DESIGN THE TOSCA OF CLOUD RESOURCES

[1]SNEHAL V. RAUT, [2]H. R. DESHMUKH

[1]M E. Scholar, Department of Computer Science, Dr. Rajendra Gode Institute of Technology and Research, Amravati, India
[2]Professor & HOD, Department of Computer Science, Dr. Rajendra Gode Institute of Technology and Research, Amravati, India
E-mail: [1]snehalraut5002@Yahoo.Com,

Abstract - With the advent of Cloud Computing, organizations are increasingly migrating their information and communication technology (ICT) resources to the cloud. Cloud computing is driving formidable change in the technology industry and transforming how to do business in around the world. [4] Using TOSCA, the cloud providers are able to define the interoperable description of services and their relationships, and to enable the portability and automated management across cloud platforms and infrastructures. However the verification of the cloud orchestration design with TOSCA is still crucial to ensure and alert when the safety properties of the cloud design are violated. In this paper, to proposed the formal verification of cloud resources design and also to describe the all the cloud resources.

I. INTRODUCTION

Topology and Orchestration Specification for Cloud Applications (TOSCA) [1] is a standard for cloud based web services design and the processes to manage them. TOSCA is developed and approved by the Organization for the Advancement of Structured Information Standard (OASIS). It helps construct agility and reduce restriction with responsibility in resource management. Using the TOSCA standard, the cloud service providers are capable of describing their cloud computing infrastructure and architecture and manage the implementation of the cloud resources. There are several cloud orchestration tools providing the automation of the deployment of implementation.

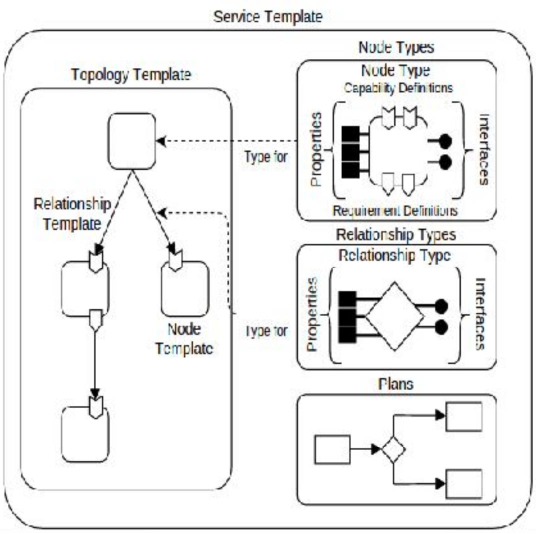

Figure 1: Structural elements of a service template and their relations [11]

II. LITERATURE REVIEW

Karn Yongsiriwit et. al. [1] proposed a semantic framework tackling this heterogeneity issue. We develop a set of ontologies to semantically represent cloud resources by looking at three cloud resource description standards: Topology and Orchestration Specification for Cloud Applications (TOSCA), Open Cloud Computing Interface (OCCI), and Cloud Infrastructure Management Interface (CIMI). Hence, our framework promotes the creation of a common semantic knowledge base of cloud resources described using these different standards. This knowledge base allows a seamless translation of cloud providers' resource descriptions. We developed an application to validate our approach as a proof of concept. We also evaluated the feasibility and completeness of our semantic framework on use cases obtained from standard specifications.

Warun Chareonsuk et. al. [2] proposed an alternative mean to do the formal verification of the cloud orchestration design by superimposing the relevant BPEL of web services over the existing TOSCA description of a cloud orchestration. The resulting formal model of the superimposition between BPEL and TOSCA defines not only the orchestration of the web services but also their service interfaces and the corresponding high level behaviors of the services. In this paper, the safety properties of the cloud orchestration are focused only and defined using the linear temporal logic formula. Our formal model is correctly written in Promela and formally verified using model checker SPIN.

Rawaa Qasha et. al. [3] shows how TOSCA, a new standard for cloud service management, can be used to systematically specify the components and life cycle management of scientific workflows by mapping the basic elements of a real workflow onto entities specified by TOSCA. Ultimately, this will enable workflow definitions that are portable across clouds, resulting in the greater reusability and reproducibility of workflows.

Beniamino Di Martino et. al. [4] this paper we propose an overview of two solutions for cloud services description and orchestration, TOSCA (Topology and Orchestration Specification for Cloud Applications) and HOT (Heat Orchestration Template), a comparison among these two solution and examples of how these two solutions correlate with cloud patterns.

Bisera Ivanovska et. al. [5] uses the P-TOSCA model for other issues that are also very important in virtualized datacenters and cloud computing, that is, to enlarge/extend the energy efficient management system. A prototype application that dynamically creates a target virtual machine on utilized physical compute node, ports the application(s) from a virtual machine hosted on an underutilized physical server to the target virtual machine in Eucalyptus cloud is presented, which is specified with P-TOSCA. After migration, the prototype application will shut down the underutilized empty physical node.

III. PROPOSED ALGORITHM

Algorithm

Step 1: Register user on cloud
Step 2: Fetch MAC address of user device and store against user details
Step 3: Detect user choice to access the resources from cloud
Step 4: Select resource uploaded to cloud by user
Step 5: Fetch attributes of resource
Step 6: Classify resource according to the attributed fetched by system
Step 7: Upload resource to cloud and record store in DB
Step 8: Configure TOSCA control panel by specifying resource selection criteria
Step 9: Fetch list of resources from cloud according to user choice and TOSCA control panel configuration.
Step 10: Maintain usage for resource for every user
Step 11: Report the usage and choice made by user

IV. RESULT ANALYSIS

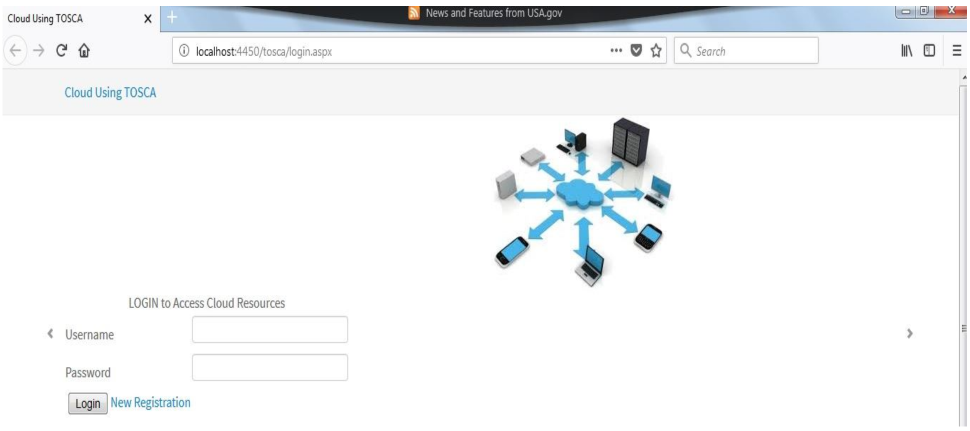

Figure 2: Home Page

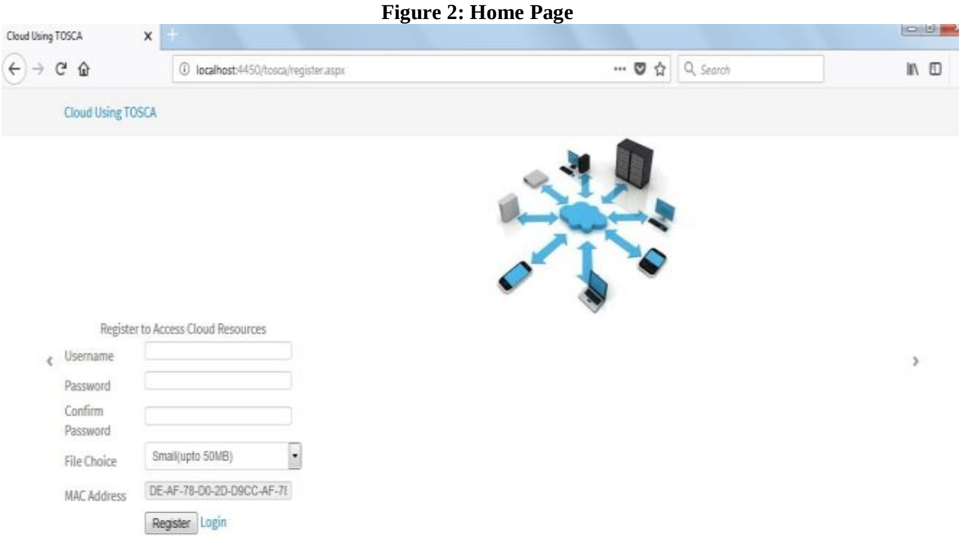

Figure 2: User must registered to cloud

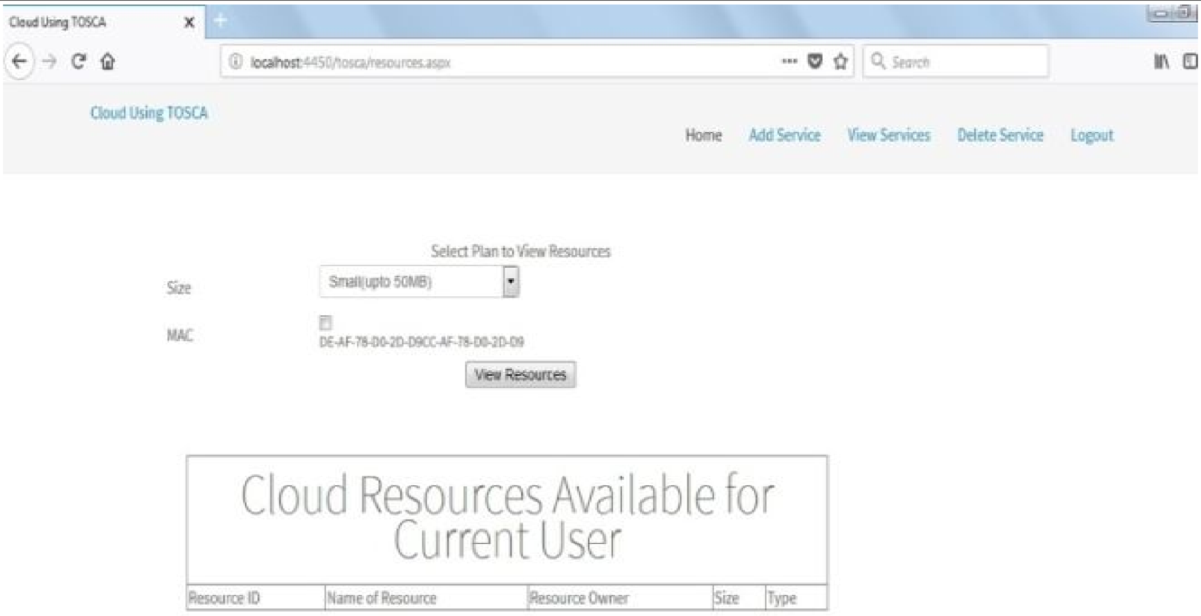

Figure 4: Resource available for User

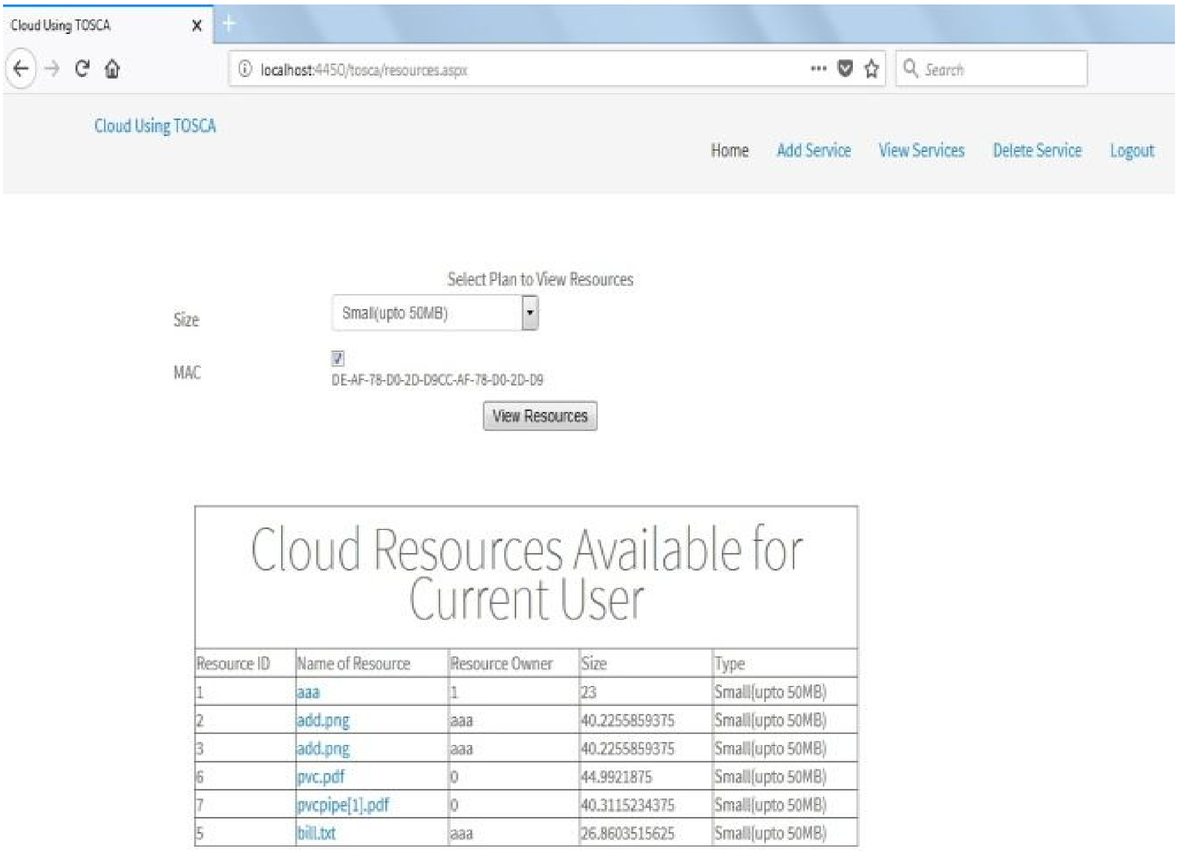

Figure 5: To Show all Resources available for users

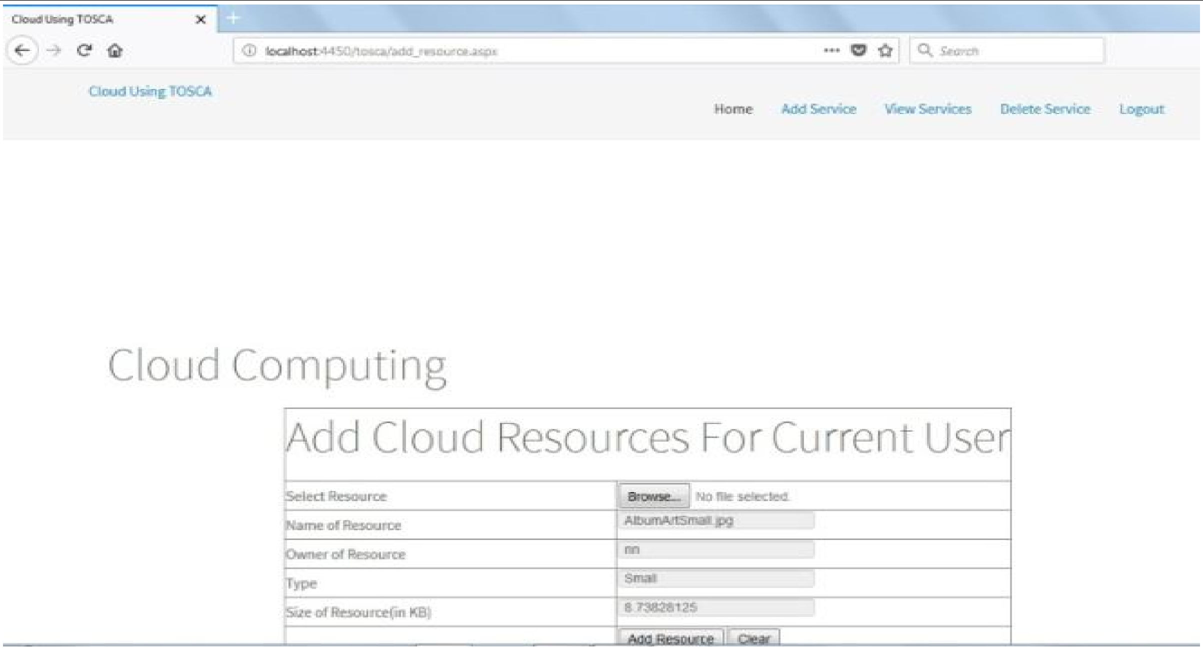

Figure 6: User can add resources to cloud

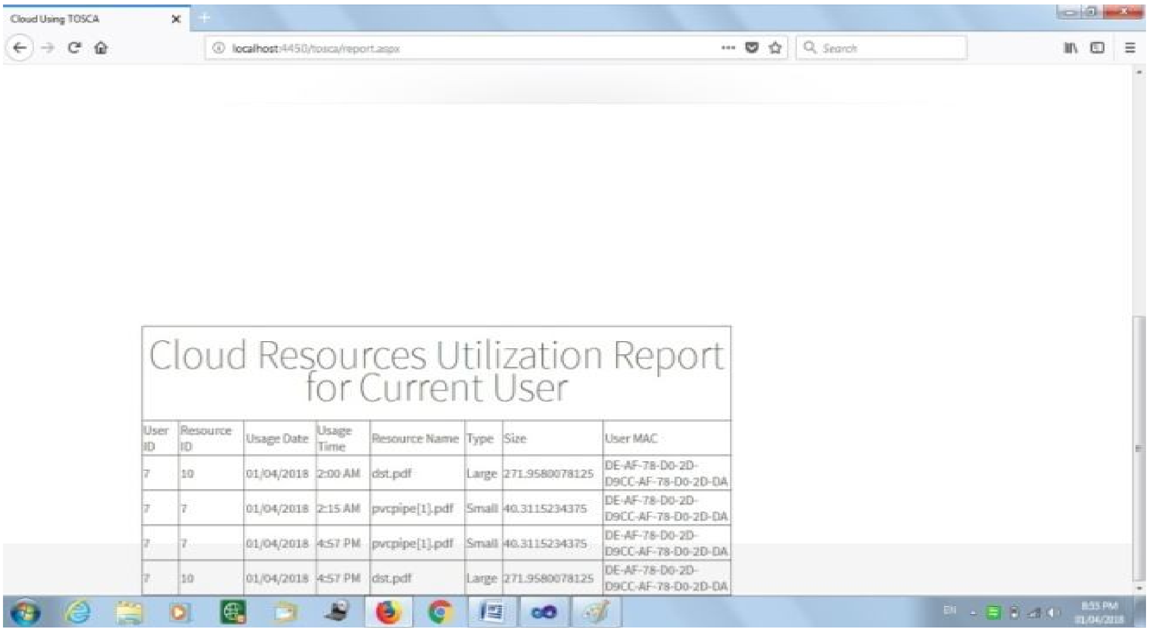

Figure 6: To show the utilization report of cloud resource of current users

V. CLOUD APPLICATION DEPLOYMENT WITH TOSCA

A cloud application in TOSCA consists of a number of connected nodes called a service template. The nodes can represent horizontally distributed functions, vertical stacks of software/middleware dependencies or both. Nodes are the building blocks of a TOSCA cloud model, shaping the way an application is designed and the way in which imperative scripts are linked to applications and resources. You can visualize TOSCA templates as a series of connected stacks of nodes, each representing an application element.

CONCLUSION

In this paper, formal verification of cloud orchestration design is described and also presents the different services of TOSCA. Cloud Computing environment, ways to define machine readable standards for the description of Cloud services and their orchestration are highly desired features. The capability of application packaging which enables the reusability and portability of applications or part of them is an important prerequisite to obtain full benefits from Cloud Computing. With the cloud, resource pools can use virtualization to abstract away the heterogeneousness and regional distribution of manufacturing resources.

REFERENCES

[1] Karn Yongsiriwit, Mohamed Sellami, Walid Gaaloul, "A Semantic Framework Supporting Cloud Resource Descriptions Interoperability", IEEE 9th International Conference on Cloud Computing, 2016.

[2] Warun Chareonsuk, Wiwat Vatanawood, "Formal Verification of Cloud Orchestration Designwith TOSCA and BPEL", IEEE, 978-1-4673-9749-0/16/$31.00, 2016.

[3] Rawaa Qasha, Jacek Cała, Paul Watson, "Towards Automated Workflow Deployment in the Cloud using TOSCA", IEEE, 8th International Conference on Cloud Computing, 2015.

[4] Beniamino Di Martino, Giuseppina Cretella, Antonio Esposito, "Defining Cloud Services Workflow: a Comparison between TOSCA and OpenStack Hot", IEEE, Ninth International Conference on Complex, Intelligent, and Software Intensive Systems, 2015.

[5] Bisera Ivanovska, Sasko Ristov, Magdalena Kostoska, Marjan Gusev, "Using the P-TOSCA Model for Energy Efficient Cloud", IEEE, MIPRO,Opatija, Croatia, 25-29 May 2015.

[6] Florian Haupt, Frank Leymann, Alexander Nowak, Sebastian Wagner, "Lego4TOSCA: Composable Building Blocks for Cloud Applications", IEEE, International Conference on Cloud Computing, 2014.

[7] Christos Tsigkanos, Timo Kehrer, "On Formalizing and Identifying Patterns in Cloud Workload Specifications", 13th Working IEEE/IFIP Conference on Software Architecture, 2016.

[8] Soheil Qanbari, Fei Li, and Schahram Dustda, "Toward Portable Cloud Manufacturing Services", IEEE Computer Society, November/DECEMBER 2014.

[9] Abdul Razaq, Huaglory Tianfield, Peter Barrie, Hong Yue, "Service Broker Based on Cloud Service Description Language", IEEE, 15th International Symposium on Parallel and Distributed Computing, 2016.

[10] Alexandru-Florian Antonescu, Philip Robinson, Torsten Braun, "Dynamic Topology Orchestration for Distributed Cloud-Based Applications", IEEE, Second Symposium on Network Cloud Computing and Applications, 2012

[11] Shin Nakajima, Lightweight formal analysis of Web service flows. In Progress in Informatics Journal, 2005.

★★★