G E N I
Global Environment for Network Innovations

Overcoming Barriers to Disruptive Innovation in Networking

Thomas Anderson
University of Washington

Larry Peterson
Princeton University

Scott Shenker
UC Berkeley

Jonathan Turner
Washington University

Editors

GDD-05-02

January 2005

Appears as NSF Workshop Report (January 2005)

Status: Final

Report of NSF Workshop on

Overcoming Barriers to Disruptive Innovation in Networking

January 2005

The NSF Workshop on Overcoming Barriers to Disruptive Innovation in Network was supported by NSF under grant CNS-0439842.

Table of Contents

1. Executive Summary

There is little argument that the Internet faces many challenges, including both correcting vulnerabilities that arise from society's increasing dependence on it [PRE05], and capitalizing on opportunities that arise as new applications. It is critical that the network research community be engaged in addressing these challenges.

The research community typically pursues one of two paths when trying to affect the Internet. The first is to incrementally evolve the network to address new vulnerabilities and opportunities as they occur. The research community, in conjunction with the commercial players that define today's Internet, have successfully followed this path for nearly 30 years, resulting in point-solutions of narrow scope, many of which step outside the original Internet architecture. The second path is to create a new Internet architecture that better addresses the many challenges on the horizon. This approach potentially involves a clean-slate design, and so is likely disruptive.

While there is no way to be certain that the incremental path will ultimately fail to address the challenges facing the Internet, there are two reasons to be concerned. The first is that the point-solutions incrementally applied to the Internet result in increased complexity. The Internet's once clean architecture has become muddied by patches, which makes it hard to reason about the network as a whole. This increased complexity makes the Internet harder to manage, more brittle in the face of new requirements, and more vulnerable to emerging threats. The second is that are architectural limits that may eventually result in a dead-end for the current incremental path. This report identifies five such limits, which we express in actionable terms:

1. **Minimize trust assumptions:** the Internet originally viewed network traffic as fundamentally friendly, but should view it as adversarial;
2. **Enable user choice:** the Internet was originally developed independent of any commercial considerations, but today the network architecture must take competition and economic incentives into account;
3. **Allow for edge diversity:** the Internet originally assumed host computers were connected to the edges of the network, but host-centric assumptions are not appropriate in a world with an increasing number of sensors and mobile devices;
4. **Design for network transparency:** the Internet originally did not expose information about its internal configuration, but there is value to both users and network administrators in making the network more transparent; and
5. **Meet application requirements:** the Internet originally provided only a best-effort packet delivery service, but there is value in enhancing (adding functionality to) the network to meet application requirements.

Considering the risks of solely pursuing the incremental path, the workshop participants believe it is important that the research community also pursue the design, evaluation, and deployment of disruptive network architectures. This path is not without its own risks, however. First, researchers need more realistic evaluations of architectural proposals. New architectures need to be evaluated experimentally, operating at scale, and under real-world

conditions. Second, there must be a plausible deployment plan for any new architecture. Expecting global agreement about (and uptake of) a new network architecture is not realistic in an environment dominated by commercial considerations.

Despite these risks, there is a new approach to experimental network testbeds that both permit realistic experimental evaluations and have the potential to lead to wide-spread deployment. The key features of the new approach include (1) an overlay infrastructure with global reach that can be shared among multiple candidate network architectures; (2) interposition mechanisms that allow users to opt-into new architectures on a per-user/per-application basis, thereby providing real user traffic and facilitating incremental deployment; and (3) a high-performance substrate that provides sufficient capacity to make successful architectures viable on a larger scale.

In light of the current situation and opportunities, the workshop participants make the following recommendations to the National Science Foundation:

Recommendation 1: Immediately initiate a research program on experimental architectural research in networking. If successful, the potential benefits are enormous, easily justifying the modest initial outlay. A dedicated research program will pull together the research community to tackle the broad scope of thorny architectural questions that we have outlined in this report, and that must be addressed for the program to be successful.

Recommendation 2: Foster experimental validation of new architectural research in networking. Paper designs, although thought provoking, are unconvincing, both to the companies that need to adopt them, and to the research community in evaluating ideas and in gaining insight into design tradeoffs. Thus, to maximize our chance of success, NSF must foster an expectation within the experimental architectural research program that research ideas should normally be validated under real use

Recommendation 3: Fund the development and deployment of suitable testbeds. Since experimental validation is an important component of this research program, it is essential that researchers have access to suitable testbeds NSF should therefore endeavor to build a meta-testbed that reduces the barrier to entry for new architectural ideas. To meet short-term needs, NSF should support an initial meta-testbed that can be deployed immediately. At the same time, NSF should initiate a deliberative process through which the community can identify long-term solutions to its meta-testbed requirements.

Recommendation 4: Start a process that will lead to substantial increases in funding for a broad multi-disciplinary effort in this area over the next few years. To design, construct and widely deploy a new architecture for the Internet is an enormously difficult and, at the same time, an enormously important undertaking. To be successful, we will need to enlist the efforts of distributed systems researchers, e-scientists, application developers, computer architects, and network hardware technologists.

Recommendation 5: Find ways to promote synergy and convergence among architectural visions. Academic research focuses on novelty and, in so doing, often accentuates

differences rather than identifying commonality. The past success of the Internet strongly suggests that we will be the most successful if we can coalesce around common architecture features. Architecture, by its very nature, "defines that on which we must agree." Thus, to be effective, architectural researchers should seek convergence rather than divergence.

Recommendation 6: Help the community learn from industry. Disruptive architectural research should not be fettered by today's problems and practices, but it must be informed by them if we are not to simply repeat the mistakes of the past. The large gap between the research and commercial communities often prevents effective communication between the two, to the detriment of both. To bridge this gap, NSF should facilitate interactions between researchers and practitioners.

The workshop participants recognize that different outcomes are possible. One possibility is that multiple promising architectures bloom, but that over time, there is convergence on a single new architecture for the Internet. Ideally, the incremental deployment story proves successful, bringing the new architecture to the verge of commercialization. Another possibility is that many valid architectures emerge, but there is no consensus as to a single correct architecture. Instead, the experimental testbed that supports multiple architectures emerges as the substrate for a future global communications infrastructure. A third possible outcome is that the ideas developed as a part of this program provide new insights and architectural clarity, but these ideas can be incrementally retrofitted into today's Internet architecture. This possibility suggests that pursuing the second path (a disruptive architecture) actually improves the odds that the first path (incremental evolution) succeeds.

2. Problems, Opportunities, and the Impact Imperative

The Internet has, in a remarkably short period of time, radically transformed the world's information infrastructure. This success is in no small part due to its innovative architecture that, in several dimensions, broke with the conventional (and largely telephonic) wisdom. The architecture now accommodates a wide variety of network technologies, spans an enormous gamut of speeds, supports a broad range of applications, withstands a substantial number of failures, and scales to hundreds of millions of nodes. Moreover, the same architecture that facilitated organic and decentralized growth during the Internet's formative years has endured, without modification, the painful transition to a commercial enterprise with many competing providers. In both technical and commercial terms, the Internet architecture has succeeded beyond anyone's wildest dreams.

However, in the thirty-odd years since its invention, new uses and abuses, along with the realities that come with being a fully commercial enterprise, are pushing the Internet into realms that its original design neither anticipated nor easily accommodates. These problematic issues include: the awkwardness with which host mobility, host multi-homing, data migration and data replication are handled; the lack of protection from unwanted or harmful traffic; the increasing complexity and fragility of inter-domain routing; and the impact of radically diverse edge devices, including sensor networks. Such problems are numerous, and the Internet's emerging centrality has made these flaws all the more evident and urgent. As a result, it is now widely believed that the Internet architecture is in need of substantial change.

Unfortunately, there is increasing pessimism about the possibility of change. Adopting a new architecture not only requires modifications to routers and host software, but given the multi-provider nature of the Internet, also requires that ISPs jointly agree on that architecture. The need for consensus is doubly damning; not only is agreement among the many providers hard to reach, it also removes any competitive advantage from architectural innovation. This discouraging combination of difficulty reaching consensus, lack of incentives for deployment, and substantial costs of upgrading the infrastructure leaves little hope for fundamental architectural change. Thus, many believe that the Internet architecture, which began as a radical experiment, has now ossified into an unalterable status quo [PET04].

Freezing forevermore the current architecture would be bad enough, but in fact the situation is deteriorating. The inability to adapt to new pressures and requirements has led to an increasing number of ad hoc work-arounds, many of which violate the canonical architecture (e.g., middleboxes). While derided by architectural purists, these modifications have (usually) arisen to meet legitimate needs that the architecture itself could not. These architectural barnacles—unsightly outcroppings that have affixed themselves to an unmoving architecture—may serve a valuable short-term purpose, but significantly impair the long-term flexibility, reliability, security, and manageability of the Internet. Thus, the collision between the improbability and the necessity of change has resulted in expedient but eventually harmful architectural liberties.

While the commercial world applies point-solutions and work-arounds to the existing Internet, the research community is facing its own dilemma. A network architecture is a subtle thing that defies rigorous analysis or satisfying simulation, and is best understood through extensive live experimentation. However, current testbed paradigms are inadequate to this task. Traditional testbeds can be roughly categorized as production-oriented or research-oriented. Production testbeds, such as Internet2 [I2], support real traffic from real users, often in large volume and across many sites. As such, they provide valuable information about the operational behavior of an architecture. However, the users of such a production testbed have no choice about whether or not to participate in the testbed and usually do not even realize that their traffic is part of an experiment. They thus expect the performance and reliability to be no worse than the standard Internet. Production testbeds must therefore be extremely conservative in their experimentation, using well-honed implementations of incremental changes.

Research testbeds (such as DETER [DET]) do not carry traffic from a wide variety of real users but instead are typically driven by synthetically generated traffic and/or a small collection of intrepid users. This allows them to be much more adventurous, capable of running first-cut implementations of radically new designs. Unfortunately, this lack of real traffic also renders the results much less indicative of real operational viability. As a result, neither kind of testbed—production or research—produces the data needed to adequately evaluate new architectures. It is therefore difficult to make a compelling case for new architectural designs based on a testbed evaluation. In addition, because they utilize dedicated transmission links, both categories of testbeds involve substantial cost, and so are prohibitively expensive to operate at very large scale. Thus, they are typically of small geographic extent and arise only with substantial funding support. Given the limitations mentioned above, traditional testbeds offer far too little bang for their buck, and clearly cannot lead us into the future.

The preceding paints a depressing picture of the status quo, with an architecture incapable of change and a research community unable to validate its designs. However, within this bleakness there are seeds of hope. After roughly a decade where incremental research held

sway, there has been a resurgence of interest in more fundamental architectural questions. This architectural research is still at an early stage and needs significantly more support in order to reach fruition, but these initial architectural sprouts are very encouraging. While not providing any definitive solutions, they suggest that many of the challenges facing the Internet can be adequately addressed through architectural innovations.

In addition, there is now a promising alternative to the traditional testbed approach. Two recent trends, virtualization and overlay networks, can be combined to create effective and inexpensive testbeds. Overlay networks have often been used to augment the current Internet and deploy experimental designs. Overlay networks, in contrast to the traditional physical testbeds, are not limited geographically: in fact, overlay networks can be accessed by any user through packet-redirection implemented by host proxies. The decision about whether or not to use an overlay network can be made on a per-user, and even a per-application, basis. If the overlay network fails, the user's traffic can default back to normal Internet service. The lack of geographic limitations, the ability of fine-grained opt-in, and the presence of automatic fail-over suggest that experimental architectures could likely attract a sizable pool of volunteers willing to supply live traffic. This breaks the old dichotomy of experimental versus production testbeds; these overlays can now be both.

Moreover, overlay networks don't require significant investment in bandwidth. However, such networks require a great deal of effort to deploy and manage, and this overhead of deploying a single-purpose overlay is well beyond the means of most researchers. Fortunately, the advent of highly virtualized infrastructures, like PlanetLab [PET02, BA04], provides a solution to this problem. Virtualization allows each overlay node to emulate the actions of many logical "routers," and thereby enables such infrastructures to support many concurrent architecture experiments, each running on its own set of logical routers. The burden of running an overlay is thus shared among a large set of experiments, bringing the overhead imposed on any individual researcher to a much more manageable level. Thus, these virtualized testbeds offer new hope that large-scale live experimentation with new architectures is within reach of most researchers.

All such experimentation would be meaningless without a plausible deployment path. As argued earlier, the need for consensus and the consequent lack of competitive advantage, along with the sizable investment needed to upgrade the deployed infrastructure, makes it doubtful that the current ISPs will deploy a next-generation architecture. Thus, deployment of new architectures may rely on new entrants to the service provision market. Given the high capital costs and low operating margins of this industry, a market-entering foray by a traditional infrastructure-based ISP seems unlikely. Overlays, however, are a more cost-effective way to enter this market. A new-generation service provider could deploy an overlay supporting a new architecture and distribute proxy software that allows anyone, anywhere, to access that overlay. This deployment path would be further enhanced by a highly virtualized overlay infrastructure. Just as commercial web hosting facilities allow individual companies to easily establish production-grade web sites, a commercial overlay hosting facility could greatly lower the barrier facing entering service providers. In fact, this virtualized infrastructure need not be an overlay and could instead be based on a set of dedicated links and (virtualized) routers. As we discuss later, this would be especially relevant if a sizable market for the development and deployment of new architectures (and infrastructure-based services) develops.

While the status quo is good reason for pessimism, the new developments described above provide much hope for the future. There is growing interest in new architectural approaches,

and some of the early results are promising. Virtualized overlay infrastructures can allow extensive yet inexpensive live experimentation with these new designs, and eventual deployment may proceed through the same virtualization approach. Thus, the seeds for success are already present.

This opportunity will not be realized easily. The research community must rally around the grand challenge of designing new network architectures and following them through to deployment. This is no small task. Not only will it require abundant time and effort, it will also require a change in the community's culture. Researchers must move beyond merely academic models of success and rededicate themselves to making an impact.

Many in the community already feel this impact imperative. But they will require substantial support in order to succeed. A greater focus on architectural research would broaden the pool of interested designers and interesting designs. This endeavor will also require a greater focus on impact and a recognition of the nature of support such efforts require.

3. Challenges and Options for Meeting Them

It is clear that the Internet faces serious challenges, from improving the security and robustness of its core packet delivery service, to accommodating an explosion in the number and diversity of devices that connect to it, to enabling a new generation of applications. While a perfectly valid response to this situation is to identify the attributes an ideal Internet of 2015 might aspire to [CL05], the research community believes it is also important to re-evaluate the architectural decisions that underlie today's Internet. This research agenda involves identifying the key limitations and assumption of the current architecture and pursuing the opportunities made possible by removing these barriers, with the goal of converging on a new set of architectural features that provide the foundation for the global communications infrastructure. While there is also value in doing research that leads to incremental improvement of today's Internet, these architectural barriers must be taken head-on to fully address the challenges we face.

This section identifies seven specific architectural limitations or assumptions that the research community believes warrant investigation. The following subsections do not correspond to seven different network architectures, but rather, they identify "vectors" for architectural research that the community is already pursuing. Note that these vectors are not orthogonal; they revolve around five themes:

1. **Minimizing trust assumptions:** the Internet originally viewed network traffic as fundamentally friendly, but should view it as adversarial;
2. **Enabling user choice:** the Internet was originally developed independent of any commercial considerations, but today the network architecture must take competition and economic incentives into account;
3. **Allowing for edge diversity:** the Internet originally assumed host computers were connected to the edges of the network, but host-centric assumptions are not appropriate in a world with an increasing number of sensors and mobile devices;

4. **Designing for network transparency:** the Internet originally did not expose information about its internal configuration, but there is value to both users and network administrators in making the network more transparent; and
5. **Meeting application requirements:** the Internet originally provided only a best-effort packet delivery service, but there is value in enhancing (adding functionality to) the network to meet application requirements.

3.1. Security

Unlike the original Internet, in which the user community was a close-knit group of experts running relatively simple applications, today's user population and applications increasingly means that network traffic must be viewed as adversarial rather than cooperative. This fundamental shift makes security a major concern. In particular, the scale and heterogeneity of the network has increased dramatically to span scores of nations, thousands of network providers, and millions of users. Unfortunately, few of today's protocols are designed to minimize trust or even to recognize trust boundaries. To take one example, a single mistyped command at a router at one ISP recently caused widespread, cascading disruption of Internet connectivity across many of its neighbors. At the same time, a broad range of applications-including critical infrastructure, commerce, education, personal productivity-now depend on the Internet infrastructure. This raises both the incentives for malicious users and the consequences of successful attacks. Because of the Internet's ossification, any new security flaw in a protocol can take decades to address, handing malicious attackers a significant advantage.

Fundamentally changing the Internet architecture to assume adversarial rather than friendly use has the potential to yield dramatic benefits. Imagine, for example, a world where the Internet is a trustworthy network absent of attacks, where sensitive information is communicated safely, where corporations can rely on the Internet for their businesses without fear of disruption, and where governments can rely on it for their critical infrastructures.

Given the paramount importance of the Internet, a security-aware architecture that minimizes trust assumptions is necessary. For example, architectural support for security could (1) improve network robustness through protocols that work despite misbehaving participants, (2) enable security problems to be addressed quickly once identified, (3) isolate ISPs, organizations, and users from inadvertent errors or attacks; (4) prevent epidemic-style attacks such as worms, viruses, and distributed denial of service; (5) enable or simplify deployment of new high-value applications and critical services that rely on Internet communication such as power grid control, on-line trading networks, or an Internet emergency communication channel; and (6) reduce lost productivity currently aimed at coping with security problems via patching holes, recovering from attacks, or identifying attackers.

Several architectural approaches show promise for addressing security issues. One important thread are architectures that prevent denial of service by allowing a receiver to control who can send packets to it. Another is making firewalls a fully recognized component of the architecture instead of an add-on that is either turned off or gets in the way of deploying new applications. A clean specification for security that makes clear the balance of responsibility for routers, for operating systems and for applications can move us from the hodge-podge of security building blocks we have today to a real security architecture. A careful design of mechanisms for identity can balance, in an intentional way rather than by accident, the goals of privacy and accountability. Ideally, the design will permit us to apply real world

consequences (e.g. legal or financial) for misbehavior. This may require that the architecture be aware of such real-world attributes as boundaries of jurisdiction.

The main research challenges in defining a more secure network architecture include balancing accountability versus privacy, balancing processing overheads versus security guarantees, and determining what network information and processing to expose to the network infrastructure and to network users.

3.2. Economic Incentives

The original design of the Internet did not take into account the economic structure of the industry that would emerge to support it. The very early view of the Internet was an undifferentiated cloud of routers, with no recognition of the points where Internet service providers connect. In contrast to the telephone system, which has two kinds of phone calls: sender pays and receiver pays ("800" calls), the Internet has no equivalent of a call, and nothing to signal the direction of value flow. This lack of attention to value flow, and architectural mechanisms to underlie the flow of payments across the Internet, represents a barrier to future investment in the Internet, and a barrier to the overall economic health of the infrastructure sector. While many mechanisms have emerged in response to industry needs and in particular to the problem of bilateral connection among ISPs, it can be argued that lack of an overall architectural framework for flow of payments has hindered the deployment of inter-provider Quality of Service, of multicast, and of consumer broadband. A failure to attend to larger economic issues around the competitive nature of the industry structure can also be seen as one of the causes of poor security in the Internet, and the failure of the Internet to address larger social needs (public sector needs) such as emergency preparedness.

A future design for an Internet should take into account that a network architecture induces an industry structure, and the economic structure of that industry. The architecture can use user choice (to impose the discipline of competition on the players), indications of value flow (to make explicit the right direction of payment flow), and careful attention to what information is revealed and what is kept hidden (to shape the nature of transactions across a competitive boundary). The "architecture of economics" surrounding a new Internet must also reflect the necessity of governments to inject into the design functional objectives that do not necessarily align with the features that emerge through private sector, profit-seeking investment.

3.3. Address Binding

The way in which endpoints are identified for the purpose of directing traffic toward them is one of the most fundamental aspects of any network architecture. In today's Internet, endpoints are addressed with topologically-dependent IP addresses, and it is precisely the structured nature of the address space that enables scalable packet forwarding.

More precisely, endpoints in the Internet architecture are simply network attachment points—locations where a network device plugs in. IP addresses were not intended to say anything about the machine connected at that point. Unfortunately, due to the tight coupling between IP addresses and end hosts during the initial 20 years of Internet deployment—i.e., one of the unstated assumptions has been that machines rarely moved between attachment points, and attachment points were rarely shared between machines—IP addresses were reused as host

identifiers—that is, an IP address came to be far more than an ephemeral routing locator: *it was a machine's identity*.

A critical issue is that the use of IP addresses as end host identifiers creates a number of problems when end hosts move. Mobile IP [PER02] was developed to address this exact problem: it allowed machines to take their IP address with them, but resulted in an inefficient forwarding mechanism. A number of systems have proposed efficient mechanisms for intercepting packets near their source and forwarding them to a mobile host's current location. However, the Internet architecture effectively limits us to intercepting packets at a mobile host's home. Similarly, the design of a network that assumes that most hosts are mobile may be grossly different from either the current Internet architecture or any existing proposal. The deployment of more efficient mobile host support might greatly change the way that typical portable devices (e.g. laptops, PDA, cell phones) connect to the Internet.

In the interest of expediency, today's end hosts change IP addresses each time they move. However, operators have implemented policies that make implicit assumptions about the machines using particular IP addresses. For example, IP addresses are often used to specify security and access policies as in the case of ingress filtering to alleviate denial-of-service attacks [FE98]. To the extent that IP addresses change, we loose any identity or accountability that might have been keyed to addresses.

The current reality, then, is a mess: Internet addressing is neither secure, efficient, nor architecturally clean. Further, neither end hosts nor network attachment points are sufficient to describe the end points in today's Internet. Depending on the situation, end points may be applications (that may move between machines), sessions (that may move between applications), users (that may move between applications and machines), or data (that can exist almost anywhere).

A new architecture needs to remove the coupling between topological location and endpoint identity present in IP addresses. One proposal is that endpoints need to be given a topology independent identifier, and routing and addressing cannot depend on the identity of the endpoints. A number of proposals have explored possible avenues, each with their own strengths and weaknesses. The Host Identity Protocol (HIP) [MO05] provides each end host with a cryptographically secure identifier, which can be used to anchor transport end points, as well as input to security policies. Routing and addressing continues to be performed by traditional IP, but IP addresses are treated only as ephemeral locators. Another architectural possibility is that end-points (as equated with physical machines or operating systems) need not have any globally known identity at all. Instead, application level entities have shared identities that they use to confirm each end to the other, and higher level name spaces such as a re-designed DNS are used to give global names to services, so that they can be found.

While it's not yet clear exactly what the properties of an endpoint identifier should be, nor precisely what constitutes an endpoint, it is clear that IP addresses and network interfaces are not the right abstractions. A complete redesign of the architecture for location, global naming and shared identity will enhance the security, efficiency, ease-of-use, and flexibility of the basic forwarding infrastructure.

3.4. End Host Assumptions

The current Internet architecture makes several assumptions about the hosts that connect to its edge—that they are usually connected, that they do not move very often, that they are best

identified by relatively static names/addresses rather than more dynamic properties, and so on. This has made it difficult to incorporate devices such as sensor motes or functions such as delay tolerant network routing directly into the Internet infrastructure. Most commonly, intermediate nodes—e.g., sensor base stations, proxies, and home agents—are used to allow these devices to participate on the Internet. Unfortunately, this translation typically incurs some loss in functionality and performance.

Internet routing is based on destination address. But sensor nets often route data based on its value. Algorithms such as diffusion routing are used to build data-driven routing patterns that allow for an application-specific integrated pattern of routing and processing. Te extend sensor nets across the Internet, what is needed is support for a *sensor overlay* that allows a set of agents to recreate schemes such as diffusion routing on top of Internet connectivity. By simplifying the direct attachment of sensor networks to the Internet, we could enable a global-scale mesh of sensor networks, thus supporting a wide-variety of natural science research. In addition to the associated naming and routing architectural changes, a global sensor mesh may also require new security infrastructure. For example, while access to current sensor networks is limited by physical proximity, a global mesh of sensor networks would require enforcement of policies for access to and use of the collected sensor data.

Another limitation of the current architecture is the assumption that nodes are connected in a way that permits near instantaneous communications. Staged or delayed delivery is a part of some applications, such as email, but is not recognized as a problem at the Internet level. The correct solution to this requirement may involve a *delay tolerant* overlay, of the sort being developed by the DTN project. But it is also possible that the Internet architecture itself should better take account of nodes or regions that are poorly and intermittently connected. The deployment of transport and routing protocols that support such long-delay links would enable Internet access in a variety of impoverished and poorly connected regions. Unfortunately, the best design for supporting high delay links remains an open issue. In addition, it is unclear whether a common suite of the algorithms, protocols, and applications could support both interactive and delay-tolerant operation.

3.5. User-Level Route Choice

The current Internet architecture performs routing in an opaque manner that does not allow users control over the paths taken by traffic to and from them. In this context, a *user* could be an actual human, an application program, their Internet provider, or even an overlay service running on their behalf. This limitation restricts several desirable goals. For example, a user cannot express the choice of their ISP beyond their selection of an access provider, or direct traffic along links that have higher availability than the default path.

Relaxing this restriction provides several potential benefits, both technical and economic. Because users know whether or not a particular path is actually working for them, choosing between multiple paths in the network based upon whether they are functional can lead to improved availability and performance. Such path selection can create an enhanced competitive landscape by allowing users to easily switch between packet carriers based on their performance, cost, or availability, a choice that does not exist today. Permitting users to express their routing preferences in a more fine-grained manner may permit ISPs to offer increased service differentiation: Instead of applying a "one-size-fits-all" policy to their traffic, ISPs could perform routing and traffic engineering based upon the user traffic preferences in addition to

their own metrics and policies, or even offer unique policies such as keeping all traffic within the continental United States for security reasons.

Permitting users to control their routes opens a variety of issues. Foremost among these is resolving the conflicts between the preferences of multiple users and of the ISPs who carry their traffic. It is important that the architecture ensure the stability of the network despite the route changes induced by user choice. This selection creates a more complex economic environment; it offers potential rewards in user choice and competition, but requires solutions to issues of accounting, pricing, billing, and inter-ISP contracts. Because this architectural change involves the user or some proxy thereof, implementing a more flexible routing architecture involves changes to the entire network—including hosts. Finally, it is necessary to seriously consider the security implications of any proposed architecture (such as source routing) to ensure that they do not create additional vulnerabilities.

It is our hope that an architecture that enabled some level of user control over routes would lead to an increase in the reliability of the Internet, and an improvement in the sets of features offered to users. First, today's Internet often lacks the necessary reliability to serve as a basis for emergency services, real-time control, or for particularly time-sensitive transactions. Systems based upon user control of routing may be able to increase this availability sufficiently for the Internet to encompass a wider variety of critical services. Second, in today's Internet environment, it can be difficult to determine what party is responsible for poor (or superior) performance, and to reward them by choosing to use them for your Internet service. This inability keeps from the Internet many of the benefits of increased competition—lower costs, more efficient practices, and a rich set of services that differentiate providers from one another. User control of routes could help move the Internet in this positive direction.

3.6. Control and Management

The original Internet architecture focuses on best-effort reachability among cooperative users, which results in a primitive control/management infrastructure that bundles the reachability logic and data fowarding functions in each individual router. Today's networks, owned by competing entities and operated in different environments (data center, access/metro, enterprise, ISP) are called upon to meet far more sophisticated network wide-objectives: dependability, policy, traffic engineering, security, ease of management, cost-effectiveness, and so on. As new network-wide objectives need to be accomplished, the original box-centric control architecture (tightly coupled decision making logic and data plane in one box) forces point solutions to be invented, and then retro-fitted onto the network. This has resulted in significant complexity, with diverse and local decision logic distributed across multiple network elements. This is a fundamental reason for the fragility of the Internet, where a single local event can cause a network-wide meltdown. In short, operational complexity plagues today's Internet.

The trajectory of existing incremental efforts is to incorporate more point solutions into the control plane, which only exacerbates the problem of management complexity. If network management is re-architected to explicitly consider multiple network-wide objectives, there is the potential to reduce the fragility of today's networks and lower the complexity of the network. In general, such a change would enable rapid innovations management functions by explicitly separating the implementation of control logic from the routers that implement data

plane functions. The research community is actively pursuing this agenda [HJ00, TU01, KO00, HA02, DO02, YA04, FA05, HS03].

One of the key barriers to progress is the relative opaqueness of the network, meaning that components do not support communication of operationally relevant information to each other. Such information could be aggregated and analyzed [CL03], thereby facilitating load balancing, fault diagnosis, anomaly detection, application optimization, and other traffic engineering and network management functions [SA99, CA00, AR92, RE99, CR03, HO93, PO97, FE00a, FE00b, LA04, SH99, FE02a, FE02b]. This lack of transparency, even for components within the same administrative domain, is framed by not only technical reasons (e.g., a router can only export information about its best known routes, not all known routes, rendering it impossible to realistically simulate what-if scenarios) but also by competitive business realities (operators have a disincentive to reveal operational details about their infrastructure).

The opaqueness of the routing system merits particular consideration. It is impossible to realistically model routing behavior more than 1-hop away from a given node since the policy-rich features added to BGP (e.g., MEDs) have further removed what little transparency originally existed, and thus fatally hindered the ability to logically reason about the routing system. More fundamentally, the current market structure of the Internet promotes information hiding, and when those building and maintaining infrastructure consider opaqueness a feature rather than a limitation, an architectural position that favors transparency also needs to consider how to enforce that transparency on a market that will hide whatever it chooses, even at the expense of operational efficiency.

3.7. Meeting Application Requirements

One architectural decision of the original Internet stands out as playing a critical role in its success: the adoption of a *narrow-waisted hourglass model*. A minimal and carefully chosen set of global capabilities at the mid-level of the architecture allows both higher-level applications and lower-level communication technologies to coexist, share capabilities, and evolve rapidly. The narrow-waisted model is critical to the Internet's ability to adapt rapidly to new user demands and changing technologies.

However, new application classes place demands on core IP capabilities that many argue cannot be met within the current model. Moreover, the growing scale and increasing heterogeneity of the Internet increases the perceived value of placing functionality *within the network*, to better take advantage of localized knowledge and optimization opportunities.

There are several technical responses to these developments. One is to *widen the waist of the hourglass*, to augment the Internet's core forwarding service to include additional functionality. Proposals to satisfy new application requirements by adding new capabilities—e.g., QoS control, multicast, anycast, policy-based routing, data caching, and so on [PO81, SH97, BR97, BE00, QU01, HI03, ST93, AL99, BY98, AKA, DIG]—to the core IP protocols have dominated the last ten years of Internet networking research. Some of these have been deployed in specific circumstances; some have failed to be deployed at all. Whether or not we can deploy any specific enhancement, there is a risk to the stability and coherence of the Internet architecture if we keep adding function to the basic forwarding layer. It is a widely held belief that flexibility, deployability, and evolvability are achieved only when the truly *universal* portions of the architecture are also truly *minimal*.

A second response is to *add a layer* to the architecture, inserting purpose-tuned overlay networks between the global infrastructure and the ultimate end nodes. This strategy offers many potential advantages. Overlays constructed with application-level requirements in mind can make network-level decisions, such as routing, service model, and data manipulation, tuned to the specific application. Overlays designed to support small-scale applications can utilize algorithms that would not scale to global size. Moving functionality from shared infrastructure to multiple overlays increases decentralization by more cleanly modularizing responsibilities and administrative operations. This potential has created great interest in the overlay model, with vigorous activity in the academic and commercial communities [AN01, SU02, BA02, RA02, RO01a, RO01b, ST01, CH02, SP03, DA01, KU00, PA04, ZH04].

As important and valuable as this work is, however, virtually all of this activity has focused on understanding and increasing the functionality of specific overlay networks and algorithms, leaving unanswered the single most critical question relevant to this approach: what is it that lies underneath? What is the appropriate narrow, universally shared environment to support the overlays?

This environment, which we term the *overlay substrate,* must play three critical roles. First, it must *support* the different overlay structures and services that it underpins, in the same way that today's Internet supports end-to-end applications. This implies that the underlay must expose information about the underlying physical network that overlays need to do their job. Second, it must *protect* both overlays and underlying resources from damaging interactions, instabilities, and behaviors. Finally, it must support a *level, open playing field*, allowing technologies, services and participants to come and go while maintaining the basic integrity of the system.

A third approach is to move the narrow waist to a lower level of the protocol stack, that is, define a *network substrate* consisting of a collection of physical resources (nodes and links) on top of which multiple, alternative network architectures could co-exist. Fundamentally, this implies that virtualization would become a first-class feature of the network architecture, allowing for on-going diversity and renewal at the network layer. In this world, IP would become just one of potentially many network architectures. Others might provide alternative security or robustness properties, or simply be tailored for certain classes of applications.

A diversified network could create a range of new opportunities for current stakeholders. It would allow providers of the physical infrastructure to focus on virtual networks as their primary "customers", allowing them to distinguish themselves through the quality of their infrastructure and the support services they provide to virtual networks. Equipment vendors would have the opportunity to create new classes of equipment and provide design services to virtual network providers. Shifting the provision of end-to-end services to virtual networks would create a whole new class of business opportunities, potentially sparking a wave of entrepreneurial innovation.

Before the diversified Internet concept can be put into practice, there is a range of open issues that will need to be explored through on-going research efforts. These include: (1) defining the nature of the resource provisioning interface between substrate providers and virtual networks; (2) developing mechanisms that enable virtual networks to easily use resources provided by the substrate to implement innovative new services; (3) design systems that allow virtual networks to co-exist on a common substrate without interfering with one another, while still allowing them to interact where such interaction is desired; (4) extending

host operating systems to allow users to conveniently use the services of multiple virtual networks; and (5) develop strategies for implementing access links that would allow the access link resources to be flexibly re-allocated among multiple virtual networks, while still allowing for predictable performance.

Although these last two approaches address the problem at different layers, they both focus on designing a suitable substrate on top of which multiple network architectures and services can run. In both cases, the goal is to identify the critical balance of functionality, minimality, stability, evolvability, and deployability that will allow a shared virtualized infrastructure to spread globally, while supporting a rich and changing architectural ecosystem. Moreover, understanding which layer is the most appropriate "new waist of the hourglass" is one of the most interesting questions facing the research community.

4. Experimental Deployment of Architectural Innovations

To be effective, a research program that seeks to promote architectural innovation must enable researchers to create, deploy, and evaluate novel architectures. These architectures must both run at scale, and carry traffic from real users. This calls for the creation of a testbed of global reach and diverse capabilities. This section outlines the goals and design principles that shape this testbed.

4.1. Goals and Scope

The testbed must provide an environment in which multiple new network architectures and services can be deployed. This means there should be as few restrictions as possible on the architectures that operate on the testbed and on the capabilities provided by the testbed. Toward this end, the testbed should include a diversity of links and nodes (both physical and virtual), and permit connection of arbitrary edge devices.

The testbed should be capable of bridging the gap between so-called production testbeds, which constrain research, and research testbeds, which constrain users [KU02]. It must be capable of attracting and supporting users of its services beyond the research community. This is essential for allowing new architectural innovations to be evaluated at scale, and for creating a population of users whose demonstrated interest in a new capability can stimulate technology transfer to the commercial Internet.

To meet these goals, the testbed need not have a single architecture in the traditional sense. Instead, its role is to provide an environment in which a diverse set of experimental networks—each with its own distinct architecture—can operate. In this sense, the testbed is really a *meta-testbed* that hosts a heterogeneous collection of testbeds within it. Each of these individual testbeds is allocated a portion of the meta-testbed's resources. The meta-testbed should constrain the hosted activities hosted to the minimum extent possible, and provide for varying degrees of isolation and interconnection among these activities. The common part of the meta-testbed, which we refer to as the *substrate*, provides the mechanisms for allocating and configuring resources and ensuring the necessary isolation.

The meta-testbed should be viewed as a dynamic artifact: the physical resources, management capabilities, governance processes, implementation, and even the substrate design will evolve with time. The physical resources in the testbed may include a mix of dedicated physical links and nodes, virtual components contributed on a permanent or temporary basis

by testbed users, and resources leased from third-party providers or consortia such as NLR [NLR]. The substrate and management infrastructure should incorporate standard service policies and interfaces to enable organic growth, provide incentives to contribute, and manage dynamic resources available to the meta-testbed on a temporary basis under various terms.

4.2. Key Concepts

There are several key architectural concepts that we expect to play a central role in the design of the meta-testbed. The meta-testbed will consist of links, nodes and edge devices. Links may be implemented in a variety of ways, including direct physical links, MPLS paths, and IP tunnels. The meta-testbed links can be shared by different experimental networks running within the meta-testbed, using well-known virtual link multiplexing techniques.

The meta-testbed nodes provide a collection of memory, processing, and storage resources. They might correspond to virtual machines running on commodity processors; dedicated general-purpose processors; both dedicated and virtualized network processors and programmable hardware; and virtualized routers. The meta-testbed provides mechanisms to configure these resources for use by different experimental networks and to provide isolation between experimental networks.

Edge devices (including traditional hosts) may participate in multiple networks running within the meta-testbed. In some cases, this will require that edge devices implement separate protocol stacks, although a key to the success of the meta-testbed will be the development of mechanisms that make it easy for users to "opt-in" to experimental networks that offer some value-added capability.

Each experimental network will run on some subset of the meta-testbed resources. We call the substrate resources bound to a particular experimental network a *slice*, borrowing the term from PlanetLab [PET02, BA04]. Each slice will include some number of nodes (including both physical processors and virtual machines multiplexed shared hardware) connected by links (including both physical links and virtual links). The main responsibility of the meta-testbed management software will be to provide mechanisms that can be used to allocate resources to slices, and ensure that slices do not interfere with each other.

We note that different users of the meta-testbed will require varying degrees of isolation, connectivity, dynamism, and control in their slices. Slices that require full isolation from other slices (including traffic and performance isolation) should have a means to acquire it, subject to the availability of the required resources. At the same time, it should be possible to connect different slices to one another, where that is appropriate and mutually agreed upon. While it is likely that the meta-testbed will initially incorporate a narrow range of resources and simple assignment policies, this range should advance over time.

4.3. Design Principles

The design and development of the meta-testbed will require decisions on a wide range of issues. The workshop did not provide sufficient time to fully explore these issues, but there was substantial agreement on some core design principles, which participants felt should guide the design process and the subsequent operation of the meta-testbed. These are summarized below.

- ***Service/architecture neutrality***. What is most important for research in network architecture is that the level of abstraction be low enough to permit full experimentation at layer 3 and above. Different slices of the common testbed may reflect different layer 3 architectures at the same time. In particular, networks running in different slices may use different packet formats and service models.
- ***End-system diversity***. The meta-testbed should enable heterogeneity in the end systems that connect to it and participate in the experimental networks running within it. In particular, it should enable the connection of limited functionality end-systems (such as wireless PDAs and sensor motes).
- ***Ease of user access***. Mechanisms are needed to make it easy for users to join one or more experimental networks running in the meta-testbed, and to transparently fall back to the standard Internet whenever the experimental network cannot provide the requested service. In some cases, this can be accomplished using transparent re-direction mechanisms [KA04]. In other cases, it may require the installation of new protocol stacks in hosts.
- ***Sustainability and incentives***. To ensure the sustainability of the meta-testbed, it should be possible for participating institutions to join by contributing resources in return for access to the resources of the meta-testbed as a whole.
- ***Inter-slice composition***. The testbed infrastructure must enable interconnection among slices by mutual consent, and between slices and the external Internet. This permits slices to host network services with external users, and/or to act as transit networks. Nothing should prevent a researcher from inter-connecting a network running within a slice with another network. This other network could be running within another slice of the meta-testbed, or it could be the commodity Internet or another custom network (or testbed) that runs over standard IP protocols.
- ***Policy and governance***. Since the meta-testbed will comprise shared infrastructure, there must be a governance process to guide allocation of resources to slices, and a software architecture that implements and enforces the policies. Some slices will likely require strong performance isolation, which will make some form of admission control necessary.

There were additional issues raised for which there was not a broad consensus. While these issues should be explored further, the workshop participants felt that design and development of the meta-testbed should not be delayed until these issues can be fully resolved.

First, there was a discussion about how much effort should be focused on performance, at least when considered relative to the need to design and evaluate new functionality. For example, there is an opportunity to incorporate high performance backbone links into the meta-testbed, using fiber optic facilities available through the National Lambda Rail [NLR]. NLR links can operate at 10 Gbps, allowing the meta-testbed to carry large traffic volumes and making it possible to evaluate experimental networks operating at high speeds. For many network research purposes however, this capability is not strictly necessary, and fully exploiting this capability will require the development of high performance testbed nodes. It is difficult to know if the benefits provided by such high performance nodes would justify the cost of their development; it may be possible to accomplish most of the objectives using clusters of general-purpose processors connected by COTS switches. The question, then, is one of priorities: should sufficient funding be available, work on functionality and performance should

proceed in parallel; if not, designing new functionality that address the many challenges facing the Internet is the highest priority.

Second, while the general consensus among workshop participants was that the meta-testbed must use packet transport—thereby allowing existing network access mechanisms (primarily Ethernet) to be used to connect end users to the meta-testbed—there was an acknowledgment that the broader network community might not agree, favoring instead a circuit-based approach. However, the workshop participants believe that experimental networks running within the meta-testbed could offer circuit-like services within their own slice, for example, by using virtual links with reserved bandwidth and implementing per virtual link smoothing buffers to convert packet links with a small amount of jitter into constant delay links. Such an approach does not allow for high performance circuit switched elements (such as optical cross-connects) to be incorporated into the meta-testbed, but there was no clear suggestion for how such elements could be included, given the likely resource constraints on the meta-testbed as a whole.

4.4. Departure Point

The meta-testbed we envision is ambitious, but the networking community has a strong track record of creating testbeds and testbed technologies, and using them to evaluate and demonstrate new research ideas. Some examples of current efforts that provide subsets of the capabilities needed for the proposed meta-testbed required include:

- PlanetLab [PET02, BA04], which focuses on node virtualization and global resource management, and is widely used for research in network services.
- X-bone [TO01, TO03] and 6-bone [SXB], which define core (L3) capabilities for network virtualization and are supported by multiple operating systems;
- Emulab [EMU,WH02] and Netbed, which allocate and configure heterogeneous end-system resources and network resources (using L2 virtualization) together;

The scale and presence of these testbeds have proven to be significant enablers of new research, with strong momentum and community involvement. They serve as incubators and proofs of concept for many of the architectural ideas outlined earlier in this report, as well as for the proposed meta-testbed itself. What is needed now, however, is a more comprehensive effort that incorporates a broad range of resources and capabilities.

PlanetLab, in particular, offers a starting point that can be leveraged immediately. It provides a shared overlay infrastructure that spans over 525 nodes distributed across over 250 sites and 28 countries. It currently hosts over 350 slices—each running a different network architecture, service, or application—on the shared infrastructure. PlanetLab also includes software that allows end users to seamlessly connect their desktop machines to services they want to employ, resulting in network traffic to over 500k unique IP addresses every day.

However, this infrastructure is not sufficient by itself. It needs to be enhanced with a richer set of link technologies (e.g., by adding MPLS paths and dedicated circuits to the currently supported IP tunnels) and a more diverse set of node configurations (e.g., by adding dedicated processors and customizable hardware to the currently supported commodity processors). Over time, the resulting infrastructure will become the meta-testbed that meets our objectives. Much work remains to be done to realize the full scope of the meta-testbed, but the end-goal is clear: a meta-testbed that combines the *global reach* of overlays with the *performance realism* of physical

links and programmable routers. In making progress toward this vision, there must be an open and inclusive community process for designing the mechanisms of the meta-testbed, and this process must carefully balance competing demands and the needs of the community.

5. Recommendations

This report has explained the need for dramatic improvements in the security, robustness, manageability, flexibility, and cost-performance of the Internet as a critical piece of our societal infrastructure, and the inability of evolutionary research to effect the needed architectural improvements. Thus, we argue that the National Science Foundation should take a new approach to fostering disruptive innovation in networking. With the right support from and partnership with NSF, the network research community is poised to define the next generation of Internet technology, fundamentally and permanently addressing the problems the Internet has today. Further, a revised architecture has the potential to unleash a new class of applications, currently stalled behind the limited functionality of today's best effort Internet service. A new architecture could also better leverage technology trends towards incredibly high bandwidth optical networks and increasingly capable computation devices embedded in the network. Radically new networks such as ad hoc wireless networks and sensor networks would also be enabled.

In sum, if successful, this effort will directly benefit virtually every member of our society, enhancing homeland defense, ensuring that communication over the Internet is as reliable and secure as physically possible, delivering the raw performance of the network to demanding scientific and engineering applications, reducing the cost of Internet access for all users, and enabling the next generation of innovation in our shared network infrastructure.

However, achieving this vision will not be easy. Any replacement for the Internet's architecture must demonstrate its value via widespread use, or this effort will be pointless. This will require both NSF and the network research community to change business as usual, focusing on the construction of practical and usable systems, in addition to cutting edge research. This means that NSF must be willing to put forward sufficient funds, sustained over an extended period of time, for the community to build and operate an alternative architecture in live use by large numbers of users. And the research community must be willing to put the effort into moving their ideas into practice, rather than being satisfied with paper designs. Although progress can be made immediately, and funding ramped up in response to success in meeting milestones, make no mistake: the total scope of the effort needed is much larger than can be supported under NSF's current networking research budget. The corresponding benefit will be enormous—nothing less than putting the world's communication infrastructure on a secure, robust, flexible, and efficient basis for the foreseeable future.

Specifically, the workshop participants make the following recommendations to the National Science Foundation:

Recommendation 1: Immediately initiate a research program on experimental architectural research in networking.

Given the current architectural limitations, and the encouraging prospects for overcoming them, NSF should provide significant multi-year funding for architectural research. If successful, the potential benefits are enormous, easily justifying the modest initial outlay. The

amount of research funding may need to ramp in future years; as we gain experience with using these new architectures, additional opportunities will open up to leverage that experience. A dedicated research program will pull together the research community to tackle the broad scope of thorny architectural questions that we have outlined in this report, and that must be addressed for the program to be successful.

Recommendation 2: Foster experimental validation of new architectural research in networking.

Paper designs, although thought provoking, are unconvincing, both to the companies that need to adopt them, and to the research community in evaluating ideas and in gaining insight into design tradeoffs. Thus, to maximize our chance of success, NSF must foster an expectation within the experimental architectural research program that research ideas should normally be validated under real use. Not every research idea will be worth validating, but every successful research idea must be validated before it can be considered to have proven its worth. The burden this places on the research community should not be underestimated – working systems are much more expensive to build than paper designs, and useful systems are even more expensive. In many cases, this will involve deployment on an appropriate testbed and usage by the population-at-large. Although we should leave the specifics of how to accomplish experimental validation to the creativity of researchers and the intelligence of peer review, this is likely to mean that larger, multi-principal investigator and center-scale efforts will be needed to make progress along the most promising architectural directions.

Recommendation 3: Fund the development and deployment of suitable testbeds.

Since experimental validation is an important component of this research program, it is essential that researchers have access to suitable testbeds. The alternative, requiring each new effort to create their own testbed, is both impractical and inefficient. Instead, NSF should endeavor to build a meta-testbed that reduces the barrier to entry for new architectural ideas. To meet short-term needs, NSF should support an initial meta-testbed that can be deployed immediately. At the same time, NSF should initiate a deliberative process through which the community can identify long-term solutions to its meta-testbed requirements. NSF should support the deployment and on-going operation of the resulting meta-testbed, but this funding should not decrease that devoted to architectural research itself.

Recommendation 4: Start a process that will lead to substantial increases in funding for a broad multi-disciplinary effort in this area over the next few years.

To design, construct and widely deploy a new architecture for the Internet is an enormously difficult and, at the same time, an enormously important undertaking. It cannot be done on the cheap. Nor can it be done by the networking research community alone. To be successful, we will need to enlist the efforts of distributed systems researchers, e-scientists, application developers, computer architects, and network hardware technologists. Gaining consensus among those communities of the need and value of our efforts will be a difficult, but necessary step towards our eventual success. Equally important will be the ability to put sufficient funds behind that emerging consensus, to be able to demonstrate the value of our work in widespread practice.

Recommendation 5: Find ways to promote synergy and convergence among architectural visions.

Academic research focuses on novelty and, in so doing, often accentuates differences rather than identifying commonality. The past success of the Internet strongly suggests that we will be the most successful if we can coalesce around common architecture features. Architecture, by its very nature, "defines that on which we must agree." Thus, to be effective, architectural researchers should seek convergence rather than divergence. NSF should find ways to promote the necessary community practices, such as encouraging participation in working groups and funding synthetic research.

Recommendation 6: Help the community learn from industry.

Disruptive architectural research should not be fettered by today's problems and practices, but it must be informed by them if we are not to simply repeat the mistakes of the past. The large gap between the research and commercial communities often prevents effective communication between the two, to the detriment of both. To bridge this gap, NSF should facilitate interactions between researchers and practitioners. This extends not only to the operational community of ISPs, but also to network application developers and those developing new physical layer technologies, as the success of any new architecture will be measured by how well it matches the requirements of its hardware and the needs of its users.

References

[AL99] C. Alaettinoglu, et. al., "Routing Policy Specification Language (RPSL)," RFC 2622, June 1999.

[AKA] Akamai, www.akamai.com

[AN01] D. Andersen, H. Balakrishnan, F. Kaashoek, R. Morris, "Resilient Overlay Networks," *Proceedings of ACM SOSP*, 2001.

[AR92] R. Arun, P. Venkataram, "Knowledge Based Trouble Shooting in Communication Networks," *Proceedings of Symposium on Intelligent Systems*, November 1992.

[BE02] S. Berson, S. Dawson and R. Braden. "Evolution of an Active Networks Testbed," *Proceedings of the DARPA Active Networks Conference and Exposition*, 5/02.

[BA02] M. Balazinska, H. Balakrishnan, D. Karger, "INS/Twine: A Scalable Peer-to-Peer Architecture for Intentional Resource Discovery," *Proceedings of the 1st International Conference on Pervasive Computing*, 2002.

[BA04] A. Bavier, M. Bowman, B. Chun, D. Culler, S. Karlin, L. Peterson, T. Roscoe, T. Spalink, and M. Wawrzoniak. *Proceedings of the 1st Symposium on Network System Design and Implementation (NSDI '04)* San Francisco, CA (March 2004).

[BE00] Y. Bernet, et. al., "A Framework for Integrated Services Operation over DiffServ Networks," RFC 2998, November 2000.

[BR97] B. Braden, et. al., "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification," RFC 2205, Sepetember 1997.

[BY98] J. Byers, M. Luby, M. Mitzenmacher, A. Rege, "A Digital Fountain Approach to Reliable Distribution of Bulk Data," *Proceedings of ACM Sigcomm*, 1998.

[CA00] Z. Cao, Z. Wang, E. Zergura, "Performance of Hashing-Based Schemes for Internet Load Balancing," *Proceedings of IEEE Infocom*, 2000.

[CH02] B. Chun, J. Lee, H. Weatherspoon, "Netbait: a Distributed Worm Detection Service," Project website, http://netbait.planet-lab.org, 2002.

[CL03] D. Clark, C. Partridge, J. Ramming, J. Wroclawski, "A Knowledge Plane for the Internet," *Proceedings of ACM Sigcomm*, 2003.

[CL05] D. Clark, et. al. Making the world (of communication) a different place. January 2005. http://www.ir.bbn.com/~craig/e2e-vision.pdf

[CR03] M. Crovella, E. Kolaczyk, "Graph Wavelets for Spatial Traffic Analysis," *Proceedings of IEEE Infocom*, 2003.

[DA01] F. Dabek, M. Kaashoek, D. Karger, R. Morris, I. Stoica, "Wide-area cooperative storage with CFS," *Proceedings of ACM SOSP*, 2001.

[DET] DETER project web site. http://www.isi.edu/deter/

[DIG] Digital Fountain, www.digitalfountain.com

[DO02] A. Doria, F. Hellstrand, K. Sundell, T. Worster, "General Switch Management Protocol (GSMP) V3," RFC 3292, June 2002.

[EMU] Emulab project web site. http://www.emulab.net

[FA05] A. Farrel, J. Vasseur, J. Ash, "Path Computation Element (PCE) Architecture," Internet-Draft, March 2005.

[FE98] P. Ferguson, D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," RFC 2267.

[FE00a] A. Feldmann, A. Greenburg, C. Lund, N. Reingold, J. Rexford, F. True, "NetScope: Traffic Engineering for IP Networks," *IEEE Network*, March/April 2000.

[FE00b] A. Feldmann, A. Greenburg, C. Lund, N. Reingold, J. Rexford, "Deriving Traffic Demands for Operational IP Networks: Methodology and Experience," *Proceedings of ACM Sigcomm*, 2000.

[FE02a] N. Feamster, J. Borkenhagen, J. Rexford, "Controlling the Impact of BGP Policy Changes on IP Traffic," *Proceedings of NANOG25*, 2002.

[FE02b] N. Feamster, J. Rexford, "Network-wide BGP Route Prediction for Traffic Engineering," *Proceedings of ITCOM*, 2002.

[HA02] M. Handley, O. Hodson, E. Kohler, "XORP: An Open Platform for Network Research," *Proceedings of ACM HotNets-I*, October 2002.

[HI03] R. Hinden, S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture," RFC 3513, April 2003.

[HJ00] G. Hjalmtysson, "The Pronto Platform – A Flexible Toolkit for Programming Networks using a Commodity Operating System," *Proceedings of OpenArch*, 2000.

[HO93] J. Hochberg, et. al., "Nadir: An automated system for detecting network intrusion and misuse," *Computers & Security*, 12(3), 1993.

[HS03] H. Hsieh, et. al., "A Receiver-Centric Transport Protocol for Mobile Hosts with Heterogeneous Wireless Interfaces," *Proceedings of Mobicom*, 2003.

[I2] Internet2 project web site. http://www.internet2.edu

[I3] Internet Indirection Infrastructure project web site. http://i3.cs.berkeley.edu

[KU02] J. Kurose (editor). Report of the NSF Worksop on Network Research Testbeds. www.gaia.cs.umass.edu/testbed_workshop, 11/02.

[KA04] J. Kannan, A. Kubota, K. Lakshminarayanan, I. Stoica and K. Wehrle, "Supporting Legacy Applications over *i3*," UCB Technical Report No. UCB/CSD-04-1342, May 2004.

[KO00] E. Kohler, "The Click Modular Router," Ph.D. thesis, MIT, November 2000.

[KU00] J. Kubiatowicz, et. al., "Oceanstore: An Architecture for Global-Scale Persistent Storage," *Proceedings of ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2000.

[LA04] A. Lakhina, M. Crovella, C. Diot, "Diagnosing Network-Wide Traffic Anomalies," *Proceedings of ACM Sigcomm*, 2004.

[LBO] L-bone. www.loci.cs.utk.edu/.

[MO05] R. Moskowitz, P. Nikander, P. Jokela, T. Henderson, "Host Identity Protocol," Internet Draft, Feb. 2005.

[NLR] National Light Rail Project web site. www.nlr.net.

[PER02] C. Perkins, Ed. "IP Mobility Support for IPv4," RFC 3344.

[PET02] L. Peterson, T. Anderson, D. Culler and T. Roscoe. "A Blueprint for Introducing Disruptive Technology into the Internet," *Proceedings of ACM HotNets-I Workshop* (October 2002).

[PET04] L. Peterson, S. Shenker, and J. Turner. "Overcoming the Impasse Through Virtualization," *Proceedings of ACM Hotnets-III* (November 2004).

[PA04] K. Park, V. Pai, L. Peterson, Z. Wang, "CoDNS: Improving DNS Performance and Reliability via Cooperative Lookups," *Proceedings of Symposium on Operating Systems Design and Implementation (OSDI)*, 2004.

[PO81] J. Postel, ed., "Internet Protocol," RFC 791, September 1981.

[PO97] P. Porras, P. Neumann, "EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances," *Proceedings of NIST-NCSC National Information Systems Security Conference*, 1997.

[PRE05] President's Information Technology Advisory Committee. Cyber Security: A Crisis of Prioritization, February 2005. [QU01] B. Quinn, K. Almeroth, "IP Multicast Applications: Challenges and Solutions," RFC 3170, September 2001.

[RA02] S. Ratnasamy, M. Handley, R. Karp, S. Shenker, "Topologically-Aware Overlay Construction and Server Selection," *Proceedings of IEEE Infocom*, 2002.

[RE99] R. Reddy, D. Estrin, R. Govindan, "Large-Scale Fault Isolation," *IEEE Journal on Selected Areas in Communications*, March 1999.

[RO01a] A. Rowstron, P. Druschel, "Storage Management and Caching in PAST, A Large-Scale Persistent Peer-to-Peer Storage Utility," *Proceedings of ACM SOSP*, 2001.

[RO01b] A. Rowstron, P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems," *Proceedings of IFIP/ACM International Conference on Distributed Systems Platforms*, 2001.

[SA99] S. Savage, A. Collins, E. Hoffman, J. Snell, T. Anderson, "The End-to-End Effects of Internet Path Selection," *Proceedings of ACM Sigcomm*, 1999.

[SH97] S. Shenker, C. Partridge, R. Guerin, "Specification of Guaranteed Quality of Service," RFC 2212, September 1997.

[SH99] A. Shaikh, J. Rexford, K. Shin, "Load-Sensitive Routing of Long-Lived IP Flows," *Proceedings of ACM Sigcomm*, 1999.

[SP03] N. Spring, D. Wetherall, T. Anderson, "Scriptroute: A facility for distrubted Internet measurement," *Proceedings of USENIX Symposium on Internet Technologies*, 2003.

[SXB] 6 Bone. http://www.6bone.net/.

[ST93] R. Steenstrup, "Inter-Domain Policy Routing Protocol Specification: Version 1," RFC 1479, July 1993.

[ST01] I. Stoica, R. Morris, D. Karger, F. Kaashoek, H. Balakrishnan, "Chord: A Peer-to-Peer Lookup Service for Internet Applications," *Proceedings of ACM Sigcomm*, 2001.

[ST02] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, S. Surana, "Internet Indirection Infrastructure," *Proceedings of ACM SIGCOMM*, August, 2002.

[SU02] L. Subramanian, I. Stoica, H. Balakrishnan, R. Katz, "OverQoS: Offering Internet QoS Using Overlays," *Proceedings of ACM HotNets*, 2002.

[TO01] J. Touch. "Dynamic Internet Overlay Deployment and Management Using the X-Bone," *Computer Networks*, July 2001, pp. 117-135.

[TO03] J. Touch, Y. Wang, L. Eggert, G. Finn. "Virtual Internet Architecture," ISI Technical Report ISI-TR-2003-570, March 2003.

[TU01] P. Tullmann, M. Hibler, J. Lepreau, "Janos: A Java-oriented OS for Active Networks," *IEEE Journal on Selected Areas in Communications*. Volume 19, Number 3, March 2001.

[WH02] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb and A. Joglekar. "An Integrated Experimental Environment for Distributed Systems and Networks," *Proceedings of OSDI*, 12/02.

[YA04] L. Yang, R. Dantu, T. Anderson, R. Gopal, "Forwarding and Control Element Separation (ForCES) Framework," RFC 3746, April 2004.

[ZH04] M. Zhang, C. Zhang, V. Pai, L. Peterson, R. Wang, "PlanetSeer: Internet Path Failure Monitoring and Characterization in Wide-Area Services," *Proceedings of Symposium on Operating Systems Design and Implementation (OSDI)*, 2004.

List of Workshop Participants

David Andersen	Massachusetts Institute of Technology	dga@csail.mit.edu
Tom Anderson	University of Washington	tom@cs.washington.edu
Andy Bavier	Princeton University	acb@cs.princeton.edu
Jeff Chase	Duke University	chase@cs.duke.edu
K. C. Claffee	Coop. Assoc. for Internet Data Analysis	kc@caida.org
Patrick Crowley	Washington University	pcrowley@cse.wustl.edu
Mike Dahlin	University of Texas	dahlin@cs.texas.edu
Dave Clark	Massachusetts Institute of Technology	ddc@csail.mit.edu
Constantine Dovrolis	Georgia Institute of Technology	dovrolis@cc.gatech.edu
Joe Evans	National Science Foundation	jbevans@nsf.gov
Darleen Fisher	National Science Foundation	dlfisher@nsf.gov
Paul Francis	Intl. Computer Science Institute	francis@aciri.org
Sergey Gorinsky	Washington University	gorinsky@cse.wustl.edu
Roch Guerin	University of Pennsylvania	guerin@ee.upenn.edu
T. V. Lakshman	Bell Laboratories	lakshman@dnrc.bell-labs.com
John Lockwood	Washington University	lockwood@cse.wustl.edu
Guru Parulkar	National Science Foundation	gparulka@nsf.gov
Adrian Perrig	Carnegie-Mellon University	perrig@cmu.edu
Larry Peterson	Princeton University	llp@cs.princeton.edu
Mothy Roscoe	Intel Research	troscoe@intel-research.net
Srini Seshan	Carnegie-Mellon University	srini@cmu.edu
Scott Shenker	Intl. Computer Science Institute	shenker@icsi.berkeley.edu
Alex Snoeren	University of California, San Diego	snoeren@cs.ucsd.edu
David Taylor	Washington University	det3@arl.wustl.edu
Jonathan Turner	Washington University	jon.turner@wustl.edu
Joe Touch	Information Sciences Institute	touch@isi.edu
Arun Venkataramani	University of Texas	arun@cs.utexas.edu
Xiaowei Yang	Massachusetts Institute of Technology	yxw@mit.edu
Raj Yavatkar	Intel Architecture Labs	raj.yavatkar@intel.com
Hui Zhang	Carnegie-Mellon University	hzhang@cs.cmu.edu

The above participants attended the workshop on January 13-14, 2005, and contributed to the writing of this report. The workshop was organized by Tom Anderson (University of Washington), Larry Peterson (Princeton University), Scott Shenker (University of California, Berkeley), and Jonathan Turner (Washington University).