The Network Empowered Internetwork (NEI): An Architecture Overview (Extended Version)

Malathi Veeraraghavan
Dept. of Electrical and Computer Eng.
University of Virginia
Charlottesville, VA 22904–4743
Email: mvee@virginia.edu

Martin Reisslein
School of Electrical, Computer, and Energy Eng.
Arizona State University
Tempe, AZ 85287-5706
Email: reisslein@asu.edu

***Abstract*—After classifying internetworks into type internetworks, protocol internetworks, and organization internetworks, this paper focuses on the last of these. A key problem with organization internetworking is global routing scalability. In developing the NEI architecture to address this problem, some basic mechanisms/tenets of IP internetworking are upended. In NEI, internetworking addresses are not assigned to hosts or gateways (or even their interfaces); instead a four-tuple vector of independently administered intra-organization addresses, stub organization identifiers, transit organization identifiers, and country codes is used. With this basic change, DHCP, ARP and longest-prefix matching are no longer required. Another basic tenet for recent work on location/identifier separation, which is that stub organizations should be given provider independent addresses to minimize changes required when such organizations change their transit providers, is also confronted. A holistic solution that takes a "design-for-change" approach allows for a draconian rule of only allowing provider aggregatable addressing for stub organizations. It leverages the location-dependent four-tuple concept and combines this with careful DNS resource record structures, and DNS cache updates. Multihoming, mobility, and transit provider changes in NEI are also described.**

I. Introduction

While the current TCP/IP based Internet has been highly successful and become an integral part of our information society, critical problems with the TCP/IP internetworking mechanisms have emerged with the tremendous growth of the Internet [1]. In this paper we propose a novel solution for a future internetwork design. Our solution "empowers" networks to leverage their own *network* addressing schemes and network protocols, allowing for lightweight *internetworking* addressing and protocols.

Our *Network Empowered Internetwork (NEI)* achieves the lightweight internetworking solution through (i) combining mechanisms that improve global routing scalability, such as country codes from OSI NSAP addressing and provider aggregatable (PA) addressing, while (ii) avoiding mechanisms that contribute to scalability problems, such as per-host internetworking addresses and provider independent (PI) addresses. By empowering networks to employ their own independent addressing and routing mechanisms, the NEI achieves a global scalable internetwork that readily supports multi-homing, mobility, and traffic engineering.

II. NEI Structural Overview

A. Network

We define a **network type** as a unique combination of characteristics of the data sources/sinks (e.g., fixed or mobile), communication links (e.g., wired or wireless), and switches (e.g., packet or circuit). For instance, a set of fixed (non-mobile) data sources and sinks interconnected by wired links and connectionless packet switches are a unique network type. A set of data sources/sinks interconnected by communication links (and optionally switches) of a given network type forms a **network** if the links (and switches) run the same protocol set and belong to the same organization. For instance, an enterprise network consisting of data sources/sinks with Ethernet interface cards interconnected by Ethernet switches form a network as all components use the same protocol and are owned by the same organization.

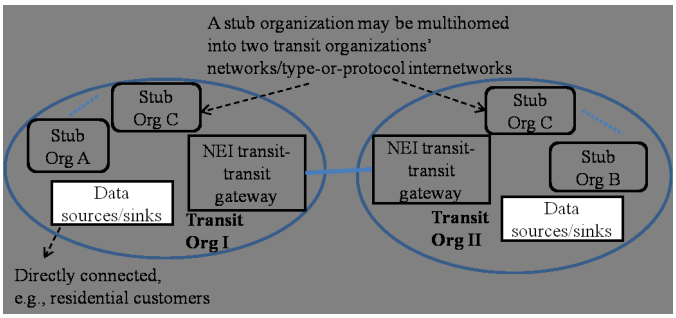
B. Internetwork

We define an **internetwork** as two or more distinct "entities" interconnected by gateways, whereby the interconnected "entities" differ according to the different types of internetworks defined below. We distinguish three types of internetworking:

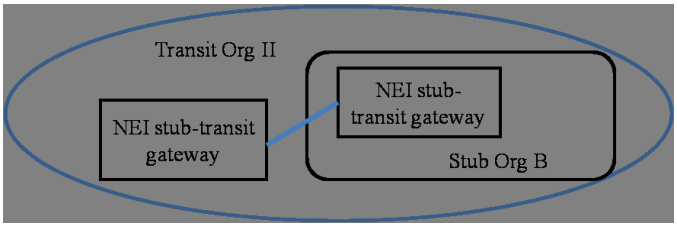
Type internetworking: The interconnected entities are networks of different network types (which usually imply different protocols). An example is a WiFi network with mobile nodes and wireless links interconnected to an Ethernet network with fixed sources/sinks and wired links.

Protocol internetworking: The interconnected entities are networks of the *same* network type but run different protocols. For instance, consider a data center with fixed data sources/sinks, wired links, and connectionless packet switches. The data center will typically use the InfiniBand protocol for interprocessor communication, but Ethernet for wide area access; hence, forming a case of protocol internetworking.

A type internetworking gateway necessarily requires functions to internetwork protocols since different protocols are needed to support different network types, but a type internetwork is a not a "protocol internetwork" as the definition of the latter requires that the connected networks be of the same type.



(a) Transit-to-transit level



(b) Stub-to-transit level

Fig. 1. Illustration of NEI organization internetworking

Therefore, "type internetworks" and "protocol internetworks" are two independent terms.

Type internetworks and protocol internetworks require **gateways** that are tailored to the specific combination of internetworked network types and protocols and we leave the study of such gateways for future research. In this paper we focus on the orthogonal problem of organization networking.

Organization internetworking: The interconnected entities can be networks, type internetworks, or protocol internetworks. Further if the entities owned by different organizations are also networks of different types or protocols, then in addition to the organization internetworking gateway functions, the type/protocol internetworking gateway functions are required. Therefore, the organization internetworking function is orthogonal to the type/protocol internetworking function.

As stated in Section I, the focus of this paper is on organization internetworking functions.

III. NEI Architecture

From an internetworking perspective, organizations are typically classified into two types: **Transit organizations**, which provide transit service, and **stub organizations**, which do not. ISPs are transit organizations, and universities/businesses/governmental agencies are usually stub organizations. Each organization can operate a **network**, as per our definition in Section II-A, or a **type internetwork or protocol internetwork**, which are defined in Section II-B.

A. Addressing

Unlike with IPv4, which assigns each data source/sink an IPv4 address consisting of a SubnetID and a HostID, in NEI there is no such global internetworking address allocation. As the future Internet is expected to be an "Internet of Things" interconnecting sensors and actuators along with computers and hand-held devices, we propose a fundamental change in thinking. Since a data source/sink is part of a network, which is owned by an organization, the NEI architecture, in its goal of empowering networks, starts by empowering individual organizations to assign `intra-org addresses` to its data sources/sinks. These addresses are the only addresses required for **intra**-organization communications. For **inter**-organization communications, all that is necessary is an additional globally unique organization identifier. In other words, there is *no* globally unique NEI address, comparable to an IPv4 address, assigned to every data source/sink. Instead, if, for example, packet based communication is used, an inter-organization packet header would contain the source and destination organization identifiers and source and destination `intra-org addresses`.

This concept is comparable to addressing in the physical world. A house (data source/sink in the analogy) in a city (organization) is identified by a combination of house number and street name (`intra-org address`) that is unique within that city (organization). This is all that is required to drive to that house if starting out from within that city. The globally unique address by which that house is known worldwide does not start from scratch. Instead it is created by appending city, state, and country identifiers to its intra-city address.

The next step is to design the assignment of organization identifiers. Having learned from provider independent vs. provider aggregatable IP addressing, the NEI design proposes to have an organization, such as the Internet Assigned Numbers Authority (IANA) control the assignment of globally unique `TransitOrgIDs`, which are limited to only transit organizations, i.e., ISPs. Stub organizations are identified by `StubOrgIDs`, the assignment of which is completely under the control of the transit organizations to which stub organizations connect.

This concept is illustrated in Fig. 1. There are two levels of organization internetworking: transit-to-transit, and stub-to-transit, and correspondingly two types of NEI gateways, an **NEI transit-transit gateway** and an **NEI stub-transit gateway**. In actual deployments of gateways it is common, as shown in Fig. 1, that each organization purchases its own gateway; and the gateways interconnects via a point-to-point link. In addition to stub organizations, a transit organization may have directly connected data sources/sinks, e.g., residential customers. We propose to use a default `StubOrgID` of 0 for these directly connected nodes. Finally, Fig. 1(a) shows an example stub organization C that is multihomed to two transit organizations' network/type-or-protocol internetwork. Multihoming is also possible for the directly connected data sources/sinks.

Effectively, this structure limits all stub organizations to be only assigned the equivalent of provider aggregatable (PA) addresses. The two noted drawbacks of PA addresses relate to **multihoming** and the need for renumbering when a stub organization **changes its transit provider**. Solutions to both are described in Section III-E.

To avoid the global routing scalability problem faced by IP, NEI adopts the concept of CountryCodes. A key point though is that the assignment of TransitOrgIDs is independent of CountryCode assignments. This allows transit organizations that operate in multiple countries to use the same TransitOrgID. The rare cases in which a country has non-contiguous regions can be handled by assigning multiple country codes.

Three points to note in comparing NEI addressing with IP addressing are as follows. **First**, TransitOrgID, StubOrgID, and CountryCode are single specific values, not ranges of addresses. **Second**, all three are variable-length identifiers as is the intra-org address. The latter has to be variable-length if organizations are to be empowered to choose their own internal addressing scheme. This also applies to StubOrgIDs, which are chosen by transit organizations. Additionally, by making the TransitOrgID and CountryCode variable-length, we avoid identifer space exhaustion. How this choice of variable-length identifiers and addresses affects packet header processing is described in Section III-G. **Third**, the inter-dependence between subnetID space and hostID space in IPv4 (owing to the fixed 32-bit length of IPv4 addresses) is not present in NEI addressing, as all three identifiers, TransitOrgID, StubOrgID, and CountryCode, and the intra-org address are completely independent of each other.

B. Naming

The need for location-independent names to complement location-dependent addressing (as described in Section III-A) is well understood, and the current Domain Name System (DNS) (with security enhancements) is reused in NEI with a few modifications. Fortunately, the query class field in DNS requests/responses, which today is set to 1 for Internet, allows for the addition of a new class for NEI.

In NEI: (i) all domain names end in a CountryExtension (today, many countries use such extensions), (ii) DNS servers implement a cache-update protocol, and (iii) DNS resource records are structured as described below. The need for these three concepts will be seen in Section III-E. The CountryExtension is needed to limit the number of DNS servers to which cache updates are propagated using the cache-update protocol, when a stub organization changes transit providers. Note the difference between CountryExtension, which is part of naming, and CountryCode, which is a part of addressing.

DNS resource records are structured as illustrated in Table I. The resource records are divided into one common row followed by **two** sets. The *first* (common) row maps the organization name to one or more {TransitOrgID, StubOrgID} pairs (multiple if the organization is multi-homed). *Set 1* maps the local name of each data source/sink (e.g., athena, wuneng in Table I) to its intra-org address. This structure is designed to ease the transition process when stub organizations change their transit providers, as only the first row needs to be changed. This is unlike with IP, where full domain names (e.g. athena.nctu.edu.tw) are mapped to IP addresses, requiring modifications of all entries if the subnetID is changed. In NEI, the full mapping is maintained only for those data sources/sinks that are owned by the organization identified in the first row, but are located in some other transit organization's network/type-or-protocol internetwork. This constitutes *Set 2* as shown in Table I.

Three points to note are as follows. **First**, the Countrycode corresponds to the country in which the named data source/sink is physically located. For example, if a transit organization spans multiple countries, the names of its servers located in different countries are mapped to the respective country codes, but the servers have the same TransitOrgID.

Second, choices for intra-org addressing are considered. For ease-of-transition, one concept is to simply use the 6-Byte IEEE 802/Ethernet address as intra-org addresses. Given the dominance of Ethernet and 802.11 wireless LANs, the use of this address is ubiquitous. Its flat addressing structure is not a handicap for small-to-medium sized organizations. The popular textbook [2] compares 6-byte Ethernet addresses to social-security numbers and refers to them as "link-layer addresses." Historically, Ethernet networks started out as single-link broadcast networks, but today's Ethernet networks are primarily switched multiple-link networks. As the OSI function classification [3] lists switching (frame forwarding) as a network layer function, we view the Ethernet address in a multiple-link switched Ethernet network as an "intra-network address." Furthermore, as the same address structure is used in other popular 802 networks, such as 802.11 wireless LANs, it also serves as an addressing solution for some type/protocol internetworks. Therefore, it is an ideal candidate for intra-org addressing in small-to-medium sized organizations.

The analogy to social security numbers is unfortunate as it raises the specter of security vulnerabilities as an NEI DNS lookup would return these 6-byte Ethernet/802 addresses as intra-org addresses for organizations that choose this approach. But we see this analogy as merely an attempt to explain the structure of why a data source/sink interface is assigned both an IPv4 address and a MAC address, as there is no discussion of security in this context. The physical-world addressing analogy described in Section III-A lays bare the oddity of this scheme. The origin of this idea of ignoring intra-network addresses and assigning a complete globally unique internetworking address dates back to Cerf and Kahn's original 1974 paper [4], which states that "the source and destination entries [referring to the equivalent of IP addresses] uniformly and uniquely identify the address of every HOST in the composite network." Effectively, this creates one global-scale homogeneous IP based network, not an internetwork of individual networks, as the NEI proposes. This assignment of a globally unique internetworking address to each data source/sink contributes significantly to the present problems, such as running out of IP addresses and global routing scalability.

TABLE I
AN ILLUSTRATIVE EXAMPLE OF DNS RESOURCE RECORDS

Name	Resource record type	Value
Common entry for the whole organization		
nctu.edu.tw	organization IDs	One or more {TransitOrgID, StubOrgID}
Set 1: Individual entries mapping local names of data sources/sinks owned by the organization listed in Set 1		
athena	intra-org address	01:89:5f:3e:6a:8b (small stubs) OR IP address (large stubs)
wuneng	intra-org address	05:91:fe:12:56:9d (small stubs) OR IP address (large stubs)
Set 2: For data sources/sinks located in other transit organizations but owned by the organization listed in Set 1		
www.nctu.edu.tw	Three-tuple vector	One or more {TransitOrgID, StubOrgID, intra-org address}

For large stub organizations where some form of hierarchical addressing may be required internally, IPv4 addressing qualifies well for intra-org addressing.

Third, an NEI DNS name lookup could result in multiple **four-tuple vector** responses, where each vector consists of {intra-org address, StubOrgID, TransitOrgID, CountryCode}. This concept will be used for multihoming, see Section III-E.

C. Parameter configuration in data sources and NEI gateways

Simply put, DHCP is eliminated in NEI. DHCP servers are required in IP to handle the assignment of the HostID portions of IP addresses. As NEI eliminates the need for such assignments, DHCP is simply not required.

What configuration of parameters, if any, is required **for organization internetworking**? The IP answer requires configuring the IP address, subnet mask, and default gateway IP address into all data sources/sinks, along with a DNS server's IP address. Such a configuration is required because in the IP solution, the IP layer runs on top of intra-network protocol layers.

In our complementary work on type/protocol internetworking, NEI **removes this requirement of having to run an internetwork layer protocol** for several types of data sources/sinks, such as **energy-constrained sensors** or **time-constrained fast moving vehicular nodes**. Protocol conversion functionality will be used in these types of environments. In cases where such data sources engage in inter-organization communications, all that needs to be configured in data sources/sinks is the intra-org address of a Stub or Transit organization's "post office" server. A data source can send its message along with only the destination name to the post office server. The post office server then looks up the DNS, adds internetwork layer protocol headers, and sends the packet to an appropriate NEI gateway for forwarding. In this mode of operation, the data sources/sinks do not require any configuration for organization internetworking.

For the more **traditional computer based data sources/sinks** where applications require many back-and-forth packet communications, the source needs to be configured with the StubOrgIDs and TransitOrgIDs for inter-organization communications, and the CountryCode based on its location. For the configuration of TransitOrgIDs and StubOrgIDs, NEI adopts the IP **gateway advertisements/solicitations** mechanism, which works well here as the same identifiers are sent to all data source/sinks. Along with TransitOrgIDs and StubOrgIDs, these advertisements also carry the intra-org address(es) of the NEI gateway(s) and a DNS server's intra-org address.

For CountryCode, the NEI solution is for a data source/sink to determine its latitude-longitude coordinates via GPS and use a prestored database to match the coordinates to a country code. Alternatively, country codes could come preloaded with the operating system.

To support the gateway advertisement/solicitation configuration, **NEI gateways need to be preconfigured**.

A Gateway is typically owned by an organization, and therefore two organizations' network/type-or-protocol internetworks are interconnected by two gateways, one within each organization. An NEI transit-transit gateway is configured with the TransitOrgID of the organization that owns it. There is only one identifier that needs to be configured for the entire gateway, unlike in IP where each interface is configured with a separate IP address and subnet mask. Similarly, an NEI stub-transit gateway within a stub organization is configured with its TransitOrgIDs, and corresponding StubOrgIDs, since such a gateway could be connected to multiple transit organizations.

D. NEI routing tables in data sources and address resolution

Data sources/sinks that are configured with organization internetworking parameters, as described in Section III-C, have an NEI internetworking layer routing table. Unlike an IP routing table, which stores the IP address of the default gateway interface, and then requires ARP, our elimination of the host ID concept at the internetwork layer, simplifies NEI routing tables, and **eliminates ARP**. ARP is required in IP internetworks because host/gateway interfaces are assigned IP addresses independent of their intra-network addresses. NEI's direct usage of intra-org addresses allows for this simplification. In the gateway column, the intra-org address(es) of the gateway(s) is (are) stored.

E. Multihoming and changing transit providers

Multihoming: Consider the multihoming of a data source/sink or a stub organization with multiple transit organizations The name of the data source/sink (domain name of the stub organization) is mapped to a set of several four-tuple vectors by the DNS as indicated in Section III-B. Data sources/sinks (stub

organizations) are addressed within each transit organization to which they are multihomed by `intra-org addresses` (`StubOrgIDs`) that are local to each transit organization. **Inbound traffic engineering** is readily achieved with the set of several four-tuple vectors. The correspondent applications can use these multiple four-tuple vectors for load-balancing and/or switch-over in response to failures.

We do not consider multihoming of a data source/sink in multiple stub organizations, as stub organizations usually provide exclusive networking services for the data sources/sinks within them and do not permit their data sources/sinks to multihome with other stub organizations. The multihoming of a data source/sink in a stub organization and directly in a transit organization appears practically plausible and would be handled analogously to the multihoming discussed above.

An analogy for this answer to handling the multihoming problem with the equivalent of provider aggregatable addressing occurs in the physical world where a individual is reachable via a phone number, email address, and road address. Others trying to reach the individual typically know all of these addresses, and try alternatives if one fails.

Changing transit providers: As noted in Section III-A, NEI addressing limits stub organizations to the equivalent of provider aggregatable (PA) addresses. The question then is what happens when a stub organization changes its transit provider. (We consider a single-homed stub organization in this section; the extension to multi-homing is straightforward.) In IP, the PA addresses for all data sources/sinks, and router interfaces have to be changed. In contrast, in NEI, all that needs to change are the `TransitOrgID` and `StubOrgID` (since the new transit organization may assign a different `StubOrgID`). First, the NEI stub-transit gateways need to be configured with the new `TransitOrgID` and `StubOrgID`. The number of such gateways is likely to be small for most stub organizations. Recall from Section III-C that data sources/sinks that engage in organization internetworking themselves store the `TransitOrgID` and `StubOrgID`. These are updated automatically by having the NEI stub-transit gateways send out gateway advertisements as soon as the change is made in the gateway configuration. As `intra-org addresses` are local to the stub organizations, **no** change is required in these addresses when a stub organizations changes transit providers.

Next, the DNS resource record that maps the stub organization name (e.g. `nctu.edu.tw`) to a `TransitOrgID` and `StubOrgID` in the authoritative DNS server is updated. Given the structure of the DNS resource records, as explained in Section III-B, only this **single record** needs to be changed in the authoritative name server, unlike in IP, where resource records map each host name to a full IP address, and consequently the whole set of records needs updating. Further, as noted in Section III-B, a cache-update protocol is used to flush out stale cached entries in DNS servers. The propagation of these updates is limited to DNS servers within a country (see Section III-G for why this is sufficient), and as cached resource records are also structured in the same manner as in the authoritative server, only the single record corresponding to the organization name needs to be changed. Stale cached entries stored in correspondent data sources/sinks are not flushed out; instead they are handled by coding applications to request authoritative answers if a communication attempt fails for address reachability reasons.

Finally, the resource record for the stub organization in the parent DNS server has to be updated to map the stub organization name to its new set of `TransitOrgIDs` and `StubOrgIDs`.

One final note is to compare this NEI solution with telephone number portability. Recent legislation requires providers to allow customers to keep their same mobile phone number after changing providers. But recall that the telephone number is effectively a "name" not an "address" as users remember/store telephone numbers of other users. In NEI, as in IP, as there is a complementary concept of location-independent names by which data sources/sinks identify each other, there is no necessity to preserve the same address when changing providers, since the cost of preserving addresses (as is possible with provider independent addressing) is the global routing scalability problem. On the flip side, a holistic "design-for-change" approach is taken in NEI to support the change of transit providers by stub organizations with the layout of DNS resource records, support for DNS cache updates, and limiting DNS lookups to countries (as is explained in Section III-G).

F. Global routing scalability

As described in Section III-A, the NEI addressing scheme is carefully constructed to achieve highly scalable global routing. Of the four tuples, only `TransitOrgID` and `CountryCode` are globally unique, while `StubOrgIDs` and `intra-org addresses` are locally unique. The global NEI routing table maintained in an NEI transit-transit gateway only contains next-hop information for `CountryCodes`, and for `TransitOrgIDs` within its own country.

Three key concepts in NEI keep the NEI transit-transit gateway (global) routing tables small. **First**, the use of country codes significantly reduces the size of global routing tables. To see this, consider the example of a new `TransitOrgID` allocated in China. Without `CountryCodes`, the reachability to this `TransitOrgID` needs to be propagated to gateways throughout the world. With `CountryCodes`, the reachability is only propagated to the gateways located in China. Furthermore, the use of country codes helps with security, when coupled with source based packet/call filtering. Given the realities of cyber-warfare, the use of country codes will be useful in the coming years.

Second, by restricting the allocation of `TransitOrgIDs` to only ISPs (i.e., essentially restricting stub organizations to PA addresses), global routing table sizes will be limited. As reported from a measurement study in [5], less than 20% of organizations are transit organizations. One possible consideration is whether small transit organizations could use the `TransitOrgIDs` of their provider transit organizations, thus further limiting the number of in-use `TransitOrgIDs`.

However, this would lead to a cascading effect of changes needed in customer stub organizations if a small transit organization changes its provider. Therefore, as transit organizations, no matter how small, provide communication service to other customers, it is advisable that they obtain and use their own `TransitOrgIDs`.

Third, longest-prefix match is eliminated. NEI transit-transit gateways do routing table lookups on destination `CountryCode` if one is present in the packet header, and destination `TransitOrgIDs` if not, while NEI stub-transit gateways perform lookups on destination `StubOrgIDs` if the first two are absent in packet headers. Our examination of a BGP routing table in a Internet2 router revealed that 45% of the entries had longer-prefix entries, many of which led to the same next-hop node as the shorter prefixes.

However, what are the drawbacks of eliminating longest-prefix match? **First**, support of **mobile** data sources/sinks is affected, as explained in Section III-H. **Second**, if a transit organization operates two separate (disconnected) network/type-or-protocol internetworks, each serving different customers, with IP's longest-prefix match it can obtain one "network ID" and divide this into two "subnetIDs," and report these with BGP. In NEI, this situation requires the transit organization to acquire two separate `TransitOrgIDs`.

A **third** case in which the lack of longest-prefix match hurts arises when large transit organizations connect to each other in multiple locations. With longest prefix matching, routing can be optimized to hand over calls/packets at gateways that are close to the shortest end-to-end route. Without longest-prefix matching, and disallowing the use of `StubOrgIDs` in global routing, a BGP like decision process [6] can cause "detours" in the routing. For example consider a large transit organization (e.g., a US-wide backbone provider) connecting to another large transit organization, such as Internet2 and ESnet connecting to each other. These two organizations interconnect at multiple PoPs; in this example, at Sunnyvale, Seattle, Chicago, New York, and Washington. Consider two enterprises, such as Lawrence Berkeley Laboratory (LBL) in California, and Brookhaven National Laboratory (BNL) in New York, both stub organizations that are customers of ESnet. If longest-prefix match is allowed, ESnet can report longer-prefix reachability to Internet2 for these two stubs. If a packet/call originates at a middle node, say at a Kansas City router in Internet2, this router would then know to route calls for LBL westward within Internet2 to a PoP, such as Seattle, at which it can route the call to ESnet, and in the opposite direction toward Chicago for packets/calls destined to BNL. However, without longest-prefix match and without the use of `StubOrgIDs` in global routing, a BGP-like decision process [6] employing the "eBGP-learned over iBGP-learned" rule (also referred to as "hot-potato routing" where one transit organization moves packets over to another as quickly as possible) and the "lowest IGP cost to border router" rule would route packets/calls from the Kansas City router to the Chicago router for both LBL and BNL destined packets/calls. This level of inefficiency does not appear to be excessive. However, if in the same example, the two transit organizations are not as richly interconnected, the level of inefficiency could be much worse. In order to mitigate this route optimization problem, large transit organizations may acquire two or more `TransitOrgIDs`, e.g., an East Coast one and a West Coast one, in return for more efficient routing.

In Section III-A, the possibility of a country requiring two country codes is mentioned. Above, two cases for a transit organization requiring multiple `TransitOrgIDs` are described. The same can happen with a stub organization. Consider a **stub organization** that has two physically separated network/type-or-protocol internetworks both connected to the same transit organization. This case is simple; the transit organization issues two separate `StubOrgIDs`. Another case is when a large stub organization has multiple physically separated network/type-or-protocol internetworks, which are connected to different transit organizations (perhaps because no single transit organization provider offers service in all its locations). In this case, the different network/type-or-protocol internetworks operate effectively as separate stubs, i.e., data sources/sinks in the different network/type-or-protocol internetworks have their names mapped in DNS resource records to different `TransitOrgIDs` and corresponding `StubOrgIDs`. Such a stub organization can always purchase leased lines or VPN service to interconnect its physically separated network/type-or-protocol internetworks, and then interconnect to one, or possibly two for reliability reasons, transit organizations. The stub organization would have to carefully manage this structure to avoid inefficient routing.

G. NEI protocol stack

Fig. 2 shows the NEI protocol stack for the basic case of organization internetworking in which the organizations operate networks (not type/protocol internetworks). More generally, the gateway shown in Fig. 2 could be a type-internetworking gateway, a protocol-internetworking gateway, or an organization-internetworking gateway if the two organizations being internetworked deploy networks (not type/protocol internetworks). Importantly, this structure should be viewed as a recursive one, in that if the internetwork of networks shown could be interconnected with other internetworks or networks. In this case, there will be multiple InterTL-InterNL protocol pairs layered on top of each other. **Two** key points are important in this NEI protocol stack. **First**, the InterTL and InterNL layers can be bypassed for intra-organization communications, as per the examples of energy-constrained sensors and time-constrained fast moving vehicular nodes in Section III-C. **Second**, through protocol conversion even inter-organization communications can be accomplished without an InterNL layer being executed at the data sources/sinks or gateways. Both cases are shown with the dotted arrows in Fig. 2. In the remainder of this section we consider internetworking with InterNL involvement.

Packet/call handling in NEI gateways: The NEI (InterNL) protocol header contains only the required tuples of the

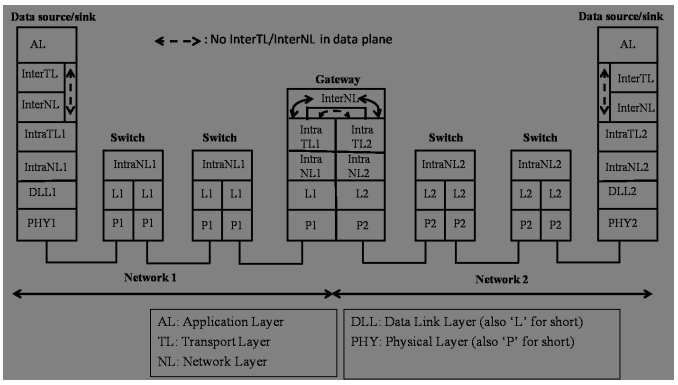


Fig. 2. NEI Protocol Model

destination and source four-tuple vectors. For example, a packet being sent from one stub organization to another within the same transit organization will only carry the source and destination `StubOrgIDs`.

When a data source needs to send a packet to a data sink in another country, (which it knows from the name of the data sink), it puts its own four tuples in the corresponding source fields of the packet, and then maps the data sink's `CountryExtension` to a `CountryCode`, and places this in a destination `CountryCode` field in the packet header. No DNS lookups are allowed for inter-country communications (loose federations of countries can request and be granted a "country code," e.g., EU for the European Union). The reason for this is to limit the extent to which DNS cache updates are required when stub organizations change transit providers. Instead, the **name of the data sink is included in the packet header**. The packet traverses from one NEI gateway to another (stub-transit and transit-transit ones) with each organization gateway looking up its `CountryCode` routing table with the destination `CountryCode` in the packet header to determine the next hop. The destination name carried in the packet header is ignored, until the packet reaches the ingress NEI transit-transit gateway within the destination's country. This gateway issues a DNS lookup to resolve the destination name and obtains the corresponding three tuples, `TransitOrgIDs`, `StubOrgIDs`, and `intra-org addresses`. Next, this gateway strips away the destination `CountryCode` field, and adds three destination fields corresponding to the three tuples. Routing within that country between the various NEI transit-transit gateways proceeds by lookups of the destination `TransitOrgID`, and through an NEI stub-transit gateway by lookups of the destination `StubOrgID`. Return packets are sent with the four-tuple vectors for both source and destination. Therefore, it is only for the first packet in a flow that the gateway at a country's edge needs to execute DNS lookups. The gateway, in its role as DNS client, can cache these mappings. This may appear strange relative to IP networking since a gateway is initiating DNS lookups. Questions of performance for high-speed links need to be considered. However, this operation can be implemented in hardware as such implementations have been demonstrated for more complex protocols [7].

When a data source sends a packet to a data sink in its own country, it creates only the requisite number of source and destination fields based on whether the packet is destined to another stub within its own transit organization or another transit organization. Intra-organization packets are handled with just the `intra-org addresses`.

All four tuples and destination names are variable length, and hence NEI packet headers or NEI signaling messages (for call setup if the networks/internetworks are of the circuit/virtual-circuit type) use the **Tag-Length-Value (TLV)** format. The length field supports variable-length parameters, and the tag field allows for position flexibility for parameter encoding. This is needed for the flexible packet headers described above, which do not always include all four tuples for source and destination, and for inter-country packets which require destination names to be carried. TLV-formatted parameters can be processed in high-speed hardware [7].

H. Mobility

The hierarchical addressing structure in NEI does not cause any new problems than those already encountered and solved in cellular networks. When a mobile connects into a foreign organization's network/type-or-protocol internetwork, it learns the foreign organization's `TransitOrgID` and `StubOrgID`, and is assigned a new `intra-org address`. Meanwhile, the new NEI stub-transit (or transit-transit gateway if it is a directly connected mobile) sends a registration message to its home NEI gateway (transit-transit gateway if it changed transit organizations, and stub-transit gateway if it only changed its stub organization but remained within its transit organization), to create a mapping of its old four-tuple vector to its new one. Packets/calls sent to its old location are forwarded to its new location with packet/call encapsulation (e.g., IP-in-IP tunneling). None of the intermediate gateways are notified as there is no longest-prefix matching. Additional steps to update the correspondent data sources/sinks, and dynamic DNS updates to its authoritative DNS server, should be taken to avoid inefficient packet/call routing.

IV. Related Work

Subsequent to a 2007 IAB workshop on routing and addressing [1], a number of research efforts have sought to develop novel internetworking architectures that support scalable global addressing and routing in conjunction with multi-homing and mobility. The recent workshop [8] reviewed several relevant ongoing research efforts. For instance, a postmodern internet architecture project [9] assigns names to communication channels and employs public/private key pairs for encryption of these names. An economic approach to structuring the future Internet is taken in the value flows project [10], which is based on contracts for communication services and develops the concepts of contract-switching and contract-routing.

Principles for a general recursive internetwork architecture employing an arbitrary number of identical distributed interprocess communication facilities are developed in [11]. Similarly, a metaprotocol for a recursive network architecture is developed in [12].

IPv6 addressing [13] embeds an extended version of the 48-bit MAC addresses as interface identifiers. This may appear similar to our concept of reusing MAC addresses as `intra-org addresses` (see Section III-B). However, NEI differs from IPv6 is many ways, starting with the concept of independent identifiers rather than internetworking addresses for data source/sinks, elimination of ARP and DHCP, limiting stub organizations to effectively provider addressing only, and use of country codes.

Most of the major efforts to handle the global routing scalability problem (LISP [14], shim6 [15], Massey et al. [5], Raj Jain led work [16], Feldman et al.'s work [15]) propose the Location/Identifier separation and the corresponding mapping service assuming that PI addressing is necessary because of the difficulty involved in changing transit providers. This is true in the IP context, but not in NEI due to several factors, all of which work together in a holistic manner to allow for the seemingly draconian rule (that all stub organizations be mandated to use PA addressing) to work. NEI also addresses other problems with IP addressing, not just global routing scalability. Elimination of DHCP and the need to configure gateway interfaces will result in lower operational costs (OPEX).

V. Conclusion

Network Empowered Internetwork (NEI) is an architecture consisting of three types of internetworks: type internetworks, protocol internetworks, and organization internetworks. This paper focuses on the last of these, and addresses problems of global routing scalability, and administrative costs of running IP networks. There are three key ideas: (a) eliminate the concept of internetwork addresses, and instead use a four-tuple vector consisting on unrelated and independently administered components, `CountryCode`, `TransitOrgID`, `StubOrgID`, and `intra-org addresses`, (b) disallow the allocation of global `TransitOrgIDs` to stub organizations (effectively mandating provider addressing for all stub organizations), and (c) handle the issue of multihoming and renumbering needed when stub organization change transit providers with the assistance of DNS. These result in the elimination of some basic mechanisms and protocols used in IP internetworks, such as DHCP, ARP, longest-prefix match, and requiring IP layer for all communications, whether intra- or inter-network.

References

[1] D. Meyer, L. Zhang, and K. Fall, "Report from the IAB Workshop on Routing and Addressing," Internet Engineering Task Force, RFC 4984, Sep. 2007. [Online]. Available: http://www.rfc-editor.org/rfc/rfc4984.txt

[2] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, 5th ed. Addison Wesley, 2010.

[3] J. Day and H. Zimmermann, "The OSI reference model," *Proceedings of the IEEE*, vol. 71, no. 12, pp. 1334–1340, Dec. 1983.

[4] V. Cerf and R. Kahn, "A protocol for packet network intercommunication," *IEEE Trans. on Communications*, vol. 22, no. 5, pp. 353–362, May 1974.

[5] D. Massey, L. Wang, B. Zhang, and L. Zhang, "A scalable routing system design for future internet," SIGCOMM IPv6 Workshop 2007. [Online]. Available: http://www.sigcomm.org/sigcomm2007/ipv6/1569043163.pdf

[6] M. Caesar and J. Rexford, "BGP routing policies in ISP networks," *Network, IEEE*, vol. 19, no. 6, pp. 5–11, nov.-dec. 2005.

[7] H. Wang, M. Veeraraghavan, R. Karri, and T. Li, "Design of a high-performance RSVP-TE hardware signaling accelerator," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 8, pp. 1588–1595, Aug. 2005.

[8] D. Massey, L. Wang, B. Zhang, and L. Zhang, "Report on the Workshop on Future Internet Routing Architecture Designs," Tech. Rep., 2008.

[9] B. Bhattacharjee, K. Calvert, J. Griffioen, N. Spring, and J. Sterbenz. Postmodern internetwork architecture. [Online]. Available: http://www.nets-find.net/Funded/PostModern.php

[10] M. Yuksel, A. Gupta, and S. Kalyanaraman. Value flows and risk management architecture for future Internet. [Online]. Available: http://www.nets-find.net/Funded/ValueFlows.php

[11] J. Day, I. Matta, and K. Mattar, ""Networking is IPC": A guiding principle to a better Internet," in *Proc. of ACM ReArch*, Dec. 2008.

[12] J. Touch and V. K. Pingali, "The RNA metaprotocol," in *Proc. of IEEE ICCCN*, Aug. 2008.

[13] R. Hinden and S. Deering, "IP Version 6 Addressing Architecture," IETF RFC 4291, Feb. 2006.

[14] D. Farinacci, V. Fuller, D. Oran, D. Meyer, and S. Brim, "Locator/ID Separation Protocol (LISP)," Internet Draft, Tech. Rep., 2009.

[15] A. Feldmann, L. Cittadini, W. Mühlbauer, R. Bush, and O. Maennel, "HAIR hierarchical architecture for Internet routing," in *Proc. of ACM ReArch*, Dec. 2009.

[16] S. Paul, R. Jain, and J. Pan, "An identifier/locator split architecture for exploring path diversity through site multihoming—a hybrid host-network cooperative approach," in *Proc. of IEEE ICC*, May 2010.