Elektrotechnik & Informationstechnik (2009) 126/7/8: 250–259. DOI 10.1007/s00502-009-0652-2

Towards the Future Internet: virtual networks for convergent services

K. Tutschku

In this paper we will investigate the deficiencies and the achievements of today's Internet. We outline how Network Virtualization (NV) can overcome the shortfalls of today's networks, how it paves the way for the Future Internet, and how it can facilitate the convergence of currently different networks.

The major building blocks of NV are the a) use of application-specific routing overlays, b) the safe consolidation of resources by OS virtualization on a generic infrastructure, and c) the exploitation of the network diversity for performance enhancements and for new business models, such as the provisioning of intermediates nodes or path oracles.

Furthermore, we detail a more elaborate concept for network virtualization, which is denoted as Transport Virtualization (TV). TV transfers the concept of location independence of resources to the area of data transport in communication networks.

Keywords: Future Internet; network virtualization; convergence

Aufbruch zum Internet der Zukunft: virtuelle Netze für konvergente Dienste.

Der Beitrag diskutiert die Unzulänglichkeiten und die Erfolge des heutigen Internets und beschreibt, wie das Konzept der ,,Netzvirtualisierung'' diese Mängel beseitigen kann. Insbesondere stellt der Beitrag dar, wie ,,Netzvirtualisierung'' das Internet der Zukunft ermöglicht und wie es zu einer geeigneten Konvergenz heute noch unterschiedlicher Kommunikationssysteme führt.

Die Bausteine von ,,Netzvirtualisierung'' sind a) die Nutzung von anwendungspezifischen Routing-Overlays, b) die sichere und zuverlässige Virtualisierung von Ressourcen auf Ebene der Betriebssysteme und c) die Nutzung der Diversität in Netzen.

Weiters führt der Beitrag in das Konzept der ,,Transport-Virtualisierung'' ein, das eine Übertragung der Ideen der ,,Netzvirtualisierung'' für einen Datentransportdienst darstellt.

Schlüsselwörter: Internet der Zukunft; Netzvirtualisierung; Konvergenz

Received: May 24, 2009, accepted May 26, 2009

1. Introduction

Communications networks are currently designed with the focus on integration. They are able to handle all kinds of communication traffic in a single network. Therefore, they apply a consistent address space, routing, and resource sharing. Prominent examples for this concept are the B-ISDN architecture and the IP hourglass model. However, it becomes increasingly obvious that the integrative approach might be useful when mainly connectivity is required, but it turns out to be inappropriate for specific blends of traffic (e.g. VoIP stream mixed with transactional traffic), certain technologies (e.g. wireless networks), and for fostering economic competition among providers (e.g. multi-domain services).

In order to overcome the impasses of the current systems and *ossification* of their protocols, a new technology is currently under development which is denoted as *Network Virtualization (NV)*. NV allows the simultaneous operation of multiple logical networks (also known as *overlays*) on a single physical platform. NV permits distributed participants to create almost instantly their own network with application-specific naming, topology, routing, and resource management mechanisms, such as server virtualization enables users to use even a whole computing center arbitrarily as their own personal computer. Therefore, NV received recently tremendous attention since it is expected to be one of the major paradigms for the Future Internet as proposed by numerous international initiatives on future networks, e.g. PlanetLab (USA, International) (*Roscoe, 2005*), GENI (USA) (*GENI Consortium, 2006; GENI Planning Group, 2006*), AKARI (JAPAN) (*NICT, 2007*), and G-Lab (Germany) (*Tran-Gia, 2008*).

In this paper we argue why and how NV constitutes a powerful technology for the future Internet and how it improves convergence of networks and services. We will outline major building blocks for NV and detail a more elaborate concept for network virtualization, which is *Transport Virtualization (TV)*. With TV, we transfer the concept of location independence of resources to the area of data transport in communication networks. We will outline the capability of TV by an example for a transport mechanism for high throughput data transmission in routing overlays. The transport mechanism is based on *Concurrent Multipath (CMP)* transmission, also known as *striping*, and discuss briefly its complexity.

The paper is organized as follows. First, we will discuss the deficiencies and achievements of today's Internet and its application and outline the new needs for convergence. After that, we discuss how recent results in overlay technology, network diversity, and operating system virtualization contribute to the capabilities of NV. The next section will detail the concept of Transport Virtualization and outline the capability of TV using the example of CMP transmission. Finally, the paper will conclude with a short summary and a discussion of requirements for a convergent Future Internet reference architecture model.

2. Some deficiencies and achievements of today's Internet

When addressing the shortfalls of the current Internet, the discussion usually focuses quickly on architectural and operational issues

Tutschku, Kurt, Univ.-Prof. Dr., Universität Wien, Professur für Future Communication, Universitätsstraße 10/T11, 1090 Wien, Österreich
(E-mail: kurt.tutschku@univie.ac.at)

such as the anticipated lack of IP addresses (*CNet News, 1999*), the complexity of today's management (*Clark et al., 2003*), or the insufficient extensibility of today's IP protocol family denoted as *protocol ossification* (*Handley, 2006*). However, this discussion neglects often the requirements of the future applications and users. Since it is particularly hard to foresee the future, we restrict this discussion to accepted requirements of current applications and usages, which are not solved, even until today. This approach provides a benchmark whether a future Internet architecture will be at least superior to today's system by solving current problems. After the discourse of the deficiencies, we acknowledge that the current Internet is still a success story. We will outline selected achievements of the current system and its applications and investigate what we can learn from these successes for a future system.

2.1 Deficiencies

A major deficiency of today's Internet is still the *missing control of the end-to-end quality of service (QoS).* Many solutions such as IntServ or DiffServ have been developed and certain *QoS islands* have been formed depending on the technology and the capabilities of the providers applying these mechanisms. As a result, a user may ask: ''Why can't I take advantage of these islands?''.

Although the protocols of the current Internet have been designed for catastrophic failures, the *reliability* of the current system and its application is very poor. However, the sophisticated resilience concepts exist, e.g. for Multiprotocol Label Switching (MPLS), and are available at experienced Internet Services Providers (ISPs). Again, this fact raises the question why the reliability islands can't be exploited for better system or service reliability.

Finally, a major deficiency is the lock-in of users to their ISPs which suppresses competition among ISPs. John Crowcroft expressed this shortfall precisely in a posting to the End2End-Interest Mailing on April 26th 2008: ''. . . I can go on the web and get my gas, electricity, . . . changed, why is it not possible to get a SPOT price for broadband Internet?". This feature is similar to the ''call-by-call'' provider selection scheme in some deregulated telephone service markets such as Germany. After passing the access system, the traffic is forwarded to the appropriate transport network. Currently, a user may ask: ''Why is the data traffic not being relayed after the access network to the most cost-efficient ISP selected by me?''.

2.2 The need for a new concept of convergence

The services in classical communication networks, such as ISDN or GSM, are rather platform-dependent. The increased application of abstraction layers which provide service interfaces that are independent from the underlying physical network, like the Internet Protocol (IP) or overlay techniques, permits services to be consumed in a variety of wireless and wireline networks such as ADSL, WLAN, or UMTS. Hence, the transition from network-centric services to application-centric *multi-network services* (*Tutschku, Tran-Gia, Andersen, 2008*) has occurred as shown in Fig. 1.

Figure 1(a) depicts the relationship of applications, services, service providers, and network providers in a typical single provider legacy PSTN (Public Switched Telephone Network). With the application of the IP protocol stack a middle layer has been introduced, cf. Fig. 1(b). Applications and services could have been provided across different technical domains, however, certain restrictions were still preserved such as limited resource management and quite inflexible routing. By the introduction of application-specific overlays, most of these limitations have been overcome and in future networks the generic overlays may even simplify today's layering architecture, cf. Fig. 1(c).

Classical services and applications, such as voice, were typically provisioned by network operators. The successful *Peer-to-Peer (P2P)*

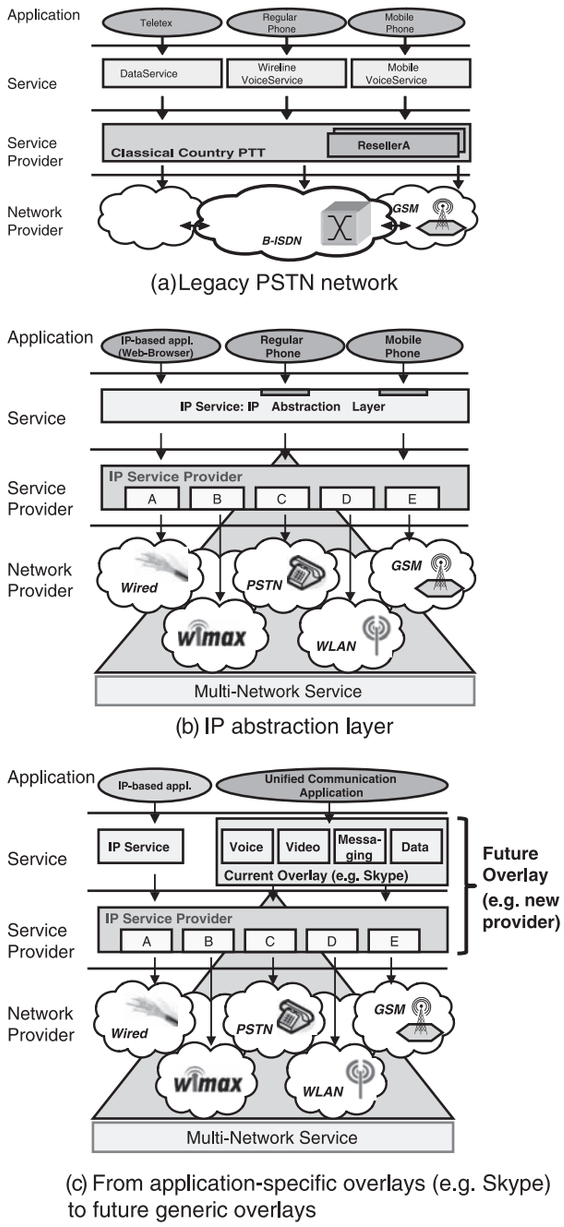

Fig. 1. Evolution to multi-network service towards future overlays

content delivery applications, cf. Sect. 2.3.1, however, have blurred the boundary between content providers and consumers. In addition, they showed that edge-based communities could easily design, deploy and offer services. The new services reveal *edge-based intelligence* and form *overlays*, which are virtual relationships within real physical network, with application-specific naming and routing concepts.

Furthermore, users transfer their social behavior increasingly to networks and networked applications. *Social networking web sites* like YouTube (*YouTube Inc., 2006*) or MySpace (*MySpace Inc., 2006*) with *user-generated content* became tremendously popular. They permit the users to structure the use of the information according to their specific social interests or social relationships.

The ubiquity and availability of networked applications in today's wired and wireless networks combined with an increasing commer-

cial significance has led to a demand for highly *dependable networks and services*. Hence, automatic resilience, fault management, and overload mechanisms have been introduced on different layers. Examples include fast reroute mechanisms on the network layer (*Martin, Menth, 2006*), multi source download in P2P content distribution networks, cf. Sect. 2.3.1, or dependable overlay services for supporting vertical handovers in mobile networks (*Tutschku, Nakao, 2009*).

The success of virtual mobile operators (*Varoutas et al., 2002*) or of the P2P VoIP service Skype (*Skype Technologies S.A., 2003*) has shown that *virtualization of telecommunication operators, services or applications* are no longer a concept discussed only by researchers. For example, Skype replaced central indices for user locations and virtualized the indices by distributed software running on the end user clients. In addition, the simple and open programming interfaces of the Skype software permits third parties to develop rapidly numerous commercial services on top of it, e.g. on-line translation services (*Skype Limited, 2006*).

As a result of the above outlined trends, a new concept for the convergence in networks is needed. While the convergence concept of current networks addresses at the interconnection of networks and services, has the new approach to aim at the contribution, sharing, and aggregation of resources. We will see below that Network Virtualization is targeting in particular at these features.

2.3 Achievements of the current Internet

Despite all its deficiencies, the current Internet has also facilitated never expected ways of using and operating networks efficiently.

2.3.1 P2P-based content distribution

One of the fastest revolutions in Internet usage was the development of P2P content distribution applications. P2P systems are a specific type of distributed systems, which consist of equal entities, denoted as *peers* that share and exploit resources in a cooperative way by direct end-to-end exchanges on application layer.

P2P content distribution systems are used to distribute very large video and audio files like DVDs or CDs. The first major P2P content distribution application was Gnutella (*Clip2, 2001*), released in 1999. After only four years, P2P contribution applications have become the major source of Internet traffic. Table 1 shows the shares of the different traffic types at a residential access system (*Gabeiras, 2004*).

Table 1. Typical traffic distribution residential access systems

Type	P2P	Not identified	Web	E-mail	FTP
Percent	67.3%	23.3%	7.9%	1.2%	0.3%

Traditional P2P content distribution applications consider a loose notion for quality, i.e. a file will eventually be downloaded after some time. P2P-based IP-TV applications are even capable to support strict quality constraints for video playback. They are capable to relay sufficient video data to an end user peer such that the peer is able to play out continuously a moving image with sound. The popularity of P2P-based IP-TV was revealed in recent studies. Table 2 depicts observed and estimated traffic volumes of different IP-TV applications reported in (*Cisco Inc., 2008*). Again, P2P-based IP-TV has gained a significant market share in very short time. It can even compete with conventional Content Distribution Networks (CDNs) as used by YouTube (*Gill et al., 2007*).

In order to understand the success of P2P-based content distribution, we will investigate now briefly the highly popular eDonkey system (*Tutschku, 2004; Wearden, 2003*). eDonkey is a typical

Table 2. Amount of IP TV traffic

Traffic Type	Terabytes per month
YouTube – worldwide (Cisco est., May 2008)	100.000
P2P Video Streaming in China (Jan. 2008)	33.000
YouTube – United States (May 2008)	30.500
US Internet backbone at year end 2000	25.000
US Internet backbone at year end 1998	6.000

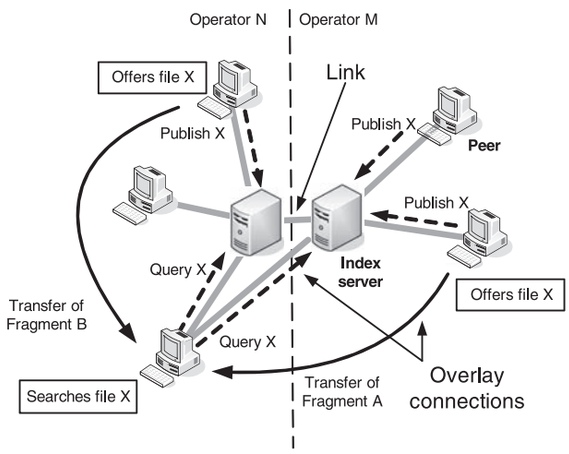

Fig. 2. Hybrid P2P content distribution application

representative for P2P content distribution applications. The eDonkey architecture is depicted in Fig. 2. eDonkey is denoted as a hybrid-P2P system since it consists of two kinds: a) end-user peers (for short denoted as peers) providing and downloading files, and b) index servers providing the information on the locations of a file or parts of it. When a peer wants to download a file, it queries the index servers and then asks the providing peers for data transmission. The data transmission can be accelerated by using the *multiple source download principle*. Here, two or more different pieces of a file are downloaded in parallel from different providing peers. Due to the availability of order information, the pieces can be reassembled appropriately. Since peers can both, downloading and providing information, the boundary between consumer and provider vanishes in P2P systems.

A closer look reveals that P2P content distribution systems form two different overlays. One overlay is dedicated to the distribution of query information, while the other one is dedicated for user data exchange, i.e. for transmitting video or audio information. It becomes also evident that the two overlays may have different topologies, addresses, and routing principles. In addition, a downloading peer remains in command where to download the data from. If numerous peers provide the same information, the downloading peer can choose the best peers to download from. This characteristic facilitates also the feature of P2P overlays to be more reliable than conventional client/server systems since they don't rely on a single source. Another feature of P2P systems is the use of their own addressing schemes. In this way, they are able to circumvent the problems of Internet hosts being behind NAT (Network Address Translation). Moreover, P2P overlays enable the integration of networks of different technologies and of different administrative domains into a single virtual structure. Thus, they facilitate the notion of *multi-network services* (*Tutschku, Tran-Gia, Andersen, 2008*).

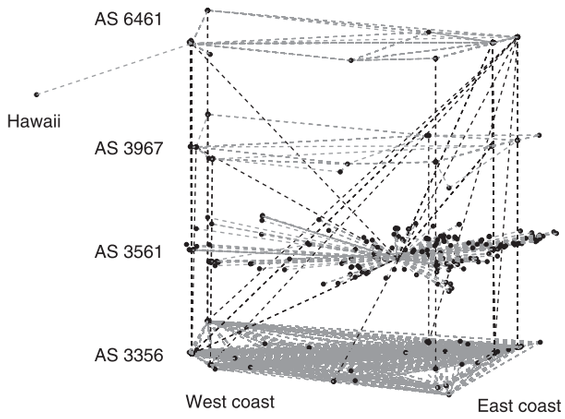

Fig. 3. Selected North-American Tier 1 provider networks

2.3.2 Diversity in connectivity and quality

Another achievement of today's Internet is its diversity in connectivity and quality. The Internet is not a homogenous network with a flat topology. Figure 3 depicts the topologies of four North-American Tier 1 network operators (AS3356, AS3561, AS3967, AS6461) on Point-of-Presence (POP) level (*Liljenstam, Liu, Nicol, 2003*). The figure reveals that a very large number of locations have many different routes to an arbitrary destination. These routes are often spread among different operators. Hence, a user would have heoretically the possibility to choose among the multiple providers and even within a provider among multiple routes. This characteristic would not only facilitate better performance but might also increase the competition among providers. A user can chose the most cost-efficient provider. Additionally, this picture shows that a significant redundancy is present in the networks. A better exploitation of this characteristic might enhance the reliability of the system.

The current Internet is not only diverse in its topology. Accompanying this feature is its *diversity in quality*. Theoretically, the current Internet protocols should find the ''shortest'' route to a destination. This feature means that theoretically the *triangle inequality (TI)* holds for the packet delay, cf. Fig. 4. However, recent measurements within PlanetLab have demonstrated that this inequality is violated more often than one has expected so far. The violation might be as high as 25% (*Banerjee, Griffin, Pias, 2004*).

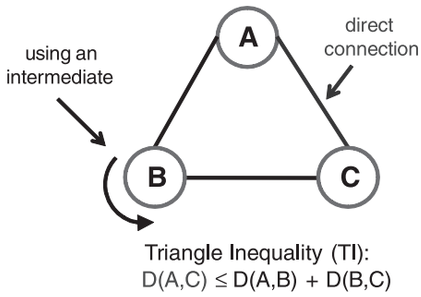

Fig. 4. Triangle inequality violation

This result shows a) that the current Internet routing is far from being optimal, b) better routes exist and sufficient capacity is often available in the networks and c) it can potentially be exploited and offered. Unfortunately, current IP transport protocols are not readily capable for multi-homing.

2.3.3 Operating system (OS) virtualization

The *virtualization* of operating systems has become very popular recently due to its capability to consolidate multiple virtual servers into a single physical machine (*Daley, Dennis, 1968; SCOPE Alliance, 2009*).

2.3.3.1 Virtualization techniques

In general, virtualization technology can be used to consolidate physical resources to reduce power consumption, maintenance and management costs. In addition, by loosening the binding of services to the physical resources providing those services, virtualization increases reconfiguration flexibility. The potential of virtualization technology to increase reliability, availability, and serviceability is thus attracting the attention of service providers as well as of system vendors. From the system point of view, virtualization is a technology that abstracts physical resources to generate logical resources. Two types of virtualization techniques can be distinguished: for *a)* sharing of resources and for *b)* aggregation of resources. While option a) is typically used when a single physical resource is considered for sharing, option b) provides for a logical resource, cf. Fig. 5.

When the *sharing* type of virtualization is required, the virtualization mechanism makes multiple virtual resources and provides them to the upper layer, cf. Fig. 5(a). A *virtual machine monitor (VMM)* (or

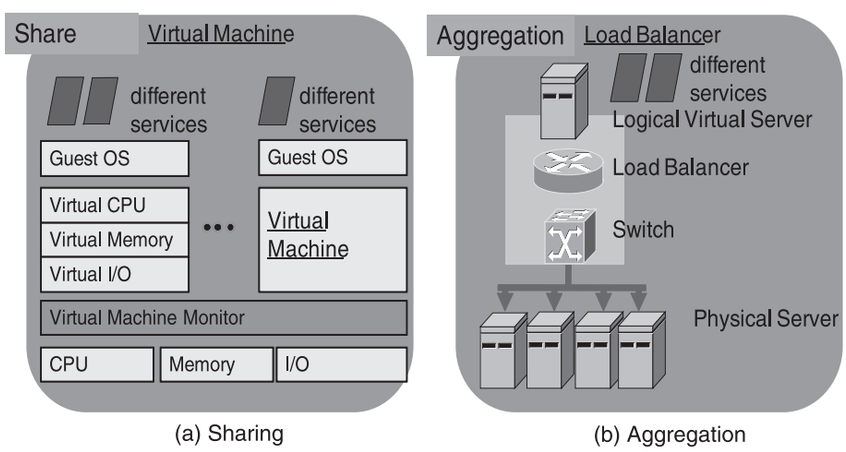

Fig. 5. Types of virtualization

hypervisor) is the typical example of a sharing mechanism in virtualization. A VMM controls the CPU scheduler to cut the CPU time into slices and provides them to the *virtual machines (VMs)* as virtual CPUs. That is, a single physical CPU resource is virtualized into multiple logical CPUs and shared by multiple VMs. Isolation and safe partitioning is a key to this type of virtualization at the hardware level.

The second option for virtualization is *aggregation*, cf. Fig. 5(b). This virtualization mechanism provides a single virtual resource out of multiple physical resources in the resource pool, often this approach is also denoted as *resource pooling* (*Wischik, Handley, Bagnulo Braun, 2008*). Load balancing is the typical example of aggregation-type of virtualization. A *scheduler* presents a single server to the network and then distributes the network accesses and their counter processing to multiple servers in a server pool. Seamless reconfiguration is a key to the aggregation-type of virtualization, which is usually provided by a software layer. A NV concept based on aggregation will be discussed in Sect. 4.2.2.

2.3.3.2 Advantages by virtualization

The application of OS virtualization reduces directly the operational expenditures (OPEX) of multiple servers. When implemented appropriately, virtualization permits fair and reliable *resource isolation* among virtual machines. In this way, virtualization allows a safe testing of server configurations without harming the other virtual machines. Furthermore, a personal machine configuration running on large servers is permitted for individual users. In this way, the users can use even a complete computer center as a PC which is located next to their desks.

Another advantage is that virtualization enables applications to be moved arbitrarily within the memory. This *memory invariance* can be exploited. Applications and systems can easily be moved and relocated to arbitrary physical locations.

In order to speed up the relocation of an instance of a virtual computer, efficient compression technologies for complete machine states such as SBUML (Scrap-Book User Mode Linux) (*Sato et al., 2003*) have been developed. SBUML can compress a state down to 10% of the real memory size. This compression ration enables a fast relocation of even large router operating system images within a network.

3. Network virtualization: solving the puzzle

The puzzle how Network Virtualization can overcome the shortfalls of today's Internet and paving the way for the Future Internet resolves when the outlined achievements are combined with the recent results on generic infrastructures, overlays and federation.

3.1 Generic infrastructure, overlays, and federation

NV enables the easy consolidation of multiple networks or overlays into a single physical system. Each virtual network can be utilized for different applications, e.g., with different QoS requirements. Hence, major concepts in NV are to provide a generic infrastructure, which supports multiple overlay networks in parallel, and to *federate* the resources provided by these infrastructures and overlays. In detail, resource federation permits the interconnection of independently-owned and autonomously administered NV infrastructures in a way that permits their owners to define resource usage and allocation policies for the infrastructure under their control, operators to manage their infrastructures, and researchers to create and populate overlays, allocate resources to them, and run, currently *experiment-specific*, software in them. The concept of federation is depicted in Fig. 6 (after (*Landweber, Falk, 2008*)). It depicts a researcher executing his experiment across an evolving federation, e.g. in the GENI project. Such federations of infrastructures can span

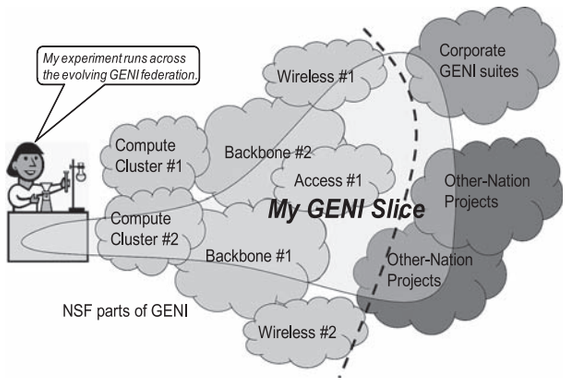

Fig. 6. Concept of federation, after (*Landweber, Falk, 2008*)

across multiple providers, technologies, and even other countries. In this example, the infrastructure is inherently planned for inter-domain management and operation. This feature gives raise to the hope that the concept of NV may be able to implement features, long hoped for in networks which are *inter-domain management* and in particular *inter-domain traffic management*.

How to achieve a federation feature is at the moment under heavy discussion (*The GENI Consortium, 2009*). The currently most promising approach is the one from Cluster B of the GENI project that is based on *components* (*Peterson et al., 2009*). For example, a component might be an edge computer, a customizable router, or a programmable access point. These components can be made available by their owners for the use in an overlay. A component encapsulates a collection of resources, including physical resources (e.g., CPU, memory, disk, bandwidth), logical resources (e.g., file descriptors, port numbers), and synthetic resources (e.g., packet forwarding fast paths). These resources can be contained in a single physical device or distributed across a set of devices. A given resource can belong to at most one component.

Each component is controlled via a *component manager* (CM), which exports well defined remotely accessible interfaces. The CM defines the operations available to user-level services to manage the allocation of component resources to different users and their applications. Typically, a component's CM runs on the component itself, although a remote proxy CM can also control components that are unable to host a CM. It must be possible to multiplex (this is often denoted a *sharing* mode in NV as defined above) component resources among multiple users. In NV, the notion of ''to slice'' is often used as synonym for ''to share''. This can be done by combining the virtualization of the components (where each user acquires a virtual copy of the component's resources), or by partitioning of the component into distinct resource sets (where each user acquires a physical partition of the component's resource). In both cases, it is said that the user is granted a *sliver* of the component. Each component must include hardware or software mechanisms that isolate slivers from each other, making it appropriate to view a sliver as a *resource container*.

The network interfaces on components are shared by VMs. The access to the interfaces is controlled by a VMM. The VMM has to accomplish here two key tasks. First, it must provide shared access to the network interface. This means that the virtual machines outgoing network traffic must be multiplexed together before being sent over the network; similarly, incoming network traffic must be demultiplexed before being delivered to the appropriate virtual machines. Second, the VMM must protect the virtual machines from each other. This means that no virtual machine must be allowed to transfer data into or out of another virtual machine's memory. Thus,

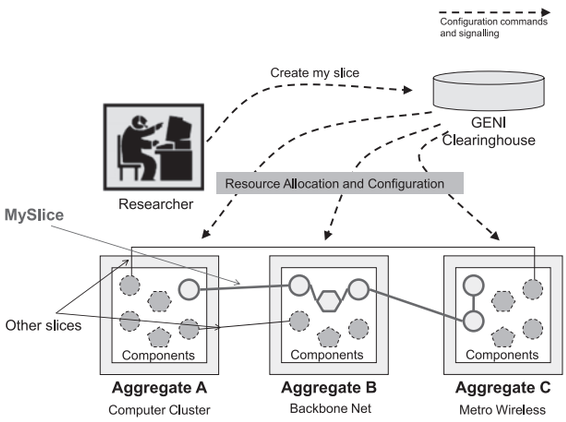

Fig. 7. Slices in GENI, after (*Landweber, Falk, 2008*)

NV achieves strong resource isolation among the virtual routers. Therefore, the challenge when implementing a generic NV node is to provide efficient, shared, and protected access to the network interface. A slice is defined by a set of slivers which spans a set of network components (plus an associated set of users), see also Fig. 7. From a user's perspective, a slice is a network of computing and communication resources capable of running an application or a wide-area network service. From an operator's perspective, slices are the primary abstraction for accounting and accountability (*Peterson et al., 2009*).

3.1.1 Virtual network and slice management

A slice is managed using three main management operations:

- *Register:* the slice exists in name and is bound to a set of users;
- *Instantiate:* the slice is configured on a set of components and resources assigned to it;
- *Activate:* the slice is booted, at which point it runs code on behalf of a user.

The detailed configuration of the components and physical resources can be achieved for the Cluster B project in the GENI system via the Raven provisioning service (*The Raven Consortium, 2009*). The Raven service is based on the Stork package management toolkit for PlanetLab (*Cappos et al., 2007*). Raven is able to provide what GENI experiment needs to run: software, runtime environments, and resources. It can be used for typical FCAPS tasks (*Subramanian, 1999*), in particular for slice management, configuration management, and monitoring and data collection.

A major feature of GENI is its capability to grow, revise and adapt slices during operation. For example, a successful, long running experiment can grow larger over time. Therefore, a researcher may issue configuration requests to the GENI clearinghouse asking for additional resources and components, allocating them, and binding them to the slice. The components may be leased from different infrastructures, i.e. aggregates, and even be of different technologies, cf. Fig. 8(a).

A *slice authority* implements security and policy management. It is associated with each slice and takes responsibility for the behavior of the slice. Every slice is registered only once, but the set of users bound to it can change over time. A slice registration has a finite lifetime; the responsible slice authority must refresh this registration periodically. The slice authority may ask for registration at a trusted authority, which is typically denoted as the clearing house, cf. Fig. 8(b). A clearinghouse is a mostly operational grouping of *a)* architectural elements including trust anchors for Management Authorities and Slice Authorities and *b)* for services including user, slice and component registries, a portal for resource discovery, a portal for managing policies, and services needed for operations and management. There can be multiple clearinghouses, which can *federate*. One application of federation is as the interface between clearinghouses. In this way, components from different administrative domains can be bound into a single slice. Thus, a slice can span over various providers, even international, semiprivate and commercial ones, with diverse technologies, cf. Fig. 8(b). As a result of these integration capabilities, the concept of federated slices is able to facilitate the *convergence* of diverse networks.

3.1.2 Benefit of network virtualization

The concept of a slice is an abstraction for networks and network applications. As a result, NV using slices permit distributed participants to create almost instantly their own network with application-specific naming, topology, routing, and resource management mechanisms such as server virtualization. Users can thus use even a whole computing center arbitrarily as their own personal computer. An immediate benefit of NV is the reduction of the required amount of hardware (capital expenditures, CAPEX) and of the operational expenditures (OPEX) of network structures, since less

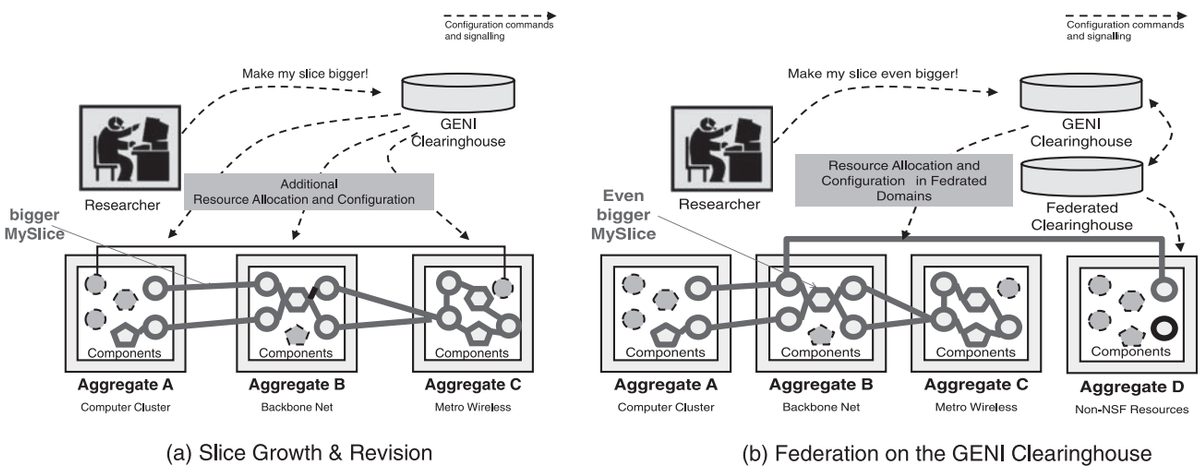

Fig. 8. Adaptation capabilities for virtual networks in the GENI architecture (*Landweber, Falk, 2008*)

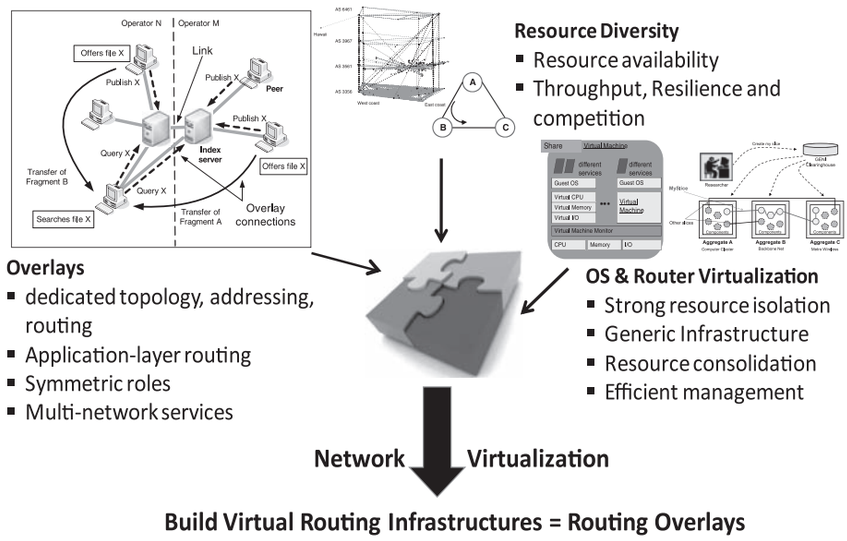

Fig. 9. Components of network virtualization

physical systems have to be configured. As a result, NV is considered currently as an operational technique. From the viewpoint of designing communication networks, the concept of NV has the potential to extend beyond operational issues. It may address some of the impasses of today's Internet such as the limited pluralism of the architectures (*Anderson et al., 2005*).

3.2 Building blockings for network virtualization

The concept of *virtual network structures*, such as P2P overlays, forms the first major building block for Network Virtualization, cf. Fig. 9. Due to their ability to form arbitrary application specific network structures, overlays can achieve higher performance and are more reliable than other network architectures. In addition, the specific ability of P2P overlays for symmetric roles prevents a look-in of users into a specific provider. The capability of overlays for bridging between various network architectures facilitates services across multiple technical and operational domains.

The second building block is *the diversity in connectivity and quality* in networks. The diversity of today's Internet will even be increased in the Future Internet due to new physical transport systems for core networks, such as 100 GB Ethernet, and more network providers. As a result, it can be assumed that high amounts of data transmission capacity will be available in the future. If one is able to locate these resources, they can be utilized for achieving high performance and reliability of the system.

Finally, *OS virtualization* and generic infrastructures constitute the third building block. It provides the opportunity to consolidate safely multiple networks in one physical platform. In addition, it may simplify the management of the system due to the reduction of physical entities.

3.3 Routing overlays: the tool for evolving today's and the future Internet

The combination of these building blocks provides the basis for Network Virtualization. The deficiencies of today's system and the foundation for the future Internet can be laid by defining a virtual routing infrastructure, also known as *routing overlays*. This infrastructure should a) enable its re-use on small scale, b) provide services invariant from the location of the service provider, and c) permit the use of application-layer mechanisms safely in lower layers of the stack.

4. Implementing advanced routing overlays

Recently, various architectures of routing overlays have been proposed (*Nakao, Peterson, Bavier, 2003; Gummadi et al., 2004*). A highly promising approach is the concept of *one-hop source routing*. Hereby, the user data is forwarded to a specific intermediate node which then relays the traffic to its destination using ordinary IP routing. The dedicated forwarding can be easily achieved by establishing a tunnel to the intermediate node. The advantage of one-hop source routing is the easy control of performance by selecting an appropriate intermediate node while still being scalable.

4.1 An efficient one-hop source routing architecture

An efficient one-hop source architecture capable of NV was suggested in (*Lane, Nakao, 2007; Khor, Nakao, 2008*). This architecture is depicted in Fig. 10. The architecture applies edge-based NV-boxes which can execute safely *virtual router software*. These software routers can accept incoming traffic from tunnel and forward this traffic to the destination using conventional IP routing protocols. When a source wants to send data with controlled performance, cf. Step 1 in Fig. 10, then it sends a signal to an

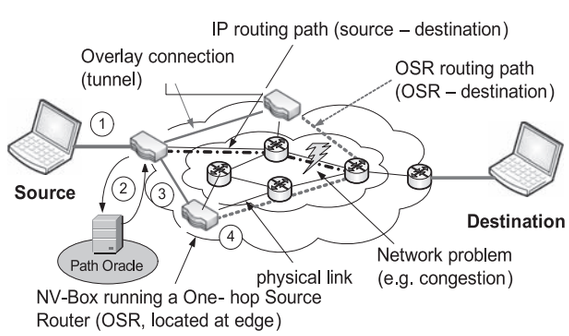

Fig. 10. Routing overlay using one-hop source routing

NV-box running the *One-hop Source Router (OSR)* software. When an OSR router receives such a signal it asks a *Path Oracle* to provide him with the address of an intermediate node which can forward this data in the required way, cf. Step 2 in Fig. 10. Subsequently, the ingress OSR router establishes a tunnel to the selected intermediate OSR router, cf. Step 3 in Fig. 10. Finally, the intermediate OSR router inserts the traffic into the conventional IP routing process, cf. Step 4 in Fig. 10.

This architecture shows a separation of the former monolithic IP system into two virtual overlays, one for signaling and one for data forwarding. This separation can be seen in parallel to the two overlays in P2P content distribution applications. The two overlays can be structured and equipped with routing mechanisms according to their specific function.

However, it also has to be mentioned here that edge-based nature of this architecture reduces the efficiency of one-hop source routing. As a consequence SOR systems should also be deployed in core networks.

Due to the use of NV-boxes and their placement at arbitrary locations, virtual SORS systems are also instantiated at arbitrary locations. Thus, virtual routing overlays can re-use the generic infrastructure available in the network.

4.2 Concurrent multipath transfer

The capability of the above introduced one-hop source architecture can be demonstrated readily by the problem of achieving very high throughput data transmissions. The solution to this problem is the combination of the multiple overlay paths into one large overall transport pipe by using concurrent multipath transfer.

4.2.1 Overall architecture

The considered CMP architecture sends data packets concurrently on different overlay paths from the source to the destination. This principle is also known as *striping* or *inverse-multiplexing*. The paths can be chosen from different overlays, which can span across different physical networks. Figure 11 shows such a case. It depicts two physical network (indicated by solid lines) and two overlays (indicated by dashed lines), which are embedded in these physical networks. The high capacity, overall transport pipe is combined from two paths from different overlays.

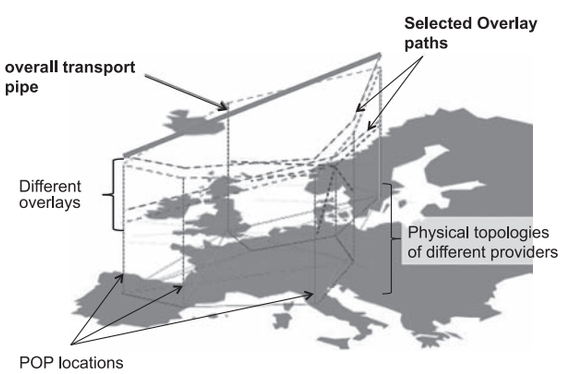

Fig. 11. Providing a high-capacity pipe by a combination of multiple overlay paths

The combination of different paths achieves a direct increase of throughput and a higher reliability since the system does not rely on a single path anymore. In addition, this architecture facilitates inter-domain traffic management and edge-based performance control due to the selection of appropriate intermediate nodes. In addition, the application of the path oracle can lead to rapid discovery of available resources in the network. Such a path oracle can be provided by the network operator or by other institutions (*Aggarwal, Feldmann, Scheideler, 2007*).

4.2.2 Transport virtualization

The idea of *transport virtualization* is motivated partly by the abstraction introduced in P2P content distribution networks (CDNs). When using the *multi-source download* mode, a peer downloads multiple parts of a file in parallel from different peers. As a result, the downloading peer doesn't rely any more on a single peer, which provides the data, and the reliability and the goodput is increased. Thus, multi-source download mode is a type of the aggregation mode of virtualization.

The above outlined abstraction of a storage resource is now transferred to the area of data transport. *Transport Virtualization (TV)* can be viewed as an abstraction concept for data transport resources. Hereby, the physical location of the transport resource doesn't matter as long as this resource is accessible. In TV an abstract data transport resource can be combined from one or more physical or overlay data transport resources. Such a resource can be, e.g., a leased line, a wave length path, an overlay link, or an IP forwarding capability to a certain destination. These resources can be used preclusively or concurrently and can be located in even different physical networks or administrative domains. Thus, an abstract transport resource exhibits again the feature of location independence.

4.2.3 Transmission mechanism

Figure 12 shows a more detailed model of the striping mechanism. The data stream is divided at the SOR router into segments which are splits into *k* smaller parts. These *k* parts are transmitted in parallel on *k* different overlay paths. The receiving SOR router reassembles these parts again into segments. The parts can arrive at the receiving router at different time instances since they are transmitted on paths with different delay distributions. Therefore, it is possible that they arrive ''out of order''. It should be mentioned here, that part reordering could only happen between different paths. The order of packets on a path is maintained since packets typically cannot overtake each other on a path.

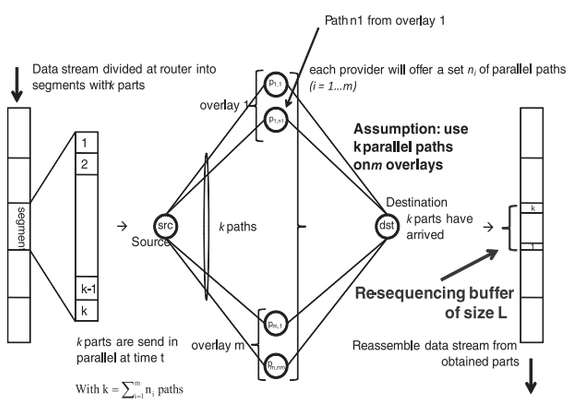

Fig. 12. CMP transmission mechanism

In order to avoid having this behavior impact on the application performance, the receiving SOR router maintains a finite re-sequencing buffer. However, when the re-sequencing buffer is filled and

the receiving router is still waiting for parts, part loss can occur. This loss of parts is again harmful for the application and should be minimized. This can be achieved by an appropriate selection of the re-sequencing buffer size. A performance investigation of the CMP transport mechanism and of the buffer occupancy is provided in (*Zinner et al., 2009*).

5. Conclusion

In this paper we have investigated the deficiencies and the achievements of today's Internet. We outlined why and how Network Virtualization (NV) can overcome the shortfalls of the current system and it paves the way for the Future Internet. NV is the technology that allows the simultaneous operation of multiple logical networks (also known as overlays) on a single physical platform. Furthermore, we argued why a new concept for convergence is needed in future networks and how NV and the idea of federation may achieve this aim.

In addition, we introduced a more elaborate concept for network virtualization, which is denoted as Transport Virtualization (TV). TV transfers the concept of location independence of resources to the area of data transport in communication networks.

The major building blocks of NV are the a) use of application-specific routing overlays, b) the safe consolidation of resources by OS virtualization on a generic infrastructure, and c) the exploitation of the network's diversity for performance enhancements as well as for new business models such as the provisioning of intermediates nodes or path oracles.

Future investigations will be focused on two areas: a) how manage and operate a generic infrastructure for virtualized networks and b) how to combine resources for TV in order to locations independence for.

Acknowledgements

The author would like to thank T. Zinner, A. Nakao, and P. Tran-Gia for stimulating discussions and D. Klein for the support in the preparation of this manuscript.

References

Aggarwal, V., Feldmann, A., Scheideler, C. (2007): Can isps and p2p systems co-operate for improved performance? ACM SIGCOMM Computer Comm. Review (CCR), vol. 37, no. 3, Jul. 2007.

Anderson, T., Peterson, L., Shenker, S., Turner, J. (2005): Overcoming the internet impasse through virtualization. IEEE Computer, Apr. 2005.

Banerjee, S., Griffin, T., Pias, M. (2004): The interdomain connectivity of planetlab nodes. In: Proc. of the 5th Passive and Active Measurement Workshop (PAM2004), Antibes Juan-les-Pins, France, Apr. 2004.

Cappos, J., Baker, S., Plichta, J., Nyugen, D., Hardies, J., Borgard, M., Johnston, J., Hartman, J. H. (2007): Stork: Package management for distributed VM environments. In: Proc. of the 21st Large Installation System Administration Conf. (LISA '08).

Cisco Inc. (2008): Approaching the Zettabyte Era. Information available at http://www.cisco.com/.

Clark, D., Partridge, C., Ramming, J., Wroclawski, J. (2003): A Knowledge Plane for the Internet. In: Proc. of the ACM Sigcomm 2003 Conf., Karlsruhe, Germany, Aug. 2003.

Clip2 (2001): The Gnutella Protocol Specification v0.4 (Document Revision 1.2). Information available at http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf/.

CNet News (1999): Net number system at a crossroads. Information available at http://news.cnet.com/Net-number-system-at-a-crossroads/2009-1023_3-225712.html/.

Daley, R., Dennis, J. (1968): Virtual Memory, Processes, and Sharing in MULTICS. Communications of the ACM, vol. 11, no. 5, May 1968.

Gabeiras, J. (2004): P2P-Traffic. Presentation at the COST 279 Midterm Seminar; Rome, Italy, Jan. 2004.

GENI Consortium (2006): GENI – Global Environment for Network Innovations. Information available at http://www.geni.net/.

GENI Planning Group (2006): GENI Design Principles. IEEE Computer, 39 (9), Sep. 2006.

Gill, P., Arlittz, M., Li, Z., Mahantix, A. (2007): YouTube Traffic Characterization: A View From the Edge. In: Proc. of the 7th ACM SIGCOMM conf. on Internet measurement (IMC 07), San Diego, CA., Oct. 2007.

Gummadi, K., Madhyastha, H., Gribble, S., Levy, H., Wetherall, D. (2004): Improving the reliability of internet paths with one-hop source routing. In: Proc. of 6th conf. and symposium on Opearting Systems Design & Implementation (OSDI'04), San Francisco, Ca., USA, Dec. 2004.

Handley, M. (2006): Why the Internet only just works. BT Technology J., vol. 24, no. 3, Jul. 2006.

Khor, S., Nakao, A. (2008): AI-RON-E: Prophecy of One-hop Source Routers: In: Proc. of the 2008 IEEE Global Telecomm. Conf. (Globecom08), New Orleans, LA., Nov./Dec. 2008.

Landweber, L., Falk, A. (2008): A GENI Use-Case. Invited Talk at Future Internet Assembly, Madrid, Spain. Available at http://www.ict-fireworks.eu/fileadmin/events/FIA-Madrid/Larry_Landweber.pdf/.

Lane, J., Nakao, A. (2007): Sora: A shared overlay routing architecture. In: Proc. of the 2nd Int. Workshop on Real Overlays And Distributed Systems (ROADS), Warsaw, Poland., Jul. 2007.

Martin, R., Menth, M. (2006): Backup Capacity Requirements for MPLS Fast Reroute. In: 7. ITG Fachtagung Photonische Netze (VDE), Leipzig, Germany, Apr. 2006.

MySpace Inc. (2006): MySpace.com. Information available at http://www.myspace.com/.

Nakao, A., Peterson, L., Bavier, A. (2003): A Routing Underlay for Overlay Networks. In: Proc. of the ACM Sigcomm 2003 Conf., Karlsruhe, Germany, Aug. 2003.

NICT (2007): AKARI Architecture Design Project for New Generation Network. Information available at http://akari-project.nict.go.jp/eng/index2.htm/.

Peterson, L., Sevinc, S., Lepreau, J., Ricci, R., Wroclawski, J., Faber, T., Schwab, S., Baker, S. (2009): Slice-based facility architecture. Technical Report of cluster B of the GENI project. Available at http://svn.planet-lab.org/attachment/wiki/GeniWrapper/sfa.pdf/.

Roscoe, T. (2005): Peer-to-Peer Systems and Applications. Berlin: Springer. 2005, ch. 33. The PlanetLab Platform.

Sato, O., Potter, R., Yamamoto, M., Hagiya, M. (2003): UML Scrapbook and Realization of Snapshot Programming Environment. In: Proc. of the Second Mext-NSF-JSPS Int. Symp. on Software Security (ISSS 2003), Tokyo, Japan., 2003.

SCOPE Alliance (2009): I/O Virtualization: A NEP Perspective. Technical Report available at http://www.scope-alliance.org/.

Skype Limited (2006): Skype Extras. Information available at http://extra.skype.com/.

Skype Technologies S.A. (2003): Skype Homepage. Information available at http://www.skype.com/.

Subramanian, M. (1999): Network Management: Principles and Practice. Addison-Wesley.

The GENI Consortium (2009): 4th geni engineering conf. (gec4). Information available at http://www.geni.net/.

The Raven Consortium (2009): The raven provisioning service. Information available at http://raven.cs.arizona.edu/.

Tran-Gia, P. (2008): G-Lab: A Future Generation Internet Research Platform. Information available at http://www.future-internet.eu/.

Tutschku, K. (2004): A measurement-based traffic profile of the eDonkey filesharing service. In: Proc. of the 5th Passive and Active Measurement Workshop (PAM2004), Antibes Juan-les-Pins, France, Apr. 2004.

Tutschku, K., Nakao, A. (2009): Towards the engineering of dependable p2p-based network control – the case of timely routing control messages. IEICE Transactions on Communications, vol. E92-B, no. 1, 2009.

Tutschku, K., Tran-Gia, P., Andersen, F.-U. (2008): Trends in network and service operation for the emerging future internet. Int. J. of Electronics and Communication.

Varoutas, D., Katsianis, D., Sphicopoulos, T., Cerboni, A., Canu, S., Kalhagen, K., Stordahl, K., Harno, J., Welling, L. (2002): Economic viability of 3 g mobile virtual network operators. In: Proc. of 3G Wireless 2002, San Francisco, USA, May 2002.

Wearden, G. (2003): eDonkey pulls ahead in europe p2p race. http://business2-cnet.com.com/2100-1025_3-5091230.html/.

Wischik, D., Handley, M., Bagnulo Braun, M. (2008): The resource pooling principle,'' ACM SIGCOMM Computer Comm. Review (CCR), vol. 38, no. 5: 47–52.

YouTube Inc. (2006): YouTube Fact Sheet. Information available at http://www.youtube.com/t/fact_sheet/.

Zinner, T., Tutschku, K., Nakao, A., Tran-Gia, P. (2009): Performance Evaluation of Packet Re-ordering on Concurrent Multipath Transmissions for Transport Virtualization. In: Proc. of the 20th ITC Specialist Seminar on Network Virtualization, Hoi An, Vietnam, May. 2009.

Author

Kurt Tutschku
holds the Chair of ''Future Communication'' (endowed by Telekom Austria) at the University of Vienna. Before that, he was an Assistant Professor at the Department of Distributed Systems, University of Wuerzburg, Germany. He led the department's group on Future Network Architectures and Network Management until December 2007. From Feb. 2008 to July 2008, he worked as an Expert Researcher at the NICT (National Institute for Information and Communication Technology, Japan).

Kurt Tutschku received a diploma and doctoral degree in Computer Science from University of Wuerzburg in 1994 and 1999, respectively, and completed his Habilitation (State Doctoral Degree) at the University of Wuerzburg in 2008. His main research interest include future generation communication networks, Quality-of-Experience, and the modeling and performance evaluation of future network control mechanisms and services in the emerging Future Internet, particular of P2P overlay networks.

Prof. Dr. Kurt Tutschku is leading and accomplished multiple industry cooperations in the field of Future Internet, P2P and Network Management with Nokia Siemens Networks, BTexact, DATEV e.G., Bosch and Bertelsmann AG. He currently leads the work package on ''Overlays for Network Control and Support of Evolved Services Infrastructures'' of the European FP7 framework project ''EuroNF''. This work package includes Network Virtualization. He has also received grants from the DFG (Deutsche Forschungsgemeinschaft) for the investigation of mobile P2P architectures.

Kurt Tutschku is an evaluator for the European Commission in the field of Information and Communication Technology (ICT) and is author of nine patent applications and around 60 publications presented in books or refereed international conferences or journals.