Application Level Network Virtualization using Selective Connection

Injung Hwang, and Dongkun Shin
Sungkyunkwan University, Suwon, Korea
{sinban, dongkun}@skku.edu

***Abstract*— Since Internet-of-Things (IoT) devices transfer several types of data such as sensor values, images, and videos, multiple heterogeneous network interfaces are required to handle different types of data communications. To utilize the multiple network interfaces efficiently, the handover technique is important. However, the current network systems cannot transfer data during the handover operation between heterogenous network interfaces. In addition, the application programmer must implement the handover operation. In this paper, we propose a novel network virtualization technique called s*elective connection*, with which an application-oblivious seamless switching between heterogeneous network interfaces is possible depending on the data type.**

I. Introduction

Generally, Internet-of-Things (IoT) devices collect several types of data, and transfer them to other devices via network interface. There are various network interfaces with different features on bandwidth, power consumption, communication range, etc. Therefore, it is important to choose the proper interface considering the requirements on the communication. For example, if only small-size sensor data are transferred intermittently, it is better to adopt the interface consuming low energy such as Bluetooth (BT) or Zigbee. Whereas, the high bandwidth network interface such as WiFi-Direct (WFD) is required to send large data such as images or videos.

Recent IoT devices such as drones and car dashboard cameras have various sensors which can collect temperature, humidity, GPS location, acceleration, image, etc. Therefore, it is hard to meet all requirements of different sensor data with a single network interface. To resolve this problem, recent IoT devices are equipped with multiple heterogeneous network interfaces, and the proper interface is enabled for the specific data type. The network interface must be selected considering several requirements such as bandwidth, power consumption, and communication range. For example, if the device transfers sensor values of small size infrequently and stays idle most of the time, we can use BT to reduce the energy consumption during the idle period. However, if the device must transfer the large data such as video clips, the WFD interface can be enabled instead of the BT to transfer the data in a high speed. In addition, the WFD interface is more energy-efficient than the BT when the transferring data is burst.

In such a device with heterogeneous network interfaces, the handover scheme between different network interfaces is important. Generally, the handover between homogenous network interfaces can be serviced by the network driver in the operation system kernel, the handover between different network interfaces must be managed by user applications because the different interface provides different network APIs to user-level applications. In addition, the data transfer operation must be halted during the handover operation. In a real-time case [1], in which data should be streamed constantly, the halted data transfer during handover is unacceptable. In this paper, we propose an application-oblivious network virtualization technique called s*elective connection*, which uses selectively a proper network interface among heterogenous interfaces while supporting data transfer during handover.

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the SW Starlab support program(IITP-2017-0-00914) supervised by the IITP(Institute for Information & communications Technology Promotion)

II. Related works

Many network management schemes are proposed, which exploit the advantages of different network interfaces. CoolSpot [2] is the hybrid solution, which uses both BT and WiFi. The technique focused on the policy of handover, i.e., when to switch, but did not handle the seamless switching. The policy is based on the bandwidth, power consumption, and communication range. Our selective connection scheme uses a simple policy based on the data size, so our scheme could be further improved with the technique of CoolSpot. WearDrive [3] is a hybrid network solution for wearable devices. When the wearable device synchronizes the data in its local storage with the remote storage in smartphone, the scheme uses BLE or WFD based on the I/O size. However, these studies did not handle the latency problem during the handover.

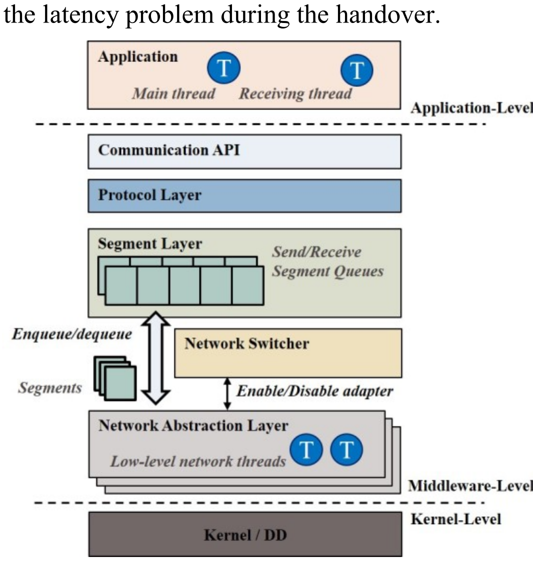

Fig. 1. Software Architecture of Selective Connection

978-1-5386-3025-9/18/$31.00 ©2018 IEEE

TABLE I
OVERHEAD OF SELECTIVE CONNECTION

RTT by Data Size	100KB	10MB
Without Selective Connection	210.98 ms	1,483.18 ms
With Selective Connection	208.70 ms	1,581.40 ms

III. SELECTIVE CONNECTION

Figure 1 shows the architecture of *selective connection* scheme. It is implemented as user-level libraries and threads and thus it is platform-independent. The application must generate a main thread and receiving thread. The threads use the *communication API* for send and receive operations. The APIs are simple and protocol-independent. When the data are transferred, they are divided into segments. A segment is 512 KB and it is the atomic unit of transferring through the network interfaces. The *Segment Layer* divides the bulk data from the user request into segments or assembles the segments into the bulk data. Two queues are managed in the segment layer, *send/receive segment queue*.

The *Network Abstraction Layer (NAL)* communicates with the kernel-level network device drivers. For each network interface (e.g., BT or WFD), the corresponding *NAL* is generated. A *NAL* has two low-level network threads for the data transfer. The *Network Switcher* enables or disables the *NAL* by observing the data traffic. It enables the high-bandwidth network adapter when there are too many segments in the segment queue, and disables it when there are few segments in the queue.

During a handover operation between different network interfaces, data can be transferred through multiple network adapters concurrently. Multiple low-level network threads fetch segments from a single queue. This enables a seamless network switching during the handover operation.

IV. EXPERIMENTS

We measured the overhead of our scheme. Table I compares the round-trip time (RTT) with two data sizes when only the WiFi interface is used with/without our scheme. When the size is large (i.e., 10 MB), a small overhead is shown although the data are divided into a lot of segments. When the data size is small, however, the selective connection shows no overhead.

Figure 2 shows the change of RTT during a handover operation between BT and WFD. At the scenario, after some idle period, the burst data transfer begins at the 3rd request, which sends 20 KB of data repeatedly without idle period. Each request usually took less than 5ms until the reply comes with BT. The dotted box area is the handover region. Since only the BT interface is enabled at the idle period, the RTTs are long and jittery. Therefore, our scheme enables the WFD interface at the 3rd request. However, there is an 8-second delay before WFD is turned on. During the handover, the BT services the communication. When WFD is turned on, the BT is disabled and RTTs decrease due to WFD.

Figure 3 shows the energy efficiency of the selective connection scheme. In this scenario, an application sends 10 KB of data repeatedly between 13 seconds of idle periods, and then it sends data of 5MB. The *active-small* and *active-5MB* mean the energy consumption while sending small data(10KB) and large data(10MB), respectively. We compared three schemes, BT-only, WFD-only, and our scheme with BT/WFD. The BT-only consumed lower energy during the idle periods than WFD, but it consumed more energy sending large data. The WFD-only scheme showed lower energy consumption during active periods. However, it showed large energy consumption during the idle period. Our scheme has low idle energy consumption and low active energy consumption by utilizing BT and WFD selectively.

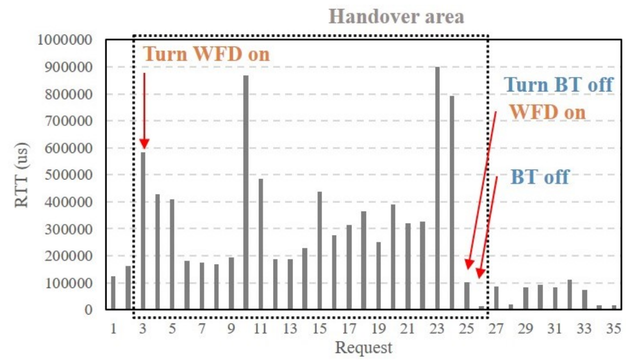

Fig. 2. RTT during Handover for streaming requests

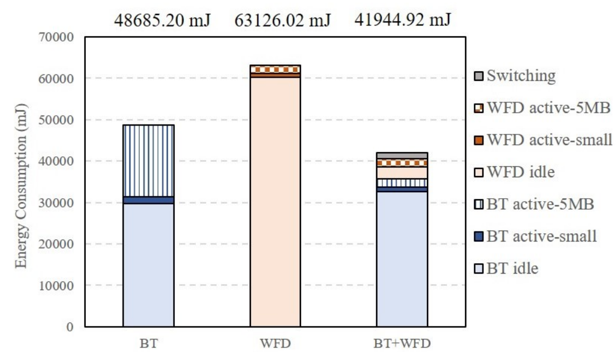

Fig. 3. Energy Consumption of each state with three schemes

V. CONCLUSION

By utilizing heterogenous network interfaces for different data types, we can improve performance and energy efficiency. However, it is difficult to implement the seamless handover operation at user applications. Our selective connection scheme enables the application-oblivious network switching for heterogenous network interface by providing user-level network virtualization. It can service data transfer during the handover operation.

REFERENCES

[1] Xing Liu et al, "Characterizing Smartwatch Usage In the Wild," *Mobisys'17.* NY USA, June. 2017.

[2] Trevor Pering, Yuvraj Agarwal, Rajesh Gupta, Roy Want, "CoolSpots: Reducing the Power Consumption of Wireless Mobile Devices with Multiple Radio Interfaces", *Mobisys'05,* June. 2005.

[3] Jian Huang, Anirudh Badam, Ranveer Chandra, and Edmund B. Nightingale, "WearDirve: Fast and Energy-Efficient Storage for Wearables", *ATC'15*, SANTA CLARA, July. 2015.