Model-Based Application Deployment on Cloud Computing

Aouat Asmaa, University of Oran1 Ahmed Ben Bella, Laboratory of Parallel, Embedded Architectures and High Performance, Oran, Algeria
Deba El Abbassia, University of Oran1 Ahmed Ben Bella, Laboratory of Parallel, Embedded Architectures and High Performance, Oran, Algeria
Benyamina Abou EL Hassan, University of Oran1 Ahmed Ben Bella, Laboratory of Parallel, Embedded Architectures and High Performance, Oran, Algeria
Benhamamouch Djilali, University of Oran1 Ahmed Ben Bella, Laboratory of Parallel, Embedded Architectures and High Performance, Oran, Algeria

ABSTRACT
Cloud Computing refers to a set of technologies and systems that provide various types of resources (computing, storage, software, etc.) on demand, through the Internet or Intranet. Thanks to these advantages many Cloud providers are available and is increasing. These cloud providers offer different PaaS platforms that must each be configured in its own appropriate way to deploy applications in the cloud. Cloud Computing is based on heterogeneity principles, which allows many configurations and sizing choices. This implies that the developer must master all deployment methods in order to benefit from all suppliers. The development and deployment of applications in the Cloud offers a new scientific challenge in terms of expression and taking into account variability. The purpose of the author’s work is to propose a deployment method and implement it to automate the process of deploying applications in a cloud environment based on model-driven engineering, to configure and provision applications to be deployed in the cloud.

KEYWORDS
Application Configuration, Application Deployment, Automation, Cloud Computing, Container, Docker, Model Driven Engineering, Runtime

1. INTRODUCTION
Cloud computing is an Internet revolution that offers many possibilities in various fields and more specifically in the processing and analysis of large data generated by the Internet. For technical and/or financial reasons, many individuals and organizations have chosen to move their servers or applications to a cloud environment in order to optimize the use of their IT infrastructure, scale-up, high availability, etc. Thus, the exploitation of infrastructure resources in a shared or pooled environment generates cost savings and increased performance.

Organizations that want to migrate (Orue-Echevarria, 2013) their applications in a cloud environment, must have qualified personnel who are familiar with the deployment and redeployment process to the various cloud providers because the latter generally do not provide any functionality to help users deploy and configure their applications. As well as organizations must master redeployment mechanisms for implementing fail-safe requirements for cloud services against hardware failures and

DOI: 10.4018/IJDST.2019040106
disasters. Cloud providers highlight the main problem of technological inconsistency for procurement between providers. For example, it is difficult to use the advantages of several providers at once to satisfy all customer needs (avoid breakdowns, available in all geographical areas, reasonable cost, availability of services). In this case, a software developer is required to master the deployment process of each cloud provider in order to exploit the advantages of each of them.

However, the most effective way to help individuals and organizations migrate their applications in a cloud environment, reduce human effort and facilitate application interoperability between vendors is to automate deployment processes (Orue-Echevarria, 2013). To this end, the objective of this paper is to propose a solution for configuring and deploying applications on different cloud providers. Several approaches and solutions have been proposed in this framework and are based on different technologies such as; Scripts (Mutiara, Refianti, & Witono 2014), Workflows (Calcaterra, Cartelli, Di Modica, & Tomarchio, 2017), Domain Specific Languages (Loope, 2011) and Model-based Approach (Ribeiro, Rocha, Santos, & Moreno, 2016). We have opted for the model-driven approach because this type of approach is designed for large applications that require major configuration changes.

In addition to the general introduction and conclusion, this document contains four sections. The second section discusses existing approaches and tools in the context of automatic deployment of applications on cloud computing. The third section introduces the concept of application deployment and implementation technologies, the fourth section is a conceptual elaboration of the solution adopted for automatic deployment. The fifth section is an implementation of the proposed deployment method and an evaluation of the product described in this manuscript against the related tools.

2. RELATED WORK

Many efforts have been made to help developers deploy and manage their applications on heterogeneous PaaS and IaaS platforms. These efforts are presented in APIs form and could be organized into three categories. These categories depend on approaches (Talwar et al., 2005) on which the majority of deployment APIs are based.

Many tasks in the management of systems and application operations are already automated using scripts. These scripts are usually copied by hand to the target system on which they are executed. Compared to plans, these scripts can be considered micro-flows: small isolated jobs that can be executed quickly and do not require transactional support (Szyperski, 2003). This type of approach reduces the risk of human error during the manual deployment process, and for the developer, the developer takes care of application development instead of a cloud computing configuration.

This type of approach reduces the risk of human error during the manual deployment process, and for the developer, the developer takes care of application development instead of a cloud computing configuration. Generally, when applications are small or system configurations rarely change, the script approach is the reasonable solution. Among the works that have adopted this approach is the work of MUTIARA (Mutiara et al., 2014).

A DSL language (Domain Specific Language) is a programming language or executable specification that aims to provide a high level of expressiveness focused on a specific domain. To do this, it offers notations and abstractions adapted to the field in question, thus facilitating their use compared to that of a more general programming language. Automatic management of application configuration is an area that has resulted in a very large number of jobs (Mernik, Heering & Sloane, 2015). In this work, the configuration is represented, explicitly, by means of a DSL. Often the largest applications in which changes involve dependencies, the language-based approach is the best choice. Among these solutions Puppet (Loope, 2011) is one of the most widely used today.

The workflow solution proposes that the developers have the services they need to deploy as well as creating the software deployment UML models. These models define all the information required for the deployment (virtual machines, services, applications, dependencies, operating systems of the virtual machines, services repositories and virtual machines, databases, services provider and
16 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the product's webpage:

www.igi-global.com/article/model-based-application-deployment-on-cloud-computing/226964?camid=4v1


www.igi-global.com/e-resources/library-recommendation/?id=2

Related Content

Collaborative e-Learning and ICT Tools to Develop SME Managers: An Italian Case
www.igi-global.com/chapter/collaborative-learning-ict-tools-develop/64456?camid=4v1a

Gridifying Neuroscientific Pipelines: A SOA Recipe and Experience from the neuGRID Project
www.igi-global.com/chapter/gridifying-neuroscientific-pipelines/45562?camid=4v1a
A Generic Reference Architecture for Collaboratory Scientific Virtual Laboratory
www.igi-global.com/article/a-generic-reference-architecture-for-collaboratory-scientific-virtual-laboratory/78734?camid=4v1a

Discovering Perceptually Near Granules
www.igi-global.com/chapter/discovering-perceptually-near-granules/44710?camid=4v1a