# IPv6 Network Virtualization Architecture for Autonomic Management of IPv6 Transition

Dujuan Gu
*Department of Research Institute of Information Technology, Tsinghua University*
*NSFOCUS Information technology CO., LTD*
*Beijing, China*
gudujuan@sina.com

Yibo Xue, Dongsheng Wang and Jun Li
*Department of Research Institute of Information Technology, Tsinghua University*
*Beijing, China*
{yiboxue, dongshengwang, junl}@tsinghua.edu.cn

*Abstract*—**We have entered the transitional period between IPv4 and IPv6. However, managing IPv4/IPv6 coexistence and transition involves some entirely new issues. Considering the management issues during IPv6 transition, we attempted to propose IPv6 network virtualization architecture (VNET6). VNET6 has its own management model based on abstraction. An evolution algorithm and autonomic control loop are specifically designed to automate provisioning of virtual resources and abstract IPv6 transition services. The evaluation of our deployment demonstrates that: VNET6, in a dynamic and autonomic managing manner, can facilitate IPv6 deployment and IPv6 transition services.**

***Keywords- Network virtualization; IPv4 and IPv6 coexistence; IPv6 transition; network management; network architecture.***

## I. INTRODUCTION

The current Internet has exposed IPv4 address space crunch. The regions of Asia, Oceania, Europe and the Middle East have exhausted their supply of IPv4 addresses [1]. With the supply of IPv4 addresses dwindling, the calls to transition to IPv6 are getting more intense. We have entered the transitional period between IPv4 and IPv6, and seen a coexistence of IPv4 and IPv6. However, it is taking on the additional burden of managing IPv4/IPv6 coexistence and transition that involves considering some entirely new issues. The heterogeneity of network devices and applications is a serious challenge facing network operators.

Unfortunately, the existing management techniques are inadequate to handle the heterogeneous environment of IPv4 and IPv6. Google's biggest challenge is not deploying IPv6 itself, but integrating IPv6 in all management procedures [2]. The latest trends in network management [3] lie on network automation, which reduces role of human operator and provides automaticity [4] (e.g. the information/knowledge flow used to drive control-loops). Auto-configuration is one of the key aspects of IPv6-based Internet [5]. We are seeking for advanced self-manageability of IPv6 deployment and IPv6 transition as a whole.

As of today, network virtualization [6] could facilitate the management of networks. Management applications are far easier to build with Onix [7] than without it. We intended to apply network virtualization to face the challenges of managing IPv6 transition. Then we proposed an IPv6 network virtualization architecture (VNET6) for managing IPv6 transitions, where borrowing from the innovative insights of network virtualization. A major challenge in this architecture is how to automatically deal with the efficient mapping among the application's end-to-end requirements of heterogeneous communication, virtual IPv6 resource and IPv6 transition services and the physical IPv4/IPv6 coexistence. Our main contributions in this paper are as follows:

- An IPv6 virtual network layer in VNET6 can be updated gradually through the seamless integration of IPv6 network resources and transition mechanisms. The network layer virtualization can facilitate IPv6 deployment and simplify the management of IPv4/IPv6 coexistence.

- IPv6 transition service abstractions encapsulate physical resources and IPv6 transition mechanisms into separate atomic function services. These refined services are used to provide flexible, reusing and unified services. Thus, the service abstractions give network operators more manageable and finer-grained control over IPv6 transition services.

- An autonomic virtualization process, together with a new evolution algorithm and an autonomic control loop, creates IPv6 virtual networks and abstract IPv6 transition services in a dynamic and autonomic manner. This process is an enabler for the autonomic management of application-specific end-to-end requirements for IPv4/IPv6 heterogonous communication.

We presented the VNET6 as the architecture of network virtualization for autonomic managing IPv6 transition. The implementation, deployment and management applications demonstrate the feasibility of VNET6. VNET6 can provide two aspects of network management: simplifying IPv6 deployment and improving IPv6 transition services to support service-oriented architecture. This architecture thus could make IPv6 deployment and transition in a stable and manageable manner.

The rest of this paper is organized as follows. Section II outlines our motivation in designing VNET6. Then in section III, we propose the network virtualization architecture of VNET6 and the related significant designs of this architecture: network layer virtualization, IPv6 transition service abstractions and an autonomic virtualization process. Section IV presents the main management facilities and implement. We

This work is jointly supported by Beijing Postdoctoral Research Foundation and the National Key Research and Development Program of China under Grant No. 2016YFB1000102, the National Key Technology Research and Development Program of China under Grant No. 2015BAK34B00 and the Natural Science Foundation of China under Grant NO. 61361126011.

describe the deployment and management applications in Section V. In section VI, the conclusions and future work are presented.

## II. MOTIVATION FOR VNET6

Before describing the architecture of our VNET6, we provide the background of network virtualization as a managing architecture for IPv6 transition, and outline our motivation in designing VNET6.

### A. Issues in Network management for IPv4/IPv6 Coexistence and Transition

The complexity of IPv6 transition process [8] makes it difficult to carry out network management. IPv6 transition in traditional network architecture is usually closely coupled with all the factors, such as network equipment, application servers and services and the peer or client systems that use or participate in those services. Hence it is highly difficult to manage IPv4/IPv6 heterogeneous networks and IPv6 transition services.

- The IPv4/IPv6 coexistent environment causes management tasks more complicated [9]. There may be the potential for IPv4 traffic to suffer at the hands of IPv6, or vice versa. Every network operator must consider managing IPv4 and IPv6 together.

- The introduction of various transition mechanisms [10] and corresponding new equipment poses new challenges to IPv6 transition management. Even worse, the challenge is deciding which of proposed transition mechanisms are the most practical for application's requirements.

- Some interoperability failure issues [11] indicate the managing challenge of IPv6 transition, because running different address families breaks application end-to-end communication. It is not acceptable to interrupt the incumbent applications and services during IPv6 transition process.

### B. Network Virtualization

Network virtualization [6] provides flexibility, promotes diversity, and increases manageability. Network virtualization enables the creation of logically isolated network partitions over shared physical network infrastructures [12]. In addition, network virtualization offers on-demand virtual networks customized for particular service and user requirements [13]. May the current trends of Network-as-a-service (Naas) [14] and Network Function Virtualization (NFV) [15] change the method for managing IPv6 transition?

However, Network virtualization and IPv6 have not been addressed together within a target architecture. How to manage IPv6 transition based on network virtualization remains unexplored. There is no research on adaptive provisioning of IPv6 transition services in the complex environment of IPv4 and IPv6 coexisting.

### C. Motivation for IPv6 network virtualization architecture

We intend to provide IPv6 network virtualization architecture (VNET6), namely, a network management architecture for IPv4 and IPv6 coexistence and IPv6 transition services using the promising way of network virtualization. Network virtualization is a powerful emerging technique with widespread applicability.

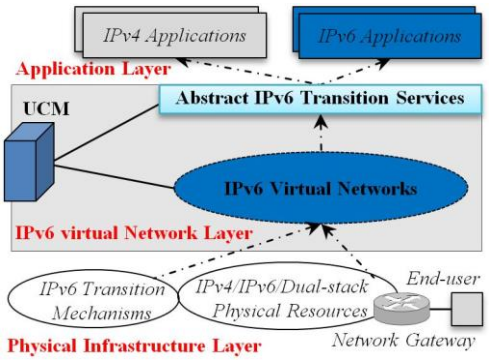


Figure 1. VNET6 architecture overview.

- The first purpose of VNET6 is to simplify the management tasks of IPv4 and IPv6. VNET6, using network virtualization, should create IPv6 virtual networks that are extracted from the IPv4/IPv6 coexistent environment. This IPv6 virtual environment should enable IPv6 resources and transition mechanisms to be integrated and supported gradually and seamlessly.

- The second purpose of VNET6 is to manage IPv6 transition services. The transition mechanisms defined in Internet Engineering Task Force (IETF) should be unified in VNET6, so as to provide high flexibility of adopting different transition mechanisms.

- The last purpose of VNET6 is to cope with the demands for dynamic IPv4 and IPv6 heterogeneous communication, especially with various IPv6 transition mechanisms. The management of IPv6 virtual networks and IPv6 transition services should become an enabler for ensure application's end-to-end communication.

## III. IPV6 NETWORK VIRTUALIZATION ARCHITECTURE

In this section, we describe the details of our VNET6.

### A. Overview

Exploring IPv6 transition scenarios, our VNET6 architecture can be logically viewed in three layers (Figure.1).

- *Physical infrastructure layer* is the complex IPv4 and IPv6 coexisting physical infrastructure of the current Internet. End-users connect Internet through network gateways in the physical networks of different IPs. Furthermore, various IPv6 transition mechanisms may have been deployed to migrate to IPv6. There are the aforementioned issues of managing IPv4/IPv6 heterogeneous networks and IPv6 transition services.

- *Virtual network layer* consists of the unified control and management (UCM), IPv6 virtual networks and abstract IPv6 transition services. UCM is a virtualization orchestrator, which extends network virtualization technology, so as to provide network layer virtualization and IPv6 transition service

TABLE I. SERVICE PROFILE EXAMPLES.

| S_ID | Service description |
|---|---|
| $T_4$ | Native IPv4 transport |
| $T_6$ | Native IPv6 transport |
| $T_{6>4}$ | Translation from IPv6 to IPv4 |
| $T_{4>6}$ | Translation from IPv4 to IPv6 |
| $T_{4>4}$ | NAT |
| $T_{4\text{-}4}$ | IPv4-in-IPv6 tunnel |
| $T_{6\text{-}6}$ | IPv6-in-IPv4 tunnel |

TABLE II. SUMMARY OF ABSTRACT IPV6 TRANSITION SERVICES

| Address Specification | Service profile | S_ID |
|---|---|---|
| IPv4 public address | Native IPv4 transport | $T_4 \mid T_{4\text{-}4}$ |
| IPv4 private address | NAT | $T_{4>4}+(T_4 \mid T_{4\text{-}4})$ |
| Global IPv6 address | Native IPv6 transport | $T_6 \mid T_{6\text{-}6}$ |
| IPv4 embedded IPv6 address | Translation from IPv6 to IPv4 | $T_{6>4}$ |
| Mapping IPv4 address | Translation from IPv4 to IPv6 | $T_{4>6}$ |
| 2002:IPv4 address/48 | 6to4 tunnel | $T_{6\text{-}6}$ |
| IPv6 prefix/64+5EFE:IPv4 address | ISATAP tunnel | $T_{6\text{-}6}$ |
| 3FFE:831/32 | Teredo tunnel | $T_{6\text{-}6}+T_{4>4}$ |

abstractions. On one hand, the network layer virtualization creates IPv6 virtual networks to be built with incremental scalability and without requiring changes to the existing networks itself. On the other hand, IPv6 transition service abstractions maintain abstract IPv6 transition services based on the resources of physical infrastructure layer. This layer enables dynamic construction and management of IPv6 virtual networks and abstract IPv6 transition services, according to application's dynamic requirements.

- *Application layer* includes IPv6 applications and IPv4 applications of the current Internet. They, over the virtual network layer, are provided abstract IPv6 transition services to ensure application's end-to-end communication in the complex IPv4 and IPv6 coexisting environment.

In the rest of this section, we describe how VNET6 virtualizes physical infrastructure into an IPv6 virtual network and provides abstract IPv6 transition services for applications automatically. UCM focuses on all visualization specific management tasks necessary in this VNET6 architecture.

### *B. Network Layer Virtualization*

Network layer virtualization provides a very effective method for managing the coexisting networks of IPv4 and IPv6 and various IPv6 transition mechanisms.

Physical nodes of the physical infrastructure layer can dynamically register or unregister in the Network Information Base (NIB) for physical infrastructure. On one hand, IPv6 topology discovery detects the presence of new IPv6 nodes, the absence of IPv6 nodes due to changing network conditions by using IPv6's Neighbor Discovery (ND) protocol. On the other hand, UCM mainly adopts SNMP to get related MIBs of physical IPv6 resources and transition mechanisms, which are maintained in the NIB for physical infrastructure.

With the available resources in the NIB for physical infrastructure, UCM supports dynamic virtual consolidation

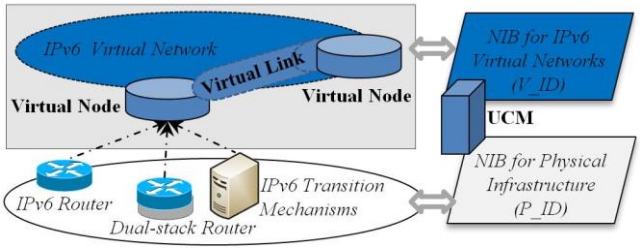


Figure 2. An IPv6 virtual network.

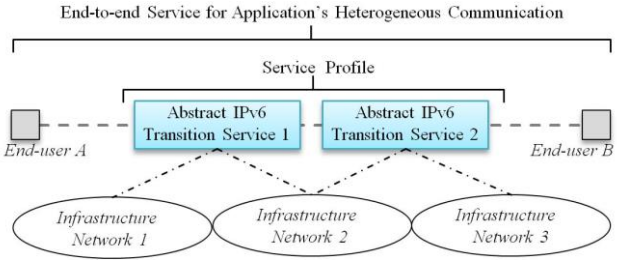


Figure 3. IPv6 transition services in VNET6.

mechanism [16] for IPv6 virtual networks. An IPv6 virtual network can be gradually updated through the seamless integration of IPv6 network resources and transition mechanisms. Due to the gradual integration of IPv6 technology into existing IPv4 Internet, an IPv6 virtual network is a dynamic network composition. Hence building this IPv6 virtual network provides flexible and incremental scalability.

An IPv6 virtual network (Figure.2) has much simpler and more abstract topology than the underlying physical infrastructure. It consists of virtual nodes and virtual links. UCM on-demand integrates various IPv6 physical resources and transition mechanisms into virtual nodes. A virtual link between two virtual nodes is a logical softwire [17] tunneling data packets. UCM maintains topologies of IPv6-based virtual networks in NIB for IPv6 virtual networks. Deploying IPv6 for network management purposes first, an IPv6 virtual network offers network operators an opportunity to get comfortable with the development of additional IPv6 capabilities.

This network layer virtualization creates IPv6 virtual networks that are decoupled from existing IPv4 physical resources. Hence, it ensures the physical infrastructure can increasingly integrate with IPv6 resources and new IPv6 transition mechanisms.

### *C. IPv6 Transition Service Abstractions*

Applications, as the practical end-users of IPv6 virtual networks, are offered IPv6 transition services for application-specific requirements during IPv6 transition. Figure.3 illustrates the representation of an end-to-end service for application's heterogeneous communication over the physical infrastructure of IPv4/IPv6 coexistence. This end-to-end service meets a service profile of the transport networks; this service profile complies with IPv6 transition services specified by end-to-end communication requirements. It is an IPv6 transition service combination, which is a chain of abstract IPv6 transition services through an application's

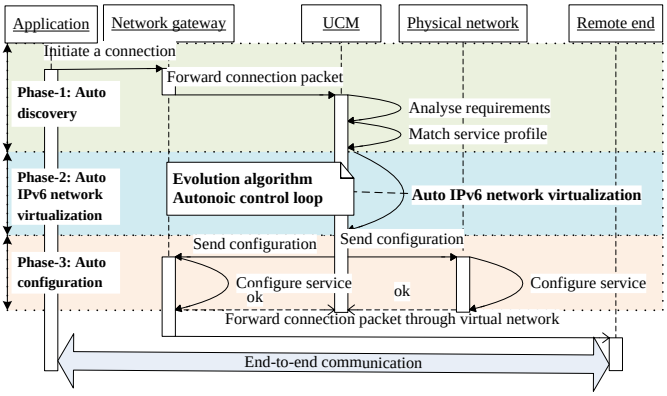


Figure 4. Autonomic virtualization process.

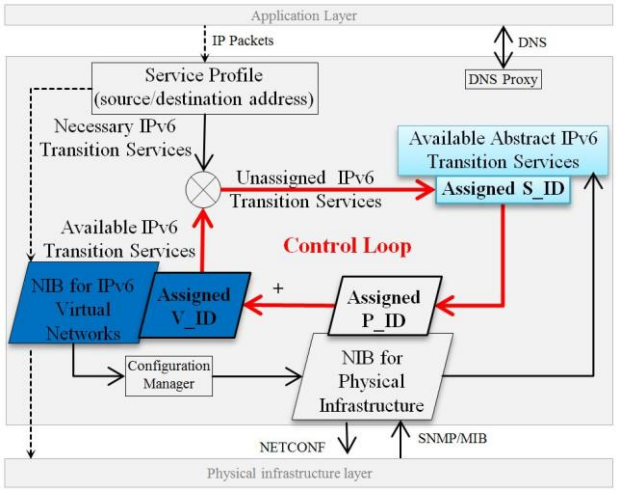


Figure 5. Autonomic control loop in UCM.

communication path. Virtual nodes encapsulate physical resources and IPv6 transition mechanisms into abstract IPv6 transition services (Table 1).

Each service addresses a separate atomic function based on IPv6 transition service level abstraction, and thus the service is more flexible and finer-grained than recent IPv6 transition techniques. For example, DS_Lite [18] takes the best of a combined mechanism of dual stack, tunneling, NAT and IPv6; then the available abstract IPv6 transition services: $T_6$, $T_{4\text{-}4}$ and $T_{4>4}$, are bound with virtual nodes and physical service nodes. An abstract IPv6 transition service is a well-defined and self-contained function performed on application packets.

Abstract IPv6 transition services enable a manageable and easy IPv6 transition service system. Even if two of more transition mechanisms are simultaneously adopted in an existing IP network, these transition mechanisms are extracted abstract IPv6 transition services. When several same abstract IPv6 transition services are available, the most adaptive one meets application-specific requirements to ensure end-to-end communication by an evolution algorithm, as described below.

### *D. Autonomic Virtualization Process*

This section provides three phases of some automated behaviors of allocating virtual networks and abstract services in a dynamic and autonomic manner. Figure.4 outlines the autonomic virtualization process to set up IPv6 virtual networks according to application-specific requirements in VNET6.

*1) Phase-1: Auto discovery for application-specific requirements of IPv6 transition services.*

When one application initiates a connection to Internet, it is a necessary to obtain a destination IP address by the DNS proxy of UCM. DNS Proxy returns a suitable answer in response to the DNS resolving request of upper applications. It is exactly the same as DNS64 [19] and DNS46 [20] in that this module is involved with the mapping IPv6 prefix of IPv4-IPv6 translators, which provide IPv6 transition services.

Upon receiving the initial connection request, the network gateway only forwards this initial connection packet to UCM, in that the default route of this gateway guides this packet with the next hop to UCM.

UCM firstly analyses the source and destination addresses of this IP connection packet, and then matches a service profile. As indicated in Table 2, the source and destination addresses are the valuable information for determining which IPv6 transition services should be adopted in a coexisting network. This service profile complies the necessary abstract IPv6 transition services specified by end-to-end communication requirements.

*2)* Phase-2: Auto IPv6 network virtualization based on service profiles.

Different service profiles enable IP packets to run in different virtual networks, now that packets are demultiplexed into the appropriate virtual network on their packet headers. UCM verifies that there are sufficient abstract IPv6 transition services available over an IPv6 virtual network. For the unassigned IPv6 transition services of an application's service profile, UCM identifies the candidate physical resources and IPv6 transition mechanisms in NIB for physical infrastructure, and then updates this IPv6 virtual network with the mapping among unassigned IPv6 transition services, the candidate physical infrastructure and this IPv6 virtual network. Eventually, these related IPv6 transition services are updated from the unassigned state to the assigned state.

Compare with the current network virtualization research, two novel proposals are designed for this autonomic virtualization process, especially. On one side, an autonomic control loop focuses on controlling and managing the virtualizing mapping behaviors in an autonomic way. On the other side, an evolution algorithm is applied to ensure an optimal end-to-end path with the optimal services. UCM assigns a specific set of physical nodes for the optimal services. Accordingly, IPv6 virtual network is maintained by UCM based on application's specific requirements. These two novel proposals are described as following sections.

*3) Phase-3: Auto configuration for IPv6 transition services*

For the assigned IPv6 transition services, UCM sends configuration commands via NETCONF interface to those specific physical nodes, including the network gateway. These configuration commands are related to the assigned abstract IPv6 transition services. Meanwhile, the network gateway is added a new route, so that the following packets can be directly transported through the optimal end-to-end path.

With above autonomic behaviors, VNET6 enables underlying infrastructure to automate provisioning of virtual resources and abstract IPv6 transition services. This process not only supports dynamic IPv6 virtual networks provisioning, mapping, and management, but also effectively ensures optimal end-to-end paths in IPv4/IPv6 heterogonous environment.

### *E. Evolution Algorithm Design*

An evolution algorithm is specifically designed to create optimal paths for application-specific end-to-end communication. These optimal paths are based on the available abstract IPv6 transition services of UCM. Nevertheless the available abstract IPv6 transition services may be imperfect for application's service profiles. On condition that UMC maintains the set of available abstract IPv6 transition services and the set of service profiles, VNET6 is able to dynamically adapt to optimal end-to-end paths with suitable abstract IPv6 transition services for service profiles.

We elaborate on this evolution algorithm. A service profile, which is considered as a set of constraints on the properties required in the necessary IPv6 transition services. In effect, UCM parses the constraints specified by the service profile, and then evaluates the suitability of any matching abstract IPv6 transition services. This suitability for application specific end-to-end communication is a metric of link performance attributes, e.g. distance, bandwidth, delay, loss. A path could present different values of such a metric according to abstract IPv6 transition services that may be used even when attached to the same network. For example, the suitability is the distance from a matching one to the network gateway of this application; the closest one is assigned among these matching ones. It is a simple evaluation policy to keep IPv6 transition mechanism simple enough to be hosted in performance constrained entities. The optimal path is bound to the most suitable abstract IPv6 transition services, which are the services with Assigned S_ID. This evolution algorithm is a critical to application's optimal end-to-end paths.

UCM brings every immediate response to a dynamic application's connection. IPv6 physical resources and services are to be preferred to IPv4 ones in IPv6 virtual networks. In a way that is IP packet aware, the evolution algorithm ensures optimal paths with suitable abstract IPv6 transition services for application's end-to-end communication.

### *F. Autonomic control loop*

Above autonomic virtualization process is achieved under an autonomic control loop (Figure.5), which is a closed-loop control of the mapping among abstract IPv6 transition services, physical infrastructure and virtual networks in UCM.

First, UCM allocates the dedicated abstract IPv6 transition services for the unsigned IPv6 transition services of an application's service profile. Then the service identities S_ID of these dedicated services are assigned (Assigned S_ID).

Second, UCM is required to enable physical infrastructure to process the services S_ID. Thus the physical resources in NIB for physical infrastructure are mapped into the services S_ID; then are labeled with assigned identities (Assigned P_ID).

Last, if a virtual node is integrated with the physical resources Assigned P_ID, it should be allocated assigned identities (Assigned V_ID) to support those dedicated services Assigned S_ID. This virtual node in an IPv6 virtual network can meet the application-specific requirements.

IPv6 transition services of an IPv6 virtual network can be updated automatically to meet different application-specific requirements, while network layer virtualization seamlessly integrates new IPv6 physical resources and transition mechanism into this IPv6 virtual network. The control loop enables UCM to be an autonomic manager, which is responsible for application's specific requirements, physical infrastructure and virtual networks management.

## IV. UCM MANAGEMENT FACILITIES AND IMPLEMENT

UCM in VNET6 touches the existing Network Management System (NMS), and the new management facilities of network virtualization are applied in UCM. Particularly, the auto IPv6 virtualization process simplifies the related management facilities in UCM.

### *A. Extensions for Management facilities of NMS*

- *Topology management* plays a critical role for the complex IPv4 and IPv6 coexisting environment. UCM is implemented to enhance the management capabilities of existing network management software tools for topology. It maintains the network topologies not only of the physical infrastructure layer in NIB for physical infrastructure, but also of the virtual layer in NIB for IPv6 virtual networks. Specially, IPv6 should be enabled on the SNMP agents and MIBs not of all management components, but of new IPv6 components.

- *DNS Proxy* provides DNS extensions for IPv6 transition mechanisms. Meanwhile, UCM gathers IPv6 prefix and IPv4 address is assigned inside the managed network to manage a coexisting network.

- *Configuration manager* simplifies managing configuration files for the IPv6 transition services of the physical infrastructure layer. It enables the real-time changes of configurations to response to application's dynamic service profile.

### *B. New Management facilities in VNET6*

- *Management of network layer virtualization*: UCM can provide IPv6 virtual networks with incremental scalability for smooth IPv6 transition. The dynamic virtual consolidation mechanism simplifies the management of IPv6 physical

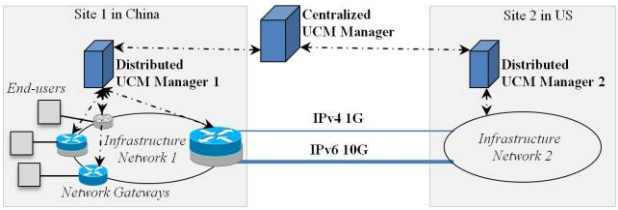


Figure 6. Hybrid Deployment of UCM.

networks and various transition mechanisms in IPv6 virtual networks.

- *Management of IPv6 transition services*: This service management adds the necessary application-to-IPv6 transition service-to-network awareness. UCM can handle management tasks: extracting abstract IPv6 transition services from IPv6 transition mechanisms and evaluates the suitability of the abstract IPv6 transition services.

### C. Implement of UCM

UCM is a control and management software tool. It is implemented to enhance the management capabilities of existing network management software tools for topology. We run this software tool on Intel Xeon E5645 with 24 CPU cores (2.4GHz), 66G memory and 64-bit CentOS as the operating system. The information in NIB for physical infrastructure is obtained by the use of SNMP protocols and related MIBs. The service configuration is implemented via NETCONF interface. The patch of ecdysis-bind [21] is modified for the DNS proxy. The autonomic control loop is implemented to manage the mapping among abstract IPv6 transition services, physical infrastructure and virtual networks. The evolution algorithm deals with service profiles to automatically create optimal paths for end-to-end communication. Those appropriate management facilities are developed and applied in UCM.

The next step implementation will extend Flowvisor [22] to support UCM function as a virtualization orchestration, which unifies the control and management of IPv6 transition.

## V. Management Applications and Deployment

To validate our design, this section describes the deployment of UCM, and two application areas of VNET6.

### A. Deployment of UCM

In our hybrid deployment (Figure.6), we have a centralized manager with a unified global view, as well as some distributed UCM managers for site management.

The centralized UCM manager, together with these distributed UCM managers, controls and manages the overall physical infrastructure layer. This centralized manager simplifies managing IPv6 transition mechanisms that are related to application's specific requirement for end-to-end communication. The centralized UCM manager means fast and easy to manage with a unified global view.

The site management is limited the system capabilities in accordance with locally-specified IPv6 transition mechanisms. It provides an integrated management to implement the most popular IPv6 transition mechanisms managed by these distributed UCM managers. A distributed UCM manager is in local centralized deployment. This manager controls and manages local overall physical infrastructure layer. It is flexible to apply new IPv6 transition mechanisms to a specific transition scenario.

This hybrid deployment leverages the benefits of the simple control as in the centralized UCM manager with the scalability and flexibility of the distributed UCM managers.

TABLE III. End-to-end performance

| Performance | Without VNET6 | With VNET6 |
|---|---|---|
| Round-trip delay (RTT) | 185.995ms | 167.04 ms |
| TCP Throughput | 18.2570 Mbps | 62.9247 Mbps |

### B. Application Area: Coordinating WAN Link Performance across Sites

WAN links are typically provisioned with significant management requirements. Based on China-US international data placement laboratory, an IPv4 WAN link has 1Gbps bandwidth, whereas an IPv6 WAN link has 10Gbps bandwidth. Two distributed UCM managers share the link state information with the centralized UCM manager.

The centralized UCM manager allows network operators to define the suitability of IPv6 transition services as the bandwidth of WAN links. Thus all UCM managers prefer the abstract IPv6 transition service $T_{4\text{-}4}$ to $T_4$. Based on this evaluation policy, IPv4 and IPv6 packets are demultiplexed into an IPv6 virtual network. An abstract IPv6 transition service $T_{4\text{-}4}$ is provided by integrating IPv4-in-IPv6 tunnel physical ends and configuring IPv4-in-IPv6 softwire mechanism for IPv4 packets. Hence, IPv4 packets within VNET6 provided higher end-to-end performance (Table 3).

### C. Application Area: Flexible IPv6 Transition Services within a Site

Most transition mechanisms meet the particular deployment. Multiple parallel mechanisms are deployed in order to fulfill diverse IPv6 transition requirements. The concurrent deployment and management of multiple mechanisms however are not cost-effective. Further, it is difficult to determine effective transition mechanisms for application-specific requirements.

The distributed UCM manager of site 1 in China has managed several abstract IPv6 transition services $T_6$, $T_4$, $T_{4\text{-}4}$, $T_{4>4}$ and $T_{6>4}$. VNET6 can maximize the re-use of existing abstract IPv6 transition services. It is through the distributed UCM manager that abstract services can dynamically compose different service combination over time, such as DS-Lite with $T_6+T_{4\text{-}4}+T_{4>4}$, Lightweight 4over6 with $T_{4>4}+T_6+T_{4\text{-}4}$. These abstract IPv6 transition services can provide a cost-effective and flexible means to define an individual service composition for application-specific requirements.

### D. Lessons learnt from the Deployment of VNET6

- *Incremental IPv6 Deployment*: VNET6 can manage IPv6 network resources and transition mechanisms to be

integrated into existing physical infrastructure gradually and seamlessly. A small set of upgraded IPv6 nodes use Softwire tunnels to form IPv6 virtual networks topology over physical infrastructure. In contrast, dual stack requires complex end-to-end upgrades and managing numerous upgraded devices. Unlike managing the co-existing network under various transition mechanisms [23], VNET6 can dynamically manage IPv6 transition service provisioning, by employing a flexible service-oriented architecture.

- *Flexible IPv6 transition services management*: By IPv6 transition service abstractions, VNET6 can be flexible, manageable and available for a broad range of IPv6 transition. In contrast, a software defined transition approach [24] has unified current IPv6 transition mechanisms. This approach, however, should deploy OpenFlow switches. Besides unifying IPv6 transition, abstract IPv6 transition services in VNET6 are more flexible and finer-granularity than current IPv6 transition mechanisms. Further, VNET6 can automatically ensure application's end-to-end communication through the optimal path with suitable abstract IPv6 transition services.

## VI. CONCLUSIONS

Considering management issues during IPv6 transition, we proposed VNET6: a combination of a unified management model for physical infrastructure layer and a service-oriental model for application layer. To minimize network operator involvement, this combination appears to align with the autonomic management of IPv6 transition.

VNET6 has its own management model based on abstraction. On one side, the network layer virtualization provides network abstractions, which increase manageability for the coexisting IPv4 and IPv6 environment. On the other side, IPv6 transition service abstractions meet the dynamic application's requirements during IPv6 transition. This management model facilitates IPv6 transition considerably smooth.

This paper only proposed a preliminary IPv6 network virtualization architecture. In future research, we will focus on the improvement of VNET6, especially how to explore the overall system performance facing large-scale application requirements, how to refine the design of IPv6 network virtualization architecture and algorithm, and how to revise our implementation in the wide range of networks. The further investigation of network virtualization, future network and IPv6 transition technologies is also our concern.

## VII. ACKNOWLEDGMENTS

Many thanks go to the president of IPv6 forum Latif Ladid and Dr. Xiaoyu Yang for their helpful discussions and feedbacks. Thank our colleagues who contributed towards this project for their collaboration.

## VIII. REFERENCES

[1] G. Huston, "Ipv4 address report", http://www.potaroo.net/tools/ipv4/index.html.

[2] H. Babiker, I. Nikolova and K. K. Chittimaneni, "Deploying ipv6 in the google enterprise network. Lessons learned", in Practice & Experience Report): https://www. usenix. org/conference/lisa11/deploying-ipv6-google-enterprise-network-lessons-learnedpractice-experience.

[3] S. KUKLINSKI and P. CHEMOUIL, "Network management challenges in software-defined networks", Ieice T Commun, vol.97, no.1, pp. 2-9 2014.

[4] R. Chaparadza, S. Papavassiliou, T. Kastrinogiannis, M. Vigoureux, E. Dotaro, A. Davy, K. Quinn, M. Wódczak, A. Toth and A. Liakopoulos, "Creating a viable evolution path towards self-managing future internet via a standardizable reference model for autonomic network engineering", in Future Internet Assembly, pp. 136-147, 2009.

[5] R. Chaparadza, R. Petre, A. Prakash, F. Németh, S. Kukliński and A. Starschenko, "Ipv6 and extended ipv6 (ipv6++) features that enable autonomic network setup and operation", in Access networks, eds. R. Szabó, H. Zhu, S. Imre and R. Chaparadza, pp. 198-213, Springer Berlin Heidelberg, 2011.

[6] A. Wang, M. Iyer, R. Dutta, G. N. Rouskas and I. Baldine, "Network virtualization: Technologies, perspectives, and frontiers", J Lightwave Technol, vol.31, no.4, pp. 523-537 2013.

[7] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue and T. Hama, "Onix: A distributed control platform for large-scale production networks", in OSDI, pp. 1-6, 2010.

[8] J. Bi, J. Wu and X. Leng, "Ipv4/ipv6 transition technologies and univer6 architecture", International Journal of Computer Science and Network Security, vol.7, no.1, pp. 232-243 2007.

[9] I. P. Hsieh and K. Shang-Juh, "Managing the co-existing network of ipv6 and ipv4 under various transition mechanisms", in Information Technology and Applications, 2005. ICITA 2005. Third International Conference on, pp. 765-771 vol.762, 2005.

[10] S. Miyakawa, "Ipv4 to ipv6 transformation schemes", Ieice T Commun, vol.93, no.5, pp. 1078-1084 2010.

[11] A. Maula, "A review and qualitative analysis of ipv6 and ipv4 interoperability technologies", Seminar on Internetworking, pp. 2-6 2010.

[12] T. Anderson, L. Peterson, S. Shenker and J. Turner, "Overcoming the internet impasse through virtualization", Computer, vol.38, no.4, pp. 34-41 2005.

[13] J. He, R. Zhang-Shen, Y. Li, C.-Y. Lee, J. Rexford and M. Chiang, "Davinci: Dynamically adaptive virtual networks for a customized internet", in Proceedings of the 2008 ACM CoNEXT Conference, pp. 15, ACM, 2008.

[14] P. Costa, M. Migliavacca, P. Pietzuch and A. L. Wolf, "Naas: Network-as-a-service in the cloud", pp. 1-1, USENIX Association, 2012.

[15] "Network functions virtualisation", http://www.etsi.org/technologies-clusters/technologies/nfv.

[16] D. Gu, X. Liu, G. Qin, S. Yan, Z. luo and B. Yan, "Vnet6: Ipv6 virtual network for the collaboration between applications and networks", J Netw Comput Appl, vol.36, no.6, pp. 1579-1588, November 2013.

[17] X. Li, S. Dawkins, D. Ward and A. Durand, "Softwire problem statement", IETF RFC4925, July 2007.

[18] Y. Lee, A. Durand, J. Woodyatt and R. Droms, "Dual-stack lite broadband deployments following ipv4 exhaustion", in August IETF, RFC 6333, 2011.

[19] M. Bagnulo, P. Matthews, A. Sullivan and I. Beijnum, "Dns64: Dns extensions for network address translation from ipv6 clients to ipv4 servers", in April IETF, RFC 6147, 2011.

[20] X. Li and C. Bao, "Dns46 for the ipv4/ipv6 stateless translator", draft-xli-behave-dns46-for-stateless-04 (work in progress) 2013, July.

[21] S. Perreault, J.-P. Dionne and M. Blanchet, "Ecdysis: Open-source dns64 and nat64", AsiaBSDCon (March 2010) 2010.

[22] R. Sherwood, G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, N. McKeown and G. Parulkar, "Flowvisor: A network virtualization layer", OpenFlow Switch Consortium, Tech. Rep 2009.

[23] I.-P. Hsieh and S.-J. Kao, "Managing the co-existing network of ipv6 and ipv4 under various transition mechanisms", in Information Technology and Applications, 2005. ICITA 2005. Third International Conference on, pp. 765-771, IEEE, 2005.

[24] W. Xia, T. Tsou, D. R. Lopez, Q. Sun, F. Lu and H. Xie, "A software defined approach to unified ipv6 transition", in Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM, pp. 547-548, ACM, 2013.