

Journal Subline

LNCS 8670

Juan Manuel Corchado · Javier Bajo
Guest Editors

Transactions on **Computational Collective Intelligence XV**

Ngoc Thanh Nguyen
Editor-in-Chief

Ryszard Kowalczyk
Co-editor-in-Chief

 Springer

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, Lancaster, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Zürich, Switzerland

John C. Mitchell

Stanford University, Stanford, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

TU Dortmund University, Dortmund, Germany

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max Planck Institute for Informatics, Saarbruecken, Germany

More information about this series at <http://www.springer.com/series/8851>

Ngoc Thanh Nguyen · Ryszard Kowalczyk
Juan Manuel Corchado · Javier Bajo (Eds.)

Transactions on Computational Collective Intelligence XV

Editor-in-Chief

Ngoc Thanh Nguyen
Institute of Informatics
Wroclaw University of Technology
Wroclaw
Poland

Co-editor-in-Chief

Ryszard Kowalczyk
School of Software and Electrical
Engineering
Swinburne University of Technology
Hawthorn
Australia

Guest Editors

Juan Manuel Corchado
Departamento de Informática y Automática
Universidad de Salamanca
Salamanca
Spain

Javier Bajo
Departamento de Inteligencia Artificial
Universidad Politécnica de Madrid
Madrid
Spain

ISSN 0302-9743

ISBN 978-3-662-44749-9

DOI 10.1007/978-3-662-44750-5

ISSN 1611-3349 (electronic)

ISBN 978-3-662-44750-5 (eBook)

Library of Congress Control Number: 2014948788

Springer Heidelberg New York Dordrecht London

© Springer-Verlag Berlin Heidelberg 2014

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Contents

Building Optimal Macroscopic Representations of Complex Multi-agent Systems	1
<i>Robin Lamarche-Perrin, Yves Demazeau, and Jean-Marc Vincent</i>	
Understanding the Role of Emotions in Group Dynamics in Emergency Situations	28
<i>Alexei Sharpanskykh and Kashif Zia</i>	
Representation of the Agent Environment for Traffic Behavioral Simulation . . .	49
<i>Feirouz Ksontini, Stéphane Espié, Zahia Guessoum, and René Mandiau</i>	
Using LCS to Exploit Order Book Data in Artificial Markets	69
<i>Philippe Mathieu and Yann Secq</i>	
Coupled K-Nearest Centroid Classification for Non-iid Data	89
<i>Mu Li, Jinjiu Li, Yuming Ou, Ya Zhang, Dan Luo, Maninder Bahtia, and Longbing Cao</i>	
An Adaptative Multi-Agent System to Co-construct an Ontology from Texts with an Ontologist	101
<i>Zied Sellami and Valérie Camps</i>	
Game Theoretical Model for Adaptive Intrusion Detection System	133
<i>Jan Stiborek, Martin Grill, Martin Rehak, Karel Bartos, and Jan Jusko</i>	
+Cloud: A Virtual Organization of Multiagent System for Resource Allocation into a Cloud Computing Environment	164
<i>Fernando De la Prieta, Sara Rodríguez, Javier Bajo, and Juan M. Corchado</i>	
Author Index	183

+Cloud: A Virtual Organization of Multiagent System for Resource Allocation into a Cloud Computing Environment

Fernando De la Prieta¹(✉), Sara Rodríguez¹, Javier Bajo²,
and Juan M. Corchado¹

¹ Department of Computer Science and Automation Control,
University of Salamanca, Plaza de la Merced s/n, 37007 Salamanca, Spain
{fer, srg, corchado}@usal.es

² Department of Artificial Intelligence, Technical University of Madrid,
Campus Montegancedo, Boadilla del Monte, 28660 Madrid, Spain
jbajo@fi.upm.es

Abstract. Nowadays Cloud Computing has gained in importance at a remarkable pace. The key characteristic of this technology is the possibility to provide new resources to the services in an elastic way according to current demand. In contrast to Cloud Computing, Multiagent Systems are focus on other features such as autonomy, decentralization, auto-organization, etc. This study demonstrates that this features of MAS are suitable to manage the physical infrastructure of a Cloud Computing environment, in other words, we present +Cloud which is a cloud platform managed by a Multiagent System.

Keywords: Cloud computing · Multi-agent system · Virtual organizations · Allocating resources

1 Introduction

The technology industry is presently making great strides in the development of the Cloud Computing paradigm. As a result, the number of both closed and open source platforms has been rapidly increasing. From an external point of view, the three most widely known services are Software, Platform and Infrastructure [5]. From an internal point of view, the services generally offered are considered elastic services. This means that it is possible to provide new resources to the services in an elastic way according to current demand. The main key factor of the rapid growth is that a high number of underlying technologies (virtualization, server farms, web services, web portals, etc.) which have reached their prime.

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* [7]) 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 [37] 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 both the hardware and software level, in incipient technologies such as virtualization [5]. 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 System (MAS) have not played an important role in the development of the Cloud paradigm. According to Talia [38] it is possible to distinguish to groups: (i) MAS that use the computational features of a Cloud environment (processing, storage, etc.) [25, 32]; 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) [25, 33].

In short, 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. MAS are suitable to help in the decision making about how vary dynamically these resources because its features (dynamicity, flexibility, autonomy, proactivity, learning, etc.) are exactly the features that a Cloud environment needs for the self-management of its resources. 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 which is development by the BISITE Research group¹. This platform allows to offer services at the PaaS (Platform as a Service) and SaaS (Software as a Service) levels. Both PaaS and SaaS layers are deployed using the internal resources of the cloud, in other words, the physical and virtual machines which provide a virtual hosting service with automatic scaling and functions for balancing workload. 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.

¹ <http://bisite.usal.es/en>

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 [7] in the underlying computational features.

Historically, the term *Cloud Computing* was first used by Professor Rammath [9]. 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* [36]. 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 [26].

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 [7, 16], following the guidelines proposed by Utility computing [31]. Additionally, the Cloud user does not have to be concerned with demand peaks, transforming passive investments in operational expenses [4].

A large number of definitions [4, 7, 16, 28] 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)² which defined Cloud Computing [5] 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.*

According to NIST, for a platform to offer Cloud Computing services, the services must contain the following characteristics [5]:

² <http://www.nist.gov/>

- Services on demand, meaning that services, regardless of their type, must be provided automatically and without human interaction according to user demand.
- Availability of services through the internet, meaning that clients should access the services through the internet and providers, as a result, must use this medium to provide their services.
- Availability of resources, meaning the provider must be able to offer services independently of their demand, using physical or virtual hardware resources assigned dynamically to each resource and reassigned according to demand. In this respect, there are authors such as [7, 43] who speak directly of high availability services, technology closely related to high availability computing.
- Elasticity, meaning that the different resources should be provided elastically and even automatically according to demand.

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 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. [40]). 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) [38].

2.1 Existing Platforms

SearchCloudComputing³ provides a list of the 10 primary cloud computing providers, which include: VMWare, Microsoft Azure, Bluelock, Citrix, Joyent, Terremark y

³ <http://searchcloudcomputing.techtarget.com/photostory/2240149038/Top-10-cloud-providers-of-2012/1/Introduction>

Amazon. Furthermore, from an academic and research scope, Corderiro et al. [13] propose the platform Euronet to interconnect multiple virtual laboratories using a Cloud environment. Given that the majority of Cloud platforms are proprietary and that the underlying infrastructure is invisible to researchers, Nurmi et al. [30] presents EUCALYPTUS, an open source framework for cloud computing that implements IaaS functions. Another proposal is provided in Malik et al. (2012), which presents a 3C model (Cooperative Cloud Computing) for research centers and universities. This model is based on the Virtual Cloud model and expects to generate a vast repository of computational resources for research centers.

All of these platforms have a very specific scope; that is, there is no platform that permits offering infrastructure, platform and software services in an integrated way. The scope of this study broaches this problem, facilitating the adoption of this paradigm by:

- developing the concept of multitenancy, which facilitates externalizing private and public Clouds to computing centers.
- developing elasticity models independent of the underlying technology, or very weakly coupled.
- developing an automated adaptation model for the internal hardware infrastructure, whether virtualized or not.

With regard to a negotiation model, cloud computing introduces a change in how to exploit and market a company's products. The model for acquiring hardware or software becomes a subscription model, or a service consumption model, which is essentially the same thing. This means that instead of acquiring pertinent computational resources, providers are hired instead. It is an attractive option for businesses, as it eliminates the requirements for the future planning of resources and permits beginning at the bottom and increasing resources only when they are actually needed. In their study, Artmbrust et al. [4] identify three case studies in which cloud computing is preferred to traditional storage: when demand for services varies over time, when it is not possible to foresee demand of services, and the computational efficiency that results from multiple machines.

2.2 Cloud and Agents

In a complex environment, such as that proposed in this project, it is difficult to determine when and how to carry out actions that imply changes in the operation or even the structure of the network. In the area of Distributed Artificial Intelligence, specifically in MAS technology, one of the goals is to create systems capable of making decisions in an autonomous and flexible way, and cooperating with other systems inside an organization. MAS technology is regarded as a potential technology to cope with the anticipated challenges of hybrid network operations. An analysis of the possibilities and benefits of implementing MAS shows that it is a suitable technology for the complex and highly dynamic operation of grid infrastructures, cloud computing, power systems or hybrid networks, among others [1, 21–24, 27].

Cloud environments require innovative architectures with advanced functionalities. To reach this objective, it is necessary to develop new functional architectures capable of providing adaptable and compatible frameworks, allowing access to services and applications regardless of location restrictions. A functional architecture defines the physical and logical structure of the components that make up a system, as well as the interactions between those components [17].

However, as indicated in the introduction, there are only a limited number of studies in the state of the art that relate Cloud Computing and agent technology [38]. 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 [11, 15] that use the computational strength of the environment to perform simulations in different fields. Another example is presented in [10], 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 [29] 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 [20, 39] that develop a Cloud service using agents for different specific purposes. Mong Sim used the Cloudle [29] which is an agent-based tool for discovering Cloud services. Some notable examples of Cloud providers combining resources include studies by Kaur Grewal [18] and Aarti Singh [35], which use shared Cloud resources to offer Infrastructure as a Service (IaaS) Examples that apply SLA to distribute services include [26, 31]. 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 [19].

Recent tendencies have led to the use of Virtual Organizations (VOs), which can be considered as a set of individuals and institutions that need to coordinate resources and services across institutional boundaries. Therefore, a VO is an open system formed by the grouping and collaboration of heterogeneous entities; the separation between form and function that exists among them requires defining how a particular behaviour will take place. Multi-agent systems (MAS) technology, which allows forming dynamic agent organizations, is particularly well suited as a support for the development of these open systems. An open MAS organization modelling makes it possible to describe structural composition (i.e. roles, agent groups, interaction patterns, role relationships) and functional behaviour (i.e. agent tasks, plans or services), it can also incorporate normative regulations for controlling agent behaviour, dynamic entry/exit of components and dynamic formation of agent groups [3, 6, 8, 41, 42].

As a conclusion, the scope of this study is dealing with the open problem of re-source allocation over a Cloud computing paradigm. In this sense, one of the most

appropriated approaches is to use MAS based on the VO, due to features such as of intelligence and adaptation that can enrich the capacities of a current Cloud Computing platform.

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 platform does not offer service at IaaS level. The internal 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 Fig. 1.

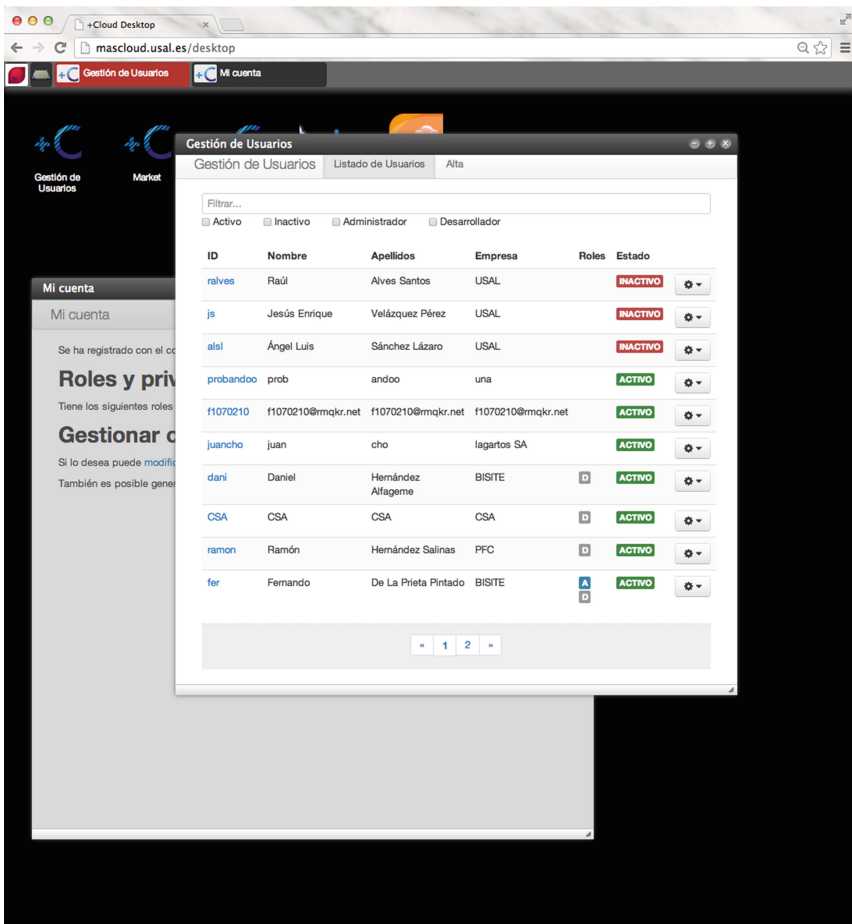


Fig. 1. SaaS in +Cloud

+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).

At this level, each user has a personalized virtual desktop from which they have access to their applications in the Cloud environment, and to a personally configured area as well. The virtual and physical resources are managed dynamically, but an overview of the internal resource can be seen through a specific web application.

- 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 services of the Platform layer are presented in the form of stateless web services. The data format used for communication is JSON [14], which is more easily readable than XML and includes enough expression capability for the present case. The existing service within the Cloud environment is:

- The FSS (File Storage Service) provides an interface to a file container, emulating a directory-based structure, in which the files are stored with a set of metadata thus facilitating retrieval, indexing, searching, etc. The simulation of a directory structure allows application developers to interact with the service as they would with a physical file system. A simple mechanism for file versioning is provided. Web services are implemented using the web application framework Tornado⁴ for Python.
- The OSS (Object Storage Service) is a document-oriented and schemaless database service, which provides both ease of use and flexibility. In this context, a document is a set of keyword-value pairs where the values can also be documents (is a nested model), or references to other documents (with very weak integrity enforcement). Nevertheless, documents are not forced to share the same structure. A common usage pattern is to share a subset of attributes among the collection, as they represent entities of an application model. The allowed types of data are limited to the basic types present in JSON documents: strings, numbers, other documents and arrays of any of the previous types. As with the FSS, the web service is implemented using Python and the Tornado framework. By not managing file downloads nor uploads, there is no need to use the reverse proxy that manages them in every node.
- The Identity Manager is in charge of offering authentication services to both customers and applications. The main capabilities of this service are (i) Single sign-on web authentication mechanism for users. This service allows the applications to check the identity of the users without implementing the authentication themselves; and (ii) REST calls to authenticate application/users and assign/obtain their roles in the applications within the Cloud.

⁴ <http://www.tornadoweb.org/>

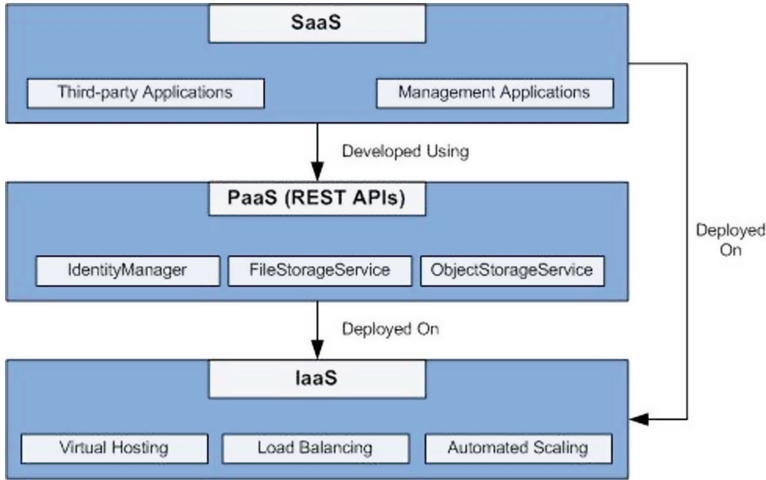


Fig. 2. +Cloud layer overview

The internal 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. It consists of a set of physical machines (servers) which contribute to the system by means of their computational resources (processing capacity, volatile memory, etc.). This level 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 Fig. 2.

The distribution of physical resources between the different virtual machines and between the different system services is a matter of current interest [19, 29, 35]. 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

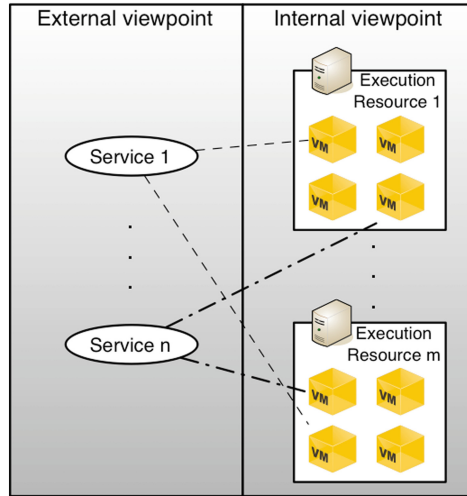


Fig. 3. Cloud computing deployment

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 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.

This agent keeps a vector with the information of the virtual machine (VR) hosted by the physical server, where each virtual machine is characterization by an id, kind of server, assigned computational capacity, assigned memory, state of the virtual machine, current use of the processing capacity and memory load.

$$VR = \{id, t, p, m, s_{VR}, u_p, u_m\}$$

In the same way, this role has an element (FR) with the characterization of the virtual server with the IP, the maximum number of virtual processors, the memory, the available memory, the minimum level of memory needed by the machine, the state of the physical machine, the percentage of current CPU use and the percentage of current memory use.

$$FR = \{ip, N, M, m_H, k_H, s_{FR}, u_p, u_m\}$$

- *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 [12] 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.

The problem of redistribution of resources at micro level is the following:

$$Problem = \{\{FR, n_p, n_m\} | \{VR, n_p, n_m\}_1, \{VR, n_p, n_m\}_2, \dots, \{VR, n_p, n_m\}_n\}$$

Where n_p y n_m are the necessities of memory and processing capability by each virtual machine.

- *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 Regular 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 [19]. 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 supervisor*, 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*, 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. 4).

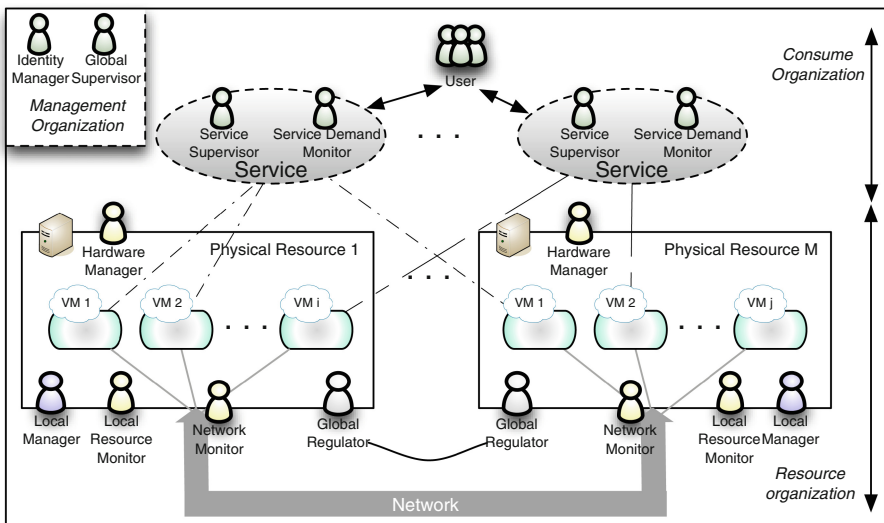


Fig. 4. MAS for resource redistribution in +Cloud

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.

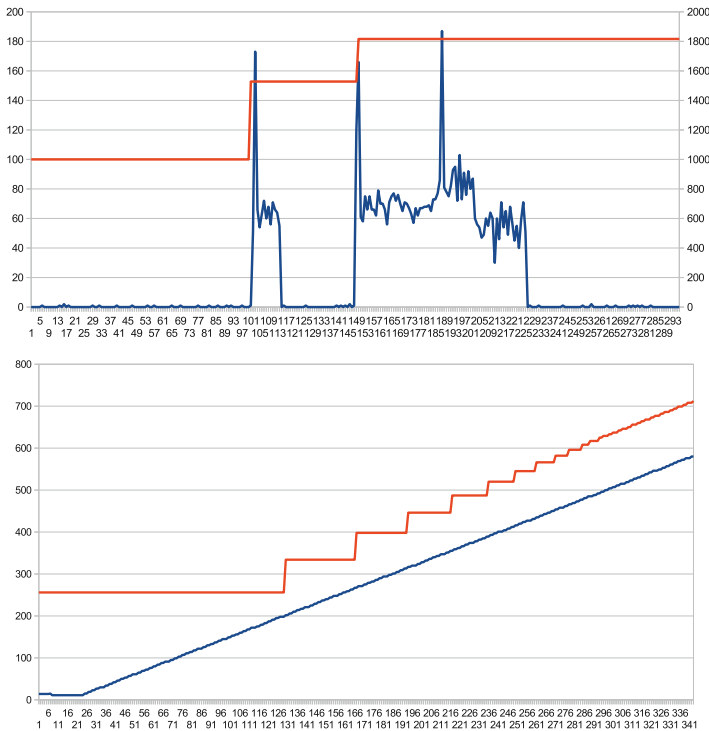


Fig. 5. Redistribution of resources at micro-level (upside: processor, downside: memory)

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 5 shows the adaptation of the system. The graph on the upside shows an increase in the use of the processor in percentage (blue line) and the corresponding increase of CPU assigned by the Local Manager (red line). The graph on the downside shows the same process in terms of gigabytes, 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.

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 [19], 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 6 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.

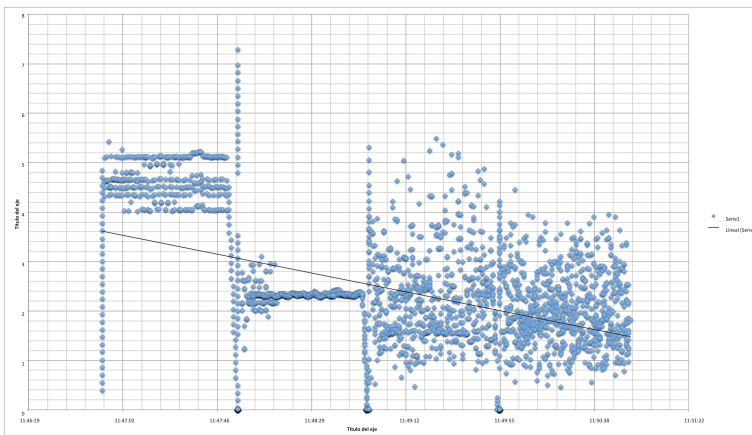


Fig. 6. Redistribution of resources at macro-level

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 [37], 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 [19], 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 advanced concepts of MAS derived from the latest MAS methodologies, such as the concept of environment or rules to manage the actions of the roles within the organizations. The use of this kind of methodologies 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) and European FEDER funds, project iHAS: Intelligent Social Computing for Human-Agent Societies (TIN2012-36586-C03-03).

References

1. Abras, S., Ploix, S., Pesty, P., Jacomino, P.: A multi-agent home automation system for power management. In: Cetto, J.A., Ferrier, J.-L., dias Pereira, J.M.C., Filipe, J. (eds.) *Informatics in Control Automation and Robotics*. LNCS, vol. 15, pp. 59–68. Springer, Heidelberg (2008)
2. An, B., Lesser, V., Irwin, D., Zin, M.: Automated negotiation with decommitment for dynamic resource allocation in cloud computing. In: *AAMAS'10 Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, vol. 1, pp. 981–988 (2010)
3. Argente, E., Botti, V., Julian, V.: GORMAS: an organizational-oriented methodological guideline for open MAS. In: Gleizes, M.-P., Gomez-Sanz, J.J. (eds.) *AOSE 2009*. LNCS, vol. 6038, pp. 32–47. Springer, Heidelberg (2011)
4. 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. *Commun. ACM* **53**(4), 50–58 (2010)
5. 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*, Bolton Landing, NY, USA, pp. 164–177 (2003)
6. Búrdalo, L., Terrasa, A., Julián, V., Zato, C., Rodríguez, S., Bajo, J., Corchado, J.M.: Improving the tracing system in PANGEA using the TRAMMAS model. In: Pavón, J., Duque-Méndez, N.D., Fuentes-Fernández, R. (eds.) *IBERAMIA 2012*. LNCS, vol. 7637, pp. 422–431. Springer, Heidelberg (2012)

7. 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, 2009, CCGRID'09, pp. 5–13 (2009)
8. Carrascosa, C., Giret, A., Julian, V., Rebollo, M., Argente, E., Botti, V.: Service oriented MAS: an open architecture. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems, International Foundation for Autonomous Agents and Multiagent Systems, May 2009, vol. 2, pp. 1291–1292 (2009)
9. Chellappa, R.: Intermediaries in cloud-computing: a new computing paradigm. In: INFORMS, Cluster: Electronic Commerce, Dallas, Texas (1997)
10. Chen, C., Wang, K.: Cloud computing for agent-based urban transportation system. *IEEE Intell. Syst.* **26**, 73–79 (2011)
11. 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)
12. Corchado, J.M., Pavón, J., Corchado, E.S., Castillo, L.F.: Development of CBR-BDI agents: a tourist guide application. In: Smith, I., Faltings, B.V. (eds.) EWCBR 1996. LNCS, vol. 1168, pp. 547–559. Springer, Heidelberg (2004)
13. Cordeiro, R.C., Fonseca, J.M., Donellan, A.: Euronet lab a cloud based laboratory environment. In: Global Engineering Education Conference (EDUCON), 2012, pp. 1–9. IEEE (2012)
14. Crockford, D.: The application/json media type for javascript object notation (json) (2006)
15. Dębski, R., Byrski, A., Kisiel-Dorohinicki, M.: Towards an agent-based augmented cloud. *J. Telecommun. Inf. Technol.* **6**, 16–22 (2012)
16. Erdogmus, H.: Cloud computing: does Nirvana hide behind the Nebula? *IEEE Softw.* **26**(2), 3–6 (2009)
17. Franklin, S., Graesser, A.: Is it an agent, or just a program? a taxonomy for autonomous agents. In: Jennings, N.R., Wooldridge, M.J., Müller, J.P. (eds.) ECAI-WS 1996 and ATAL 1996. LNCS, vol. 1193, pp. 21–35. Springer, Heidelberg (1997)
18. Grewa, R.K., Pateriya, P.K.: A rule-based approach for effective resource provisioning in hybrid cloud environment. *Int. J. Comput. Sci. Inform.* **1**, 101–106 (2012)
19. 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. *Prog. Artif. Intell.* **1**(4), 277–290 (2012)
20. Jin Kim, M., Gun Yoon, H., Ku Lee, H.: IMAV: an intelligent multi-agent model based on cloud computing for resource virtualization. In: Lee, R. (ed.) *Computers, Networks, Systems, and Industrial Engineering. Studies in Computational Intelligence*, vol. 365, pp. 99–111. Springer, Heidelberg (2011)
21. Karnouskos, S., de Holanda, T.N.: Simulation of a smart grid city with software agents. In: Third UKSim European Symposium on Computer Modeling and Simulation, 2009, EMS'09, pp. 424–429 (2009)
22. Kok, J.K., Warmer, C.J., Kamphuis, I.G.: PowerMatcher: multiagent control in the electricity infrastructure. In: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'05), pp. 75–82. ACM, New York (2005)
23. Lagorse, J., Paire, D., Miraoui, A.: A multi-agent system for energy management of distributed power sources. *Renew. Energy* **35**(1), 174–182 (2010)
24. Li, Y., Luo, Z.: A cooperative ad hoc routing based on cluster agent. In: IEEE Conference: 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), Wuhan, People's Republic of China, pp. 23–25 (2011)

25. Li, Z., Chen, C., Wang, K.: Cloud computing for agent-based urban transportation systems. *IEEE Intell. Syst.* **26**(1), 73–79 (2011)
26. Lohr, S.: Google and IBM join in cloud computing research. *New York Times* (2007)
27. McArthur, S.D.J., Davidson, E.M., Catterson, V.M., Dimeas, A.L., Hatziaargyriou, N.D., Ponci, F., Funabashi, T.: Multi-agent systems for power engineering applications—part II: technologies, standards, and tools for building multi-agent systems. *IEEE Trans. Power Syst.* **22**(4), 1753–1759 (2007)
28. Mell, P., Grance, T.: The NIST definition of cloud computing. In: *NIST Special Publication*, 800-145, pp. 1–3. NIST (2011)
29. Mong Sim, K.: Agent-based cloud commerce. In: *IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 717–721 (2009)
30. Nurmi, D., Wolski, R., Grzegorzczak, C., Obertelli, G., Soman, S., Youseff, L., Zagorodnov, D.: The eucalyptus open-source cloud-computing system. In: *9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009, CCGRID'09, May 2009*, pp. 124–131 (2009)
31. Ross, J.W., Westerman, G.: Preparing for utility computing: the role of IT architecture and relationship management. *IBM Syst. J.* **43**(1), 5–19 (2004)
32. Schuldt, A., Hribernik, K., Gehrke, J.D., Thoben, K.D., Herzog, O.: Cloud computing for autonomous control in logistics. In: Ehrich, H.-D. (ed.) *GI Jahrestagung. LNI*, vol. 175, pp. 305–310. Springer, Heidelberg (2010)
33. 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)
34. Siebenhaar, M., Nguyen, B., Lampe, U., Schuller, D., Steinmetz, R.: Concurrent negotiations in cloud-based systems. In: Vanmechelen, K., Altmann, J., Rana, O. (eds.) *GECON 2011. LNCS*, vol. 7150, pp. 17–31. Springer, Heidelberg (2012)
35. Singh, A., Malhotra, M.: Agent based framework for scalability in cloud computing. *Int. J. Comput. Sci. Eng. Technol. (IJCSSET)* **3**, 41–45 (2012)
36. 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'01*, pp. 149–160 (2001)
37. 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)
38. Talia, D.: Clouds meet agents: towards intelligent cloud services. *IEEE Internet Comput.* **16**(2), 78–81 (2012)
39. Venkataramana, K., Padmavathamma, M.: Agent-based approach for authentication in cloud. *IRACST – Int. J. Comput. Sci. Inf. Technol. Secur.* **2**(3), 598–603 (2012)
40. Wang, L., Tao, J., Kunze, M., Castellanos, A.C., Kramer, D., Karl, W.: Scientific cloud computing: early definition and experience. In: *HPCC'08: 10th IEEE International Conference on High Performance Computing and Communications*, pp. 825–830 (2008)
41. Zato, C., Sanchez, A., Villarrubia, G., Rodriguez, S., Corchado, J.M., Bajo, J.: Platform for building large-scale agent-based systems. In: *2012 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, pp. 69–73 (2012)

42. Zato, C., Villarrubia, G., Sánchez, A., Barri, I., Soler, E.R., del Viso, A.F., Sánchez, C.R., Cabo, J.A., Álamos, T., Sanz, J., Seco, J., Bajo, J., Corchado, J.M.: PANGEA - platform for automatic coNstruction of orGanizations of intElligent agents. In: Omatu, S., De Paz Santana, J.F., Rodríguez-González, S., Molina, J.M., Bernardos, A.M., Rodríguez, J.M.C. (eds.) *Distributed Computing and Artificial Intelligence. Advances in Intelligent and Soft Computing*, vol. 151, pp. 229–239. Springer, Heidelberg (2012)
43. Zhang, Q., Cheng, L., Boutaba, R.: Cloud computing: state-of-the-art and research challenges. *J. Internet Serv. Appl.* **1**(1), 7–18 (2010)