A Multiagent System for Resource Distribution into a Cloud Computing Environment

Fernando De la Prieta[1], Sara Rodríguez[1], Javier Bajo[2], and Juan Manuel Corchado[1]

[1] Department of Computer Science and Automation Control, University of Salamanca,
Plaza de la Merced s/n, 37007, Salamanca (Spain)
[2] Department of Artificial Intelligence, Technical University of Madrid, 28660,
Campus Montegancedo, Boadilla del Monte, Madrid (Spain)
{fer,srg,corchado}@usal.es, jbajo@fi.upm.es

Abstract. It is undeniable that the term Cloud Computing has gained in importance at a remarkable pace. It is a technology which is becoming a common element of our life, due to the variety of devices related to the Internet of Things. In this technological frame, there are not many studies in which a Multiagent system has facilitated the management of a cloud-based computational environment; although a first sight its features (autonomy, decentralization, auto-organization, etc.) seem suitable for the task. This study presents the +Cloud which is a cloud platform managed by a Multiagent System.

Keywords: Cloud Computing, Multi-agent System. Allocating resources.

1 Introduction

Cloud computing is being positioned as one of the main technological paradigms to have appeared in recent years. Among its most notable features [14] is its capability to offer any type of computational services; in addition to a pool of variable resources that satisfy the computational needs of the offered services. This means that it is possible to provide new resources to the services in an elastic way according to current demand.

The reasons for the quick growth of the computational paradigm are varied, but it is possible to group them into three main categories. The first group is formed by the main technology companies (IBM, Google, Amazon, Microsoft, etc.) who have an economic interest in this paradigm as a new market. While previously an emergent market, its current dominance can be explained, in part, by a new business model which does not require an initial investment; instead, the client simply pays according to the resources used (*pay-as-you-go* [4]) These companies oriented their efforts (economic, technological and human) to the development of this technology by creating various pilot projects (Sun Cloud by Sun, Blue Cloud by IBM, etc.), in addition to other open approaches [21] which eventually resulted in what we know as Cloud Computing. Secondly, the quick birth has been possible as a result of the maturity of the variety of technological components (server, cluster, high availability, grid computing) that form the computational paradigm; as well as the tremendous research, at

Y. Demazeau et al. (Eds.): PAAMS 2013, LNAI 7879, pp. 37–48, 2013.

both the hardware and software level, in incipient technologies such as virtualization [3]. Finally, there cannot be the slightest doubt that positive public reception has been a key factor in its rapid development. From the public's perspective, a cloud environment makes it possible not only to synchronize data, information, or even tasks, projects, etc., but also to work in a delocalized way through the use of online tools. And, from the companies' perspective, the main advantage is that a cloud environment does not require an initial investment, making it possible to pay only for those resources that are required at a particular moment.

Multi-agent Systems (MAS) have not played an important role in the development of Cloud technology, as Talia presents in this study [22]. According to Talia, it is possible to distinguish to groups: (i) MAS that use the computational features of a Cloud environment (processing, storage, etc.); and (ii) Cloud environments that use MAS for the internal management of their resources, or to offer intelligent services. As is shown in the following section, the state-of-the-art indicates that the majority of the applications are related to the former group (agents using Clouds) [12][17]. Within the latter group, however, it is only possible to find incipient research lines focused on applying agreement and negotiation technologies to the task of allocating the internal resources of the cloud [1], or reaching an agreement on the quality of the cloud services being offered [15].

Although there is no clear convergence between MAS and Cloud Computing, it is obvious that the MAS features (dynamicity, flexibility, autonomy, proactivity, learning, etc.) are exactly the features that a Cloud environment needs for the self-management of its resources. A Cloud Computing environment has a set of resources (physical and virtual) which have to vary dynamically in order to cope with the demand of the computational services being offered. Within this model, the decision making process is complex, due to the variability of the demand for services and the lack of information on the decision components. This is the why an agent-based Cloud computing environment is suitable for the efficient allocation of computational resources, enabling the dynamic and automatic readaptation of each element which forms part of the cloud environment.

This study presents the +Cloud (masCloud) platform that offers Cloud services such as storage of information and files at the platform level, as well as a set of native software applications. The core of this platform is a MAS based on a Virtual Organization (VO), which makes it possible to automatically manage the computational resources of the system, adapting them in an elastic way according to demand.

The present paper is structured as follows: the following section presents the state-of-the art of Cloud computing system and its relationship with MAS. The +Cloud architecture is then presented in detail. This study finalizes with a review of the initial tests of the system, conclusions and future research lines.

2 Cloud Computing and Multiagent Systems

Cloud Computing, understood as computational paradigm, is emerging recently with great importance. Although it may be initially considered another computational paradigm, reality indicates that its rapid progression is motivated by economic interests [4] in the underlying computational features.

Historically, the term *Cloud Computing* was first used by Professor Rammath. [5]. However, the concept was becoming popular through Salesforce.com, a company that focused its market strategy to offer software as a service (SaaS) to big companies. However, IBM was the first company to detail the specific terms of the guidelines of this technology (auto-configuration, auto-monitorization, auto-optimization) in the document *Autonomic Computing Manifesto* [20]. By 2007, Google, IBM and others had joined together to form a research consortium which resulted in the birth of this technology as we know it today. [13]

For the large companies, knowledge about this technology is a competitive advantage. First of all, the Cloud provider can offer its services through a pay-as-you-go model [4] [9], following the guidelines proposed by Utility computing [16]. Additionally, the Cloud user does not have to be concerned with demand peaks, transforming passive investments in operational expenses [2].

A large number of definitions [4][14][9][2] have emerged at both a company and academic level. In each one, the authors try to highlight the most relevant features from their point of view. When a wide number of definitions are analyzed, it is possible to distinguish two big groups:

1. Those whose interests are focused on defining the technological aspects of the computational paradigm; these can be further divided in those who focus on defining either hardware or software characteristics
2. Those whose interest is to highlight the aspects related to the negotiation model, which is intrinsically associated with a Cloud environment.

It was the American NIST (National Institute of Standards and Technology)[1] which defined Cloud Computing [3] as a *model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*. Further to this definition, NIST presents a set of features, different deployment models (private, public Community Cloud); and most importantly, three models of service. Understanding service as a capability that the Cloud offers to the end users, we can underscore the following three service models:

- Software as a Service (SaaS). This capability allows the provider to supply the user with applications that can be directly executed on the cloud infrastructure. This entails a number of advantages such as the ubiquity of the applications or the use of light clients. However, there are also a number of difficulties (which in some cases are strengths) directly related to the consumer's loss of control over the infrastructure (network, storage, operating system, difficulty to configure, etc.).
- Platform as a Service (PaaS). This capacity is supplied by the provider and allows the consumer to use the necessary tools to create their own applications within the

[1] `http://www.nist.gov/`

Cloud environment. Some of these services include programming, libraries, tools, etc.
As with the services in the previous level, the programmer does not control the underlying infrastructure, nor the environment where the applications are deployed.

- Infrastructure as a Service (IaaS); (o Hardware as a Service for Wang *et al.* [24]). This capability provides to the consumer include different kinds of hardware such as processing, storage, network, etc.
  This capacity can be provided to consumers with the ability to install their own software in an operating system deployed in a hardware environment, obviously virtualized, with characteristics defined by the actual user.

This division of models leads some authors to speak of Something as a Service (*ssS) [22].

2.1 Cloud and Agents

As indicated in the introduction, there is only a limited number of studies in the state of the art that relate Cloud Computing and agent technology [22]. In general terms, a Cloud system may use MAS applications in a Cloud environment for deployment, and there are also Cloud environments that use agent technology to manage their resources. Some of those applications include:

- **Agents using Cloud.** Within this group, the main state of the art applications use computational resources from the Cloud environment. For example, there are systems such as those described in [7][8] that use the computational strength of the environment to perform simulations in different fields. Another example is presented in [6], where the Cloud environment is used as a persistence engine for information.
- **Cloud using Agents.** Within this subgroup, the range of possibilities is even further extended. Mong Sim [15] highlights three subgroups of applications: (i) combination of resources among Cloud providers; (ii) planning and coordination of shared resources; (iii) establishing contracts between users and Cloud service providers. As Mong Sim points out, it is possible to find studies such as [11][23] that develop a Cloud service using agents for different specific purposes. Mong Sim used the Cloudle [15] which is an agent-based tool for discovering Cloud services. Some notable examples of Cloud providers combining resources include studies by Kaur Grewal [10] and Aarti Singh [19], which use shared Cloud resources to offer Infrastructure as a Service (IaaS) Examples that apply SLA to distribute services include [15][18]. Finally we should point out the application of negotiation and agreement algorithms applied to different levels and processes within the framework of a cloud computing environment [25].

3 Proposed Architecture: +Cloud

+Cloud is a platform based on the cloud computing paradigm. This platform allows offering services at the PaaS and SaaS levels. The IaaS layer is composed of the physical environment which allows the abstraction of resources shaped as virtual machines; however, the system does not offer this kind of service to the end users as shown in Figure 1.

+Cloud has a layered structure that covers the main components of cloud computing:

- The SaaS layer is composed of the management applications for the environment (control of users, installed applications, etc.), and other more general third party applications that use the services provided by the next layer (PaaS).
- The PaaS layer provides services through REST web services in API format. One of the more notable services among the APIs is the identification of users and applications, a simple non-relational database service and a file storage area that controls versions and simulates a directory structure.
- The IaaS layer is used to deploy all management and general-purpose applications, in addition to the all services at the platform layer. This layer provides a virtual hosting service with automatic scaling and functions for balancing workload.

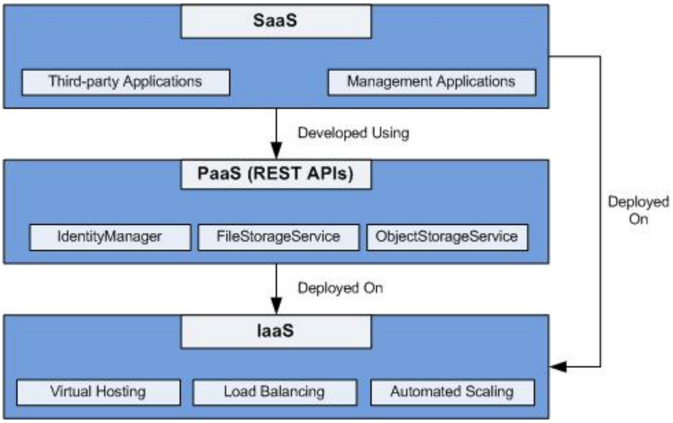

Fig. 1. +Cloud layer overview

From an internal viewpoint at a technological level, a Cloud Computing system is formed by a large set of computational resources, referred to in previous technologies as a server cluster or server farm. Abstractions are performed over these hardware resources, as virtual machines, which allows the easy and dynamic management of computational resources. Although performance decreases as a result of the computational needs of managing virtual computational resources, the advantages exceed the disadvantages, since complex tasks, such as the creation/destruction of virtual machines based on templates, the dynamic configuration of assigned resources, or even the migration of virtual machines between physical servers without stop the pending task, are made possible by virtualization.

In conclusion, a Cloud computing environment such as +Cloud platform can be viewed, at an external level, as a set of computational resources offered to end users. At an internal level, these services are deployed into a set of virtual machines that are hosted by physical server of the computational environment, as shown in Figure 2.

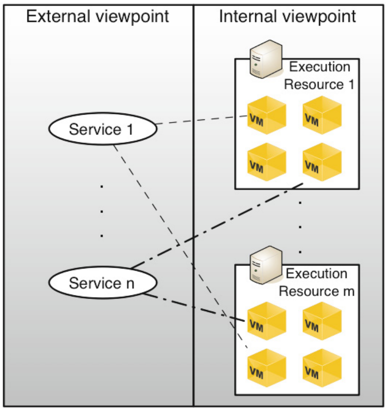

Fig. 2. Cloud Computing deployment

The distribution of physical resources between the different virtual machines and between the different system services is a matter of current interest [25][15][19]. The redistribution of resources can be seen from both a micro and macro level point of view. From a micro point of view, there is a distribution of resources between the virtual machines that accommodate a single host. In other words, a physical server has a set of physical resources available (processing, memory and drive) that must be shared among the different virtual instances that it hosts, leaving a set of minimum resources available for its own host. At the macro level, there is a redistribution of resources at a global level in the Cloud, which entails migrating virtual machines in use between different servers, and turning on and off the physical machines that provide or consume resources within the Cloud environment.

The +Cloud platform uses a virtual organization of agents to manage the system resources. MAS can be perfectly adapted to solve this problem, as it allows making decisions in an open environment where the availability of information is limited and agents are thereby required to make decisions, amidst great uncertainty, that affect the entire system. As the decision making is a distributed process, the system has greater availability than other systems in which decision making is a centralized process.

Figure 3 provides a high level description of the system. As shown, the system is divided into the following agent organizations:

- **Resource Organization.** This agent organization is charge of managing both the physical and virtual system resources. The agents are distributed throughout the hardware elements of the Cloud environment. Their main goal is to maximize the use of resources. It is intended that there are no active resources that are underutilized, which implies that there must be the smallest possible number of active physical machines to satisfy the current demand. At the same time, the computational load of the active physical resources must be high. Within this organization includes the following roles:

  - *Local Manager*, in charge of allocating the resources of a single physical machine among its virtual machines and its own physical machine. There is one in each physical server.

    In terms of its internal architecture, a CBR-BDI [26] agent can redistribute the resources of each physical machine among the different virtual machines according to the partial information that it has. It can modify (increase or decrease) the resources of the physical machines in use.

    Additionally, it can start up or shut down virtual machines within the local server; it does not do so by its own initiative, however, since it is the Global Regulator agent within the same machine that gives the order.
  - *Local Resource Monitor*, in charge of knowing the usage level of the virtual resources of each virtual machine. There is one monitor for each physical machine and it has all the knowledge about the physical machine as well as its virtual machines.
  - *Global Regulator*, in charge of negotiating with its peers regarding the redistribution of the resources at a global level. There is one in each physical server.

    The Global Regulator uses agreement algorithms between peers to distribute resources at a global level. When the service does not have the desired quality, all agents throughout the system with this role must reach an agreement as to how to solve the problem. To do so, they will use a distributed CBR system and algorithms according to [25]. Once the decision has been made, it is applied to the system in order to solve the problem.
  - *Network Monitor*, this role can monitor the network from the point of view of each single physical machine. There is one in each physical server.
  - *Hardware Manager*, the goal of this role is to manage at all times the hardware that is both in use and on standby. There is one in each physical server, each of which acts as coordinator.

- **Consumer Organization.** At the technological level this organization deploys over the computational resources offered by the organization described in the previous section. The services encompassed by this organization will, therefore use the system resources according to existing demand. Its main goal is to maximize the quality of service, which requires monitoring each service individually, keeping in mind that each service will be deployed simultaneously on various virtual machines located in different physical services.
  - *Service Supervidor,* this role is responsible for making decisions about each individual service. There is one for each service, each of which is located in the same virtual machine hosting the SDM of the same service, which in turn incorporates the load balancer service.

- *Service Demand Monitor*, in charge of monitoring each demand service which is offered by +Cloud. There is one agent of this type per each kind of service. They incorporate a load balancer to redirect requests to the different virtual machine which are offering the service at that time.
- *User,* represents the system users that use the services. As such, they are the ones that ultimately use the system resources. There can be different types of users: SaaS User, Cloud User and Administrator.

- **Management Organization.** This organization is in charge of ensuring that the entire system functions correctly, which is in fact its primary goal. There are two types of roles:
  - *Global Supervisor,* which ensures that the other roles and agents of the VO work correctly. If something fails, or one of the agents does not respond to its messages, this role will take the necessary actions to restore the system to a functioning state.
  - *Identity Manager*, which is in charge of allowing other agents to enter or exit the system. Within the framework of this system, each time that a service is initiated or suspended, as with a physical machine, agents will enter or exit the system. This manager is also in charge of logging the User agents. As with all other services, this role, which itself constitutes a service, must be monitored as such.

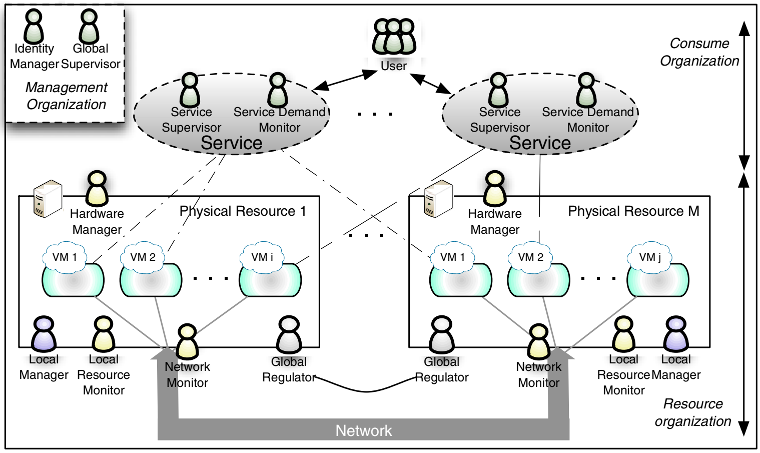

Fig. 3. MAS for resource redistribution in +Cloud

4 Preliminary Results

The +Cloud architecture has been continually evolving since its beginning. Beyond the multiagent system that governs the computational environment, it is also necessary to develop a set of services that are offered to the end user. Some of the more important services include FSS (*File Storage Service*), which stores files, and OSS (*Object Storage Service*), which stores information in a non-SQL database. The elasticity of these Cloud services is supported in the +Cloud architecture previously presented.

In order to obtain data regarding the performance of the Cloud environment and its ability to adapt to changes in the demand of services, a Denial of Service (DoS) attack is executed over the FSS. This is done by sending a constant stream of requests to the service over a period of 300 seconds. The number of requests is continually increased during the time of the test. This section presents the results obtained.

As indicated, the adaptation can be seen from both a micro and macro perspective. At the micro level, the adaptation takes place in each of the individual servers within the Cloud environment. This provides the physical server with limited capabilities (processing, memory, hard drive, etc.), which it must then share among the virtual machines it hosts. The adaptation takes place through the *Local Manager* agent, which works closely with the *Local Resource Monitor* agent. These agents are local to each machine and do not have information about the other nodes in the Cloud environment. The *Local Manager* agent uses a CBR as a support system for making decisions. The information that this agent has is provided by the monitor, which is continually gathering data about the computational load of each machine in the physical server. Figure 4 shows the adaptation of the system. The graph on the left shows an increase in the use of the processor (blue line) and the corresponding increase of CPU assigned by the Local Manager (red line). The graph on the right shows the same process, except that the memory is assigned to a virtual machine; the blue line is the memory used while the red line is the memory assigned.

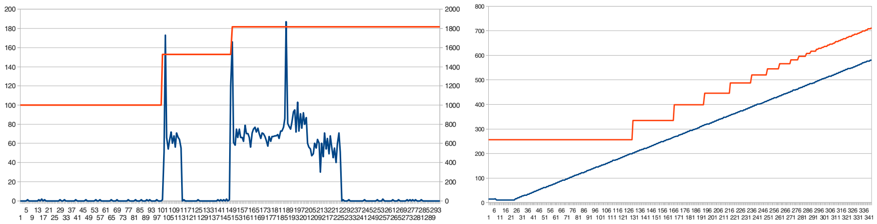

Fig. 4. Redistribution of resources at micro-level (Left: processor, Right: memory)

When the *Local Manager* detects insufficient resources, or the *Service Supervisor* detects decreased quality of service, an adaptation process takes place at a macro level. This process, which can be reviewed in detail in [25], is based on the negotiation among the Global Regulator agents for different nodes in the Cloud environment. During this negotiation process, the agents decide how to redistribute resources among the different nodes (and not just internally) to rectify the problem of demand. Figure 5 shows how quality of service is improved. While the response time (y-axis) in the first part of service is very high, the quality of service improves considerably after two consecutive readaptations.

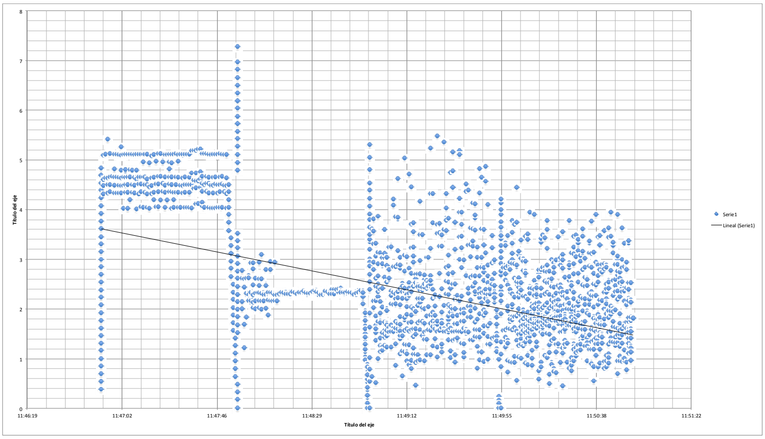

Fig. 5. Redistribution of resources at macro-level

5 Conclusions and Future Work

This study has presented the +Cloud platform, which is a Cloud Computing platform that is managed at internal level for a MAS based on a VO. As indicated in the introduction, initial results have shown that MAS technology is ideal for managing the computational resources of this type of system.

State of the art Cloud environments follow a centralized model for making decisions [21], which can lead to different problems such as (i) the need to centralize information; (ii) the need for a large computational load in the nodes where the decision making process takes place; and (iii) finally, the ability to recover from mistakes that can arise with a centralized decision making process. Using a model such as that proposed in this study, all of these problems can be resolved. This is due, in large part, to the system agents, which are able to make decisions in a Cloud environment where only partial information is available. Even if one or more nodes fail, it is still possible for the readaptation process to take place in the available nodes. Finally, the use of MAS allow using such techniques as the agreement techniques described in [25], which makes it possible to make decisions for readaptation at a global level without needing to centralize the information.

A final note with regard to future lines of work, given the great technological component of the system and its dependency on the environment, we expect that future versions of the Cloud environment will include the model proposed by Ricci in [27], which introduces the concept of Artifact to interact with the environment. The use of this model will facilitate the evolution of the platform and its independence from the underlying technological environment.

Acknowledgment. This work is supported by the Spanish government (MICINN), project OVAMAH, reference: TIN2012-36586-C03-01.

References

1. An, B., Lesser, V., Irwin, D., Zin,, M.: Automated Negotiation with Decommitment for Dynamic Resource Allocation in Cloud Computing. In: AAMAS 2010 Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems, vol. 1, pp. 981–988 (2010)
2. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: A view of Cloud Computing. Communications of the ACM 53(4), 50–58 (2010)
3. Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A.: Xend and the Art of Virtualization. In: ACM Symposium on Operating System Principles, Boltoin Landing, NY, USA, pp. 164–177 (2003)
4. Buyya, R.: Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, CCGRID 2009, pp. 5–13 (2009)
5. Chellappa, R.: Intermediaries in cloud-computing: A new computing paradigm. In: Informs. Cluster: Electronic Commerce, Dallas, Texas (1997)
6. Chen, C., Wang, K.: Cloud Computing for agent-based urban transportation system. IEEE Intelligent Systems 26, 73–79 (2011)
7. Cheng, Y., Low, M.Y.H., Zhou, S., Cai, W., Seng Choo, C.: Evolving agent-based simulations in the clouds. In: Third International Workshop on Advanced Computational Intelligence (IWACI), pp. 244–249 (2010)
8. Dębski, R., Byrski, A., Kisiel-Dorohinicki, M.: Towards and agent-based augmented cloud. Journal of Telecommunications and Information Technology, 16–22 (2012)
9. Erdogmus, H.: Cloud Computing: Does Nirvana Hide behind the Nebula? IEEE Sofware 26(2), 3–6 (2009)
10. Grewa, R.K., Pateriya, P.K.: A Rule-based Approach for Effective Resource Provisioning in Hybrid Cloud Environment. International Journal of Computer Science and Informatics, 101–106 (2012)
11. Jin Kim, M., Gun Yoon, H., Ku Lee, H.: IMAV: An intelligent Multi-agent model based on Cloud computing for resource virtualization. In: Computers, Networks, Systems, and Industrial Engineering, pp. 99–111 (2011)
12. Li, Z., Chen, C., Wang, K.: Cloud Computing for Agent-based Urban Transportation Systems. IEEE Intelligent System 26, 73–79 (2011)
13. Lohr, S.: Google and ibm join in cloud computing research. New York Times (2007)
14. Mell, P., And Grance, T.: The Nist Definition of Cloud Computing. In: NIST Special Publication 800-145. pp. 1–3. NIST (2011)
15. Mong Sim, K.: Agent-based Cloud Commerce. In: IEEE International Conference on Industrial Engineering and Engineering Management, pp. 717–721 (2009)
16. Ross, J.W., Westerman, G.: Preparing for utility computing: The role of IT architecture and relationship management. IBM Systems Journal 43(1), 5–19 (2004)
17. Schuldt, A., Hribernik, K.A., Gehrke, J.D., Thoben, K.D., Herzog, O.: Cloud Computing for Autonomous Control in Logistics. In: 10th Annual Conference of the German Society for Computer Science (2010)
18. Siebenhaar, M., Nguyen, T.A.B., Lampe, U., Schuller, D., Steinmetz, R.: Concurrent Negotiations in Cloud-based Systems. In: Vanmechelen, K., Altmann, J., Rana, O.F. (eds.) GECON 2011. LNCS, vol. 7150, pp. 17–31. Springer, Heidelberg (2012)

19. Singh, A., Malhotra, M.: Agent based framework for scalability in Cloud Computting. International Journal of Computer Science & Engineering Technology (IJCSET), 41–45 (2012)
20. Stoica, F., Morris, D., Karger, M., Kaashoek, F., Balakrishnan, H.: Chord: A scalable peer-to-peer lookup service for internet applications. In: Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM 2001, pp. 149–160 (2001)
21. Takato Endo, P., Estácio Gonçalves, G., Kelner, J., Sadok, D.: A survey on Open-source Cloud Computing Solutions. In: Brazilian Symposium on Computer Networks and Distributed Systems, pp. 3–16 (2010)
22. Talia, D.: Clouds Meet Agents: Towards Intelligent Cloud Services. IEEE Internet Computing 16(2), 78–81 (2012)
23. Venkataramana, K., Padmavathamma, M.: Agent-based approach for Authentication in Cloud. IRACST - International Journal of Computer Science and Information Technology & Security 2(3), 598–603 (2012)
24. Wang, L., Tao, J., Kunze, M., Castellanos, A.C., Kramer, D., Karl, W.: Scientific Cloud Computing: Early Definition and Experience. In: HPCC 2008: 10th IEEE International Conference on High Performance Computing and Communications, pp. 825–830 (2008)
25. Heras, S., De la Prieta, F., Julian, V., Rodríguez, S., Botti, V., Bajo, J., Corchado, J.M.: Agreement technologies and their use in cloud computing environments. Progress in Artificial Intelligence 1(4), 277–290 (2012)
26. Corchado, J.M., Pavón, J., Corchado, E., Castillo, L.F.: Development of CBR-BDI agents: A tourist guide application. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 547–559. Springer, Heidelberg (2004)
27. Ricci, A., Omicini, A., Viroli, M., Gardelli, L., Oliva, E.: Cognitive Stigmergy: Towards a Framework Based on Agents and Artifacts. In: Weyns, D., Van Dyke Parunak, H., Michel, F. (eds.) E4MAS 2006. LNCS (LNAI), vol. 4389, pp. 124–140. Springer, Heidelberg (2007)