EXTENSION OF THE SUBMITTED PAPER AS COMPARED TO THE ORIGINAL ONE, PUBLISHED IN SOICT 2017

We are grateful to submit our paper in SOICT17 to a special section of Informatica journal. In this submitted paper, we have improved the presentation and clarity of the original one, as demonstrated in the submitted manuscript. In what follows, we explain the extension of the submitted paper to the original one with appropriate reference to the specific section/paragraph of this submitted manuscript:

1. We already corrected grammatical errors to make the paper clearer. In addition, we renamed the proposed work as **SHIOT: A novel SDN-based framework for the Heterogeneous Internet of Things**.

2. In the section 2, beside the related works, we added a brief introduction related to SDN:

2.1 Software Defined Networking

Over the past decade, the need for services that span multiple IoT application domains is growing in order to realize the efficiency gains, promised by the IoT. End-users, however, have to face the heterogeneity issue, which arises when a wide variety of devices, wireless communication solutions, and access technologies are implemented in the IoT. SDN is considered as a good solution to handle such issue because of its centralized and programmable controller that enables simple pro- grammatic control of the network data-path. The main idea of SDN is to separate the control and data planes. The control plane creates and modifies the forwarding rules, which are subsequently sent to network devices. The network devices (e.g., switches and routers) in turn, just forward packages based on the received rules. The controller is therefore considered as control logic precept to examine the overall network behavior. Using SDN-based controller, network administrators can easily program, manipulate and configure network protocols in a centralized way. Fig. 1 shows a general architecture of SDN, which consists of three main layers: Application layer, Control layer and Infrastructure layer.

In order to deploy a SDN architecture, it is essential to have an interface that ensures the communication between the data and control plane. Such interface is called South-bound Interface (SBI) and should be standardized. SBI defines a protocol to facilitate the diversity of network devices and controller software. There are variety of SBI protocols (e.g. ForCES [8]), but the most typical one seems to be OpenFlow [19]. An OpenFlow-enabled networking switch needs to maintain a forwarding flow table that has three types of information: rules, actions associated with each rule, and the statistics that count the number of packets and bytes for the flow. The OpenFlow-enabled switch also creates a secure channel to communicate with the SDN controller. We have chosen OpenFlow in the present work because it is capable of reducing management complexity, handling high bandwidth and implementing new policies when required. A detailed description on the advantages of OpenFlow can be found in [19].

3. We enrich the related works in adding more recent works from the field of SDN and IoT as follows.

Wei et al. [25] introduced a hash-based distributed strategy while integrating SDN into IoT in order to solve the problem of storage limitation of forwarding nodes. In their work, the multi-dimension selection method was utilized for finding the suitable storage. The hash space was formed by using the IoT data flow. However, the authors did not consider the QoS requirements related to the different IoT applications/services.

Sharma et al. [26] introduced DistBlockNet, a distributed secure SDN for the IoT, where blockchain technology was exploited to verify a version of the flow-rule table. How- ever, in their experiments, they did not consider the average end-to-end delay, the most important QoS parameter that has a significant impact on the user experience while running an IoT service.

4. We have reconstructed the ontology to illustrate the applicability of the proposed SHIOT. In the revised manuscript, we set our focus on the IoT that is implemented in E-healthcare system. We refer the reviewers to Section 3 for the detailed description:

The present work constructs an ontology and utilizes semantic technologies to describe the IoT context as well as the devices and their characteristics. We set our focus on the IoT that is deployed in E-healthcare system. Fig. 3 shows the constructed ontology, which involves three main classes as follows.

1. ***Applications**: We considers five different healthcare applications:*
    - *Monitoring: This is used to capture and record the various healthcare indicators including the physiological (i.e., ECG, EMG, EEG), chemical (sweat, glucose, saliva), and optical (oximetry, the properties of tissues) metrics.*
    - *Therapeutic: The goal is to monitor the treatment of a given disease. This consists of medication (drug delivery patches), stimulation (chronic pain relief) and emergency (defibrillator).*
    - *Fitness and Wellness: The application aims to observe the motion and location indicators such as physical activity, calorie count, GPS information and indoor localization.*
    - *Behavioral: This application is used to maintain regular surveillance over the patient's activities (fall, sleep, exercise), emotions (anxiety, stress, depression) and diet (calorie intake, eating habits).*
    - *Rehabilitation: This application is used to monitor the rehabilitation of patients like speech (language development) and camera (technology for blinds).*
2. ***Devices**: There are two main types of devices, i.e., on-body contact sensors and peripheral non-contact sensors.*
3. ***Positions**: The positions where the devices locate include the laboratory, operating rooms, casualty rooms, consulting rooms, day rooms, emergency rooms, pharmacy, high dependency unit, maternity ward.*

5. We also re-implemented the testbed using three Request Analysis PCs (RA1-3) and one coordinator PC for the load balancing (see Figure 5 in the revised manuscript). This implementation has been proven to be more effective through experimental results as illustrated in Section 4.2.

6. For the scalability, the reconstructed ontology is implemented with 250 devices and 60 rooms. We also implemented a denser network topology, having 150 nodes (abbreviated as 150N topology). The new obtained experimental results are impressive. The scalability of LARE has been proven in 150N topology, i.e., a large network having high node density, where the performance gap between SHIOT and the other methods become more significant. We have added lot of new experimental results in Section 4, which was divided into four parts: Section 4.1 describes in detail the testbed; Section 4.2 and 4.3 analyze the experimental results related to the *request analysis* and *routing* layers. Section 4.4 compares SHIOT with the traditional system that is deployed without SDN.