

Evolvable Internet Architecture (EIA)

Bi Jun, Lin Pingping, Hu Hongyu

(Network Research Center of Tsinghua University, Beijing 100084, P. R. China)

Abstract:

The end-to-end attribute of the Internet enables easy modification and deployment of applications running at the host. Competition among these applications promotes the development of the Internet. However, new protocols related to the core layer, and network routers and switches are often hard to successfully implement. This paper proposes an Evolvable Internet Architecture (EIA). It suggests that new network architectures can be plugged into network equipment or into a host through interfaces provided by EIA for network experimentation or actual network deployment. Users can independently select network architectures, and use one or some of these architectures at the same time. The diversity provided by EIA will promote the evolution of the Internet.

After more than 30 years development, the Internet has achieved unprecedented prosperity and has become an essential infrastructure in our daily work and lives. However, the Transfer Control Protocol (TCP)/Internet Protocol (IP)-based architecture of the Internet was born with defects. The best-effort forwarding strategy, for example, cannot provide users with required Quality of Service (QoS); network management is hard to implement; network security vulnerabilities are increasingly exposed; IP address resources are almost exhausted; access devices are ubiquitously mobile and heterogeneous; and routing in a large-scale network suffers poor scalability.

Efforts to reform TCP/IP-based architecture have been unceasing, with new protocols and algorithms being introduced one after another. The end-to-end attribute of the Internet enables application layer protocols at

the host end to be easily modified and deployed, and through competition between all applications (both old and new), development of the Internet is driven forward. However, this attribute limits innovation in the core part of the Internet. In order to deploy new protocols designed for the Internet core or core network devices, global deployment or modifications on all network devices is necessary. In reality, however, network operators and device vendors are often disinclined to make such large-scale deployments or modifications due to lack of incentive mechanisms. Also, because new protocols have often not been thoroughly tested in a real network environment, they fail to win the trust and support from network operators. This hinders the protocols from maturing in their implementation, and affects operators' decision-making on deployment. As a result, new protocols have not been applied on a large scale and the evolution of Internet core technology has come to a standstill.

To overcome the defects of TCP/IP-based architecture, network researchers have put forward new

architectures including Role-Based Architecture (RBA)^[1], Recursive Network Architecture (RNA)^[2], and architecture for Services Integration, Control, and Optimization (SILO)^[3]. Each of these architectures provides a solution to certain problems with existing architectures, but due to the following reasons none of them come close to meeting all requirements of the future Internet.

(1) Multiple architectures are needed at the same time in order to solve the different aspects of network problems, and to meet the demands of future applications. Network problems can be solved by using multiple architectures together rather than by improving existing TCP/IP-based architecture. Employing one architecture to satisfy all demands is something akin to seeking the optimal solution within multiple constraints. Where there are contradictory constraints, there cannot be an optimal solution. Hence, a single, standalone architecture is insufficient to meet the demands of future applications. Multiple architectures can deliver services in a complementary or competitive way so as to drive the

This work was funded by the Research Fund for the Doctoral Program of Higher Education of the Ministry of Education of China under Grant No. 200800030034.

development of the Internet.

(2) Network architecture is always enhanced and optimized. It is difficult to predict demands of the future Internet because such demands change with time. The need for new functionality may give rise to new protocol stacks (or network architectures), and a new mechanism is required to allow these protocol stacks to coexist with the old architecture. Conversely, an old protocol stack may be completely abandoned if it has not been used for a long time.

(3) Experimental network traffic should be tested in a real network and coexist there with user traffic. Issues such as competition, strategy, and incentive do not arise in any test bed with no user traffic. Therefore, a prototype system that has been proven successful in a test bed is not sure to work well in the real Internet. Researchers should conduct experiments in real networks rather than in test beds. In this way, when a new protocol is accepted by users, it can be directly put into practical use, avoiding the transplant from experimental network to real network.

This paper proposes an Evolvable Internet Architecture (EIA) that supports the coexistence of multiple architectures. EIA is a structure that is able to contain several network architectures and that allows new architectures to join. It functions as a “dock” or “socket” for these architectures, enabling them to plug in. Existing TCP/IP stack is one of these plugged in architectures. Architectures using the EIA can supplement each other to solve different problems, or can contend to solve a same problem—like the relationship between Skype, QQ, MSN and Gtalk in real-time communications. Users can randomly select different architectures, and use one or some network architectures simultaneously.

1 Evolvable Internet Architecture

1.1 Evolution Strategy of EIA

There are two main approaches to the

evolution of the Internet: involution and evolution.

The projects that adopt the involution approach include Global Environment for Network Innovations (GENI), Future Internet Design (FIND) and Future Internet Research and Experimentation (FIRE).

The evolution approaches attempt to mend the existing Internet.

Taking the involution approach involves reconstructing an experimental network, and this inevitably leads to high costs. And forecasting to the future can also be erroneous. Only continuous reform to the current Internet is possible. Therefore, EIA takes the evolution approach, which enables researchers to implement and deploy new protocols within existing networks.

The evolution approach used for EIA differs from existing ones.

Existing approaches mend the current TCP/IP-based architecture, find new problems, solve the problems, and then mend it again. However, a certain mend may not be compatible with future mends. Moreover, these approaches, including Network Address Translation (NAT), focus on short-term return but may cause long-term damage.

EIA is different. It supports the coexistence of multiple architectures, and TCP/IP-based architecture is just one of them. Through competition, the best solutions survive and develop.

1.2 Objectives of EIA

The main objectives of EIA are as follows:

Researchers can develop and test new protocols on real network devices. Experimental data flow and normal flow run parallel in the real network and do not affect each other.

EIA is able to accommodate a variety of architectures. EIA works as a platform enabling competition among these architectures; they can contend for solving the same problem or supplement each other to solve different problems. They can also share their hardware resources with each other.

Various network architectures have been proposed, each of which has its

own special design. However, EIA enables the most suitable architectures to survive and develop by means of competition. As an inclusive network platform to the upper architectures, EIA must take the following issues into account:

Each architecture may cover a part of the Internet only.

Terminal users may use one or several architectures, but not all the architectures, in a similar way that some Internet users use QQ and Gtalk, but not MSN, in real-time communications.

1.3 Makeup of EIA

The EIA is made up of two parts:

- Normal hosts with EIA software to support multi-architecture;
- EIA supported network devices (switches or routers).

1.3.1 EIA Host

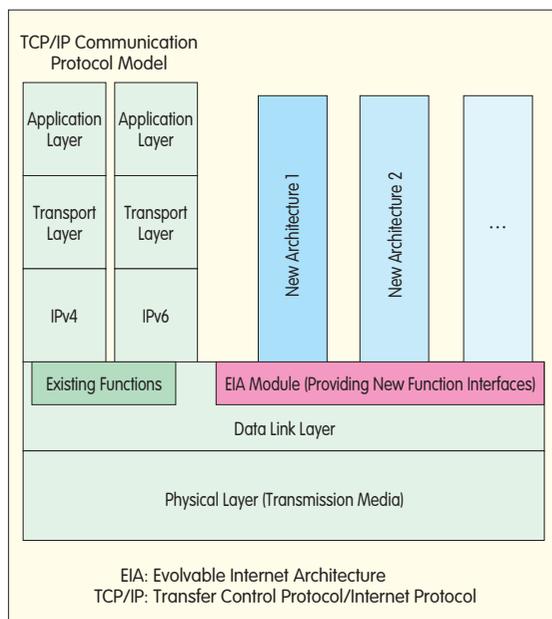
(1) Basic Mode

Figure 1 shows the multiple protocol stacks of an EIA host. From this figure, it can be seen that the EIA module is at the data link layer, and provides some function interfaces to support multiple architectures. In existing architecture, the host has to be configured with the EIA module in order to enable the coexistence of several architectures. Some architectures may use existing function interfaces rather than those provided by the EIA module to meet service requirements.

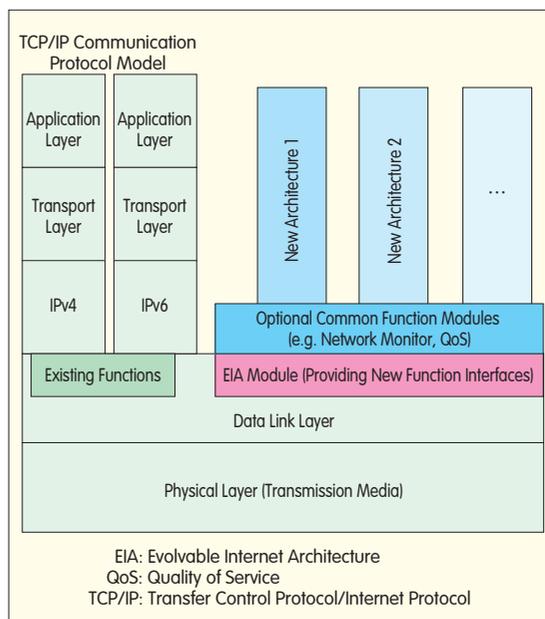
The EIA module transfers data packets to different architectures based on their architecture ID (defined as architecture protocol numbers). The existing TCP/IP communication protocol model acts as a special architecture and coexists with other architectures.

After the EIA module is added, the data link layer works as follows:

- Upon receiving a packet from the lower layer, the data link layer deframes the packet and uploads it (according to the protocol ID) to a related protocol stack (i.e. architecture) above.
- When it receives a packet from the upper architectures, the data link layer forwards the packet to a port or calls related hardware resources to process the packet, depending on the task



▲ Figure 1. Multiple protocol stacks of the EIA host.



▲ Figure 2. EIA with common functional modules added.

virtualization of Personal Computer (PC), acts as the primitive operation to directly operate hardware resources. The packet action interface layer—also called the network virtualization layer—encapsulates the instructions of the lower layers as a packet-level interface, and provides it to the upper layer architectures for forwarding, discarding, rewriting and other functions.

The packet action interface has the

assigned by the upper protocols.

Compared with the existing Open System Interconnection (OSI) model, the EIA model moves the socket interface down to the physical layer. The architectures above the data link layer are arranged vertically and in parallel.

(2) Advanced Mode

In the basic model, the data link layer is mostly responsible for deframing packets and encapsulating frames. To enable the easy development of new architecture, advanced EIA abstracts the common functional modules of new architectures, such as network monitor and QoS, and provides them to the developer as optional. Figure 2 illustrates EIA with common modules added.

(3) Storage Location of Architecture ID

Architecture ID can be stored in one of the following ways:

- In the frame head of data link layer—as the packet format differs from one architecture to another.
- In a field after the frame head. All architectures can negotiate on a common data field or location used to save the architecture codes. This is similar to the location of the version number in IPv4/IPv6 dual-stack structure. When the data link layer

receives a data packet, it splits the packet and checks the packet head. If the first field of the packet head includes 4 (meaning the version number of the IP packet is 4), this packet will be processed by IPv4 stack; if the first field is 6, the packet will be processed by IPv6 stack. This working mode is similar to that of an IPv4/IPv6 dual-stack node.

1.3.2 Network Device Supporting EIA

(1) Principle

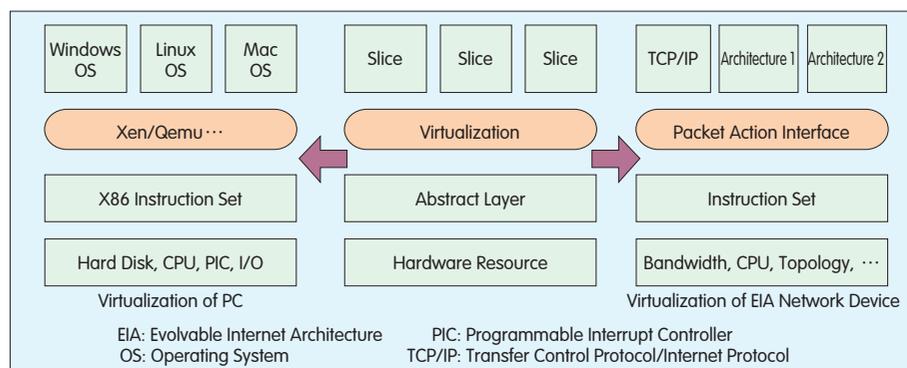
The coexistence of several architectures, supported by EIA, requires the underlying hardware resources to be shared.

As shown in Figure 3, the instruction set in virtualization of EIA network device, similar to x86 instructions in

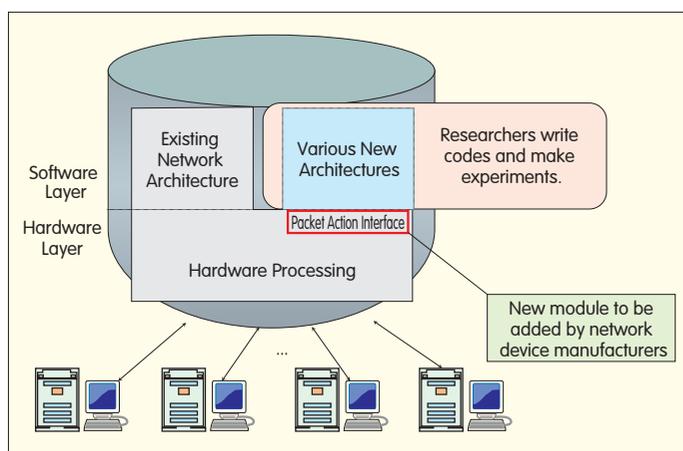
following two tasks:

- To receive data packets from a port and upload them to related protocol stacks according to protocol numbers.
- To accept tasks from upper layer protocols and then call the instruction set to perform corresponding operations on the data packets (e.g. to forward data packets to related network device interfaces). The operations on data packets are defined and processed by each protocol, while the virtualization layer only forwards data packets and calls functions to execute lower-layer primitive functions.

Above the packet action interface, each architecture runs independently and has no impact on the other. Therefore, each protocol sees a



▲ Figure 3. PC virtualization vs. virtualization of EIA network device.



◀ Figure 4.
Enabling network
devices to support EIA.

different network view or network topology from others. In addition, the communication between architectures can be further defined.

(2) Enabling Commercial Network Devices to Support EIA

On condition that developers promise not to perform any operation affecting normal data flows (for instance restarting or shutting down a network device), network device manufacturers can open some interfaces to developers without exposing details of their devices. In Figure 4, these interfaces refer to packet action interfaces, which are the modules to be added by the manufacturers. The relationship between a network operator and the developers is similar to the relationship between the super administrator and common users in Linux Operating System. Once such interfaces are provided, users can write codes and test them directly on network devices.

When the primitives of a network device are not enough for a new protocol—for example, some special encryption algorithms are needed for hardware implementation—the developer can perform the new function by adding a hardware module with solidified software to interconnect with the packet action interface. Developers can make independent and creative experiments, or even directly run a new architecture or protocol to deliver services, once the programmable platform of EIA is set up and opened to the public. And they can do this without the assistance of network device

manufacturers and network infrastructure operators.

2 Compatibility of EIA with Existing Architecture

Adopting the evolution strategy, EIA treats TCP/IP protocol stack as a special case among its several architectures to facilitate compatibility with existing networks. That is to say, existing IP and hierarchies still work in the network, but only act as one of the competitors. EIA does not add new protocol layers to existing architecture, as does Shim6^[4], but adds some interfaces. Hence, it is not necessary to change the old TCP/IP-based architecture. Compared with layer-adding methods, this approach can be more quickly implemented with existing devices, making the transition easier.

In the era of IP/IPX, several protocol stacks are already supported by the host. Current IPv4/IPv6 dual-stack structures can upload the data packets to a specified protocol stack by identifying the version number, and the original socket can receive the entire data frame. All these show that it is feasible for the host to support EIA. To enable their network devices to support EIA, manufacturers need only add interfaces for the developers of new architectures without changing the existing network.

Once network device manufacturers accept and support EIA, experimenters can program on Internet devices. As existing architectures and protocols

take different processing procedures from the newly-added modules, such experiments will not have an impact on the running of other protocols. Once a new architecture or protocol proves successful, it can be put directly into actual use, and the deployment of the new architecture or protocol will be quite smooth.

3 Comparison with Current Works

Among evolution approaches, the most typical is OpenFlow^[5]. OpenFlow provides Internet researchers with experiment methods for processing real flows. It adopts the structure of OpenFlow controller plus OpenFlow switches, where controller software instructs devices to perform related operations. This technology can be used in software defined networks. At present, OpenFlow can only run in local networks and can only support IPv4. That is, the flow table can only process IPv4 packet format and does not support any new protocol. The operations on the flow table are limited to flow-based forwarding, discarding, forwarding packets to the controller, and common forwarding, and the statistics function of the table is quite rough. New protocols may require more detailed statistics. As for packet processing, OpenFlow forwards the packets to specific external hardware for processing. This method is passable for trial networks, but for real networks (e.g. software defined networks), it suffers forwarding bottleneck and has poor scalability. Before it can be used in real networks, problems such as the scalability and redundancy of the controller, and the secure channel bottleneck (similar to the internal channel bottleneck of a router), must be solved. In contrast, EIA allows different protocol stacks to be installed in the devices as pluggable software modules. If the basic primitives are not enough, the hardware module with solidified software can be plugged. So EIA can be more easily applied in real networks than OpenFlow, which consists of remote controller and external hardware.

In OpenFlow, other operations, such as adding a flow table onto existing switches, and looking up each data packet in the table, are involved. EIA is free from the burden of table lookup and only forwards the packet to a related protocol stack based on the protocol ID. OpenFlow contains devices such as controller and NetFPGA, whereas EIA does not require any additional devices. EIA also provides developers with a real, programmable network; therefore, it is more likely to facilitate the smooth transition of the Internet than OpenFlow.

4 Conclusions

Currently, difficulty in implementing and applying new protocols in the core network layer and in network devices (e.g. switches and routers) hinders the development of core Internet technologies. Although researchers have proposed diversified Internet architectures, so far there has not arisen an architecture that can be widely applied to the existing Internet. This paper proposes EIA as a carrier of all network architectures. Adopting a special structure, EIA provides an ideal experimental platform for researchers of new architectures. It allows several architectures to be smoothly applied and to coexist in the future Internet. Thus, it promotes the continuous

evolution of core Internet technologies. EIA ends the situation where only one architecture is used for all networks, and lays a solid foundation for the graceful evolution of Internet.

On the programmable platform EIA, Internet researchers can experiment with their research results. As a result, EIA can better solve the existing problems of Internet, which in turn will speed up the evolution of the Internet and enable the Internet to become increasingly powerful. In the future, the study of EIA will mainly focus on clarifying the actions of the network device virtualization layer so as to determine the interfaces that should be provided by network device manufacturers.

References

- [1] BRADEN R, FABER T, HANDLEY M. From protocol stack to protocol heap: role-based architecture [J]. *Computer Communication Review*, 2003, 33(1): 17–22.
- [2] TOUCH J D, WANG Y S, PINGALI V. A recursive network architecture [R]. ISI-TR-2006-626. Marina del Rey, CA, USA: ISI, 2006.
- [3] DUTTA R, ROUSKAS G N, BALDINE I, et al. The SILO architecture for services integration, control, and optimization of the future Internet [C]// *Proceedings of IEEE International Conference on Communications (ICC '07)*, Jun 24–28, 2007, Glasgow, UK. Piscataway, NJ, USA: IEEE, 2007: 1899–1904.
- [4] NORDMARK E, BAGNULO M. Shim6: Level 3 multihoming shim protocol for IPv6 [R]. IETF RFC 5533. 2009.
- [5] The OpenFlow switch consortium [EB/OL]. [2009-01-31]. <http://openflowswitch.org>.

Biographies

Bi Jun



Bi Jun is a professor and director of the Network Architecture & IPv6 Division, Network Research Center of Tsinghua University. His research interests include network architecture and protocol, and next generation Internet. He has led over several national scientific research projects and was once awarded the second prize of National Science and Technology Achievement Prizes. He has published over 70 academic papers and has applied for 15 national patents of invention.

Lin Pingping



Lin Pingping is a doctoral candidate at the Network Research Center of Tsinghua University. She is mainly engaged in researching network architecture and protocol, and next generation Internet. She has published 15 academic papers.

Hu Hongyu



Hu Hongyu is a postdoctoral researcher at the Network Research Center of Tsinghua University. She is mainly engaged in research on network architecture and protocol, and next generation Internet. She has published 20 academic papers.

Roundup

ZTE and Vivo Unveil World's First Low-Cost Handset with Digital TV in Brazil

ZTE Corporation, a leading global provider of telecommunications equipment and network solutions, announced the Brazilian launch of N290, a low-cost cell phone with digital TV in Brazil on April 21, 2010.

Available exclusively through Vivo, the mobile phone are available in Rio de Janeiro and São Paulo, priced at BRL 399 for the pre-paid version and BRL 199 for the post-paid plan. It will be available in other Brazilian cities in coming weeks where Digital TV

signals are just available.

The ZTE N290 incorporates an MF645 3G modem for Digital TV reception, and supports Vivo's strategy to bring mobile Digital TV to its customers in time for the World Cup in South Africa in June.

With a 3.2 inch widescreen, the handset works with the Brazilian TV digital system ISBD-T which allows the access to all digital open channels at no cost. The new handset also features a 2.0 MP camera and

supports a Maxim 8G TF card.

Importantly, Digital TV services are free of charge. This means that once a handset is purchased, there is no need for a monthly subscription to access Brazil's broadcast TV.

According to Eliandro Avila, ZTE's President in Brazil, the manufacturer's goal is to offer an innovative and low-cost device in a market that, until very recently, only had expensive deals to offer with Digital TV.

(ZTE Corporation)