A Multi-Agent based Architecture for Cloud Infrastructure Auto-adaptation

Hanen Chihi[1], Walid Chainbi[2] and Khaled Ghedira[3]
[1]University of Tunis, Higher Institute of Computer Science/SOIE, Tunis, Tunisia
[2]University of Sousse, Sousse National School of Engineers/SOIE, Sousse, Tunisia
[3]University of Tunis/SOIE, Tunis, Tunisia

Keywords: Cloud Computing, Self-adaptation, Autonomic Computing, Multi-Agent Systems.

Abstract: Cloud computing including hardware, software, communication and networks are growing towards an ever increasing scale and heterogeneity, becoming overly complex. To manage such growing complexity, autonomic computing focuses on self-adaptable computing systems to the maximum extent possible without human intervention or guidance. In this paper, we design a multi-agent-based architecture for Cloud infrastructure auto-adaptation. This model is based on distributed multi-agent systems which collaborate to enrich the Cloud with self-* capabilities. The proposed model represents an effective method to reduce servers' power consumption while achieving the required performance and providing trusted Cloud.

1 INTRODUCTION

In the recent years, Cloud computing has attracted considerable attention (Mell and Grance, 2011). The concept of Cloud computing abstracts the runtime infrastructure to the user. It is a type of distributed system consisting of a set of interconnected and virtualized computers based on service-level agreements established through negotiation between providers and users. In this way, users will be able to access Cloud data and applications through Internet anywhere, at any time, on-demand and pay-per-use (Hurwitz et al., 2009).

To adapt the dynamic behaviours of Cloud infrastructure and to meet performance requirements, fault tolerance, reliability, security, etc., without manual intervention, techniques and methodologies are needed to be designed and implemented. Autonomic Computing (AC) may help providers and users to reach this goal. AC is a suitable candidate to endow the Cloud with self-* capability (Solomon et al., 2010).

An autonomic system is a set of autonomic elements, which implement intelligent control loops to monitor, analyse, plan and execute using knowledge of the environment. Several research efforts focused on enabling the autonomic properties address four main areas (IBM, 2003): Self-Configuring, Self-Optimizing, Self-Healing, Self-Protecting.

Many researchers have addressed Cloud autonomic behaviors at all levels, from the hardware level to software systems and applications (IBM, 2003). At the hardware level, data-center and operating system may be dynamically adapted and reconfigured. At the application level, databases, application server and web servers may be dynamically reconfigured to satisfy needed performance. Moreover, efforts have also focused on autonomic middleware, programming systems and runtime (Van et al., 2009).

Deriving from distributed artificial intelligence technology, multi-agent technology rising recent years is a fusion of distributed computing, artificial intelligence and other disciplines of programming methods. Agent-based approaches have been a source of technologies to a number of research areas. These include distributed planning and decision-making, communication languages, automated auction mechanisms, coordination mechanisms and learning mechanisms. Moreover, agent technology offers key advantages for the development of AC systems as it supports autonomy, adaptability, etc.

Motivated by the fact that agent technology is already ready for being integrated into a framework of AC (Chainbi, 2010), Multi-Agent Systems (MAS) can be used as basic components for implementing

Chihi H., Chainbi W. and Ghedira K..
A Multi-Agent based Architecture for Cloud Infrastructure Auto-adaptation.
DOI: 10.5220/0004374700900095
In *Proceedings of the 3rd International Conference on Cloud Computing and Services Science* (CLOSER-2013), pages 90-95
ISBN: 978-989-8565-52-5

intelligence in Clouds making them more adaptive, flexible, autonomic and smarter.

This work focuses on the Infrastructure-as-a-Service delivery Cloud computing class. It proposes a general multi-agent-based architecture for Cloud infrastructure adaptation. Agents are used to help the intelligent provisioning of basic resources to user applications, optimize the use of infrastructure provided as services and the management of the essential hardware maintaining the requested QoS, protect and repair Cloud infrastructure.

The remainder of this paper is organized as follows. Section 2 presents the related work. Section 3 describes the proposed MAS-based architecture for Cloud infrastructure management. Cases studies are detailed in Section 4. Finally, we conclude our study in Section 5.

2 RELATED WORK

Over the last decade a number of researchers describe the use of intelligent agents in service oriented architecture based system (Li et al., 2008); (Overeinder et al., 2008); (Chainbi, 2010). Agents can effectively and automatically provide some basic levels of computer and network defense that could be applied to Cloud infrastructure. In addition, intelligent agents offer various solutions to control access and authentication, distributed trust management, audit and intrusion detection, and diagnostic and system restoration.

Mazur et al. (Mazur et al., 2011) propose an autonomic monitoring defensive mechanism for Cloud integrating intelligent agents, computational intelligence, and ontologies. Distributed intelligent agents collect data within the Cloud by monitoring devices, data streams and code execution. This information is handled in the form of ontology-based models. Within the system, these ontologies can be operated on individually or brought together. Authors propose a new method that can enrich each generation of ontology during its creation by using related, known data to seed the process.

Collazo-Mojica et al. (Xabriel et al., 2012) propose a SOA API, in which users provide a Cloud application model and get back possible resource allocations in an IaaS provider. The solution emphasizes the assurance of quality of service (QoS) metrics embedded in the application model. An initial mapping is done based on heuristics, and then the application performance is monitored to provide scaling suggestions.

Cao et al. (Cao et al., 2009) propose a service oriented QoS-assured Cloud architecture that includes physical and virtual resources for Cloud service provisioning. They propose an autonomic strategy that assures users' request QoS.

Kim et al. (Kim et al., 2011) propose an intelligent multi-agent model based on virtualization rules for Cloud resource management. This model automatically allocates resources suitable to optimize Cloud performance. It infers user's request by analyzing and learning user context information. This multi-agent model is composed by three main agents: User Agent, Gathering Agent, Agent Manager and Virtualization Register that synchronizes with distributed agent to manage log data from the creation of VM to its destruction.

Frincu et al. (Frincu et al., 2011) propose a self-adaptive distributed scheduling platform composed of MAS implemented as an intelligent feedback control loops supporting defined policies and exposing self-healing capabilities. To face the challenge of building an autonomous self-healing system, an adaptive inter-provider MAS was proposed. The scheduling module is able to offer fully distributed storage and communication mechanisms, support fault-tolerance using agents as recoverable modules, support autonomy by dynamically changing scheduling policy, and adapt the negotiation policy.

In (Zarrabi and Zarrabi, 2012), the author introduces a Cloud intrusion detection system which is developed based on Cloud computing and can make up for the deficiency of traditional intrusion detection. They introduce intrusion detection system as a Service in a Cloud to protect user network. It exploits some characteristics in network traffics that make it possible to extract the required data from the user network for evaluation. This architecture is intended to be scalable by allowing users to combine the features of different intrusion detection systems' services for more reliable solution.

Although current approaches offer significant solutions to the problem of Cloud adaptation, their main drawback is that they don't clearly describe the adaptation process of Cloud resources and how adaptation characteristics of Cloud are addressed (e.g. security, optimization, context change, etc.). In addition, existing works are interested in only one property of AC.

Our work focuses on the autonomic resource management on the Cloud environment. However, with an extended scope aimed at jointly considering resource healing and protection needs and also power consumption and energy costs.

Another characteristic of the proposed model is

the use of distributed MAS that communicate locally (intra-MAS communication) and in a global level (intrer-MAS communication).

3 MULTI-AGENT BASED ARCHITECTURE FOR CLOUD AUTO-ADAPTATION

Cloud infrastructure can be represented as a tree of components reflecting the spatial arrangement and interconnection of Cloud resources. Each level contains a particular category of resource. The Cloud central manager is located in the first level. The second level presents regions distributed over the world. The third level contains data-centers which are deployed in regions. The forth level comprises PM which are located in distributed data-centers. In each PM, at least one VM is deployed.

In this work, we propose to auto-adapt the Cloud infrastructure based tree structure. Therefore, we propose a distributed Cloud agents-based architecture that uses the tree-based Cloud infrastructure. Each node in the tree is considered as a MAS.

Components of different levels have different functions. The role of a component is ensured regarding its location by a MAS composed mainly by four agents: self-configuration agent, self-optimisation agent, self-protection agent and self-healing agent.

Each agent in the multi-agent-system integrates functionality for data management. They accept information from the next-level agents or previous-level agent, and continuously learn and updates according to their own environment. Cloud adaptation evolves as a result of VMs being created, migrated, and terminated regarding users' request. In addition, Cloud adaptation incorporates the optimization and the healing of Cloud resources and ensures Cloud trustworthiness.

Figure 1 shows a multi-agent Cloud infrastructure architecture which includes physical resources layer and virtual resources layers to support autonomic Cloud infrastructure. In particular, Figure 1 shows an example of an autonomic Cloud tree with three regions. The region (in the left) is composed of two data-centers. The data-center (in the left) contains two physical machines. The PM (in the left) hosts two VMs which deploy services used by customers.

In the multi-agent autonomic Cloud architecture, distributed agents interact via message exchanges with the environment in the form of perceptions/effects. Regarding the exchange of messages between agents, two types of message exchanges may take place:

- Exchange local messages between agents of a single MAS based on the means of communication at their disposal. For example, a self-optimization agent invokes a self-configuration agent of the same resource for reconfiguration purpose.
- Exchange of messages between agents located on remote MAS. For example, to ensure the migration of a VM, a VM-level agent can contact a DC-level agent to add a new PM which will host the migrated VM.

3.1 Self-configuration Agent

The proposed self-configuring module provides abstraction with complete customization of Cloud resources. Reporting directly to the hardware environment, the self-configuring agent at the IaaS comes down to managing resources. The most common configuring operations in IaaS are: resources' allocation, resources' release, resources' migration.

If the current active resources don't support users' request, the autonomic manager should update infrastructure by adding new resources. It deploys and startup resources in the IaaS systems. The starting operation of VM requires prior configuration.

If the autonomic agent detects that the VM has terminated services execution, it saves its context and stops it, then it updates the state of the agent of the previous layer. The same process is repeated for the PM and data-center level agents.

The self-configuration agent should ensure VM migration. The VM is moved from on PM to another while continuously running, without any remarkable effects from the point of view of the end users.

3.2 Self-optimisation Agent

The Cloud-based virtualization offers a margin to the administrator of the IaaS for intervention on PM. It allows, among others, the Cloud providers to implement different policies for the allocation or re-allocation of resources in order to make savings or to fulfill a contract customer. The IaaS should regularly scan its environment and reorganizes available VMs on machines and data-centers to free up some of them. This release of machines reduces the energy consumption and cost of IaaS.

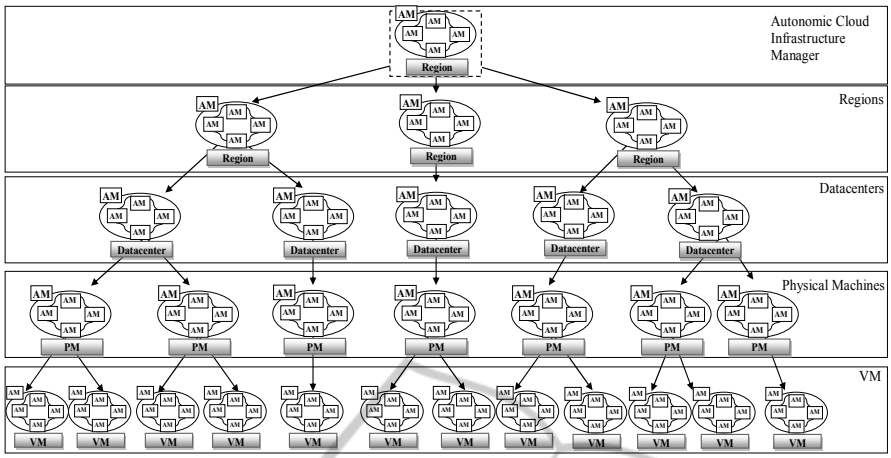

Figure 1: Multi-agent Architecture for Cloud Infrastructure Adaptation.

The self-optimisation agent collects and analyses some indicators (reliability, power quality…), then, it updates Cloud infrastructure by minimizing the number of active PMs.

3.3 Self-protection Agent

The self-protection agent is built around the IaaS model for providing security to any Cloud infrastructure. A self-protection agent is composed of these modules: detection, analysis and reaction.
Detection module is responsible for intrusion detection. It collects information form network and received by the resource. It selects items of interest and forwards them to the analysis module. Thereafter, the analysis module should take a decision and pattern matching plan. The reaction module is responsible for remote configuration and control of the resource by applying the defined protection plan.

3.4 Self-healing Agent

Self-healing agents are able to recover resource from failures. It is characterized by an intelligent feedback loop that supports data collection and analysing, module of recovery plan definition and module for applying recovery plan.

The recovery module analyses and defines several healing plans. Then according to collected information, it selects healing plan. Then the self-healing agent executes the selected recovery plan.

Infrastructure failures are failures of components that cannot be handled without interrupt services' running on it. In the proposed architecture, we propose, before dealing with recovery plan, to migrate connected VMs to others PMs.

4 CLOUD INFRASTRUCTURE ADAPTATION SCENARIOS

This section discusses the autonomic adaptation of Cloud infrastructure. For this, it is necessary to adapte VMs that will support the execution of cloud applications. Once instantiated, each VM is subject to an autonomic adaptation that ensures the configuration, optimisation, protection and healing of the distributed resources.

4.1 Self-configuration Scenarios

The self-configuration agent mainly focuses on the management of distributed resources in order to maximize the system utilization.

4.1.1 Resources Allocation Scenarios

Figure 2 describes an example representing adding new resources. The ACIM deduces the need for two additional VMs. A distributed agent decides that we need to create new PM in an existing data-center and deploys two VMs in order to satisfy users' request.

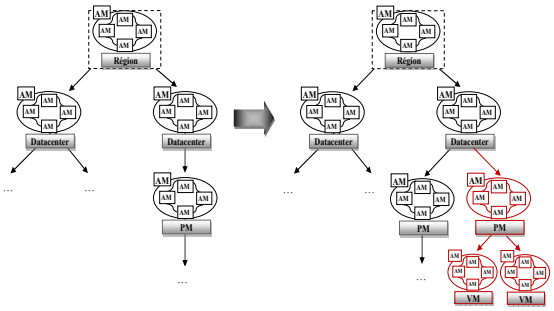

Figure 2: Allocate new VM and PM resources.

4.1.2 Resources Release Scenario

In the VM level, self-configuration agents verify the state of active resource. If the agent indicates that the resource (VM, PM or DC) state is idle then the agent should save resource context, update PM agent context and stop the resource. Figure 3 represents an example of releasing resource task.

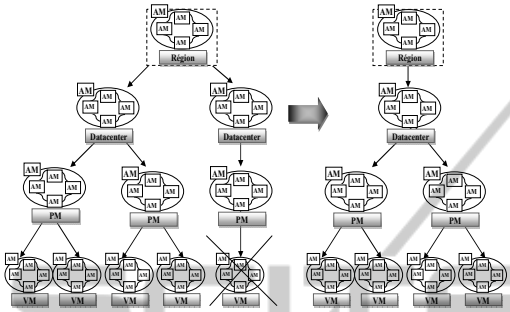

Figure 3: Release resource.

4.1.3 Resources Migration Scenario

The self-configuration agent of each VM verifies continuously the state of the associated VM. If the VM performance is lower than fixed threshold then the self-configuration agent creates needed VMs with appropriate features. Thereafter, agents of destination VMs and source VMs communicate information about application being running in the VM and they migrate VM context. Then, the self-configuration agent of the VM, PM and DC levels update their context and they stop the resource associated to the idle agent (see Figure 4).

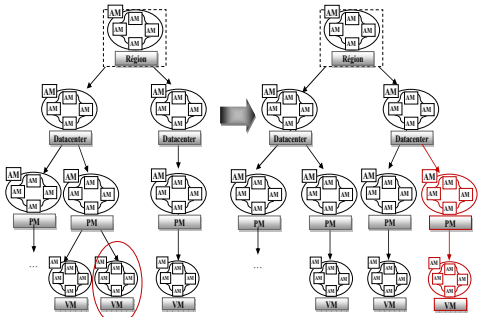

Figure 4: Migration of VM.

4.2 Self-optimization Scenario

The Cloud computing technology makes the resource as a single point of access to the client and is implemented as pay per usage. To reduce infrastructure cost, the proposed autonomic strategy self-optimizes the Cloud infrastructure by optimizing the used VM number.

Figure 5 presents an example of Cloud infrastructure optimization scenario. The optimization process consists of the following process. First, distributed MAS find PMs that are partially used, then, self-configuration agent of selected PMs migrate the associated resources in order to free up more resources. Then, self-configuration agents update their context and they stop the resource associated to the idle agent.

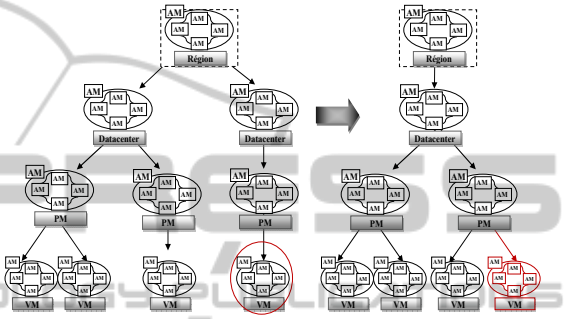

Figure 5: Self-optimization of Cloud resources.

4.3 Self-protection Scenario

At the VM and PM levels, each self-protection agent should collect continuously security information. Then, it analyzes them and executes the appropriate protection plan in order to:

- Ensure the availability of the information held within or between systems participants.
- Maintain information integrity exchanged within Cloud nodes.
- Detect and correct intrusions.

For example, if there are failed resources, then autonomic agents should apply a healing plan in a transparent way and let the Cloud infrastructure evolves in a safe manner.

4.4 Self-healing Scenario

The self-healing agent detects failed resources (Figure 6). The correction process begins by migrating application to other VMs then the self-healing agent reconfigures failed resources. Once resources are repaired they can be activated.

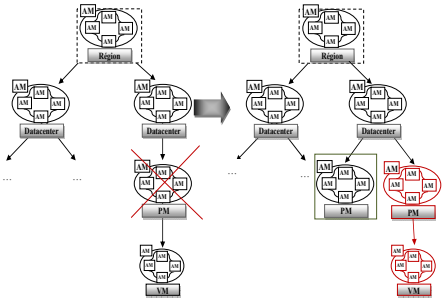

Figure 8: Self-healing of Cloud resources.

5 CONCLUSIONS

By combining AC and agent technology, this paper proposes a new multi-agent based architecture for Cloud infrastructure adaptation which includes physical resources (PM and data-center), virtual resource (VM) and an autonomic Cloud infrastructure manager. The proposed autonomic Cloud infrastructure strategy instantiates and coordinates the self-* capabilities for Cloud infrastructure adaptation.It is based on MAS that are composed mainly by four agents:

- Self-configuration agent able to allocate, release or migrate resources.
- Self-optimization agent used to optimize the amount of active resources in order to reduce energy consumption.
- Self-protection agent that ensures the credibility security of Cloud infrastructure.
- Self-healing agent that repairs failed resources in a transparent way for the users.

Moreover, we have analysed the behaviour of the proposed model through simulation scenarios.

For future work, we intend to expand our solution to propose a multi-objective algorithm for Cloud infrastructure self-optimization that dynamically modifies the Cloud infrastructure in order to reduce energy consumption.

REFERENCES

Cao, B. Q., Li, B., Xia, Q. M., 2009.A Service-Oriented Qos-Assured and Multi-Agent Cloud Computing Architecture, *CloudCom '09 Proceedings of the 1st International Conference on Cloud Computing*, 644 – 649.

Chainbi, W., 2010. An Agent-Based Methodology for Self-* Systems ", International Journal Multiagent and Grid Systems Journal, Volume 6, Number 1, pp. 55-69, IOS Press, 2010.

Frincu, M. E., Villegas, N. M.,Petcu, D., Muller, H.A.,Rouvoy, R., 2011. Self-Healing Distributed Scheduling Platform, 11th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing (CCGrid), 225-234.

Hurwitz, J. Bloor, R. Kaufman, M., Halper, F., 2009. Cloud Computing For Dummies, no.**2**.

IBM Group, 2003: An architectural blueprint for autonomic computing. http://www-03.ibm.com/autonomic/pdfs/AC.

Kim, M., Lee, H., Yoon, H., Kim, J. I., Kim, H., 2011. IMAV: An Intelligent Multi-Agent Model Based on Cloud Computing for Resource Virtualization, *International Conference on Information and Electronics Engineering, IPCSIT,* vol.6, 199-203.

Li, J.,Ma,D., Li L., Zhu,H., 2008. AADSS: Agent-based Adaptive Dynamic Semantic Web Service Selection,*Int. Conf. on Next Generation Web Services Practices NweSP 08)*.

Mazur, S., Blasch, E., Chen, Y., Skormin, V., 2011.Mitigating Cloud Computing Security Risks using a Self-Monitoring Defensive Scheme, *Aerospace and Electronics Conference (NAECON),* 39-45.

Mell, P., Grance, T. 2011. The NIST Definition of Cloud Computing (Draft), National Institute of Standards and Technology, vol.53, no.**6**, 1-7.

Overeinder, B. J.,Verkaik P. D., Brazier, F. M. T.,2008. Web service access management for integration with agent systems, *Proc.Of the 23rd Annual ACM Symp.on Appl. Computing, Mobile Agents and Syst. Track*, Mar. 2008.

Solomon, B., Ionescu, D., Litoiu, M., Iszlai, G. 2010. Designing autonomic management systems for Cloud computing, International Joint Conference on Computational Cybernetics and Technical Informatics (ICCC-CONTI) , 631–636.

Van, H. N., Tran, F. D., Menaud, J.M. 2009. Autonomic virtual resource management for service hosting platforms, Workshop on Software Engineering Challenges in Cloud Computing, 1-8.

Xabriel J., Collazo-Mojica, S., Sadjadi, M., Ejarque, J. Rosa, Badia, M., 2012. Cloud Application Resource Mapping and Scaling Based on Monitoring of QoS Constraints, Knowledge Systems Institute Graduate School, 88-93.

Zarrabi, A., Zarrabi, A., 2012. Internet Intrusion Detection System Service in a Cloud, *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 5, No 2, 308-315.