EIE: An Evolvable Internet Environment

Pingping Lin
Network Research Center, Tsinghua University
Department of Computer Science, Tsinghua University
Tsinghua National Laboratory for Information Science and Technology
Beijing, China
linpp@netarchlab.tsinghua.edu.cn

Jun Bi
Network Research Center, Tsinghua University
Department of Computer Science, Tsinghua University
Tsinghua National Laboratory for Information Science and Technology
Beijing, China
junbi@tsinghua.edu.cn

Hongyu Hu
Network Research Center, Tsinghua University
Department of Computer Science, Tsinghua University
Tsinghua National Laboratory for Information Science and Technology
Beijing, China
huhongyu@cernet.edu.cn

***Abstract*—The end-to-end characteristic of Internet enables easy deployment and modification to the application layer protocols running on host. And the competition among these applications promotes the development of the Internet. However, new protocols related to core network layer or network forwarding equipment are hard to deploy, which hinders the evolution of the core network technology. To solve this problem, this paper proposes an Evolvable Internet Environment (EIE), which runs as a programmable platform and provides a mechanism for normal network researchers (not only the equipment vendors like Cisco and Juniper) to implement and deploy new network architectures or try their new ideas on the forwarding device and operate the hardware resources on it. EIE supports incremental deployment and provides a collaborative experimental environment for academia, industry achievements. With EIE, various Internet architectures or protocols can plug themselves into EIE network forwarding equipment and run simultaneously for experiment and actual development. Architectures upon EIE coexist and compete with each other, and some of them maybe succeed or are eliminated under natural selection. EIE uses such competition to promote the evolution of the core network.**

Keywords-Internet Architecture; Evolution; Competition; Platform.

I. INTRODUCTION

With 30 years of development, Internet itself got an unprecedented development and prosperity. However, new protocols involving core network layer or core network forwarding equipment (mainly refers to switches or routers) such as Differentiated services, IP Multicast, RBA [1], RNA [2], SILO [3], secure routing are suffering from the large-scale deployment embarrassment:

1) New protocols need to be standardized.
2) Need large-scale test.
3) Need realization into forwarding equipment.
4) Deployments are always costly, and lack of enough deployment incentives.
5) Furthermore, there is no large real user traffic, economic factors, routing strategy or deployment incentives in testbed. Experiments running successfully in testbed cannot prove that they will run successfully in real network. This further hinders the realization maturity of equipment and the deployment decision of operators.

The evolution of Internet core technology is at a standstill.

Besides, only one Internet architecture cannot meet all requirements of the future Internet; if we try to meet all requirements by improving TCP/IP, it would be a problem of seeking for an optimal solution in multiple constrained conditions. If these constraints have conflicts, there would be no optimal solution. It is necessary to solve various problems by a coexistence of varieties Internet architectures.

Furthermore, in the past years, it was impossible for commercial switches and routers to provide an open software platform due to competition. And the internal detail of network equipment is hidden for new experiments; this might lead to the network crash down. However, in recent years, with the development of the technology and thousands of demands from the vendor's customers, the equipment vendors start to open up more and more programming interfaces for customers to develop their needs by themselves.

Based on the analysis above, this paper proposes an evolvable platform EIE to support the coexistence of various Internet architectures. It allows normal researchers to participate to the development of the core network. Internet architectures can be easily plugged into EIE platform, and the TCP/IP stack is one of them. The EIE upper architectures can form a competitive relationship to solve a same problem, similar to the relationship among Skype, MSN, Google Talk, or complement each other to solve different problems. Users can become clients of one or several network architectures.

The following sections are organized as follows: Section II describes the related work. Section III gives a full explanation of the EIE mechanism. Then, the deployment issues are discussed in Section IV. Finally, this paper presents evaluation in Section V and conclusion in Section VI.

II. RELATED WORK

During the past few years, several experiment platforms have been proposed for the network Innovation. OpenFlow [4] is a solution providing real data flows for researchers to carry out their Internet innovations. Currently, it works in the local area network, and mainly deals with IPv4 protocol. It faces lots of challenges to achieve the worldwide deployment, such as the scalability problem, management problem and so on.

PlanetLab [5] is an open, shared testbed for developing, deploying, and accessing planetary scale applications. Researchers can use a slice made up of several dedicated hosts or servers to carry out their experiments. In contrast, EIE is to achieve a multi-architecture coexisting Internet, and focuses on programming in network forwarding devices rather than hosts.

The Global Environment for Network Innovations (GENI) [6] is organized around several focus areas, facility architecture, the backbone network, and distributed services. GENI is a clean-slate solution and possesses its own experiment devices,

 ISBN: 978-1-61208-148-9

this is the main difference between GENI and EIE, as EIE is dirty-slate and running on production network.

Active network [7] allows customized computations on packets in programmable routers to accelerate infrastructure innovation. However, its design is based on TCP/IP by adding an Active Layer upon TCP/UDP layer. Besides, it does not support the deployment of revolutionary architectures like RBA and RNA, which are equal to TCP/IP stack and need the hardware change on routers. Thus, it is hard to deploy them.

Virtual router [8] achieved the router underlying resource isolation. But the function in virtual router is a subset of the master router. Currently, it is used in several special occasions and does not support the proposed novel architectures or protocols by researchers like a new OSPF or BGP.

So far, the easy and efficient deployment issue of new architectures is still unresolved. Novel ideas related to core network layer or core network device are still need years to be deployed to the production network.

III. THE EIE ARCHITECTURE AND MECHANISM

A. EIE targets

EIE mainly has two goals: 1) Researchers can program and easily try their new architectures or protocols in production network device. 2) EIE is able to accommodate varieties of other architectures, as a platform to enable the competition among the upper architectures. Through natural selection, the fittest survive. Besides, different architectures share the same underlying hardware resources.

B. EIE model

symbol	meaning
NE	network entity
NC	network connector
CS	constrains, the relationship between NE and NC
h	host
le	logical forwarding equipment
sas	safety assurance system

EIE = Host (*NEh*)
+ Logical forwarding device (*NEle*)
+ Logical link (*NC*)
+ Safety assurance system (*NEsas*)
+ EIE mechanism (*CS*)

Definition: *EIE = (NE, NC, CS)*

EIE = (network entity, network connector, constrains)

$$NE = \sum_{k\in(h,le,sas)} NE_h + NE_{le} + NE_{sas}$$

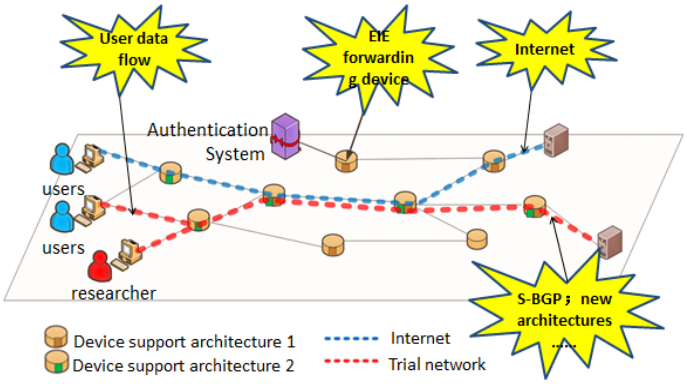

Figure 1. Future Internet with EIE

In Figure 1, the trial architecture network in blue line and current Internet in red line coexist in the future. And the trial architecture network runs in production environment. Internet users can contribute traffic to new architecture network. Each architecture network may only cover a part of the entire Internet; if one architecture has not been used for a long time by users, finally, it will completely be out of stage by natural selection.

The authentication system or safety assurance system controls which researchers can deploy resource code to the EIE forwarding equipment.

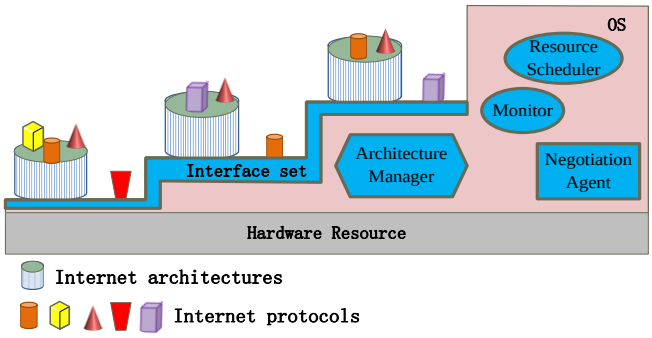

Figure 2. Components in EIE forwarding device

In Figure 2, EIE forwarding device contains five components:

① Interface Set: open up to researchers for programming. Ladder type interfaces means supporting various granularity functions. New protocols proposed by researchers can run directly on the EIE interfaces or on an architecture running in EIE.
② Negotiation Agent: as the representative to negotiate new protocol deployment requests from Internet.
③ Architecture Manager: responsible for architecture access control, management, withdraw, and maintaining records {resource size, lease length, developer ID, process ID …}
④ Monitor: monitoring the health status of architectures and feedback to Architecture Manager.
⑤ Resource Scheduler: responsible for resource allocation and isolation.

Outside EIE forwarding equipment, there are two other components in the whole EIE platform:

⑥ Safety assurance system: protecting the safety of programmable forwarding device.
⑦ Virtual build environment: for developers.

Figure 3 below describes how the seven components work in EIE mechanism.

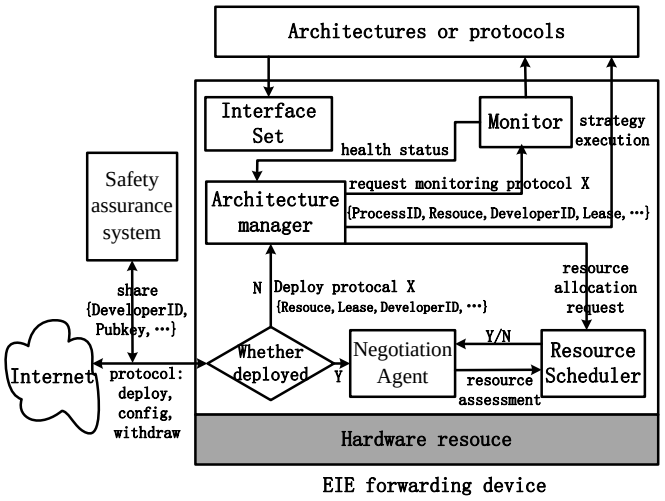

Figure 3. The whole EIE components interaction mechanism

C. EIE compositions

EIE includes two parts: 1) Normal hosts with EIE software to support multi-architecture; 2) EIE supported network devices (switches or routers). Besides, there should be a safety assurance system to control which researchers can deploy programs to EIE device in Figure 1.

D. EIE host

1) Basic model

As the figure below, the existing hosts need an additional EIE module in the link layer to support multiple architectures. Users just need to download the EIE software and install it.

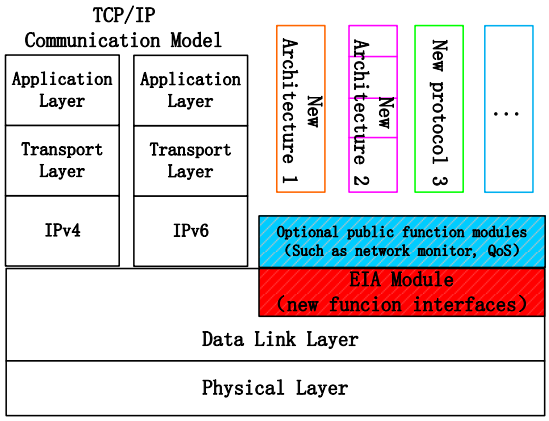

Figure 4. Basic model of EIE host to support multiple architectures

The existing TCP/IP communication protocol stacks (IPv4, IPv6) run as two special architectures and coexist with other new architectures.

After installing the EIE module, the tasks of the link layer are as follows: it receives a packet from the Internet, then passes the packet to the appropriate upper protocol stack according to the architecture or protocol ID; when receives a packet from the upper architectures, it will forward this packet to a specific port or make a call to the underlying hardware resources according to the instruction of upper architectures.

2) Implementation in host

EIE on host is achieved by WinPcap [9], which allows application to capture and transmit packets bypassing the TCP/IP stack. In this way, EIE can get an entire data frame, and directly put it onto the network adapter and then to the Internet.

E. EIE network equipment

1) How to support multiple architectures for EIE network equipment

EIE supporting multiple architectures to coexist requires those architectures to share the underlying hardware resources.

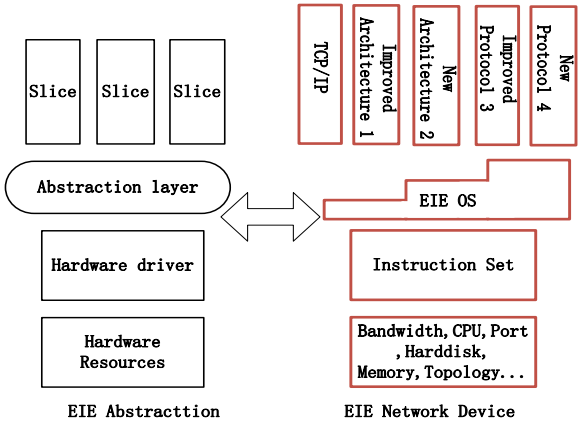

Figure 5. EIE network equipment

As shown above, the instruction set in EIE network device will be provided as primitive operation to operate the hardware resources, similar to the x 86 instructions in computer. EIE operating system layer encapsulates instructions into interfaces and provides them to the upper architectures or protocols, such as packet forwarding, discarding, and rewriting functions.

2) Resource management

Each architecture can enjoy a certain number of hardware resources like hard disk space, memory, bandwidth, and so on. Once the amount of usage exceeds the pre-negotiated size, EIE will carry out strategies like suspending this process for a certain time. In this way, EIE guarantees that new architectures have negligible effect to the existed data traffic.

3) How to support EIE for commercial network equipment

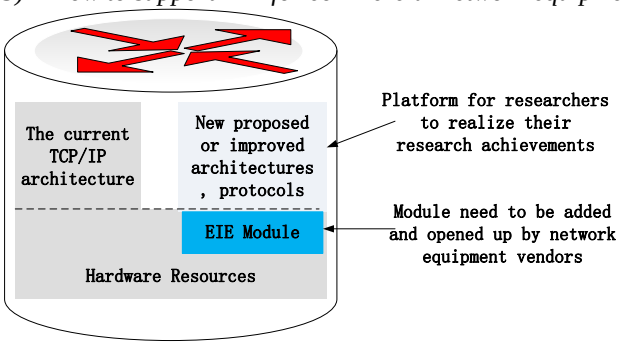

Figure 6. Commercial forwarding equipment to support EIE

The EIE forwarding equipment does not change the current TCP/IP architecture. Network equipment vendors just need to add EIE module and open up some interfaces to developers as the figure above. Once researchers gain permission to deploy a new architecture to EIE equipment, they also can send commands dynamically to change their code, configurations, parameters, and others.

 ISBN: 978-1-61208-148-9

II. DEPLOYMENT

A. The compatibility of EIE with the current Internet

EIE achieves compatibility with the existing network, by taking the current TCP/IP protocol stack architecture as a special case among the multiple competing architectures. EIE does not add any new protocol layer to the current TCP/IP architecture, and just adds a software mechanism to the traditional network equipment. So the EIE solution can achieve faster implementation and easier transition.

B. EIE supports incremental deployment and arbitrary virtual topologies

In the initial stage, EIE uses the Overlay technology [10] for a packet of new architecture to transport from one EIE network device to another to achieve the incremental deployment.

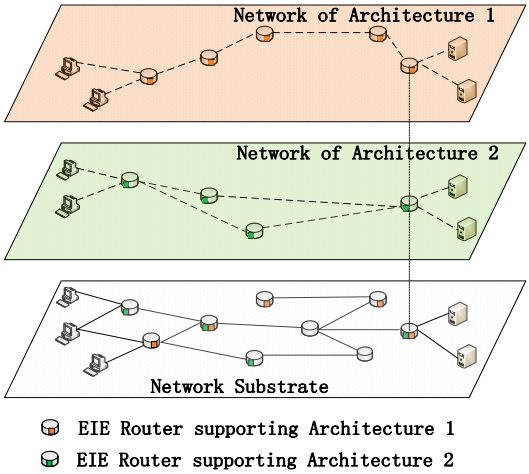

Figure 7. Current network infrastructure with EIE routers

As in the figure above, different architecture networks have different views of topology, such as the topology of network 1 and the topology of network 2.

III. EVALUATION

A. The advantages of EIE

1) The benefit to researchers

Researchers can do experiments in the real world; experiments will carry real traffic of end users at a large scale. Compare with the traditional way to deploy a new protocol to the real network in section I, in EIE the cycle time for a new protocol from the conception design to the commercial use is greatly shortened.

2) Delployment Incentives for Internet Service Provider

There are mainly two ways to motivate the EIE deployment for an ISP (Internet Service Provider). : 1) Through government support. Currently academic achievements are always difficult to be deployed to the real network. EIE can increase the conversion rate of the scientific research achievements. 2) ISP can charge equipment rental fees from researchers. Researchers maintaining their protocols reduce the management cost of ISP.

B. Performance of EIE equipment

Each architecture can enjoy a certain number of hardware resources, once the usage amount exceeds the pre-assigned number, EIE will carry out certain strategies. New proposed architectures or protocols also can run as pluggable hardware modules. In this way, EIE guarantees that new architectures or protocols do not affect the existed traffic, and that the new architectures on EIE can get a high forwarding speed.

IV. CONCLUSION AND FUTURE WORKS

At the current stage, new architectures or protocols involving network core layer or core forwarding equipment (switches, routers and others) are hard to deploy. This impeded the evolution of the Internet core network. Although researchers have already proposed various Internet architectures, almost no architecture has been large-scale deployed and widely used on the Internet so far. The main contribution of this paper is the proposed EIE theory and its model, which runs as an open and programmable platform achieving easy deployment of innovative architectures or ideas from researchers.

Researchers participate in the core network development to realize their ideal future Internet, rather than relying on vendors to implement. Architectures upon EIE can coexist and compete to achieve the continual evolution of the core network.

With EIE, we believe that the Internet will become more and more powerful, and meet the growing demands of users. The future work of EIE will focus on defining the detail of the interface set for network equipment vendors, improving the EIE theory, and implementing the EIE prototype.

ACKNOWLEDGMENTS

Supported by National Science Foundation of China under Grant 61073172, Program for New Century Excellent Talents in University, Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant 200800030034, and National Basic Research Program ("973" Program) of China under Grant 2009CB320501.

REFERENCES

[1] R. Braden, T. Faber, and M. Handley. “From protocol stack to protocol heap role-based architecture.” Comp. Comm. Rev., Jan. 2003, pp. 17-22.

[2] J.D. Touch, Y.S. Wang, and V. Pingali. “A Recursive Network Architecture”, ISI Technical Report ISI-TR-2006-626, December 2006.

[3] R. Dutta, G. N. Rouskas, I. Baldine, A. Bragg, and D. Stevenson, “The SILO Architecture for Services Integration, Control, and Optimization for the Future Internet”, in Proc. IEEE International Conference on Communications ICC ’07, G. N. Rouskas, Ed., Glasgow, Scotland, 2007, pp. 1899–1904.

[4] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. “OpenFlow: Enabling Innovation in Campus Networks” , Openflow wihtepaper, March 14, 2008.

[5] PlanetLab: http://www.planet-lab.eu/ 05.29.2011

[6] GENI: Global Environment for Network Innovations. http://www.geni.net. 05.29.2011

[7] Active/ProgrammableNetworking:http://www.run.montefiore.ulg.ac.be/Research/Topics/index.php?topic=Active 05.29.2011

[8] Virtual router: http://www.cisco.com/web/CN/products/products_netsol/switches/products/ca7000/nexus_7000_virtualization.html?pnBAC=OTC-CC5OB001RF01 05.29.2011

[9] The Windows Packet Capture Library, http://www.winpcap.org/ 05.29.2011

[10] E.K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. “A Survey and Comparison of Peer-to-Peer Overlay Network Schemes”. IEEE Communications Surveys and Tutorials, Second Quarter, 2005, pp. 72–93.

 ISBN: 978-1-61208-148-9