A strawman proposal for future diverse internets

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

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

***Abstract*—This paper puts forth a straw man proposal for future diverse internets. Guided by a principle that we call *Align-and-Decouple (AD)*, our AD protocol architecture design calls for lightweight transport protocols that allow for decoupled operation of networks that are either owned by different organizations, or are networks of different types/protocols. TCP's fragmentation and congestion control mechanisms modify the original data stream so significantly that all new networks are essentially designed to just carry "TCP transformed" traffic. In contrast, with the AD principle, an AD-API is defined to consist of atomic transfer and pipe communication services, which can be supported by different providers with their own specific networking technologies, and yet allow for internetworking.**

***Index Terms*—future internet architecture; protocols; layering**

I. Introduction

There is a growing interest in designing and evaluating new architectures for future internets [1]. Several problems have been identified with the current Internet architecture, one of which is the difficulty of introducing new communication services, such as IPv6, IP multicast, and IntServ/DiffServ (QoS), as noted by researchers funded through the NSF Future Internet Design (FIND) program [2]–[4].

To address the above problem as well as other limitations with today's internet (e.g., security, global routing scalability), we have proposed a clean-slate future internet design based on a principle which we call **Align-and-Decouple (AD)** [5]. The *align* part refers to making architectural choices that align the internets to real-world administrative entities such as stub organizations (enterprises) and transit organizations (ISPs), while the *decouple* part of the principle is to free individual communication service providers to experiment with new communication services that inspire new applications. It is our hypothesis that internets based on this AD principle will enable the introduction of new services without requiring agreement among service providers on the networking technologies used to support the service. While the align part of the AD principle affects the addressing scheme and security concerns, the decouple part is the focus of this paper.

After introducing a taxonomy for terms such as "networks" and "internetworks" in Section II, we list several example networks in Section III. Our strawman proposal for future internets is outlined in Section IV. The paper is summarized in Section V.

II. Taxonomy

A. Network

We define a **network type** as a unique combination of characteristics on several unrelated dimensions as listed in Table I. For example, we define a network type `A` as consisting of (i) fixed (non-mobile) data sources and sinks interconnected by (ii) wired links and (iii) connectionless packet switches (with no long link disruptions). We define a **network** as follows: (i) it is of a particular network type, (ii) the same protocol stack is used on all its links, and (iii) all its components (data sources/sinks and switches) are owned by the same organization. For example, an enterprise network consisting of data sources/sinks with Ethernet interface cards interconnected by Ethernet switches form a network (i) of the example network type `A` defined above, (ii) run the same protocol (Ethernet) on all its links, and (iii) all its components 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 internetworks:

Type internetwork: The interconnected entities are networks of different network types. An example is an IEEE 802.11 wireless LAN interconnected to a multiple-link Ethernet switched network.

Protocol internetwork: 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. Data centers typically use the InfiniBand protocol for interprocessor communication, but Ethernet for wide area access, making this an example of a protocol internetwork.

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

Organization internetwork: The interconnected entities can be networks, type internetworks, or protocol internetworks. Further if the entities owned by different organizations

978-1-4577-0681-3/11/$26.00 ©2011 IEEE

TABLE I
A FEW DIMENSIONS FOR CLASSIFYING NETWORKS; MORE CAN BE ADDED AS NEEDED

Dimensions	**Set of values**				
1. Switch/MAC type	**Multiple-link switched networks**			**Single-link switchless networks**	
	Connectionless packet switch	Circuit switch	Virtual circuit switch	broadcast-select (random-access)	polling
2. Type of links	Wireless			Wired	
3. Data sources/sinks	Fixed			Mobile	
4. "Long" link disruptions	Do not occur			Do occur	

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.

III. EXAMPLES OF TYPE/PROTOCOL INTERNETWORKS

A diverse set of networks will need to be interconnected in any future internet. In designing transport protocols for future internets, we use current-day networks as examples of entities that need to be interconnected, while ensuring that these protocols are flexible enough to accommodate any new future networks. A few examples of current-day networks are:

1) Wireless sensor networks: Zigbee, IEEE 802.15.4
2) Vehicular networks: IEEE 1609.3 over 802.11p
3) Cellular networks: 3G and 4G (WiMAX and LTE)
4) Wireless LANs: IEEE 802.11
5) Satellite networks and disruption tolerant networks
6) Residential access networks, such as Passive Optical Networks (PONs) and cable (DOCSIS)
7) Data center networks, such as InfiniBand and FibreChannel
8) Dynamic circuit networks such as Internet2 and ESnet's deployments, AT&T's Optical Mesh Service, and Verizon's Bandwidth-on-Demand (BoD) service
9) Supervisory Control and Data Acquisition (SCADA) networks for electric grid, water/sewage, etc.
10) Enterprise Ethernet switched networks
11) Backbone single-link networks, where the links are physical 10/100 GigE links, or logical leased lines provisioned across SONET/SDH or WDM optical circuit networks.

The first five are wireless networks, while the next four are wired, multiple-link switched networks.

IV. A STRAW MAN PROPOSAL FOR FUTURE INTERNETS

To develop transport protocols for operation across different types of internetworks described in Section II, we first need to define an Application Programming Interface (API) offered by future transport protocols to applications. The API is essentially a definition of the communication services. Next, we describe the AD protocol architecture. A principle of least perturbation is put forward to ensure that the internet transport protocols support the decoupling part of the AD principle.

A. API and Services

Ideally, data communication services should include the following two types of services:

- **Atomic transfer service**: A user provides a one-off message or file along with destination names(s) and asks its network to simply deliver this message or file.
  - Parameters of an atomic transfer request: destination(s), message location plus length or file descriptor, confirmed or unconfirmed delivery, deadline (o)[1], security properties (o), mobility-support (o)
- **Pipe service**: A user asks its network to open a pipe to a destination or group of destinations, and leave it open until it asks to close the pipe during which time the user is free to insert data (in the form of messages, files or streamed bits) into the pipe and expect delivery at the specified destination/group of destinations.
  - Parameters used in pipe establishment: destination name(s), uni- or bi-directional, confirmed or unconfirmed delivery, in-sequence delivery or not, security properties (o), traffic characterization (o), rate (o), delay (o) and jitter (delay variance) (o), schedulable-or-not (o), mobility-support (o)
  - Parameters passed down to the pipe after establishment: message location plus length or file descriptor or a bit stream

These communication services are supported through an **Application Programmer Interface (API)**, i.e., a set of functions offered to user applications. We refer to this as the **AD-API**. As several combinations of the parameters are possible, there are a large number of variants of these two basic communication service types. This is an extension of today's Internet offerings of TCP sockets (reliable pipe or atomic service with none of the other options listed above), real-time transport protocol API (for streamed A/V traffic), and UDP and raw IP sockets (both atomic unreliable services).

B. AD protocol architecture

Fig. 1 illustrates the AD-internet protocol stack. As per the network classification of Table I, some networks may be single-link switchless networks, while others may be multiple-link switched networks. For intra-network communication across single-link networks all that is required are the application layer, AD-API, Data-Link Layer (DLL) and physical

[1] "(o)" is used to indicate optional parameters

978-1-4577-0681-3/11/$26.00 ©2011 IEEE

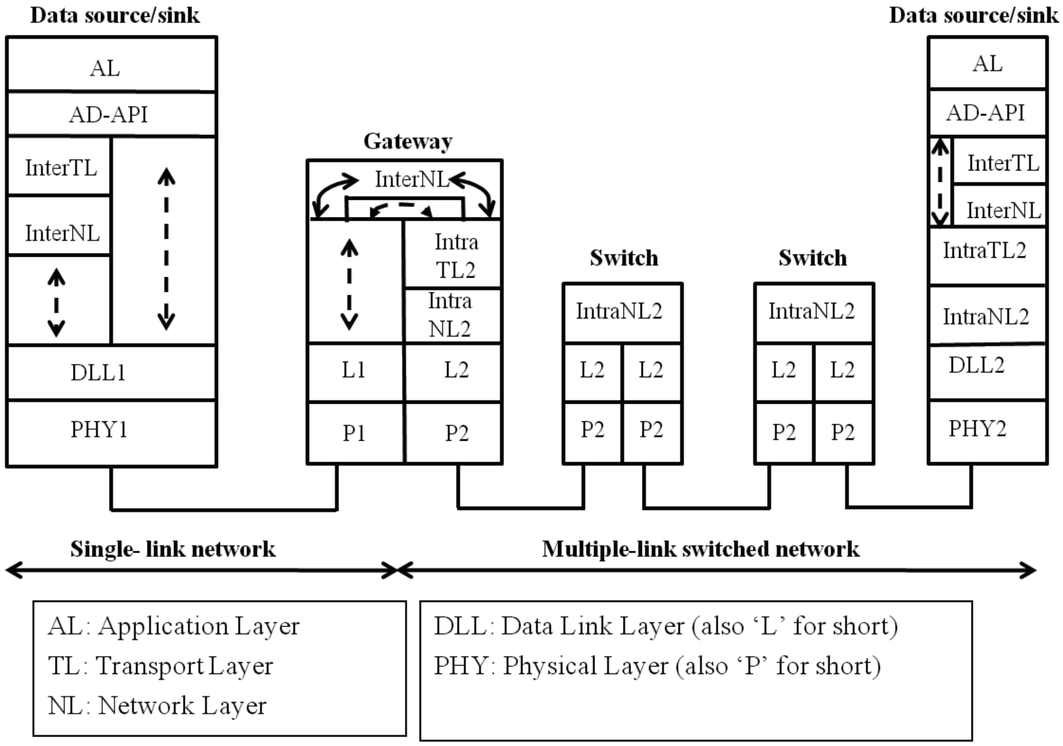

Fig. 1. AD internet protocol architecture

layer (PHY). A network can choose to support some subset of the communication services listed in Section IV-A. If a user of this network wants to communicate with the user of another network, e.g., one located on the multiple-link switched network shown in Fig. 1, then the Inter-Transport Layer (Inter-TL) and Inter-Network Layer (Inter-NL) protocols are involved. As per the parameters listed for the atomic and pipe services in Section IV-A, messages/files in the atomic service and pipe establishment requests are passed from applications down to the AD-API layer shown in Fig. 1 with destination name(s). The AD-API executes the name-to-address translation function. An addressing scheme for AD internets is described in [5]. Based on whether the destination is in the same network or across an internetwork, the AD-API issues the correct primitives to communicate directly with the DLL (for intra-network communications within a single-link network), Intra-TL (for intra-network communications within a multiple-link switched network), or to the Inter-TL (for inter-network communications). These are illustrated with the dashed arrows within the data sources/sinks.

The gateway shown in Fig. 1 is an example of an AD gateway that interconnects a single-link network with a multiple-link switched network. A gateway can have the capability to connect many different networks, each of which could be a single-link or multiple-link switched network. It would need to run the DLL/PHY layers of single-link networks, and the four-layer Intra-TL, Intra-NL, DLL and PHY of the multiple-link networks to which it connects.

Consider the dashed line shown within the gateway in Fig. 1 that effectively allows for bypassing the Inter-NL. This is feasible for the AD-API pipe service, whereby in the pipe establishment phase, a mapping of DLL/transport layer ports can be stored in the gateway[2]. We provide two examples. The InfiniBand transport protocol uses “queue pairs” that are similar to TCP destination and source ports. If TCP were the transport protocol run over say an Ethernet switched network on one side of a gateway with InfiniBand on the other side, TCP port numbers could be mapped to Infiniband queue pairs. Another example occurs in vehicular networks. The IEEE 1609.3 standard defines a technique for mapping a Wave Short Message Protocol (WSMP) Provider Service Identifier (PSID) to a UDP/IP port/address pair within a server connected to a road-side unit allowing for an internetworking of WSMP over 802.11p with the Internet without requiring the vehicular onboard units to themselves run UDP/IP.

C. *Principles governing AD internetworking*

The decoupling aspect of the Align-and-Decouple (AD) principle introduced in Section I requires a supporting principle that we call Principle of least perturbation, which is described below.

Network technologies: Networks of different types, for example, `datagram` networks, packet-switched networks with

[2]The IEEE 802.2 Logical Link Control (LLC) sublayer of the DLL supports DSAPs/SSAPs (Destination and Source Service Access Points) that serve the same purpose as TCP/UDP ports; however this particular protocol is limited by legacy assignments of SAPs. A comparable concept can be implemented in future DLLs of single-link switchless networks.

978-1-4577-0681-3/11/$26.00 ©2011 IEEE

`priority queueing`, `circuit` networks or `virtual circuit (VC)` networks, can support multiple variants of the atomic and pipe transfer services with clever engineering ideas. For example, at first glance, a datagram network seems ill-suited for a continuous audio/video data stream. But this is done today, using the simple techniques of overprovisioning and adaptive buffering at the receiver to smooth out jitter. The implication of this argument is that a provider has some options to enable a new communication service across its existing network without changing network technology.

Principle of least perturbation: Given our decoupling principle, each organization can choose its own network technologies to support some subset of the large number of variants of the AD-API/communication services. When an organization's network/TP-internetwork is a transit network on an end-to-end path, it participates in pipe establishment during which it is provided a set of optional parameters as specified by the data source. When data starts flowing down the pipe, the upstream organization's gateway should restore the traffic patterns to be as close as possible to those specified during pipe establishment. Effectively, this leads to our principle of least perturbation, which states that the internetwork transport and network layers should be such that the traffic delivered by each gateway to the next organization network/TP-internetwork is in a format as close as possible to that offered by the data source to the first organization network/TP-internetwork on the path. TCP modifies the data source's traffic pattern considerably because of its congestion control and maximum segment size based fragmentation mechanisms.

Decoupling enables independent service introduction: Using the above approach, not all transit organizations will want to support a new communication service, potentially because their existing network technologies do not allow it and the cost of upgrades is not justifiable for business reasons. This just limits the reachable set of destinations for the customers of an organization that deploys a new communication service. An analogy is that US customers of Fedex cannot use its deadline-based delivery service to reach destinations in a country where there is no such delivery service with whom Fedex can establish a partnership. In AD internets, each organization will be able to choose its own set of technologies, and the communication services that it supports. When a new communication service is offered by an organization to its customers, the **reachable set of destinations** parameter is limited to those that can be reached via other organizations that offer the same service without requiring that the other organizations deploy the same networking technologies. This mitigates a difficulty noted by Turner in [4] that introducing new capabilities into the Internet is difficult because "universal agreement" is required before such capabilities can be deployed. Decoupling removes the need for such agreements on particular technologies to realize a new communication service.

The cost of such decoupling is that gateways will need disks and more memory buffers than current-day IP routers. At the simplest level, if two networks on an internetwork path have different available bandwidth, with TCP, the ACK self-clocking will essentially slow the traffic on the entire end-to-end path. We argue that this is not always desirable, as a network may operate a different bandwidth sharing scheme in which its preference is to move the data for the flow at high speed across its network, and free up its bandwidth for other transfers. This is not possible without storage in the gateways. With decreasing disk storage costs ("Kryder's Law" [6]), and improving speeds of access, adding disks to gateways seems like a natural extension to the current design of allowing memory buffers in IP routers.

V. Summary

One of the problems identified with today's Internet is the difficulty in introducing new communication services. This paper argues that with lightweight internetwork transport protocols that do not modify the original traffic patterns, individual networks can develop their own mechanisms to support new services on their existing networks, while at the same time allowing to be internetworked to offer these services between their customers and those of other providers.

VI. Acknowledgment

The University of Virginia portion of this work was supported by the DOE grant DE-SC002350 and NSF grant 1038058. The authors thank Robert D. Russell, University of New Hampshire, for his valuable comments.

References

[1] D. Clark, "NSF Future Internet Summit Washington, DC, October 12–15, 2009, Meeting Summary Version 7.0," 2010. [Online]. Available: http://www.nets-find.net/Summit-summary-7.pdf", month = January 5
[2] N. Feamster, L. Gao, and J. Rexford. Cabo: Concurrent architectures are better than one. [Online]. Available: http://www.nets-find.net/Funded/Cabo.php
[3] D. Massey, L. Wang, B. Zhang, and L. Zhang. Enabling future Internet innovations through transitwire (efit). [Online]. Available: http://www.nets-find.net/Funded/eFIT.php
[4] J. Turner, P. Crowley, S. Gorinsky, and J. Lockwood. An architecture for a diversified Internet. [Online]. Available: http://www.nets-find.net/Funded/DiversifiedInternet.php
[5] M. Veeraraghavan, M. Reisslein, J. Li, and R. D. Williams, "A clean-slate align-and-decouple (AD) addressing architecture for future internets," Tech. Rep., submitted to IEEE ICCCN. [Online]. Available: http://mre.faculty.asu.edu/NEI_ov_ext.pdf
[6] "Kryder's law," scientific American, Aug. 2005. [Online]. Available: http://www.scientificamerican.com/article.cfm?id=kryders-law