

## PRFC : A New Programmable Router Architecture Providing Flexible Customization for Service

Yan Zhang<sup>1, a\*</sup>, Jian-yang Huang<sup>2, b</sup>

<sup>1,2</sup>National Digital Switching System Engineering and Technological R&D Center  
Zhengzhou, 450002, China

<sup>a</sup>zy249876242@163.com, <sup>b</sup>huangjy@126.com

**Keywords:** routing, programmable router, virtualization, various service, NetFPGA

**Abstract.** The current routing mechanism is more and more difficult to satisfy the various service demands due to the rigid structure and simple ability. Aimed at this problem, this paper proposes a new programmable router architecture to support diverse applications and emerging communication paradigms—PRFC. PRFC supports the self-adjust of routing protocols dynamically to adapt to the different requirements of applications, supports coexistence of multiple routing paradigms, and provides a feasible solution for the network compatibility and evolvement. Finally, the prototype was implemented based on the NetFPGA platform. Compared with the existing programmable routers, PRFC realizes the coexistence and customization of various routing protocols, and achieves better performance.

### Introduction

In recent years with the rise of research towards new network architecture<sup>[1][2]</sup>, in order to support the long-term coexistence between the future diversified network architecture and the existing TCP/IP network and the progressive test and deployment of future network architecture<sup>[3]</sup>, some new type of programmable virtual routers were studied by academics. Currently Eddie Kohler<sup>[4]</sup> et al. designed and implemented the modular software router Click, but because of its software architecture, whose forwarding performance is poor. In order to improve the forward rate, Anwer<sup>[5]</sup> et al. designed a set of programmable virtual routers named SwitchBlade based on FPGA hardware. But because of the limitation of FPGA resource, this system can support up to four isomorphism hardware virtual data plane. Han<sup>[6]</sup> et al. put forward high-speed software router named PacketShader using general-purpose GPU. However, Packetshader does not support non IP forwarding requirements in the study of future network. Dobrescu<sup>[7]</sup> et al. put the high-performance software router named RouteBricks using cluster technology. However it performs poor portability because the data path uses software to build completely.

In order to improve the flexible programmable ability of routing system, this paper designed a new programmable virtual router named PRFC, which support service customization and heterogeneous network coexistence. The contributions of this study are threefold. First, this paper proposed the overall architecture of PRFC, designed the routing derived algorithm based on the control plane and the routing decision algorithm based on the data plane and realized the on-demand agreement customization and routing choice. Second, this paper designed and implemented the PRFC prototype system. Third, this paper tested and analyzed the function, forwarding performance and service quality assurance performance of the prototype system.

### PRFC DESIGN ARCHITECTURE

**Overall Architecture.** In the impulse of diversified business that needs and network dynamic behavior, PRFC can choose corresponding routing protocol according to different business and specific network type requirement, and construct various service paths with constraint attribute to meet specific application requirement. Based on the programmable hardware and virtualization technology<sup>[8]</sup> as the

core, this paper designed and implemented the PRFC prototype system. The system can support multiple protocol, flexible programmable protocol and high speed data packet forwarding. As shown in figure 1, the overall architecture is divided into control plane and data plane of two parts.

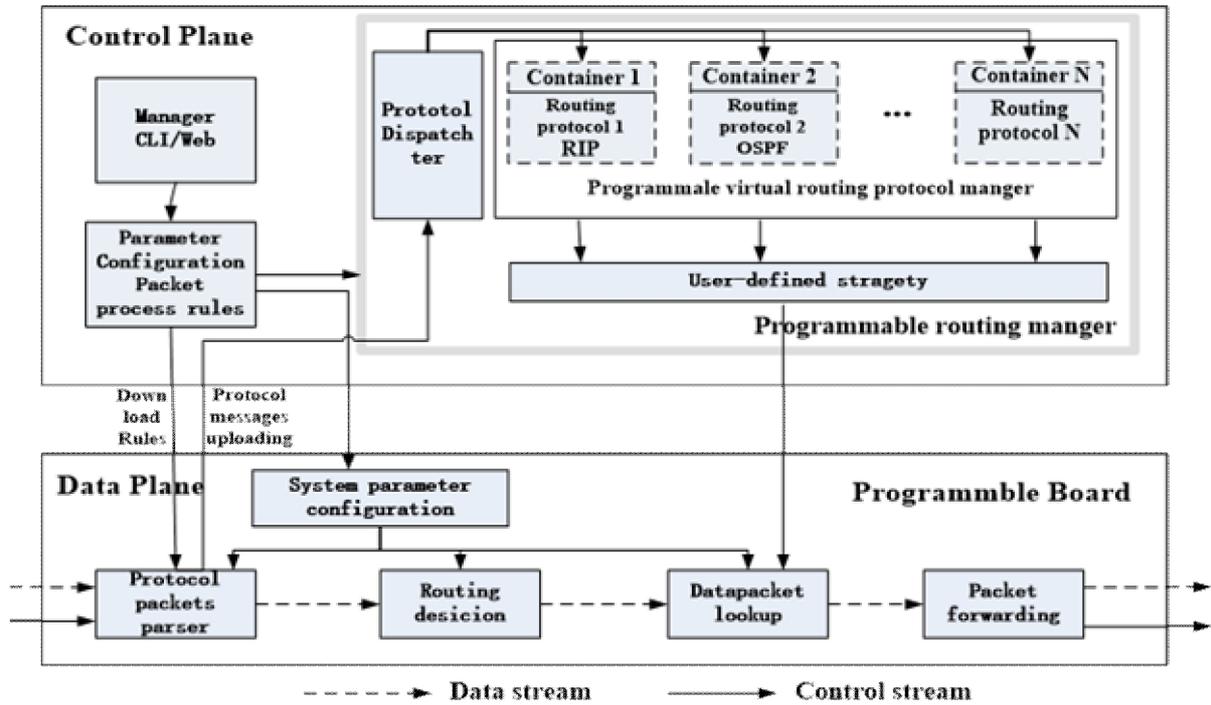


Fig.1. The Overall Architecture of PRFC

**The Structure of Control Plane.**The main function of the control plane is to complete the derivation of routing protocol and the corresponding configuration according to the requirements. In order to achieve the effective separation of multiple routing protocol instances, the way of data communication link of virtual container and physical host is shown in figure 2.

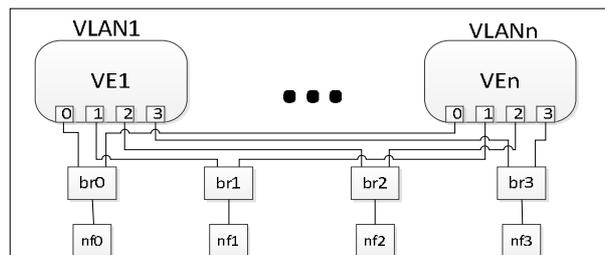


Fig.2. The Mapping of Virtual Network Interfaces

In the figure, the bridge has realized the message communication between virtual container and physical interface card, the VLAN achieves the effective separation of different routing protocol message interaction. So as to realize the on-demand derivation of routing protocol, PRFC need to flexibly control the protocol message according to the needs of the business. Therefore, we designed the routing derivation algorithm that it is as follows.

---

**Algorithm 1:** The Routing Derivation

---

```

0.   for each packet arrived
1.     set vid  $\beta$  get VLAN tag
2.   for i = 1:N
3.     if  $\exists i, \text{Container}[i] == \text{vid}$ 
4.       strip VLAN tag , sent packet to ith container
5.       rtable[i]  $\beta$  calculate routing table
6.     else
7.       start a new container N
8.       set Container[N]  $\beta$  vid
9.       Protocol[N]  $\beta$  Configure a new routing protocol
10.      end if
11.    end for
12.  end for

```

---

With the change of business requirements, PRFC can change the VLAN-ID mapping table to modify the agreement distribution rules to update the different interaction objects of routing protocol and establish new routing forwarding tables. This provides the judgement for the underlying hardware routing decision module. At the same time, it also provides a flexible programmable interface for the user to write a complex routing control strategy. Finally, the routing management platform can also develop new type of routing protocol for requirements according to the upper constantly emerging business, therefore realizing the routing mechanism coexistence of heterogeneous network.

**The Structure of Data Plane.** Data plane is mainly to process the packet at a high speed according to the ruler generated by control plane configuration. At first, the packet header classification module distinguishes different business category according to classification rules. Then, the routing decision module forwards the packet to different routing forwarding planes for processing according to the TOS-RID mapping table. The design of the hardware data plane structure is shown in figure 3.

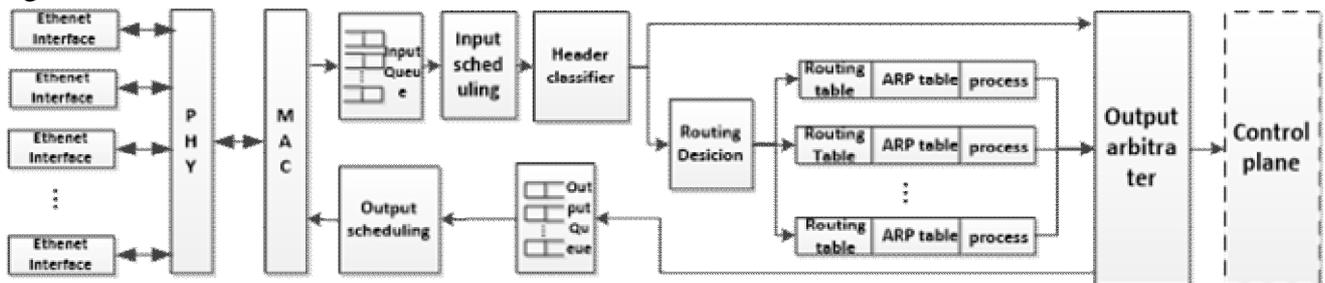


Fig.3. The Structure of Hardware Data Plane

The current designs uses the eight bits in the TOS field in the packet header to identify the types of upper businesses, and the specific identification scheme can refer to the related design in the DiffServ<sup>[9]</sup>. In this design, the high-speed packet header classification parsing module mainly distinguishes the controlled messages of routing protocols and ordinary packets. The routing decision module implements the look-up and matching efficiently of business types and hardware routing tables according to a piece of TOS-RID mapping table saved in hardware TCAM. In order to guarantee the high speed forwarding performance of the PRFC, the routing table and the ARP table all comply with the longest prefix algorithm in hardware TCAM. In order to implement the customized routing forwarding function, the system combines the classification of packet header parsing module and routing decision module to run a routing decision algorithm. The algorithm is shown as follows.

In the algorithm 3, the number of routing forwarding plane cannot completely meet the needs of the control layer routing protocol due to resource constraints of data plane. However, the table updates policy support the user-defined programming implementation to periodically choose and update

hardware routing forwarding table to meet the forwarding needs of data, therefore this promoting the system scalability. When the heterogeneous networks coexist, the routing table and ARP table in this design can be generalized through the way that corresponding look-up logic and packet processing operation can be designed according to the requirements.

**Algorithm 2: The Routing Decision**

0. for each packet arrived
1. set MACn  $\beta$  get ingress port
2. if it is VLAN packet
3. sent packet to CPU<sub>n</sub>
4. else
5. set tos  $\beta$  get TOS tag
6. rid  $\beta$  Lookup TOS--RID table
7. If rtable[rid] exist in hardware
8. nexthop  $\beta$  Lookup routing table rtable[rid]
9. MAC address  $\beta$  Lookup arp table atable[rid]
10. sent packet to egress port MAC<sub>x</sub>
11. else
12. through the user-defined strategy
13. write rtable[rid] in hardware based on user-defined strategy
14. update TOS—RID table
15. goto (8)---(10)
16. end if
17. end if
18. end for

**The Prototype Implementation.** The detailed design of PRFC routing prototype is shown in figure 6. The control plane of prototype system adapts the architecture of OpenVZ<sup>[10]</sup> and Quagga<sup>[11]</sup>. In the prototype, the data plane can parallelly look up routing and ARP tables based on the NetFPGA-10G<sup>[12]</sup> board. All of the number of tables is 32, which can be expended according to need.

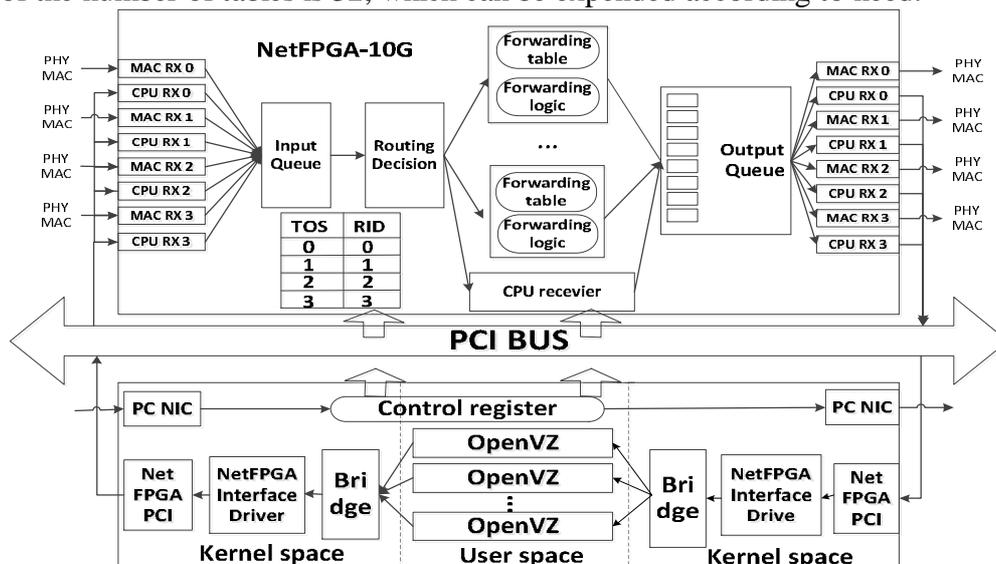


Fig.4. the PRFC Prototype

**EXPERIMENTAL RESULTS**

The test scene of PRFC prototype is composed of five nodes. Each node builds three virtual containers through the OpenVZ virtualization technology and runs the RIP, OSPF and ISIS routing

protocol in the same time. In the network, there are three types of business deployed, namely VOIP, FTP and Video, whose TOS field is set to 0x00, 0x01, 0x02 respectively.

**Forwarding Rate.** The forwarding rate is an important performance index to measure the router system. This paper tests the actual forwarding rate of PRFC prototype system under the circumstance of different packet size by using the Iperf3.

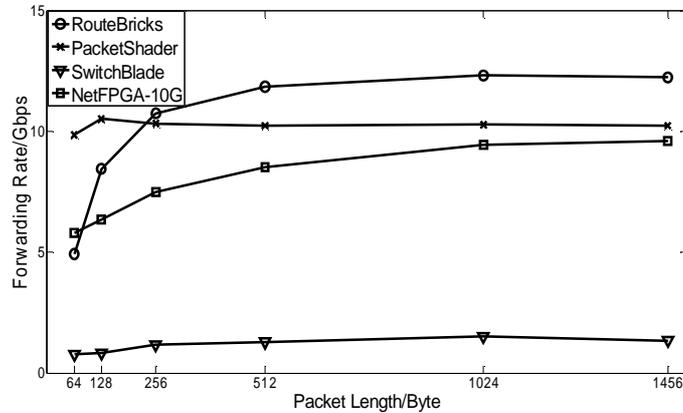


Fig.5. The Forwarding Rate based on Different Packet Length

It can be seen that the maximum single port forwarding rate of the PRFC prototype is 9.6 Gbps, which is close to the theoretical maximum 10 Gbps. The maximum throughput of the system is close to  $9.6 \times 4 = 38.4$  Gbps. Compared to the SwitchBlade based on the NetFPGA-1G interface card, PRFC improves the system forwarding performance dramatically. Meanwhile, PRFC's forwarding rate close to the highest level, compared with the PacketShader using the technology of GPU acceleration and the RouteBricks using cluster technology.

**QoS Assurance.** In order to verify the performance of PRFC routing system in improving the QoS of business, the comparison of packet loss rate and actual link transmission bandwidth is shown in figure 6 and 7 respectively.

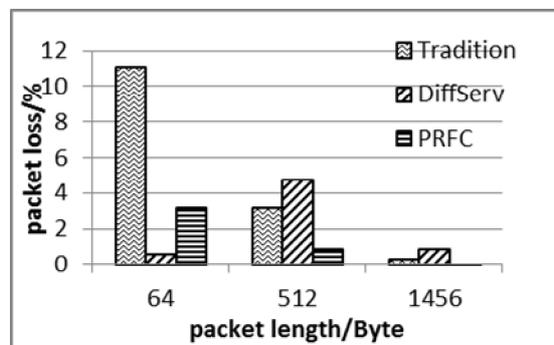


Fig.6. the packet loss based on different business

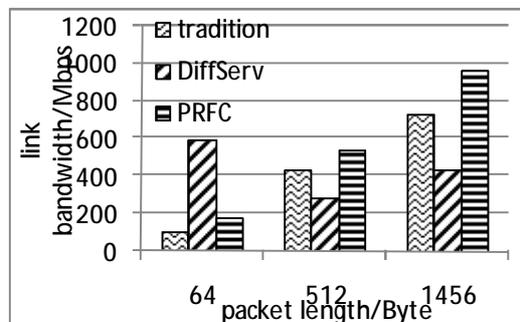


Fig.7. the link bandwidth based on different business

The experimental results show that the DiffServ only guarantees the transmission bandwidth of high priority business ( 64Byte ). The transmission bandwidth of the rest of the two kinds of business is less than traditional routing scheme, whose packet loss rate is higher. Therefore, DiffServ can only guarantee the QoS requirement of single business. Compared with the traditional routing way, the

packet loss rate of business flow transmission was reduced by 73%, the link transmission bandwidth increased 28% on average. Therefore, PRFC well promoted the quality of service when many business coexist compared with Diffserv through distinguishing the business flow and using customized transmission routing scheme.

## CONCLUSION

Aimed at the rigid architecture and single function of the current routing mechanism, we proposed PRFC routing module based on business needs and the concept of routing protocol self-adaptive. Furthermore, we design a routing prototype system named PRFC which supports diverse business needs and support heterogeneous networks coexist, using the open programmable hardware platform of NetFPGA-10 and the operating system virtualization technology OpenVZ and programmable routing control platform Quagga as the core. Compared with the existing system, the system not only supports the business custom routing service path, but also promotes in the forwarding performance and QoS at the same time. In general, this paper is of great significance for meeting the diverse routing descriptions of future business and supporting the progressive test and deployment of the future new network.

## ACKNOWLEDGMENT

This material is based upon work supported by the “973 Program” of China under Grant Nos. 2012CB315901 and 2013CB329104, in part by the Natural Science Foundation of China (NSFC) under Grant Nos. 61309019 and 61372121, in part by the National High Technology Research and Development Program (863 Program) of China under Grant No. 2013AA013505 and 2015AA016102.

## REFERENCE

- [1] Bell. Failure to thrive: QoS and the culture of operational network in Proceedings of the ACM SIGCOM M Workshop on Revisiting IP QoS (RIPQoS03). New York, USA, 2003: 115 -120.
- [2] McKeown N. Software-defined networking [J]. INFOCOM keynote talk, 2009, 17(2): 30-32.
- [3] Van Jacobson, Diana K Smetters, James D Thornton, et al . Networking Named Content[A] . Communications of the ACM, 2012, 55(1): 117-124.
- [4] E Kohler, R Morris, B Chen, et al. The click modular router [J]. ACM Transactions on Computer Systems, 2000, 18(3): 263-297.
- [5] Anwer M B, Motiwala M, Tariq M B, et al. SwitchBlade: A platform for rapid deployment of network protocols on programmable hardware [C] Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications(SIGCOMM’10), New Delhi, New York, 2010: 183-194.
- [6] HAN Sangjin, JANG Keon, PARK Kyoung Soo, et al. PacketShader: a GPU-accelerated software router. ACM SIGCOMM Computer Communication Review, 2010, 40(4): 195-206.
- [7] DOBRESCU M, EGI N, KATERINA J, et al. RouteBricks: Exploiting parallelism to scale software routers [C] Proceedings of the 22nd ACM Symposium on Operating Systems Principles (SOSP’09), Big Sky, MT, USA. New York, NY, USA, 2009: 15-28.
- [8] N. M. Chowdhury, R. Boutaba. A survey of network virtualization. Computer Networks, 2010, 54(5): 862–876.
- [9] Blake S, Black D, Carlson M, et al. An architecture for differentiated services. RFC 2745, December, 1998.
- [10] OpenVZ: Server Virtualization Open Source Project. [EB/OL].<http://www.openvz.org>.
- [11] Kunih iro Ish iguro. Quagga-A routing software packagefor TCP/IP networks. [EB/OL]. <http://www.nongnu.org/quagga/>.

[12] NetFPGA-10G project [EB/OL], 2014.