A Survey of the Research on Future Internet and Network Architectures

Toru HASEGAWA(✉), Senior Member

SUMMARY The Internet was designed for academic use more than 40 years ago. After having been used commercially, many unpredictable requirements have emerged, including mobility, security and content distribution. In addition, the Internet has become so ossified that fulfilling new requirements is difficult. Instead of developing ad-hoc solutions, redesigning clean-slate Internet architectures has become a key research challenge in networking communities. This survey paper addresses key research issues and then introduces ongoing research projects from Japan, the United States and the European Union.

key words: New Generation Network, future Internet, clean-slate design

1. Introduction

The Internet, originating from Advanced Research Projects Agency Network (ARPANET), was born about 40 years ago. The Network Control Program of ARPANET was replaced with TCP/IP protocol suites in 1983 and then the Internet, which was originally used as an academic network, was commercialized in 1988. The Internet has been so successful that a number of applications including Web, Voice over IP (VoIP), and streaming have been deployed over it.

Despite the success, the current Internet is so ossified as not to be able to keep up with current and future requirements. The ossification of the Internet is criticized from many aspects. First, the end-to-end principle, which is one of the most important design principles of the current Internet, might not hold [1] because the increasing processing speed and memory space of routers' hardware would enable nodes in the Internet to do more than just forwarding packets. Second, many new network functions proposed to fulfill requirements from new applications are not widely (commercially) deployed over the Internet because they require changes in routers and host software. Such network function examples include mobility and multicast. Third, since security was not originally built into the Internet, experimental testbeds and network virtualization as their base technology have been studied across the broad areas of network architectures, principles, mechanism designs and so forth.

The research topics include principles of rethinking the narrow waist, new architectures including the Content Centric Networking (CCN) and service centric architectures, integration of mobility/security/management functions that were not originally built into the Internet, experimental testbeds and network virtualization as their base technology and network science foundation for future Internet/network architectures.

After the initial phase when each research topic was undertaken independently, large-scale research projects that focus on total architectures have come to be funded. The Future Internet Architecture (FIA) program [6] of NSF is such a funding program and the four large projects including NDN [7], MobilityFirst [8], NEBULA [9] and XIA [10]–[12] are on-going. In Japan, the multi-institutional project AKARI [13] is developing an initial specification [14] for achieving the New Generation Network. In the European Union, many projects including 4WARD [15] and SAIL [16] are on-going.

As researches on future network architectures have been progressing, international workshops focusing on them have been held. ReArch, VISA, FutureNet, NOMEN and ICN workshops are such examples. Articles on future Internet architectures [17]–[19] have been published by magazines. Among them, [18] provides a series of survey articles and one of the articles [19] surveys the above research projects.

In Japan, IEICE has published the three special sections related to the New Generation Network [20]–[22].

This paper surveys research challenges in terms of ar-

Copyright © 2013 The Institute of Electronics, Information and Communication Engineers
architectures and protocols for the future Internet/ network architectures. (The words Internet and network are interchangeably used in this paper.) The rest of paper is organized as follows. Section 2 discusses the host and information-centric models as the narrow waist of the future Internet. In Sect. 3, the IP layer which is the core of internetworking is re-thought. Section 4 discusses the Information Centric Networking (ICN) which proposes the narrow waist of naming layer. Section 5 discusses mobility issues of the future Internet focusing on MobilityFirst project [8]. Section 6 discuss security issues of the future networks. Sections 7 and 8 discuss network management functions and service centric architectures focusing on how flexible new services are developed. In Sect. 9, network virtualization and its application to experimental testbeds are discussed. In Sect. 10, other issues including cloud networking architectures, network sciences and substrate technologies are briefly summarized.

2. Narrow Waist

The current Internet protocol stack has a layered architecture and resembles an hourglass. Its center is a universal network layer, i.e., the Internet Protocol (IP) layer, which implements the minimal functionality necessary for global interconnectivity. Some studies consider that the ossification of the Internet architecture, which does not support new applications well, originates from the current narrow waist of the IP layer [1], [23], [24], [25] addresses the IP layer’s limitations in terms of flexibility in forwarding, data distribution and so forth. Having to introduce caches and proxies in the middle of the Internet is a limitation example. It proposes using HTTP because HTTP is a de-facto protocol for emerging content delivery services.

The narrow waist issue is related to drawbacks coming from location dependent IP addresses. The location dependency causes the IP layer to fail to build in mobility, multi-homing, multicast and security. There are two approaches that address this issue. The first approach is the locator/identification separation scheme [26]–[33] which solves such drawbacks in the host-centric model which the current Internet adopts. The basic idea behind the separation is as follows: An IP address in the current Internet is used as a session identifier in TCP, as well as a locator for routing. This coupling leads to problems. It breaches the independence between the layers. Mobility and multi-homing are not well supported because an IP address is bound to a physical location.

The second approach is the information-centric model known as ICN [34], [35], CCN [36] and the Named Data Networking (NDN) [7]. (We call architectures based on the information-centric model ICN hereafter.) ICN proposes a different model that replaces where with what. In this model, named contents such as Web pages and videos are transferred to hosts requesting them. Content names do not have location information that would be required for end-to-end packet delivery. ICN takes a more general approach and regards the naming layer as the narrow waist.

The narrow waist of host and information-centric models correspond to the IP layer and the naming layer, respectively. Figure 1 shows how names, identifiers and locators are handled in the two models. In the current IP layer, a human-readable host name is resolved to an IP address as a locator by Domain Name System (DNS) and the IP layer delivers packets to the locator by the routing service (a-1).

In locator/identifier separation, a human-readable host name is resolved to a location independent identifier by a DNS-like name resolution system. Then the identifier is resolved to the locator and the packet is routed based on the locator (a-2). In some cases, a packet is directly routed based on the identifier itself. In this case, the identifier is not resolved to the locator (a-3).

On the contrary, there are two approaches to forwarding packets in the information-centric model. In the first approach, a packet is directly routed by the content name (b-1). (This is called name-based routing.) In the second approach, the content name is resolved to a locator and the locator is used by routing the packet.

Developing future networks based on the above approaches raises various issues such as scalability, mobility and security. Sections 3 and 4 discuss scalability issues in the both models, respectively. Sections 5 and 6 discuss mobility and security issues, respectively.

3. Internetworking Architectures

3.1 Overview

This section addresses two challenges in the host-centric model. The first challenge is overcoming the location dependency of IP addresses. The second challenge is achieving scalability and source routing that have not been well provided.

3.2 Locator/Identifier Separation

Table 1 summarizes locator/identifier separation network ar-
The architectures are classified by how to assign an identifier to a host. A router-based scheme assigns an identifier that specifies a network endpoint through which a host is connected to the network [26], [27]. Identifiers can be aggregated to achieve scalability in routing; however, this causes this scheme to fail to naturally support mobility [26].

On the contrary, host-based scheme [28]–[33] assigns a flat identifier that identifies the host. A flat identifier is usually a self-certifying address [37] and thus provides accountability. Accountability means that who (or which host) sent this packet is validated by unforgeable ways. There are two approaches for forwarding packets as shown in Fig. 1 of Sect. 1: (a-2) name resolution and (a-3) name-based routing. Routing based on flat identifiers is discussed in Sect. 3.3.2.

Table 1 also summarizes how mobility, multi-homing and security are supported by the individual architectures. In the rest of the section, individual architectures are described.

Since mobility and multi-homing are not built into the Internet, ad hoc solutions have been developed such as Mobile IP [38] and Shim6 [39]. These solutions introduce shim layers between the IP and TCP layers instead of recreating the IP layer. Locator/identification separation proposes a more elegant solution by re-thinking the role of IP address.

The Host Identity Protocol (HIP) [28], [29] is an incremental solution and introduces HIP sublayer within the TCP/IP stack as illustrated in Fig. 2. HIP introduces a new namespace of identifiers called Host Identity (HI). HI is a 128-bit long public key based value called Host ID Tag (HIT). HITs are translated into IP addresses in kernels of the host. HIP discovers and authenticates bindings between public keys and IP addresses. In order to support mobility, rendezvous servers that map HITs to IP addresses are prepared. Before sending a packet, a host sends a request asking the current IP address of the mobile host to the rendezvous server. Thus HIP is based on an end-to-end approach that does not require any change in routers.

Some proposals follow a core-edge separation approach [26], [27] wherein the separation is achieved in routers instead of hosts. Despite Locator/Identifier Separation Protocol (LISP) [26] not introducing any changes to the stack of hosts, it does not support mobility due to their router-based identifiers.

Inspired by these anticipatory studies, many future Internet/network architectures have adopted locator/identifier separation using flat identifiers. There are two issues for implementing it. One issue is scalability in routing. Some studies on name resolution [28]–[31] address how an identifier is mapped to a locator in a scalable manner. Since an identifier is defined in a flat name space, such flat name spaces make it difficult to establish a hierarchy in name spaces. On the contrary, [32] proposes name-based routing without using name resolution. This issue is discussed in Sect. 3.3 and it is revisited in Sect. 5 since it is related to mobility support. Another issue is how the accountability of packets is achieved using public key based addresses as identifiers. The security issue is revisited in Sect. 6.

3.3 Routing

Scalable inter-domain routing is still a challenge for the current Internet as well as locator/identifier separation schemes. First, in the workshop held in fall 2006 by the Internet Architecture Board (IAB), scalability and stability of routing in the Internet were discussed [40]. The Border Gateway Protocol (BGP) [41], the de-facto inter-domain routing protocol, suffers from many problems such as route oscillations, instabilities [42] and slow convergence [43]. In addition, BGP does not provide multi-path and user controlled routing. Thus there are two series of researches addressing scalability [44], [45] and multi-path/user control source routing [46]–[48].

Second, although these studies focus on location dependent IP addresses, scalability is an important research issue for routing based on flat identifiers in locator/identifier separation schemes [32], [33]. The proposals for such routing issues are summarized in Table 2.

3.3.1 Routing on Location Dependent IP Addresses

Hybrid Link State Path-Vector (HLP) [44] is a step forward to re-think BGP. HLP assumes that routing events are prop-
agated as Autonomous System (AS) path vectors to all ASes in BGP and this leads to poor scalability. Leveraging peering, customer provider relationships between ASes, HLP proposes a hierarchical routing structure as shown in Fig. 3.

It is a hybrid link-state and path-vector protocol. The experiments show that HLP reduced the churn-rate of route updates by a factor 400 compared to that of BGP. (The churn rate is the rate of routing announcements received by a given router.) On the contrary, Routing Control Platform (RCP) [45] focuses on iBGP scalability, i.e., full-mesh communication among iBGP routers, and proposes a centralized routing control. The main idea is separating the routing state from the routers and it introduces robustness, scalability speed and so forth.

Another important issue is tussles in the Internet [49]. Since the Internet has become an infrastructure of the society, different stakeholders participate in it. In some cases, stakeholders have different interests from each other and such a process is called a tussle. A typical example is a tussle between users’ requirements to control the end-to-end path and operators’ policies to optimize their networks. Some studies [46]–[48] propose a source-controlled multipath routing.

Pathlet routing [46] proposes a source-controlled multipath routing. Networks, i.e. Autonomous Systems (ASes), advertise pathlets that are fragments of end-to-end paths along which they are willing to route. A sender concatenates such pathlets into a full end-to-end source route. A pathlet is contained in each packet and routers forward the packet based on the pathlet and thus source-controlled multipath routing is achieved. New Inter-domain Routing Architecture (NIRA) [47] offers multiple paths, too, but it allows only valley-free [50] paths and has some assumptions about the network topology.

A project in NeTS FIND proposes a “PostModern Internet Architecture” [48] which addresses a tussle between users and operators at the IP layer. The key idea of this architecture is to separate the policy plane from the data plane. The IP layer is redesigned so that diverse functions can support diverse policies. This architecture adopts source-controlled routes as pathlet and NIRA do. Recently, the new project ChoiceNet [51] has been funded by the FIA program and it focuses on choices more generally than user controlled routing.

3.3.2 Routing on Flat Identifiers

The flat identifier raises a scalability issue because it causes routing protocols to fail to use the hierarchy of location dependent addresses which is used for achieving scalability in routing by current routing protocols such as BGP and OSPF.

The scalability has been addressed in both name resolution and name-based routing. First, many studies use central servers that map an identifier to a locator [26]–[30] such as LISP and HIP. HIMALIS uses a name resolution mechanism slightly different from the rendezvous servers of HIP. Its name registries are organized in two groups to store static and dynamic mapping information separately so that it can facilitate faster name resolution as well as update of the dynamic mapping records in the mapping registries.

Some studies [31] apply Distributed Hash Table (DHT) [52] to achieve distributed resolution. SEATTLE [53], which does not focus on location/identifier separation, provides name resolution from an IP address to a MAC address using DTH. DMap [31] provides a fast global dynamic name resolution service based on DTH focusing mobility. This issue is revisited in Sect. 5.3.

On the contrary, Routing on Flat Labels (ROFL) [32] applies DHT to a physical network rather than to an overlay network and routes a packet directly using a flat identifier. It assigns identifiers to all hosts and routers and wraps these identifiers to create a circular name space as in Chord [52]. However, such a DHT-based name routing suffers from the problem that the stretch, which is defined as the ratio of route length to the shortest path length, tends to become high. Achieving shorter stretches in the flat name space is still an open question. Some studies [33] address this issue by applying compact routing [54].

4. Information Centric Networking

4.1 Overview

ICN architectures adopt a different model from the current host-centric model. It addresses location dependent addresses and content distribution dominant in future networks. Table 3 summarizes the principal architectures such as Data-Oriented Network Architecture (DONA) [55], CCN [36], NetInf [15], [34] and PSIRP/PURSUIT [56]–[58] from the design choices described in Sect. 4.3. After describing background and CCN as an ICN architecture example in Sects. 4.1 and 4.2, respectively, Sect. 4.3 describes the architectures in detail from the design choices.

4.2 Content Delivery

Although the primary usage of the Internet was host-to-host communication in the past, the Internet has come to be dominated by content distribution and retrieval. As incremental approaches, Content Delivery Networks (CDNs) and Peer-to-Peer (P2P) networks have become deployed widely. In
Table 3: Comparison of architectures.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name Space</td>
<td>Flat with structure</td>
<td>Hierarchical</td>
<td>Flat with structure</td>
<td>Flat with structure</td>
</tr>
<tr>
<td>Name-data Integrity</td>
<td>Signature</td>
<td>Signature</td>
<td>Signature or content hash,</td>
<td>Self-certifying</td>
</tr>
<tr>
<td>Name</td>
<td>Self-certifying</td>
<td>Human-readable</td>
<td>Self-certifying</td>
<td>Self-certifying</td>
</tr>
<tr>
<td>Trusted Third Party</td>
<td>-</td>
<td>Yes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Routing</td>
<td>Name-based Routing</td>
<td>Name-based Routing</td>
<td>Hybrid (Name-based and Name resolution)</td>
<td>Name-based Routing</td>
</tr>
</tbody>
</table>

a CDN network such as AKAMAI [59], contents such as images and videos are cached at thousands of edge servers around the world to improve performance. In a P2P network [60], peers (or hosts) share contents among themselves to mitigate the load on servers. Despite their success, they do not solve all problems due to their overlay approach. Overlay networks have no controls of physical networks and thus inefficient usage network resources are quite often in many overlay networks.

ICN is proposed to natively support efficient content distribution. But it is more than that and it becomes a common concept of several future Internet architectures.

4.3 Content Centric Networking (CCN)

In this section, CCN [36] is explained as reference architecture. CCN is a proposal to switch from host-oriented to content-oriented networking to meet data-intensive application needs. This means that although in the current Internet, accessing content and services requires mapping from what users care about to the network, future network architectures take content as a primitive to decouple location from identity, security and access, and retrieve content by name.

There are pioneering architectural studies including i3 [61], TRIAD and DONA [55]. DONA proposes to replace DNS names with flat, self-certifying names [37] and a name based anycast primitive.

Inspired by the architectures, CCN [36] proposes that the narrow waist is changed from the IP layer to content or data distribution, i.e., the name layer, as shown in Fig. 4. (CCN is the core concept of Named Data Networking (NDN) project [7] in the NeTS FIA project.) CCN adheres to the following principles.

1. It adopts an hour glass architecture where the narrow waist is content or naming layer as shown in Fig. 4.
2. It builds security into the architecture despite the fact that the current Internet does not.
3. It retains the end-to-end principle of the current Internet and expands this principle.
4. It adopts routing and forwarding plane separation as many future Internet studies do.

The main difference from the host-to-host communication of the current Internet is that the name of the content (data) is used to specify the communicating entity. Instead in the current Internet, the host that serves the content is specified. (Such host identifiers of the current Internet and HIP are the IP address and the Host ID Tag, respectively.) Thus communication in CCN is receiver-oriented as follows: To receive data, a receiver sends out an Interest packet, which carries a name that identifies the desired data. A router remembers the interface from which the request comes in, and then forwards the Interest packet by looking up the name in its Forwarding Information Base (FIB). Once the Interest reaches a node that has the requested data, a Data packet is sent back. CCN uses routing protocols similar to the current Internet, i.e., BGP and OSPF. Content names are aggregated to prefixes and the prefixes are propagated to FIBs of CCN nodes by BGP and OSPF-like routing protocols.

As shown in Fig. 5, a CCN node (router) consists of Faces (Interfaces), FIB which is a name-based routing table, Pending Interest Table (PIT) which records tentatively Interest packets and Content store which caches contents. This name and cache-based forwarding naturally supports multicast, mobility, multi-homing which are not inherently supported by the current Internet.

4.4 Design Choices

There are several important design choices of ICN.

1. Naming Scheme

ICN gives contents (objects in ICN) unique names that are independent of locations and hosts. An important role is
to establish a verifiable binding between the content and its name so that a receiver can authenticate the content. The authenticity is beneficial to prevent Denial of Service (DoS) attacks.

Names play a crucial role in routing and thus names may reflect a hierarchical structure or include embedded location information in order to assist routing. Two naming schemes are proposed: the hierarchical name space [36] and the flat name space [34], [55]–[58] with the self-certifying name.

The hierarchical name space has a similar structure to current Uniform Resource Locators (URLs). A packet of CCN consist of the hierarchical name like a URL, the content and the digital signature information. It provides content-based security [62], which means that CCN authenticates the binding names and content by letting all contents have digital signatures based on public key cryptography. COPSS [63] extends the CCN naming scheme to enable efficient large-scale dissemination of contents.

The hierarchical name space has the following advantages over the (flat) self-certifying name. First, such names are human-readable. Second, they provide better routing scalability through name-prefix aggregation. On the contrary, one drawback of adopting human-readable names is that the architectures should have a trusted third party like a PKI infrastructure.

The self-certifying name allows direct verification of the binding between the name and the content. [34], [55]–[58] use the self-certifying name in the flat name space as the name of the information (content, data). For example, the name of DONA has the form \( P : L \) wherein \( P \) is the globally unique identifier that contains the cryptographic hash of the information owner’s (publisher’s) public key and \( L \) is the unique label of the information. Self-certification is achieved by binding the hash of content and its name (identifier) and its benefit is that there is no need to have either a public key infrastructure or other trusted third parties.

Which one between the two approaches is better remains controversial [64]. [62] advocates the hierarchical name space claiming that flat/opaque names in the flat name space requires an indirection architecture similar to today’s DNS and that the location independent flat name makes it difficult to construct an efficient mechanism to retrieve a nearby copy of content. [64] compares the two name spaces and concludes that there would be no instinct differences between them.

(2) Routing

Achieving scalability is an important issue in both name-based routing and name resolution. In name-based routing, nodes holding copies of named content are found to route request packets, i.e., Interest packets of CCN. In name resolution, a path from a requesting node to such nodes holding copies is found.

First, name-based routing is used both in a hierarchical name space like CCN and in a flat name space. In CCN, a requester asks for a content by sending Interest packets. Interest packets are forwarded to upstream (CCN) nodes that hold a copy or an original of the content as shown in Fig. 5. Names of contents can be aggregated as prefixes in a FIB.

However, the number of name prefixes would be far larger than that in the current BGP routing tables because the number of contents would be far larger than that of hosts. This raises a scalability issue. Besides [65] shows that FIB lookup based on the longest-prefix matching of the name-based routing is the most critical bottleneck of CCN routers. This is because CCN routers should look up content names in a huge FIB by longest prefix matching, which requires seeking the longest matching prefix through all candidate prefix lengths. [65] studies the feasibility of the Bloom filter based FIB for CCN routers and [66] proposes distributing a FIB to many line cards. Designing high performance CCN routers is still an open issue.

Second, name resolution used in a flat name space is similar to that of locator/identifier separation described in Sect. 3.3.2. In the flat name space, a content name need be mapped to a locator of the original content or locators of its copies. A name resolution service which stores binding from content (object) names to locations pointing to storages containing copies or originals of contents is needed for obtaining the location (i.e., the IP address) of the self-certifying flat name. How scalability is achieved in name resolution is a challenge because hierarchy of names cannot be utilized. PRISP [56] uses a name resolution server called a rendezvous point. It adopts source routing using a Bloom filter. NetInf adopts a hybrid approach of name-based routing and name resolution. It proposes a distributed name resolution system called Multi-level Distributed Hash Table (MDHT) [67] which provides a nested, hierarchical DHT for scalable distributed name resolution and efficient anycast routing.

(3) In-Network Cache

Nodes in ICN naturally support in-network storage to cache contents. Content is cached at nodes on a path from a requesting host to nodes holding the content or its copies. On the contrary, nodes in ICN can cache contents on nodes that are not on such paths by announcing cached contents in a name-based routing protocol or a name resolutions service [15].

Although caching is an old issue, its reality in actual environments is a hot topic [68], [69] in ICN research. These studies show that the current router implementation technology enables replacement of cached chunks at line speed. Since the caching capacity is smaller than the amount of forwarded contents and central management of cache placement is not feasible, caching techniques in a decentralized manner has been drawing attention. The breadcrumbs [70], [71] propose an implicit, transparent, and best-effort approach towards caching. A breadcrumb, which is a minimal piece of information, stores the direction in which a content was sent in the past, thus tying content routing with content location and caching. [72] focuses on caching chunks of a content. Some studies [73], [74] quantitatively analyze the
performances of caching techniques.

4) Mobility

No session is established between hosts and no concept of session helps ICN build in mobility. Two types of mobility should be supported in ICN in the following way. Mobility of a client, i.e., a host requesting content, is naturally supported [75]. Wherever the requesting host is, the requestor can access the content or its copy only by sending an Interest packet. Mobility support requires scalability and responsiveness in name-based routing and name resolution as described in (2) of this section. On the contrary, when content moves, new routing information is propagated in name-based routing or a new locator is registered in a name resolution server. NetInf addresses this mobility support for content [76], but it is an open issue. For example, how mobility is supported to contents identified by hierarchical names is not well addressed.

5) Other Issues

This subsection summarizes other issues including transport protocol, host-to-host communication and availability.

First, transport protocol functions including flow (congestion) control and reliability is different from end-to-end control of the host-centric model. ICN provides hop-by-hop congestion control based on interest packets. [77], [78] propose link layer transport protocols implementing an additive increase and multiplicative decrease similar to TCP congestion control algorithms.

Second, how (conventional) host-to-host communication is supported is an issue and [79] proposes a solution for VoIP (Voice-over IP). Third, there are still challenges about inter-domain routing [57] including routing policies [80] and incentives [81]. Fourth, availability issues, i.e., security issues, have started to be addressed including cache pollution [82] and prefix hijack [83].

Finally, one study [68] is skeptical about the value of CCN. The two questions, “what benefits do we think ICN designs offer, and are ICN designs the best way to achieve those benefits? (cited from [68])” are raised.

5. Mobility

5.1 Overview

Mobility was not inherently built into the current Internet. Locator/identifier separation schemes described in Sect. 3.2 and ICN described in Sect. 4 are candidate solutions and they naturally support mobility. Focusing on mobility support in a locator/identification scheme, this section describes MobilityFirst, a well-known architecture which addresses many design issues while focus on mobility.

There are two research challenges. One research challenge is scalability in identifier resolution mapping a host identifier to a (current) location. The current resolution techniques used by Mobile IP [38] and HIP [28] and dynamic DNS [84] are not so scalable to be used in highly dynamic environments. Another challenge is mobility support in challenged networks.

5.2 Challenged Networks

Challenged network studies [85], [86] focus on networks where end-to-end connectivity is not assured, contrary to that the current Internet implicitly assumes that hosts are continuously connected. Examples of such networks include wireless ad-hoc/sensor networks, vehicular networks, post-disaster networks [87] and so forth. Many future network architectures use techniques developed by challenged network studies because accommodating vehicular networks and sensor networks is one of the future network objectives.

In challenged network environments, which are called as Delay Tolerant Network (DTN) environments, connectivity is intermittent due to node failures, mobility, limited powers, disconnected links and so forth. In DTN studies, the end-to-end principle is re-defined. The bundle layer [85], [88] is introduced to provide a store-and-forward service as an overlay [88]. When a DTN node receives (accepts) a bundle, it is responsible for delivering the bundle. Since bundles should be stored at nodes’ buffers until they are delivered to receiver hosts, they tend to be stored for a long time. The custody transfer of bundle protocol lets the nodes delete (release) the bundles from their buffers.

Routing in a DTN is an important research issue [89]. While routing in deterministic contexts where it is known beforehand when and where nodes or hosts contact each other is relatively easy, the routing under random conditions is difficult. An epidemic routing [90] scheme is designed to forward bundles to all nodes except the one on which the message arrived. Please see the survey of routing protocols in DTNs [91].

5.3 MobilityFirst

Inspired by DTN, a network architecture based on the cache-and-forward paradigm, which is called Postcards from the edge [92], is proposed. This architecture is a transport layer solution that operates in a hop-by-hop store-and-forward manner with large files (chunks). It exploits the decreasing cost and increasing capacity of storage devices to provide unified and efficient transport services to hosts that may be wireless.

MobilityFirst [8], [93] is one of the network architectures that deals with many design issues while focusing on mobility. It is mainly inspired by the locator/identifier separation scheme and the cache-and-forward paradigm. The design goals of MobilityFirst are mobility as the norm with dynamic host and network mobility, robustness with respect to intrinsic properties of wireless medium, trustworthiness and privacy, usability features such as support for context-aware pervasive mobile services (cited from [93]).

It consists of key components as illustrated in Fig. 6. Actually some of challenges such as scalability in name
resolution are similar to those of ICN described in Sect. 4. MobilityFirst takes the locator/identifier separation scheme, wherein the host identifier is a self-certifying public key and the locator is a network address. A host identifier is a self-certifying public key network address that supports authentication and security. A key research challenge is the fast global dynamic name resolution service. DMap [31] distributes names to address mappings among routers using an in-network single-hop hashing technique. [94] compares the performances of the name resolution of DMap with those of name based routing of CCN.

MobilityFirst provides generalized delay-tolerant routing [95]. It proposes the concept of storage aware routing (STAR), which gives routers the option to temporarily store data as a routing decision. This implies that transport is conducted in a link-by-link fashion contrary to (conventional) end-to-end flow/retransmission controls in TCP [96]. This significantly decreases the role of the end-to-end control.

### 6. Security

#### 6.1 Overview

The current Internet does not build in security and it suffers from source spoofing, denial-of-service, route hijacking, and route forgery. Thus most future network architectures take security as one of the goals. Table 4 summarizes approaches of the architectures. As shown in the table, although built-in security is a fundamental requirement to future networks, issues other than accountability are not enthusiastically studied. Some new future network architectures such as MobilityFirst [8] and XIA [10] focus on trustworthiness in networks; however, trustworthy network research is still in an early stage.

#### 6.2 Accountability

The current Internet is prone to various attacks including source spoofing, denial-of-service, route hijacking, and route forgery. The vulnerabilities are caused by the lack of accountability. On the contrary, real-world security depends on accountability (Imagine, if you will be in a world where all actions are anonymous (cited form [97]).)

Accountable Internet Protocol (AIP) [97] provides accountability as a first-order property and proposes a hierarchy of self-certifying addresses like [37]. The AIP address is 160 bit long and includes this self-certifying address, i.e., a 144 bit public key hash as shown in Fig. 7. Such self-certifying addresses are assigned for both domains and hosts and they allow domains and hosts to prove they have the address they claim to have without any trusted third party. And thus AIP follows the principle of locator/identifier separation. [98] takes an incremental approach and uses only self-certifying AS identifier with the help of DNSSEC.

Many new network architectures adopt accountability as one of their major design principles. The adoptions are classified into host-based identifiers and content-based identifiers. In the host-centric model, an identifier of a host is a hash of host’s public key [8], [10], [27], [28], [30]–[32]. On the contrary, in the information-centric model, an identifier of the content is self-certified [15], [55]–[57]. An advantage of self-certifying addresses and identifiers is that there is no need of trusted third party like a PKI infrastructure. On the contrary, content-based security [36], [62] of CCN needs a trusted third party to let all contents have digital signatures.

#### 6.3 Privacy and Trustworthiness

Protecting privacy is an important research issue because future networks/Internet inherently support mobility in a wireless environment as MobilityFirst [8] does. Although protecting privacy is a design principle, there are not many studies in this area. The main research motivation is that concealing endpoints’ information from all routers and hosts unless they need the information to perform their assigned network functions. A research project in NeTS FIND initiative [99] proposes a solution by using encrypted addresses. Anonymity communication such as tor [100] is another solution. [101] proposes a lightweight anonymous communication protocol and shows that it can be built into future network architectures.
6.4 Security Policy

Security policy management is studied in XIA [10] and Secure Architecture for the Networked Enterprise (SANE) [102]. SANE provides a centralized network security policy management in an administrative domain. However, it focuses on enforcing security policies in an enterprise network and its control is strict. A SANE network consists of switches that are not capable of IP routing and all communications between hosts in the SANE network are controlled by the domain controller. The domain controller constructs a spanning tree rooted at it. However, the scalability at the Internet scale is not well addressed in SANE.

7. Network Management

7.1 Overview

The Internet began with a few hundreds of nodes and has become a massive distributed system consisting of millions of nodes. Since management of such a large system is much more complex than the original Internet, a new management framework leveraging knowledge [103], self-management and evolvability are required.

The complexity originates because forwarding, routing control and management planes are tightly coupled in the current Internet and thus many studies re-consider network management architectures focusing on the complexity. Some network management architectures [45], [104]–[107] focus on the separation of routing and forwarding, others [108], [109] focus on that of control and management planes. Some network architectures [45], [104]–[108] adopt a centralized approach and thus the scalability at the Internet scale is not still well addressed. Others [109]–[111] adopt a decentralized or autonomous approach. Table 5 summarizes the features of such network management architectures.

7.2 Routing Management

Routing in the current Internet is an important issue among many other issues including security, bootstrapping and failure recovery. The problem originates because the Internet bundles control logic and packet handling into the individual routers [45], [104]. For example, the path-computation logic is governed by distributed routing protocols. They let the routers learn about the topology as well as select paths. Another example is load-balancing implemented by carefully tuning OSPF link costs.

4D architecture [104], [105] proposes a clean slate approach to data-network control and management, contrary to the current distributed routing control. It is based on three principles: Network level-objectives are the goals or objectives for the network. For example, a reachability policy objective could be stated as “do not allow hosts in subnet B to access the accounting servers in subnet A.” A network-wide view is a coherent snapshot of the state of each network component. Direct control means the ability to manage all states in the data plane so as to direct packet forwarding.

The control plane is decomposed to achieve the direct control as illustrated in Fig. 8. The dissemination plane provides a robust and efficient communication substrate that connects routers and switches with decision elements. (In this sense, the control and data planes are independent like circuit switch networks.) The discovery plane discovers the physical components in the network and creates logical identifiers to represent them. Replacing today’s management plane, the decision plane makes operations in real time on a network-wide view and makes all decisions driving network control to achieve network-level objectives. The data plane is responsible for forwarding packets.

There are several studies that adopt a centralized solution for routing management such as 4D. RCP [45] is similar to the 4D architecture; however, it is incremental (backwards compatible) and focuses on scalability in i-BGP full-meshes. [106] extends RCP and proposes a routing architecture wherein the routers act like forwarders while the routing computation is performed centrally.

7.3 Management Model

Complexity Oblivious Network Management (CONMan) [108] recognizes the complexity of the control and management planes, too, and restructures them inspired by the 4D architectures’ discovery plane. Not restricting it to just routing management, CONMan provides self-configuration, abstraction, and declarative specification for more general network management. For example, self-configuration of

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Central or Distributed</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCP [45]</td>
<td>Central</td>
<td>Separation of Routing and Forwarding</td>
</tr>
<tr>
<td>4D [104], [105]</td>
<td>Central</td>
<td>Separation of Routing and Forwarding</td>
</tr>
<tr>
<td>[106]</td>
<td>Central</td>
<td>Separation of Routing and Forwarding</td>
</tr>
<tr>
<td>OpenFlow [107]</td>
<td>Central</td>
<td>Separation of Routing and Forwarding, Flow management</td>
</tr>
<tr>
<td>CONMan [108]</td>
<td>Central</td>
<td>Separation of Control and Management Planes, Declarative Specifications, Cross-layer Management</td>
</tr>
<tr>
<td>[110], [111]</td>
<td>Distributed</td>
<td>Autonomous Network Management, Separation of Control and Management Planes</td>
</tr>
</tbody>
</table>

Fig. 8 4D architecture (cited from [105]).
networks contributes to decreasing humans’ configuration errors because configuration errors are a major reason for IP backbone failures [112].

Thus in CONMan, protocols are abstracted as modules and network configuration is represented by piping the modules so that the management plane can understand the potential of the underlying network and configure it without dealing with the details of the protocol/devices. CONMan also covers important network design choices: declarative specifications for routing management [113], [114] and the database and interfaces for the cross-layer management [115].

[110], [111] propose an in-network management paradigm that builds management functions into the network and network elements. The goals are evolvability, self-management and so forth. Autonomous management enables management systems to be independent of any external technical interference and do self-management. ANA [109] also proposes a generic architectural framework for autonomic systems composed of autonomic devices including autonomic network management.

7.4 Flow Management

Centralized management is the current trend of network management for enterprise, data center and Internet management, too. OpenFlow [107] takes a centralized approach by which a centralized controller manage all flows (paths) among all hosts in an administrative domain. Software Defined Network (SDN) takes a similar approach to OpenFlow. Although many centralized solutions have been proposed, most solutions focus on routing/path management in a single administrative domain. Thus studies on other management functions in interdomain environments are desired.

8. Service Centric Architecture

8.1 Overview

In the 1970’s, one goal of the Internet was to have a simple packet-switched communication and the end-to-end argument has dominated the current Internet design. The Internet itself is kept simple and most of the complexity is implemented on end-systems. However, although commercialization of the Internet introduces diverse service requirements, the end-to-end design lacks flexibility to adapt to new service requirements [1]. Inspired by this discussion, skepticism has been raised regarding protocol implementations based on the layered architecture. Thus frameworks for flexibly implementing end systems and services have been proposed. These frameworks raise issues of modularity and a layer structure to achieve the flexibility. Table 6 summarizes the approaches to the two issues adopted by the frameworks.

8.2 Modularity

Modular implementation is a solution for achieving the flexibility and thus modular frameworks supporting protocol implementations based on the layered structure have been studied such as x-Kernel [116] and the Click router [117]. On the contrary, addressing the ossification caused by the layered architecture, many network architectures propose a modular framework for providing a new service and a new protocol stack in a flexible way, avoiding the ossification of the current Internet, mapping service requirements to low layer implementations and so forth. The basic idea of the architectures is how a service is formed by combining elementary blocks in a network.

RNA [118] supports non-layered protocol implementations contrary to [116], [117]. It is a framework wherein protocols may be dynamically composed depending on the context relative to the layers below and above. The idea is to provide a meta-protocol, i.e., a single protocol that is instantiated and customized at different layers of a protocol stack as illustrated in Fig. 9.

Several service centric frameworks [119]–[121] are proposed to flexibly develop a service using building blocks, too. [119] defines a service architecture, called Information Transfer and Data Service (ITDS), which bases communication abstractions on the transfer of information rather than the process of sending data (cited from [119]). The separation of communication and processing enables to provide various information transfer patterns. ITDS uses router-based functionality to implement services on nodes in the network. The key idea is to help nodes decide where to assign the processing task across them in the network.

RNA [118] supports non-layered protocol implementations contrary to [116], [117]. It is a framework wherein protocols may be dynamically composed depending on the context relative to the layers below and above. The idea is to provide a meta-protocol, i.e., a single protocol that is instantiated and customized at different layers of a protocol stack as illustrated in Fig. 9.

Several service centric frameworks [119]–[121] are proposed to flexibly develop a service using building blocks, too. [119] defines a service architecture, called Information Transfer and Data Service (ITDS), which bases communication abstractions on the transfer of information rather than the process of sending data (cited from [119]). The separation of communication and processing enables to provide various information transfer patterns. ITDS uses router-based functionality to implement services on nodes in the network. The key idea is to help nodes decide where to assign the processing task across them in the network.

RNA [118] supports non-layered protocol implementations contrary to [116], [117]. It is a framework wherein protocols may be dynamically composed depending on the context relative to the layers below and above. The idea is to provide a meta-protocol, i.e., a single protocol that is instantiated and customized at different layers of a protocol stack as illustrated in Fig. 9.

Several service centric frameworks [119]–[121] are proposed to flexibly develop a service using building blocks, too. [119] defines a service architecture, called Information Transfer and Data Service (ITDS), which bases communication abstractions on the transfer of information rather than the process of sending data (cited from [119]). The separation of communication and processing enables to provide various information transfer patterns. ITDS uses router-based functionality to implement services on nodes in the network. The key idea is to help nodes decide where to assign the processing task across them in the network.

RNA [118] supports non-layered protocol implementations contrary to [116], [117]. It is a framework wherein protocols may be dynamically composed depending on the context relative to the layers below and above. The idea is to provide a meta-protocol, i.e., a single protocol that is instantiated and customized at different layers of a protocol stack as illustrated in Fig. 9.

Several service centric frameworks [119]–[121] are proposed to flexibly develop a service using building blocks, too. [119] defines a service architecture, called Information Transfer and Data Service (ITDS), which bases communication abstractions on the transfer of information rather than the process of sending data (cited from [119]). The separation of communication and processing enables to provide various information transfer patterns. ITDS uses router-based functionality to implement services on nodes in the network. The key idea is to help nodes decide where to assign the processing task across them in the network.

RNA [118] supports non-layered protocol implementations contrary to [116], [117]. It is a framework wherein protocols may be dynamically composed depending on the context relative to the layers below and above. The idea is to provide a meta-protocol, i.e., a single protocol that is instantiated and customized at different layers of a protocol stack as illustrated in Fig. 9.

Several service centric frameworks [119]–[121] are proposed to flexibly develop a service using building blocks, too. [119] defines a service architecture, called Information Transfer and Data Service (ITDS), which bases communication abstractions on the transfer of information rather than the process of sending data (cited from [119]). The separation of communication and processing enables to provide various information transfer patterns. ITDS uses router-based functionality to implement services on nodes in the network. The key idea is to help nodes decide where to assign the processing task across them in the network.

RNA [118] supports non-layered protocol implementations contrary to [116], [117]. It is a framework wherein protocols may be dynamically composed depending on the context relative to the layers below and above. The idea is to provide a meta-protocol, i.e., a single protocol that is instantiated and customized at different layers of a protocol stack as illustrated in Fig. 9.
ing blocks are services. A service is a self-contained operation that is relevant to a specific communication task such as packet fragmentation and encryption. SILO allows any set of services to be selected dynamically for a task contrary to that the existing frameworks [116] for realizing protocols take the layered approach.

8.3 Cross-Layer Design

The above frameworks adopt modular building blocks to provide flexibility and some [118], [121] adopt a non-layered structure that enables cross-layer design. In wireless networks, protocols according to the layered architectures may not always provide good performance due to the highly variable nature of wireless links and the resource nature of (mobile) hosts [122]. Cross-layer feedback wherein the upper-layer utilizes the under-layer’s performances as hints is proposed to improve performances [123], [124]. Cross-layer design is one of the main goals of AKRAI project [125].

Thus a cross-layer design is supported by some frameworks [118], [121]. SILO [121] aims at a cross-layer design to optimize performances and [126] proposes a cross layer negotiation mechanism that sets up a complete stack of connection-oriented protocols of which multiplayer protocols perform a handshake. However, cross-layer design frameworks require further research.

9. Network Virtualization and Testbeds

9.1 Overview

It is difficult to deploy and evaluate new technologies and protocols in realistic size testing environments. There is a growing interest in virtualized networks as a means of enabling experimental evaluation of new network architectures on a realistic scale. Network virtualization provides efficient methods for resource sharing by multiple concurrent experiments on the same testbed.

Inspired by the success of early testbed facilities [127], [128], network virtualization architectures are expected to play an important role as testbeds as well as commercial network infrastructure. The challenges of network virtualization are classified into isolation, security, scalability, programmability, performance and management [129], [130]. Isolation of resources is one of the most important challenges to which most network virtualization architectures give importance. Security and privacy issues specific to network virtualization must be identified. For example, programmability increases vulnerability without secure programming models. Scalability is the number of slices how many architectures a testbed can support.

Table 7 summarizes how network virtualization architectures handle these research issues.

<table>
<thead>
<tr>
<th>Research Issues</th>
<th>Network Virtualization Architectures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isolation, Security, Scalability</td>
<td>PlanetLab [127], VIINI [128], CoreLab [131], Diversified Internet [132], OpenFlow [107], GENI [140], VNode [41]</td>
</tr>
<tr>
<td>Programmability</td>
<td>PEARL [134], Supercharging PlanetLab [135], NetPGA [137], [138]</td>
</tr>
<tr>
<td>Performance</td>
<td>[139]</td>
</tr>
<tr>
<td>Management</td>
<td>GENI [140], VNode [41]</td>
</tr>
<tr>
<td>Service Model</td>
<td>Cabot [42], MDR [143]</td>
</tr>
</tbody>
</table>

9.2 Network Virtualization

9.2.1 Early Testbed

PlanetLab [127] and VIINI [128] are early testbed facilities that effectively mimic the scale of the Internet. In these testbeds, Internet nodes run Linux virtual software to virtualize its resources to allocate isolated resources to individual distributed applications/protocols which are concurrently under experimentation. Experiments are allocated a slice that is composed of multiple slices spanning multiple sites. Inspired by its success, network virtualization architectures and their application to testbeds for future network architectures are being studied. CoreLab [131] incorporates flexibility and code-usability to PlanetLab by employing the hosted virtual machine monitor.

9.2.2 Network Virtualization Architectures

Most virtualization network architectures proposed so far focus on isolation, security, scalability and programmability. [132] proposes to make network virtualization as a core capability of a future diversified Internet with the objectives of isolation and fairness. The fundamental abstractions for a virtualized network are substrate routers, which are connected to each other by physical links, and metarouters, which are hosted on substrate routers and are connected to each other by virtual metalinks carried over physical links.

Programmability for packets as well as for flows [107] is an important issue inspired by the active network researches [133]. Most virtualization architectures provide programmability for packets and some of them provide fast path data processing. Performance and management among the above challenges and service models (business models) are important so that such architectures can be applied commercially in the future.

Performance, which virtualization based on the overlay network approach sacrifices, is an important challenge [134]. Routers supporting virtualization (virtual routers) should provide good performance of the data plane with maintaining isolation among slices especially in the case that the number of slices increases. [135] proposes a supercharged PlanetLab platform that implements separate slow and fast paths for packet processing and forwarding. The slow path is chosen for application specific processing while the fast path is optimized for line-rate packet forwarding.
proposes a virtualized data plane based on the programmable network processing hardware NetFPGA [137]. [138] uses programmable Network Interface Cards (NICs) for high performance. [139] considers implementing a high performance router on commodity hardware.

Management issues must be resolved in order to operate testbeds as well as commercial networks wherein thousands of slices are used in the experiment. Some testbed projects based on network virtualization network architectures start developing management tools/frameworks such as GENI [140] and Vnode [141].

9.2.3 Service Model

Network virtualization may change the conventional model of Internet Service Providers (ISPs) in the current Internet. It may decouple infrastructure providers who maintain network equipment from service providers who offer end-to-end services. Concurrent Architectures are Better than One (Cabo) [142] uses network virtualization to allow a service provider to simultaneously run multiple end-to-end services over different infrastructure providers. This decoupling may raise problems in the confidentiality of Service Providers (SPs). [143] addresses the problem that routing information of SP is leaked to Infrastructure Providers (IPs) if the IPs are competitors to the SP. It proposes a Minimum Disclosure Routing (MDR) that hides the routing information of the SP to underlying IPs. The solution is based on secure multi-party computation and applies it to distributed routing.

9.3 Testbed Projects

Developing future network architectures requires large-scale testbeds and thus many testbeds such GENI [3], New Generation Network Testbed JGN-X [144] and Future Internet Research and Experimentation (FIRE) [145] have been constructed. These testbeds are based on network virtualization. Among them GENI is one of the first testbeds based on network virtualization. The key idea of GENI is to build multiple slices out of the substrate for resource sharing and experiments. It consists of physical network substrates and a global control and management framework. The research projects developing the testbeds try challenges related to large-scale hardware, software, distributed system tests and maintenance, security and robustness, coordination, openness, and extensibility.

10. Other Issues

This paper surveys mainly clean-slate proposals related with the network layer and the above. This section discusses other issues such as evolvability, cloud networking architectures and network science.

10.1 Evolvability

The first issue is the debate between clean-slate versus evolutionary (incremental) approaches [146]. Many clean-slate designs surveyed in this paper impose no restrictions or assumptions on the current Internet/network architectural designs. Even if they are revolutionary, they should be evolutionary so as to accommodate legacy networks including the Internet and circuit networks as well as to handle new requirements.

Either approach raises an issue of how to support evolvability. We cannot predict how the Internet would be used 50 years hence and thus the future Internet should be evolvable to keep up with the progress of substrate technologies and applications.

Some studies [10], [23] address this issue. [147] proposes a model to mathematically analyze how layer structures are evolved and implies that the narrow waist of the future Internet would be above the current IP layer. Shad-owNet [148] which is based on network virtualization, is designed to accelerate network evolution. The idea is that the infrastructure is connected to, but functionally separated from a production network, thus enabling realistic testing. (Network virtualization is discussed in Sect. 9.)

eXpressive Internet Architecture (XIA) [10]–[12] considers that today’s incremental deployment has the drawback. Such a deployment hides new functionalities usually implemented by tunnels and overlays from the underlying IP network. XIA proposes to incorporate expressiveness and evolvability into the network layer (i.e., the IP layer). The key element is the abstraction of a principal that is a receiver of a packet such as a host and an application. Each type of principal is associated with a different contract with the network and applications are able to specify their intent by choosing the appropriate principal types. This expressiveness enables evolvability.

10.2 Cloud Networking Architecture

Cloud computing based on data centers is rapidly becoming popular and many could services including Infrastructure as a Service (IaaS) and Software as a Service (SaaS) are commercially available. Although this computing model offers significant economic advantages of scale by sharing hardware among applications, the network architectural consideration is missing. And thus cloud networking architectures becomes a hot research topic and they take an evolutionary approach.

NEBURA [9] proposes a future cloud networking architecture that interconnects data centers and connects users to their data. It addresses the three challenges: (1) it is intrinsically more secure, (2) it provides flexibility for further applications, (3) it provides a viable path for migration and deployment that is conscious of technical feasibility, economics, and regulation. The architecture consists of the data plane protocol for establishing policy-compliant paths, the control plane providing access to application-selectable services and the core that redundantly interconnects data centers.
10.3 Network Science

Network science takes the most clean-slate approach for the future Internet/networks. Over the past forty years, computer networks especially the Internet have changed in terms of size (the number of nodes) and Quality of Service (QoS). The networks have become more complex than the conventional theoretical and mathematical techniques can handle. Thus deep, new scientific understanding about their complexity is required for their future design. In recognition of the importance of this area, the United States has launched a research program called as Network Science and Engineering (NetSE) [149] and in Japan, IEICE has published the special sections related to network science [150]. Research issues include routing [151]–[154], biology [154] or brain [155] inspired network control, autonomous algorithms [156], [157] for dealing with complexity in the future Internet/networks and socio-technological issues.

Among the issues, routing has been well studied. 

Greedy Routing on Hidden Metrics (GROH) [151] is an innovative approach. The study assumes that the scalability problem would not be due to large storage space in routers, but due to the churn as a result of routing updates. GROH proposes a greedy forwarding based on a hidden metric space by leveraging the small world phenomena in the Internet. The idea behind that is “Nodes in complex networks exist in some spaces that underlie the observable network topologies (called as hidden metric spaces (Cited from [151]))”. Routing control messages become unnecessary due to such an observable network topology. However, the study itself is in an early stage. This study follows greedy routing in wireless sensor and ad-hoc networks wherein geographical information such as virtual coordinate information is used to reduce routing control messages [152], [153].

10.4 Miscellaneous Issues

Other important issues are energy efficiency (energy saving) [158] and substrate technology innovations in wireless [159] and optical networks [160]. The papers [158]–[160] explain the current trends in these research areas.

11. Conclusion

This paper surveys clean-slate future Internet/network architectures in terms of architectures and protocols. Subjects discussed include the narrow waist in the future, naming, security, mobility, service, network virtualization and so forth. The idea behind this discussion is how ossification of the Internet can be resolved and how the functions that the current Internet lacks such as security and mobility can be built into future Internet/networks. Many architectures and protocols have been proposed and their deployment and evaluation over large-scale testbeds have just begun. However, there are many challenges to be tackled in achieving future Internet/networks. It is hoped that this paper will encourage young researchers to take on such research challenges.

References


[38] ChoiceNet, https://code.renci.org/gf/project/choicenet/


IEICE TRANS. COMMUN., VOL.E96–B, NO.6 JUNE 2013


Toru Hasegawa received the B.E., the M.E. and Dr. Informatics degrees in information engineering from Kyoto University, Japan, in 1982, 1984 and 2000, respectively. Since joining KDD in 1984, he has been working in the field of formal description technique (FDT) of communication protocols, ATM and high-speed protocols. From 1990 to 1991, he was a visiting researcher at Columbia University. His current interests are future Internets, mobile computing and so on. He is currently a professor at Osaka University. He was a guest professor at National Institute of Informatics from 2002 to 2006. He received IPSJ Convention Award in 1987 and the Meritorious Award on Radio of ARIB in 2003. He is a fellow of IPSJ.