Development of User Plane Control for vEPC

TSUKAGOSHI Tsutomu, SAEKI Shuichi, TERADA Junpei, SUNAGAWA Yuta, SHIMAMOTO Hiroshi, IGUCHI Tomohito

Abstract

The key to achieving an effective SDN/NFV solution is the development of an efficient software-based means of handling user data transfer processing among mobile core network devices. We focus on the performance-related components of telecommunications networks such as high throughput and low latency, elucidating the technical problems involved in using standardized software to accomplish user data transfer processing - a task conventionally performed by customized, proprietary hardware. This paper describes how we have solved those problems and describes how to operate the system in a virtualized environment.

Keywords

NFV, network function virtualization, EPC, vEPC, mobile core network, DPDK, PCI passthrough, SR-IOV

1. Introduction

The days when telecommunication networks carried voice calls and simple online data are long past. Today's networks must handle a burgeoning array of ever more complex applications and streaming data, all while dealing with an exponential increase in mobile traffic, as well as the rise of new services such as Internet of Things (IoT) and over-the-top (OTT) content. In the face of this, telecom operators find themselves confronting a daunting range of challenges - from how to maximize the potential and increase the flexibility of existing resources to bringing new communication services on stream as and when required to meet consumer expectations.

Simply adding more dedicated hardware or building more network capacity will not solve the problem. Managing this explosive growth in both traffic and services requires a new approach. To meet this challenge, software-defined networking (SDN) and network function virtualization (NFV) were developed to enable telecom networks to scale up capacity and capabilities quickly, efficiently, and flexibly.

In this paper we will focus on the performance requirements (high throughput and low latency) - which are particularly important for data transfer processing - for these virtualization systems, with reference to our virtualized Evolved Packet Core (vEPC) solutions, our core mobile data communication network products. In addition to reviewing the component technologies, we will also discuss some of the issues we encountered when implementing these products and how they were resolved.

2. Overview of Conventional EPC Products

2.1 Overview of EPC

The architecture of mobile core networks has been specified by the 3rd Generation Partnership Project (3GPP), a telecommunications standardization organization, and is composed of functional components - such as Mobility Management Entity (MME), Serving Gateway (S-GW), and Packet Data Network Gateway (P-GW) in Evolved Packet Core (EPC) that houses LTE lines (**Fig. 1**). These components help mobile core networks manage network control processing (control plane management) and packet transfer processing (user plane

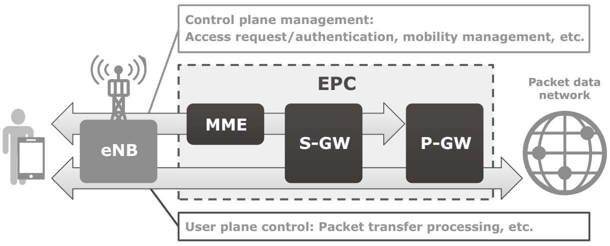

Fig. 1 EPC core components in 3GPP standard architecture.

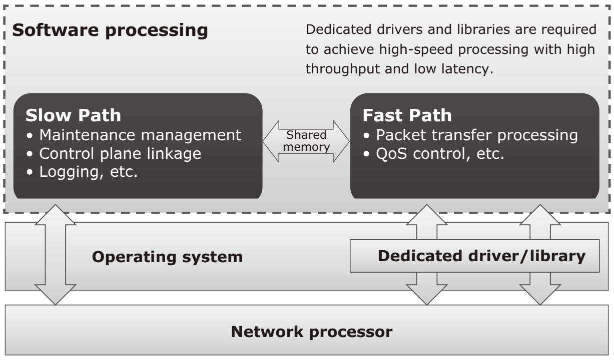

Fig. 2 System architecture of user plane control section (conventional system).

control) of user data - including access request/authentication, mobility management, and handover.

In order to avoid network-wide slowdowns caused by processing delays, the device that controls the EPC's core components needs to be able to suppress the occurrence of low latencies and latency fluctuations, while at the same time processing a large amount of network control requests as part of control plane management. For user plane control (packet transfer processing, quality of service [QoS] control, etc.), even higher performance is required in terms of throughput and latency.

2.2 Previous Method (before virtualization)

Previously, in order to meet the telecom network performance requirements described above, we used software design and memory control design intended for adoption of Carrier Grade Linux (CGL) in the implementation of control plane management. However, to achieve the high-performance throughput and latency required for user plane control, we had to use dedicated drivers and libraries compatible with dedicated hardware (network processors and dedicated chips) (**Fig. 2**).

In user plane control, high performance is critical for packet transfer processing and QoS control. To handle this, network processor functions are implemented as fast paths via dedicated drivers and libraries. For processing of maintenance and control plane linkage where the performance requirements are less stringent, general-purpose functions are implemented as slow paths.

When implementing the user plane control function section with conventional technology, software and hardware must necessarily be tightly coupled in the resulting architecture. It is this issue that must be overcome in order to successfully achieve NFV.

3. NFV Compatibility and EPC Component Technologies

3.1 Overview

Implementing NFV while maintaining network performance quality (high throughput and low latency) requires two things. First, the processing conventionally handled by dedicated hardware must be delegated to software on a commercial-off-the-shelf (COTS) server. Second, any deterioration in quality that could be caused by the overhead of the virtualized environment must be prevented.

With NEC's vEPC solutions, technologies designed for a virtualized environment such as Data Plane Development Kit (DPDK), Peripheral Component Interconnect (PCI) passthrough, and Single Root I/O Virtualization (SR-IOV) are used to solve these issues. However, replacing dedicated hardware and libraries with software on a COTS server generates new challenges. In the following sections, we will discuss some of those challenges and the measures we took to solve them. We will also discuss the technology we use to eliminate overhead in the virtualized environment.

3.2 Processing Using Software and DPDK

vEPC is able to deliver processing performance on a COTS server equivalent to what you would get with a conventional dedicated hardware and libraries thanks to the use of DPDK technology for packet transfer processing (which requires especially high performance).

The simplest way to implement user plane control processing with software on a COTS server configured with an IA CPU and standard network interface card (NIC) would probably be to use a Linux network stack and a NIC driver (**Fig. 3**, left). However, this setup would generate a number of problems with packet transmission/reception:

- Processing would be interrupted by a transmission/reception event
- Packets loss between processing of devices and EPC software
- Numerous memory copies at data handover

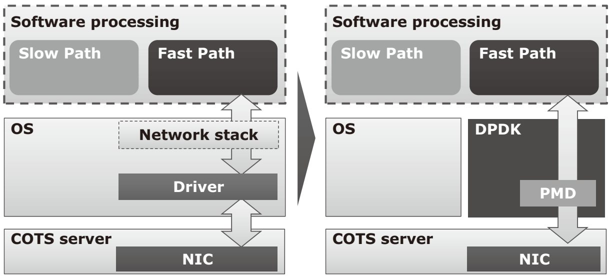

Fig. 3 Schematic diagram of software stack when DPDK is adopted (driver on left and comparison with DPDK on right).

The overhead resulting from interrupt and processing overload and the delays caused by multiple memory copies would make it impossible to meet requirements for user plane control. In fact, this implementation method results in packet loss due to buffer overflow even with extremely low packet traffic.

DPDK is a technology that enables high-speed packet processing on a COTS server - something that previously required dedicated hardware and libraries. To achieve high-speed packet transfer processing, while avoiding the factors which cause a decline in throughput and deterioration in latency compared to packet processing on a Linux network stack, the following measures are applied (Fig. 3, right).

- Use of a polling method driver called Poll Mode Driver (PMD)
- Direct exchange of data between user programs and devices so that unnecessary memory copying doesn't take place

By implementing the user plane control processing using DPDK, you can now achieve vEPC user plane control on a general-purpose IA server without using dedicated hardware and libraries.

3.3 Processing Using Virtualized Machines and Virtualization-related Technology

The usual way to build a virtualized environment is for the host OS to provide a guest OS with a virtual CPU and virtual NIC using an emulator such as QEMU. The guest OS uses the provided virtual CPU and virtual NIC to perform processing (**Fig. 4**).

In this type of configuration, the load and processing delay caused by the NIC driver and network stack first take place at the stage of virtual switch processing in the host OS layer, and then in the emulation processing performed by the virtual NIC. Additionally, the load and delay in the processing also take place due to the NIC driver and network stack on the guest OS side as well. To

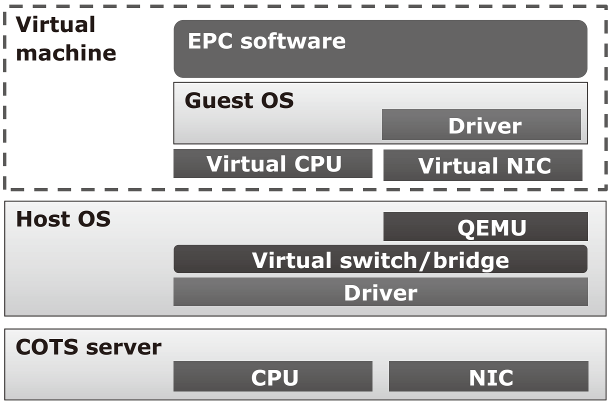

Fig. 4 System configuration with conventional virtualization.

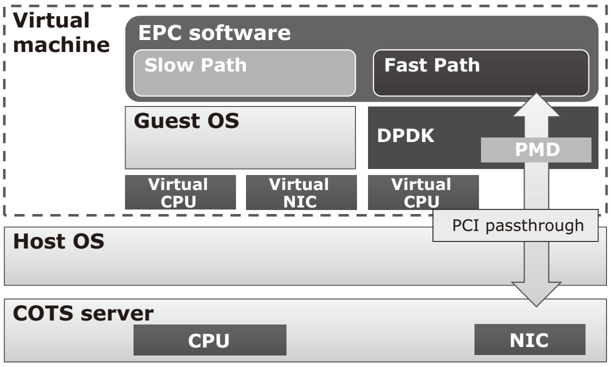

Fig. 5 System configuration using PCI passthrough and DPDK.

facilitate software-controlled user plane processing in a virtualized environment, the issues in both the host layer and guest layer need to be solved.

An effective way to solve these issues is to minimize overhead caused by virtualization by using DPDK in the guest layer and PCI passthrough in the host layer as described in 3.2 above (**Fig. 5**).

With PCI passthrough, the mapping of physical addresses for I/O in the guest layer and host layer are solved by using the CPU's virtualization support mechanism to make it possible for the virtual machines in the virtualized environment to directly refer to and control the host's PCI devices. As a result, the host layer's processing overhead in the virtualized environment can be reduced by eliminating the virtual NIC's emulation processing while skipping the virtual switch processing in the host layer.

Also thanks to the use of PCI passthrough, the physical NIC can be controlled on the virtual machine side, so the DPDK can be used for packet transmission/reception processing on the virtual machine side. In this way, NEC's vEPC is able to effectively manage user plane control using software processing, while assuring first-rate performance even on virtual machines.

However, when PCI passthrough is used, the targeted devices are occupied by the relevant virtual machines, resulting in lack of flexibility in simultaneous operation of multiple virtual machines. We've solved this problem by using a technology called SR-IOV. When using SR-IOV, we can show the PCI device as multiple virtual devices. When this is combined with PCI passthrough, a single NIC can be used from multiple virtual machines.

4. Commitment to the Future

In this paper we have shown how NEC's vEPC solution provides a low-cost, scalable, and easy-to-implement alternative to dedicated hardware and libraries. Operating as a group of virtual machines in a virtualized environment, this system is able to deliver EPC functions while meeting the performance and quality standards required for telecom networks, including user plane control. Among the technologies that make this possible are DPDK, PCI passthrough, and SR-IOV. Going forward, we intend to continue refining and developing our vEPC components, with a focus on improving management and operation flexibility on a system-wide basis, particularly in areas such as topology concealment, separation of processing and management data, and achievement of N-ACT configuration. At this time, it is important for us to look at ways of meeting the needs of telecom networks that take account of key requirements in processing and controlling billing information such as assuring compatibility between advanced services and consistent, reliable performance as an essential tool in emergency, as well as performance requirements such as high throughput and low latency.

5. Conclusion

In this paper, we focused on the performance requirements that must be met in order to successfully implement NFV on telecom network systems, as well as the component technologies used to resolve those issues. Already implemented in a number of proof-of-concepts (POCs) and commercial networks, NEC's vEPC solutions have successfully demonstrated that the system's core network processing performance - including user plane control - is more than sufficient to operate in a virtualized environment. At NEC, we are committed to providing telecom operators and end users with high-value products and services, while continuing to advance and improve the technologies on which these products are based.

* LTE is a registered trademark of European Telecommunications Standards Institute (ETSI).

* Linux is a registered trademark or trademark of Linus Torvalds.

* Intel is a trademark of Intel Corporation in the U.S. and other countries.

Reference

1) 3GPP:3GPP TS23.401, General Packet Radio Ser vice (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access

Authors' Profiles

TSUKAGOSHI Tsutomu
Senior Manager
Carrier Services Division

SAEKI Shuichi
Manager
Carrier Services Division

TERADA Junpei
Assistant Manager
Carrier Services Division

SUNAGAWA Yuta
Carrier Services Division

SHIMAMOTO Hiroshi
Software Expert
Common Platform Development Division
NEC Communication Systems, Ltd.

IGUCHI Tomohito
Assistant Manager
Common Platform Development Division
NEC Communication Systems, Ltd.

Information about the NEC Technical Journal

Thank you for reading the paper.
If you are interested in the NEC Technical Journal, you can also read other papers on our website.

Link to NEC Technical Journal website

Japanese | English

Vol.10 No.3 Special Issue on Telecom Carrier Solutions for New Value Creation

Vol.10 No.3
July 2016

Special Issue TOP