://www.ieeecomputersociety.org/10.1109/MIC.2009.94
34. Kondo D, Javadi B, Malecot P, Cappello F, Anderson DP (2009) Cost-benefit analysis of cloud computing versus desktop grids. In: IPDPS '09: Proceedings of the 2009 IEEE international symposium on parallel & distributed processing, IEEE Computer Society, Washington, DC, USA, pp 1–12. http://dx.doi.org/10.1109/IPDPS.2009.5160911
35. Kothe D, Kendall R (2007) Computational science requirements for leadership computing. Technical report, Oak Ridge National Laboratory. http://www.nccs.gov/wp-content/media/nccs_reports/ORNL_TM-2007_44.pdf
36. Long Jump (2009) http://www.longjump.com/. Accessed 9 July 2009
37. Mell P, Grance T (2009) The Nist definition of cloud computing. http://www.csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc, v15
38. Microsoft Azure Services Platform (2009) http://www.microsoft.com/azure/default.mspx. Accessed 8 July 2009
39. Mietzner R, Unger T, Leymann F (2009) Cafe: a generic configurable customizable composite cloud application framework. In: Meersman R, Dillon TS, Herrero P (eds) OTM conferences (1), Springer, Lecture notes in computer science, vol 5870, pp 357–364
40. Moreno-Vozmediano R, Montero RS, Llorente IM (2009) Elastic management of cluster-based services in the cloud. In: ACDC '09: Proceedings of the 1st workshop on automated control for datacenters and clouds, ACM, New York, NY, USA, pp 19–24, http://www.acm.org/10.1145/1555271.1555277
41. Nilsen JK (2007) Python in scientific computing: applications to Bose–Einstein condensates. Comput Phys Commun 177(1–2):45
42. Nurmi D, Wolski R, Grzegorczyk C, Obertelli G, Soman S, Youseff L, Zagorodnov D (2009) The eucalyptus open-source cloud-computing system. In: Cappello F, Wang CL, Buyya R (eds) CCGRID, IEEE Computer Society, pp 124–131
43. Palankar MR, Iamnitchi A, Ripeanu M, Garfinkel S (2008) Amazon s3 for science grids: a viable solution? In: DADC '08: Proceedings of the 2008 international workshop on Data-aware distributed computing, ACM, New York, NY, USA, pp 55–64. http://www.acm.org/10.1145/1383519.1383526
44. Parabon Frontier (2009) http://www.parabon.com/. Accessed 8 July 2009
45. Paraview (2009) http://www.paraview.org/. Accessed 16 June 2009
46. Pautasso C, Heinis T, Alonso G (2006) Jopera: autonomic service orchestration. IEEE Data Eng Bull 29(3):32–39
47. Sameh A, Cybenko G, Kalos M, Neves K, Rice J, Sorensen D, Sullivan F (1996) Computational science and engineering. ACM Comput Surv 28(4):810–817. http://doi.acm.org/10.1145/242223.246865
48. Science VC, 2009 EW (2009) http://www.infosys.tuwien.ac.at/autocompwiki/index.php/CSE09. Accessed 16 June 2009
49. Simpson AD, Bull M, Hill J (2008) Identification and categorisation of applications and initial benchmarks suite. http://www.prace-project.eu/documents/documents/Identification_and_Categorisatio_of_Applications_and_Initial_Benchmark_Suite_final.pdf
50. Six Benefits of Cloud Computing CCND (2008) http://www.web2.sys-con.com/node/640237
51. Skytap (2009) http://www.skytap.com. Accessed 2 July 2009
52. SOASTA (2009) http://www.soasta.com/. Accessed 16 June 2009

53. Sotomayor B, Montero RS, Llorente IM, Foster I (2009) Virtual infrastructure management in private and hybrid clouds. IEEE Internet Comput 13(5):14–22. http://dx.doi.org/10.1109/MIC.2009.119
54. Sterling T, Stark D (2009) A high-performance computing forecast: partly cloudy. Comput Sci Eng 11(4):42–49. http://www.ieeecomputersociety.org/10.1109/MCSE.2009.111
55. Sullivan F (2009) Guest editor's introduction: Cloud computing for the sciences. Comput Sci Eng 11(4):10–11. http://www.ieeecomputersociety.org/10.1109/MCSE.2009.121
56. Taylor IJ, Deelman E, Gannon DB, Shields M (2006) Workflows for e-Science: scientific workflows for grids. Springer New York, Inc., Secaucus, NJ, USA
57. Truong HL, Dustdar S (2010) Composable cost estimation and monitoring for computational applications in cloud computing environments. In: The international conference on computational science 2010 (ICCS 2010). Tools for program development and analysis in computational science, Elsevier, Amsterdam, The Netherlands
58. VAMP/VASP (2009) http://www.cms.mpi.univie.ac.at/vasp. Accessed 16 June 2009
59. Vaquero LM, Rodero-Merino L, Caceres J, Lindner M (2009) A break in the clouds: towards a cloud definition. SIGCOMM Comput Commun Rev 39(1):50–55. http://www.acm.org/10.1145/1496091.1496100
60. Walker E, Brisken W, Romney J (2010) To lease or not to lease from storage clouds. Computer 43:44–50. http://doi.ieeecomputersociety.org/10.1109/MC.2010.115
61. Wang L, Tao J, Kunze M, Castellanos AC, Kramer D, Karl W (2008) Scientific cloud computing: early definition and experience. In: HPCC, IEEE, pp 825–830
62. Watson GR, DeBardeleben NA (2006) Developing scientific applications using eclipse. Comput Sci Eng 8(4):50–61. http://www.dx.org/10.1109/MCSE.2006.64
63. Wien2k (2009) http://www.wien2k.at. Accessed 16 June 2009