SAVI Testbed: Control and Management of Converged Virtual ICT Resources

Joon-Myung Kang, Hadi Bannazadeh, and Alberto Leon-Garcia
Department of Electrical and Computer Engineering
University of Toronto, Toronto, ON, Canada
Email: {joonmyung.kang, hadi.bannazadeh, alberto.leongarcia}@utoronto.ca

***Abstract*—This paper introduces the SAVI testbed for control and management of converged virtual ICT resources. The SAVI testbed attempts to create an open applications marketplace that operates on converged virtual computing and networking infrastructures and supports experimentation with emerging Future Internet protocols and applications. In this paper we briefly review the Smart Applications on Virtual Infrastructure (SAVI) network and the Virtualized Application Networking Infrastructure (VANI) which are the basis for the SAVI testbed. We then present the architecture and design of the SAVI testbed. Finally, we report on the current status of development and share our experience with the SAVI testbed.**

I. Introduction

Many nations are investing in national-scale research programs focused on the Future Internet and applications [1]. They have been addressing content-oriented paradigms, mobility and ubiquitous access to networks, cloud-computing-centric architectures, security, and experimental testbeds. The *Smart Applications on Virtual Infrastructure (SAVI)* project (involving nine universities and multiple industrial partners) has been established with a focus on future application platforms designed for applications enablement. A major obstacle to innovation in network aspects of application platforms is the inability to test and deploy new Internet protocols and applications at scale.

In the US, the Global Environment for Networking Infrastructure (GENI) is a major effort to create a testbed by advancing and federating different testbeds such as PlanetLab and Emulab on top of a dedicated research network [2]. GENI follows a slice-based architecture, and different testbeds will be able to connect to each other through GENI wrappers. Other related efforts include ProtoGENI [3] based at the University of Utah and Future Internet Research for Sustainable Testbed (FiRST) [4]. Recently, ExoGENI [5] has been proposed to link GENI to two advances in virtual infrastructure services outside of GENI: open cloud computing (OpenStack) and dynamic circuit providers, through their native Infrastructure-as-a-Service (IaaS) interfaces, that provide links to other GENI tools and resources. In Europe, Scalable and Adaptive Internet Solutions (SAIL) [6] addresses design concepts and technologies for the networks of the future and develops techniques to move from today's to future networks. BonFIRE [7] offers a multi-site cloud testbed that supports large scale testing of applications, services and systems over multiple, geographically distributed, heterogeneous cloud testbeds. BonFIRE includes OpenNebula based testbeds and non-OpenNebula based testbeds.

In this paper we present the SAVI Testbed (TB) that enables experimentation in future applications platforms and future Internet alternatives by controlling and managing converged computing and network resources. SAVI investigates multi-tier cloud computing where infrastructure includes core massive datacenter cloud computing and additional computing resources at many smart edges. SAVI has developed a smart-edge cluster that provides converged ICT resources in support of computing, networking, and applications, and a control and management framework to create virtual networks. These virtual networks provide researchers with virtual nodes, consisting of associated virtual resources, at different SAVI Smart Edge sites that are interconnected with research network connectivity to support experimentation at scale. SAVI nodes incorporate open source software and hardware (OpenStack [8], Openflow [9], NetFPGAs, and others as they become available). To enable experiments that require high performance, SAVI provides a method to facilitate the programming of FPGA resources. In this paper we present the current status and share our experience in demonstrating SAVI capabilities in a workshop setting.

The paper is organized as follows. Section II briefly reviews the SAVI network and our previous work on testbed development. Section III presents the SAVI system architecture and a design for integrating ICT resources in the SAVI testbed. In Section IV, we present our experience from a hands-on workshop for the SAVI Edge Node. Finally, conclusions and future work are presented in Section V.

II. SAVI Network and VANI Overview

A. SAVI Network

The objective of the SAVI project is to address the design of future application platforms built on flexible, versatile and evolvable infrastructure that can readily deploy, maintain, and retire the large-scale, possibly short-lived, distributed applications that will be typical in the future applications marketplace. An application platform brings together the infrastructure (personal devices, wireless and wired access networks, Internet, and computing clouds) and software resources required to deliver applications. In a future setting, the end user will typically use the application platform through a mobile device that connects to a ubiquitous very-high-bandwidth, integrated wireless/optical access network. The application platform provides network connectivity to services that support the application of interest. Most services will be supported by massive-scale distant data centers located at sites of renewable energy. Some services will require low latency (e.g. alarms in smart grids, safety applications in transportation, monitoring in remote health) or the processing of large volumes of local

978-3-901882-50-0

information (e.g. video capture in lecture rooms) provided by intelligent converged network and computing resources at the "Smart Edge" of the network, for example in the premises of traditional telecom service providers.

SAVI considers the extended computing cloud to be multi-tiered and to include a Smart Edge, and it investigates the interplay between the Smart Edge and remote massive-scale data centers in the delivery of applications. The need for a Smart Edge tier is driven by the latency requirements of time sensitive requirements in smart grids as well as safety and fast response applications in transportation and health. The Smart Edge is also made necessary by the high cost and delay involved in transferring large volumes of information across the network core. SAVI investigates the hypothesis that all computing and networking resources can be virtualized and managed using Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) principles. The SAVI Smart Edge includes a greater variety of resources than conventional datacenters. The SAVI project also investigates the virtualization of the wireless access network and its control in the smart edge. This aspect however will not be discussed in this paper.

B. VANI Testbed

The SAVI testbed builds on the earlier Virtualized Application Networking Infrastructure (VANI) testbed [10]. VANI is a testbed that allows researchers to utilize virtualized resources to rapidly create and deploy networked systems and distributed applications. VANI developed a virtualized converged computing and networking cluster that can serve as a prototype for the Smart Edge node converged network node which assumes: 1) service-oriented application creation, 2) IaaS methods for configuring and scaling resources to support applications, and 3) virtualization of physical resources. The VANI node offers processing, storage, networking, and FPGA resources, and can be viewed as a hybrid between a computing cluster and a high-end router, where packet and higher-layer and application processing can be performed on shared resources accessed across a 10GE fabric. VANI is unique in testbed systems in using off-the-shelf components in converging computing and communications resources, and in allowing for the addition of additional resource types in its design. We developed a prototype of a node consisting of multiple programmable and non-programmable resources that have been virtualized as services that can be used and networked together [11]. The SAVI converged network cluster has been built on VANI by: 1) addressing outstanding design issues in VANI, 2) adding resources and capabilities to VANI, 3) adding federation to facilitate collaboration with data centers and testbeds, and 4) utilizing open source cloud computing and Software Defined Network (SDN) software to control conventional cloud and networking resources.

III. Architecture and Design of SAVI Testbed

In this section, we describe the system architecture for the SAVI TB. First, we describe a SAVI TB platform architecture including its main components and interfaces. Then, we present a reference architecture with two planes and their logical components: Control and Management (C&M) plane and Application and Experiment (A&E) plane. Finally, we present a design of SAVI TB for the Edge Node in the reference architecture.

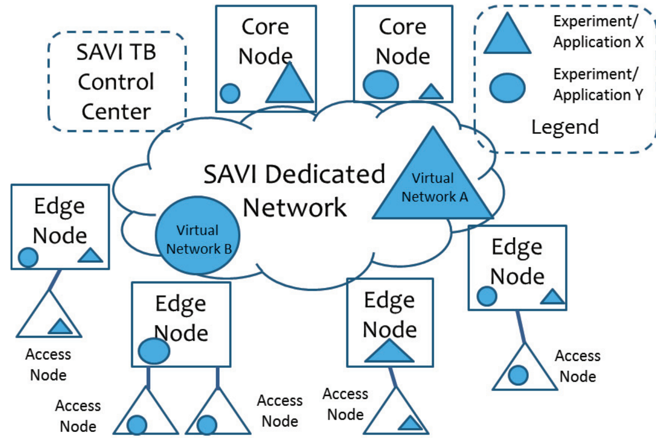

Fig. 1. SAVI TB main entities, hosting two sample applications

A. High-Level Architecture

The SAVI TB platform architecture, including its main components and interfaces, is described here. The interfaces described in this paper are for both internal (between components of the SAVI TB) and external (between the SAVI TB and external entities) communications. The SAVI TB is comprised of five main physical entities: 1) Core Nodes, 2) Smart Edge Nodes, 3) Access Nodes, 4) SAVI Network, and 5) SAVI TB Control Center.

Figure 1 depicts a high level view of these main entities. Core nodes, Edge Nodes and Access Nodes contain resources that are used for creating an application. Examples of these resources are compute, storage, network, optical access, wireless access and reconfigurable hardware resources (e.g. BEE4 and NetFPGA). The Core nodes are accommodated by conventional cloud computing resources (compute, storage and basic networking), while Edge nodes include more advanced resources such as reconfigurable hardware resources. The SAVI network is also considered a resource in the SAVI TB. Core nodes, Edge nodes, and SAVI TB control center are all inter-connected by the SAVI network which is a dedicated research network. The Access Nodes are connected to the SAVI Edge Nodes using dedicated links. Core and Edge Nodes together are referred to as the extended cloud in SAVI. A SAVI node is defined as a set of physical resources that are interconnected by an internal high-speed fabric with negligible latency. In contrast, two separate SAVI nodes interconnected through SAVI TB dedicated network would have significantly higher latency.

In a typical SAVI deployment, Edge Nodes are deployed on sites located at participating universities. The Access Nodes also are deployed on various universities. Note that, it is possible for a university to only have Access Nodes and be connected to an Edge Node in a different university. The Core Nodes, on the other hand, are deployed fewer universities compared to the Edge Nodes. For instance, there could be one or two Core nodes across the SAVI TB platform.

Applications and experiments are deployed on different components of SAVI TB. In SAVI, applications are different than experiments in a sense that they are aimed at delivering a feature to end users, and need to guarantee an agreed level of service, while experiments are comparably shorter-lived, used by a researcher, and aimed at gathering measurement data or user feedback. In essence, however, both applications and experiments are treated equally in/by SAVI TB, and are used interchangeably in the rest of this document.

An application is deployed on SAVI TB by allocating a slice of resources to that experiment or application. All SAVI resources are virtualized in SAVI TB, and their allocation to each application or experiment is performed by the TB C&M plane. A SAVI TB Control Center hosts clearinghouse functionalities described in the SAVI proposal such as Authentication, Authorization and Accounting (AAA). Moreover, it hosts parts of SAVI C&M plane functionalities that are platform-wide. The internal components of SAVI C&M plane and their responsibilities are explained in the next section.

In Figure 1, the SAVI TB Control Center is depicted apart from the rest of the entities. Although the SAVI TB Control Center can be separately deployed on a group of servers independent from other SAVI nodes, it could also be physically co-located/distributed on the Core and/or the Edge nodes. However, to emphasize on its testbed-wide (platform-wide) logical importance and authority, it has been depicted separately in this figure (with dashed line).

Figure 1 also shows how two sample Applications (X & Y) coexist in the SAVI TB by getting a slice of resources of Core nodes, Edge nodes, Access nodes and SAVI network. In the figure, application X, which is represented by a triangle, uses virtualized resources in all Core nodes and Edge nodes (except one Edge node). The amount of resource used by each Application is depicted by the size of application legend (e.g. triangle for Application X). For instance, Application X uses more resources in one of the Core nodes and Application Y (represented by circle) uses more virtualized resources on the other Core node. SAVI nodes are also securely connected to the Internet to access resources that are available outside SAVI.

From the logical point of view, the SAVI TB follows a two-plane basic reference architecture depicted in Figure 2. This two-plane architecture includes two main planes: C&M and A&E. The two-plane basic reference architecture depicts main components in SAVI-TB.

The C&M plane's main task is to allocate and secure a slice of virtualized resources to an application. The application then will run in the A&E plane over the allocated virtual resources throughout SAVI TB including any combination of Edge nodes, Core nodes, Access nodes, and network resources. It performs the traditional FCAPS management tasks (Fault, Configuration, Administration, Performance and Security) on SAVI resources, as well as control related tasks such as on-demand allocation/de-allocation of virtualized resources to/from experiments and applications.

The A&E plane is the place in which researchers can run their application on a set of virtualized resources allocated by the SAVI C&M plane. In this plane, applications are free to follow any architecture (e.g. peer-to-peer, star topology)

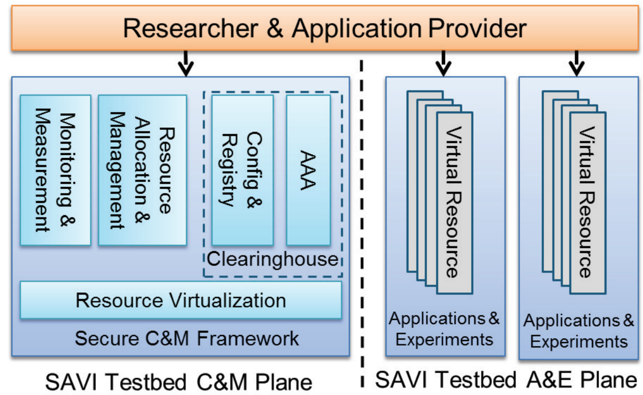

Fig. 2. SAVI basic reference architecture

that they wish to satisfy their specific design goals and requirements. Therefore, we try to enforce minimum constraints on the internal architecture and interactions in this plane. To facilitate rapid creation of (short-lived) applications, however, we try to provide reusable services (resources) required for future smart applications in this plane.

B. Design of SAVI Testbed for Edge Node

VANI provided several virtualized physical resources such as reprogrammable hardware, processing, storage, and fabric resources. Except for reprogrammable hardware resources, open source based integrated IaaS solutions can be utilized instead for virtualizing the VANI physical resources. The IaaS solutions provide a more holistic approach by integrating all of the necessary functionality in a single package including virtualization, management, interfaces, and security. For this reason, the SAVI TB adopts open source IaaS solutions where possible and retains VANI solutions where necessary, e.g. for programmable hardware.

To run experiments on the SAVI TB, we need the control and management plane. We define the management plane as the high-level manager for allocation of resources, and the control plane as the lower level unit dealing with physical components and resources. The management plane is used to manage the allocation of virtual slices of access networks, converged nodes and network bandwidth to virtual networks dedicated to specific experiments and/or applications running on the SAVI application platform testbed.

Figure 3 shows a conceptual layered design of SAVI TB for an Edge Node. In the Edge Node cloud computing resources, Edge Node network, and other SAVI resources are available as physical resources. We manage all physical resources by virtualizing in the virtualization layer. We provide control and management APIs for cloud computing resources thanks to OpenStack [8] which is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter. A network manager for Edge Node networks provides APIs for supporting an SDN based on Openflow [9] which is an open standard to deploy innovative protocols in production networks. Finally, the control and management APIs for other SAVI resources, such as reconfigurable hardware resources (FPGA or NetFPGA) or any type of

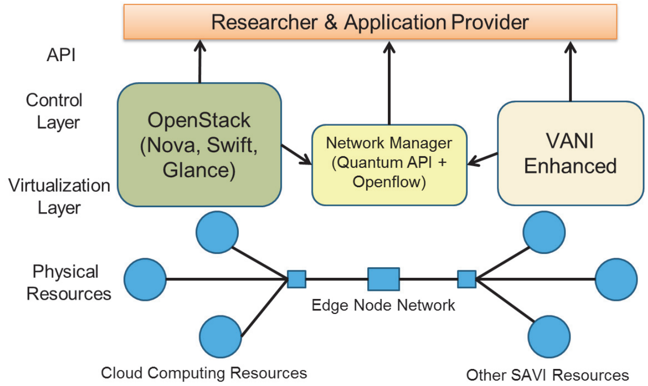

Fig. 3. Conceptual design of SAVI testbed for Edge Node

hardware-based acceleration resources, can be provided by a VANI-enhanced module which is the enhanced and improved version of VANI C&M design as described in Section II-B.

IV. Current Status and Experience

We have implemented the SAVI TB C&M software using Java and Python based on the design described in the previous section. We have also published all projects as an open source in a github public repository (https://github.com/savi-dev). We have put our SAVI TB core logic, resource management webservices, clients to the github repository.

In order to show feasibility of our SAVI TB, we organized a one-day hands-on workshop for fifty researchers and trained participants to use SAVI resources and develop useful applications through our SAVI TB. First, we deployed a SAVI TB client and seven SAVI Edge Nodes at our university. Each Edge Node had Intel i7 3.4GHz CPU, 12GB of RAM memory, 1TB SATA HDD and a running Ubuntu 12.04 server. We installed all SAVI TB clients and OpenStack clients at the SAVI TB client which was a main gateway for connecting our SAVI TB. In the workshop we demonstrated how to control and manage resources in the SAVI TB, develop cloud-based applications using the computing and storage resources, accelerate the applications using reprogrammable FPGA, control virtual networks using Openflow, and set up and operate the SAVI TB. Furthermore, we guided researchers to use computing, storage, and networking resources on SAVI TB and develop two sample cloud-based applications in the virtual computing instance. From the experience of the hands-on workshop, we enabled researchers to efficiently develop any applications and experiments using virtual resources provided on SAVI TB. We have shared a video including demonstration from the workshop using the SAVI TB[1]

V. Conclusion

In this paper we have presented the SAVI TB for applications and experiments by control and management of converged virtual ICT resources. First, we discussed the objectives of the SAVI network to provide a motivation of our TB work. We then described a system architecture of the SAVI TB based on improvements of VANI, our previous testbed effort. We used opensource software including OpenStack, Openflow, controllers in addition to in-house developed code based on VANI C&M to implement a practical testbed. Finally, we described how our SAVI TB C&M can be used to test and deploy applications and experiments and we reported on our hands-on workshop to university researchers and industry partners. As future work, we are designing platform-wide components for integrating all Edge Nodes, Core Nodes, and SAVI networks. We are also working on monitoring and measuring physical and virtual resources as a platform service.

[1] See the video titled "SAVI Testbed Workshop 2012 - High-Level Demonstration" at http://www.savinetwork.ca/research/documents/

Acknowledgment

This research was supported by NSERC of Canada. The authors would like to thank Hesam Rahimi, Mohammad Faraji, Soheil Hassas Yeganeh, Maxim Podlesny, Mohammad Faten Zhani, Rafael Esteves, David Mann, Thomas Lin, and Lilin Zhang for their good comments and contributions for SAVI TB design and development.

References

[1] J. Pan, S. Paul, and R. Jain, "A survey of the research on future internet architectures," *Communications Magazine, IEEE*, vol. 49, no. 7, pp. 26 –36, July 2011.

[2] "Global environment for network innovations (geni) project," http://www.geni.net, [Online; accessed 13-August-2012].

[3] "ProtoGENI," http://www.protogeni.net, [Online; accessed 13-August-2012].

[4] S. Lee, S. Han, J. Kim, and S. Lee, "FIRST: korean future internet testbed for media-oriented service overlay network architecture," *Journal of Internet Technology*, vol. 11, no. 4, pp. 554–559, july 2010.

[5] I. Baldine, Y. Xin, A. Mandal, P. Ruth, A. Yumerefendi, and J. Chase, "Exogeni: A multi-domain infrastructure-as-a-service testbed," in *TridenCom*, 2012.

[6] "Scalable Adaptive Internet Solutions (SAIL)," http://sail-project.eu, [Online; accessed 13-August-2012].

[7] A. C. Hume, Y. Al-Hazmi, B. Belter, K. Campowsky, L. M. Carril, G. Carrozzo, V. Engen, D. G. Pérez, J. J. Ponsatí, R. Kübert, Y. Liang, C. Rohr, and G. V. Seghbroeck, "Bonfire: A multi-cloud test facility for internet of services experimentation," in *TridenCom*, 2012.

[8] "Openstack community," http://www.openstack.org, [Online; accessed 13-August-2012].

[9] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[10] H. Bannazadeh, A. Leon-Garcia, K. Redmond, G. Tam, A. Khan, M. Ma, S. Dani, and P. Chow, "Virtualized application networking infrastructure," in *TRIDENTCOM*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 46. Springer, 2010, pp. 363–382.

[11] K. Redmond, H. Bannazadeh, P. Chow, and A. Leon-Garcia, "Development of a virtualized application networking infrastructure node," in *Proc. IEEE GLOBECOM Workshops*, 2009, pp. 1–6.